

COBOL Resource.

Release 2.05

Features

VS Cobol 85 Compiler, Procedure Interpreter, Command processor, Operator mode, Utilities and tools on Unix platform.

Cobol source code and VS-Procedures moved to unix with minimal modifications being required.

VS XDMS and DMS compatibility.

A \times C= API set to allow full Resource file system access.

Support for CONTROL, DATENTRY and REPORT.

VS Print queue and Job queue.

COBOL Resource is a complete development and runtime environment that allows you to move your Cobol applications to unix platforms, retain the look and feel of your current VS development environment, and run your existing VS COBOL programs on an unix platform. With COBOL Resource, you do not need to recode your source files, convert your Procedure Language files, or learn a new development environment.

Cobol Resource consists of the ANSI Standard Wang VS COBOL 85 compiler, the VS Procedure Interpreter, most of the VSSUBS, most major utilities, an interactive symbolic debugger and a VS DMS / XDMS file system, and the familiar Command Processor user interface. (See below).

Moving your VS COBOL applications to a unix platform can be a simple, straightforward, and cost

```
Help F13 F14 F15 F16 End Insert Delete Erase Home BackSpace BackTab Pickup Putdown
*** COBOL ReSource - Version 02.05.00 ***
Copyright, Wang Laboratories, Inc. 1994

Workstation pts0 Ready          9:54 pm   Thursday   August  3, 2000

Hello CFA
Welcome to ReSource

Press (HELP) at any time to interrupt your program or to stop
processing of the current command.

Use the function keys to select a command:

(1) RUN program or procedure          (9) Enter WORD PROCESSING
(2) SET usage constants              (11) Enter OPERATOR mode
(3) SHOW program completion report   (12) SUBMIT procedure
(4) Manage QUEUES                    (15) PRINT COMMAND screen
(5) Manage FILES/LIBRARIES           (16) LOGOFF
(6) Manage DEVICES
```

effective process with Cobol Resource. There are a number of proven procedures to move both source code and data files from the VS platform to the unix platform.

Thomas Junker (SRDI Representative)
tjunker@tjunker.com
www.tjunker.com

SRDI (Software Resource Development & Integration.)

If required Email softdevl@sunshine.net.au for the name of your Dealer / Representative.

Other product names mentioned in this data sheet may be registered trademarks or trademarks of their respective companies.

This Document is Copyright SRDI and may only be copied in its entirety. Document Version 1.01, September 14, 2000

The Resource Development Package

The central component of the COBOL Resource Development Package is the VS COBOL 85 compiler, which has been ported to run on unix. This is not a rewritten compiler, but rather a port of the same ANSI 85 compiler that you have been using on the VS.

However, the COBOL Resource Development Package is more than just a compiler. It features the Resource Symbolic Debugger - an interactive, easy-to-use, source-level, program debugging tool that supports Resource COBOL applications. Each Development Package also includes the LINKER utility and a complete Runtime Package for debugging and testing.

The COBOL Resource Compiler

The COBOL Resource compiler is a port of the VS COBOL 85 compiler with a new code generator tuned for the specific unix platform. The compiler generates an executable object module on UNIX, rather than an interpretive program, and therefore takes full advantage of the specific platform.

Compatibility - The COBOL Resource compiler is compatible with your VS COBOL programs, ensuring that you can continue to use your existing COBOL application source code for VS or UNIX operation.

Portability - Source files can be written and compiled on either the VS system or open system, without any need for conversion. This feature allows you to develop applications on both systems, without concern for the target platform. This coexistence capability ensures that your move to unix systems is accurate, seamless, and at your own pace. Development can continue on your VS as your staff gradually move to the unix systems, with no loss in productivity.

High Productivity Code Generation ---

Like its VS Counterpart, the COBOL Resource compiler generates executable object code and therefore has been tuned to take advantage of the target platform's architecture.

ANSI Compliance -The COBOL Resource compiler contains the same ANSI COBOL 85 extensions as the VS COBOL85 compiler.

Full Transaction Processing - This feature allows you to bracket a group of updates so they can be either committed or rolled back as a single, complete transaction, just as on the VS.

Workstation I/O - The popular Wang workstation I/O extension, DISPLAY AND READ. is fully supported in the COBOL Resource compiler. Thus, you do not have to change your VS Source programs to communicate with the workstation, and your end users continue using the same VS interface they are familiar with, including the use of tabs, default fields, and field validation.

