

IBM VisualAge TeamConnection



# ENVY-Manager TeamConnection Bridge User's Guide

*Version 2.0*



IBM VisualAge TeamConnection



# ENVY-Manager TeamConnection Bridge User's Guide

*Version 2.0*

**Second Edition (December 1997)**

**© Copyright International Business Machines Corporation 1997. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

## **Notices**

**Note**

Before using this document, read the general information under “Notices” on page vii .



---

# Contents

<b>Notices</b> . . . . .	iii
<b>Notices</b> . . . . .	vii
<b>Trademarks</b> . . . . .	ix
<b>Chapter 1. Overview of the ENVY/Manager-TeamConnection Bridge</b> . . .	1
Scope of this documentation . . . . .	1
Description of the ENVY/Manager-TeamConnection Bridge . . . . .	2
Basic functionality . . . . .	2
How the bridge communicates with TeamConnection . . . . .	2
<b>Chapter 2. Preparing to use the ENVY/Manager-TeamConnection Bridge</b> .	5
Setting up the bridge environment . . . . .	5
Prerequisites . . . . .	5
Environment variables . . . . .	5
Installing and activating the ENVY/Manager-TeamConnection Bridge. . . . .	6
Loading the ENVY/Manager-TeamConnection Bridge . . . . .	6
Testing the ENVY/Manager-TeamConnection Bridge . . . . .	7
<b>Chapter 3. Using the ENVY/Manager-TeamConnection Bridge</b> . . . . .	9
Setting default properties . . . . .	9
Context page . . . . .	10
Operations page . . . . .	10
Import page. . . . .	11
Export page . . . . .	12
Exporting ENVY components to TeamConnection . . . . .	12
Exporting configuration maps and applications . . . . .	13
Exporting files . . . . .	14
Importing ENVY components from TeamConnection . . . . .	14
<b>Chapter 4. Using the ENVY/Manager-TeamConnection Bridge: a simple   scenario for VisualAge Generator developers</b> . . . . .	17
Scenario assumptions . . . . .	17
Exporting ENVY components to TeamConnection . . . . .	17
Object mapping in TeamConnection . . . . .	18
Build generation . . . . .	19
Making a change to a member. . . . .	20





---

## Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY, USA 10594.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact the Site Counsel, IBM Corporation, P.O. Box 12195, 3039 Cornwallis Road, Research Triangle Park, NC 27709-2195, USA. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

IBM may change this publication, the product described herein, or both. These changes will be incorporated in new editions of the publication.

This publication contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.



---

## Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

AIX	MVS/XA
Common User Access	NetView
CUA	Operating System/2
C/370	OS/2
ENVY*	TeamConnection
IBM	VisualAge
MVS	XGA
MVS/ESA	

\* ENVY is a registered trademark of Object Technology International, Inc.

The following terms are trademarks of other companies:

**ObjectStore**

ObjectStore Design, Inc.

**UNIX** X/Open Company Limited

Microsoft, Windows, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.



---

## Chapter 1. Overview of the ENVY/Manager-TeamConnection Bridge

ENVY provides a repository with operational support tailored specifically for highly-interactive, prototyping environments that emphasize iterative development, such as VisualAge for Smalltalk or VisualAge Generator. A bridge from ENVY to TeamConnection provides access to the powerful software configuration management (SCM) support provided by ENVY, along with the scalable, enterprise-level support provided by TeamConnection. TeamConnection's ability to manage all development artifacts (not just source code), to share information in a common model, and to integrate multiple tools and multiple languages across the enterprise on a single baseline extends the capabilities of software development groups. The ENVY/Manager-TeamConnection Bridge (also referred to as ***the bridge*** in this documentation) will provide essential integration for VisualAge tools which use ENVY as their day-to-day operational library.

VisualAge Generator Version 3.0 has access to the TeamConnection-ENVY Bridge through VisualAge for Smalltalk, which can interface directly with ENVY/Manager. The bridge supports the import and export of VisualAge Generator objects (*parts*) to and from TeamConnection.

ENVY/Manager provides a collaborative component development environment for application development and integration using fine-grained object languages, such as Smalltalk. The ENVY repository is designed for languages that run on the universal virtual machine (uVM). The repository includes persistence, versioning, and configuration management.

TeamConnection can be used to manage artifacts (*parts*) that need to be shared with non-uVM based languages or tools for purposes of build management, problem tracking, and other configuration management functions. These artifacts can be exported to the TeamConnection server through the ENVY/Manager-TeamConnection Bridge and stored as TeamConnection parts.

