



DI-Guy Expressive Faces

Users Guide



VT MÄK
A company of VT Systems



DI-Guy Expressive Faces

Users Guide

Copyright © 2014 VT MÄK

All rights Reserved. Printed in the United States. No part of this document may be copied or reproduced in any form without prior written consent of VT MÄK.

VR-Exchange™, VR-TheWorld™, VR-Vantage™, DI-Guy™, and DI-Guy Scenario™ are trademarks of VT MÄK. MÄK Technologies®, VR-Forces®, RTIsby®, B-HAVE®, and VR-Link® are registered trademarks of VT MÄK.

3ds Max® is a registered trademark of Autodesk, Inc.

OpenGL is a registered trademark of Silicon Graphics, Inc.

Vega Prime and OpenFlight are registered trademarks of Presagis.

Visual C++ and Direct3D are registered trademarks of Microsoft Corp.

FLEXIm is the registered trademark of Flexera.

COLLADA is a trademark of Sony Computer Entertainment, Inc.

Certain models were provided by CG2, Presagis, Antycip, Engineering and Computer Simulations, Inc., and Viewpoint DataLabs International.

E-Town™ Building Library models are from CGSD Corporation, © 1999.

libjpeg - this software is based in part on the work of the Independent JPEG Group

libtiff © 1988-1997 Sam Leffler, Silicon Graphics, Inc. www.libtiff.org

Ogg Vorbis © 2002 Xiph.org Foundation www.xiph.org

OpenAL is a trademark of Creative Labs, Inc.

Perl - © 1989-1999, Larry Wall www.perl.org, www.activestate.com

Qt © 2005-2007 Trolltech ASA www.trolltech.com

zlib © 1995-2005 Jean-loup Gailly and Mark Adler www.zlib.net

pthread sourceware.org/pthreads-win32

Qscintilla © 2008 Riverbank Computing Limited www.riverbankcomputing.co.uk

FaceFX. ©2002-2013, OC3 Entertainment, Inc

All other trademarks are owned by their respective companies.

For third-party license information, please see "Third Party Licenses," on page xi.

VT MÄK
150 Cambridge Park Drive, 3rd Floor
Cambridge, MA 02140 USA

Voice: 617-876-8085

Fax: 617-876-9208

info@mak.com

www.mak.com

Revision DIG-13.0-4-140523



Contents

Preface

Documentation	v
Derivative Work	vi
MÄK Products	vi
How to Contact Us	ix
Document Conventions	x
DI-Guy Conventions	xi
Mouse Button Naming Conventions.....	xi
Third Party Licenses	xi
Boost License.....	xi
libXML and libICONV	xii
Lua	xii

Chapter 1. DI-Guy Expressive Faces

1.1. Introduction	1-2
1.2. How to Work with DI-Guy Expressive Faces	1-3
1.2.1. Create a Script	1-5
1.2.2. Select an Avatar	1-7
1.2.3. Prepare a Base Scenario	1-8
1.2.4. Record Voice Performances	1-9
1.2.5. Create Lip-Synching Animations in FaceFX Studio Professional	1-10
1.2.6. Create a Simple Local Path for Each Response (Animation) .	1-16
1.3. Creating a FaceFX Head Appearance	1-17
1.3.1. Creating and Exporting Data from 3ds Max	1-18
1.3.2. Setting Up and Publishing a New FaceFX Actor	1-20
1.3.3. Setting Up a Head for Use in DI-Guy Expressive Faces	1-20
1.4. Considerations for Using Expressive Faces	1-22

Index



Preface

This manual is written for persons who want to create head animations for use with the Expressive Faces module for DI-Guy. The manual assumes that you are familiar with your operating system and that you know how to work in your computer's graphical window environment. It also assumes that you are familiar with DI-Guy Scenario.

Please see release documentation for information about changes and updates to DI-Guy since this manual went to press.

Documentation

DI-Guy comes with the following documentation:

- ♦ *DI-Guy SDK Developers Guide*. Explains how to program using the DI-Guy SDK.
- ♦ *DI-Guy Scenario Users Guide*. Explains how to use DI-Guy Scenario to create scenarios with realistic human activity.
- ♦ *DI-Guy Content Reference Manual*. Explains how to customize DI-Guy characters and add new character models.
- ♦ *DI-Guy AI Users Guide*. Explains how to use DI-Guy AI to add intelligent behavior to DI-Guy characters.
- ♦ *DI-Guy Motion Editor Users Guide*. Explains how to import, edit, and customize DI-Guy motions.
- ♦ *DI-Guy Expressive Faces Users Guide*. Provides a brief introduction to using the Expressive Faces module.
- ♦ *DI-Guy Master Index*. A PDF master index to the PDF manuals.
- ♦ DI-Guy Master Index. An HTML page that links to all documentation.
- ♦ Class documentation. HTML documentation describing all classes and functions.

These documents are available from the start menu under DI-Guy, or by browsing to *./doc*. Of particular interest is the DI-Guy Documentation Master Index, which lets you quickly access all the documents above along with additional technical information about DI-Guy.

Derivative Work

Any DI-Guy content (model, texture, motion, and so on) that you modify or copy is a derivative work that is protected under copyright law by the same rules that apply to content delivered with DI-Guy. This derivative content may be used only on computers that have a valid DI-Guy license. Please contact VT MÄK to discuss licensing for other uses of derivative models.

MÄK Products

DI-Guy Expressive Faces is a member of the VT MÄK line of software products designed to streamline the process of developing and using networked simulated environments. The VT MÄK product line includes the following:

- **VR-Link® Network Toolkit.** VR-Link is an object-oriented library of C++ functions and definitions that implement the High Level Architecture (HLA) and the Distributed Interactive Simulation (DIS) protocol. VR-Link has built-in support for the RPR FOM and allows you to map to other FOMs. This library minimizes the time and effort required to build and maintain new HLA or DIS-compliant applications, and to integrate such compliance into existing applications.

VR-Link includes a set of sample debugging applications and their source code. The source code serves as an example of how to use the VR-Link Toolkit to write applications. The executables provide valuable debugging services such as generating a predictable stream of HLA or DIS messages, and displaying the contents of messages transmitted on the network.

- **MÄK RTI.** An RTI (Run-Time Infrastructure) is required to run applications using the High Level Architecture (HLA). The MÄK RTI is optimized for high performance. It has an API, RTIsPy®, that allows you to extend the RTI using plug-in modules. It also has a graphical user interface (the RTI Assistant) that helps users with configuration tasks and managing federates and federations.
- **VR-Forces®.** VR-Forces is a computer generated forces application and toolkit. It provides an application with a GUI, that gives you a 2D and 3D views of a simulated environment.

You can create and view local entities, aggregate them into hierarchical units, assign tasks, set state parameters, and create plans that have tasks, set statements, and conditional statements. VR-Forces also functions as a plan view display for viewing remote entities taking part in an exercise. Using the toolkit, you can extend the VR-Forces application or create your own application for use with another user interface.

