

The logo features a stylized white 'G' inside a dark red circle with a glowing effect, followed by the text 'GENESYS™ 3.0' in a bold, black, sans-serif font.

GENESYS™ 3.0

The title 'New Features Guide' is centered on the page in a large, bold, black serif font. The background of the page is a light gray grid pattern with faint, semi-transparent images of software interface elements like charts and data tables.

New Features Guide



Copyright 2014 Vitech Corporation.
All Rights Reserved

GENESYS 3.0 New Features Guide

Copyright © 2011-2014 Vitech Corporation. All rights reserved.

No part of this document may be reproduced in any form, including, but not limited to, photocopying, translating into another language, or storage in a data retrieval system, without prior written consent of Vitech Corporation.

Restricted Rights Legend

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7013.

Vitech Corporation

2270 Kraft Drive, Suite 1600

Blacksburg, Virginia 24060

540.951.3322 FAX: 540.951.8222

Customer Support: support@vitechcorp.com

www.vitechcorp.com

GENESYS™ is a trademark of Vitech Corporation.

Other product names mentioned herein are used for identification purposes only, and may be trademarks of their respective companies.

Publication Date: April 2014



TABLE OF CONTENTS

Integrating State and Behavior – The State Transition Diagram 5

Connecting with IBM® DOORS® 5

Connecting with Microsoft® Excel® 7

 Changing Data 7

 Examine Data 7

Exposing Critical Design Parameters 7

Exploiting the Model – Auto-Checking with Model Diagnostics 9

Refining diagrams – Richness, control, and utility 10

 Drag-drop visual enhancements 10

 Inserting Generic Graphics 10

 All Entities tab 10

 Mouse-over Details 10

 Nudging 11

 Grid support 11

 Drag snapping and guides 12

 Inserting Notes 12

 Inserting Generic Shapes 12

 Layout dialog simplification 12

 Preferences quick access on ribbon 12

 Image Library 12

 Opacity 13

 Z-Order support 13

 Toggle diagram Pan and Zoom tool 13

Performance and Scalability Improvements 13

 64-Bit Support 13

 Simulation Load and Reset 14

 Database access of folders, entities, relationships, and attributes 14

Release of Help Files 14

Expanding the Model 14

 Migrating Projects from pre-v30 Schemas 14

 General Changes Included in All Schemas 15

 Class changes 15

 Additions 15

THIS PAGE INTENTIONALLY BLANK

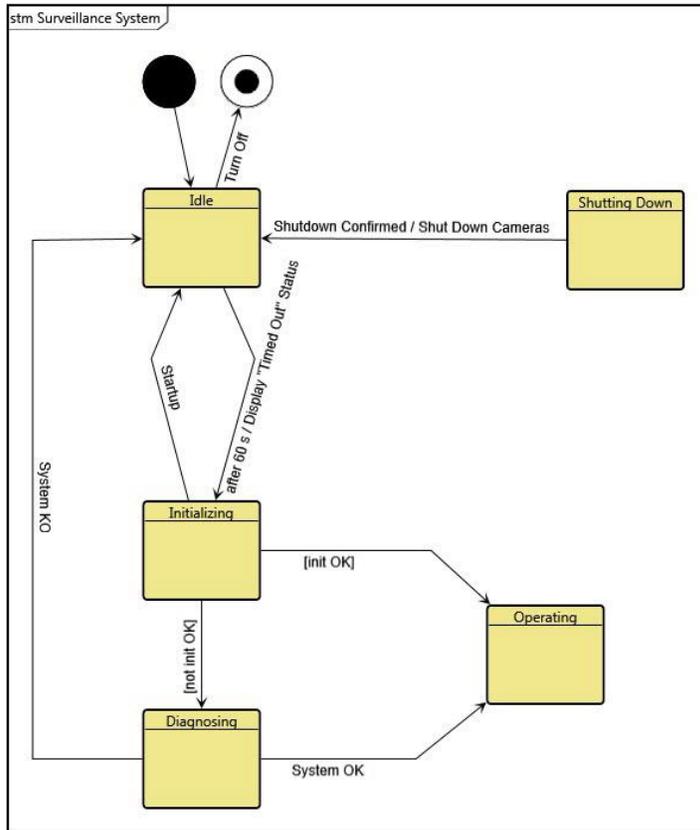
INTEGRATING STATE AND BEHAVIOR – THE STATE TRANSITION DIAGRAM

State models and behavioral models are often used to represent the same concept space but do so in completely different ways such that a common underlying model generally can't drive both...until now. GENESYS 3.0 introduces the state transition diagram side-by-side with all of the behavioral representations you have leveraged in the past. It's up to you whether you model a specific concept using state transition or behavioral approaches. GENESYS 3.0 simply expands your range of options to model and represent the way you choose.

States – and state transition diagrams – are a specific modeling approach available for use and decomposition to any level desired. You can model completely using state concepts, begin using state concepts which then incorporate lower-level behavior, or model from beginning to end using behavior.

