GENESYS™2023

SBE Vision Adapter Guide

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

Publication Date: June 2023

TABLE OF CONTENTS

GENESYS Adapter to the SBE Vision Digital Thread Platform 1
Definitions 2
Capability and Usage Overview 5
Export (Publishing) 6
Import (Subscribing) 7
One-way Data Transfer and Round-Tripping 8
One-Way Data Transfer 8
Subsequent Update to the ASoT 9
Round-Tripping 10
Using the GENESYS SBE Vision Digital Thread Adapter 11
Installation of the SBE Vision Schema Extension 11
Accessing the GENESYS – SBE Vision Digital Thread Adapter 13
Procedures for performing Imports, Exports, one-way Transfers, and Round-tripping 13
Export… 13
Import… 14
One-Way Data Transfer 16
Round-tripping 16
THE GENESYS – SBE Vision DT Platform Adapter UI 17
Input Screen 1 – Viewpoint, Credentials, SBE Connection Settings 17
Input Screen 2 – Specifying Import or Export 18
Input Screen 3 – Specification of the Data Location on the SBE DT 19
Input Screen 4 – Scope and Location 20
Results Screen 1 – Operation Summary and Issues 21
List of Non-isomorphic transformation rules 22
GENESYS Entity’s Parameters to SBE DT Parameter class entities 22
Egress / Publish 22
Ingest 23
SBE Visions DOORS Classic & DOORS NextGen Integrations 24
General Notes on Data Processing Rules 24
Attribute Type Mappings 24
Relationship Attributes 25
Embedded Images and Tables 25
Inherited Parameters and Attributes 25
Parameter Substitutions 25
Parameter Bindings 26

CUSTOMER RESOURCE OPTIONS

Supporting users throughout their entire journey of learning model-based systems engineering (MBSE) is central to Vitech's mission. For users looking for additional resources outside of this document, please refer to the links below. Alternatively, all links may be found at www.vitechcorp.com/online-resources/.

Webinars

Immense, on-demand library of webinar recordings, including systems engineering industry and tool-specific content.

Screencasts

Short videos to guide users through installation and usage of GENESYS.

A Primer for Model-Based Systems Engineering

Our free eBook and our most popular resource for new and experienced practitioners alike.

Help Files

Searchable online access to GENESYS help files.

Technical Papers

Library of technical and white papers for download, authored by Vitech systems engineers.

Technical Support

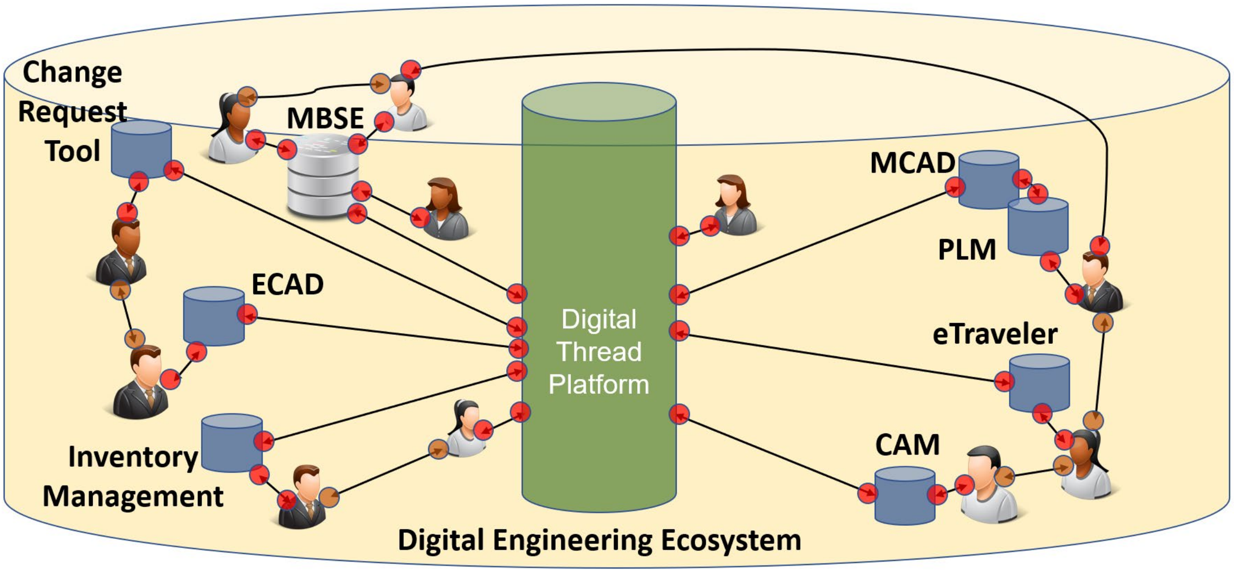
Frequently Asked Questions (FAQ), support-ticket web form, and information regarding email, phone, and chat support options.

GENESYS ADAPTER TO THE SBE VISION DIGITAL THREAD PLATFORM

Many companies are recognizing the need to move from siloed, stand-alone engineering and product lifecycle-related software tools to a connected, integrated network of applications, each capable of maintaining their own authoritative sources of data while sharing subsets of it with other tools and users with a need to know. Historically, this has been achieved with either one-time exports, manual manipulation of the data, and import into another tool or via custom-built point-to-point adapters developed between specific tools. While these approaches can work up to a certain scale and level of complexity, they have drawbacks such as the possibility of introducing human error and slow transfer time in the former and costly development and maintenance of an ever-increasing number of point-to-point connectors in the latter.

In the connected Digital Engineering Ecosystem (DEE), model-based systems engineering (MBSE) is not the center of the universe, but instead, a critical cog in the integrated software tool chain supporting the full product development life cycle. The center of gravity within the DEE will shift depending on the phase of the product development lifecycle. Whereas MBSE is the “connective tissue” for requirements, behavior, physical design, and verification and validation, a Digital Thread tool, such as SBE Visions’ Digital Thread platform, is the “connective tapestry” for a connected Digital Engineering Ecosystem consisting of a rich mix of engineering, program management, manufacturing, and enterprise management software tools. Rather than relying on point-to-point connections, a Digital Thread tool provides a many-to-many central hub that handles the inter-tool data exchanges. A data-driven approach is required that determines the key information that needs to be shared between tools. It’s important to remember that data-driven does not mean “gather all your data and proliferate it everywhere”; rather, it should be question-based: what information is needed where and at what time to answer a specific question.

The diagram below shows an illustration of a connected Digital Engineering Ecosystem, comprised of tools, connections, and the people executing the work and making decisions. In this view, the Digital Thread platform acts as the central inter-tool data translator and broker, providing a many-to-many connection approach.



GENESYS' built-in SBE Vision Digital Thread platform adapter enables a GENESYS project to connect to an instance of SBE Vision's Digital Thread platform[1] and exchange digital engineering (DE) data with a broad ecosystem of other DE tools. SBE Vision uses a hub-and-spoke, ontology-first approach to data ingestion, mapping, translation, and transfer. This approach allows the ontological concepts in an entity-relationship-attribute (ERA)-based tool to be mapped to SBE's master ontology and published to a defined channel. The paired tool on the other side of the channel also maps to the SBE master ontology, and the data is then translated. A list of the software tools that SBE currently has integrations with is available at: https://www.sbe-vision.com/adapters-sdk.

Definitions

Authoritative Source of Truth (ASoT) – An entity such as a person, governing body, or system that applies expert judgement and rules to proclaim a digital artifact is valid and originates from a legitimate source. The authoritative source of truth for a digital artifact serves as the primary means of ensuring the credibility and coherence of the digital artifact that its creators share with a variety of stakeholders. It gives stakeholders from diverse organizations and distributed locations the authorization to access, analyze, and use valid digital artifacts from an authoritative source. The owners of digital environments or the community for digital engineering ecosystems provides stakeholders with an authoritative source of truth that assures confidence in the quality of the digital artifact across disciplines, domains, and life cycle phases.[2]

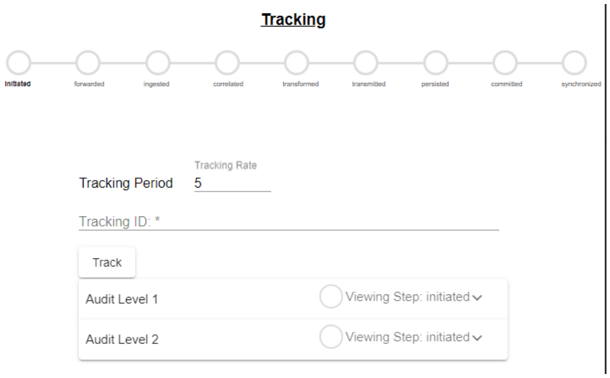
[1] SBE Vision's Digital Thread Platform requires separate licensing. For more information, contact them at: https://www.sbe-vision.com/contact

[2] From: https://www.omgwiki.org/MBSE/doku.php?id=mbse:authoritative_source_of_truth#:~:text=An%20authoritative%20source%20is%20known,Primary%20sources%20articulate%20the%20law.%20%E2%80%A6

Branch - A kind of Entity Set that holds a collection of Entities that can be revised because they are considered to be Work-in-Process (WIP).

Channel - A bi-directional data pipeline for the transmission of artifacts between an Entity Set and the smallest unit of versionable containment in a Data Source.

