

Migration to VisualAge TeamConnection Version 3

Migration from CMVC 2.3.1 to VisualAge TeamConnection Enterprise Server Version 3

Kevin Postreich, Lee Perlov, Angel Rivera, Mark Dunn

TR 29.3113

VisualAge TeamConnection Development
IBM Software Solutions
Research Triangle Park, North Carolina, USA

© Copyright 1999, IBM Corp.

Migration to VisualAge TeamConnection Version 3

Migration to VisualAge TeamConnection Version 3

ABSTRACT

This technical report provides a procedure and tools that aid in the migration of a Configuration Management and Version Control (CMVC) Version 2.3.1 family to its successor product, VisualAge TeamConnection Enterprise Server Version 3. The objective is to provide a structured process, to reduce errors and to save time by selecting only appropriate data for the migration.

ITIRC KEYWORDS

- VisualAge
- TeamConnection
- CMVC
- Migration

Migration to VisualAge TeamConnection Version 3

ABOUT THE AUTHORS

Kevin Postreich

Mr. Postreich is a staff software engineer with the VisualAge TeamConnection development group. He joined IBM in 1980 as an electronic engineer in Charlotte, North Carolina. He relocated to RTP as an MVS systems programmer.

Mr. Postreich is currently a member of the TeamConnection development/test team, and a member of the first defense team for customer support.

Lee R. Perlov

Mr. Perlov is an advisory software engineer in the VisualAge TeamConnection/CMVC development team. He started working for IBM in 1985 in Gaithersburg, Md, working in the Federal Systems Division on various projects for the United States intelligence community. He then moved to RTP to work on library development and support.

Mr. Perlov received a B.S. degree in Accounting from the University of Florida in 1983. He also completed two years of graduate work in the Department of Computer Science at the University of Florida.

Mr. Perlov is currently a member of the VisualAge TeamConnection Services team, as well as family, system and web administrator for the services TeamConnection family.

Angel Rivera

Mr. Rivera is an advisory software engineer with the VisualAge TeamConnection/CMVC development team. He joined IBM in 1989 and since then has worked in the development and support of library systems.

Mr. Rivera has an M.S. in Electrical Engineering from The University of Texas at Austin, and a B.S. in Electronic Systems Engineering from the Instituto Tecnológico y de Estudios Superiores de Monterrey, Mexico.

Mr. Rivera is currently the team lead for the development of CMVC 2.3 and provides service support for customers of VisualAge TeamConnection.

Mark Dunn

Mr. Dunn is a advisory software engineer with the VisualAge TeamConnection development group. He rejoined IBM in 1996 and is currently in charge of development and support of TeamConnection migration.

Mr. Dunn has a B.S. degree in Biology from Western Kentucky University and a B.A. degree in Computer and Information Systems from Florida Atlantic University.

Migration to VisualAge TeamConnection Version 3

ABSTRACT	- 3 -
ITIRC KEYWORDS	- 3 -
ABOUT THE AUTHORS	- 4 -
Introduction	- 10 -
Limitations on the migration from CMVC to VisualAge TeamConnection	- 11 -
Acknowledgments	- 11 -
How to get the most up to date version of this technical report	- 12 -
Overview of the migration process	- 13 -
General suggestions	- 13 -
Notes for practice migrations	- 13 -
Preparing the migration plan	- 14 -
Important assumptions about the migration from CMVC	- 14 -
Planning considerations	- 14 -
Important migration notes about the sequence number for defects and features	- 15 -
Whether to clone or not to clone a CMVC family	- 15 -
Whether to prune a CMVC family	- 16 -
Disk space considerations	- 16 -
Time considerations	- 16 -
Day by Day Plan	- 16 -
Day: TC - 1 day (family servers)	- 16 -
Day: TC - 1 day (clients and build servers)	- 17 -
Day: TC - 1 day (users and usability)	- 17 -
Day: TC - 1 day (test migration)	- 17 -
Day: TC (server side)	- 18 -
Day: TC (client side for the test migration)	- 18 -
Day: TC + 1 day (server side)	- 18 -

