In Silico: 3D Animation and Simulation of Cell Biology with Maya and MEL

In Silico: 3D Animation and Simulation of Cell Biology with Maya and MEL

Jason Sharpe
AXS Biomedical Animation Studio

Charles John Lumsden
University of Toronto

Nicholas Woolridge
University of Toronto

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO
Morgan Kaufmann Publishers is an imprint of Elsevier

MORGAN KAUFMANN PUBLISHERS

Acquisitions Editor: Tiffany Gasbarrini
Publishing Services Manager: George Morrison
Project Manager: Mónica González de Mendoza
Assistant Editor: Matt Cater
Cover Design: Jason Sharpe / Alisa Andreola
Cover Illustration: Jason Sharpe

Morgan Kaufmann Publishers is an imprint of Elsevier.
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA

This book is printed on acid-free paper.

Library of Congress Cataloging-in-Publication Data
Sharpe, Jason.
In Silico: 3D Animation and Simulation of Cell Biology with Maya and MEL / Jason Sharpe, Charles John Lumsden, Nicholas Woolridge.
p. ; cm.
Includes bibliographical references and index.
ISBN-13: 978-0-12-373655-0 (pbk. : alk. paper) 1. Cytology—Computer simulation. 2. Maya (Computer file) 3. Computer animation. 4. Computer graphics. 5. Three-dimensional display systems. I. Lumsden, Charles J., 1949– II. Woolridge, Nicholas. III. Title. IV. Title: Cell biology art and science with Maya and MEL.
[DNLM: 1. Cells—Programmed Instruction. 2. Computational Biology—Programmed Instruction. 3. Models, Biological—Programmed Instruction. 4. Motion Pictures as Topic—Programmed Instruction. 5. Programming Languages—Programmed Instruction. QU 18.2 S532s 2008]
QH585.5.D38S53 2008
571.601′13—dc22

2007053013

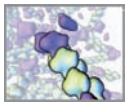
ISBN: 978-0-12-373655-0

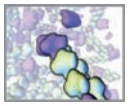
For information on all Morgan Kaufmann publications,
visit our Web site at www.mkp.com or www.books.elsevier.com