Commit ID - A tracking identifier for any entity sets published to the Digital Thread. See the Tracking UI for more context:



Tracking

Tracking Period

Tracking Rate: 5

Tracking ID: *

Audit Level 1 ○ Viewing Step: initiated

Audit Level 2 ○ Viewing Step: initiated

Digital Engineering Ecosystem – A digital engineering ecosystem includes enterprises' interconnected digital environments, stakeholder-networks, and semantic data that allows the exchange of digital artifacts from an authoritative source of truth to serve the stakeholder communities' interests.[3]

Digital Thread – A data-driven architecture that links together information generated from across the product lifecycle and is envisioned to be the primary or authoritative data and communication platform for a company's products at any instance of time."[4] It consists of a network of connections, both connections inside each domain model (intra-model) and the connections between models in different repositories (inter-model)[5] and the transfer of data and information across the network.

Entity Set - A managed collection of Digital Thread Entities within a Partition. Each Entity Set can have zero or more dependent Entity Sets which together define a Configuration Context.

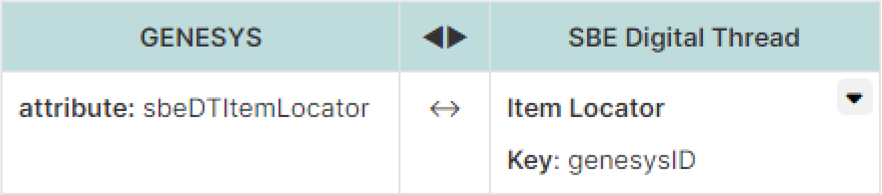
Item Locator - A global unique identifier (GUID) that GENEYS uses to correlate entities in GENESYS with their counterparts on the SBE Vision DT platform. Within the SBE Vision DT platform, an "Item Locator" is a key value-pair object associated with each entity. On the SBE DT Platform, GENESYS creates a new key called "genesysId"

[3] https://www.omgwiki.org/MBSE/doku.php?id=mbse:digital_engineering_ecosystem

[4] 2018 by Singh and Willcox at MIT in their paper entitled Engineering with a Digital Thread

[5] From: https://intercax.com/2020/08/21/syndeia-and-building-the-digital-thread/

within the Item Locator that stores the GENESYS-created unique ID string. The mapping between GENESYS and the SBE DT Platform is:



GENESYS	◀▶	SBE Digital Thread
attribute: sbeDTItemLocator	↔	Item Locator ▾ Key: genesysID

Non-isomorphic Transformation – In mathematics, non-isomorphic means "not having the same form" and is used in many branches of mathematics to identify mathematical objects which are structurally distinct.[6] Here, it means a data object in an Entity-Relationship-Attribute (ERA) ontological model of a specific classification in one software tool that is mapped and transformed to a different ERA classification in another tool. For example, an *attribute* of an entity in Tool A is mapped and transformed to a first-class *entity* in Tool B.

Ontology - Includes a representation, formal naming, and definition of the categories, properties, and relations between the concepts, data, and entities that substantiate one, many, or all domains of discourse. More simply, an ontology is a way of showing the properties of a subject area and how they are related, by defining a set of concepts and categories that represent the subject.[7]

Partition - An SBE container that owns a cohesive collection of Entity Sets. A partition can contain any number of channels.

Single Source of Trust (SSoT) - A single source of truth is a single point of access and modification for a data element. It requires both a normalized data model and integration such that producers and consumers of a data element all access the same copy. Single source of truth is an architectural practice and data strategy that helps to ensure that data is reliable, accurate and secured.[8]

Subscription - An operation to include a named subset of the Entities that have been cloned & transformed beneath another Branch or Tag for inclusion into the External System in its own native format.

[6] https://mathworld.wolfram.com/Nonisomorphic.html#:~:text=The%20term%20%22nonisomorphic%22%20means%20%22,are%20said%20to%20be%20isomorphic.

[7] https://en.wikipedia.org/wiki/Ontology_(information_science)

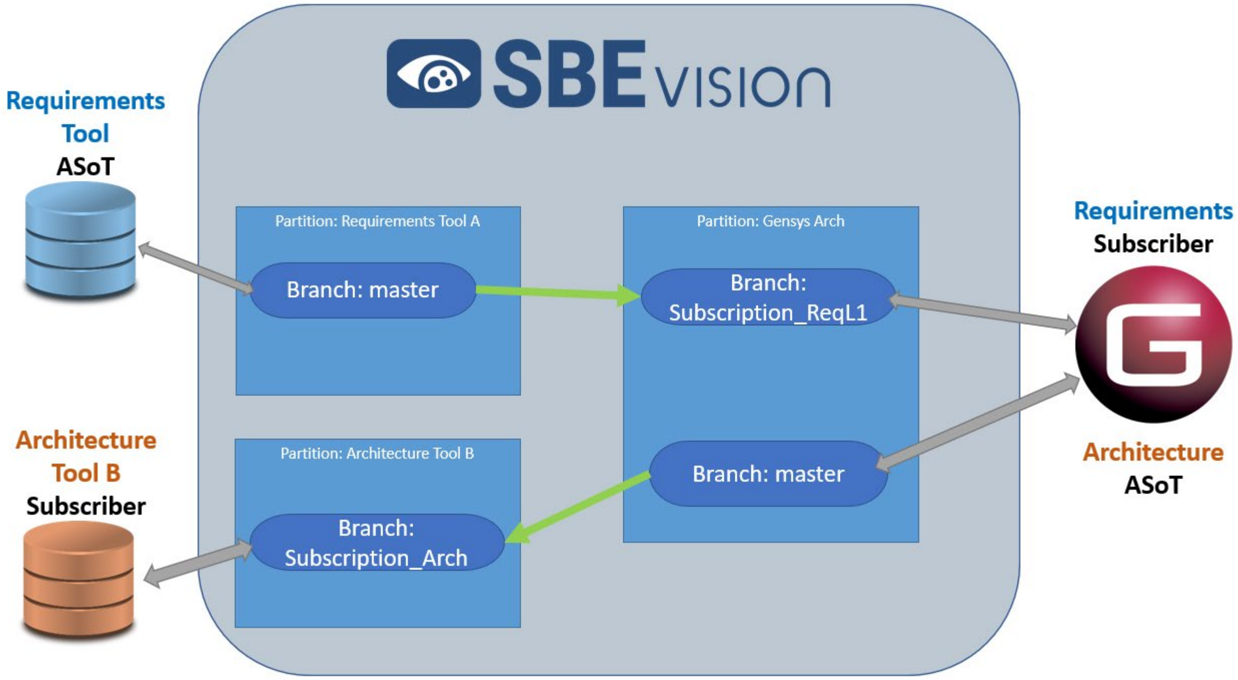
[8] https://www.omgwiki.org/MBSE/doku.php?id=mbse:authoritative_source_of_truth#:~:text=An%20authoritative%20source%20is%20known,Primary%20sources%20articulate%20the%20law.%20%E2%80%A6

CAPABILITY AND USAGE OVERVIEW

The transfer of a data package from one software tool to another using the SBE Vision Digital Thread platform requires a coordinated interplay between the two tools and the DT platform. The diagram below will be used to illustrate the export (publishing), import (subscribing), one-way data transfer, and round-tripping processes.

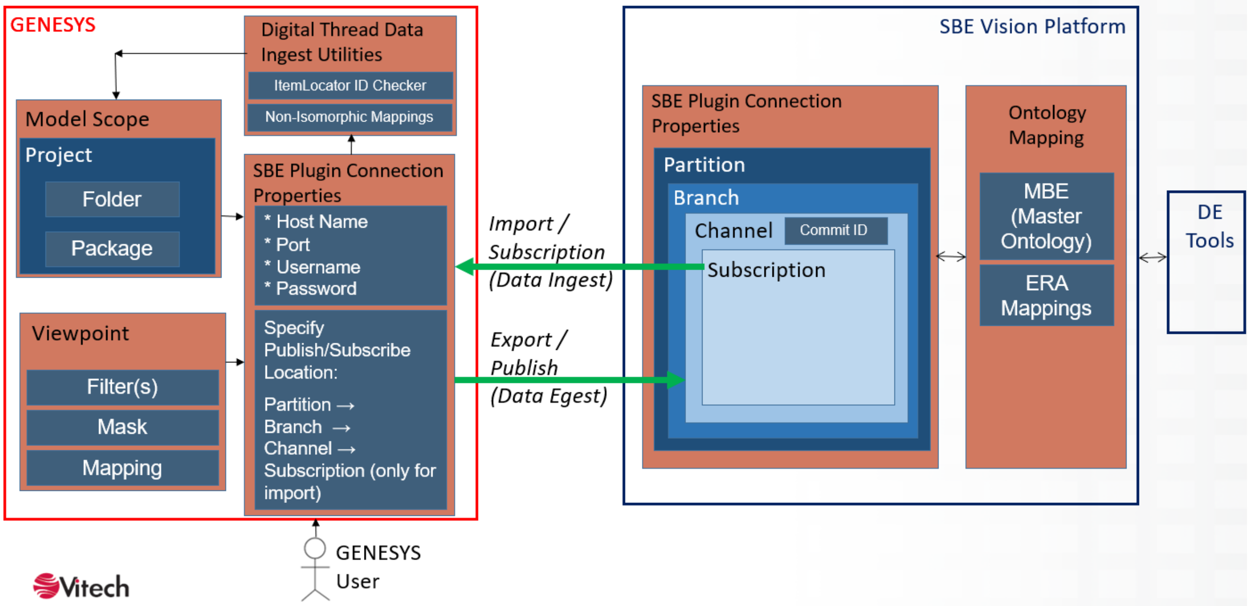
In this example, we have a Requirements Tool that acts as the ASoT for all requirements on the program. This is the tool where the requirements engineers perform their work. The systems engineers working in a GENESYS project on the systems architecture need an authoritative set of the requirements in their model to ensure that the functional and physical architectures that they are deriving are compliant with the program's requirements and to establish proper traceability. For this use case, the Requirements Tool acts as the ASoT for requirements and GENESYS acts as the subscriber for requirements.