Migration to VisualAge TeamConnection Version 3

Gathering all the materials	- 19 -
Finding documentation about the migration process	- 19 -
Migration tools	- 19 -
Checklist of material related to CMVC and TeamConnection	- 20 -
Prepare the CMVC family for the migration	- 22 -
Checklist for migrating a CMVC family from 2.1, 2.2 or 2.3.0 to the new 2.3.1	- 22 -
Checklist for preparing the CMVC 2.3.1.x family for migration	- 23 -
What to do to CMVC before migration	- 25 -
Significant differences between CMVC and VisualAge TeamConnection	- 27 -
Prepare the target VisualAge TeamConnection family	- 28 -
Checklist for creating a new family in VisualAge TeamConnection	- 28 -
Check for FixPaks and HotFixes	- 29 -
Important information about installing the code and creating user ids	- 29 -
Create family using tcadmin or dbcreate	- 30 -
Environment setup for running mkmiglst	- 30 -
Performing the migration	- 31 -
Checklist for migrating the CMVC family to VisualAge TeamConnection	- 31 -
Background on mkmiglst	- 32 -
Compiling mkmiglst	- 32 -
Usage statement of mkmiglst	- 32 -
Help statement of mkmiglst	- 32 -
Description:	- 32 -
Details:	- 33 -
Options:	- 33 -
Notes:	- 33 -

Migration to VisualAge TeamConnection Version 3

Example of using mkmiglst to generate the migration script	- 34 -
Example of migration file 1 generated by mkmiglst: myfile1.lst	- 34 -
Example of migration file 2 generated by mkmiglst: myfile2.lst	- 35 -
Example of file with migration instructions generated by mkmiglst: myfile.txt	- 36 -
Hints and tips for migrating the data from CMVC	- 38 -
Maintenance mode	- 38 -
Take note of the time that it took to perform the test migrations	- 38 -
Qualify appropriate migration steps	- 38 -
Redirect output during migration	- 38 -
Executing the migration scripts	- 39 -
Updating the configuration files	- 39 -
Updating the configurable fields	- 40 -
Updating the user exits	- 40 -
Be aware of the difference in user exits between TeamConnection and CMVC	- 40 -
Verification of the new TeamConnection family	- 41 -
Checklist for verifying the new family in VisualAge TeamConnection	- 41 -
Post-migration tasks for the new TeamConnection family	- 42 -
Checklist for performing post-migration tasks for the new TeamConnection family	- 42 -
Updating the sequence number for defects and features	- 43 -
Our experiences migrating to TeamConnection Version 3	- 44 -
Migration decisions	- 44 -
Do we migrate all of our releases or just the active ones?	- 44 -
Do we migrate the change history or just the currently committed version of each file?	- 44 -
Do we clean up the family before the migration?	- 44 -
Results of migration decisions	- 44 -

Migration to VisualAge TeamConnection Version 3

Migration from CMVC95	- 46 -
AIX Environment Variables	- 46 -
Required	- 46 -
Optional	- 46 -
Decisions that will greatly increase the migration times	- 46 -
Migrating of CMVC95 Notes	- 47 -
Migrating from Oracle, Informix or Sybase	- 47 -
Migrating from DB2	- 47 -
Migration shell scripts	- 48 -
Updating the CMVC ChangeView (DB2, Oracle, Sybase, Informix): CmvcUpdateChangeView.ksh	- 48 -
Import a release into CMVC before migration: file_import.ksh	- 48 -
Automated verification tool: migVerify.ksh	- 48 -
Renaming Defects That have the same name as existing Releases in CMVC: processDefects.ksh	- 49 -
Renaming Levels That have the same name as existing Tracks in CMVC: processLevels.ksh	- 49 -
Reassign user's work and delete users: reassign_work.ksh	- 49 -
Completing a release in CMVC before migration: release_complete.ksh	- 50 -
Delete a release from a CMVC family: release_delete.ksh	- 50 -
Warning and Error Messages	- 51 -
Obtaining the Tools	- 52 -
IBM Intranet	- 52 -
Web Home Page for TeamConnection	- 52 -
FTP site for TeamConnection	- 52 -
Public Internet	- 52 -
Web Home Page for TeamConnection	- 52 -
FTP site for TeamConnection	- 52 -
Obtaining Info-ZIP	- 53 -