ENVY objects stored in a TeamConnection database can be queried and retrieved back into the ENVY/Manager development environment as needed. The units of storage in TeamConnection include exported ENVY components (such as applications and configuration maps) and large grained objects (files). Small-grained objects, such as VisualAge Generator data items, are imported and exported as constituents of applications. The data items in an application are exported to TeamConnection in an array that makes their definitions available to other tools through the data model.

---

### Scope of this documentation

This documentation is intended for users and administrators installing and using the bridge. It is assumed that you familiar with both the VisualAge for Smalltalk and TeamConnection products.

The following subsections describe the mechanics of enabling the ENVY/Manager-TeamConnection Bridge for VisualAge for Smalltalk Pro (Version 4.0 or later), the process of exporting ENVY components to TeamConnection, and the process of importing these components back into ENVY/Manager. See the

VisualAge Generator documentation for tool-specific details. TeamConnection information related to change tracking and build processing are addressed in the TeamConnection documentation.

Many terms used by VisualAge for Smalltalk and TeamConnection are problematic because the tools may define these terms differently. *Release* and *component* are typical examples. To avoid any ambiguity, such terms may precede the name of the tool they are applied to, such as *TeamConnection release*.

---

## Description of the ENVY/Manager-TeamConnection Bridge

### Basic functionality

It makes sense to describe the functionality of the bridge from the perspective of a Smalltalk developer, because it is through the Smalltalk image that the user drives the bridge. The bridge is an import/export facility for three types of entities:

- Smalltalk configuration maps
- Smalltalk applications
- Files residing on local and networked file systems that are accessible through the image

The bridge allows a Smalltalk developer to store any of these entities in a TeamConnection database and retrieve them at a later time. As is the case when exporting to other ENVY/Manager libraries, configuration maps and applications must be versioned before they can be exported. This enforces the notion that the Smalltalk developer uses the bridge and TeamConnection to maintain baselines rather than for managing work-in-progress.

Developers use ENVY/Manager's fine-grained support to facilitate the process of shared development in open editions of components on a daily basis. At appropriate junctures, components are versioned and promoted to TeamConnection, where together with other project elements, they form a baseline across an entire project. The resulting baseline may contain objects such as program elements, file, and metadata.

From the perspective of the TeamConnection user or administrator, the bridge allows the Smalltalk image to function as a TeamConnection client, storing and retrieving parts in a TeamConnection family database.

### How the bridge communicates with TeamConnection

The bridge functions are initiated from within the VisualAge for Smalltalk environment. Each operation that interacts with TeamConnection runs for some time in the Smalltalk image, but at some stage will make use of functions built into an appropriate version of the TeamConnection client and server. The bridge itself is implemented in Smalltalk, with the primitive functions provided in one of the DLLs in TeamConnection.

The unit of transfer used by the bridge for Smalltalk components is an ENVY/Manager library. Each library stored in TeamConnection contains one of the following:

- a Smalltalk application and its released subapplications (and their released subapplications, and so forth)

- a configuration map without any of its released applications

**Note:** Subapplications cannot be exported through the bridge without an enclosing application.

ENVY/Manager libraries are stored in TeamConnection databases as TeamConnection parts. When the bridge exchanges a library with TeamConnection, the target in a TeamConnection database is specified by a TeamConnection context. A TeamConnection context is comprised of the following TeamConnection parameters:

- Family name
- Release name
- Work area name

**Note:** Each TeamConnection context can contain only one version of any named application or configuration map. This is unlike ENVY/Manager libraries, in which multiple versions of a named Smalltalk component can co-exist.

The bridge is aware of the various relationships between ENVY/Manager components. When an application is transferred through the bridge, all of its released subapplications are transferred with it. When a configuration map is transferred through the bridge, the bridge will also transfer the released applications in separate operations. Depending on a user-specified setting, the bridge can also transfer required maps of configuration maps.





---

## Chapter 2. Preparing to use the ENVY/Manager-TeamConnection Bridge

The bridge is delivered as a configuration map suitable for loading into a VisualAge for Smalltalk Version 4.0 (or later) image. The library, TCEMBR.DAT, will contain the configuration map **ENVY/Manager-TeamConnection Bridge** and, for VisualAge Generator build support, **VAGen ENVY/TC Bridge**.

These configuration maps should be imported into your development library so that it can be loaded by all of the users sharing that library. The step-by step instructions are described in "Installing and activating the ENVY/Manager-TeamConnection Bridge" on page 6.