Additionally, GENESYS acts as the overall ASoT for the systems architecture. A partner organization working on the program uses another MBSE tool to develop the architecture for one of the subsystems. They are using Architecture Tool B and need a subset of the overall systems architecture model captured in the GENESYS architecture ASoT project to provide them context from which to derive their subsystem's functional and physical architectures and to establish traceability to the higher-level architectural model. For this use case, GENESYS acts as the ASoT for architecture and Architecture Tool B acts as the subscriber for architecture.


SBE VISION
Requirements Tool ASoT
Partition: Requirements Tool A
Branch: master
Partition: Gensys Arch
Branch: Subscription_ReqL1
Requirements Subscriber
Architecture Tool B Subscriber
Partition: Architecture Tool B
Branch: Subscription_Arch
Branch: master
Architecture ASoT

In general, GENESYS always processes data exchanges to and from the DT through a Viewpoint (see the **VIEWPOINT** help topic for more information), which allows the data to be transformed into the local language and expectations of the end user. Additionally, any applicable non-isomorphic transformation rules in play are also processed. GENESYS utilizes a unique ID (stored in an attribute within GENESYS called “sbeDTItemLocator” and stored within the SBE Vision DT as the value associated with the key called “genesysID” that is housed within *ItemLocator* attached to each entity), to determine if data in the GENESYS project has previously been on the SBE Vision DT or not. This check is performed to ensure that entities are not incorrectly duplicated or replicated upon multiple import and export operations.

An overall nested physical architecture of the key interacting components of GENESYS and the SBE Vision DT Platform client are shown in the diagram below.



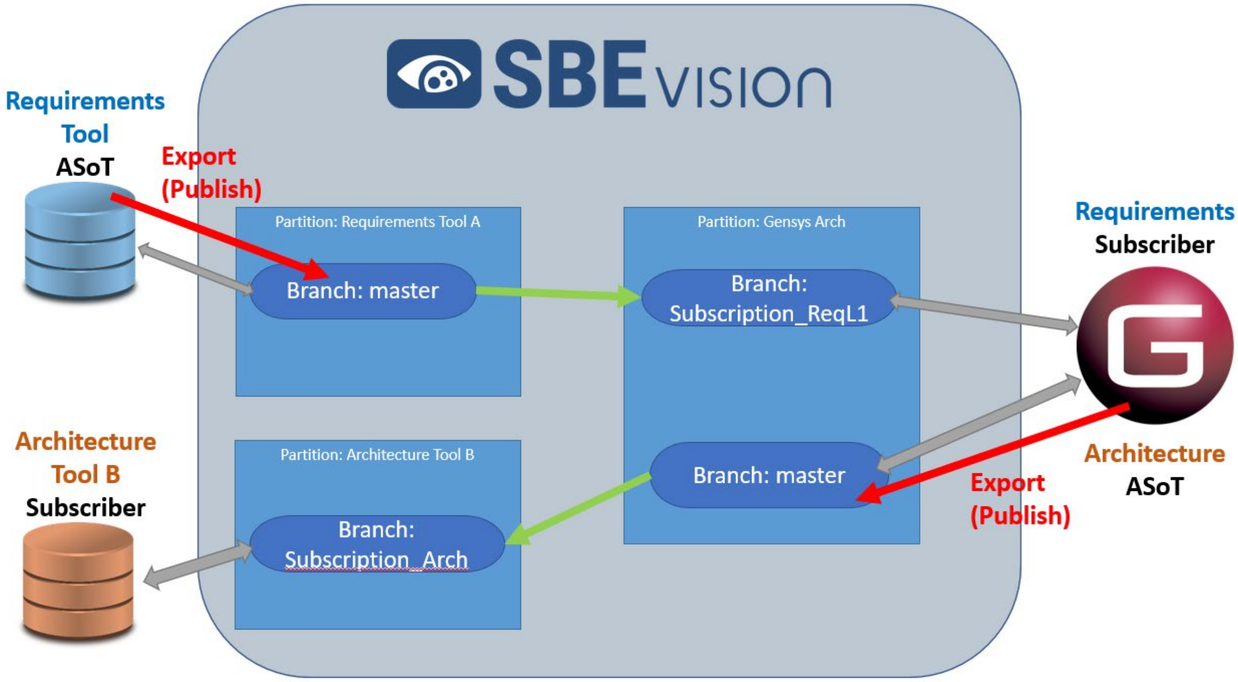
A GENESYS schema extension is provided that must be installed prior to using the SBE Vision DT adapter. For more information, see the section entitled “Installation of the SBE Vision Schema Extension”.

For the steps involving actions that must be taken within the SBE Vision DT platform client, please refer to SBE Vision’s help documentation, as the details may change over time.

The following sections describe the general process for exporting (publishing), importing (subscribing), one-way data transfers, and round-tripping. After those sections, the mechanics of how to execute each process in GENESYS is described.

Export (Publishing)

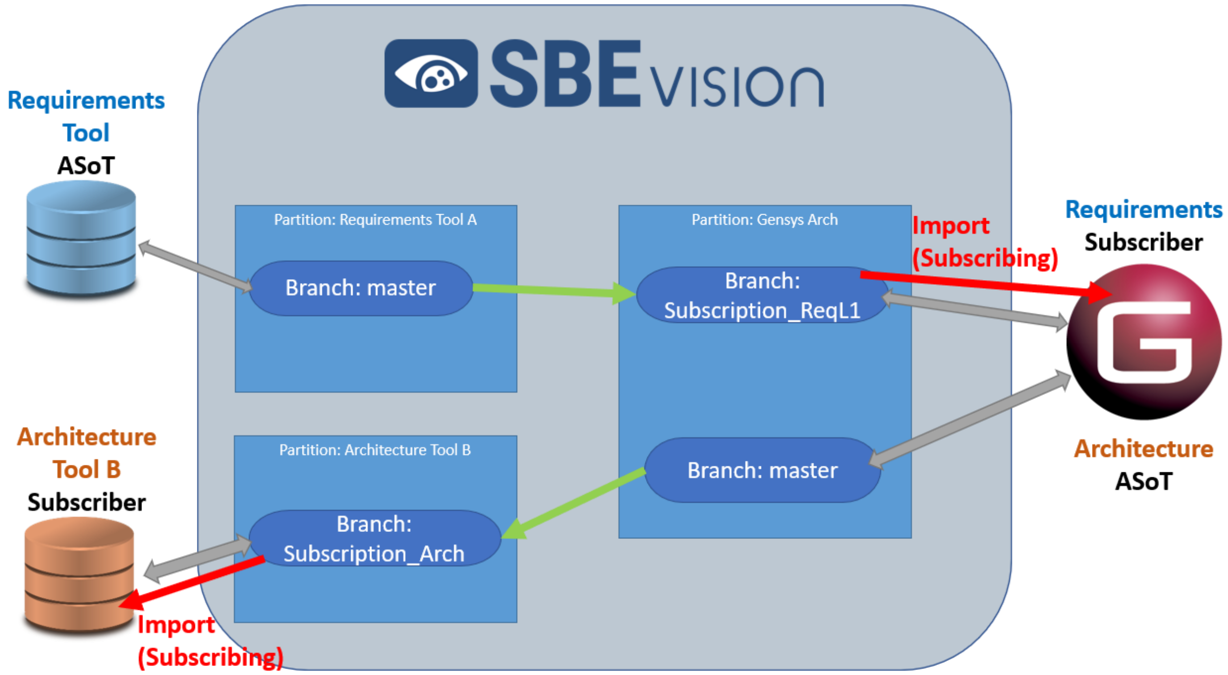
In our example, there are two tools that are exporting (publishing) data: the Requirements Tool and GENESYS, shown in the diagram below.



A partition is created on the SBE Vision DT platform for each associated tool. As a best practice, ASoT information should be published to a master branch. In this example, the Requirements Tool publishes its requirements to the “Branch: master” within the Requirements Tool A partition. The requirements entity set within the channel are now available to any other connected tool connected to the SBE Vision DT once ontologies are mapped between the two tools, which in this example, will be GENEYSYS. Similarly, the GENESYS project publishes its ASoT data related to functional and physical architecture to the “Branch: master” within the GENESYS Arch partition. The architecture entity set within the channel is now available to any other connected tool once ontologies are mapped, which in this example will be Architecture Tool B.

Import (Subscribing)

In our example, there are two tools that are importing data: GENESYS and Architecture Tool B, shown in the diagram below.



Imports are typically conducted from a branch labeled as a subscription. In this example, there is a "Branch: Subscription_ReqL1" in the GENESYS Arch partition. This subscription branch has been populated with all of the "level 1" requirements from the Requirements Tool. These requirements can then be imported into GENESYS via the import procedure described later. Similarly, there is a "Branch: Subscription_Arch" in the Architecture Tool B partition that is populated with the functional and physical architecture previously published out from GENESYS. The architecture entity set can then be imported into Architecture Tool be via the import procedure described later.