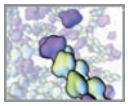
08 09 10 11 12 13 10 9 8 7 6 5 4 3 2 1

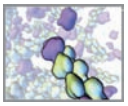
Printed in China

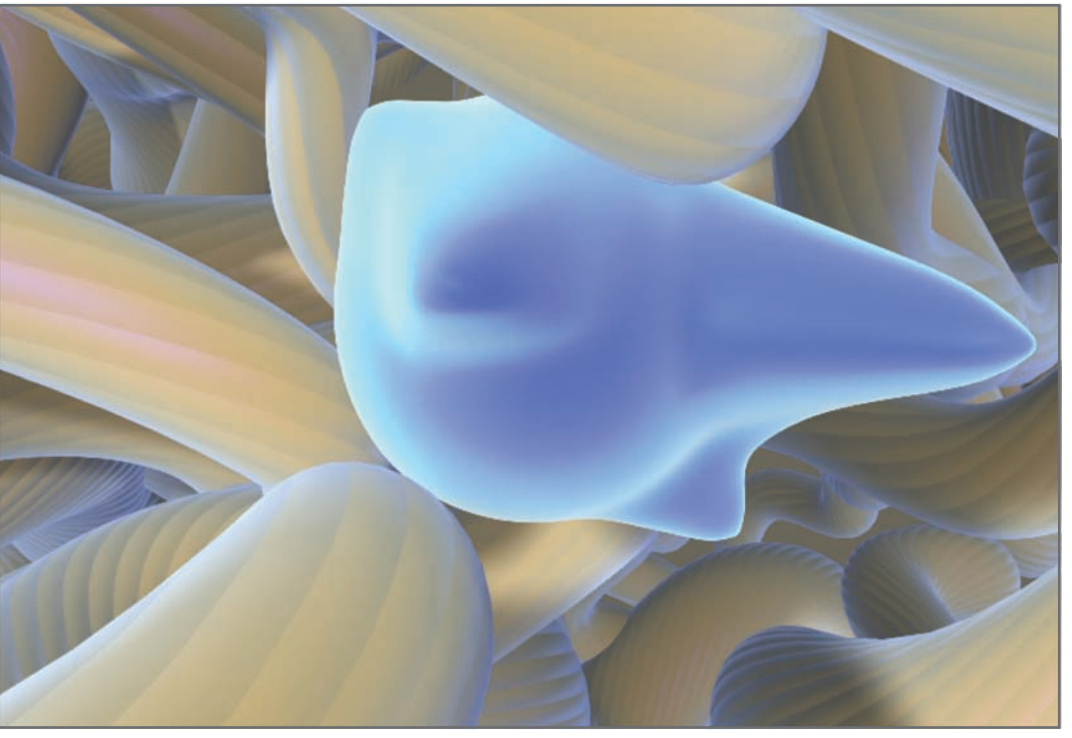
CONTENTS





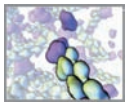






Still image from a Maya simulation model of cell migration in a 3D scaffold. The cell extends protrusions in search of scaffold fibers. When it contacts a fiber, the protrusion adheres to it. The cell body then contracts, pulling it in the direction of the adhesion. Maya's extensive 3D modeling toolset and programming capabilities make it well suited to 3D visual simulations of biological phenomena such as cell migration.

Preface



Who is this book for?

If, like us, you are involved with the study of cells and cell biology, or if your work takes inspiration from the organic world, this book is for you. We have written *In Silico* for the diverse creative community—scientists, artists, media designers, students, and hobbyists—now deeply involved with the living cell as a key to unlocking the complexity of organic matter and a gateway to powerful new understanding of disease. In the scientific area, cell and molecular biologists and their research partners today have little time to spare developing complex computer programs from the ground up. High-end three-dimensional (3D) computer programs like Autodesk Maya provide the busy scientist with a robust, flexible development environment in which state-of-the-art computer methods can be used to analyze, model, and visualize cell data. Equipped with deeply customizable user and application programming interfaces, Maya and other top-tier 3D animation programs afford rapid prototyping of data analysis and models through advanced graphics, physics, and rendering systems. Output capability embraces both crisp numerical data and polished 3D dynamic visualizations of cell physiology. These tools have enough programming flexibility that the working researcher can concentrate on the functional aspects of the data mapping or simulation capability they wish to create.

In the communications field are individuals and groups immersed in the burgeoning marketplace of biocommunications, especially medical and scientific animation. The telling of stories is a human universal, common to all peoples and cultures. The increasingly complex world enabled by science and technology makes the accurate, compelling telling of scientific stories more important than ever. Constantly, animators of medical and scientific subjects are called on to present ever more intricate, unusual phenomena involved in understanding how cells work and what goes wrong with them to cause devastating illnesses like cancer and heart disease. At the same time, the expectations of a media-savvy public for concise, truthful, entertaining visual stories rise even higher. Taking control of a program like Maya can empower the media artist to better interpret and visualize wonderfully intricate cellular phenomena—such as the crowded molecular landscapes of the cell interior, the cell waves coursing through the embryo's interior, or the skein of blood vessels healing a wound—that would be impractically tedious or impossible to animate by hand.

And too numerous to count, surely, are the artists and citizens everywhere who draw inspiration from biology and the natural world, and who dream of imparting some facet of organic vitality and complexity to their creative work or personal appreciation of nature. The ideas and methods of this book will, we believe, inform and inspire everyone with such interests. Although the focus of our applications is the exciting realm of the living cell, those whose interests embrace other parts of living nature will find the knowledge and techniques they learn here of useful in many different ways.

Why Maya?

Although Maya is a top-tier product used worldwide for 3D animation in entertainment, gaming, and manufacturing, this Academy Award® winning program does not stand alone in representing the cutting edge of high-end 3D. Superb tools such as SoftImage XSI, Maxon Cinema 4D, NewTek LightWave 3D, Autodesk 3ds Max, and