Usually, the **Library Supervisor** or the first user to use the bridge will perform this operation and then inform other users that the tool is available in the library.

The sections that follow describe the steps necessary to set up the bridge and verify that it is functional.

---

### Setting up the bridge environment

The following information is especially pertinent to the individual(s) responsible for bridge setup and administration.

#### Prerequisites

Before the bridge will work, you must have the following:

- VisualAge for Smalltalk Pro Version 4.0 or later installed. See the VisualAge for Smalltalk documentation to confirm that you have the appropriate hardware and software prerequisites available.
- A TeamConnection server that is running.
- A TeamConnection GUI client installed on the machine where you are running your Smalltalk image.

**Note:** This release of the bridge only runs on OS/2 and Windows platforms.

You should verify that you are able to communicate with the relevant TeamConnection server by using the TeamConnection GUI client. If you cannot communicate with the TeamConnection server in this manner, the bridge will definitely not function correctly.

Only certain releases of TeamConnection support the bridge. If you have received the bridge with your release of TeamConnection, you probably have a matching version. If not, then you may have to upgrade your release of TeamConnection. The DLL TCEMBR.DLL should be available to programs in your environment, because this is the DLL that contains the primitives used by the bridge.

#### Environment variables

The bridge relies on the user to specify the various parameters that make up the TeamConnection context. By default, the bridge will query the variables in the

environment that the image is running. These variables, **TC\_FAMILY** , **TC\_RELEASE** , and **TC\_WORKAREA**, are used as initial values for the default TeamConnection context.

There are two additional environment variables that can be defined for the bridge, as follows:

- **TC\_COMPONENT** is used as the default TeamConnection component for parts stored in TeamConnection through the bridge. If **TC\_COMPONENT** is not defined or is empty, the value root is used.
- **TC\_RELATIVE** is used to specify the initial destination path for files retrieved from TeamConnection through the bridge. If **TC\_RELATIVE** is not defined or is empty, the current directory according to the image is used.

It is not necessary to define any of these variables for the bridge to work. Defining them only makes setting up the default bridge configuration in the image easier for a bridge user.

A system administrator may want to have the environment variables automatically defined in a network login script. When a user logs into a LAN and then uses the bridge, the user will be provided with the defined values as hints for setting up the default TeamConnection configuration.

---

## Installing and activating the ENVY/Manager-TeamConnection Bridge

The bridge is loaded into the image like any other configuration map using the **Load** option from the **Editions** menu of the **Configuration Maps Browser**.

Once the bridge is loaded, the submenu **TeamConnection Bridge** will appear on the **Tools** menu of the **System Transcript** window. This submenu is referred to as the *bridge menu*. The bridge menu is the launching point for all of the bridge operations.

“Loading the ENVY/Manager-TeamConnection Bridge” provides step-by-step instructions for the bridge loading process.

## Loading the ENVY/Manager-TeamConnection Bridge

Follow these steps to load the ENVY/Manager-TeamConnection Bridge:

1. Open the VisualAge for Smalltalk Pro - Client.
2. Go to the **System Transcript** window and select **Browse Configuration Maps** from the **Tools** pulldown menu.
3. In the Configuration Maps Browser window, select Import from the **Names** pulldown menu.
4. A dialog will prompt you to enter the full path name of the library that you want to import. For purposes of activating the bridge, you will need to supply a TeamConnection pathname (determined by where you have installed TeamConnection) for the file called TCEMBR.DAT. Select the **OK** pushbutton to continue and display the **Selection Required** window.
5. In the **Selection Required** window, select ENVY/Manager-TeamConnection Bridge in the **Names** list, which will prime the **Versions** list with a version number.
6. Select the version in the **Versions** list and move it to the **Selected Versions** list using the right-arrow pushbutton.

7. For the additional interoperability with VisualAge Generator described in “Chapter 4. Using the ENVY/Manager-TeamConnection Bridge: a simple scenario for VisualAge Generator developers” on page 17, you must also import the configuration map called VAGen ENVY/TC Bridge, as described in the previous steps.
8. Select the **OK** pushbutton to initiate the import process. During the process of importing the TCEMBR.DAT file into the VisualAge for Smalltalk Pro manager.dat file, the **System Transcript** window will issue a message stream that confirms the success of the import.
9. In the **Configuration Maps Browser** window, select ENVY/Manager-TeamConnection Bridge from the **Names** list.
10. Select the item (there should only be one available) in the Editions and Versions list.
11. Select all of the items in the **Applications** list, click mouse button 2, and select **Load** from the pop-up menu.