Migration to VisualAge TeamConnection Version 3

How to unzip files	- 53 -
Copyrights, Trademarks and Service marks	- 54 -

Migration to VisualAge TeamConnection Version 3

Introduction

This technical report provides a procedure and tools that aid in the migration of a Configuration Management and Version Control (CMVC) Version 2.3.1 family to its successor product, VisualAge TeamConnection Enterprise Server Version 3.

This technical report is a complement to (rather than a replacement of) the published documentation provided with VisualAge TeamConnection Enterprise Server Version 3.

The current Migration facility (migcmvc) to migrate a CMVC family to VisualAge TeamConnection is a general purpose tool with considerable capability and flexibility. Unfortunately, it does not enforce a methodology and this may result in mistakes that could be avoided if the user follows a well defined procedure.

The objectives of this technical report are:

- To describe a structured process.
- To provide checklists to help you execute each migration step.
- To provide tools for migration and hints on how to use them.
- To reduce errors and to save time by selecting only appropriate data for the migration.
- To provide a tool is provided to help you verify the success of the migration.

We discuss also our experience of migrating from CMVC 2.3 to VisualAge TeamConnection Version 2 and then to Version 3.

We provide also the specific migration details to migrate from CMVC95 (similar to CMVC 2.3.1, available only for IBM internal users).

Finally, we provide the details on the tools used in this document and how to get them (see *Obtaining the Tools* at the end of this document).

Notes:

- CMVC 2.3.1 is a prerequisite for migration to VisualAge TeamConnection, due to support of the **Report -general** command used during migration.
- This technical report refers to VisualAge TeamConnection V3.01 (fixpak 3.0.1) which has significant enhancements to the migration tool.
- Important enhancements (hot fixes) to the migration tool are occasionally made available between versions and fixpaks. These **hotfixes** are available on the external web:
<http://www.software.ibm.com/ad/teamcon/downloads/>
When you are placed in the appropriate FTP site, go to the directory **hotfixes**.

Migration to VisualAge TeamConnection Version 3

Limitations on the migration from CMVC to VisualAge TeamConnection

The Report -general action was added in CMVC 2.3.1.2 to allow the migration of ChangeView. The implementation of Report -general has the following limitations:

- The Report -general was only implemented on the CMVC client for Unix and not for the CMVC client for Intel, therefore you cannot migrate into an Intel TeamConnection family.
- As of CMVC 2.3.1.4, the Report -general has not being implemented yet for the Sybase databases. It is planned to implement this support in a future refresh for CMVC. In the meantime, you may want to migrate the CMVC family on Sybase to another supported database, and then to migrate to TeamConnection.
- As of CMVC 2.3.1.4, the Report -general does not work on Oracle databases for HP-UX and Solaris. However, the CMVC server for Oracle on AIX works fine with Report -general. It is planned to fix this defect in a future refresh of CMVC. In the meantime, you may want to migrate the CMVC family to another supported combination, and then to migrate to TeamConnection.
- Based on the above, the list of CMVC combinations that support Report -general is:
 - AIX 4:
 - DB2 V2
 - DB2 UDB V5
 - Oracle 7.1 and 7.2
 - Oracle 7.3
 - Informix 7
 - HP-UX:
 - Informix 7

Acknowledgments

Many of the questions and answers that are compiled in this technical report were obtained from the TEAMC and CMVC forums in the IBMPC conferencing disk and from the CMVC6000 forum in the IBMUNIX conferencing disk. We want to thank the main participants in these electronic forums for their support!