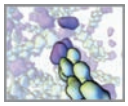
Side Effects Software's Houdini, stand alongside Maya to define the state of the art in 3D animation capability. Maya is our subject in this book for three reasons. First, despite the excellence of alternative tools Maya currently enjoys a pre-eminent status in top-end 3D animation work. Second, the Maya programming interfaces—accessed through a C++ application toolset (the API—which we plan to deal with in a subsequent book), via scripting in the Python language, and through Maya's own scripting language MEL, which we treat in this book—allow enormous power and flexibility in customizing Maya for scientific applications. Third, the academic outreach initiatives supported by Autodesk, the firm that makes and sells Maya, have enabled us to test Maya and some of its predecessors (such as Alias PowerAnimator) in demanding real-world science projects in cell and medical science. As a threesome, we have between us accumulated roughly 40 person-years of experience across a wide range of such applications. We find Maya worthy of close attention whenever there is a need to model and visualize 3D cell biology using a computer. Since our origins trace back to the early days, in which such computer methods were lab-written custom jobs in languages like Fortran, C++, and OpenGL, Maya for us means shorter time to software completion while increasing the power of the animated visualization.

If you are already a user of a 3D animation package other than Maya, you will still find considerable useful material in the pages to follow. The book is going to show you how to approach complex biological problems effectively, by means of a workflow in 3D visual computing. We have developed this workflow over the years of our medical and biocommunications research and use it daily in our teaching and scientific investigation. By working through the book's projects and case studies, you will be able to adapt our workflow to other 3D animation products as well as take them much further in Maya itself.

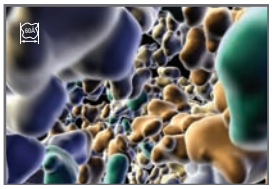
What the book offers

In the world of computer graphics software, Maya is a relatively complicated application. Learning and, eventually, some degree of genuine mastery, take time, but don't despair. Page by page, the learning map we have set up will take you from one productive result to the next. You will deal throughout with learning content that has genuine interest and significance in the world of science and cell biology. In *Part 1* you will meet the key ideas and terms from scientific computer graphics needed to dive into Maya while assessing its historic relevance to leading edge visualization. In *Part 2*, you will receive a self-contained introduction to Maya and to our workflow that will take you from starting the program through to a polished animation rendering of a complex protein. With this foundation you are ready to meet MEL, the programming language by which you will harness Maya's ability to model and render complex events. Then in *Part 3*, we put this all to work. You will develop a portfolio of case studies ranging from the single biological molecule to populations of interacting macromolecules, and then on to mobile cells as they move through their tissue environment. As you complete each element in the portfolio, you will have taken command of powerful new strategies for using MEL to control Maya's numerical and visual rendering activity.

Here's what you can expect in the rest of the book.

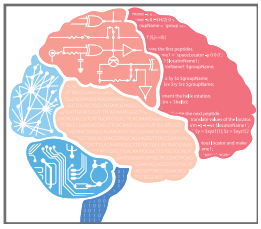


Part 1: Setting the stage



01 Introduction

To get started, we attempt to answer the question: "Why visualize?" We briefly discuss the power of visual perception in human learning and discovery, and how we can leverage our innate visual intelligence to advance understanding in science. The role of structural hierarchy in biology is explored, and we take this opportunity to introduce some of the "major players" at the levels of molecules, cells, and tissues. Maya is introduced, and some of its history traced. Finally, we celebrate the advances in 3D computer animation that have provided powerful, yet affordable tools for conducting visual explorations of complex systems.



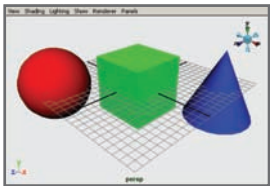
02 Computers and the organism

This chapter will survey the basic idea of computation and how it should be done automatically, by a machine. We will see to that a core tenet of information processing, *conditional control*, is used by both computer programs and living organisms to regulate activity. This sets the stage for understanding how computer programs can illuminate the structures and functions of biological systems.

03 Animating biology

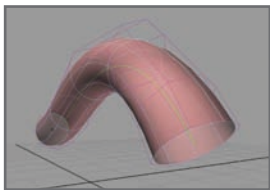
In this chapter, you'll explore the standard animation workflow, and see how it can be adapted to the needs of a biomedical researcher or animator. We examine the preproduction process, where a story is developed and refined, and a plan for the execution of the film is made. In the production phase, the hard work of building, texturing, animating, and rendering of the story elements takes place. In postproduction, the media developed in production are composited, edited, and packaged for delivery. These steps are applicable to most science communication contexts, and we propose a modified version of them to accommodate the unique requirements of biological systems visualization.

