

Aktuelle Implementierungen des PEARL Systems von Werum

H. Windauer, Werum GmbH

Zusammenfassung

Dieser Beitrag beschreibt die wesentlichen Eigenschaften und den Stand der aktuellen Implementierungen des portablen PEARL-Systems von WERUM für die Rechenanlagen COBRA 530, DEC LSI 11, DEC VAX 11, NOVADATA, PCS QU 68000, Siemens 7.xxx (BS 2000), SISCO MB 8000 und Zilog Z 8000.

Schlüsselwörter: PEARL Compiler, Testsystem, Datenbank-system

Summary

This paper describes the essential properties and the state of the actual implementation of the portable PEARL system of WERUM for the computers COBRA 530, DEC LSI 11, DEC VAX 11, NOVADATA, PCS QU 68000, Siemens 7.xxx (BS 2000), SISCO MB 8000 and Zilog Z 8000.

Key words: PEARL compiler, test system, database management system

1. Einleitung

Zum besseren Verständnis der aktuellen Implementierungen werden zunächst die Komponenten des portablen PEARL-Systems vorgestellt, bevor auf die einzelnen Implementierungen eingegangen wird.

2. Das portable PEARL-System von WERUM

Das portable PEARL-Programmiersystem der Firma Werum besitzt folgende Komponenten /10/ :

- Compiler und Ablaufsystem mit dem PEARL-Betriebssystem BAPAS-K
- Testsystem
- Datenbanksystem BAPAS-DB

Es ist bisher in Teilen (z.B. nur der Compiler) oder vollständig auf folgenden Rechenanlagen installiert: HP 1000, HP 3000, Intel 8086 /6/, ND 100, RDC, Siemens 300-R-Serie.

Zu Zwecken der Analyse und Cross-Compilierung ist der Compiler auch auf Gastrechnern folgenden Typs installiert: Amdahl 470/6, IBM 4341, Siemens 7.760.

2.1 PEARL-Compiler und Ablaufsystem

2.1.1 Eigenschaften des Compilers

Der PEARL-Compiler kann PEARL-Programme gemäß DIN 66 253, Teil 1, sowie u.a. die folgenden Sprachelemente aus DIN 66 253, Teil 2, /2/ verarbeiten:

- Benutzerdefinierte Datentypen (TYPE)
- Benutzerdefinierte Operatoren (OPERATOR)
- Referenzen (REF)
- Felder und Strukturen in Strukturen
- Prozeduren in Tasks und Prozeduren

Der genaue Sprachumfang ist in /9/ beschrieben.

Der Compiler verfügt über vielfältige Parameter, deren Handhabung der Bedienungsart angepaßt wird, die auf der jeweiligen Rechenanlage üblich ist. Mittels solcher Compiler-Parameter können u.a. Listings der Quelle und des Übersetzungsergebnisses verlangt werden, wobei im Listing des Übersetzungsergebnisses Hinweise auf die Ursprungszeilen im Quellprogramm eingefügt sind. Außerdem erstellt der Compiler auf Wunsch eine Cross-Referenz-Liste aller benutzten Objekte mit Angaben der Quellzeilen-Nummern für ihre Definition und Benutzung.

Durch Compiler-Parameter kann die Überwachung von Feldgrenzen und Referenzen zur Laufzeit ein- oder ausgeschaltet werden.

Der Compiler analysiert Programme umfangreich und recht genau. Die Fehlermeldungen erfolgen im Klartext mit Angabe der Quellzeilen-Nummer.

Ein Preprozessor gestattet das Einfügen von Programmentexten aus Files (%INCLUDE) und bedingte Compilierung (%IF).

Die Systemteil-Auswertung wird durch eine sogenannte Konfigurationsliste gesteuert, die alle Konfigurationsmöglich-

keiten des Zielrechners beschreibt. Sollen diese Möglichkeiten erweitert werden, z.B. beim Anschluß eines bisher nicht vorgesehenen Gerätes, so kann der Anwender selbst mit einfachen Mitteln diese Konfigurationsliste anpassen und dem Compiler (sogar dynamisch) bekanntmachen.

Der Compiler kann Code und Daten trennen.

2.1.2 Struktur des Compilers

Der PEARL-Compiler besteht aus einem analytischen Teil ("Oberteil"), der ein PEARL-Programm in die rechnerunabhängige Zwischensprache IL1 (Intermediate Language 1) übersetzt und - in der Regel - aus einem codegenerierenden Teil ("Codegenerator"), der das PEARL-Programm aus der IL1-Form in den Zielcode überführt. Der Oberteil ist rechnerunabhängig und deshalb nur einmal programmiert; die rechnerabhängigen Codegeneratoren werden für jeden Typ Rechner, auf dem PEARL-Programme ablaufen sollen, neu erstellt.

