



**Poseidon House
Castle Park
Cambridge CB3 0RD
United Kingdom**

TELEPHONE:
INTERNATIONAL:
FAX:
E-MAIL:

**Cambridge (01223) 515010
+44 1223 515010
+44 1223 359779
apm@ansa.co.uk**

ANSA Phase III

ANSAware/RT 1.1 Release Notes

Ian Macmillan

Abstract

This document forms the release notes for ANSAware/RT 1.1. It describes the changes since the previous release. As such it is supplementary to the ANSAware/RT 1.0 Manual [APM95] and the previously issued ANSAware 4.1 manual set [APM92], both of which are included with the distribution.

APM.1675.00.01

Draft
Technical Report

20th December 1995

Distribution:
Supersedes:
Superseded by:

Copyright © 1995 Architecture Projects Management Limited
The copyright is held on behalf of the sponsors for the time being of the ANSA Workprogramme.

ANSAware/RT 1.1 Release Notes



ANSAware/RT 1.1 Release Notes

Ian Macmillan

APM.1675.00.01

20th December 1995

The material in this Report has been developed as part of the ANSA Architecture for Open Distributed Systems. ANSA is a collaborative initiative, managed by Architecture Projects Management Limited on behalf of the companies sponsoring the ANSA Workprogramme.

The ANSA initiative is open to all companies and organisations. Further information on the ANSA Workprogramme, the material in this report, and on other reports can be obtained from the address below.

The authors acknowledge the help and assistance of their colleagues, in sponsoring companies and the ANSA team in Cambridge in the preparation of this report.

Architecture Projects Management Limited

Poseidon House
Castle Park
CAMBRIDGE
CB3 0RD
United Kingdom

TELEPHONE UK
INTERNATIONAL
FAX
E-MAIL

(01223) 515010
+44 1223 515010
+44 1223 359779
apm@ansa.co.uk

Copyright © 1995 Architecture Projects Management Limited
The copyright is held on behalf of the sponsors for the time being of the ANSA Workprogramme.

Architecture Projects Management Limited takes no responsibility for the consequences of errors or omissions in this Report, nor for any damages resulting from the application of the ideas expressed herein.

Contents

3	1	Overview
3	1.1	Introduction
3	1.2	Upgrade from ANSAware/RT 1.0 to 1.1
3	1.2.1	Windows NT (WIN32) platform
3	1.2.2	Bug fixes
4	1.3	Requirements
4	1.3.1	ANSAware/RT platforms
5	1.3.2	System requirements
5	1.4	Release media
5	1.5	Support
6	1.6	Source code roadmap
9	2	ANSAware/RT for the WIN32 Platform
9	2.1	Introduction
9	2.2	Loading the ANSAware/RT distribution
9	2.2.1	ANSAware/RT distributions
9	2.2.2	The ANSAware/RT distribution tape
10	2.2.3	Loading the generic distribution
11	2.2.4	Loading the WIN32 distribution
11	2.3	Building ANSAware/RT
11	2.3.1	Establishing the build environment
12	2.3.2	The Build Process
13	2.3.3	<i>Bootstrapping</i> the build system
13	2.3.4	Building the utilities
13	2.3.5	Running the site configuration utility
14	2.3.6	Generating the makefiles
14	2.3.7	Compiling and Linking
15	2.4	Testing ANSAware/RT
15	2.4.1	Creating a binary installation
15	2.4.2	Running ANSAware/RT

1 Overview

1.1 Introduction

This document forms the release notes for ANSAware/RT 1.1. It describes the changes since the previous release. As such it is supplementary to the ANSAware/RT 1.0 Manual [APM95] and the previously issued ANSAware 4.1 manual set [APM92], both of which are included with the distribution in the form of postscript files.

1.2 Upgrade from ANSAware/RT 1.0 to 1.1

The main areas of change at this release are:

- Support for an additional platform: Microsoft Windows NT (WIN32) running on Intel 32-bit hardware
- Various bug fixes.

1.2.1 Windows NT (WIN32) platform

Support has been added for running ANSAware/RT on Windows NT. The port uses the latter's WIN32 programming interface and in principle should also run on Windows 95. However, it has only been tested on the former. Windows NT Workstation 3.51 was used in the development of the WIN32 distribution, but Windows NT Server should also be suitable. Support for WIN32 in this release is limited to Intel 32-bit hardware although it is considered a small and relatively straightforward task to extend this to other hardware platforms on which the WIN32 programming interface is available.