Presented in a freeform layout, the state transition diagram represents states, the transitions that connect them, and the events that trigger transitions. The implementation aligns with the SysML representation:

- The state transition diagram is opened on a State. Child states are shown as rounded rectangles.
- The lines between states are Transitions. Transitions are directional, exiting from one state and entering another. States may have multiple transitions, but transitions are limited to a single entry and a single exit.
- Transitions are triggered by Events and serviced by Functions / Operational Activities. Though it is possible to display Transition information on the transition line, most commonly the transition line is represented by some combination of the triggering event(s), the guard condition (shown in square brackets), and the function which services the transition.
- The state nodes themselves can optionally display the entry, exit, and do (root) functions related to them.



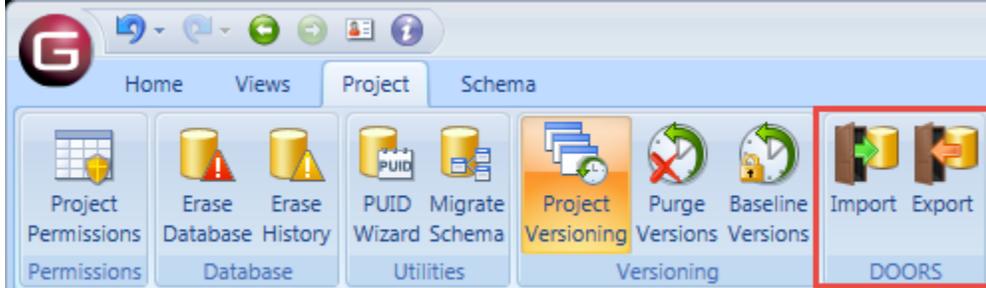
CONNECTING WITH IBM® DOORS®

DOORS has long been a defacto standard for requirements management. Though GENESYS provides rich capabilities spanning from concept definition through test – certainly encompassing requirements management – many teams find themselves mandated to house their requirements in DOORS. Over the years, many different solutions have been used to help transfer information between these two mainstays in the systems engineering world – some from Vitech, some third party, some formal, and most ad hoc. With this release, a new standard approach is available to simplify and streamline this connectivity.

GENESYS 3.0 New Features Guide

Leveraging the GENESYS API, you can now connect to DOORS via Open Services for Lifecycle Collaboration (OSLC.) Developed by IBM, OSLC serves to interface between multiple tools and enables bi-directional exchange of data with DOORS.

You'll find DOORS import and export icons on the Project tab.

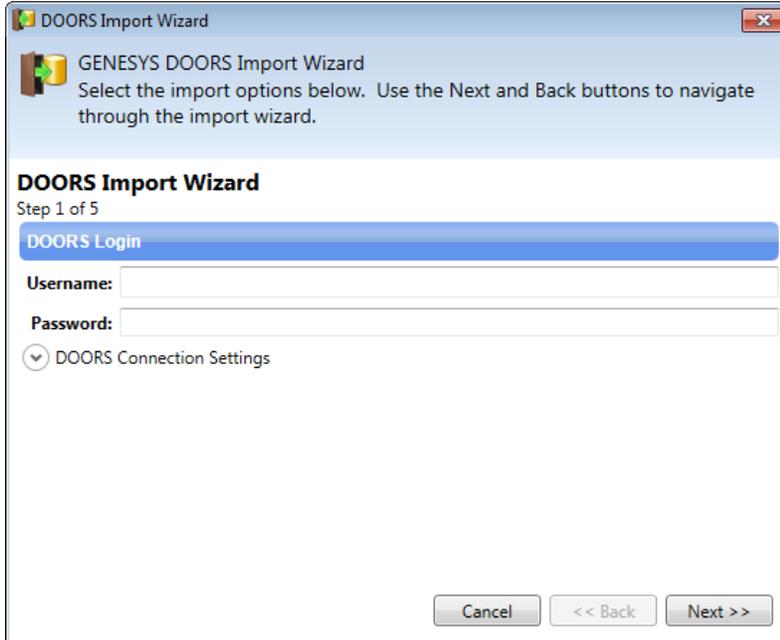


The DOORS Connector supports three fundamental concepts of operations when leveraging both

GENESYS and DOORS in a joint environment:

GENESYS for Requirements Generation. In this concept of operations, the user is focused on the operational and problem analysis. The user has not started with configuration controlled requirements. Instead, the generation of system requirements is the objective. GENESYS is the initial tool, operational analysis proceeds, and the final system-level requirements are to be baselined in GENESYS. The fundamental need in this case is a one-way export of requirements from GENESYS to DOORS.

DOORS as the Initial Requirements Source. In this concept of operations, the initial requirements (system-level or otherwise) have been captured in DOORS. The team wishes to use this as the starting point for systems analysis in GENESYS. Though DOORS does not have a major ongoing role in this scenario, it is possible that requirements changes in DOORS will occur. The fundamental need in this case is a one-way export of requirements from DOORS into GENESYS with subsequent export of changes or change orders.