- ♦ VR-Vantage™. VR-Vantage is a line of products designed to meet your simulation visualization needs. It includes three end-user applications (VR-Vantage Stealth, VR-Vantage PVD, and VR-Vantage IG) and the VR-Vantage Toolkit.
 - VR-Vantage Stealth displays a realistic, 3D view of your virtual world, a 2D plan view, and an exaggerated reality (XR) view. Together these views provide both situational awareness and the big picture of the simulated world. You can move your viewpoint to any location in the 3D world and can attach it to entities so that it moves as they do.
 - VR-Vantage IG is a configurable desktop image generator (IG) for out the window (OTW) scenes and remote camera views. It has most of the features of the Stealth, but is optimized for its IG function.
 - VR-Vantage PVD provides a 2D plan view display. It gives you the big picture of the simulated world.
 - The VR-Vantage Toolkit is a 3D visual application development toolkit. Use it to customize or extend MÄK's VR-Vantage applications, or to integrate VR-Vantage capabilities into your custom applications. VR-Vantage is built on top of OpenSceneGraph (OSG). The toolkit includes the OSG version used to build VR-Vantage.
- ♦ MÄK Data Logger. The Data Logger, also called the Logger, can record HLA and DIS exercises and play them back for after-action review. You can play a recorded file at speeds above or below normal and can quickly jump to areas of interest. The Logger has a GUI and a text interface. The Logger API allows you to extend the Logger using plug-in modules or embed the Logger into your own application. The Logger editing features let you merge, trim, and offset Logger recordings.
- ♦ VR-Exchange™. VR-Exchange allows simulations that use incompatible communications protocols to interoperate. For example, within the HLA world, using VR-Exchange, federations using the HLA RPR FOM 1.0 can interoperate with simulations using RPR FOM 2.0, or federations using different RTIs can interoperate. VR-Exchange supports HLA, TENA, and DIS translation.
- ♦ VR-TheWorld™ Server. VR-TheWorld Server is a simple, yet powerful, web-based streaming terrain server, developed in conjunction with Pelican Mapping. Delivered with a global base map, you can also easily populate it with your own custom source data through a web-based interface. The server can be deployed on private, classified networks to provide streaming terrain data to a variety of simulation and visualization applications behind your firewall.
- ♦ VR-inTerra. VR-inTerra is a C++ API for adding terrain agility to applications. It can load, page, or stream terrain from a wide variety of formats or sources into a single, consistent run-time representation, consisting of a collision graph and vector network.

- ♦ **DI-Guy™.** The DI-Guy product line is a set of software tools for real-time human visualization, simulation, and artificial intelligence. Every DI-Guy software offering comes with thousands of ready-to-use characters, appearances, and motions. DI-Guy enables the easy creation of crowds and individuals who are terrain aware, autonomous, and react intelligently to ongoing events. Save time, money and create outstanding simulations with DI-Guy. The DI-Guy product line includes the following products:
 - The DI-Guy SDK. Embed the DI-Guy library in your real-time application and populate your world with lifelike human characters.
 - DI-Guy Scenario™. Author and visualize human performances in a rich, user-friendly graphical environment. Use DI-Guy Scenario as an end visualization application or save scenarios and load them into your DI-Guy SDK enabled application.
 - DI-Guy AI. Generate crowds of autonomous characters to quickly populate your worlds with hundreds and thousands of terrain-aware, collision avoiding DI-Guys. Used as a module on top of DI-Guy Scenario and DI-Guy SDK.
 - Expressive Faces Module. Enable DI-Guy characters to have faces that display emotion, eyes that look in directions and blink, and lips that sync to sound files.
 - DI-Guy Motion Editor. Create or customize motions to your particular needs in an easy-to-use graphical application.

How to Contact Us

For DI-Guy technical support, information about upgrades, and information about other MÄK products, you can contact us in the following ways:

Telephone

Call or fax us at:	Voice:	617-876-8085 (extension 3 for support)
	Fax:	617-876-9208

E-mail

Sales and upgrade information:	info@mak.com
Technical support:	support@mak.com
VR-Vantage support:	vrv-support@mak.com

Internet

MÄK web site home page:	www.mak.com
License key requests:	www.mak.com/support/get-licenses.html
Product version and platform information:	www.mak.com/support/product-versions.html
For the free, unlicensed MÄK RTI:	www.mak.com/resources/bonus-material/cat_view/16-bonus-materials/24-mak-high-performance-rti.html
MÄK Community Forum:	www.mak.com/community-forum/1-forum.html

Post

Send postal correspondence to:	VT MÄK 150 Cambridge Park Drive, 3rd Floor Cambridge, MA, USA 02140
--------------------------------	---

When requesting support, please tell us the product you are using, the version, and the platform on which you are running.

Document Conventions

This manual uses the following typographic conventions:

Monospaced	Indicates commands or values you enter.
Monospaced Bold	Indicates a key on the keyboard.
<i>Monospaced Italic</i>	Indicates command variables that you replace with appropriate values.
Blue text	A hypertext link to another location in this manual or another manual in the documentation set.
{ }	Indicates required arguments.
[]	Indicates optional arguments.
	Separates options in a command where only one option may be chosen at a time.
()	In command syntax, indicates equivalent alternatives for a command-line option, for example, (-h --help).
/	Indicates a directory. Since MÄK products run on both Linux and Windows PC platforms, we use the / (slash) for generic discussions of pathnames. If you are running on a PC, substitute a \ (backslash) when you type pathnames.
<i>Italic</i>	Indicates a file name, pathname, or a class name.
sans Serif	Indicates a parameter or argument.
➤	Indicates a one-step procedure.
Menu → Option	Indicates a menu choice. For example, an instruction to select the Save option from the File menu appears as: Choose File → Save .
	Click the icon to run a tutorial video in the default browser.
	Indicates supplemental or clarifying information.
	Indicates additional information that you must observe to ensure the success of a procedure or other task.

Directory names are preceded with dot and slash characters that show their position with respect to the DI-Guy home directory. For example, the directory *vrforces13/doc* appears in the text as *./doc*.

DI-Guy Conventions

Table 2-1 lists the conventions used for entity coordinates in DI-Guy documentation.

Table 2-1: Coordinate system conventions

Convention	Indicates
XYZ	X = forward, Y = to the left Z = up
Yaw, Roll, Pitch	Orientation is applied in the order YRP. Yaw = rz – orientation about the vertical axis Roll = rx – orientation about the fore-aft axis Pitch = ry – orientation nose down, nose up

Mouse Button Naming Conventions

An instruction to click the mouse button, refers to clicking the primary mouse button, usually the left button for right-handed mice and the right button for left-handed mice. The context-sensitive menu, also called a popup menu or right-click menu, refers to the menu displayed when you click the secondary mouse button, usually the right button on right-handed mice and the left button on left-handed mice.

Third Party Licenses

MÄK software products may use code from third parties. This section contains the license documentation required by these third parties.

Boost License

VR-Link, and all MÄK software that uses VR-Link uses some code that is distributed under the Boost License. All header files that contain Boost code are properly attributed. The Boost web site is: www.boost.org.

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the “Software”) to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

libXML and libICONV

VR-Link and all MÄK software that uses VR-Link, links in libXML and libICONV. On some platforms the compiled libraries and header files are distributed with MÄK Products. MÄK has made no modifications to these libraries. For more information about these libraries please see the following web sites:

- The LGPL license is available at: <http://www.gnu.org/licenses/lgpl.html>.
- Information about IconV is at: <http://www.gnu.org/software/libiconv/>.
- Information about LibXML is at: <http://xmlsoft.org/>.

Lua

VR-Forces and DI-Guy use the Lua programming language (www.lua.org). Its license is as follows:

Copyright © 1994–2012 Lua.org, PUC-Rio.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.



1. DI-Guy Expressive Faces

This chapter explains how to use the DI-Guy Expressive Faces plug-in.