The WIN32 port is a complete implementation of ANSAware/RT, i.e. it supports trader, factory and node manager. This is in contrast to previous releases of ANSAware, for Windows 3.1 for example.

The WIN32 port is application source code compatible with ANSAware/RT running on other platforms. It will also interwork with ANSAware 4.1 using the UDP protocol.

1.2.2 Bug fixes

Various minor bugs have been fixed since the initial release and these will not be listed here. Major faults that have been corrected are:

- Use of multiple RPCs (e.g. REX and TREX) within one capsule no longer causes eventual failure;
- Multiple initiates and redeems, particularly where the order of the latter does not match that of the initiates, no longer causes the capsule to terminate with an error;
- Multiple client and server objects in a single capsule no longer cause ANSAware/RT to deadlock.

1.3 Requirements

1.3.1 ANSAware/RT platforms

As with the previous release, ANSAware/RT has certain requirements on any *platform* on which it executes. For the most part, these are similar to those of ANSAware 4.1 but with the additional requirement that the underlying operating system must have real-time capability. In particular, ANSAware/RT tasking is implemented over POSIX threads (p-threads).

The following table lists the platforms on which this release of ANSAware/RT is supported.

Table 1.1: Supported platforms

Platform	CPU	System	O/S	Libraries
alpha_osf_1.3.rt	alpha	DEC Alpha	OSF1	-lpthreads -lmach -lc_r
alpha_osf_3.0.rt	alpha	DEC Alpha	OSF1	-lpthreads -lmach -lc_r
win32	Intel 32-bit	IBM compatible	Microsoft Windows NT	wsock32.lib

ANSAware/RT runs over DEC Alpha/OSF1, versions 1.3 and 3.0.¹. As of this release, it also runs over Microsoft Windows NT (WIN32) on Intel 32-bit hardware.

1.3.1.1 Disc requirements

The following table lists the disk space requirements for ANSAware/RT on the supported platforms.

Table 1.2: Disk space requirements in megabytes

Platform	Distribution	Build Tree	Installation	Total
generic	25	n/a	n/a	n/a
alpha_osf_1.3.rt	20	88	37	145
alpha_osf_3.0.rt	20	88	37	145
win32	4	28	11	43

1.3.1.2 Communication requirements

ANSAware/RT uses only the internet family of network protocols. At this release it is further restricted to UDP.

1. ANSAware/RT has been successfully ported to HP/RT and LynxOS [APM.1207], but since these systems are not generally available, these ports are not being provided with this release. It is anticipated that further ports will be produced as suitable platforms become generally available.

1.3.2 System requirements

This section lists the system requirements, in terms of system software needed by ANSAware/RT.

1.3.2.1 *Alpha OSF/1 requirements*

ANSAware/RT should be compiled on DEC Alpha systems with the standard OSF1 compiler (cc) and using the standard make utility (/usr/bin/make). The compiler option -std1 is required to ensure ANSI compatibility and to define some additional conditional compilation macros (see the OSF1 man pages for details).

Results using other tools cannot be guaranteed and in particular, the GNU versions of the compiler (gcc) and make utility (gmake) are known to fail.

1.3.2.2 *WIN32 requirements*

In principle the WIN32 distribution may be compiled with any suitable 32 bit C compiler. However, the only compiler tested is Microsoft Visual C++ version 1.10, 32-bit edition.¹ Using this compiler, no compiler options need to be explicitly specified over the default values supplied by the ANSAware/RT build system.

The build process makes heavy use of the `nmake` utility which is supplied with Visual C++.

The build makes use of the winsock network library `wsock32.lib`, which is part of Windows NT.

The WIN32 distribution should run on any version of Windows NT (both Workstation and Server) for Intel 32-bit hardware and also on Windows 95. However, it has only been tested using Windows NT Server 3.1 and Windows NT Workstation 3.51: no testing has been performed on Windows 95.

This release is dependent on file systems supporting filenames longer than the DOS 8.3 convention, i.e. it cannot be used on a standard DOS FAT file system.

1.4 Release media

ANSA sponsors may access the released software by ftp ([ftp.ansa.co.uk](ftp://ftp.ansa.co.uk)). Currently only the generic distribution is available, packaged in tar format.

The software and manuals are normally available on 4.0mm DAT written in tar format. Other formats may be provided on request and will depend on the facilities available to APM at the time.