We want to thank in particular the following coworkers:

- Gary Greig, CMVC95 support team, IBM Austin, Texas
- Keith Purcell, OEM Lab in IBM RTP, North Carolina.
- Jim Austin, AIX Lab in IBM RTP, North Carolina.
- Jon Gregory, TeamConnection team, IBM RTP, North Carolina.
- Brent Wunderlich, TeamConnection team, IBM RTP, North Carolina.
- Clifford Meyers, TeamConnection team, IBM RTP, North Carolina
- Tim Orłowski, TeamConnection team, IBM RTP, North Carolina.

Migration to VisualAge TeamConnection Version 3

How to get the most up to date version of this technical report

The most up to date version of this technical report can be obtained from the IBM VisualAge TeamConnection Enterprise Server:

- Library home page by selecting the item **Library** at URL:
<http://www.software.ibm.com/ad/teamcon>
- FTP site by accessing the URL (See the file README.index.txt for details):
<ftp://ftp.software.ibm.com/ps/products/teamconnection/papers>

Migration to VisualAge TeamConnection Version 3

Overview of the migration process

General suggestions

Here is a broad overview of the steps needed to migrate a CMVC family to VisualAge TeamConnection. The ordering of the chapters in this technical report reflect this migration sequence:

1. Prepare the migration plan.
 - We strongly suggest a rigorous plan for migrating CMVC families to TeamConnection that requires the participation of your entire team.
 - Train users on TeamConnection prior to migration. Key differences are identified in the technical report **Comparison between CMVC 2.3.1 and VisualAge TeamConnection Enterprise Server 3**.
2. Gather all materials needed for planning and migration.
3. Prepare the CMVC family for migration.
4. Prepare the new VisualAge TeamConnection family.

A new family needs to be created in VisualAge TeamConnection in order to have a place for the data to be migrated from the CMVC family.
5. Perform the migration.

It is recommended that you perform a practice migration before the real migration. Whether the migration is for practice or a real one, these are the basic steps:

 - Select the releases to be migrated.
 - Construct a migration script using **mkmiglst**.
 - Review the generated migration list and make changes as needed.
 - Migrate the releases using the instructions generated by **mkmiglst**.
6. Verify the new family.
 - Verify the migration in Unix using **migVerify**.
7. Perform post-migration tasks on the new family.

Notes for practice migrations

- Your team should extensively exercise your new TeamConnection family after migrating from CMVC for a very limited period of time, such as one full day. Do everything that you do normally in your development and reporting efforts. You can get a feel for a day's activity by looking at your CMVC audit log for a given family. The audit log is \$HOME/audit/log of the CMVC family user id.
- At the end of testing, have a quick meeting to assess the results and refine the plan for the final migration.
- Once you pass a day of testing your new TeamConnection family with no "show-stoppers", then discard this testing TeamConnection family.
- Finally, perform the migration again for real, and start using your new TeamConnection family.

Migration to VisualAge TeamConnection Version 3

Preparing the migration plan

This chapter provides several checklists that can be used to determine the progress of the migration process.

Notes:

- The Estimated Time for those tasks that take more than 5 minutes has been rounded up to the nearest ½ hour.
- It assumes that all the prerequisites have been completed.

	Description of task	Est. Time	Comment
	Preliminary planning activities.	½ day	See the <i>Planning Considerations</i> in this chapter.
	Prepare day-to-day plan.	½ day	See <i>Day by Day Plan</i> in this chapter

Important assumptions about the migration from CMVC

Throughout this document, we will assume the following:

- You should plan to do practice migrations because there are things you need to find out for yourself.
- You should include in your plan enough time to do backups of the source CMVC family and of the target VisualAge TeamConnection family at several points in the migration (such as just before the migration, after all the data except the files have been migrated, and after all the files have been migrated).
- Take into account that the migration facility is very flexible and thus, you can expect to decide to change options and try again.
- Most tools described in this document are written for Unix. However, **mkmiglst** is written in C, and includes instructions for compiling it on Intel and on Unix.