**Note:** For the additional interoperability with VisualAge Generator described in “Chapter 4. Using the ENVY/Manager-TeamConnection Bridge: a simple scenario for VisualAge Generator developers” on page 17, you must also load VAGen ENVY/TC Bridge, as described in the two previous steps. ENVY/Manager-TeamConnection Bridge *must be loaded first*.

12. After the application loading progress dialog completes without errors, the ENVY/Manager-TeamConnection Bridge should be functional. You can close the **Configuration Maps Browser** window at this time.

## Testing the ENVY/Manager-TeamConnection Bridge

To verify that the bridge is active and ready for ENVY component export/import functions, follow these steps:

1. Go to the **System Transcript** window and select the **Tools** pulldown menu. Then select **Default Properties** from the **TeamConnection Bridge** cascade menu. This will display the **Default Properties** notebook.
2. Verify that the TeamConnection family in the **Family** field on the **Context** page of the **Default Properties** notebook is appropriate for your project. You may need to coordinate your access to the family with your family administrator.
3. Select the **Test Server** pushbutton. If the bridge is properly configured, the server connection test will return an information window that provides server-specific information. Select the **OK** pushbutton to dismiss the server information window.

You are now ready to export ENVY components to a TeamConnection server.

**Note:** When exiting VisualAge for Smalltalk Pro - Client, you should save your image so that the ENVY/Manager-TeamConnection Bridge will be preserved for future use.



---

## Chapter 3. Using the ENVY/Manager-TeamConnection Bridge

You can perform TeamConnection functions on ENVY components, provided that you supply parameters necessary to identify a *bridge configuration*. Bridge configuration parameters are defined by the **Default Properties** notebook, as described in “Setting default properties”.

Each time the bridge interacts with TeamConnection, it uses the parameters in a bridge configuration to ensure that the behavior of the operation is in accordance with the users’ specifications. Because specifying a configuration for each operation would be time-consuming and most operations would use the same configuration, you can specify a default configuration. Each time the user initiates an operation, you can use the default configuration or modify it.

The default configuration is stored in the image so that once it is setup, it will be maintained until the bridge is reloaded from the library.

---

### Setting default properties

To set properties for import and export actions across the ENVY/Manager-TeamConnection Bridge, open the **Default Properties** notebook as described in “Testing the ENVY/Manager-TeamConnection Bridge” on page 7. The **Default Properties** notebook contains four pages of settings, as follows:

- Context page
- Operations page
- Import page
- Export page

Each page of the **Default Properties** notebook includes the following controls:

#### ***Show this dialog when exporting and importing*** checkbox

The **Show this dialog when exporting and importing** checkbox specifies whether the dialog should be shown each time an import/export operation for the bridge is initiated by the user. If the dialog is shown, it gives the user the opportunity the default configuration for a particular operation only.

#### **push buttons**

**OK** Saves the current settings as default setting. This option may not be available if some fields are left incomplete or contain invalid values.

**Cancel** Closes the **Default Properties** notebook and ignores any changes made in the dialog.

#### **Defaults**

Updates the dialog fields with the values in the current default bridge configuration.

**Reset** Updates the dialog fields with the initial values that are set when the bridge is first loaded into the image.

## Context page

The context page is used to specify the TeamConnection family, release, and work area used as the context for the default bridge configuration.

### Family

Use this field to input the name of your TeamConnection family server. Select the **Test Server** pushbutton to return an information window that provides server-specific information. If you cannot successfully communicate with the TeamConnection server, you may have specified an invalid family name. Your TeamConnection family administrator may be of some assistance at this point.

### Release

The TeamConnection release. By selecting the **Query releases** pushbutton, you can prime the **Release** field drop-down menu with valid release choices based on the **Family** field value.

### Work area

The TeamConnection work area in which you will perform TeamConnection actions. By selecting the **Query work areas** pushbutton, you can prime **Work area** field drop-down menu with valid work area choices based on the **Release** field value.

**Note:** Any communication with a TeamConnection server takes time. Querying the available releases and work areas typically takes a few seconds, which is the reason that this data is not automatically used to populate the dialog.

## Operations page

The **Operations** page determines how operations are performed in TeamConnection, including whether operations are forced and how parts in the database are locked.

