



DI-Guy 13.0.1 Release Notes

This document provides the following release-specific information:

Systems Supported	2
Renderer Support	2
Network Compatibility	3
RTI Support	3
FOM Support	3
New Features and Updates	4
API Changes	6
Bug Fixes	6
Known Problems	8

Copyright © 2015 VT MÄK, 150 Cambridge Park Drive, 3rd Floor, Cambridge, MA 02140 All rights reserved. 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.

Document ID: DIG-13.0.1-8-150320

Systems Supported

Table 1 lists the platforms currently supported by MÄK DI-Guy 13.0.1. Application code must be built with the indicated compilers in order to link to DI-Guy libraries.

Table 1: Platforms supported

Platform	Compiler
Red Hat Enterprise Linux Workstation 6.0	Default compiler (64-bit)
PC with Windows XP/Vista/Windows 7	Microsoft Visual C + + 9.0 (32-bit and 64 bit) Microsoft Visual C + + 10.0 (32-bit and 64 bit)



You must be an administrator to install MÄK products on Windows Vista.

Renderer Support

DI-Guy 13.0.1 supports the following renderers:

- Windows 32 bit: OpenGL, OpenGL Graphics API, DirectX 9 Graphics API, OSG Graphics API, Unity, VegaPrime.
- Windows 64 bit: OpenGL, OpenGL Graphics API, OSG Graphics API
- Linux: OpenGL, OpenGL Graphics API, OSG.

License Manager

To run DI-Guy 13.0.1, you must install MÄK license management software. DI-Guy 13.0.1 uses FLEXlm 11.11. You must obtain a new license file from MÄK. If you are running a previous version of DI-Guy, your old license will continue to be valid for that version.

The License Manager files are not part of the DI-Guy installer. You can download them at:

- Windows:
<ftp://ftp.mak.com/out/MAKLicenseManager-win-setup.exe> (for 32-bit systems)
<ftp://ftp.mak.com/out/MAKLicenseManager-win64-setup.exe>
- Linux:
<ftp://ftp.mak.com/out/MAKLicenseManager-linux-setup.tar.gz> (for 32-bit systems)
<ftp://ftp.mak.com/out/MAKLicenseManager-linux64-setup.tar.gz>

A license manager FAQ is available at:

<http://www.mak.com/license-support.html>

Network Compatibility

HLA only

DI-Guy 13.0.1 was built against VR-Link 5.1.2 and is compliant with:

- ♦ RPR-FOM 1.0 and a subset of 2.0 (draft 6, 14, and 17).
- ♦ MÄK RTI 4.1 or later .

DIS only

DI-Guy 13.0.1 supports DIS 4, 5, 6, and 7.

RTI Support

We recommend using the latest version of the MÄK RTI. However, DI-Guy should work with other RTIs that conform to the HLA 1.3 specification, the IEEE 1516 SISO DLC API or HLA Evolved and are built with the same compiler as DI-Guy.

When using the MÄK RTI, remember to make sure that DI-Guy can find your FED file, and optional *rid.mtl*, file by either putting them in the directory from which you are running, or by setting the environment variable to the directory that contains them.

FOM Support

DI-Guy 13.0.1 has built-in support for versions 0.5, 0.7, 0.8, 1.0, and 2.0, drafts 6, 14, 17 and 20, of the RPR FOM. It also supports VR-Link's ability to support alternative FOMs through the FOM Mapper. By default, DI-Guy 13.0.1 uses RPR FOM 1.0. It is recommended that you use the DI-Guy FOM Extensions. DI-Guy includes two custom FOMs, one for HLA 1.3 and HLA 1516, and one for HLA Evolved.

New Features and Updates

DI-Guy 13.0.1 is primarily a maintenance release. It has the following updated features:

- ♦ Improved support for collada mesh data exported from Maya.
- ♦ Added `diguyCharacter::set_desired_position_double()` and `get_desired_position_double()`
- ♦ Improved guide code to handle large value locations better.
- ♦ Added `diguyCharacter::set_is_network_paused()` allows end users to duplicate network paused behavior without using our network layer.
- ♦ Updated instancing to respect texture swaps. Can be toggled off for increased performance. Added `instancing_respect_graphics_state_switch` to `bdiGraphicsInitGraphicsAPI` struct.
- ♦ Increased precision of positions to better support geocentric coords.
- ♦ Characters with custom up vectors will move better along the coordinate system that the up vector defines. previously limited to motions with forward direction and other limitations.
- ♦ Added new z velocity threshold option for forcing a packet update, this can help with up and down movement over the network.
- ♦ Added a new UI for editing packet thresholds.
- ♦ New draw pass system allows client applications to register draw passes with DI-Guy that are called instead of the native rendering when calling `diguyScenario::draw()`. Clients can add any number of draw passes each with an optional `preDraw()` and `postDraw()` callback. The example `programming_examples/diguy_plugin/diguy_plugin_sensor_fusion` exercises the draw pass functionality.
- ♦ Calling `diguyCharacter::set_speed()` while a character is transitioning into a still motion, will now no-op. Previously setting speed to 0 when a character played certain dead transitions could wind up freezing the character in mid fall.
- ♦ Motions that are slowed down will automatically interpolate.
- ♦ Increased the maximum speed of I-Guy strafing.
- ♦ Added `diguyCharacter::set_local_to_world_orientation_matrix()` experimental function allowing a rotated plane to be used for character movement displacement calculations. This is useful for geocentric rendering and animation.
- ♦ Improved `diguyCharacter::force_action()` with a speed, to no longer speed warp transition animations.
- ♦ Improved IK system to properly handle uniformly scaled characters.
- ♦ Changing character appearance no longer resets a specified hand item, but will still change to a new default if no item is specified.
- ♦ Improved Vehicle based I-Guy to turn camera better, ground clamp properly and be generally nicer.

- ♦ New draw pass system allows client applications to register draw passes with DI-Guy that are called instead of the native rendering when calling `diguyScenario::draw()`. Clients can add any number of draw passes each with an optional `preDraw()` and `postDraw()` callback.

Camera

- ♦ Merged 12.5.4 camera functionality dealing with supporting camera roll into 13.0.1.
- ♦ Better handling of camera smoothing, keeps camera motion smooth while resolving jitter when tracking characters.

Networking

- ♦ Upgraded to VR-Link 5.1.2.
- ♦ Added debugging visuals for received actions.
- ♦ Added culling to debugging visuals to keep performance reasonable.
- ♦ Added `diguyCharacter::set_is_network_paused()` allows end users to duplicate network paused behavior without using our network layer.
- ♦ Improved warning messages from Lua when invoked from script beads and a couple other fixes.
- ♦ Networking module will always send out action speed as part of a custom PDU. This was only happening if it was a non-default speed.
- ♦ Added new z velocity threshold option for forcing a packet update, this can help with up and down movement over the network.
- ♦ Added a new UI for editing packet thresholds.
- ♦ Rewrote data that went out describing human articulated aiming data. Previous solution was world space xyz positions, that wouldn't really work in other coordinate systems. We are now sending more portable azimuth and elevation data that should be cleaner. A similar change was made for gaze descriptions.
- ♦ DI-Guys have marking text set to their names by default.

Instancing

- ♦ Updated instancing to respect texture swaps. Can be toggled off for increased performance.
- ♦ Added `instancing_respect_graphics_state_switch` to `bdiGraphicsInitGraphicsAPI` struct.

API Changes

- ♦ New API functions:
 - `diguyApp::add_draw_pass()`
 - `diguyApp::remove_all_draw_passes()`
 - `diguyCharacter::get_lod_ranges()`
 - `diguyCharacter::get_num_lods()`
 - `diguyScenario::calculate_character_lods()`
 - `diguyGraphicsInstanceGroup::get_state_index()`
- ♦ New class: `diguyDrawPass`.
- ♦ The example `./programming_examples/diguy_plugin/diguy_plugin_sensor_fusion` exercises the draw pass functionality.
- ♦ To run the DirectX 9 programming example, you must have the DirectX9 Nov. 2007 runtime. For details, go to <http://www.microsoft.com/en-us/download/details.aspx?id=317>

Graphics API Changes

- ♦ In OpenGL reference Code, added nicer support for asynchronous texture loading proxy textures. Please see `diguyOglGraphicsTexture::build_opengl_texture()` for updated sample code.
- ♦ Fixed issues with `diguyGraphicsLinks::get_rotation_euler()`, and `diguyGraphicsLinks::get_rotation_quat()` to return valid values even if the underlying data is still in eulers or quats (currently happens with some vehicles.)
- ♦ OSG reference Code:
 - Large rewrite of culling/lod system, a new custom node early outs culling based on simplified bounding volume. Also short circuits LOD picking and OSG character traversal. drops cost of culling by 90%.
 - Added custom TBO class to remove limitation of instancing only working on OSG 3.2 or greater.
 - Changed `diguyOsgGraphicsShape::update()` to check for transposed shader matrices before pushing data to OSG instead of via an `#ifdef` guard.