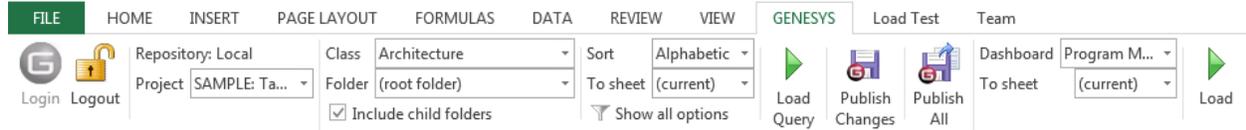


DOORS as Authoritative Requirements Repository. In this concept of operations, the initial requirements have been captured in GENESYS. The systems engineering component of the team wishes to use GENESYS for systems analysis, but DOORS will serve as the authoritative requirements repository – at the system and subsystem level. As lower-level requirements are generated in GENESYS, they are reflected in the DOORS repository. In this case, the GENESYS involvement with the project begins with a one-way export of requirements from DOORS into GENESYS. After that, the primary need is the publication of lower-level requirements from GENESYS to

DOORS. In both directions, there exists the possibility of change orders. It is also possible that once requirements are exported into DOORS, the corresponding GENESYS objects may be baselined to avoid further property changes.

CONNECTING WITH MICROSOFT® EXCEL®

Further leveraging the GENESYS API, we've built a connector to Microsoft Excel. This separately installed extension will add a GENESYS tab to the ribbon in Excel. From there, you can login to your GENESYS repository and access data.



Log in to your project from the ribbon for simple queries, such as all elements in a class or folder. Then click the Load button to bring your GENESYS data into Excel. This simple query will return the class, folder, number, and name.

If you need a more complex query, or would like to change what attributes and relations are returned, then you can expand the query builder using the Show all options command. Here you can use the multi-select drop-downs to choose exactly what data you want to see in Excel. You can even apply a GENESYS filter or sortblock.

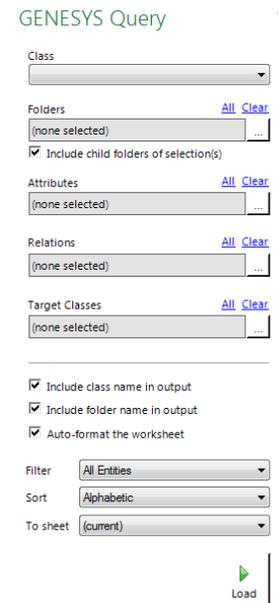
Once you have your data in Excel, there are two primary paths. Some users will want to change data, others will want to examine the data.

Changing Data

If you can edit Excel data, then can edit your GENESYS data in Excel. Just make the changes in Excel, then click the Publish Changes button in the toolbar.

It's important to note the difference between the Publish Changes button, and the Publish All button. The behavior is exactly what you might expect them to be. Publish Changes will take only what you've changed in Excel and push that data to GENESYS as the current value. Publish all will take all data in Excel and push that data to GENESYS as the current value. We've added both buttons because we know you'll want the flexibility.

If you have other users working in GENESYS, then using either Publish command will overwrite the GENESYS data. So, if you have another user who changed an attribute, then you push your change of the same attribute to GENESYS, last in wins. Remember that many attributes are versioned in GENESYS, in which case change history will be tracked.



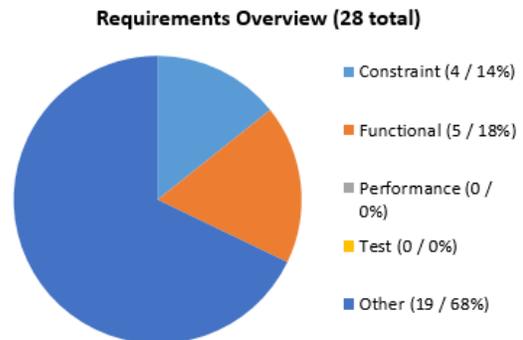
Examine Data

Getting your data out to Excel opens up a wealth of options for the examination of your data. Many of our customers have been doing this for years with Excel dashboards they built on their own.

We're now sharing with you a set of dashboards in Excel that we've built for you. To see your data in a dashboard, load the data, then in the ribbon, select the dashboard you want to see!