Planning considerations

Moving your data from a CMVC family to a TeamConnection family will require careful planning. These are the questions you need to ask yourself:

- Will all of the releases from my current CMVC family be migrated, or only some of them?
- If not all of the releases will be migrated, are there any releases that need to be kept in CMVC as reference only, while other releases will be migrated to be used in future development?
- How important is the history of each file change? Could the history of each file change be left in CMVC (which means that the CMVC family needs to be active) as a reference family and migrate only the latest file change to TeamConnection? If only the last file change will be migrated, then migrating only the latest file change will save time during the migration and disk space with the new TeamConnection family.

Migration to VisualAge TeamConnection Version 3

- Is there enough disk space to keep both CMVC and TeamConnection families at the same time in the same host?
- Is there enough CPU to run CMVC and TeamConnection families at the same time in the same host?
- Are NFS mounts for holding the version control tree data for the CMVC family data being used? If so, after migrating the CMVC family you may not need them anymore if your CMVC family is retired from service.
- How much time can the CMVC family be placed in maintenance mode in order to perform the migration? Because the migration utility will request data from the CMVC family, this family should be up and running; however, it is not desirable to allow users to modify the CMVC family while its data is being migrated. For this reason it is recommended to place the CMVC family in **maintenance mode** (that is, read-only).
- Decide how you are going to stage your migration from CMVC to TeamConnection: if you have multiple releases or even multiple families, then the migration may take more than one day of work, so you need to be ready to run both CMVC and TeamConnection in parallel while all the families are being migrated.

The answers to these questions are critical because they can help you determine whether you will need to:

- Clone any CMVC families from your current host to another host that has more resources and/or you want to continue working with some releases in CMVC after migration.
- Prune data from CMVC families (using our shell scripts).
- Remove NFS mounts, add disk space, move around disk space, etc.
- Limit the amount of data to be migrated to save time or disk space; for example, get a snapshot of a release instead of the entire file change history.

Important migration notes about the sequence number for defects and features

If you do not migrate all the defects and features from the CMVC family (including all completed and canceled ones), then there might be gaps in the numbering sequence for the defects and features that will be opened in the TeamConnection family after the migration.

In order to avoid these gaps being filled in by TeamConnection, you will need to find out what is the latest sequence number for a defect in CMVC and then update with this number the sequence number for defects in TeamConnection. In that way you will avoid having one defect in TeamConnection that has the same number as a closed defect in CMVC but that refer to different problems.

For details on how to modify the sequence number in the target TeamConnection family, see section *Updating the sequence number for defects and features*.

Whether to clone or not to clone a CMVC family

If you are going to keep some releases in a CMVC family around for reference and if you have other releases that are going to continue to stay in production in CMVC, then you should consider cloning your CMVC family.

Migration to VisualAge TeamConnection Version 3

By duplicating the contents of a CMVC family in another family, you can make one read-only and continue work in the other. Cloning a CMVC family is documented in the technical report **CMVC Frequently Asked Questions: Migration of Families**.

Whether to prune a CMVC family

Pruning a CMVC family is a good way to save space after migrating releases to TeamConnection or cloning a family. Pruning a CMVC family eliminates confusion about which release in which family is the one to be updated.

Disk space considerations

CMVC and TeamConnection use lots of disk space, as does all software development, therefore, we recommend pruning any duplicate releases, for example:

- After migrating a release to TeamConnection use the pruning tool to remove the migrated release from the CMVC family.
- After cloning a CMVC family into a read-only CMVC clone family and a CMVC development family, use the pruning tool **release_delete.ksh** to make sure there is only one copy of each release across CMVC families. The pruning tool will remove files from the vc directory of the CMVC family whenever possible, thus saving the most disk space. See the section **Migration Shell Scripts** in this document.