Part 2: A foundation in Maya



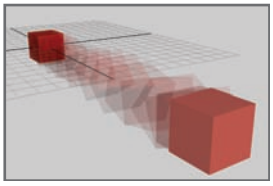
04 Maya basics

This chapter will get you immediately familiar with Maya, via a tour of the primary features of the user interface (UI). You'll learn about Maya's program architecture—the proprietary Dependency Graph and Scene Hierarchy—and get a sense of what's actually happening when you start pressing Maya's buttons. A basic understanding of "Maya behind the scenes" will greatly extend what you can accomplish with the software. We'll continue to develop this understanding in the subsequent chapters.



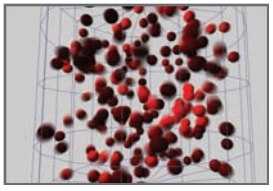
05 Modeling geometry

In this chapter you will learn to make geometric models. A discussion of different model types and their components gives an understanding of how complex surfaces are created from relatively simple beginnings. You'll also see how models are composed of nodes and attributes—the stuff of Maya's Dependency Graph—via practical examples.



06 Animation

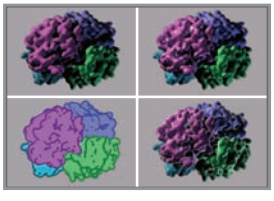
With animation, you'll bring your models to life. In Maya, to animate is to change some attribute over time—be it position, color, or speed, for example. You will see this definition applied as you learn to work with the tools of animation—keyframes and animation curves—to make objects move around and change shape. You'll wrap up the chapter with your first procedural—or algorithm-driven—animation, and a taste or what's possible when you set aside the standard UI animation tools and begin using written expressions to simulate motion.



07 Dynamics

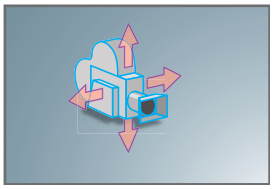
One of the truly powerful features of Maya is that it's a sophisticated, built-in dynamics engine that you can use to simulate real-world physics. It calculates forces and collision dynamics for soft- and rigid-bodied objects and for entities called particles. In this chapter you will create animations driven entirely by Maya Dynamics, in which objects are moved about by forces and collide with one another. These ready-made physics simulation capabilities are a boon not only to visual effects artists looking to emulate real-world phenomena, but also to the computational biologist looking to breadboard dynamic modeling scenarios before going through the effort and expense of building a custom physics engine.

With Maya, you have at your fingertips the same tools for rendering proteins, cells, and tissues that professional CGI artists use to create the stunning imagery that has revolutionized Hollywood visual effects. In each of the following four chapters, you'll focus on an aspect of Maya's extensive rendering capabilities. Together these chapters will take you through the process of preparing an animated scene (showing the four subunits of the blood protein hemoglobin) for rendered output.



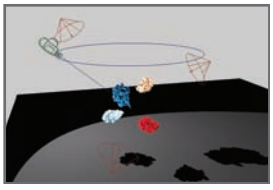
08 Shading

In this, the first chapter on the rendering process, you'll learn how to make and apply shading networks, or *shaders* for short. Shaders work with the lights in a scene to determine the appearance—color, texture, opacity, etc.—of objects in your finished renderings. You'll learn how to quickly create and apply shaders to multiple objects in preparation for rendering.



09 Cameras

Like a real movie camera, a Maya camera defines what your audience will see. Many features are available with a real camera are embodied in the Maya version, allowing you to set up and record shots in virtual 3D space much as you would in the real world. The Maya camera also defines your view of the 3D scene as you work with it, and is therefore an indispensable tool, whether or not you plan to make finished (rendered) movies with Maya. By the end of this chapter, you'll know how to set up and animate a camera along a track called a motion path—much the way a movie camera is set up on a track to move as it records the action.

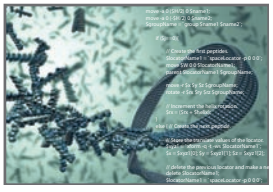


10 Lighting

If the camera is a cinematographer's *brush*, then light is the paint. Just like in the real world, light defines what is visible in your Maya scenes, and the quality of its appearance. We'll show you how to achieve professional illumination with minimal effort in order to get the most out of your images.