1.5 Support

ANSAware software is being made available in source form to enable recipients to undertake experiments and porting. It is not guaranteed to be free of defects although it has been extensively tested with all the facilities which the ANSAware/RT implementation team have available. ANSAware/RT is not supported as a warranted software product and any or all parts of ANSAware/RT may change in future versions. The ANSA architecture will

1. There is no C++ specific code in the WIN32 distribution of ANSAware/RT.

continue to evolve and these changes will be reflected in the design and implementation of ANSAware and ANSAware/RT.

If you experience any trouble with porting or using ANSAware/RT please contact Architecture Projects Management Ltd. (APM) at the address inside the front cover of the associated manual [APM95]. APM wish to be informed of all problems and difficulties encountered and, within the constraints of the ANSA work programme, will give advice on how to deal with them.

If your use of ANSAware/RT is governed by a Software Licence, then this will tell you in detail the nature and extent of any support that you are entitled to receive.

APM will be pleased to receive any extensions, improvements, new ports or additional services for incorporation in future releases, at its discretion. Conditionally compilable sources are preferred.

Please e-mail ANSAware/RT bug reports to: ansaware@ansa.co.uk.

1.6 Source code roadmap

The ANSAware/RT master tree is structured into a number of independent components as follows. Although the differences are comparatively minor, the entire structure is reproduced here for completeness.

- **master/util:** utilities
 - **config:** ansaimake configuration files
 - **prepc:** prepc source code
 - **stubb:** stubc source code
 - **imake:** ansaimake source code
 - **scripts:** package, ansaconfig and other shell scripts
 - **makedep:** ansadepend source code
 - **sutils:** simple utility programs
- **master/include:** header files
 - **ansi:** header files for use with ANSI C
 - **lib:** header files for use with application libraries
 - **capsule:** platform independent header files
 - **capsule/CORE_unix:** UNIX specific header files
 - **capsule/CORE_w32:** WIN32 specific header files
 - **capsule/stack:** platform specific stack handling header files
 - **config:** platform specific capsule configuration files
 - **machine:** machine or hardware specific header files
 - **opsys:** operating system specific header files
 - **stubb:** header files for stub generation
 - **stubb/MoveMacs:** machine specific header files for stub generation
 - **idl:** contains IDL files for ANSA services
- **master/src/ansa/capsule:** capsule libraries
 - **CORE_unix:** UNIX specific capsule code

- **CORE_w32**: WIN32 specific capsule code
- **master/src/ansa/test**: test programs for capsule libraries
- **master/src/ansa/trader**: trader sources
- **master/src/ansa/factory**: factory sources
- **master/src/ansa/nodemgr**: node manager source
- **master/src/ansa/reloc**: relocation service example
- **master/src/lib**: user libraries
 - **reloc**
 - **checkpt**
- **master/src/examples**: examples
 - **Alarm**
 - **BTest**
 - **Cksum**
 - **DHStone**: real-time distributed Hartstone benchmark
 - **DHStcb**: as above, communications bandwidth variation
 - **DHStpp**: as above, protocol preemptivity variation
 - **DHStpq**: as above, priority queuing variation
 - **Echo**
 - **Except**
 - **Mgmt**
 - **Netinfo**
 - **RTEcho**: real time variation of Echo example
 - **SBank**
 - **Sample**
 - **SlowInt**
 - **Stubtest**
 - **bindEcho**: explicit binding example
 - **test1**
 - **timeEcho**: real time variation of Echo example
- **master/contrib**: contributed applications: these are not produced or supported by APM.

The ANSAware/RT root directory contains a number of subdirectories which contain scripts and configuration files; these are as follows:

- **master**: the master source tree described above
- **ANSAware**: contains ANSAware/RT configuration files and platform specific utility scripts for ANSAware.sh
- **examples**: contains examples parameters used as input to ANSAware.sh
- **bin**: contains scripts used by ANSAware.sh and packages
- **awkscrs**: contains awk scripts used by ANSAware.sh and packages

- **dirs:** contains the list of directories describing the master source tree structure
- **packages:** contains package configuration files and package specific scripts

hardcopy: contains an on-line, PostScript copy of this manual.

2 ANSAware/RT for the WIN32 Platform

2.1 Introduction

This chapter describes the procedure for producing the set of executable and configuration files required to run ANSAware/RT on a WIN32 platform. There are three main steps in the process:

- Loading the ANSAware/RT distribution
- Building ANSAware/RT
- Testing ANSAware/RT

2.2 Loading the ANSAware/RT distribution

2.2.1 ANSAware/RT distributions

There are two ANSAware/RT distributions available at Release 1.1:

- the generic one, which at this release may be built for the OSF1 Unix platform, or may be used to produce a WIN32 distribution for subsequent building on a Windows NT platform¹.
- the WIN32 distribution for Microsoft Windows NT on Intel 32-bit hardware.