**Force** The **Force** and **Don't force** radio buttons are mutually exclusive.

In TeamConnection terms, *force* is an indication that changes should be forced into the TeamConnection repository, possibly breaking links with the part in other version contexts. Its intent is to indicate that, although the specified version might not match the current set of versions applicable to the object in the persistent store, the changes in those versions specified in the version string are to be made, breaking the links to those current versions not specified.

The force option is important only if you specify that a part version is to be locked. If you want to retrieve or store a locked part in a particular release or work area that is linked to another release or work area, you might want to specify the force option when you are checking in or checking out the part, even if someone else might have the part checked out in another context. See the discussion of locking below for a description of TeamConnection locking options.

### Locking

These mutually-exclusive radio buttons enable you to instruct TeamConnection cache services (TCCS) on how to manage the locking behavior for parts that you are exporting to or importing from the TeamConnection repository.

**Obtain and release**

Also known as *optimistic* locking, TCCS will attempt to check out the part(s) before checking in changes that you have made in the ENVY environment. If this action is successful, the part(s) will not be locked in TeamConnection after the export.

**Obtain and retain**

TCCS will attempt to check out the part(s) before checking in changes that you have made in the ENVY environment. If this action is successful, the part(s) will remain locked in TeamConnection after the export.

**Retain**

For parts already locked in TeamConnection, after changes are exported from ENVY the locked parts should remain locked (i.e., the lock is *retained* by the original owner).

**Release**

For parts already locked in TeamConnection, after changes are exported from ENVY the locked parts should be unlocked, and therefore available to other developers in that context.

## Import page

The **Import** page provides default settings options when importing ENVY components or files previously exported to a TeamConnection database.

**Configuration Maps**

If the **Import all required maps too** checkbox is checked, it specifies that when a configuration map is imported, its required configuration maps (along with any other required configuration maps, recursively) should be retrieved from TeamConnection as part of the import action. If this option is enabled, and a configuration map being imported does have required maps, the maps can only be imported if they actually exist in the TeamConnection database.

**Note:** The checkbox is checked as the default.

**Destination for Files**

The **Destination path for files** field identifies the target (base) directory for imported files.

**Replacing Existing Files**

These mutually-exclusive radio buttons enable you to select a desired default method for overwriting files (or not) in your working target directory.

**Ask user**

This choice enables you to choose which files are to be overwritten.

**Do not replace existing files**

Files that currently exist in the target directory will not be overwritten.

**Replace existing files**

Files that currently exist in the target directory are automatically overwritten.

## Export page

The **Export** page provides default settings options for exporting ENVY components or files to a TeamConnection database.

### Storage in TeamConnection

The **Component** field identifies the TeamConnection target component for your export action. This component designation, along with TeamConnection family, release, and work area values supplied in the **Context** page of the **Default Properties** notebook, is necessary to define the context for any new TeamConnection parts created by an export action.

### Configuration Maps

If the **Export all required maps too** checkbox is checked, it specifies that when a configuration map is exported, its required configuration maps (along with any other required configuration maps, recursively) should be exported TeamConnection. If this option is enabled, and a configuration map being exported does have required maps, the maps can only be exported if they actually exist in the TeamConnection database.

This option is used to prevent version mismatches when a configuration map requires other configuration maps, as in the following case:

1. For a configuration map that requires other configuration maps, you do an export *with* the required maps.
2. At some later time, you export again *without* the required maps.
3. When you attempt to import *with* the required maps, the import may fail, because a configuration map level in TeamConnection does not match the level previously exported from ENVY.

**Note:** The checkbox is checked as the default.

---

## Exporting ENVY components to TeamConnection

The **TeamConnection Bridge** cascade menu provides an **Export** choice, which offers the following choices:

- Configuration Maps
- Applications
- Files

**Note:** You must have the appropriate authority to update all parts associated with the configuration maps, applications, or files to be exported.

As a general rule, it is advisable to export applications and configuration maps along with any configuration maps required by these ENVY components to avoid version mismatches. If you make a change to an application, it is important to update all the exported configuration maps that contain the application and to export all of the configuration maps again.

**Note:** The **Export all required maps too** checkbox located on the **Export** page of the **Default Properties** notebook defaults to this behavior.

The following describes two simple cases in which a mismatch might occur:

1. Export a configuration map that contains several applications.
2. Make a change to one of the contained applications.



3. Export the changed application only.
4. Attempt to import the configuration map.

**or**

1. Export two configuration maps that contain the same application.
2. Make a change to the common application and export only one of the configuration maps that contains the application.
3. Attempt to import the second configuration map.