Introduction	1-2
How to Work with DI-Guy Expressive Faces	1-3
Create a Script	1-5
Select an Avatar	1-7
Prepare a Base Scenario	1-8
Record Voice Performances	1-9
Create Lip-Synching Animations in FaceFX Studio Professional	1-10
Create a Simple Local Path for Each Response (Animation)	1-16
Creating a FaceFX Head Appearance	1-17
Creating and Exporting Data from 3ds Max	1-18
Setting Up and Publishing a New FaceFX Actor	1-20
Setting Up a Head for Use in DI-Guy Expressive Faces	1-20
Considerations for Using Expressive Faces	1-22

1.1. Introduction

DI-Guy is software for realistic human simulations. Standard DI-Guy characters have static faces and are often viewed from far enough away that this is a satisfactory visual result (and a win for high performance visualization).

The DI-Guy Expressive Faces module enables you to enhance DI-Guy characters with expressive face geometry to create a more intimate performance with their characters including:

- ♦ Faces that emote.
- ♦ Lips that synchronize to audio.
- ♦ Eyes that blink and gaze.

DI-Guy Expressive Faces users also tend to make heavier use of some standard features of DI-Guy such as gesturing, pointing, and gazing.

DI-Guy Expressive Faces is the perfect solution for simulations where communicating with other human characters is critical and characters are close to the camera. DI-Guy Expressive Faces have been used for:

- ♦ Interrogation training for criminal investigators.
- ♦ Cultural training.
- ♦ Use-of-force training.

Developers will often couple speech recognition technology with DI-Guy Expressive Faces characters to communicate naturally and have conversations with their lifelike Expressive Faces avatars. The avatars in turn often have sophisticated branching logic and performances that allow the avatar to react intelligently to questions, building on previous questions, and so on.

OC3's FaceFX software underpins DI-Guy Expressive Faces. Use of FaceFX provides the following benefits:

- ♦ The FaceFX Studio Professional application. This software enables you to create lip-synching and other expressive performances. The user interface lets you edit parameters for facial animation, such as blink, eye brow raise, head tilt, and so on, and has a timeline-based editor that lets you tune the performance. FaceFX Studio Professional comes as part of the DI-Guy Expressive Faces development license.

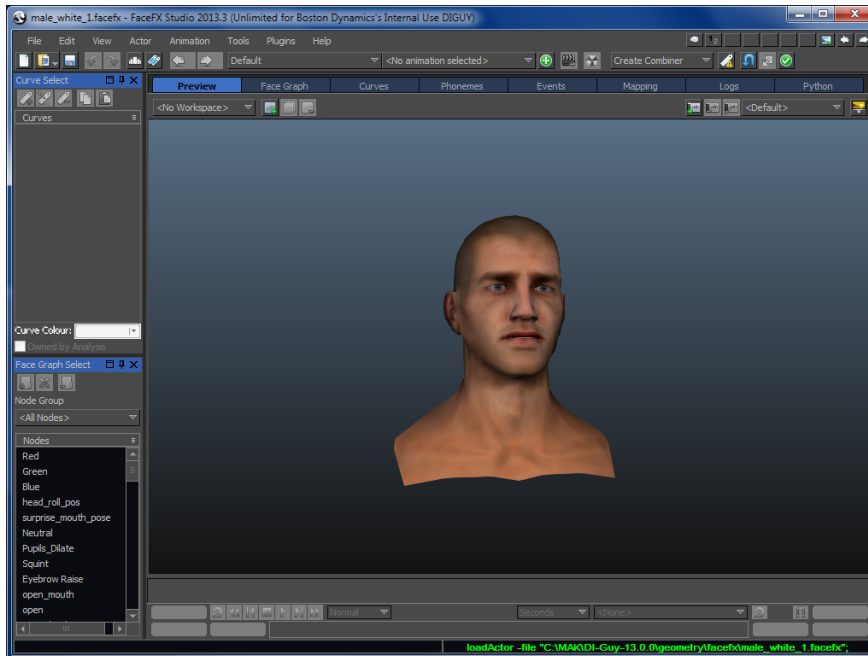


Figure 1-1. FaceFX

- ♦ FaceFX is a popular choice in the video game industry. Like DI-Guy, OC3 is investing in the continuous upgrade of FaceFX, so that users who choose DI-Guy Expressive Faces can feel confident that their investment will keep pace with the ever-advancing changes in human and facial simulation.

1.2. How to Work with DI-Guy Expressive Faces

There are many ways to work with DI-Guy Expressive Faces, including:

- ♦ Using a DI-Guy Expressive Faces AI hierarchical finite state machine (HFSM) to control logic.
- ♦ Have users (typically trainees) interact with an avatar
- ♦ Using speech.
- ♦ Using a button-pushing user interface.

This document does not try to address all possible modes of using DI-Guy Expressive Faces. It describes a technique called the expressive local paths technique that has been successful with DI-Guy customers and DI-Guy developers. The goal is to quickly introduce you to using DI-Guy Expressive Faces.



We recommend that you use simple nomenclature for files and other assets. If possible, avoid using non-alphanumeric characters, dashes, spaces, and so on.

The expressive local paths technique assumes:

- ♦ The scenario to be created will be interactive between a user (referred to as the trainee) and one or more DI-Guy characters equipped with DI-Guy Expressive Faces (referred to as avatars).
- ♦ The scenario follows a question and answer format between the trainee and the avatar. Occasionally actions performed by the trainee, such as drawing a weapon, will also influence avatar behavior.
- ♦ Responses to a particular question or action may be variable and will be affected by what questions or actions have previously occurred in the simulation.

The following steps summarize the expressive local path technique:

1. Create a script.
2. Select an avatar.
3. Select a terrain, props, and background characters and prepare a base scenario.
4. Record voice performances.
5. Create lip-synching animations in FaceFX Studio Professional.
6. Create a simple path for each avatar response.
7. Create signals and a pushbutton interface for signaling questions to the avatar.
8. Create an expressive local path for each avatar response by adding gestures, gazes, and actions to the simple local path.
9. Build up logic for how the avatar chooses a response.
10. Create and train a vocabulary for accepting speech from the trainee and translating it into a signal.

In the following sections, we will examine each step in detail.

1.2.1. Create a Script

The first step to creating your performance is to think about your goals for the simulation and how they can best be achieved. Consider the following questions:

- ♦ Is your application training the trainee on how to ask certain questions, or perform a particular task?
- ♦ How can your scenario most effectively get at this core education?
- ♦ Is the scenario meant primarily to train, demonstrate, or test?
- ♦ How can you set boundaries on the simulation?

It will not be possible to have your avatar respond to every conceivable question. You will have to fit your development to the goals and the budget you have for authoring.

For this demonstration, we have created a simple interrogation scene. It tries to do the following:

- ♦ Create a simulation likely to be useful to DI-Guy customers.
- ♦ Demonstrate a wide range of emotion for the avatar.
- ♦ Demonstrate logic that shows that previous questions affect future responses.
- ♦ Create a simulation that can be run in 2-3 minutes.
- ♦ Choose neutral subject matter.

i

- ♦ The initial script differs from the demonstration scenario included in the distribution of DI-Guy Expressive Faces (Exface_Sales_Demo). Like all scripts, it undergoes changes as voice actors and developers acting as directors make modifications during the process of simulation creation.
 - ♦ This is a purely fictitious scenario intended for demonstration purposes. It does not claim to follow real-world interrogation practices.
 - ♦ There are two answers to most of the questions; this is because these questions can be asked twice.
-

Sample Interrogation Script

0? Can we ask you some questions? [Miranda rights]

No, it's OK, I've got nothing to hide. I don't even know what this is about. (friendly)

1? What is your name?