The loading procedure differs depending upon which distribution you have requested. Normally the generic distribution is supplied and the procedure for this is described in 2.2.3. Loading of a WIN32 distribution is described in 2.2.4.

2.2.2 The ANSAware/RT distribution tape

Both distributions of ANSAware/RT are normally distributed on 4.0mm DAT tape, written in `tar` format. The tape is written on Unix using the GNU version of `tar` with the following options:

```
tar cvof /dev/rmt11
```

which specify old format archives and no hardware tape compression. This combination offers the maximum portability in terms of being able to restore the tape using different versions of `tar` and various types of DAT device.

1. ANSAware/RT has been successfully ported to HP/RT and LynxOS [APM.1207], but since these systems are not generally available, these ports are not being provided with this release. It is anticipated that further ports will be produced as suitable platforms become generally available.

2.2.3 Loading the generic distribution

Note: This stage requires about 25Mb of disk space for the generic distribution and a further 4Mb for the WIN32 shadow tree, i.e. 29Mb in total. This space may be reclaimed after the WIN32 shadow tree has been transferred to the target WIN32 machine (described in 2.2.3.4).

The generic distribution must be loaded (restored and *installed*) on a Unix machine. A WIN32 distribution suitable for building on a Windows NT platform is then generated from the generic distribution and transferred to the target WIN32 platform. These steps are described in more detail in the following sections.

2.2.3.1 Restoring the generic distribution

Restoring the generic distribution on a Unix platform is the same procedure as for the previous release of ANSAware/RT. Briefly, this involves creating a directory to hold the distribution, making it the current directory and using a suitable `tar` command. For example:

```
mkdir ANSAwareRT1.1
cd ANSAwareRT1.1
tar xvf /dev/rmt1
```

On your system the tape device will probably have a name other than `/dev/rmt1`; if this is so then substitute your device name in the above command. After reading the tape, the restored directory structure should appear as described in 1.6.

2.2.3.2 Installing the Generic Distribution

Note: This stage is dependent on an ANSI compliant C compiler being available on the Unix host as it involves compiling `imake`, and generating `makefiles` from `imakefiles` which use ANSI C constructs.

Once the tape has been read, the distribution must be *installed*. This is a process which performs some initial checking and setup that must be done before ANSAware/RT can be built and before further distributions can be produced. It is accomplished by running the `InstallAW.sh` script on the host Unix machine. At this release, only the `GENERIC` distribution may be installed and this should be specified at the prompt asking which distribution you wish to install.

2.2.3.3 Creating a WIN32 Distribution on Unix

A WIN32 distribution must be created on the host Unix machine after *installation* of the generic distribution, by running the `ANSAware.sh` script. Selecting `win32_dist` as the platform will cause a *shadow tree* to be produced which may be subsequently transferred to a WIN32 (e.g. Windows NT) platform for building.

Note: It is not possible to build within the WIN32 distribution tree on Unix: the contents of the tree must first be transferred to a WIN32 platform. This is described in more detail in 2.2.3.4.

2.2.3.4 Transferring a WIN32 distribution from Unix

Note: The file and directory structure of a WIN32 distribution created from a generic distribution, and that in a WIN32 specific distribution, are very similar. The former has a few additional files which are not required, but which will not cause any problem if they are transferred to a WIN32 platform.

The WIN32 shadow tree may be transferred from the Unix host to the WIN32 target platform by any convenient mechanism, e.g. ftp, tape, nfs. Since ftp is provided with Windows NT, two scripts are provided to ease this type of transfer.

Located in the WIN32 shadow tree in the directory `<top level>/util/win32/scripts`, are two scripts:

- `maketree.bat`
- `install.ftp`

These should be individually transferred to the target machine.

Maketree.bat may then be run on the target platform, in a console window, to create the ANSAware/RT directory structure:

```
maketree <drive> <top level directory>
```

For example:

```
maketree C awrt1.1
```

would create a directory called `awrt1.1` on the C drive.

Install.ftp is a script containing ftp commands to fetch the required files into their respective directories on the target platform. Before use, it must be customised by changing the first two lines to correspond to a suitable user name and the location of the WIN32 shadow tree on the source machine. It is then used on the target machine from a console window as follows:

```
ftp unix-host-name <install.ftp
```

where `unix-host-name` is the name of the machine on which the generic distribution was loaded. This will transfer the entire directory structure, ready for building.