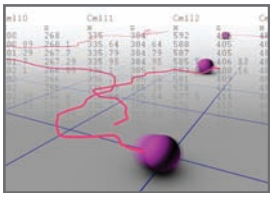
11 Action! Maya rendering

In this final chapter on the rendering process, you'll see how Maya integrates shaders, camera view, and lights to produce one or more image files. We'll explore the different render "engines" available in Maya and their relative advantages.



12 Mel scripting

At this point in the book, you'll know your way around the UI and be familiar with the concepts and terminology involved in modeling, animating, and rendering in Maya. You'll be ready to depart somewhat from the standard UI tools and start exploring Maya's scripting capabilities. This chapter introduces Maya's scripting (or programming) language, MEL (short for Maya Embedded Language). You'll learn how to run individual MEL commands and how to compose a script—or short computer program—out of multiple MEL statements in order to automate tasks in Maya. Readers new to computer programming will learn the basic concepts—syntax, variables, operators, flow control, etc.—in the context of MEL. Those with previous programming experience can scan the chapter to pick up the MEL basics. In either case, plentiful examples and a short tutorial will have you coding Maya tasks using MEL in no time.



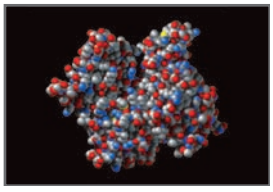
13 Data input/output

Ready-made software plug-ins are available for porting some of the more common 3D data formats to and from Maya. However, if you're working with a format for which no plug-in exists, such as experimental data formatted in a spread sheet, you may want to create your own importer

or exporter. This chapter shows you how to do just that using a suite of MEL commands for reading and writing external files. You'll also learn the MEL commands useful for formatting the text that you read and write. In the chapter's tutorial, you'll extract 3D coordinates from a cell migration data file, use them to visualize the moving cells, and then save out a report summarizing key migration statistics.

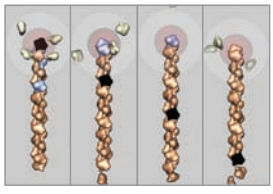
Part 3: Biology in silico—Maya in action

In this part of the book, you'll explore and use a workflow for in silico modeling and simulation that builds on your knowledge of Maya's UI and scripting capabilities. We present five tutorial-style projects, each dealing with a different level of biological organization—from a single protein up to a population of cells in a tissue matrix. In each project we'll guide you, step by step, through the composition of custom MEL scripts that automate the model building and/or dynamic simulation. Whether you're a scientist looking to explore Maya techniques in 3D computation or an artist visualizing topics in cell science, you'll learn a range of useful techniques that can subsequently be applied to your own projects.



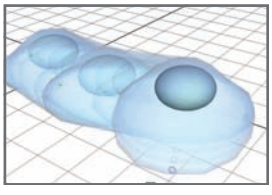
14 Building a protein

The ability to work with molecular models is essential to any 3D in silico approach to cell (and molecular) biology. To begin, one must first be able to build models using structural data. Once built, these models can be used to study and simulate a range of phenomena from protein folding to shape complementarity. In this chapter, you'll build a custom script to make a protein model using an external Protein Data Bank (PDB) file. You'll be able to use this script to make models from other PDB files and revise it to suit other data formats. Moreover, the chapter doesn't end when your model is built: we'll guide you through setting up and rendering a finished picture worthy of a book cover or wall poster.



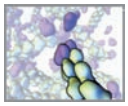
15 Self-assembly

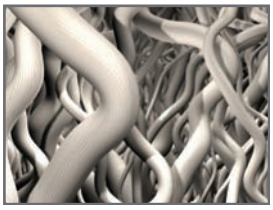
The self-assembly of macromolecular structures is key to the organization and function of cells and tissues. In this chapter you'll create a dynamic model of regulated self-assembly featuring an actin protein filament. You'll do this with custom MEL scripts that emulate molecular diffusion and chemical reaction dynamics.



16 Modeling a mobile cell

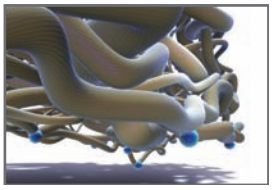
The study of mobile cells spans a huge range of biomedical research, from the spread of cancer to tissue regeneration. In this chapter you will create a simple cell model in Maya and make it crawl in response to a simulated chemical stimulus. By setting up parameters that control the cell's motion, including the degree to which it responds to the stimulus, you'll see how such a model could be extended to simulate and predict different modes of cell behavior.