EXPOSING CRITICAL DESIGN PARAMETERS

Systems engineering requires a host of information – objects, their interrelationships, their structures, and a mountain of properties. Some of these are design descriptors: numbers, types, rationale, etc. When



GENESYS 3.0 New Features Guide

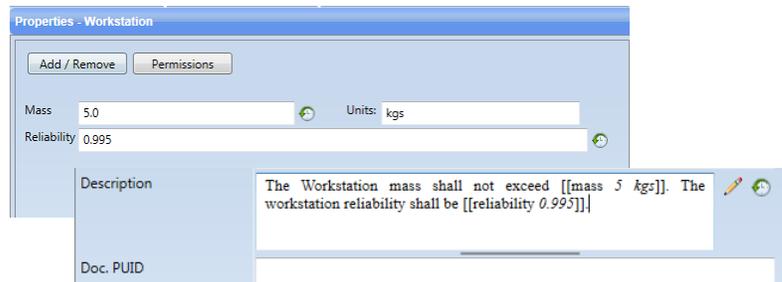
leveraging a tool, some are artifacts of the design environment itself: creator, modification time, and more. Buried within this mass of information – sometimes literally embedded within blocks of text itself – is the most critical information of all: design-dependent and design-independent parameters.

There have always been approaches to manage this key information. In the simplest and most frequent case, teams simply incorporate these values within descriptive text opting for convenience and the benefits of natural language. In the most complex and rarely used approach, key values are identified and managed as first-class elements, often at great inconvenience. GENESYS 3.0 introduces design parameters, exploiting the power and value of separately managed values with the convenience of natural text.

Within GENESYS, parameters are similar in many ways to attributes. Both exist at the individual element level. Both are permission controlled. Both are most frequently managed through the property sheet. There are several key differences:

- Attributes are descriptive aspects of the design (description, rationale, etc.) as well as GENESYS environment data. Parameters are values – generally numeric with units - that define the design. These could be targets and thresholds for aspects of performance or current design values.
- Attributes are pre-defined by the project administrator (though extensible at any time). Parameters are user-defined at “runtime.” Whenever you decide you have a value that should be independently tagged and managed, you can define a new parameter.
- Attributes can leverage the full range of data types in GENESYS. Parameters are limited to an appropriate subset: integer, float, date, time, timestamp, and string.
- Attributes are stand-alone values (within the context of the element). Parameters generally have an associated units field. With the exception of a string parameter, it is generally appropriate to specify a unit (weight in kilograms, temperature in Celsius, etc.).
- Attributes are *shown* for every element in a given class. Parameters are *available* for every element in a given class but only shown when populated. This allows you to establish parameters such as mass (appropriate for a hardware component) and lines of code (appropriate for a software component) without cluttering all elements in a class.
- Versioning is optional for attributes. Versioning is always enabled on parameters (when versioning is enabled for the project). Parameters represent key design aspects that should be tagged and managed carefully through the life of the project.
- Most importantly, parameters can be embedded in text attributes for the given element. This enables you to expose and manage this critical design data with the ease of natural text.

When working with parameters, you will see a third tab on the element property sheet. Here you can select one of the pre-defined parameters (over 70 have been established in the default schemas) or create one of your own. When editing a text field, right-click to embed a parameter – either one that already exists or one that you wish to create on the fly. When working within the tool, the current



parameter is shown in square brackets with the current value in italics – `[[mass 5.0 kgs]]` – to visually highlight the value. In outputs – whether working documents, formal specifications, or web publication – the parameter field is replaced with the current value in a seamless manner. (And if you are wondering, parameters are completely permission-controlled so those without read permission for a parameter will see the parameter indicator but not the value itself – even within GENESYS.)

Leveraging parameters in this way truly enables the best of both worlds – the power, explicitness, and management of models with the ease and simplicity of text.

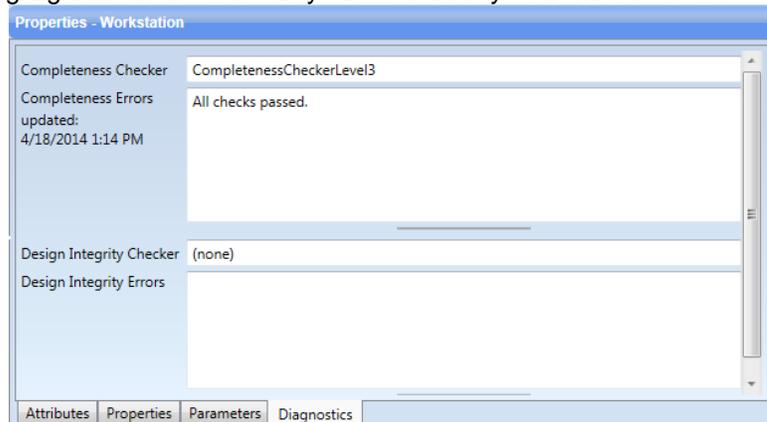
EXPLOITING THE MODEL – AUTO-CHECKING WITH MODEL DIAGNOSTICS

Teams derive a great deal of benefit simply from adopting a model-based approach. They expose their system model for greater team alignment and understanding (in document-centric approaches, the model is still there, it's simply obscured by the communication medium). They gain insight from the thought process of developing the model. They gain richness and explicitness not generally achievable in document-centric approaches. Plus they achieve savings in countless ways – from increased productivity, from early detection of errors, from improved and streamlined impact analysis, and even from the elimination of production costs for specifications. All that said, the greatest value of the model-based approach comes not from developing the model but from exploiting the model.