As the number of programmers authorized to version components and the complexity of your applications increase, so does the possibility for these types of problem to occur. Therefore, it is important to coordinate update authority in such a way that all affected parties are notified about configuration changes, and that someone in the development group has authority over all levels of components. It may also be advisable to limit export actions to higher levels of authority than you have previously.

Exporting components is a substantial operation that typically takes at least ten to twenty seconds (possibly minutes for a large collection of components). Such an operation begins with the bridge exporting the components to temporary ENVY/Manager libraries and then generating detailed descriptions of the library contents for the benefit of TeamConnection. To guarantee atomicity and minimize the number of times that the bridge must communicate with TeamConnection (thus avoiding unnecessary overheads), all components are transferred in one primitive operation.

Even a single configuration map usually counts as more than one component, because it typically contains at least one release application. Once the primitive operation is invoked, control of the process is in the TeamConnection client code, which is effectively blocked against the TeamConnection server. Because the Smalltalk image is blocked waiting for the primitive to return, the user interface will not update, and the user cannot halt the operation.

## Exporting configuration maps and applications

The process for exporting ENVY-based configuration maps and applications to a TeamConnection family database includes the following steps:

1. Select **Configuration Maps** or **Applications** from the **Export** cascade menu. You will be prompted to select an appropriate version of the configuration map or application that you want to export.

**Note:** ENVY components must be versioned in ENVY before being exported to TeamConnection.

2. In the **Selection Required** window, select the desired configuration map or application in the **Names** list, which will prime the **Versions** list with a version number.
3. Select the version in the **Versions** list and move it to the **Selected Versions** list using the right-arrow pushbutton. Because only one version of any named configuration map can exist in a TeamConnection context, it is only possible to choose one version for any particular name.
4. Select the **OK** pushbutton to initiate the export process.
5. If the **Show this dialog when exporting and importing** option has been set in the default bridge configuration, you will be presented with the **Export Properties** notebook, which is primed by values in the **Default Properties**

notebook. If you are satisfied with the current values in the **Export Properties** notebook, select the **OK** pushbutton to initiate the export process.

If the export succeeds without errors, a message is logged to the **System Transcript** window. Users are informed of any errors with a message box.

## Exporting files

The process for exporting ENVY-based files to a TeamConnection family database includes the following steps:

1. Selecting **Files** from the **Export** cascade menu.
2. You will be prompted to select the files that you want to export. You can add to or delete files from the list using the **Add**, **Remove**, or **Remove All** pushbuttons.
3. Select the **OK** pushbutton to initiate the export process.
4. If the **Show this dialog when exporting and importing** option has been set in the default bridge configuration, you will be presented with the **Export Properties** notebook, which is primed by values in the **Default Properties** notebook. If you are satisfied with the current values in the **Export Properties** notebook, select the **OK** pushbutton to initiate the export process.

If the export succeeds without errors, a message is logged to the **System Transcript** window. Users are informed of any errors with a message box.

---

## Importing ENVY components from TeamConnection

The **TeamConnection Bridge** cascade menu provides an **Import** choice, which offers the following choices:

- Configuration Maps
- Applications
- Files

The process for importing any of these ENVY components is essentially the same, and includes the following steps:

1. Select **Configuration Maps**, **Applications**, or **Files** from the **Import** cascade menu.
2. If the **Show this dialog when exporting and importing** option has been set in the default bridge configuration, you will be presented with the **Import Properties** notebook, which is primed by values in the **Default Properties** notebook.
3. When you are satisfied with the current values in the **Import Properties** notebook, select the **OK** pushbutton.
4. You will be prompted to supply a query pattern to further reduce the number of candidates for import. Select the **OK** pushbutton to launch the query of the TeamConnection context that you have specified up to this point.

**Note:** Use the wildcard characters (\*) and (?) as delimiters for your queries.

5. A list of ENVY components matching your query is returned. Each of these components exists in a TeamConnection database specified by the TeamConnection context in the configuration used for this operation. Select the objects you want to import from this list and select the **OK** pushbutton to initiate the import action.

6. In the case of configuration maps and applications, the selected components will be imported into the default ENVY/Manager library that the image is connected to. For files, the selected files will be written to the path specified by the **Destination path for files** option in the bridge configuration used for this operation. If any of the files already exist, they may be overwritten, or the user may be prompted, depending on the value of the **Replace existing files** option.

If the import succeeds without errors, a message is logged to the **System Transcript** window. Users are informed of any errors with a message box.