17 Modeling an ECM scaffold

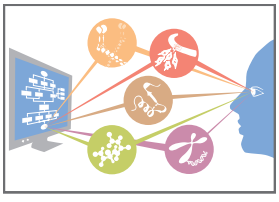
In the body, cells live in complex 3D environments of the various tissue types. Research in regenerative medicine is increasingly focused on the relationships between cells and their surroundings, with a growing awareness that 3D tissue architecture plays a key role in cell behavior. In this project you'll use our in silico workflow to build a fibrous tissue matrix. A set of model parameters will let you vary the structure of each matrix you create. You'll see that, given a set of model criteria, you can leverage MEL to create structures of a complexity that would be impractical to attempt using the standard modeling tools available through Maya's UI.



18 Scaffold invasions

In this, the final project of the book, you'll model the penetration of your tissue matrix by a mobile group of cells—using only MEL and some custom methods we developed for mapping 2D cell motion onto 3D surfaces.

In no way does this chapter represent the limit of what's possible for modeling cell biology in Maya. On the contrary, we have only scratched the surface! We hope that this and the projects before it will inspire you to create new developments in this exciting field of 3D in silico biology.



19 Conclusion

In this chapter we revisit the themes and methods covered in the book and look ahead to the future of biocommunications and computational cell science.

Further reading

We tour the cell biology, 3D visual computing, and Maya tools and techniques in sufficient detail to advance you quickly and efficiently through each chapter in the book. Nonetheless, practical constraints have made it necessary to be brief in our treatment of many of the subjects. Where you desire more information, we encourage you to explore the *Further reading* we've listed according to topic.

Glossary

This book was written for artists and scientists alike. Depending on your field of work or study, you may encounter terminology and concepts that are new to you. In the *Glossary*, we've compiled many of the key terms used throughout the book. They are listed with references to the pages on which they're used.

CD-ROM and companion Website

Everything you need to work through the examples, tutorials, and projects—background information, step-by-step instructions, and MEL code listings—is provided on the printed pages. In addition, we've enclosed a CD-ROM with supplemen-

tary material. It includes MEL scripts, Maya files, and rendered animations from various chapters. The read_me.txt file in the root directory of the CD-ROM includes an index of the enclosed computer files.

On the books's companion Website you'll find updates and corrections (when necessary) to the files provided on the CD-ROM.

www.insilico.book.net.

Computer hardware and software

The Maya files and MEL scripts listed in this book and included on the CD-ROM were created and tested on a mid-range consumer-level PC with the following specifications:

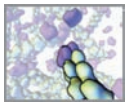
Software	Maya 8.5 for Windows
OS	Windows XP Professional 2002 (Service Pack 2)
PC	Dell Dimension 8300
CPU	Pentium 4, 3.20 GHz
RAM	1 GB
Graphics adapter	ATI Radeon 9800 XT, 256 MB DDR

The book's tutorials and projects have been developed over a number of versions of Maya, both in Windows and Mac OS. They have been **tested to work in Maya 8.5 for Windows**. Users of older versions of Maya may have to look around for commands whose names have changed, but the MEL code will probably work largely unaltered. As this book went to press, a new version was announced (Maya 2008). Although we have not had the opportunity to test our projects against Maya 2008, we have no reason to believe that the techniques we rely on would have altered enough to have broken them.

Similarly, the instructions for accessing Maya menus and tools, along with references to the Maya Help Library, are specific to Maya 8.5 for Windows. With a little adaptation they can readily be applied to learning Maya in other environments, namely Mac OS and Linux.

If you are considering purchasing Maya, we strongly recommend you ensure its compatibility with your hardware and software configuration by consulting the *system requirements* and *qualified hardware* specifications available via Autodesk's website:

www.autodesk.com/fo-products-maya



About the authors

Jason Sharpe is a cofounder of the award-winning AXS Biomedical Animation Studio in Toronto. Trained in mechanical engineering at Queen's University, fine arts at Emily Carr Institute of Art and Design and biomedical communications at the University of Toronto, he has worked on a wide range of Maya-based 3D animation projects for research, education, and entertainment.