To complement the embedded ability to directly simulate your model for dynamic verification, GENESYS 3.0 introduces a rich framework of embedded model diagnostics. With a library of over 70 completeness checks and countless design integrity diagnostics, all in a customizable framework for including your own rules, these model diagnostics go far beyond simple diagram-centric checks. Put simply, these diagnostics aid in the bookkeeping and validation of systems engineering, freeing you to focus your valuable time on the critical inspiration of systems engineering.

The first component of model diagnostics – the completeness checks – evaluates your model completeness on the fly. Much like TurboTax® dynamically informs you of missing data, GENESYS will tell you which key attributes and relationships have not yet been completed. Of course, not all rules are valid on day one of your project. As your design matures, your completeness checks should become more sophisticated and robust. With this in mind, for each project you specify one of three levels of completeness checks – from none to rich – tuning to your specific needs.

The second component of model diagnostics – the design integrity checks – is a pre-defined (and extensible) set of script-based rules to highlight inconsistencies in your model. As you work across the systems engineering domains or across levels of detail, it's trivial to introduce inconsistencies as your problems become more complex. With the push of a button, you can highlight these issues for a given element, a folder, a package, or the entire project. Unchecked, these inconsistencies can doom your project or the system you produce. Addressed, these inconsistencies can be easily managed.



Both completeness and design integrity checks are implemented via script in GENESYS. This provides a framework into which you can plug your specific rules. Whether reflecting a style guide for your project team, a corporate or customer standard, or an extended underlying schema, complying with your custom rules is critical for project success. By providing an easily extended framework alongside the existing capability, your project can quickly implement the specific diagnostics you need.

The results of model diagnostics are shown on a fourth tab added to the element property sheet. A new pre-defined filter titled Diagnostic Errors has been included to quickly identify those elements with diagnostic issues. All of the diagnostic results are available via reporting so that you can generate the work products needed for online or offline review.

One of the rules of systems engineering is that it costs 3 to 10 times as much to correct an error for each successive stage it moves through the life cycle (assuming it can still be resolved without degraded system performance). Helping to identify these issues early on and expose them for your review is priceless.

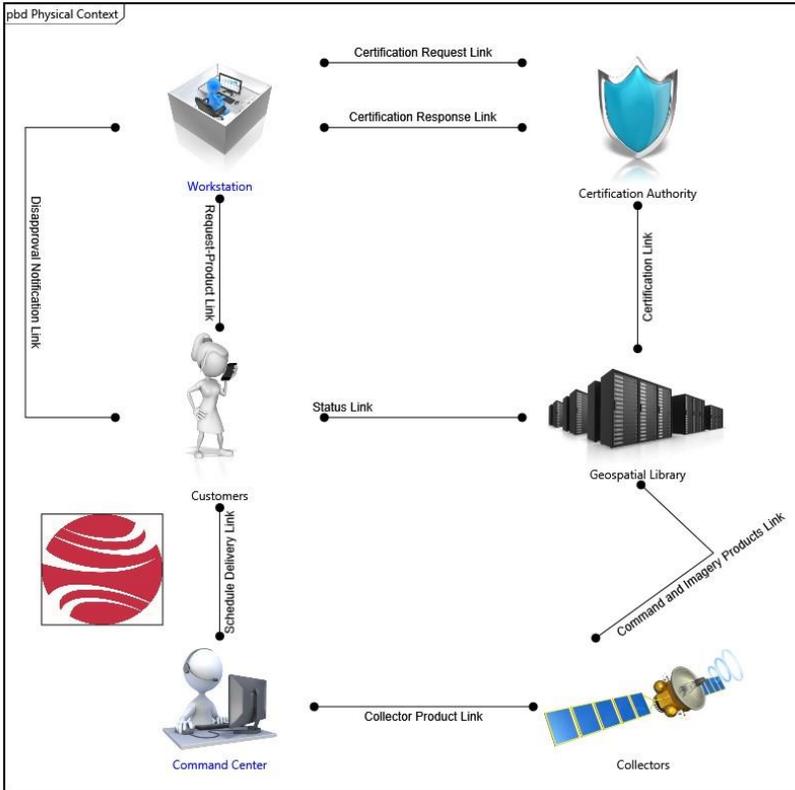
REFINING DIAGRAMS – RICHNESS, CONTROL, AND UTILITY

GENESYS 3.0 represents the transformation of visualization and communication with diagrams. While every bit as technically correct, diagrams now have the flexibility needed to help you communicate most effectively.