Command Processor Shell

COBOL Resource users are greeted with the familiar VS Command Processor menu and PF key-driven interface after login, rather than with a UNIX command line. You can set usage constants for default source, run, object, print, and work libraries and volumes. With this shell, end users become immediately productive with COBOL applications, without retraining. The Resource shell also allows you to manage files, libraries, devices, and queues. The Procedure Queue and the Print Queue enable you to change the status of a job, change the position of the job in the queue, and remove jobs from the queue.

Procedure Interpreter

Cobol Resource includes a port of the VS Procedure Interpreter, providing compatibility with current VS Procedures. Simply move your procedure files to the new platform and immediately execute them.

PUTPARAM and GETPARAM

The PUTPARAM and GETPARAM facilities of the VS have also been preserved for COBOL Resource. PUTPARAM and GETPARAM, used by all VS applications to acquire information needed for execution at runtime, are fully integrated with the COBOL Resource Procedure Interpreter.

VSSUBS

Popular VSSUBS are re-created for COBOL Resource with the same programmatic interfaces that are used on the VS. Your COBOL applications calling these SUBS need only be recompiled on the new platform to access the identical system services.

Utilities

Like the VSSUBS, VS utilities are critical components of current VS COBOL 85 applications. To make the transition to unix systems easy, COBOL Resource provides many of these utilities, which can be run from the Command Processor shell. These utilities preserve the vast majority of the functions found on the VS.

CONTROL - With this utility, you can define the characteristics of a data file thus creating the CONTROL file that contains these specifications.

The CONTROL files are used by both DATENTRY and REPORT. You can move existing control files from VS to Resource where you can use these unchanged with the Resource CONTROL, DATENTRY, and REPORT utilities.

DISPLAY - The DISPLAY utility enables you to display source, print, and data files,

using a menu and PF key interface.

COPY allows you to copy files and libraries and contains many of the VS COPY features, including reorganization.

SORT includes most supported options found in the VS SORT utility.

DATENTRY - This utility enables you to create a data File in the format defined by the CONTROL file. Using DATENTRY, you can add, modify, or delete records in a file, as well as change their display characteristics.

REPORT utility enables a user to create customized reports based on the files created by the DATENTRY utility.

EDITOR - This easy-to-use text editor is similar to the VS Editor, with some features of ADEPT.

PERSON - Most of the VS PERSON utility is included, allowing you to modify keyboard and workstation personalities and colors.

VERIFY - Similar to VS VERIFY, this utility analyzes the structure of an Indexed file and reports any errors found. Several utilities are also included that allow system administrators to monitor a PDMS installation.

Symbolic Debugger

The Resource Symbolic Debugger (RSDB) is a complete- interactive, source-level, program debugging tool. Similar to its counterpart on the VS, RSDB allows you to single step through a COBOL application while you look at the source file. RSDB also features breakpoint setting, symbolic field referencing, searching the source, display of program screens and data windows. Two unique features of the debugger are 1. The ability to debug across link levels, and 2, The use of the Help key to enter debug during program execution.

LINKER

The COBOL Resource LINKER allows you to statically link compiled object and runtime modules to form single, executable objects. You create this link between two COBOL programs or between a COBOL ReSource program and a native C program. Additionally, you can link COBOL Resource to a dynamic link library (DLL), which allows you to create smaller executable objects.

Common Runtime System API (CRS)

Version 2.0 of Resource contains a program interface API. Using this API you can write programs to manipulate Resource files, and use other features of resource from most languages. Examples are supplied in $\times C=$.

The ReSource Runtime Package

Each Runtime Package includes the Resource Common Runtime System, that provides support for all the VS I/O features, including the VS DMX and XDMS functionality.

The Resource shell, complete with a Run screen, Manage Files and Libraries, And other VS command processor functions is a key component of the Runtime

Portable Data Management System

The Resource Common Runtime System utilizes the Portable Data Management System (PDMS) to provide its support for VS Indexed and Alternate Indexed files. PDMS is a data management system that provides Indexed file compatibility with VS DMS and XDMS files.

True Variable Length Records - True variable length record files utilize much less disk space, especially when they are large Indexed files common to COBOL applications.

Duplicate Alternate Key Access in Primary Key Order - This is a Wang extension that ensures complete source code compatibility.

Sharing and Locking of Resources - These facilities make it possible for more than one program to perform concurrent updates to different records in the same file. The system locks those records, preventing concurrent access until the update is complete.

Copyright Wang Laboratories, Inc.

Extensions to product are the Copyright of SRDI.

Other product names mentioned in this data sheet may be registered trademarks or trademarks of their respective companies.