Note: When using the Windows NT version of ftp, ignore the repeated error messages that are produced before the transfer proper begins. These are a result of this version of ftp not understanding the comment symbol and hence trying to interpret the comments as commands.

2.2.4 Loading the WIN32 distribution

Restoring a WIN32 specific distribution from tape will depend on the distribution media, how it was written and the facilities available on the target machine for restoring the data. As described in 2.2.2, ANSAware/RT is normally distributed on 4.0mm DAT tape in tar format. This may be restored using one of the publicly available GNU tar implementations available for Windows NT.

2.3 Building ANSAware/RT

2.3.1 Establishing the build environment

The build procedure is command line driven and carried out entirely in a console window (CMD.EXE on Windows NT). Before the build can proceed, an appropriate command environment must be established.

This includes setting up your path to provide access to compilers and libraries, values for environment variables used to direct the build and so on. To assist you in this, a batch file `ansaenv.bat` is provided in:

```
<top level directory>/util/win32/scripts
```

and contains suitable default values which may need to be customised to suit your particular environment.

The script may be executed explicitly in a new console window created to perform the build, or it may be specified in a PIF file and associated with a suitable Program Manager icon. The former requires less effort to set up and is more straightforward if the build is only to be performed once, whilst the latter is more suitable for performing repeated builds.

2.3.1.1 Creating a build icon and PIF file

If it is desired to set up a Program Manager icon for automating the starting of a console window with a suitable build environment as described above, it is necessary to first create a PIF. This is performed using the PIF editor and specifying:

- `CMD.EXE` as the Program Filename
- Any suitable Window Title, e.g. ANSAware/RT Build
- The directory where you installed ANSAware/RT as the Startup Directory

and in the Windows NT dialogue:

- The full pathname of `ansaenv.bat` as the Autoexec Filename
- The default `CONFIG.NT` as the Config Filename.

Having created the PIF and saved it to a suitable location, proceed by creating a Program Manager icon and specifying the following in the properties dialogue:

- Any suitable Description, e.g. ANSAware/RT Build
- The PIF you just created as the Command Line
- Your ANSAware/RT installation directory as the Working Directory.

Double clicking on the newly created icon should then start a console window with an appropriate ANSAware/RT build environment. This window is also suitable for running ANSAware/RT applications.

2.3.2 The Build Process

The build process comprises the building of binary versions of libraries, servers and commands and consists of the following steps:

- *Bootstrapping* the build system
- Building the essential build and configuration utilities
- Running the configuration utility
- Constructing the makefiles from the imakefiles
- Using `nmake` to control compilation, linking and binary installation.

A script is provided to automate the build, called `ansaware.bat` and located in the script directory as before. It is invoked using the same parameters as `maketree.bat`, i.e. the drive and top level directory where the WIN32 tree was

installed. Before running this script you need to have certain information to hand in order to respond to the prompts issued by the configuration utility. This is described in 2.3.5.

`Ansaware.bat` is a short script and each of the steps may be performed individually. In particular, if changes are made and need to be incorporated, the whole build need not usually be repeated. It is usually quicker under these circumstances to perform only the required steps by manually invoking a subset of the commands in `ansaware.bat`.

For example, if a source file is modified it is only necessary to compile and link the changes, which may be achieved by the running `nmake` with the default target `all`. The build steps and what they do are described in more detail in the following sections.

2.3.3 *Bootstrapping the build system*

Since the build system uses `imake` to construct the makefiles which will subsequently be used by `nmake`, the first step is to bootstrap the system from an initial makefile, `Makefile.b`. The command is as follows:

```
nmake -f Makefile.b MasterMakefile
```

and this builds the `imake` utility and then runs it on the master `Imakefile` to produce the master `Makefile` from which everything else is built.

2.3.4 *Building the utilities*

The next step in the build process is to build the configuration utility `sitecnfg` and the well-known-interface-reference generator `wkifref`. This is achieved by the following command:

```
nmake Boot
```

2.3.5 *Running the site configuration utility*

Before running the configuration utility, you must have the following information:

- The full path of where ANSAware/RT is installed
- Where you would like the binary installation to be placed
- Which MPS modules you require (at this release only UDP is supported)
- Which machine is to run the trader, you need:
 - Node name
 - IP address, e.g. 192.5.254.29
- Whether you wish to use an alternative compiler
- Any additional compilation options you require.