Charles J Lumsden is Professor of Medicine at the University of Toronto. Trained as a theoretical physicist, he studies the mathematical logic behind illnesses such as Alzheimer's disease and cancer. He and his students have explored and championed a variety of 3D graphics software as aids to biomedical discovery, including top-tier commercial tools such as Maya and MEL.

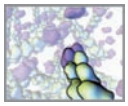
Nicholas Woolridge, Associate Professor of Biomedical Communications at the University of Toronto, has played a major role in the development of the visualization design field in the university's renowned Master's Degree in Biomedical Communications. His current research focuses on the optimization of visual media for medical research and education.

Acknowledgments

The splendid staff at Morgan Kaufmann, our publisher, has given us essential aid—mixed with clearheaded expertise and unquenched enthusiasm—as *In Silico* found its way through the press and into your hands. Tim Cox, then a senior editor at Morgan Kaufmann, saw sense in our idea that time was right for a richly cross-disciplinary book exploring Maya and its programming language, MEL, as tools for adventure and discovery in biology and medicine. Tim also got behind our conviction that such a book would be at its best if written for a use by a diverse audience of artists, scientists, and highly motivated private citizens. Morgan Kaufmann is a world leader in producing texts that map the subtle intricacies of MEL programming; we were, and remain, honored to have *In Silico* at home in this distinguished setting. Once Tim had the project launched, our Editor, Tiffany Gasbarrini, and Assistant Editors Michele Cronin and Matt Cater, helped us survive the twists and turns of bringing the book to life. Through our publisher we benefited from the comments of expert readers, who responded to drafts of *In Silico* either in whole or in part. Our thanks to these hard-working colleagues for their generous allotment of time and attention: Prof. Klaus Mueller of Stony Brook University; David F. Wiley, President and CEO of Stratovan Corporation; Azam Khan, research scientist at Autodesk Corporation; and five anonymous reviewers. Their input, uniformly deft and relevant, has helped *In Silico* complete its journey with enhanced strength.

In addition, two student reviewers—Lori Waters (of the Biomedical Communications graduate program) and Tatiana Lomasko (PhD candidate in the Institute of Medical Science), both at the University of Toronto—completed many of the tutorials, providing valuable feedback that helped us to hone our approach.

Throughout their history, Maya and MEL were invented and advanced by a community of brilliant computer graphics innovators principally located in Toronto, Canada (with colleagues in offices in Paris and California). The software was originally developed by Alias, Inc., and is now under the banner of the Autodesk Corporation. We cannot overstate our appreciation to Autodesk and to its staff of Maya and MEL experts in assisting us on occasional technical questions and allowing us to present the many illustrations in which Maya's user interface is depicted. As well, *In Silico* takes the view that influential inventions like Maya are what they are not only through the genius of their creators, but also because they appear at a specific time and place in human history. Therefore, appreciating historic trends in computer technology, computer programming, and 3D computer animation gives us better understanding of Maya and MEL. The history of Maya and MEL has not been written up extensively, and what sources exist we found to be occasional and widely scattered. We are therefore most grateful to Autodesk for granting us discussions with members of its staff, who number among the original inventors of Maya and MEL. These incredibly busy people answered our questions about origins and inspirations with patience, grace, and good humor. We are delighted to be able to incorporate the gist of those discussions here, by way of introducing you to the depths of Maya and MEL. In particular we must thank Joyce Janczyn, lead designer of MEL, as well Mike Taylor,



Duncan Brinsmead, and Jos Stam for talks that opened our eyes to the inner life Maya.

Ravi Jagannadhan gave considerably of his own time to review and test the many MEL scripts published here. And, during this entire time Azam Khan (research scientist at Autodesk) never tired of his informal role as our advisor and principal facilitator amidst the elite world of those charged with inventing the latest versions of Maya and Maya programming.