My name's Timothy Johnson, you can call me Tim. (smile)

I already told you, my name's Tim...

2? Where were you last night?

Last night I was with my buddy Johnny, you know, having a few beers. (laughs) We watched TV at his house all night. (happy)

Yea, like I was telling you, I was with my pal Johnny. That guy's the greatest. We watched the baseball game all night at his place.

3? Really? We arrested your friend Pete Kenny last night for burglary, and he said you were driving the car... [establishing witness]

I don't believe you. (disgust) *I was with Johnny last night.*

Why would Pete say that? (disgust)

4? Do you know that an officer was shot last night? [establishing guilt]

What? Really? (confusion) *I...I didn't know that.*

I understand, that's terrible. (sympathy, empathy)

5? Listen Tim, we're here to help, we just want to know what happened. [establishing trust]

I dunno...I was with Johnny last night. (confusion, weakness)

Listen, I didn't do it, but if I did, it would have been for the money, not to hurt anybody!

(Once 3,4,5 have been asked at least once, and perhaps forcing 2 or 3 of the questions to be asked twice, he then confesses.)

Alright, listen, I did it, it was a stupid idea. Pete put me up to it. I didn't want anyone to get hurt. That was Pete!...I think I should get a lawyer.

1.2.2. Select an Avatar

Selecting the best starting avatar involves three things:

- ♦ Selecting the best DI-Guy Expressive Faces character type.
 - ♦ Selecting the best body appearance.
 - ♦ Selecting the best facial appearance.
1. Open the DI-Guy Character Viewer. DI-Guy Character Viewer lets you audition all of the content that comes with DI-Guy. (For details about the Character Viewer, please see Chapter 5, *Using the Character Viewer*, in *DI-Guy Content Reference Manual*. If you have created any custom content, you can view it as well.)
 2. Choose the character type that has the most actions similar to what you are looking for in your performance. The “suspect_09” character type is a particularly rich character type that has many interesting actions relevant to simulations that might need facial animation.

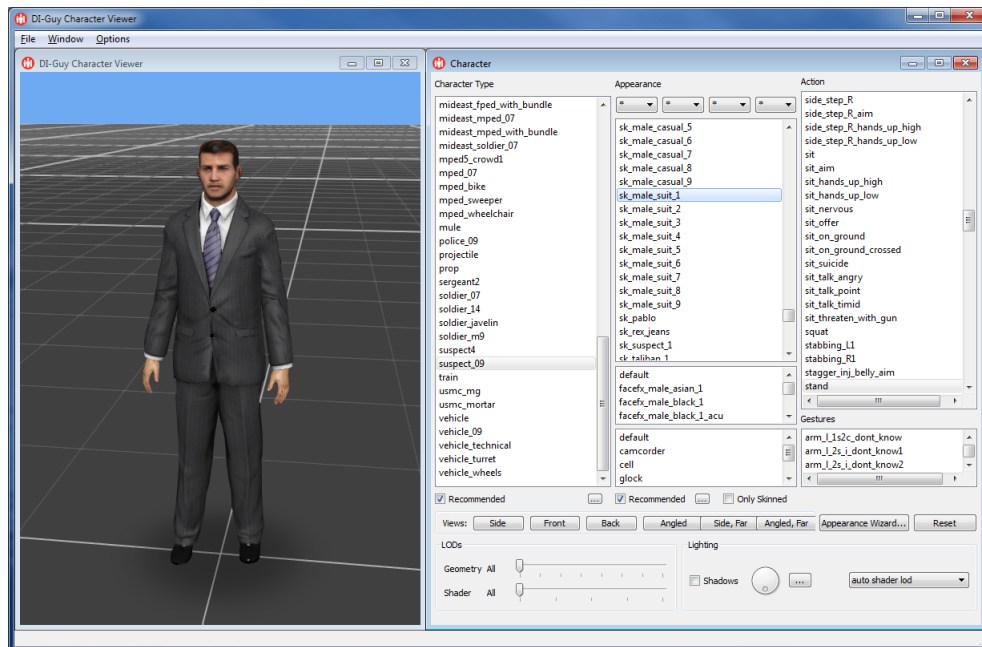


Figure 1-2. suspect_09

3. Find the best appearance for the body and the facial appearance. Use the Appearances column to select different body appearances. Not all appearances have corresponding expressive face appearances. Those that do have a secondary appearance list under the main appearance list in which one or more expressive face appearances are listed.



If you cannot find the right combination of character type, appearance, and facial appearance, please contact DI-Guy Support at support@mak.com for assistance. Because FaceFX-style expressive faces were introduced in DI-Guy 12.0, there are only a small number of sample faces available for certain appearances. Let us work with you to get you the character type, appearance, and facial appearance that will let you be successful.

DI-Guy Expressive Faces is fully customizable, so if there are additional motions, character appearances, or facial appearances required, these can be added by the developer. Contact support@mak.com for assistance with facial customization. See *DI-Guy Content Reference Manual* for customizing character appearances and *DI-Guy Motion Editor Users Guide* for instructions about customizing actions for DI-Guy.

You can also contract with DI-Guy Services to create customizations, as well as building a starting scenario or other assistance.

1.2.3. Prepare a Base Scenario

1. Start DI-Guy Scenario¹.
2. Choose a scene object (terrain). It can be one provided by DI-Guy or one that you supply.
3. Determine where you would like your avatar(s) to be located, create a character of the character type and appearances decided in “[Select an Avatar](#),” on page 1-7.
4. Choose one or more interesting camera angles to help bring the performance to life.
5. Place appropriate vehicles and props in your scene.
6. If there are any supporting characters, add them to the scenario.

1. The use of DI-Guy Scenario is recommended but not required for implementing Expressive Faces. Developers choosing not to use DI-Guy Scenario need to perform these operations using their own tools.

1.2.4. Record Voice Performances

The quality of the voice recordings will be a major factor in the believability of your end performance.¹ Two factors are important:

- ♦ Technical quality. Use a quality microphone. Have post-processing tools that, at a minimum, allow you to crop audio recordings, reduce noise, and output *.wav* files.
- ♦ Performance quality. Choose a voice actor that has a voice that matches your avatar and is not a distraction. Be sure that the voice actor not only has a fine voice, but can emote and act convincingly. When producing the vocal performance, give strong direction to the actor to assure you get the voice capture that will make your scenario successful.

Consider video recording the performance to assist with later steps where the facial animation will be emotively tuned in FaceFX Studio Professional and DI-Guy Scenario to help bring the performance to life. Larger budget programs can even consider performing a motion capture session to capture skilled actor performances for reproduction in the simulation.

Final *.wav* files should be placed in *./custom/sfx/<recording session name>*. Consider using descriptive filenames and perhaps including the actor's name in the filename.

1. Advanced Developers may consider using speech generation for their avatar. Contact support@mak.com for support and advice.

1.2.5. Create Lip-Synching Animations in FaceFX Studio Professional

1. Start FaceFX Studio Professional.
2. Choose **File** → **Open Actor**. Available actors are in `./geometry/facefx`. These files have a `.facefx` suffix. You do not need to create a custom file. The face is displayed in the window
3. Press **Alt** and move the mouse to rotate the face to a frontal view (Figure 1-3).