Each prompt has a default which may be selected by responding with just a carriage return. The defaults are as follows:

- The current directory
- `<installation directory>\install\win32`
- UDP
- The machine performing the build

- CL (the Visual C++ compiler)
- No additional compiler options.

The configuration utility is invoked by typing its name:

```
sitecnfg
```

It will respond with an interactive dialogue requesting the above information.

The configuration utility generates a header file `options.h` (in `<top level>\include`) containing information about the trader and protocols selected. It also generates a file `site.def` (in `<top level>\util\config`) containing pathnames and compiler details which will be used in the construction of the makefiles in the next build step.

2.3.6 Generating the makefiles

The makefiles are now generated recursively down the directory structure from the `Imakefiles` and using information contained in `site.def` placed there by the configuration utility. The command used is:

```
nmake Makefiles
```

Note: At this point in the Unix build, the header file dependencies would be placed in the generated makefiles using the `depend` rule to `nmake`. This is not recommended for the WIN32 build as the utility which does this (`makedep`) does not understand the `#if` pre-processor directive. As a result it adds some incorrect dependencies which will cause the build to fail, saying it does not know how to make these files.

2.3.7 Compiling and Linking

To compile and link ANSAware/RT, the following command is used:

```
nmake all
```

this will cause `nmake` to begin building in the current directory and then recursively build in each subdirectory. Partial builds are therefore possible by invoking this command in a lower level directory, rather than in the top level installation.

2.3.7.1 Compiler and Link Warnings

When using Visual C++ with the default build options, several compiler warnings are emitted which are expected and should be ignored. Most of these relate to code automatically generated by Unix tools such as `yacc`. The following lists the most common types of error although this list is not necessarily exhaustive.

- Unreferenced local variables
- Parameter mismatches
- Conversion between different floating point types
- Signed/unsigned mismatches

The following are expected errors from the linker:

- Inconsistent DLL linkage (dll export assumed)
- `NULL_IMPORT_DESCRIPTOR` already defined in `ansa.lib`

These are a result of the existing source structure being incompatible with building the capsule library as a dynamic link library (dll). This could be fixed and is not difficult, but would require significant effort as many declarations need modification in various source files.

2.4 Testing ANSAware/RT

2.4.1 Creating a binary installation

Once ANSAware/RT has been built, it is possible to create a binary installation that contains only the files that are essential to the running of ANSAware/RT and its applications. This is achieved by typing:

```
nmake install
```

which will copy all the required files to the binary directory that you specified when ANSAware/RT was configured (see 2.3.5).

If you are running or developing applications, but are not planning to modify the source of ANSAware/RT itself, creating a binary installation allows the space occupied by the source to be reclaimed, i.e. the remainder of the shadow tree may be deleted. Alternatively, if you are running ANSAware/RT on several WIN32 platforms, only the binary installation need be copied to those platforms being used for developing or running ANSAware/RT applications.

2.4.2 Running ANSAware/RT

The WIN32 distribution is a full implementation of ANSAware/RT in that it supports all the ANSAware/RT services, e.g. trader, factory, node manager. However, it only supports the UDP network protocol.

Unlike the Unix distribution, ANSAware/RT for WIN32 builds the capsule library as a dynamic link library (DLL). This significantly reduces the size of applications and removes the need to re-link after making changes to the library.

Note: Because the capsule library is a DLL, it is essential that the directory in which it is placed appears in your path. Failure to ensure this will result in applications failing to start because they cannot locate the library. The `ansaenv.bat` script described in 2.3.1 ensures that the appropriate directory is placed in your path.

ANSAware/RT applications and services all run as console applications under WIN32. This was done to speed the implementation time and reduce the differences between the Unix and WIN32 distributions. There is no reason why applications cannot be produced which present a more conventional GUI based look and feel.

Applications may be launched from an existing console window, or may be associated with program manager icons, invocation of which will cause a new console window to be started (see 2.3.1).

Command line options and environment variables are identical to the Unix distribution and are described in the ANSAware 4.1 manual [APM92].

References

[APM.1207]

G Li, Real-Time ANSAware Version 1.0: Programming and System Overview, APM document 1207, APM Ltd., Cambridge CB3 0RD, U.K., May 1994.

[APM92]

APM Ltd., ANSAware Version 4.1 Manual, APM Ltd., Cambridge CB3 0RD, U.K., May 1992.

[APM95]

APM Ltd., ANSAware/RT 1.0 Manual, APM Ltd., Cambridge CB3 0RD, U.K., May 1995