Since this book hopes to be useful to readers who are new to computers, computer programming, or 3D animation—as well as an efficient self-contained resource for experienced science researchers and computer artists—we have used key moments from computer history and animation history to lay newcomers a congenial path to MEL programming. It is a pleasure to thank all the computer historians, collectors, and archivists who helped us with information, recollections, and photographs. In particular we must note the extended assistance generously given our history frame by: portraitist Louis Fabian Bachrach III for his photograph of programming language pioneer John Backus, lead inventor of the Fortran language; computer scientist John Bennett (Sydney, Australia) for his assistance and support in presenting his early computer graphic of structure pattern data for the protein myoglobin; Deirdre Bryden, Queen's University (Kingston, Ontario) archivist, and Marnee Gamble, University of Toronto archivist, for mainframe history and photographs at these Canadian research centers; Martin Campbell-Kelly, University of Warwick (Coventry, UK), for early computer history and photographs, especially the EDSAC; Annette Faux, archivist at Cambridge University's Molecular Biology Laboratory, for early 3D models of the myoglobin protein; PDP-8 microcomputer collector and archivist David Gesswein, his wife Janet Walz, and their cats Khym and Py for the PDP-8 microcomputer photograph shot specially for the book; Calvin Gotlieb, University of Toronto, for access to his archives on that institution's computer center history; Bonnie Ludt, California Institute of Technology Archives, for her help with the Linus Pauling photographs; Dawn Stanford of the IBM Corporate Archives for assistance with IBM mainframe history; Peter Strickland, Managing Editor of the *Acta Crystallographica* journals, for his assistance with early computer visualizations of protein structure; Bjarne Stroustrup, inventor of the universally used C++ programming language, for his photograph; Marcia Tucker, Institute for Advanced Study Archives (Princeton, NJ) for assistance with the John von Neumann photograph; and Martin Zwick, Portland State University (Portland, OR), for information and photographs on key early work in molecular computer graphics. Our photo editor, Jane Affleck, also gave us strong assistance in sourcing hard-to-find images.

In Silico celebrates as well creative work by many of our colleagues who advance the visual interpretation of cell structure and dynamics through 3D computer graphics and animation. We especially thank Drew Barry, Marc Dryer, Stephen Ellis of Ellis Entertainment, David Goodsell, and Jenn Platt for letting us include their work here; Eddy Xuan and Sonya Amin of AXS Studio for their tremendous support and generous contributions to the book's illustrations; and Christina Jennings of Shaftesbury Films for letting us include animation stills from her pioneering dramatic series, *Regenesis*. Stunning visualizations in biology and medicine of course use technology other than computer graphics, such as photographic microscopy and video capture. We are indebted to: Peter Friedl and Katarina Wolf, University of Würzburg, Germany; Sylvia Papp and Michal Opas, University of Toronto; and Alexis Armour, Hôpital Hôtel-Dieu du CHUM, Université de Montréal, for their help and consent in

using their micrographs and/or video capture of cellular and tissue engineering materials. A special thanks goes to John Semple of Sunnybrook Health Sciences Centre for his expertise and guidance in regenerative medicine that helped shape the book's two final projects. John, who is both an artist and a scientist, also provided feedback at an early stage that helped us craft the book for researchers and artists alike.

This book would not exist without the support we received from NSERC, the Natural Sciences and Engineering Research Council of Canada, in the form of a three-year grant under NSERC's Collaborative Research and Development (CRD) program. The CRD program brings University-based researchers in Canada together with companies that share common interests in science and technology—in this case the idea that a top-tier 3D animation package like Maya (itself a Canadian invention) can be a powerful tool in the hands of biomedical scientists and teachers. The NSERC-CRD initiative seeks outcomes with broad relevance to the advanced training needs and research application requirements of citizens in Canada and indeed worldwide. It has therefore been a special pleasure to design our work under this grant program, through NSERC-CRD Grant Number CRD 270158-03, entitled "Interpretive Visualization: Understanding cell systems dynamics through computer animation"; so that our findings can communicated in a book for working professionals and for trainees in both the sciences and the digital media arts. Our corporate partners, Bell Canada Enterprises in grant year 1 via the Bell University Grants Program at the University of Toronto, and Alias (now Autodesk) in grant years 2 and 3, were essential to the success of our CRD project and we are deeply grateful for their participation. At each step NSERC personnel at various levels—Eileen Jessop, Pamela Moss, Anne-Marie Monteith, Sylvie Boucher, and Lise Desforges—assisted us with practical guidance and patient advice.

No book large or small gets done without evenings, late nights, and weekends nipped from time otherwise owed to kith and kin. So we must end with our deepest thanks to our families, who have put up with all the stolen hours and steadfastly supported us throughout the book's creation.