

LIST OF TABLES

Table 1-1.	Distribution Concerns and Solutions.....	10
Table 4-1.	Micro-benchmark: overhead of method calls with one extra argument.....	107
Table 4-2.	Macro-benchmarks: cost of a universal extra argument.....	109
Table 4-3.	Overhead of Maintaining Thread Equivalence Classes.....	111
Table 4-4.	J-Orchestra worst-case indirection overhead as a function of average work per method call (a billion calls total).....	127
Table 4-5.	Effect of lazy remote object creation (local-only objects) and J- Orchestra indirection.....	129
Table 5-1.	Baseline 1—Local Execution (processing overhead) on both the fast (750MHz) and the slow (440MHz) machine.....	153
Table 5-2.	Baseline 2—RMI Execution, without Restore (one-way traffic).....	153
Table 5-3.	Baseline 3—RMI Execution with Restore on local machine (no network overhead).....	153
Table 5-4.	RMI Execution with Restore (two-way traffic).....	153
Table 5-5.	NRMI (Call-by-copy-restore). Both the portable and optimized version shown for JDK 1.4.....	154
Table 5-6.	Call-by-Reference with Remote References (RMI).....	154
Table 5-7.	Software Complexity Metrics.....	175
Table 6-1.	Type-based analysis of used system classes.....	199