Introduction to Tao Documentation

1 About this document

This document provides an overview of the documentation provided in pdf format The material is organised, for convenience, into categories based on projected needs of different users. This classification of the documentation is somewhat arbitrary and it should be realised that most users of Elate® will almost certainly find that the various areas overlap.

Please note that the documents listed here are intended to provide a general introduction to parts of the system. As such, they are complementary to, and not intended as a replacement for, API or other documentation elsewhere in Elate® or int**e**nt®. The areas are:

♦	Introduction	♦	VP / C programmer
♦	Device driver programmer	♦	System programmer
♦	Application Programmer	♦	Java programmer

At the end of this document, there is a summary list of all manuals.

2 Introduction

2.1 Getting Started with int**e**nt on Windows®

This document is designed as a general introduction for the programmer using int**e**nt with Windows[®] for the first time. It includes step-by-step guides to installing and configuring int**e**nt, a description of the user interface is described, and an introduction to int**e**nt 's 'help' facilities. *Getting Started with int**e**nt on Windows* also provides a basic introduction to system features such as the shell, the debugger, the Audio Visual Environment, the VP assembler, the C and C++ compilers, Sysbuild, Sysgen, the translation procedure and use of Java[TM] Byte Code. All of these are treated in greater detail elsewhere. Since this document differs from the other *'Getting Started'* documents only in platform specific details, it is unnecessary to read all of these documents.

2.2 Getting Started with int**e**nt on Linux®

This document is designed as a general introduction for the programmer using Elate with Linux for the first time. As in the case of *Getting Started with int**e**nt on Windows* this document includes step-by-step guides to installing and configuring Elate, a description of the user interface, and an introduction to Elate's 'help' facilities. Similarly, *Getting Started with int**e**nt on Linux* provides a basic introduction to system features such as the shell, the debugger, the int**e**nt media libraries, the VP assembler, the C and C++ compilers, Sysgen, Sysbuild,the translation procedure and use of Java[TM] Byte Code. All of these are treated in greater detail elsewhere. Since this document differs from the other 'Getting Started' documents only in platform specific details, it is unnecessary to read all of these documents.

2.3 Technical Glossary

This relatively short glossary is designed to complement the rest of the system documentation. It defines a number of basic terms, concentrating upon those that the documentation uses with the greatest frequency, or uses in a fashion that differs from the commonly accepted sense. This glossary is aimed at readers who already possess a degree of technical knowledge, but who are new to Tao's technologies

3   Java^{TM} Programmer

3.1   Introduction to Tao's Java^{TM} Technology solution

This is designed to explain the benefits of combining Java^{TM} technology with int**e**nt, Java technology edition (JTE), to a reader that may be unfamiliar with even the basic principles of the Java^{TM} language and the int**e**nt content foundation. The document is not designed to train the reader in the use of the Java^{TM} language, and only a general overview of the language is provided. The concepts behind int**e**nt are described in basic terms, and the advantages of combining Java^{TM} technology with int**e**nt^{JTE} are listed. Some practical guidance on the use of the Java^{TM}-language-to-VP translator are supplied. The document does not presuppose any prior knowledge of the concepts involved in object-orientated programming, and a simple account of this is also provided.

3.2   Using Java^{TM} Technology with int**e**nt

This guide is designed for the Java programmer wishing to write code for Java^{TM} technology on int**e**nt^{JTE}. Like the *Introduction to Tao's Java^{TM} Technology Solution*, this document does not provide training in the Java^{TM} language itself, although in this case because a prior familiarity with the language is assumed. One section describes the architecture of the int**e**nt^{JTE}. Another details the improvements effected upon the Java^{TM} platform by re-implementation as VP code. The document also supplies guidelines on using int**e**nt^{JTE}, and running applications that have been translated to the Java^{TM} language from VP. *Using Java^{TM} Technology with intent* provides no detailed introduction to int**e**nt. The reader is advised to consult the *Tool Programming Guide(VP)*, the *Object Based Programming Guide*, the *VP Reference Manual*, the *Sysgen Reference Manual* and the *Reference Manual for the Kernel* before reading this document.

