

Preface

This book combines two emergent research areas of the computer graphics world. On the one hand, there is software visualization. After 30 years, it remains an under-achiever with enormous potential to impact how we develop software. This book looks at how to visualize dynamic program behavior, while it is happening. The techniques are readily applicable to other areas of software visualization. Our ultimate goal is to make visible many practical aspects of program behavior that are currently invisible or difficult to see at best, such as how to find bugs and performance bottlenecks.

A second major focus of this book is on virtual environments. Lots of programmers want to create them, but for most that challenge is just *too* hard; virtual environments integrate advanced 3D graphics, animation and networking in ways that most ordinary developers can't manage in a practical time frame. We are writing this book to do what we can to help conquer that obstacle.

It is obvious that these two genres of computer graphics should be combined. The networking and persistence afforded by virtual environments are exactly what software visualization needs in order to become more collaborative, shared, ubiquitous, and educational. All we have to do is figure out how to build it.

Unicon is an innovative, very high-level programming language hosted at unicon.org and the popular open source site SourceForge. It is descended from Icon, a language developed at the University of Arizona as a successor to the SNOBOL language family from AT&T. Unicon's 3D and networking facilities turn out to be great for virtual environments, and its execution monitoring facilities make it second to none in the area of software visualization. Programs in other languages can often be instrumented to extract similar program behavior information, but in Unicon 120 or so kinds of program execution behavior are continuously available, any time they are of interest.

This book is organized into two parts. Part I is an overview of execution monitoring and program visualization. It presents Unicon's monitor architecture and the framework for monitoring Unicon programs using a series of example visualization tools that observe many kinds of execution. Part II presents virtual environments, including 3D modeling and network communication necessary for multi-user collaboration. It introduces methods of propagating visualization

information and collaborating within a multi-user virtual environment. The ultimate goal is to produce game-like real-time graphical ways of understanding and talking about bugs, bottlenecks, and other aspects of program behavior. Following Part II is a collection of appendices including some detailed program examples and a description of the implementation of the monitoring framework.

Writing Virtual Environments for Software Visualization

Jeffery, C.; Al-Gharaibeh, J.

2015, XVII, 155 p. 65 illus., 24 illus. in color., Hardcover

ISBN: 978-1-4614-1754-5