One-way Data Transfer and Round-Tripping

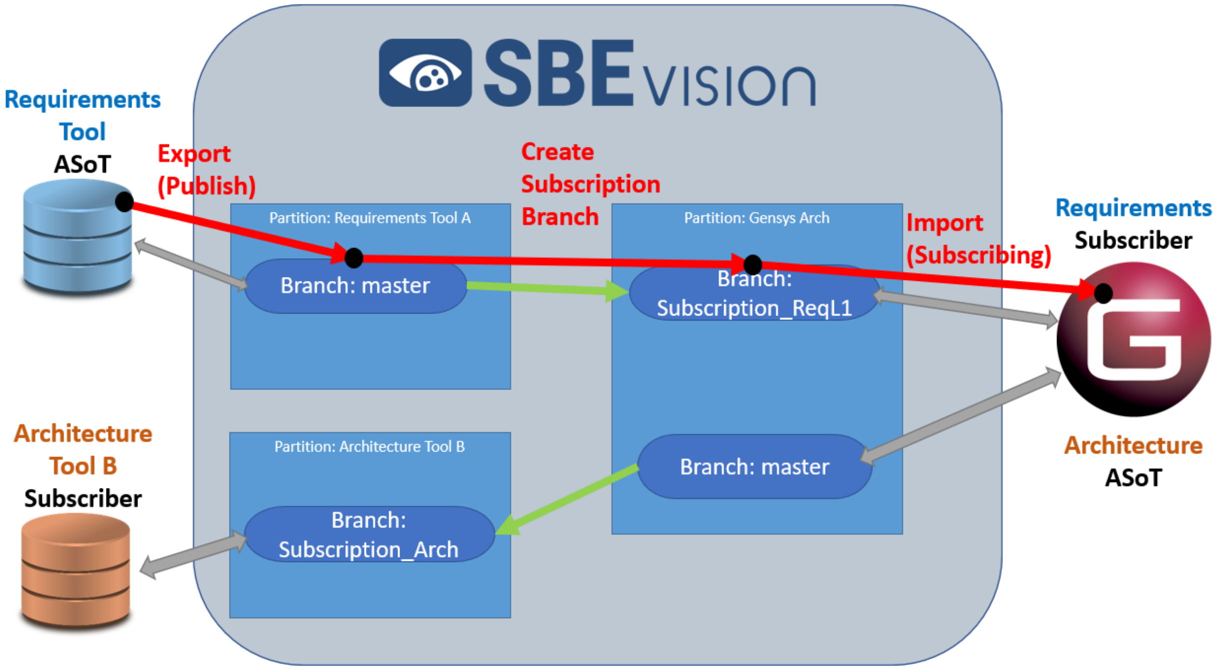
Combinations of exports, imports, branches, and diffs/merges across the SBE DT platform can be joined to transfer data between associated software tools. The two most common scenarios are discussed in the subsequent sections.

Within the SBE Digital Thread platform client, an additional step must be performed for all tool-to-SBE Vision DT-to-tool data transformations. The ontological concepts in each tool must be related to one another through the platform's ontology mapping interface. Please see SBE Vision's help resources for more information.

One-Way Data Transfer

In our example, the Requirements Tool is the ASoT for all requirements on the program. The systems engineers working in a GENESYS project on the systems architecture need an authoritative set of the requirements in their model to ensure that the functional and physical architectures that they are deriving are compliant with the program's requirements and to establish proper traceability. The requirements can be placed into

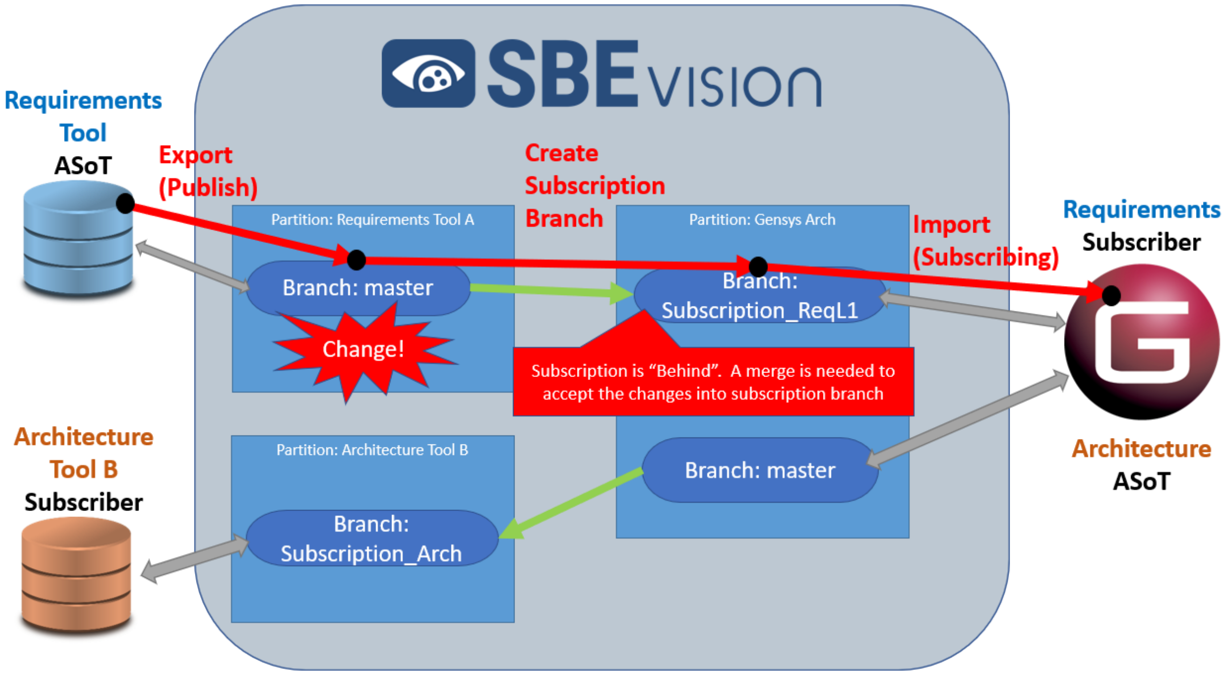
GENESYS using a combination of an export, merge, and import across the DT. The following diagram summarizes the operations to be performed.



The requirements are first exported from the Requirements Tool to the “Branch: master” in the Requirements Tool A partition on the SBE DT platform. Then, a merge operation is performed between the “Branch: Subscription_ReqL1” in the GENESYS Arch partition and the “Branch: master” in the Requirements Tool A partition. This then updates the “Subscription_ReqLq” branch to be consistent with the entity set in the “Branch: master.” Finally, an import can be performed from the GENESYS – SBE Vision DT adapter interface to ingest the requirements entity set contained within “Subscription_ReqLq” branch into a specified location within the GENESYS project.

Subsequent Update to the ASoT

After the requirements from the Requirements Tool have been ingested into GENESYS, it is highly probable that the requirements will go through another round of updates or refinements at some point in the product lifecycle. If the requirements are updated after the initial data transfer, a refresh operation needs to be performed. This is shown in the following figure.

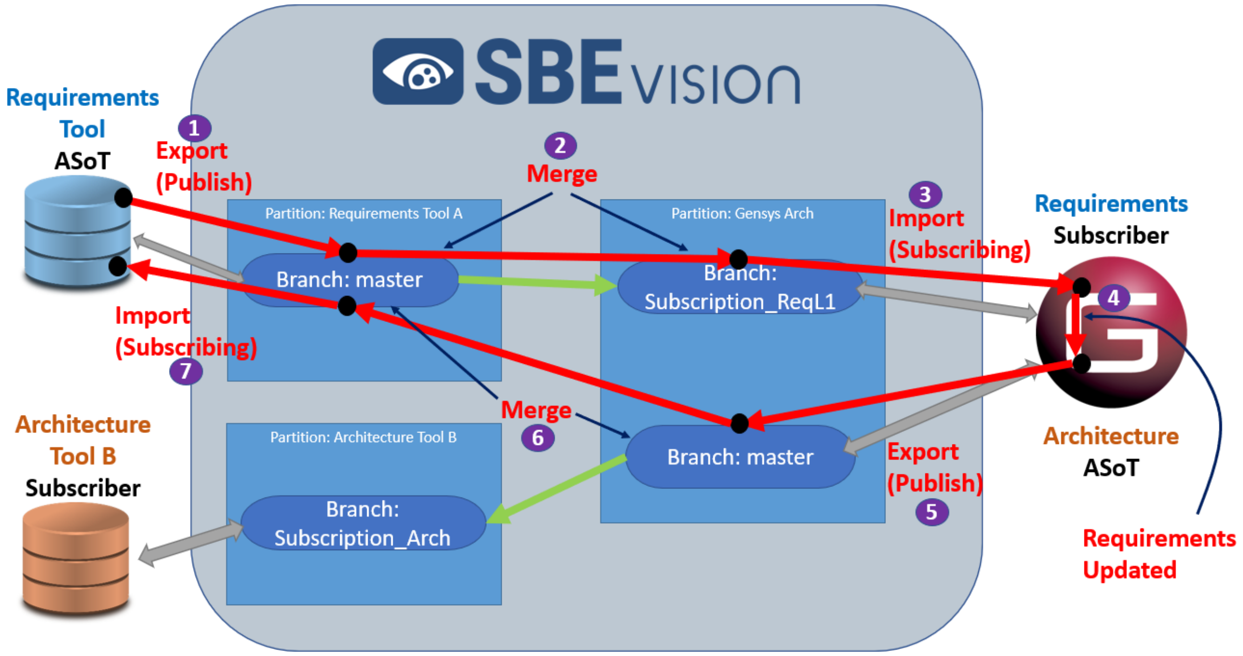

SBE VISION
Requirements Tool
ASoT
Export (Publish)
Create Subscription Branch
Partition: Requirements Tool A
Branch: master
Change!
Partition: Gensys Arch
Branch: Subscription_ReqL1
Subscription is "Behind". A merge is needed to accept the changes into subscription branch
Import (Subscribing)
Requirements Subscriber
Architecture ASoT
Architecture Tool B
Subscriber
Partition: Architecture Tool B
Branch: Subscription_Arch
Branch: master

The steps to ingest an update of the requirements into GENESYS are the same as the specified in the section above with the only difference being that the Subscription branch is no longer empty but populated with an out-of-date requirements data package. Step 2 to perform a merge will bring the subscription branch into alignment with the master branch.