Drag-drop visual enhancements

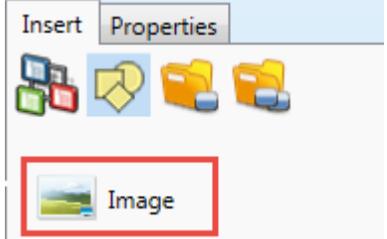
When you drop an item from the Toolbox onto a diagram, there are rules that govern where you are allowed to drop the item. If you are not highly familiar with the diagram, it can be challenging to determine exactly where you can drop that item. GENESYS now highlights in blue the position where a selection

can be dropped from the Toolbox onto a diagram.



Inserting Generic Graphics

With the introduction of image representation for elements in GENESYS 3.0, the natural next question is “can I put a picture of



X on my diagram?’ There are many useful cases for this, most notably team or project logos as independent graphics on diagrams.

Shown on the Toolbox Utilities tab, the Image command prompts you to select an image from either the stock image libraries or from your own file collection. The image is then

inserted at full size in the middle of the screen. You can then easily drag the graphic to the position desired.

All Entities tab

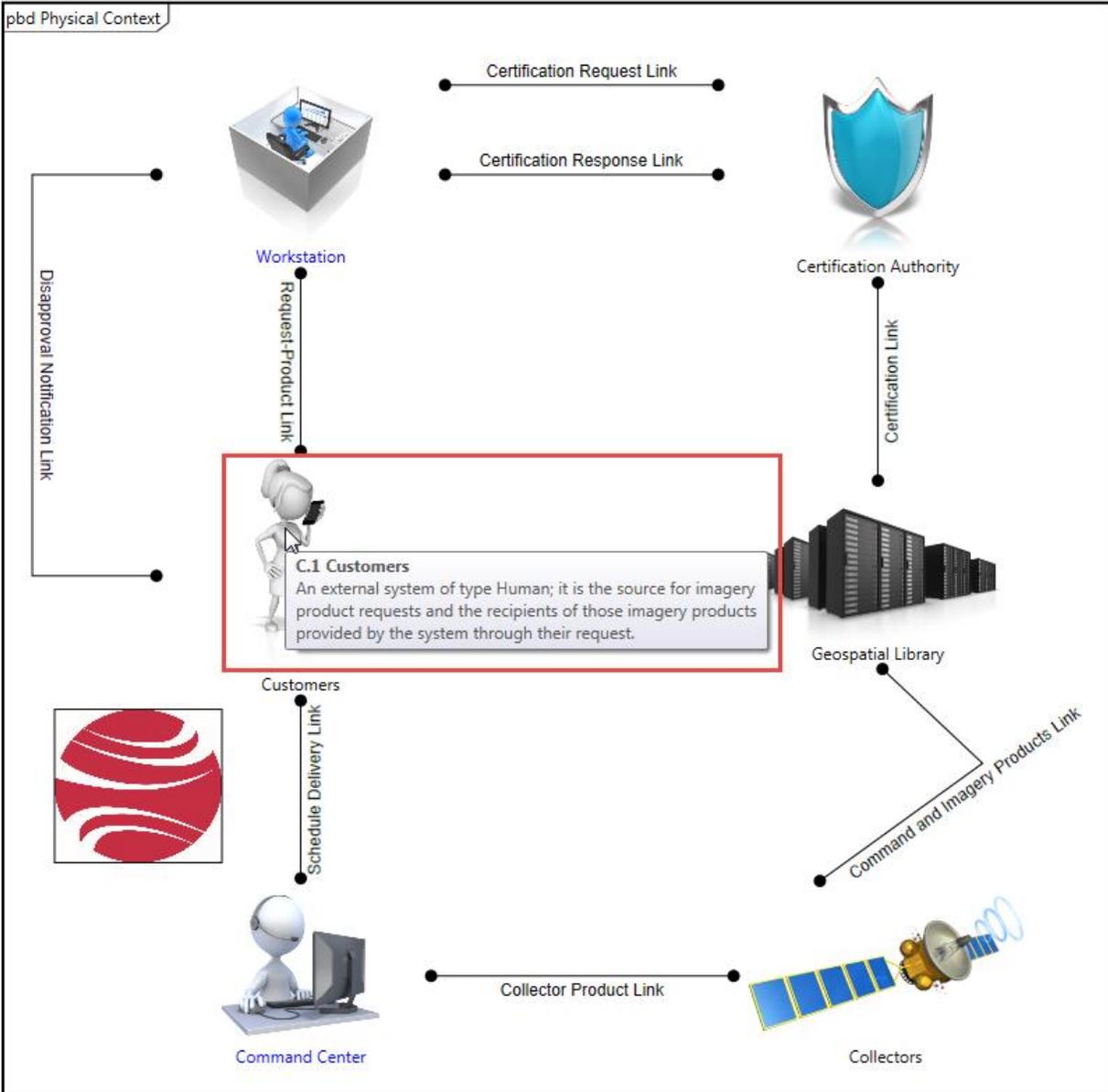
While each diagram focuses on specific model aspects, we are always looking at representations of the real underlying model. In many cases, we may want to create a relationship immediately – even if it won’t be shown on this particular view – rather than flipping from view to view. For this reason, we have added the All Entities tab. Drag any element onto a mode, and GENESYS will prompt you with the possible ways to relate the two elements. Not only does the ease manipulation, it significantly lowers the learning curve on the language of MBSE.

