

# Table of Contents

## Chapter 1

### Overview

1.1	About this document	1-1
1.1.1	Who should read this document?	1-1
1.1.2	How to use this document	1-1
1.1.3	Terminology	1-2
1.2	Other sources of information.	1-3

## Chapter 2

### The Arrange data model

2.1	Document building blocks	2-1
2.2	Primitive data types	2-2
2.2.1	arNoteID	2-3
2.3	Selections	2-3
2.4	Current document	2-4

## Chapter 3

### The Arrange API

3.1	Overview	3-1
3.2	Hooks	3-1
3.2.1	Module parameter block	3-2
3.3	Callbacks	3-3
3.3.1	Conventions	3-3
3.3.2	Data Callbacks	3-4
3.3.3	NoteCalls	3-8
3.3.4	SelCalls	3-11
3.3.5	SearchCalls	3-14
3.3.6	SysObjCalls	3-15
3.3.6.1	Topics, folders, and views	3-15
3.3.6.2	Type definitions	3-17
3.3.6.3	Field definitions	3-17
3.3.6.4	Windows	3-18
3.3.7	ListCalls	3-19
3.3.8	ViewCalls	3-23
3.3.9	DocCalls	3-23
3.3.10	IOCalls	3-26
3.3.11	UICalls	3-27
3.3.12	DialogCalls	3-31
3.3.13	ModMgrCalls	3-34
3.3.14	MemCalls	3-35
3.3.15	UtilCalls	3-36

## Chapter 4

### What is a plug-in?

4.1	Overview	4-1
4.2	The plug-in operating environment	4-1
4.2.1	Arrange startup	4-1
4.2.2	Arrange exit	4-2
4.2.3	Toolbox calls	4-2
4.2.4	Memory management	4-3
4.3	Plugin file structure	4-4

## Chapter 5

### Writing a plug-in

5.1	Creating a new plug-in	5-1
5.1.1	Defining your plug-in	5-1
5.1.2	Choosing a sample file	5-2
5.1.3	Writing a plug-in	5-2
5.1.4	Building a plug-in	5-3
5.1.4.1	Building under MPW C++	5-3
5.1.5	Testing and debugging	5-4
5.2	Naming and registering your plug-in with Common Knowledge	5-4

## Chapter 6

### Debugging a plug-in

6.1	Debugging standalone code	6-1
6.1.1	Debugging using Macsbug	6-1
6.1.2	Debugging using Jasik's Debugger	6-1
6.2	Special debugging features in Arrange	6-2
6.2.1	Callback error checking	6-2
6.3	The Arrange debugging window	6-2
6.3.1	Logging to the debug window	6-3
6.3.2	Debug window commands	6-3