4   VP programmer

4.1   Tool Programming Guide (VP)

This document is designed for the programmer who understands the basic concepts behind int**e**nt, and is starting to learn how to program in Virtual Processor Code. The document gives details of the syntax and structure of the source code of a VP tool, and an overview of the 'virtual processor.' Example tools and example programs are included. The guide also provides information on the use of library functions, VP instructions and macros.

4.2   Object Based Programming Guide

This guide is designed for programmers learning to program using Virtual Processor Code in an object based manner. It would be advisable to read the *Tool Programming Guide (VP)* before this document, which provides information upon how to use facets of ordinary VP tool programming in an object based fashion. *The Object Based Programming Guide* offers guidelines on defining classes, and on allocating and initialising class instances.

4.3   VP Quick Reference

Whereas the VP Reference Manual provides a comprehensive description of the VP language, this brief document contains a simple quick reference sheet on the same subject.

4.4   VP Reference Manual

This manual can be used in conjunction with the *Tool Programming Guide*. The two documents have some overlap, in that both discuss the structure of the Virtual Processor. However, while the *Tool Programming Guide* is designed to coach a programmer through the process of learning to program

Introduction to Tao Documentation

Elate tools in VP, the *VP Reference Guide* is a reference source of details on the VP programming language. The VP Reference Manual includes an alphabetical list of all VP instructions.

5   C Programmer

5.1 The C Tool Programming Guide

This document is designed for the programmer who understands the basic concepts behind Elate, and is intending to write tools in C, which will then be compiled using the Elate C Compiler. The document provides guidelines upon building and using tools in C. The guide also supplies an architectural overview of the Elate Native C Compiler, describes its use in compiling an example program. In addition, it discusses aspects of the Elate native port of the GNU C compiler that may be required by users of the compiler, particularly with respect to the interfacing of assembler code with the output of the compiler. An appendix for the aid of programmers writing in C++ is attached.

6   System Programmer

6.1 System Programming Guide

This document aims to provide the programmer with an overview of the int**e**nt and Elate architecture as well as fundamental information about its various sub-systems. The guide intends to introduce the programmer to key concepts and provide a cookbook of initial ideas. The programmer can then build on this and improve his knowledge of Tao's technology by consulting the appropriate reference manual for further in-depth information. This guide assumes that the programmer has worked through the VP Tool Programming Guide and the Object Based Programming Guide and is familiar with the basics of assembling and running VP tools.

6.2 The Reference Manual for the int**e**nt Kernel

The int**e**nt kernel is in effect a collection of functions that may be subdivided in terms of the facilities that they provide for the system. *The int**e**nt Kernel Reference Manual* dedicates a section to each of these 'groups' of functions. Each section describes the API of the relevant functions, and provides an overview of the way in which the kernel interacts with the rest of the system. Some internal kernel functions are not described in this document. Unlike some other operating systems, the Elate kernel does not contain the system's device drivers, which are categorised and documented separately. The kernel functions covered within this document include those dedicated to tool handling, process management, process scheduling, process synchronisation, memory management, signal management, callback manipulation, the atomic adjustment of linked lists, entropy collection, timer management, and the manipulation of the atom table, the named data area table and the device table. This document presupposes a level of familiarity with the VP language, and should be read after the VP *Tool Programming Guide* and the *VP Reference Manual.*

6.3 Reference Manual for the Sysbuild Utility

At design time, int**e**nt uses a number of mechanisms to ensure that the system image from which the application is to be run contains the minimum of tools which are needed to successfully run the application. For the creation of such system images, int**e**nt provides two image generation utilities, the System Generation Utility (Sysgen), and sysbuild, which draws upon the underlying functionality provided by sysgen. This document outlines the basic instructions for using the sysbuild utility, with appropriate examples.

6.4 Sysgen Reference Manual

The *Sysgen Reference Manual* is intended for those programmers who need to use the System Generation Utility (Sysgen) to generate a bootable system image from a collection of tools and a specification of the user's requirements. Sysgen takes an instruction file, produced through analysis of the references made by the applications and device drivers to the kernel, library and user tools, and uses the instructions contained within that file to make one or more system images. The reference manual provides an overview of Sysgen's functionality, including a list of command line options, and a detailed syntactic description of each instruction that may feature in an 'instruction file.' An account of the system boot sequence is also included, as are a description of the behaviour of the Sysgen pre-processor, and details of NDAs which have special significance in the images generated by sysgen. It is recommended that the reader refers to *the Tool Programming Guide (VP)* and the *Reference Manual for the intent Kernel* before reading this document.