Mouse-over Details

We’ve expanded the notion of tooltips onto diagrams. Mouse-over an icon in a diagram and GENESYS will display the number, name, and description of that element. Move your mouse away and the

information disappears. This is great for getting key details quickly without the need to navigate away from the view.

Nudging



Sometimes adjusting the placement of a diagram entity by mouse just isn't working for you. In the event you prefer adjusting via the keyboard, you can now move entities with the keyboard. You can move entities on a diagram using either larger jumps, or pixel by pixel (use the Control key plus the arrow keys to move in the smallest increment.)

Grid support

Aligning nodes on your diagram is a lot easier when you have a visual reference. We've added a simple grid to all diagrams to help.

By default, this grid is light blue. If you change the background color of the diagram, then grid color will adjust so that you have sufficient contrast.

This grid can be toggled on/off in the user preferences.

Drag snapping and guides

With such a wealth of diagram usability enhancements in GENESYS 3.0, it just wouldn't be complete without helping you to better align nodes. When dragging a node, if you get in close proximity to aligning with another node, alignment guides will appear and you can drop the node in alignment. It will pick up on center, middle, top, bottom, left, right, and the background grid.

Inserting Notes

A note on a diagram can convey key context information, pose a question, or capture any key information you need to log on the fly in a simple text box. This text box becomes part of the view, saved immediately, and available to all users who have permission to view the diagram.

Available from the Toolbox Utilities tab, drop the note icon on the diagram. GENESYS will prompt you to enter note text, then place a sticky note on the diagram.

Inserting Generic Shapes

You now have the ability to insert generic shapes on diagrams. Rectangles, rounded rectangles, ellipses, circles, stars, triangles, and more are all supported. Like notes, diagram shapes also can optionally include text. This enhances the representation and communication value by emphasizing key aspects or implying groups, clusters, etc.

From the diagram Toolbox, click on the Utilities icon, then drop the desired shape onto your diagram.

Layout dialog simplification

We agree that giving you a variety of diagram layout options is good. Sometimes we provide so many, that we actually degrade the user experience. Such is the case with the layout options. We've removed a few of the layout options to simplify and streamline the user experience.

Preferences quick access on ribbon

Many users have found that they use some diagram preferences more than others. Some users even toggle on and off frequently. These most used preferences have been made available on the ribbon so you'll have them at your fingertips each time you open a diagram.

This preference setting will apply to all diagrams, but are user-specific.

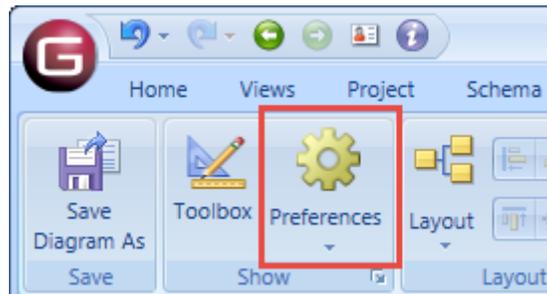


Image Library

In GENESYS 2.0 we gave you the ability to replace a node on a diagram with an image. In this release we extended that to allow you to place an image anywhere on a diagram. These greatly enhance your opportunities for communication. But this capability is only as valuable as the images you have available.

To make sure you have a wealth of images at your fingertips, we've added a stock image library in GENESYS. With a library of over 3,000 images you should be able to meet most image needs.



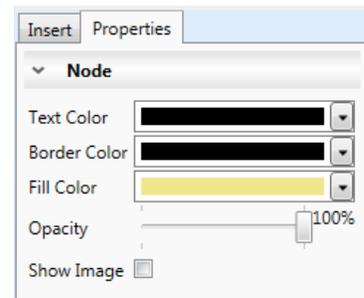
Further, with an image library of that size, text descriptions just won't do. When you drop down the image list, you'll see a preview of the image. You can quickly scroll through the list to find the image that best meets your needs.

Opacity

To give you even more power in enhancing the communication power of your diagram, you can now decrease the opacity of the entities/items that you do not want to focus on.

Select the desired element, then drag the slider on the Properties tab left or right to adjust.

The default setting will always be 100% until you adjust it otherwise.



Z-Order support

Layering objects, especially shapes, on a diagram allows you even more communication power. GENESYS now supports adjusting Z-order, bringing elements forward and backward on the diagram. You'll find Z-order options in the right-click menu.

Toggle diagram Pan and Zoom tool

In the top left corner of every diagram is a simple navigation tool. This tool allows you to quickly move to viewing other areas of the diagram and is especially helpful when working on a very large diagram. If you are working on a smaller diagram, you might find this navigation tool to be something you'd rather not see.