---

## Chapter 4. Using the ENVY/Manager-TeamConnection Bridge: a simple scenario for VisualAge Generator developers

The following scenario is a generalized case used to illustrate the way that VisualAge Generator developers might use the ENVY/Manager-TeamConnection Bridge to accomplish change tracking and build processing. An actual implementation requires that a Smalltalk development team begin with versioned ENVY components and a plan for sharing common applications.

After you have installed the ENVY/Manager-TeamConnection Bridge, you can export a versioned configuration map to a TeamConnection family database. For VisualAge Generator developers, this means that you can generate programs, tables, and map groups using the TeamConnection build interface. See the VisualAge Generator *Generator's Guide* and in this document for details.

---

### Scenario assumptions

For purposes of describing the scenario, the following assumptions are established:

- A development team using VisualAge Generator wants to perform problem tracking and build generation.
- A family is created in TeamConnection with a release r1, defined with a track-driver process (i.e., all part changes are made in reference to work areas).
- A build agent and its corresponding build processor has been started to handle build requests for generation, and similarly for preparation.
- Data item definitions and records used to access data in a database are kept in a "common" application, while programs and their other associates are kept in a separate application.

**Note:** This assumption enables the scenarios to include application prerequisites.

---

### Exporting ENVY components to TeamConnection

To prepare for exporting the ENVY components to TeamConnection, perform the following activities:

1. **In TeamConnection:**
  - a. Create a feature called f1 and accept the feature.
  - b. Create a work area called wa1 for implementation of the feature.
2. **In ENVY:**
  - Create applications for the common data and for other VisualAge Generator parts.
3. **In VisualAge Generator Developer:**
  - Create programs and their associates.
  - Create generation option, linkage table, resource association, link edit, and bind parts as necessary.
4. **In ENVY:**
  - a. Create a configuration map that gathers the common data application and the application containing all the other parts.
  - b. The class developers version their classes.

- c. The class owners release versioned classes into the two applications and the application managers version the applications.  
If more than one developer has been working on the feature, each may have opened a new edition of a part's class extension, so a merge of the method editions will have to be performed.
- d. The configuration map manager releases the versioned applications into the configuration map and versions the configuration map.
- e. An administrator uses the ENVY/Manager-TeamConnection Bridge to export the configuration map to the work area associated with feature f1. See "Exporting ENVY components to TeamConnection" on page 12 for ENVY/Manager-TeamConnection Bridge export instructions.

After the ENVY/Manager-TeamConnection Bridge export action is completed, there will be an EmLibrary part for the configuration map and for each of its applications, and proxy parts for each entry in each application's BOM file. The BOM file for each application contains an entry (at least the name, edition/timestamp, and TCPart type) for each class and method in each application.

## Object mapping in TeamConnection

After a ENVY/Manager-TeamConnection Bridge export action, the following parts are created in TeamConnection in wa1 for f1 in the component and release specified as context for the export action:

- For each application of the configuration map there will be an application part.
- For each entry in the BOM, a proxy part with a name qualified by the application name for uniqueness, as described in Table 1.

The ENVY/Manager-TeamConnection Bridge must map ENVY components to part names in TeamConnection in such a way that the parts can be retrieved in a reusable form when they are imported from TeamConnection back into the ENVY environment.

Table 1. Name generation mapping for the ENVY/Manager-TeamConnection Bridge

Class Type	Naming Convention	Mapped Name Example
EmLibrary	<class_name_of_blob_object>.<name_of_blob_object>	EmApplication.MyApp, EmConfigurationMap.MyConfigMap
EmConfigurationMap	<config_map_name>	MyConfigMap
EmApplication	<application_name>	MyApp
EmSubapplication	<app_name>.<subapp_name>	MyApp.MySubapp
EmClass OR EmClassExtension	<app_name>.<subapp_name>.<class_name>	MyApp.MyClass, MyApp.MySubapp.MyClass
EmInstanceMethod OR EmClassMethod	<app_name>.<class_name>.<method_name> OR <app_name.subapp_name>.<class_name>.<method_name>	MyApp.MyClass.MyMethod, MyApp.MySubapp.MyClass.MyMethod

See the VisualAge Generator *Generator's Guide* for additional information related to generation part output names in TeamConnection.

---

## Build generation

The VisualAge Generator *Generator's Guide* provides detailed VisualAge Generator build generation instructions. The following overview is provided to place these activities in the context of the ENVY/Manager-TeamConnection Bridge:

### 1. In VisualAge Generator Developer:

- For each program, the Options Override (OVR) part that has been exported to TeamConnection creates an initial build tree for VisualAge Generator applications in TeamConnection.

