infoWAT

Volume 1, Issue 7

September 1983

WATCOM Software, Version 2.0

One of our key objectives is to ensure the relevance of our products to consumer needs. As part of our commitment to listen carefully to and encourage user feedback, we have responded with the product enhancements and additions described in this newsletter.

The five languages and editor formerly known as Waterloo micro APL, BASIC etc., have been extensively rewritten or enhanced to elevate them beyond the traditional 'micro' limitations. This development was undertaken to accommodate increasing user needs for these products in interactive mainframe environments as well as in networking between micro and mainframe computers. Significant performance improvements have also been effected in the areas of common need.

Extensive new documentation as well as significant improvements in existing documentation have been created to support these new products, which include WATCOM APL, BASIC, COBOL, FORTRAN, Pascal and the Editor. These compatible versions are supported on the IBM Personal Computer with DOS, the IBM 370 with VM/SP CMS, and the DEC VAX with VMS.

WATCOM interactive language interpreters are designed specifically but not exclusively for educational use in teaching computer programming. Of special significance is the emphasis on effective error diagnosis and powerful debugging capabilities.

Enhancements to WATCOM APL and BASIC have been made with specific business usages in mind.

WATCOM APL V2.0

WATCOM APL is an extensively rewritten version of Waterloo microAPL version 1.2. Many new features have been added and existing features improved. For the IBM Personal Computer significant improvements have been made in performance, floating-point arithmetic (8 byte extended precision utilizing the 8087 co-processor if it is present; emulated in software otherwise), and workspace size (limited only by the amount of memory present in the PC).

New features include:
- The Random Array File System allows the user to store and retrieve APL objects from a disk file in random order (similar to the Sharp APL file system).
- Extensive report formatting facilities include ⎕FMT (from Sharp APL) and extended ⍕ picture formatting (from IBM's APL2). ⎕FMT uses FORTRAN-like formatting clauses, while extended ⍕ uses a "picture example" character left argument.
- String searching and substitution built in as a natural extension to indexing/indexed assignment.
- Input numeric processing allows verification (⎕VI) and forming (⎕FI) of character strings into numeric data.
- Event Trapping and Simulation gives the user the ability to trap and handle errors using Execute Alternate (⎕EA), Event Type (⎕ET), Event Message (⎕EM) and Event Simulation (⎕ES). (Modeled on APL2.)
- Workspace transfer functions allow the user to create files containing a form of

the workspace which can easily be converted back to the original workspace on any version of WATCOM APL on any computer system.

- Automatic workspace loading allows the user to develop an application which will run without the user requiring any knowledge of the underlying APL support.
- Multiple statement separator (◊) allows the user to place several related statements on one function line.
- User defined collating sequence (⎕CS) allows sorting of character arrays according to different character sets.
- Ability to lock user-defined functions.

Improved features:
- Major competitive speed improvements have been made in the following areas:
  - primitive function execution
  - user-defined function execution
  -)COPY and)PCOPY system commands
- Multiple numeric datatypes: boolean, integer and floating point numbers now take the minimum amount of room whenever possible for more efficient usage of the workspace.
- Gamma function is used for factorial (monadic !) and binomial (dyadic !).
- ⎕MEMALLOC allows the user to dynamically allocate memory for use with ⎕PEEK, ⎕POKE and ⎕SYS.

Additional features:
- The Native File system allows communication with the file system of the underlying machine.
- The Compress primitive function has been extended to allow an integer left argument, which allows the replication of items rather than just selection.

Special features for the IBM Personal Computer:
- The workspace includes all available memory.
- Arrays can have up to 32767 elements regardless of the data type. This allows floating point arrays of up to 256K bytes.
- Several features of APL may be optionally included when APL is invoked. Any memory not used as a result of a feature not being loaded is freed for use in the workspace.
- Support for 8087 co-processor allows fast, extended precision numeric computation (performed in software if 8087 not present)
- Colour Graphics support allows easy access to the PC graphics capabilities.
- 40 user-programmable function keys are provided.
- Host communication allows full access to the file system of a host machine.
- The PC may be used as a terminal using the system functions ⎕SETUP and ⎕PASSTHRU.
- ⎕SETUP allows the user to set and query serial port settings. The Native File system may be used to access the serial ports.
- ⎕CURSOR allows the user to change the position of the cursor as well as query its current position.

WATCOM BASIC V2.0

WATCOM BASIC is an extensively enhanced version of Waterloo microBASIC version 1.2. The major addition is formatted I/O capabilities with INPUT USING and PRINT USING. Many formatting features are available including credit and debit, floating dollar, and asterisk fill. Some examples of the new PRINT USING features follow (the computer response to the user "immediate mode" command is underlined in order to distinguish it):

```
? using '@### diggety ###', 'hotdog'
hot diggety dog
money$='@$$$,$$$V.## CR'
print using money$, 12345.67;-123
$12,345.67        $123.00 CR
```

Other enhancements are:
- Free format structured control statements allow you to enter statements like:
  ```
  for i=1 to 10:a(i)=i:next i
  ```
- Block labels on structure statements allow you to quit from more than one level without using a flag variable.
- Short and long precision floating point arithmetic.
- On the IBM Personal Computer, support for 8087 co-processor allows fast, extended precision numeric computation (performed in software if 8087 not present).