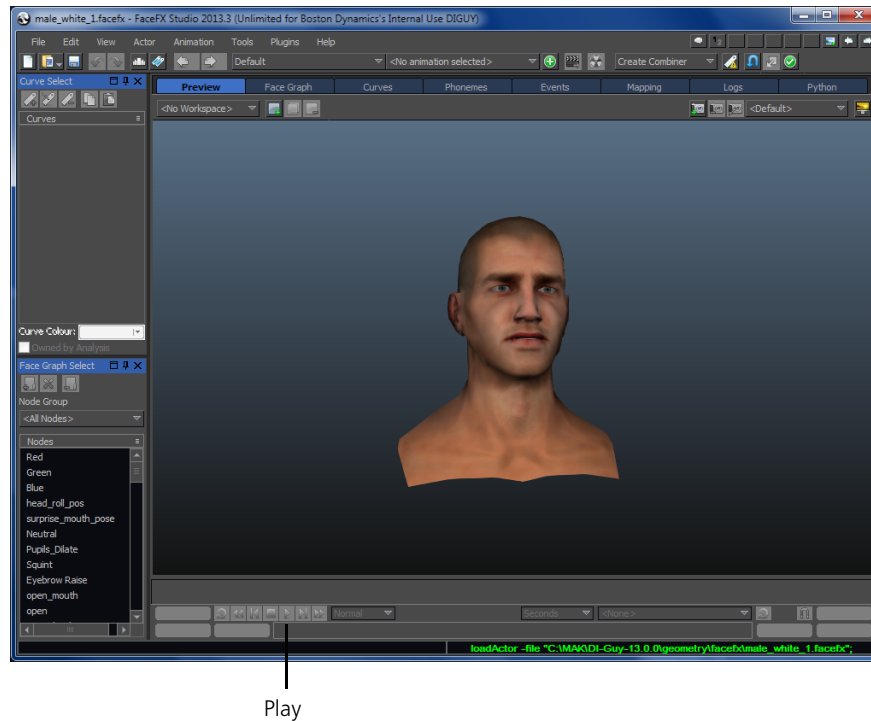


Figure 1-3. FaceFX Studio Professional

4. Generate a new animation group for the actor. Choose **Actor** → **Animation Manager**. The Animation Manager opens (Figure 1-4).

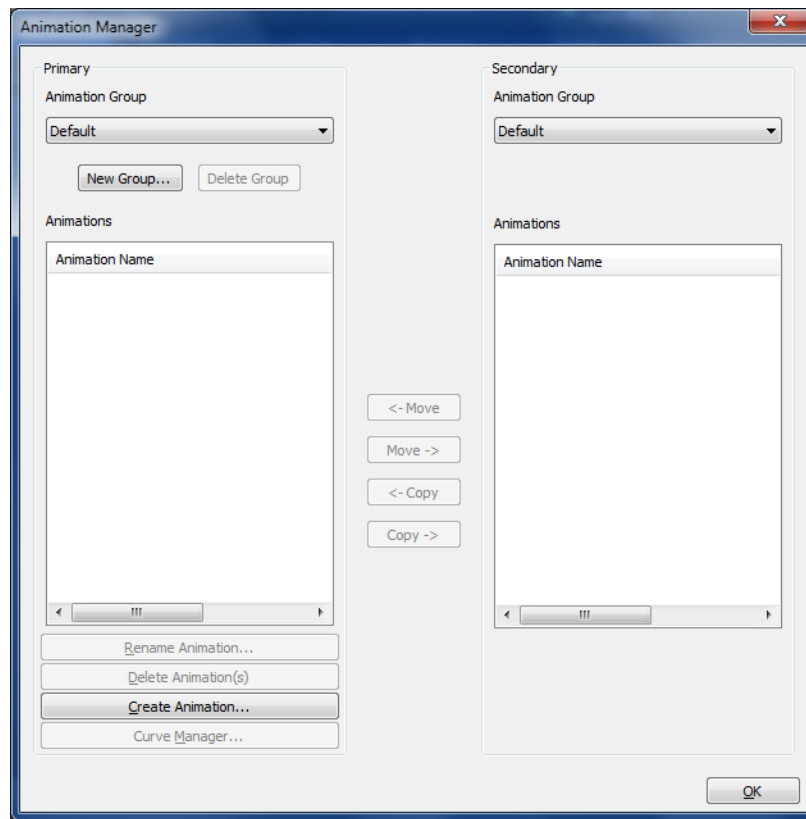


Figure 1-4. Animation Manager

5. Click **New Group**. The Create Animation Group dialog box opens.
6. Type a name for your animation group. We will use “interrogation”.
7. Click OK.
8. Click Create Animation. The Create Animation wizard opens.
9. Select Create an Animation from an Audio File and click Next.
10. Select a *.wav* file and choose Next. The wizard prompts you for the text of the sound file. (If you do not have your own wave file and just want to try this out, select *./sfx/exface_sales_demo/bob_now_were_talkin.wav*. The dialog text is “Now we’re talking.”)
11. Enter the dialog text and select Next.
12. Click Next to accept the language and default analysis actor file. The wizard prompts you with an animation name derived from the wave file name.
13. Click Finish. The animation name is added to the list (Figure 1-5).

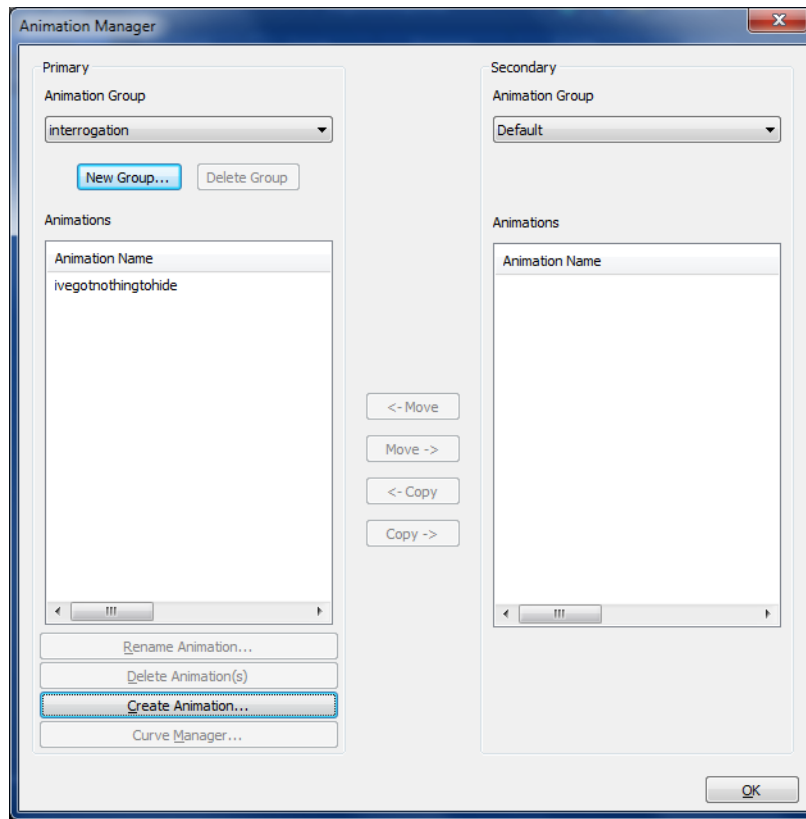


Figure 1-5. Animation Manager with new animation

14. Click OK.
15. In the main window, click Play (Figure 1-3). You should see the character speak the text you have entered.

The character will not only perform the lip-synching, but other facial animation will be created by the system, attempting to get you started on a convincing performance. These may include various head tilts, eye brow motions, and blinks. If you are lucky, the performance will be satisfactory and you can consider publishing the animation and testing it in DI-Guy.