Round-Tripping

Round-tripping is when data is exchanged between paired tools in a full cycle, with the primary direction being data flowing from the ASoT to the subscriber but also sometimes between the subscriber and the ASoT. It may be the case that the individual or team working in the subscriber tool finds something that needs to be corrected in the ASoT's data and may make a request to push an update from the subscriber back to the ASoT. When this occurs, the individual data elements make a full "round-trip" from the ASoT tool to the subscribing tool back to the ASoT tool.

In our example, requirements that have been ingested to GENESYS from the Requirements Tool have been found to potentially need an update in the ASoT due to an inconsistency found by the systems engineer.



The steps are the same as described above in the "One-Way Data Transfer" section up through the ingestion of the entity set into GENESYS. The systems engineers within GENESYS update some of the requirements and then publish the broader Architecture data set out to the "Branch: master" within the GENESYS Arch partition. A merge operation must be performed between the "Branch: master" within the Requirements Tool A partition and the "Branch: master" within the GENESYS Arch partition. If the requirement updates are accepted by the designated authority during the merge process, the requirements "Branch: master" is then brought into alignment with the GENESYS "Branch: master." Finally, an import can be performed to ingest the updated requirements entity set contained within "Branch: master" into a specified location within the Requirements Tool.

USING THE GENESYS SBE VISION DIGITAL THREAD ADAPTER

For the GENESYS 2023 release, the GENESYS-SBE Vision Digital Thread adapter has been robustly tested with DOORS classic and DOORS Next Generation. SBE Vision's other supported tools will work with GENESYS to a degree, although the fidelity of the data transformations and exchanges has not yet been benchmarked.

Installation of the SBE Vision Schema Extension

The provided SBE Vision schema extension must be imported into the GENESYS project before data can be transferred to/from the digital thread. This extension captures SBE Vision Digital Thread metadata associated with each GENESYS entity as attributes. The schema extension includes the following attributes:

Attribute Alias	Attribute Name	Class Applicability	Description
SBE Digital Thread URL	sbeDTURI	All Classes	A hyperlink to the object entity in the SBE Vision Digital Thread Platform Web client that corresponds to the same entity in GENESYS.
SBE Digital Thread ASoT URL	sbeDTASOTURL	All Classes	A hyperlink to the object in another software tool within the enterprise's SBE Vision-connected DE environment that has been flagged as the Authoritative Source of Truth (ASoT) for the data item.
SBE Digital Thread unique identifier	sbeDTItemLocator	All Classes	A global unique identifier (GUID) used to associate GENESYS elements with their counterparts on the SBE Digital Thread platform. These are assigned by GENESYS and populated in a key called "genesysID" within the Item Locator object associated with each entity in the SBE Vision DT platform.
SBE Digital Thread Subscription ID	sbeDTSubscriptionID	Package Class	Captures the associated SBE Vision Subscription ID to facilitate publishing data back to a specific subscription in round-tripping scenarios.

GENESYS users can find this extension in the GENESYS installation folders installed on their computer, typically at the following location:

> *Program Files > Vitech > GENESYS 2023 > Extensions > SBE Vision Digital Thread Schema Extension.gnsx*

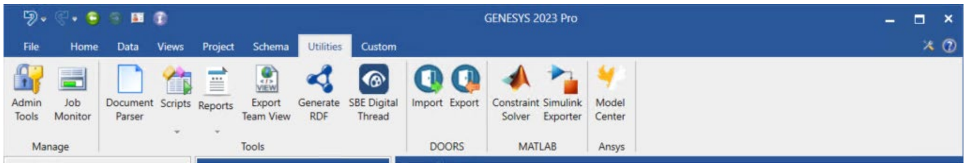
NOTE: When importing the schema extension, in Step 2 of the Import Wizard be sure to select the second option, ***Import into Project***, and select the project that you want to import the extension into, rather than creating a new project.

The import may take several minutes. Ignore the yellow WARNING message that is not applicable to this import.

Users and Groups, repository scripts (excluding project entity attribute scripts), and reports will not be imported.

ACCESSING THE GENESYS – SBE VISION DIGITAL THREAD ADAPTER

The ***SBE Vision Digital Thread*** adapter can be accessed via the ***Utilities*** tab on the ribbon and clicking on the applicable icon.



The adapter user interface is comprised of four input screens and one results screen. These screens are used to perform both *export* and *import* operations, which are described in the section of this document entitled "**Importing and Exporting**".

PROCEDURES FOR PERFORMING IMPORTS, EXPORTS, ONE-WAY TRANSFERS, AND ROUND-TRIPPING

The following sections describe the step-by-step procedures for performing common operations with the SBE Vision Digital Thread. Please refer to the UI screen references captured in [brackets] for more information on the user interface screen where the step is performed. Each of the UI Screens referenced is described in the section entitled "The GENESYS – SBE Vision DT Platform Adapter UI".

Export

The steps to export from GENESYS to the SBE DT Platform are:

1. **In the SBE Vision client**: Create a partition on the SBE DT platform (this step is required on the first pass only).
2. **In the SBE Vision client:** Create a master branch (this step is required on the first pass only).
3. **In GENESYS:** Using the SBE Vision adapter, select the Viewpoint that will be used to process the data package through upon export. [Input Screen 1]
   a. Optionally, one may include "context entities." If checked, all entities that are related in a direct way (not including grandparents or grandchildren entities related through another entity or another relationship) to the entities in the export data package's scope are included when exported to the digital thread. Note that GENESYS does not include cross-project relationships in the included scope of an export.

4. **In GENESYS**: Using the SBE Vision adapter, establish a connection to the SBE DT platform using the specified User Name and password as well as the *SBE Connection settings* established for your organization's specific installation instance of the SBE Vision client. [Input Screen 1]
5. **In GENESYS:** Using the SBE Vision adapter, select that an *Export* operation is to be performed. [Input Screen 2]
6. **In GENESYS:** Using the SBE Vision adapter, select the data placement location on the SBE platform, which will be the master branch created in Step 3. Publishing of ASoT information, as a best practice, should always be published to a master branch. [Input Screen 3]
7. **In GENESYS:** Using the SBE Vision adapter, select the data package scope (project, folder, or package) to be published. [Input Screen 4]
8. **In GENESYS:** Using the SBE Vision adapter, publish the data package to the specified SBE DT Channel. GENESYS will perform the following actions as part of the publishing process:
    a. Process the data package through the selected **Viewpoint's *masks***, ***mappings***, and ***filters***.
    b. Process the data through any non-isomorphic transformation rules assigned to the GENESYS project.
    c. For each entity or relationship that has been previously published to the DT from GENESYS, GENESYS finds the associated SBE Digital Thread object via its ***gensysId*** in the ItemLocator and updates all attributes of the class, or end point connections of the relationship, with the values stored in GENESYS.
    d. For any new entities that have never been previously published to the DT from GENESYS, GENESYS creates new objects in the SBE Vision DT Channel of the appropriate class or relationship and ShapeID based on the established mappings, assigns the GENESYS GUID to the GENESYS attribute "sbeDTItemLocator" in the object's ItemLocation, and writes all attributes from the GENESYS object to the SBE Vision DT object.
9. **In GENESYS:** GENESYS will report back to the user a summary of the data transfer operation's results, including a list of all conflicts generated, if any, during the import. [Results Screen 1]

Import

The steps to subscribe and ingest a data package from the SBE Vision DT platform into GENESYS are:

1. **In the SBE Vision client**: Create a partition on the SBE DT platform (this step is required on the first pass only).

2. **In the SBE Vision client:** Create a master branch (this step is required on the first pass only).
3. **In GENESYS:** Using the SBE Vision adapter, select the Viewpoint that will be used to process the data package through upon import. [Input Screen 1]
4. **In GENESYS**: Using the SBE Vision adapter, establish a connection to the SBE DT platform using the specified User Name and password as well as the *SBE Connection settings* established for your organization's specific installation instance of the SBE Vision client. [Input Screen 1]
5. **In GENESYS:** Using the SBE Vision adapter, select that an *Import* operation is to be performed. [Input Screen 2]
6. **In GENESYS:** Using the SBE Vision adapter, select the *Subscription* on the SBE Vision Platform that you wish to import. Note that for an import, in addition to specifying the *Channel*, you must also select a *Subscription*, which is part of a *Channel*. A *Subscription* consists of data from the *Channel* that has been cloned & transformed into the format that the ingesting tool is expecting. [Input Screen 3]
7. **In GENESYS:** Using the SBE Vision adapter, select the location where the data from the SBE Digital Thread subscription will be published in the GENESYS project. Note that for an import, the location should be a package with the same name as the SBE Digital Thread subscription name. [Input Screen 4]
8. **In GENESYS:** Using the SBE Vision adapter, import the data package to the specified GENESYS project location. GENESYS will perform the following actions as part of the import process:
    a. GENESYS will query the subscription and cache a list of all entities, attributes, and relationships in the set.
    b. For each entity in the SBE DT subscription, GENESYS will check to see if it has an existing "genesysId" in its ItemLocator or not. If the Entity does not have an existing "genesysId", that means that object has not previously been imported into the GENESYS workspace previously. In this case, GENESYS will assign it a new unique id.
    c. GENESYS will then perform a "sync" operation of the dataset back to the SBE Vision DT Channel, which will ensure that all entities in the set have their ItemLocator "genesysId" established on the SBE Vision platform. This is necessary for entity matching back to the DT in future transactions to ensure that elements are not incorrectly duplicated/replicated in future transactions.
    d. Process the data through any non-isomorphic transformation rules assigned to the GENESYS project.

e. Process the data package through the selected **Viewpoint's *masks*** and ***mappings***. Note that for an import, the Viewpoint's associated ***filters*** are not processed in reverse order, as they lose context in the reverse direction.

f. For Entities and relationships with a pre-existing "ItemLocator" genesysID prior to the sync operation (i.e. entities and relationships that already exist in the GENESYS project), GENESYS overwrites the existing object's attributes & parameters with the new DT values.

g. For Entities and Relationships that are brand new to the GENESYS project, GENEYSYS creates new GENESYS objects for each and populates their attributes, parameters and relationships with the information included in the SBE DT subscription.