Der Compiler-Oberteil, alle anderen o.g. portablen Komponenten und die Codegeneratoren selbst sind automatisch nach IL1 überführbar, können also mittels eines Codegenerators automatisch in den entsprechenden Zielcode übersetzt werden.

2.1.3 Betriebssystem-Kern

Für die PEARL-gerechte Organisation und Ausführung von Tasks, für die Synchronisation und die Prozeß-E/A hat WERUM den rechnerunabhängigen PEARL-Betriebssystem-Kern BAPAS-K (Basis für Prozeßautomationssysteme - Kern) entwickelt, der bereits bei verschiedenen PEARL-Systemen im Einsatz ist.

2.1.4 Laufzeitpakete für binäre und formatierte E/A

Alle Laufzeitroutinen, die für einen PEARL-gerechten Ablauf der PEARL-Anweisungen für binäre E/A (einschließlich File-Organisation) und formatierte E/A erforderlich sind, hat WERUM - soweit möglich - rechnerunabhängig in PEARL programmiert und in den Laufzeitpaketen BAPAS-FILE und BAPAS-FORMEA zusammengefaßt.

2.2 PEARL-Testsystem

Für den interaktiven Test von PEARL-Programmen auf Sprachebene verfügt WERUM über ein portables, in PEARL programmiertes Testsystem /1/.

Dieses Testsystem bietet für den Einzeltest von Programmkomponenten komfortable Display-, Trace- und Unterbrechungsmöglichkeiten (Breakpoints). Darüberhinaus unterstützt das System auch den Integrations- und Gesamttest durch die Möglichkeiten der Diagnose und Veränderung der Zustände von Tasks und Synchronisationsvariablen sowie durch Schnittstellen zur Simulation der Prozeßperipherie und durch Simulation

der Zeitachse.

2.3 PEARL-Datenbanksystem

Das PEARL-Datenbanksystem BAPAS-DB (Basis für Prozeßautomationssysteme - Datenbanksystem) wurde von WERUM mit der Zielsetzung entwickelt, ein offenes Datenbanksystem auf Prozeß- und Minirechnern bereitzustellen, bei dem anwendungsabhängige Komponenten wie Zugriffsverfahren und Ausprägung der Benutzeransprache einfach angepaßt, erweitert oder sogar ausgetauscht werden können, falls die standardmäßig hierfür gebotenen Komponenten speziellen Anwendungsanforderungen nicht genügen. Weitere wesentliche Eigenschaften sind:

- Verschiedenen Dateien können verschiedene Zugriffsverfahren zugeordnet werden.
- PEARL-Tasks können konkurrierend zueinander und konkurrierend zu Dialog-Benutzern auf die Daten zugreifen.
- Höher priorisierte Tasks setzen sich auch im Datenbanksystem durch.

Eine umfassendere Charakterisierung findet sich in /4/ oder /10/.

3. PEARL Engine 68000

Der Einsatz von Mikroprozessoren auf breiter Basis bringt notwendigerweise Programmierengpässe, die nur durch den Einsatz höherer Programmiersprachen und die Verfügbarkeit kostengünstiger Stand-alone-Systeme überwunden werden können. Dies trifft insbesondere für den Bereich der Echtzeit-Anwendungen zu. Wie die Beispiele Energieverteilung und Fertigungstechnik zeigen, sind in diesem Bereich außerdem zunehmend Probleme der Datenhaltung zu bewältigen. Deshalb entwickeln die Firmen IRT GmbH, PCS GmbH und WERUM GmbH ein vollständiges PEARL-Programmiersystem auf einem leistungsfähigen, zukunftsorientierten Mikrocomputersystem - die PEARL Engine 68000.

Das Mikrocomputersystem besteht aus dem System PCS QU 68000 (CADMUS 9.200) mit dem Betriebssystem UNIX, das um Mehrprozessorfähigkeit und numerischen Koprozessor vervollständigt wird /3/. Das PEARL-Programmiersystem setzt sich aus den Komponenten Compiler, Ablaufsystem, Testsystem und Datenbanksystem zusammen /8/.

Besonderes Gewicht wird dabei auf die Portierbarkeit des Programmiersystems gelegt, um es mit geringem Aufwand an viele und insbesondere zukünftige 16/32-bit-Mikroprozessortypen anpassen zu können.

Da sich die Systemprogrammiersprache C /7/ in Verbindung

mit dem Betriebssystem UNIX weltweit durchgesetzt hat, liegt es nahe, anstelle eines nur für Motorola 68000 verwendbaren Codegenerators IL1→Bindercode 68000 einen Umsetzer IL1→C zu entwickeln und damit das gesamte PEARL-Programmiersystem in C bereitstellen zu können, wodurch seine Portabilität wesentlich erhöht wird.