However, you will usually want to improve the performance. FaceFX Studio Professional has a wealth of features for directly editing the performance. The following sections demonstrate some basic tools you can use to refine the performance. See FaceFX Studio documentation for more details.

Refine the Lip-Synching

Examine the lip-synching in detail. If you do not believe that the character's mouth is truly speaking the phonemes you hear, then the quality of the rest of the performance does not matter.

1. Select the Phonemes tab. [Figure 1-6](#) illustrates the phoneme tab for the text, “No. OK. I’ve got nothing to hide.”

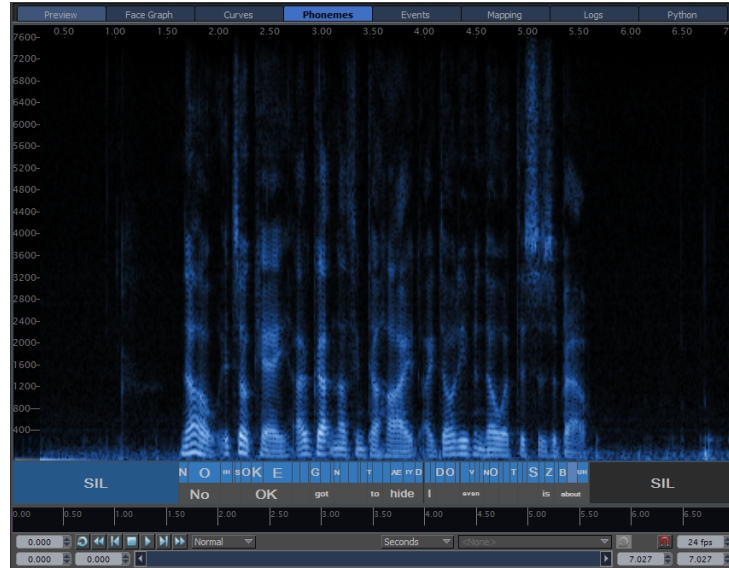


Figure 1-6. Phonemes tab

The display shows the audio wave. Beneath it is the phoneme breakdown of the sentence as matched to the actual audio wave. Beneath the phoneme breakdown is the exact text that you entered when you created the animation. Timing, in seconds, is shown along the top and bottom of the screen.

2. Adjust the location and width of the phonemes to match the lip movements.
3. Go back and forth between this window and the Preview window. Try playing the performance at quarter time by changing the Normal pulldown on the playback controller to Very Slow. While the voice will be much deeper, it is much easier at this speed to see and correct lip-synching timing issues.

Besides editing the phonemes, you can insert or delete them. One thing you might want to do is close the mouth during pauses. We will see how to do this in the next section.

Curve Editing

One way to edit an animation is to edit curves for the different facial movements, such as opening and closing the mouth, blinking, head movement, and so on.

1. Select the Curves tab (Figure 1-7).

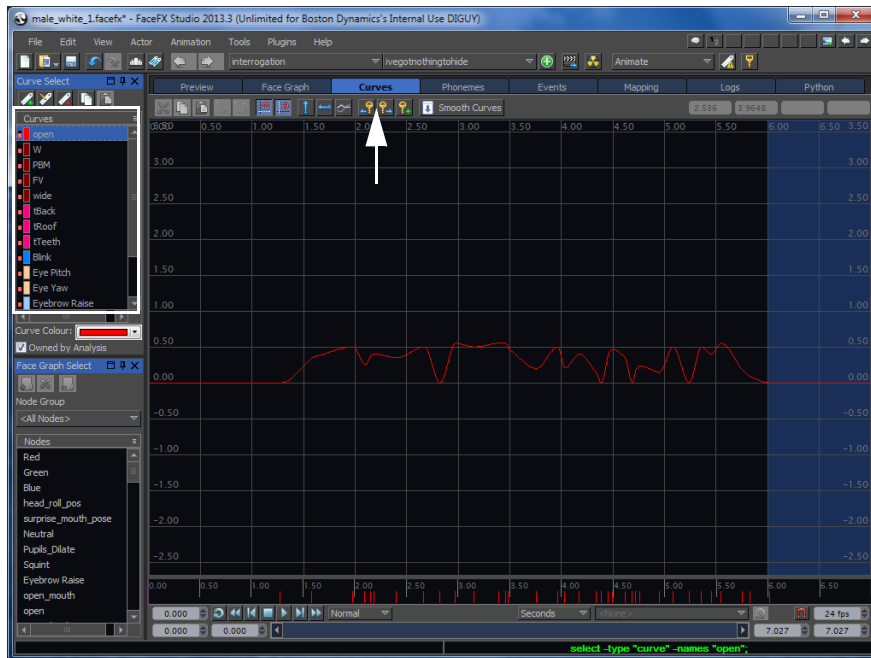


Figure 1-7. Curve editing

2. On the Curves palette, select open. Open controls how far open the mouth is.
3. To edit the curve, clear the Owned by Analysis check box. Points are added to the curve (Figure 1-8). You can move the control points up, down, or laterally in the curve editor. You can add new control points.

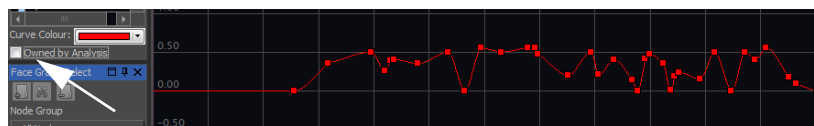


Figure 1-8. Curve points

4. Try editing other curves. Look at some of the other curve types, such as Blink, Squint, and Head Roll.



FaceFX adds head movements based on its analysis of the sound file and text. If the voice recording is fairly short, such as “Now we’re talking,” it may not add many of the possible types of curves. You might want to add curves that it did not insert or try a longer recording to see if you get more movement added automatically.

To add a new curve type:

1. On the Curve Select palette, click the Add button (). The Select Node to Drive or Type New Curve Name dialog box opens.
2. Select a curve in the list.
3. Click OK. The curve is added to the list. In the window it is a straight line.
4. Edit the curve to add the motion you want.

Working with these parameters in FaceFX is where technology meets art. The variation you can add to your performance is infinite. If you have made video recordings of your actor, you can watch those videos and duplicate the facial animations you liked the best.

Saving and Publishing Your Work

When you are satisfied with the new animation, save your actor and animation set. If this is the first time you are saving the animation set, you also need to save (export) it.

1. Choose **Actor** → **Export Animation Set**. A file browser opens.
2. Save the Animation set to `./custom/sfx/<your project name>/<character name>`. For this demo we called the project name *interrogation* and the character name *suspect*. Once you have done this, the Export Animation Set will be unavailable and the update of the animation set happens automatically during actor save or game publishing.
3. Select **File** → **Publish to Game**. The Publishing Options dialog box opens.
4. Set the Output directory to `./custom/sfx/<your project_name>` and accept the default checkboxes.

Reopening Your Work

When you want to do some more work:

1. Start FaceFX Studio Professional.
2. Choose **File** → **Open Actor** and select the same actor as before.
3. Choose **Actor** → **Mount Animset** to retrieve your older animations.