9. **In GENESYS:** GENESYS will report back to the user a summary of the data transfer operation's results, including a list of all conflicts generated, if any, during the import. [Results Screen 1]

One-Way Data Transfer

The steps to perform a one-way data transfer are as follows:

1. Export a requirements data package from the Requirements Tool to the Branch: master on the SBE DE platform. [see Export section]
2. Create a Subscription branch in the GENESYS partition (or locate it if it already exists. Initially, the Subscription branch will be empty. On the SBE DE platform, perform a merge operation to bring the GENESYS Subscription branch into alignment with the Requirements Tool Master branch. [see SBE Vision's help for more information on the merge operation]
3. Import the requirements data package into GENESYS. [see Import section]

Round-tripping

The steps to perform a round trip are as follows:

1. Export a requirements data package from the Requirements Tool to the Branch: master on the SBE DE platform. [see Export section]
2. Create a Subscription branch in the GENESYS partition (or locate it if it already exists. On the SBE DE platform, perform a merge operation to bring the GENESYS Subscription branch into alignment with the Requirements Tool Master branch. [see SBE Vision's help for more information on the merge operation]
3. Import the requirements data package into GENESYS. [see Import section]

4. A systems engineer in GENESYS finds requirements that they believe need to be updated. The requirements are updated in GENESYS with rationale documented.
5. The updated data from GENESYS is published to its Master branch. [see Export section]
6. On the SBE DE platform, perform a diff/merge operation to determine which updates in GENESYS Master branch should be aligned with the Requirements Tool Master branch. Note that the diff/merge process allows discrete differences to be accepted or rejected. [see SBE Vision's help for more information on the diff/merge operation]
7. Perform an import/sync from the Requirements Tool Master branch back into the Requirements Tool. The Requirements Tool now has the accepted and approved updated requirements, which originated in GENESYS. [see Import section]

When round-tripping, it is best practice to ensure that the scope of the GENESYS model that is included is based on a ***Package***. This is the easiest and most reliable way to ensure that the scope remains the same with multiple round trips. It also allows for data within a single GENESY project to be compartmentalized for use with multiple SBE Vision channels and end-point tools.

THE GENESYS – SBE VISION DT PLATFORM ADAPTER UI

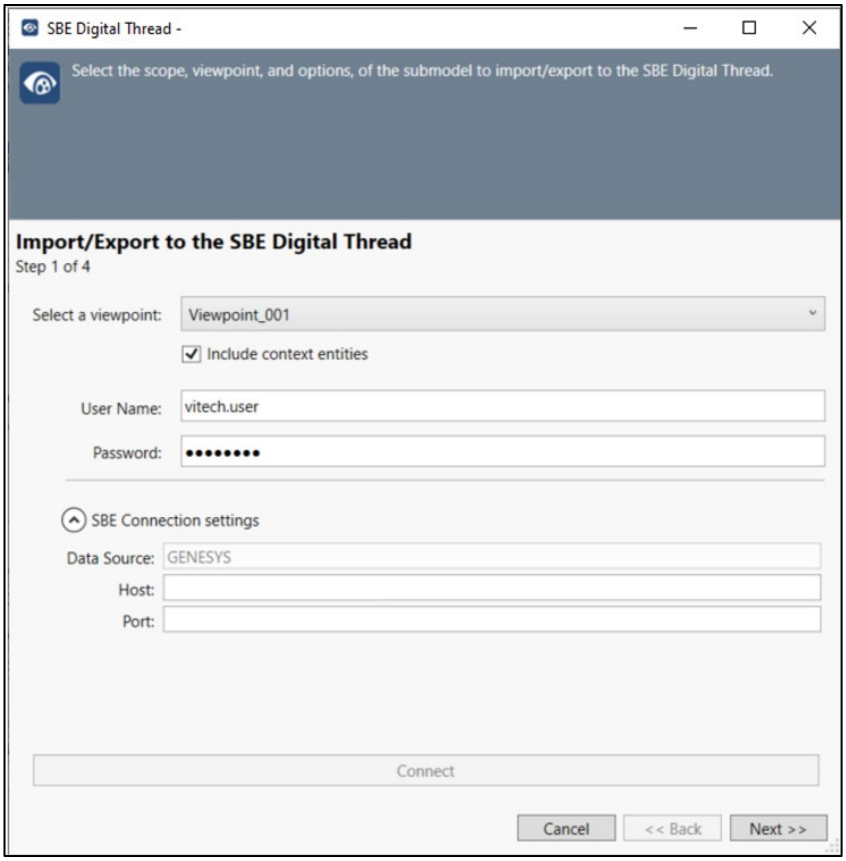
Input Screen 1 – Viewpoint, Credentials, SBE Connection Settings

The first screen allows the user to select the Viewpoint that will be applied to the data transfer operation. Additionally, the user can optionally select whether to include context entities.

The user's credentials to connect to log into the SBE Vision DT Platform instance are also entered here. The ***User Name*** and ***Password*** are not persisted and will need to be entered for any interaction with the digital thread.

The last section of this screen is dedicated to specifying the ***connection settings*** to the instance of the SBE Vision platform that GENEYS is being connected.

1. The Data Source field is filled in automatically. For advanced workflows, this can be changed through the GENESYS configuration file. Please contact support for details on changing the SBE data source.
2. Enter the host name assigned to your company in the ***Host*** field. This data is persisted and will need to be updated only when importing or exporting data to a different SBE portal than the last digital thread interaction with GENESYS.
3. Enter the port in the ***Port*** field only if you are working on a local machine.



SBE Digital Thread -

Select the scope, viewpoint, and options, of the submodel to import/export to the SBE Digital Thread.

Import/Export to the SBE Digital Thread

Step 1 of 4

Select a viewpoint: Viewpoint_001

☑ Include context entities

User Name: vitech.user

Password: ••••••••

SBE Connection settings

Data Source: GENESYS

Host:

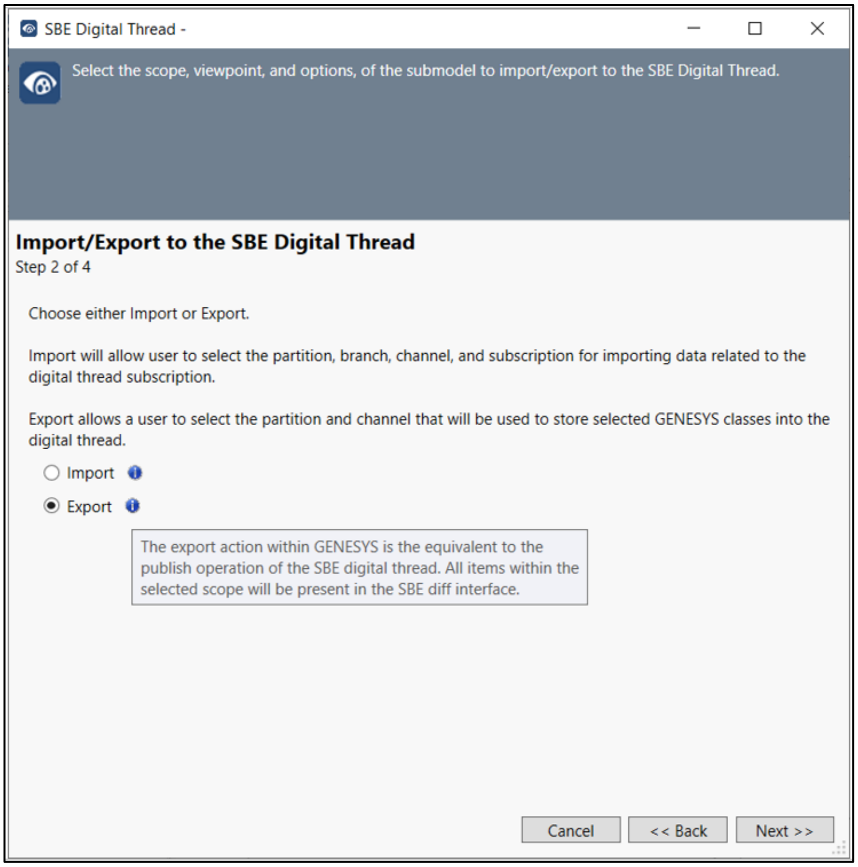
Port:

Input Screen 2 – Specifying Import or Export

The second screen allows the user to specify whether an ***Import*** or ***Export*** operation will be performed.

SBE Vision uses different terms for these operations. The equivalent terms are shown in the table below.

GENESYS Term	SBE Vision Term	Description
Import	Synchronize	This operation first does a refresh and then a publish in the same action.
Export	Publish	Publishes all the data within the specified scope to the selected SBE Vision Channel.



Select the scope, viewpoint, and options, of the submodel to import/export to the SBE Digital Thread.

Import/Export to the SBE Digital Thread

Step 2 of 4

Choose either Import or Export.

Import will allow user to select the partition, branch, channel, and subscription for importing data related to the digital thread subscription.

Export allows a user to select the partition and channel that will be used to store selected GENESYS classes into the digital thread.