Die Übersetzung der PEARL-Programme und die Ein/Ausgabe mit Standardperipherie-Geräten erfolgt unter Regie des Teilnehmer-Betriebssystems UNIX, während die Ausführung der PEARL-Programme auf demselben oder einem benachbarten Prozessor unter Regie des PEARL-Betriebssystems BAPAS-K geschieht.

Die Realisierung der PEARL Engine 68000 erfolgt in zwei Schritten: Im April 1984 werden PEARL-Programme auf QU 68000 übersetzt und ausgeführt werden können, Testsystem und Datenbanksystem werden im Oktober 1984 implementiert sein.

Die Entwicklung der PEARL Engine 68000 wird im Rahmen des Projekts "Software-Technologie" der GMD vom BMFT gefördert.

4. Cross-PEARL-Programmiersystem VAX - Z 8000

4.1 PEARL für DEC VAX 11/750

Bereits abgeschlossen ist die Implementierung des PEARL-Compilers, -Ablauf- und Testsystems auf VAX 11/750 unter dem Betriebssystem VMS. Wesentliche Bedingungen des entsprechenden Auftrages von AEG waren:

- Übersetzung und Test von PEARL-Programmen für Zilog Z 8000 auf VAX 11/750.
- Ausführung von mehreren PEARL-Programmen auf VAX 11/750 simultan zueinander und simultan zu anderen Programmen von Timesharing-Benutzern.
- Übersetzung der PEARL-Programme nach FORTRAN IV zur Einbettung in ein existierendes Simulationssystem auf einem Großrechner.
- Anschluß von FORTRAN-Programmen an PEARL-Programme und umgekehrt auf VAX 11/750.

Deshalb wurde zur Übersetzung der PEARL-Programme der bei WERUM vorliegende Umsetzer IL1→FORTRAN IV auf VAX installiert, d.h. bei dieser Implementierung werden die PEARL-Anweisungsprogramme durch den PEARL-Compiler nach FORTRAN IV übersetzt und anschließend durch den FORTRAN-Compiler der VAX weiter verarbeitet. Die Ausführung und der Test der PEARL-Programme auf VAX 11/750 erfolgen mit den beschriebenen BAPAS-Komponenten und dem PEARL-Testsystem, wobei die reale Zeit auf eine virtuelle Zeitachse abgebildet wird.

Da das PEARL-System auf VAX 11/750 als "Entwicklungssystem" für Zilog Z 8000 eingesetzt wird, kann der durch den Umweg über FORTRAN bedingte Effizienzverlust auf VAX in Kauf genommen werden. (Im Gegensatz zu C kennt FORTRAN weder getrennte Compilierung von Moduln mit Import/Export-Beziehungen noch Strukturen und Pointer.)

Ein VAX-PEARL-Produktionssystem würde einen Codegenerator für VAX erfordern.

4.2 PEARL für Zilog Z 8000

Der Mikroprozessor Zilog Z 8000 wird in den geplanten Anwendungen ohne Standardperipherie, d.h. nur mit Prozeßperipherie, eingesetzt. Zur Ausführung von PEARL-Programmen wurden daher nur das PEARL-Betriebssystem BAPAS-K mit einigen zusätzlichen Laufzeitroutinen, nicht aber die E/A-Pakete, direkt auf der Hardware des Z 8000 implementiert. BAPAS-K und die Laufzeitroutinen zusammen beanspruchen nur 10 KB Speicher.

Die Übersetzung von PEARL-Programmen für Z 8000 erfolgt auf VAX 11/750 mittels des o.g. Compiler-Oberteils und eines Codegenerators, der sehr effizienten Am Z 8001 Assembler-Code erzeugt.

Die Assemblierung und das Binden der Programme erfolgen auf einem AmD-Entwicklungssystem.

Der Cross-Compiler PEARL→Am Z 8001 Assembler ist bei WERUM auch auf Siemens 330/R 30 installiert.

5. PEARL-Programmiersystem für Siemens 7.xxx unter BS 2000

Im Rahmen eines entsprechenden Auftrags entwickelt WERUM einen Codegenerator für Siemens 7.xxx unter BS 2000 und bereitet die Implementierung des PEARL-Betriebssystems, der E/A-Pakete und des PEARL-Testsystems für diese Rechner vor. Ziel ist, im Spätsommer 1984 PEARL-Programme auf Siemens 7.xxx unter BS 2000 simultan zu anderen Programmen übersetzen, ausführen und testen zu können, wobei die reale Zeit auf eine virtuelle Zeitachse abgebildet wird.