A quick setting changed in the user preferences can toggle the navigation tool on or off.

PERFORMANCE AND SCALABILITY IMPROVEMENTS

GENESYS 3.0 includes a host of new capabilities and refinements throughout its engineering environment. In addition to the major capabilities noted before, we made several improvements to performance and scalability.

64-Bit Support

The terms 32-bit and 64-bit refer to how much of your computer's memory (RAM) can be used by applications. A 32-bit application can use up to 4GB of RAM. This is fine for many users.

If you have a particularly large data set, you may find that a bit more memory would be helpful. A 64-bit application can use a theoretical maximum of 16 exabytes of RAM – much more RAM than your computer actually has. This means you can work faster with larger data sets.

GENESYS 3.0 New Features Guide

While GENESYS has always been able to run on 64-bit computers, it has been a 32-bit application and Windows handled the compatibility. We now have 64-bit installers for our customers to use.

Simulation Load and Reset

Loading data in the simulator takes a few moments as the model is loaded. That loading takes a little less time now that our developers have improved the performance of simulation loading.

This same loading occurs when you use the Reset button, and you'll find that faster as well.

On a small model, you might not notice but if you have a large data set, it's a pretty big deal!

Database access of folders, entities, relationships, and attributes

Not only are we loading simulations faster, we're loading everything fast. Folders, entities, relationships, and attributes all load faster!

RELEASE OF HELP FILES

Great tools are great. But great tools with great documentation are...better than great! We're documenting every capability of GENESYS, every button, every diagram, and every nook and cranny. At initial release, we'll be including the first installment of the Help documentation. Click the help icon in the top right corner of GENESYS to access it.

We're adding content incrementally with each service pack release. So as GENESYS 3.0 matures, be sure to check back in the help tool for new content.

EXPANDING THE MODEL

With each release, Vitech continues to enhance the underlying schemas to reflect our lessons learned and best practices in the application of MBSE.

A complete list of all schema changes is included at the end of these release notes. At the highest level, the key schema changes include:

- Adjustments to fully support State Transition diagram
- Expanding to support parameters
- Addition of default images for schema classes
- Removal of a duplicative attribute

Migrating Projects from pre-v30 Schemas

Given the nature of the schema changes made in the v30 base and DoDAF schemas, minor project migration is required to transform specific classes and specific attributes. To best support this migration, GENESYS 3.0 provides a schema migration utility on the Project tab.

This utility is targeted at the standard Vitech schemas. If your project uses a customized schema, additional migration support may be required. If you have made schema extensions and would like to understand what special steps – if any – are required to support your extensions, please contact [Vitech Customer Support](#) with a copy of your extensions. We are happy to review these and advise you in your migration accordingly.

Note: You do not have to migrate your v20 schema in order to make use of GENESYS 3.0. You should assess your project needs, your project lifecycle, and the changes present in the v30 schemas to determine if, and when, to migrate your schemas. Project teams nearing a major milestone or approaching conclusion should strongly consider remaining with their current schema. We recommend that others consider moving to the v30 schemas to take advantage of the latest improvements in the MBSE language.

General Changes Included in All Schemas

Class changes

- Nexus (and its subclasses)
 - Added target class State to generated by relations
- State
 - Added target class Nexus (and its subclasses) to generates relation
- ImplementationUnit (and its subclass)
 - Removed values attribute (Rationale: With the addition of parameters, this attribute was duplicative.)

Additions

- In the schema, a node image can now be associated with a class.
- A default set of parameters have been added.
 - Link – availability, downtime, errorRate, failureRate, pressure, pressureGradient
 - Requirement – adaptability, flexibility, interoperability, missProbability, repairTime, sustainability, traceability
 - Component – availability, decisionThreshold, density, downTime, durability, errorRate, failureRate, falseAlarmProbability, force, height, length, loadingCoefficient, logisticalDelay, maintainability, mass, missProbability, MTBF, MTTF, MTTR, power, pressure, pressureGradient, reliability, repairability, repairTime, resilience, robustness, scalability, speed, temperature, tensileStrength, torque, volume, width
 - Function – decisionThreshold, errorRate, falseAlarmProbability
 - VerificationRequirement – adaptability, availability, decisionThreshold, density, downTime, durability, errorRate, failureRate, falseAlarmProbability, flexibility, force, height, interoperability, length, loadingCoefficient, logisticalDelay, maintainability, mass, missProbability, MTBF, MTTF, MTTR, power, pressure, pressureGradient, reliability, repairability, repairTime, resilience, robustness, scalability, speed, sustainability, temperature, tensileStrength, torque, traceability, volume, width

THIS PAGE INTENTIONALLY BLANK



Vitech Corporation
2270 Kraft Drive, Suite 1600
Blacksburg, Virginia 24060
540.951.3322 FAX: 540.951.8222
Customer Support: support@vitechcorp.com
www.vitechcorp.com