○ Import

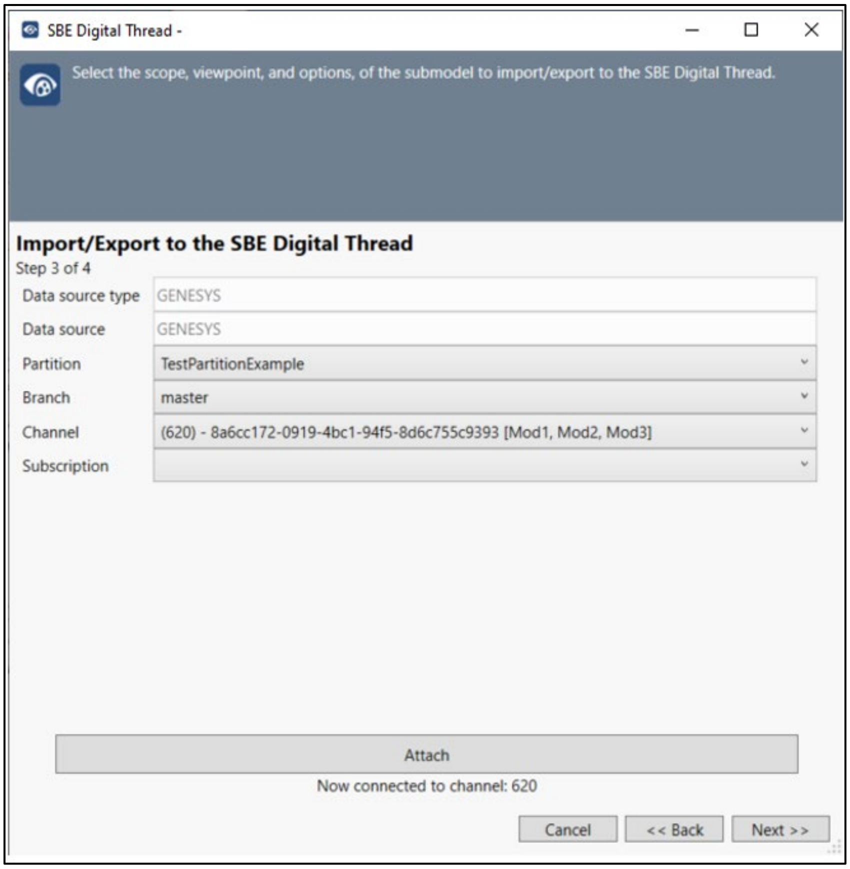
● Export

The export action within GENESYS is the equivalent to the publish operation of the SBE digital thread. All items within the selected scope will be present in the SBE diff interface.

Input Screen 3 – Specification of the Data Location on the SBE DT

For an *Expor*t operation, the location where the GENESYS data package will be placed on the SBE Digital Thread platform is specified from this screen. For an *Import* operation, the location of the data on the SBE Digital Thread that will be ingested into GENESYS is specified from this screen.

1. At the window that appears, the ***Data source type*** and ***Data source*** should automatically appear in the window. Advanced users can reach out to Vitech support for details on how to update these values if a different name was chosen when setting up the SBE platform.
2. Select the specific partition, branch and channel that you wish to connect to from the drop-down lists on the ***Partition, Branch***, and ***Channel*** fields, respectively.
3. ***Subscription*** must be specified for *Import* operations only.



Select the scope, viewpoint, and options, of the submodel to import/export to the SBE Digital Thread.

Import/Export to the SBE Digital Thread

Step 3 of 4

Data source type: GENESYS

Data source: GENESYS

Partition: TestPartitionExample

Branch: master

Channel: (620) - 8a6cc172-0919-4bc1-94f5-8d6c755c9393 [Mod1, Mod2, Mod3]

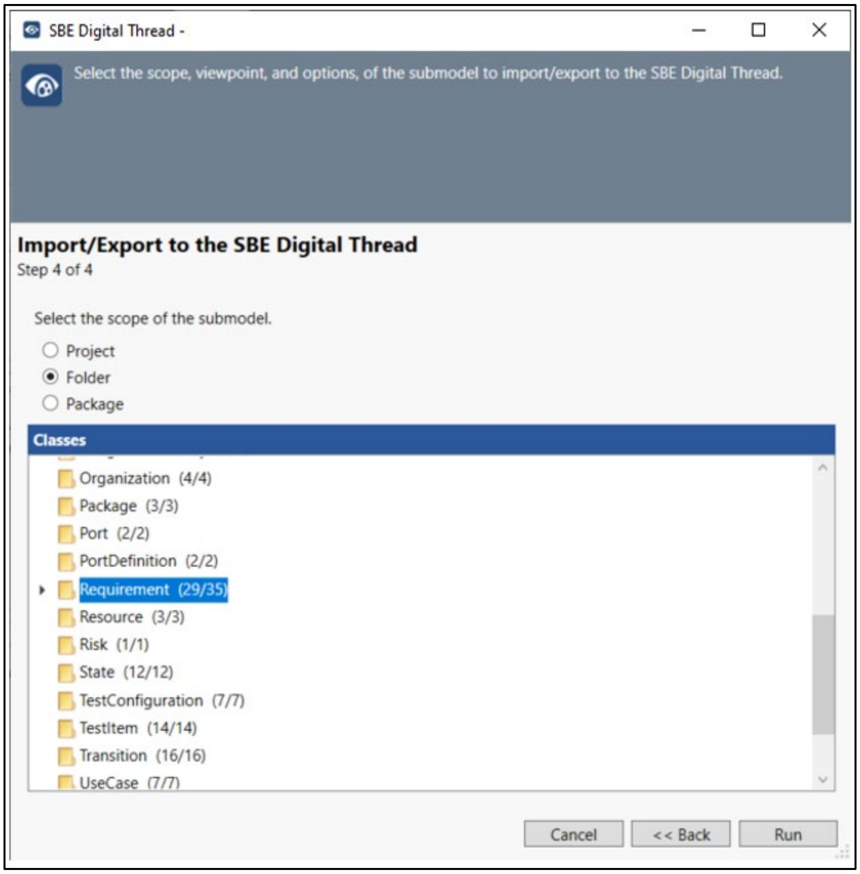
Subscription:

Now connected to channel: 620

Input Screen 4 – Scope and Location

The last input screen allows the user to specify the scope and location of the data transfer. For an *Export* operation, the user is selecting the scope of the entities that are included in the data package that will be exported to the specified location on the SBE Digital Thread platform. For an *Import* operation**,** the user is selecting the location where the imported data package will be placed within the GENESYS project file.

The options are **Project, Folder** *(including all sub-folders)*, or **Package** *(including all sub-packages)*. For *Import* operations, the selected scope should be consistent with the entity types contained in the data package that is being imported. Note that this is a concern if **Folder** is selected, as **Folders** contain only one class; if the data package contains conflicting classes, the import operation will fail to import the non-supported classes and will be noted in the *Issues* section of the Results UI screen.



SBE Digital Thread -

Select the scope, viewpoint, and options, of the submodel to import/export to the SBE Digital Thread.

Import/Export to the SBE Digital Thread

Step 4 of 4

Select the scope of the submodel.

○ Project

● Folder

○ Package

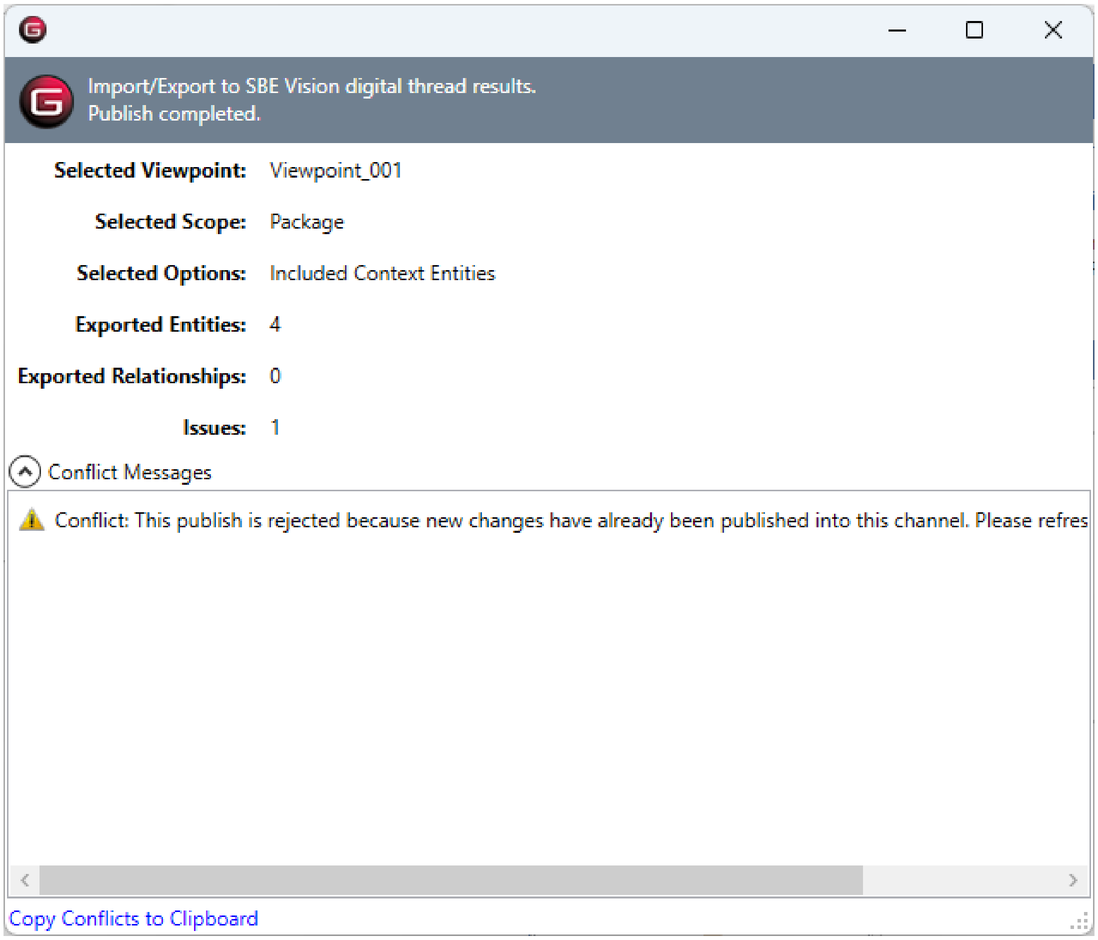
Classes

- Organization (4/4)
- Package (3/3)
- Port (2/2)
- PortDefinition (2/2)
- Requirement (29/35)
- Resource (3/3)
- Risk (1/1)
- State (12/12)
- TestConfiguration (7/7)
- TestItem (14/14)
- Transition (16/16)
- UseCase (7/7)

Results Screen 1 – Operation Summary and Issues

The final operation summary and issues screen is presented after the *Import* or *Export* operation has been completed.