6. PEARL-Implementierungen in Brasilien

Nachdem im Herbst 1982 sechs brasilianische Software-Entwickler bei WERUM in PEARL und in das portable PEARL-Programmiersystem von WERUM eingeführt worden waren, begannen Anfang 1983 diese Entwickler sowie zwei Mitarbeiter von WERUM in Sao Paulo mit der Implementierung von PEARL für drei brasilianische Rechenanlagen /5/ :

Die COBRA 530 ist ein 16-bit-Minirechner der Staatsfirma COBRA. Der Prozessor ist aus Bit-slice-Prozessoren aufgebaut. Die Implementierung des PEARL-Systems, d.h. Erstellung des Codegenerators, Anschluß des Betriebssystems und der E/A-Pakete sowie Installation des Testsystems, wird durch die private Firma THEMAG durchgeführt. THEMAG arbeitet mit mehr als 1000 Ingenieuren u.a. an der Planung und Durchführung von Energie-Erzeugungs- und -Verteilprojekten einschließlich kompletter Staudamm- und Kraftwerksprojekte.

Die SISCO MB 8000 ist ein 16-bit-Minirechner der Privatfirma SISCO. Die Hardware und die Schnittstellen der System-Software entsprechen der NOVA 2 von Data General. Das PEARL-System wird von dem privaten Systemhaus ESCA implementiert, das vor allem Erfahrung aus Prozeßrechner-Projekten in den Bereichen Stahlproduktion und öffentlicher Transport und Verkehr besitzt. ESCA beschäftigt ca. 400 Mitarbeiter.

Die dritte PEARL-Implementierung wird von WERUM für die Rechenanlage Novadata durchgeführt, eine Architektur der Privatfirma Novadata auf Basis von Intel 8086.

Das Projekt wird auf brasilianischer Seite durch die Staatsfirma DIGIBRAS, Brasilia, auf deutscher Seite durch das Internationale Büro der GMD koordiniert. Der BMFT unterstützt die Arbeiten mit ca. 25% Zuschuß im Rahmen des Abkommens über Zusammenarbeit in der wissenschaftlichen Forschung und technologischen Entwicklung mit Brasilien.

7. PEARL-Compiler für LSI 11

Das Fraunhofer-Institut IITB, Karlsruhe, und WERUM entwickeln gemeinsam ein Cross-PEARL-System für LSI 11, bei dem PEARL-Programme auf Siemens R 10 zur Ausführung auf LSI 11 übersetzt und gebunden werden. Die Arbeiten sind Teil eines Projekts des IITB mit einer Universität in China.

Die zu entwickelnden Komponenten (Codegenerator und Ablaufsystem) werden so erstellt, daß sie für eine PEARL-Implementierung auf PDP-Rechenanlagen in möglichst großem Umfang wiederverwendbar sind.

Literatur

- /1/ Brunner, P.J.; Meffert, K.; Windauer, H.: PEARL Testsystem. PEARL Rundschau, Band 2, Nr. 6, Dezember 1981, S. 8 - 13.
- /2/ DIN 66 253, Programmiersprache PEARL. Teil 1: Basic PEARL, Beuth, Berlin 1981, Teil 2: Full PEARL, Beuth, Berlin 1982.
- /3/ F ä r b e r, G.: PEARL Engine 68000: Hardware. Tagungsband des Workshops "PEARL in der Rundfunktechnik", 29.09.83, Institut für Rundfunktechnik, München.
- /4/ G o e d e, K.; Landwehr, K.: BAPAS-DB - Ein portables offenes Datenbanksystem für Prozeßrechner. Informatik-Fachberichte 39, S. 443 - 452, Springer, Berlin-Heidelberg 1980.
- /5/ H a d d a d, M.; Tsai Chung Hsin: Linguagem de Tempo Real - PEARL. Anais "1. Congresso Nacional de Automacao Industrial - CONAI", Sao Paulo, 11. - 15.07.83, S. 522 - 524. SUCESU-SP, Sao Paulo 1983.
- /6/ H e i n e, P.: Wirtschaftliche Implementierung von Realzeit-Hochsprachen auf Mikrorechner-Systemen am Beispiel des RMX86-PEARL. Regelungstechnische Praxis, 25. Jahrgang 1983, Heft 7, S. 286 - 289.
- /7/ K e r n i g h a n, B.W.; Ritchie, D.M.: The C Programming Language. Prentice-Hall, Inc., 1978.
- /8/ S a u t e r, D.; Windauer, H.: PEARL Engine 68000: Software. Tagungsband des Workshops "PEARL in der Rundfunktechnik", 29.09.83, Institut für Rundfunktechnik, München.
- /9/ W e r u m, W.; Windauer, H.: Introduction to PEARL. Vieweg, Braunschweig 1982.
- /10/ W i n d a u e r, H.: Portables PEARL-Programmiersystem von Werum, Charakterisierung. PEARL Rundschau, Band 2, Nr. 1, April 1981, S. 28 - 32.

Anschrift des Autors

Dr. Hans Windauer
 WERUM Datenverarbeitungssysteme GmbH
 Glogauer Straße 2 A
 2120 Lüneburg
 Tel. 04131/53066