1.2.6. Create a Simple Local Path for Each Response (Animation)

1. Start DI-Guy Scenario and load the scenario you created in “[Prepare a Base Scenario](#),” on page 1-8.
2. Open the Character Objects:Character page and select the character for which you made the animations.
3. In the Head Appearance list, select facefx_male_white_1. This gives the character the head appearance you created.
4. Select the Character Objects:Face Exp. page ([Figure 1-9](#)).
5. Select the FaceFx Assets tab.
6. Select your FaceFx Actor (male_white_1).
7. Click Add Animset. The Select Animset to Mount on Actor window opens.
8. Select the animset you saved (interrogation).
9. Click OK. The animation set is added to the Animation Groups section of the Assets list [Figure 1-9](#).
10. Expand the interrogation section. It lists the wave files that you saved in FaceFX.
11. Double-click the ivegotnothingtohide line. The character executes the animation.

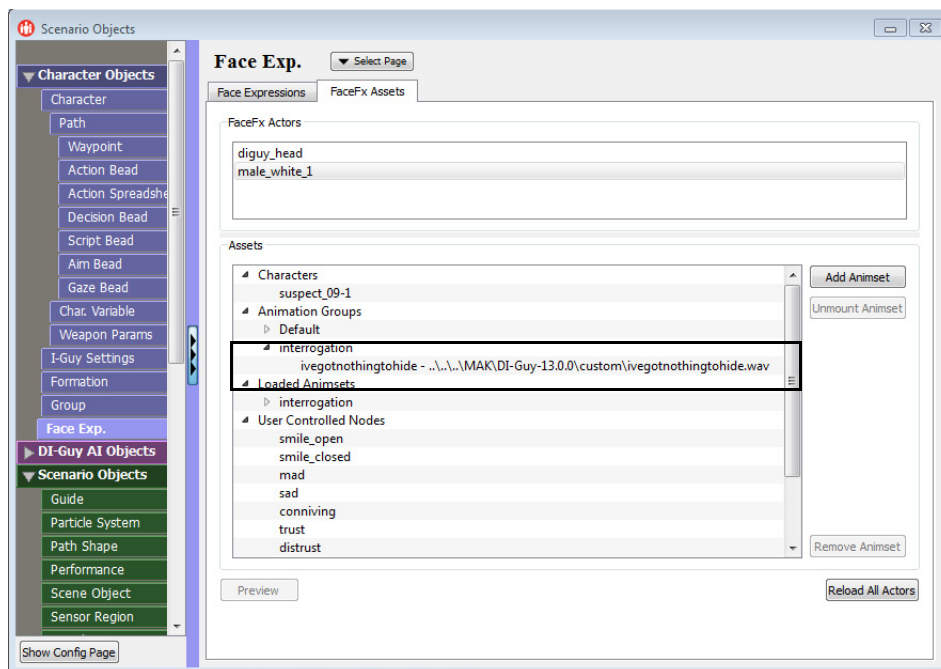


Figure 1-9. Face Expression page

At the bottom of the FaceFx Assets tab there is the Reload All Actors button. This button is very useful. You can keep your FaceFX Studio Professional session running, and as you modify or add animations, once you have saved those changes, you do not need to restart DI-Guy Scenario to use those changes, just click the button.

Create a path for the animation.

1. Since the animation is called ivegotnothingtohide, select the Character Objects:Path page.
2. Add a path.
3. Change the name from path0 to ivegotnothingtohide.
4. Select the Character Objects:Decision Bead page.
5. Add a decision bead and logic to play your animation (Figure 1-10).

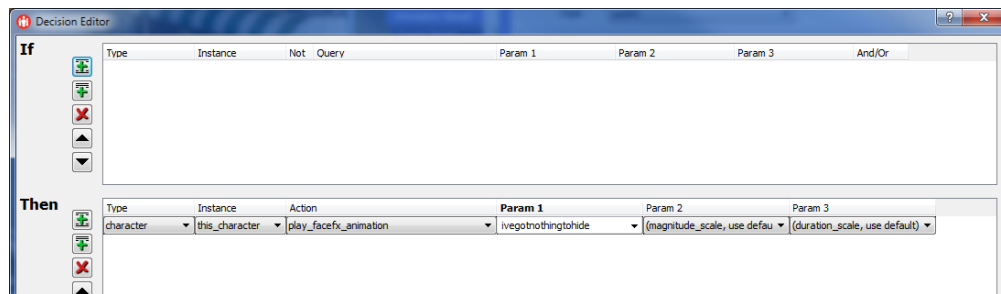


Figure 1-10. Animation setup

6. To audition the path, go to the Character Objects:Character page and make this path be the Initial Path.
7. Click Play. You may want to change the character's action to sit or stand, because the default is to walk.
8. On the Path page, click the Copy button for each animation you created. It is recommended that you set the "Path Paste Offset" to 0, 0, 0 in the Preferences dialog box so that each new copy is in the exact same place.
9. Update the new paths with better names and point the decision bead entries at the appropriate animation.

We now have one or more simple paths, but they are not local. Making these paths be local means that when the path starts, it will begin where the last path left off. This is useful when stringing paths together, the last thing you want is a discontinuity when a new path is activated.

1.3. Creating a FaceFX Head Appearance

This section explains how to create and export data from 3ds Max and how to set up and publish a new FaceFX actor.

1.3.1. Creating and Exporting Data from 3ds Max

1. Open a sample FaceFX model from `./geometry/facefx`. There should be a number of `.max` files to use as a base.
2. Modify geometry, textures and so on. We recommend using the skin wrap modifier if you need to do any significant changes to vertex topology. The wrap modifier allows you to reference a second model and copy its weights by proximity.
3. Verify that the character can go through all the emotions and poses in the animation data.
4. Pick the FaceFX utility plug-in ([Figure 1-11](#)).

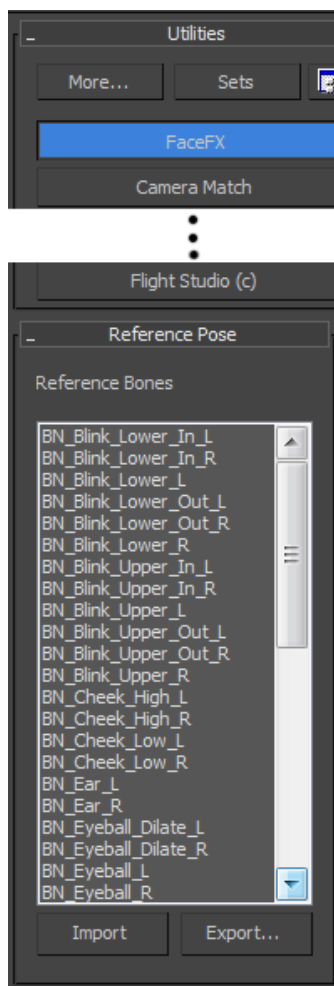


Figure 1-11. FaceFX plug-in

5. Click Create Actor and give the character a reasonable name that will match the associated head appearance.

6. On the Reference Bones panel, click Export and select all bones with the BN_ prefix.
7. On the Nodes Panel, click Batch Export.
8. Pick `./geometry /facefx/diguy_batch_export.txt`. This script has a list of poses and the keyframes they happen at.
9. Click Save Actor and place the new actor file in `./custom/geometry /facefx`.
10. Select the head of the character and execute the Export Selected command. The file should be exported as a Ogre `.Mesh` file. Give it the same name as the actor and save the file in `./custom/geometry /facefx/`.