The screen summarizes the options selected for the performed operation, provides summary statistics on the number of ***Entities*** and ***Relationships*** that were exported or imported, and summarizes any issues that were encountered. Any *Issues* that were encountered will have additional details displayed in the *Conflict Messages* pane. These can be copied to the clipboard via the option displayed at the bottom of the window for archiving or further investigation.



Import/Export to SBE Vision digital thread results.
Publish completed.

Selected Viewpoint: Viewpoint_001

Selected Scope: Package

Selected Options: Included Context Entities

Exported Entities: 4

Exported Relationships: 0

Issues: 1

Conflict Messages

⚠ Conflict: This publish is rejected because new changes have already been published into this channel. Please refres

LIST OF NON-ISOMORPHIC TRANSFORMATION RULES

As of the release of GENESYS 2023, the following non-isomorphic transformation rules are implemented:

GENESYS Entity's Parameters to SBE DT Parameter class entities

Parameters of any entity in GENESYS are transformed to first class **Parameter** entities on the SBE Vision platform. This transformation is performed to facilitate interaction with other MBSE tools which treat values and units (parameters in GENESYS's CSDL) as first-class entities.

Egress / Publish

Parameters are published to the SBE Vision DT as **Parameter** entities that are related to the host entity through a "*contained in*" relationship. Each parameter entity on the DT shall has its ***Shape Name*** assigned the same name as the ***Parameter Name*** in GENESYS.

On the SBE Vision DT, there are 3 sub-classes of the Parameter entity with the following GENESYS parameter fields mapped to it:

SBE Vision::Parameter Sub-class	GENESYS::Parameter Field
Constraint Parameter	Minimum Maximum Objective units
Design Parameter	Design units
Requirement Parameter	Observed units

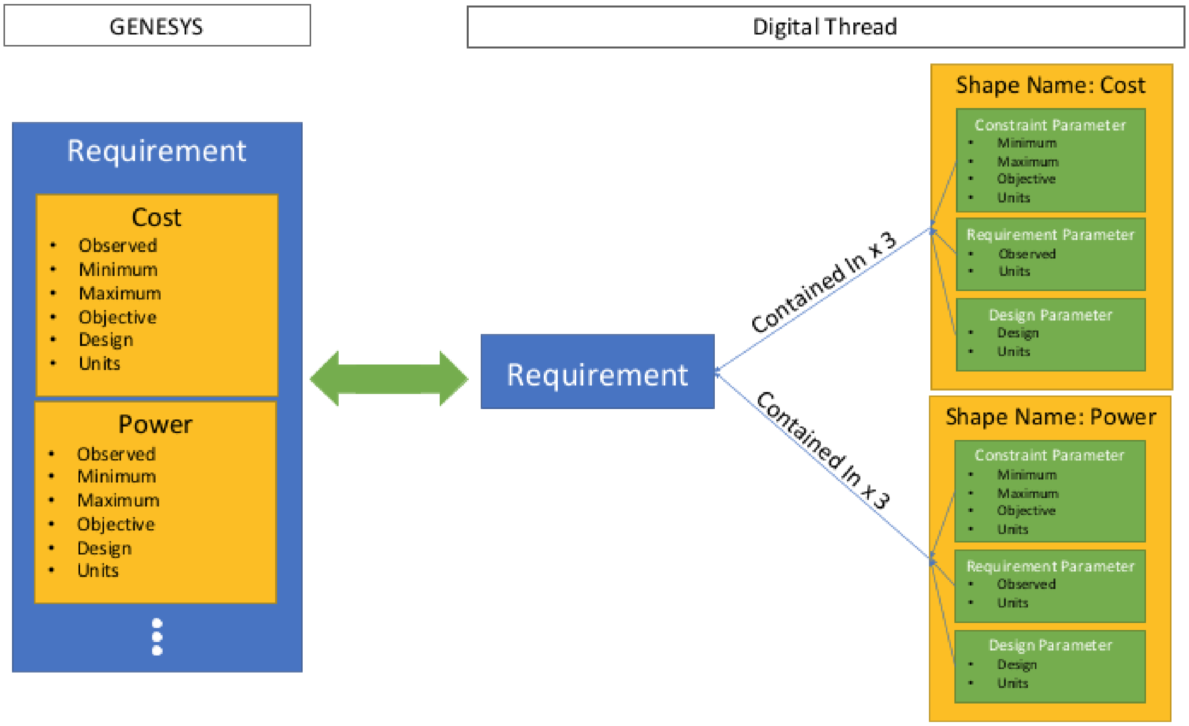
Units are mapped to each of the subclasses. The reason for this is that parameter entities may be synchronized to different Digital Engineering tools based on need and may not be shipped as a complete set. As such, each parameter entity needs to hold its units.

The SBE Vision sub-class names as well as the mappings of GENESYS parameter fields to them are modifiable by the SBE Vision user.

Ingest

GENESYS looks for all entities in the SBE Digital Thread subscription collection of type "**Parameter**" that have a relationship of “parameterrelation" to another entity and with a shape name of “"genesysrequirementparameter," "genesysdesignparameter," or "genesysconstraintparameter". Any elements meeting this criterion have their ***minimum, maximum, objective, design, observed***, and ***unit*** parameter fields mapped back into the associated GENESYS element's parameter fields on impot to GENESYS.

An illustration of this non-isomorphic transformation is shown in the following figure.



SBE Visions DOORS Classic & DOORS NextGen Integrations

DOORS *Requirement* elements that have *Stakeholder* attributes of attribute type “multi-select” are transformed to GENESYS *Organization* class entities of the same name. In GENESYS, a *“has stakeholder/is stakeholder of”* relationship is created between the Organization entity(ies) and the related Requirement entity. This is included by request of the launch customer for the GENESYS-SBE Vision adapter.

GENERAL NOTES ON DATA PROCESSING RULES

This section describes general notes on data processing rules applied to imports and exports.

Attribute Type Mappings

GENESYS has many specific attribute types, some of which do not have a direct corollary on the SBE Vision DT platform. To account for this, the following mappings are made:

GENESYS Attribute Type	SBE Vision DT Platform Type
Boolean	Boolean
Collection	Array
Date	String
Date Time	String

Entity Reference	String
Enumeration	String
Float	Float
Hierarchical Number	String
Integer	Integer
Number Spec - Constant	Float
Number Spec - Random	--NOT SUPPORTED-- / --NOT EXPORTED --
Reference Spec	String
Script Spec	--NOT SUPPORTED-- / --NOT EXPORTED --
String	String
Text	String

Relationship Attributes

GENESYS supports the concept of *relationship attributes*. However, the SBE Vision DT currently does not. As such, *relationship attributes* are not currently exported to the SBE Vision DT platform.

Embedded Images and Tables

The SBE Vision DT Platform currently does not support the transport of embedded images or tables in entity attributes. As such, all embedded images and tables are neither exported from GENESYS nor available from other tools to be imported into GENESYS.

Inherited Parameters and Attributes

GENESYS supports the capability of entities to inherit parameters and attributes from parent entities through the *specialization* relationship. However, data imported from the SBE Vision may have parent-child entities that are not consistent with this inheritance. If this occurs, GENESYS will import the data as defined in the SBE Vision DT subscription and override the attribute and parameter values, as needed.

Parameter Substitutions

Parameter Substitutions are a useful way in GENESYS to avoid having to hard code numerical values and units in a textual attribute field. Rather, they allow the user to create a coded reference to the values, which are replaced any time the entity attribute is exported from GENESYS, ensuring that the textual description is always consistent with the parameters whenever they change. However, this coded reference creates a challenge when exporting and importing data from the Digital Thread where other tools do not have a corresponding concept.

The way that GENESY handles this is as follows:

- On *Export*, parameter substitutions are published with the coded references replaced with their associated parameter's values.
- On *Import*, any attributes of type "text" are imported as-is. GENESYS does not attempt to try to resolve the text strings with any associated parameters. The user must review and update these fields manually, if needed.

Parameter Bindings

Parameter Bindings provide a mechanism to ensure that parameters remain consistent between associated entities. However, this concept creates a challenge when exporting and importing from the Digital Thread where other tools do not have a corresponding concept.

The way that GENESY handles this is as follows:

- On *Export*, bound parameters are published as "unbound" copies of the parameters in each bounded entity (i.e. the child entity). This essentially is a dumb export where the concept of binding is removed and each parameter is exported independently.
- On *Import*, if there are any conflicts that would violate existing parameter bindings in GENESYS, the binding in GENESYS is removed and the values from the DT are published to the appropriate fields.

THIS PAGE INTENTIONALLY BLANK

Vitech
A Zuken Company

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