- Settable print zone width is provided.
- Renumber of a subset of the program allows portions of the program to be moved.
- Statement trace control has OPTION TRACEON and OPTION TRACEOFF.
- Logging of terminal input and output to a file can be done.
- Workspace name, date and time are displayed by the LIST command.
- Local variables and arrays can be defined inside functions and procedures. For example:

```
40 def fn_sum(mat x, n%),i%,sum
45 ! i%, sum are local variables
50   sum = 0
55   for i% = 1 to n%
60     sum = sum + x( i% )
65   next i%
70   fn_sum = sum
75 fnend
```

- There are new matrix operations for sorting.
- Matrix inverse has been added.
- New trigonometric functions such as COSH, LOG10 have been added.
- Several new string functions have been added. Some examples:

```
s$ = 'Mixed'
! show upper case, lower case
print uprc$( s$ ); lwrc$( s$ )
MIXEDmixed
! show substring replacement
print srep$( s$, 'ed', 'ing' )
Mixing
! show translate function
print translate$( s$, 'dM', 'sf' )
fixes
```

WATCOM COBOL V2.0

It is designed to support a substantial implementation of ANS COBOL (1974). The language supported includes level 1 of the following modules: Nucleus, Sequential I-O, Relative I-O and Table-handling. As well, parts of level 2 are supported in these modules including full support for the PERFORM, STRING and UNSTRING verbs. WATCOM COBOL includes the WATCOM Editor for preparing programs as well as a powerful interactive debugging facility.

Improvements have been made in performance and diagnostics. Several problems of a minor nature have been corrected. No extensive new features have been added.

WATCOM FORTRAN V2.0

The original Waterloo microFORTRAN was essentially a dialect that varied in many respects from standard FORTRAN-77 in order to accommodate specific teaching aids. Strong user feedback indicated that a much closer implementation of the standard was required. WATCOM FORTRAN V2.0 has had extensive and highly detailed changes made to many areas of the language in order to satisfy user needs and includes a significant portion of the FORTRAN-77 definition.

The powerful editor and interactive debugger that are included in the interpreter enhance the utility of the product during program development and debugging.

New features include:
- The COMMON and DIMENSION statements have been added.
- CHARACTER support now corresponds to the FORTRAN 77 standard.
- Additional functions such as REAL and ATAN2 have been added.
- Variable definition checking has been made optional.
- On the IBM Personal Computer, support for 8087 co-processor allows fast, extended precision numeric computation (performed in software if 8087 not present).

Language changes include:
- Subprogram recursion is no longer allowed in conformance with the FORTRAN-77 standard. As a result the name of a function can be used within that function as a variable, without it causing an invocation of the function.
- The A format code used in formatted IO has been modified to operate according to the FORTRAN 77 standard.
- CHARACTER variables in previous versions of WATCOM FORTRAN were of variable length. They are now of fixed size, and are declared with a length

attribute. This change will affect existing programs which use CHARACTER variables.

WATCOM Pascal V2.0

WATCOM Pascal corresponds very closely to the ANSI Pascal standard and to the ISO Pascal committee draft proposal. This implementation includes sophisticated features such as text file support, pointer variables, multidimensional arrays, string functions, assembly-language interface, interactive debugging, and support for colour graphics on the IBM PC.

Significant performance improvements have been made in the interpreter, and the following new features have been added:

- Procedure and function names can now be passed as parameters to other procedures and functions.
- Random file access can be performed using the "Update" and "Setnext" procedures. "Update" opens a file such that both input and output operations can be performed on it. "Setnext" specifies the element in a file to which the next input or output operation applies.
- The "append" procedure opens a file such that subsequent output operations to the file will append to its previous contents.
- The "else" clause in a "case" statement is an extension to standard Pascal which permits a statement to be executed when no match is found for a case selector expression. The "case" statement is illustrated by the following:

```
case character of
  'a' : Process( y, y );
  'b', 'c' : ; { null action }
  'd' :
      begin
        a := 1;
        b := 2;
      end;
  'e' : Process( n, y );
  else Other( y );
end
```

- The "initrandom" procedure and "random" function provide a pseudo-random number generation mechanism.
- On the IBM PC, support for 8087 co-processor allows fast, extended precision numeric computation (performed in software if 8087 not present).

WATCOM Editor V2.0

The WATCOM Editor is a general-purpose text editor that supports both a command-oriented and on-screen mode of editing. Function key support further enhances the text manipulation capabilities of the editor.

Command mode features include:
- line positioning by number or by context using powerful string searching
- line insertion or deletion
- global and selective text changes
- command recall and modification
- file manipulation commands such as directory listing, file rename, deletion, etc.
- ASCII terminal emulation for connection to remote host computers

On-the-screen mode features include:
- positioning using cursor movement keys
- overtyping of text
- insertion and deletion of text

Function key support includes:
- up or down movement through text a line or a screen-full at a time
- insertion or deletion of lines
- selection of foreground and background colour on the IBM PC
- new function key support; line split, line join, and a single command/screen mode key

Enhancements to the software include:
- a new meta-character "%#" that substitutes for the line number of the line being changed thus permitting numbering of lines in the workspace
- "edit" and "exit" commands
- "get" and "put" commands no longer change the current file name
- the current file name is displayed on the screen as a convenient reminder of the name of the file being editted
- a revised SET command on the IBM PC that displays serial port characteristics