Bug Fixes

This release fixes the following problems:

- ♦ Fixed animation rotation interpolation on joints when imported into motion editor via collada.
- ♦ Animation system regression fix to 12.5, restored a while loop that was needed when update loop was fed a long time step.

- ♦ Fixed potential issue in animation system when update loop was fed a long time step.
- ♦ Merged 12.5.4 camera functionality dealing with supporting camera roll into 13.0.1
- ♦ Fixed calculation of script/decision bead position when multi-arc rep motion instances are used.
- ♦ Fixed hard crash when networked entity goes on/off-line and changes appearance.
- ♦ Changed `diguyOsgGraphicsShape::update()` to check for transposed shader matrices before pushing data to OSG instead of via an `#ifdef` guard.
- ♦ Fixed fatal errors in `lqt` when dealing with certain widget types.
- ♦ Regression fix, `diguyCharacter::create_and_force_bridge_path()` didn't work properly due to action editing improvements.
- ♦ Fixed hard crash interaction between destroying networked characters and instancing.
- ♦ Fixed networking sending of velocity of characters playing speed scaled motions.
- ♦ Fixed asynchronous texture load shader ugliness when switching between shaders when spec/normal maps haven't been loaded yet.
- ♦ Fixed bug where ground clamped vehicles weren't following the terrain.
- ♦ Fixed a crash that could occur when a character playing an animation had its character type changed.
- ♦ Fixed issue where `diguy 12.5` style DIS packets were being sent.
- ♦ Fixed bug where characters that had their default hand item set and then the item that the default actually equaled could lose what was in their hand.
- ♦ Fixed issue where capitalized FLT extension could confuse material system.
- ♦ Fixed issue where dragging an AI could reset their mind and confuse their location.
- ♦ Fixed network teleporting on reset and other large discontinuous position updates.
- ♦ Fixed a number of drift and adaptive guide parameter bugs.
- ♦ Fixed a number of speed scale and velocity scaling issues. Regressions to 12.5 as well as long standing issues.
- ♦ Fixed force action with a non-default speed with a scaled character/or different skeleton then expected (for example, female soldier).
- ♦ Fixed a bug where up vector setting was being ignored on `vehicle_09's` on reset.
- ♦ Fixed a bug where vehicles on reset could have oddly behaving wheels.
- ♦ Fixed bug where changing turreted vehicle's appearances could have incorrect aiming/firing behavior.
- ♦ Regression fix, preloading a character type did not properly preload animation data.
- ♦ Fixed bug with reporting initial action for a path with an action that didn't have a transition out.
- ♦ Fixed hard crash when networked entity goes on/off-line and changes appearance.

- ♦ Fixed hard crash interaction between destroying networked characters and instancing.
- ♦ Fixed issue where DI-Guy12.5 style DIS packets were being sent.
- ♦ Fixed network teleporting on reset and other large discontinuous position updates.
- ♦ Fixed a number of drift and adaptive guide parameter bugs.
- ♦ Fixed bug where teleporting vehicles did not have proper orientation.
- ♦ Fixed networking sending of velocity of characters playing speed scaled motions.
- ♦ Fixed asynchronous texture load shader ugliness when switching between shaders when spec/normal maps haven't been loaded yet.
- ♦ Fixed issue where dragging an AI could reset their mind and confuse their location.
- ♦ Fixed small nav mesh glitch which caused a failure to detect some impassable polygons.
- ♦ Fixed Doppler Effect bugs: improper velocity updates caused stuck/distorted sounds.
- ♦ Fixed fatal errors in lqt when dealing with certain widget types.

Known Problems

DI-Guy 13.0.1 has the following known problems:

- ♦ The first time you load the IGuy_Interaction scenario, the terrain may disappear at certain camera angles. If you reload the scenario, the problem is no longer present.