Refer to the VisualAge Generator *Generator's Guide* for more details on the OVR part.

### 2. In TeamConnection (build function):

- a. The build administrator selects the EZEPREP collector of the initial build tree of a program proxy in wa1 for f1, and requests a build.
- b. TeamConnection places the generator build event on the build queue, and the generator build agent detects a new build event that it can service.
- c. The build processor invokes the generator build script, which parses the name of the generation configuration map name from the generated part's build parameters.
- d. The build script invokes the generator, which imports the configuration map and its references to the generation ENVY manager. The ENVY manager used by generation is identified by a VisualAge INI file on the generation build server.

The ENVY/Manager-TeamConnection Bridge determines whether each application referenced already exists in the generation Envy manager, and only imports an application if that version of the application is not already in the manager.

**Note:** For VisualAge Generator builds, you can use the environment variable TC\_ENVY\_REFRESH to control when VisualAge Generator builders will import the required configuration map from TeamConnection. TC\_ENVY\_REFRESH can be used to affect the following behaviors:

- If TC\_ENVY\_REFRESH=null, the configuration map will not be imported from TeamConnection if that version of the configuration map is already in the connected Envy Manager.
- If TC\_ENVY\_REFRESH=null, the configuration map will be imported from TeamConnection if that version of the configuration map is not already in the connected Envy Manager.
- If TC\_ENVY\_REFRESH=notnull, the configuration map will always be imported from TeamConnection. This setting is important if you use VisualAge DataAtlas to modify data elements that are dependents of the configuration map. In that case, if the data elements have been modified since the configuration map was exported to TeamConnection, the builder will warn you that the configuration map is not synchronized with its dependent data elements *only if* TC\_ENVY\_REFRESH=notnull. Such a warning allows you to import the changed data elements into a new edition of the configuration map and export the resulting new version to TeamConnection, before trying the build again

Setting TC\_ENVY\_REFRESH is only relevant in the environment of the TeamConnection build server that performs the VisualAge generator builds.

- e. The generator uses the ENVY/Manager-TeamConnection Bridge to update the outputs and dependencies in the build tree.
- f. If there are tables and/or map groups used by the program, the generator determines whether there is already a build tree for them. If not, initial build trees are added for them using the program's OVR part.
- g. TeamConnection re-examines the build tree of the EZEPREP collector and determines that new build events have been added to the build scope for preparation of the generation outputs, and possibly for generation and preparation of tables and map groups. Build events are started to complete the preparation of generation outputs, and generation and preparation of tables and map groups if necessary.

**Note:** See the VisualAge Generator *Generator's Guide* for greater detail on this process.

3. **In TeamConnection (change control):**

- A project administrator completes the fix record(s) for the feature f1 and adds the work area wa1 to a system test driver. Eventually the driver is committed to the release and the feature is completed.

---

## Making a change to a member

1. **In TeamConnection:**

- a. Defect d1 is created and accepted in TeamConnection
- b. Work area wa2 is created for the implementation of the defect d1.

2. **In ENVY:**

- a. An application manager creates new edition of an application that requires a change.
- b. A developer makes a change to one or more parts.
- c. The class developer of the changed parts versions the class, the class owner releases the class into the new edition of the application, and the application manager versions the application. If more than one defect is in progress, the class owner must release only the versions that apply for defect d1.
- d. The configuration map owner opens a new edition of the configuration map used to generate the program being changed, and releases the new application version into the configuration map. The configuration map owner versions the configuration map.
- e. The administrator uses the ENVY/Manager-TeamConnection Bridge to put the configuration map back into the work area wa2 for defect d1.

3. **In TeamConnection (build function):**

- a. Build administrator builds the program(s) affected by the change. This can be done by selecting the preparation collector for each program and requesting a build, or by selecting a collector for a subsystem, and building the subsystem collector. Only programs, tables, or map group affected by the changes to proxy members will be rebuilt.
- b. The generation process continues as it did for the initial build (see "Build generation" on page 19 for details), except that there should be no need to add new build trees unless a new table or map group was added to a program being built



4. **In TeamConnection (change control):**

- A project administrator completes the fix record(s) for the defect d1 and adds the work area wa2 to a system test driver. Eventually the driver is committed to the release and the feature is completed.



Part Number: 33H2571



Printed in the United States of America  
on recycled paper containing 10%  
recovered post-consumer fiber.

SC34-4551-00



33H2571