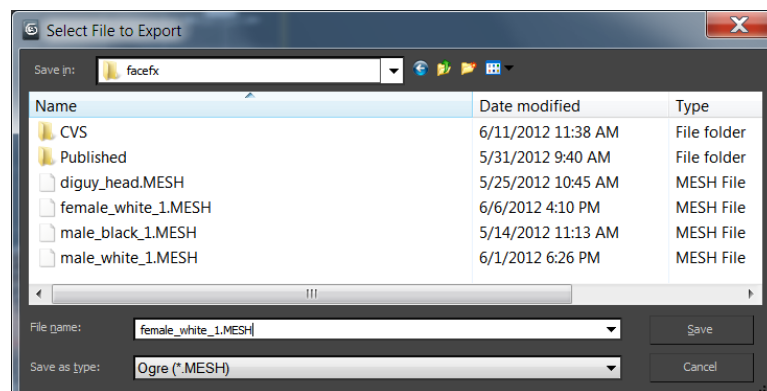


Figure 1-12. Exporting a file



Any change in the hierarchy of character bones or changes to the neutral pose will probably require that the actor is recreated. When in doubt go through the export process again. Select the various mesh LODs of the character and all of the BN_joints and export them as a collada DAE file in `./custom/geometry /dae`.

1.3.2. Setting Up and Publishing a New FaceFX Actor

1. Start FaceFX Studio.
2. Open the actor file that was saved in *./custom/geometry/facefx*. You should now see the face that was exported as an Ogre mesh file.
3. If you switch to the face graph view there will be a unstructured node graph that does not have constraints or connections.
4. Choose **Actor** → **Sync to Template** and import *./geometry/facefx/diguy_head_face_graph.fxt*. This should now give you a character that can animate and lip sync to audio.
5. Please see the FaceFX Studio documentation for the procedures to author animation and create animsets for use in DI-Guy.
6. Choose **File** → **Publish to Game** and publish the new actor in *./custom/geometry/facefx/published*.

1.3.3. Setting Up a Head for Use in DI-Guy Expressive Faces

Example of a FaceFX shape set:

```
shape_set facefx_male_w1
  actor = greg
  is_base_shape = 0
  class_type = head
  multi_lod_filename = facefx_male_w1.dae
  shape_attachment_data = head position facefx
  shape_suffix_lod1 = _lod1Mesh
  shape_suffix_lod2 = _lod1Mesh
  shape_suffix_lod3 = _lod1Mesh
  shape_suffix_lod4 = _lod2Mesh
  shape_suffix_lod5 = _lod2Mesh
  shape_suffix_lod6 = _lod2Mesh
  shape_suffix_lod7 =
  parameters
    facefx_actor = male_white_1
```

Special flags for a FaceFX head shape set:

- ♦ `class_type = head`. Indicates that this shape replaces other items with class type of head, without this the character will have two heads.
- ♦ `shape_attachment_data = head position facefx`. Indicates that the head mesh should be attached to the position link and be initialized with the FaceFX module shape callbacks.
- ♦ `facefx_actor = male_white_1`. Indicates which FaceFX actor should be used when this shapset is loaded.

Example of a FaceFX head appearance:

```
head_appearance facefx_male_white_1
  item = facefx_male_w1
  parameters
    head_type = male_large
    graphics_state_switch_map1 = sk_male_w1_hands:male_w1_head
```

Special flags for a FaceFX head appearance:

- ♦ `head_type = male_large`. Indicates what body appearances this head is compatible with. Body appearances are also tagged with a `head_type` field to allow the user interface to only present the appropriate heads for a given appearance.
- ♦ `graphics_state_switch_map1 = sk_male_w1_hands:male_w1_head`. This state switch applies to the character's hand item and keeps the color of the hands synchronized with the color of the head. This is applied after any state switches in the body appearance.



The head appearance functionality is not limited to FaceFX. Non-FaceFX heads can be made swappable as well. The `shape_attachment_data` should just not specify FaceFX.

1.4. Considerations for Using Expressive Faces

- ♦ DI-Guy Expressive Faces uses a shader lookup table to establish a mapping between 3ds Max bones and DI-Guy links. The lookup table is specified in `./config/common/shader_matrix_index_tables.cfg`. If you plan to use a different naming convention for your face bones you must add a new `shader_matrix_index_table` entry to a custom version of `shader_matrix_index_tables.cfg`.
- ♦ Number of Joints. DI-Guy uses a maximum of 50 link matrixes in shaders. If you plan to build an expressive face with more than 50 joints, you must modify both the shaders and the `shader_matrix_index_tables`.
- ♦ Scale mismatch between DI-Guy and FaceFX. The animation data that FaceFX produces is scaled by .0254 when mapped into DI-Guy's animation system. A few models have required a different value. It has been unclear where in a 3 application/2 file format export pipeline the problems arise. [Figure 1-13](#) illustrates this problem.



Figure 1-13. Scale mismatch

To fix this, add the following meta data to `./custom/config/diguy/facefx.cfg`:

```
face_data female_white_1
unit_conversion_factor = .01
```

- ♦ Scaling of heads in 3ds Max is also very tricky, it may require scaling the `BN_Head` node, disconnecting all children and scaling the bones back to a scale factor of `<1.0,1.0,1.>`, renormalizing all scale keyframe data on the joints, and then reconnecting the hierarchy. After a mesh is scaled the mesh will need a `Reset XForm` modifier added above the mesh and below the skin and collapsed.

After changes to joint position it is usually necessary to reset the skin by checking and unchecking `Always Deform` in the advanced section of the `Skin Modifier`.



Index

Numerics

3DS Max, exporting data from 1-18

A

actor

- opening 1-10, 1-15

- publishing 1-20

- saving 1-15

- setting up 1-20

animation

- creating 1-11

- decision bead 1-17

Animation Manager 1-10

animation set

- opening 1-15

- saving 1-15

appearance, FaceFX 1-17

audio, recording 1-9

avatar, selecting 1-7

C

Character page 1-16

character type 1-7

Character Viewer 1-7

Create Animation Group dialog box 1-11

Create Animation wizard 1-11

creating

- animation 1-11

- local path 1-16

- scenario 1-8

- script 1-5

curve, editing 1-14

Curve Select palette 1-15

Curves palette 1-14

D

decision bead, animation 1-17

Decision Bead page 1-17

dialog box, Create Animation Group 1-11

DI-Guy Character Viewer 1-7

DI-Guy Scenario 1-8

E

editing

- curves 1-14

- lip-synching 1-13

exporting, data from 3DS Max 1-18

expressive local path technique 1-4

F

Face Exp. page 1-16

FaceFX 1-2

- appearance 1-17

FaceFX Studio Professional 1-2

- lip syncing 1-10

finite state machine 1-3

H

head, setting up for Expressive Faces 1-20

hierarchical finite state machine 1-3

J

joint, number of 1-22

L

lip syncing, FaceFX Studio Professional 1-10

lip-synching, editing 1-13

local path, creating 1-16

M

machine, finite state 1-3

O

opening

actor 1-10, 1-15

animation set 1-15

P

page

Character 1-16

Decision Bead 1-17

Face Exp. 1-16

Path 1-17

palette

Curve Select 1-15

Curves 1-14

path

local, creating 1-16

Path page 1-17

Phonemes tab 1-13

publishing, actor 1-20

R

recording, voice 1-9

S

saving

actor 1-15

animation set 1-15

scale mismatch 1-22

scenario, creating 1-8

script, creating 1-5

selecting, avatar 1-7

shader 1-22

state machine 1-3

T

tab, Phonemes 1-13

V

voice, recording 1-9

W

wizard, Create Animation 1-11



Link - Simulate - Visualize

DIG-13.0-4-140523