7 Device Driver Programmer

7.1 The Elate Device Driver Design Guide

This guide is designed to provide a programmer with all the information necessary for writing a device driver. Since within Elate device drivers are objects, it is advisable to read the *Object Based Programming Guide* prior to reading this document. A reader who wishes to write device drivers in VP code should also be familiar with after the *Tool Programming Guide* and the *VP Reference Manual*. A programmer intending to write device drivers in C or C++ is advised to refer to the *C Tool Programming Guide* before reading the *Device Driver Design Guide*. This guide describes the architecture of a device driver, and reveals how a device object can be created, called from applications, loaded from memory, or utilised across a network. It also includes the API specifications for methods used by device driver objects.

8 Multimedia Programmer

8.1 The intent Media Libraries Programming Guide

This document is designed for those who need to program within the intent Multimedia Toolkit. The guide includes an account of the different types of Audio Visual Object (AVO), the fundamental building block of the media libraries. It also provides guidelines for programming for the toolkit, and outlines the essential components of an intent multimedia application. It is recommended that before reading this document, the reader familiarise themselves with the VP language and the use of objects in Elate and intent, by reading documentation such as the *Tool Programming Guide(VP.*

9 General

9.1 The intent Shell User Guide

The *Shell User Guide* is a user-oriented document that provides an introduction to the shell. Only a few of the more commonly used shell commands are listed, and the user guide can be seen as complementing the *Shell Reference Guide*, which provides a complete list of the shell commands. The user guide gives a practical introduction to several facets of the shell, such as the reconfiguration of shell interaction, running shell scripts, generating filenames, and multi-tasking functions. It also provides a general reference guide to the grammar of the shell programming language.

9.2 Reference Manual for the intent Shell and Shell Commands

This reference document lists the commands available for the shell. Since this list is prefaced by only a brief introduction to shell command conventions and the use of options, this document should be

used in conjunction with the *Shell User Guide*. The commands are listed in alphabetical order, each command supplied with a synopsis, a description and an example to illustrate its functionality.

10 Using the Developer Documentation

A huge quantity of information is available to the developer as part of the standard int**e**nt release. The documentation consists of a large number of HTML files. Generally, the files have the name api.html or user.html. Each file being located in the appropriate directory for the part of the system it is describing. A convenient way of accessing this information is to simply create desktop shortcuts to *contents.html* and *index.html*. You can then access the documentation using your favourite web browser.

The documentation has an automatic indexing system designed by Tao. This allows the programmer to use a program called 'help'. Help scans through indexes of the HTML documents looking for certain tagged keywords. An example of its use is:

```
$ help printf
```

Help will then give you a selection of pages that have this keyword tagged and you can select the most suitable. It will then display the relevant section of the requested HTML file. This facility is quite useful for the programmer who knows the name of a particular tool but wishes to check the exact inputs and outputs required.

A simple HTML viewer is also available. This allows you to view any HTML file from the int**e**nt shell. To use the viewer simply type html <filename.html> e.g. $ *html dev/api.html*

11 Summary list of documentation

11.1 Manuals

- Introduction to Elate Documentation

- Getting Started with Elate on Windows
- Getting Started with Elate on Linux
- Introduction to Tao's Java Technology Solution
- Using Java technology with int**e**nt
- int**e**nt Media Libraries Programming Guide
- VP Tool Programming Guide
- VP Quick Reference
- Object Based Programming Guide
- VP Reference Manual
- C Tool Programming Guide
- System Programming Guide
- Elate Kernel Reference Manual
- Sysgen Reference Guide
- Sysbuild Reference Manual
- Debugger User Guide
- Shell User Guide
- Shell Reference Manual
- Technical Glossary

Please note that the material previously contained within the 'Porting Elate' document is now located within the System Programming Guide.

Introduction to Tao Documentation