Time considerations

Time spent migrating your families is time that your families are down, therefore the less data you migrate the less down time for your families. This is one of the best reasons to migrate only the releases you plan to continue developing. Leaving the other releases in CMVC will get your TeamConnection family up and available much faster.

In order to save time and effort, once you have migrated the active releases, put the CMVC family in read-only mode and remove it from your backup list. There is no reason to backup a family that does not change.

Day by Day Plan

You have to begin by selecting an appropriate day in which all the members of your team will test the migrated family. Select an appropriate target date and call it **TC day**. Based on that target date there are several activities that need to be done prior, during and after that date.

Day: TC - 1 day (family servers)

Source system (CMVC):

- Ensure that all users of the new TeamConnection family have the proper host lists and access in the current CMVC family. In that way, the new family will be ready to handle the authentication of users after the migration from CMVC.

Migration to VisualAge TeamConnection Version 3

- Get the migration tools mentioned in this document. See *Obtaining the Tools* at the end of this document.

Target system (VisualAge TeamConnection):

- If you have a pilot project with beta code from TeamConnection, you may want to delete the beta TeamConnection code from the server host.
- Remove any old versions of TeamConnection so that the server points to the current code.
- Install TeamConnection and DB2 updates from the latest fixpak. Follow the directions in the README.TXT and INSTALL.* (INSTALL is available in .TXT, .PDF and .HTM). See the section *Gathering all the materials*.
- Plan the backup and restore strategy for your new family.
- Get the migration tools mentioned in this document. See *Obtaining the Tools* at the end of this document.

Day: TC - 1 day (clients and build servers)

- Ensure that everybody in the team has installed the TeamConnection client with the latest FixPak and HotFix.
- Remove any old versions of TeamConnection, such as the beta versions used in a pilot project.
- Prepare a list of team members that will test the new TeamConnection family from unique combinations of client platforms, such as Windows NT and AIX. The idea is to ensure that all relevant client platforms can work with TeamConnection in your network configuration.

Day: TC - 1 day (users and usability)

- Schedule and deliver training sessions for the users of the CMVC family to provide them with the essentials of how to use a TeamConnection family. A starting point is to present the technical report **Comparison between CMVC 2.3.1 and VisualAge TeamConnection Enterprise Server 3**.
- Review with key users how you use your CMVC family (such as component structure, naming convention) and decide what changes, if any, will be done on the migrated data.
- Do not take for granted that the current setup in the CMVC family is perfect and keep in mind that it might be possible to do some fine tuning with the setup in the new TeamConnection family just after the migration but before overall production use.

Day: TC - 1 day (test migration)

- Migrate the latest snapshot of the CMVC family into the new test TeamConnection family.
- The new TeamConnection family is started and a migration test is done by selected team members to ensure that so far the migration went fine and that the family is ready for TC day.
- The new TeamConnection family should be up for a few hours to allow the users to test their access to the family.
- Rehearse the backup and restore strategy for the new test family.
- All team members ensure that they are ready for TC day: they should be able to connect to the new TeamConnection family from the connectivity point of view. This can be checked by

Migration to VisualAge TeamConnection Version 3

issuing the command **ping \$TC_FAMILY** (where \$TC_FAMILY is the name of the family) and if this is successful, then issue the command **teamc report -testServer**.

- They should have the proper host list and access list. This can be checked by issuing the command **teamc user -view userName**, where userName is the name of each user.

Day: TC (server side)

- The new TeamConnection family is started with the desired number of family servers, which is roughly the cubic root of the total number of users. For example, if you have 50 users, the closest cubic root is 4 (for 64): **teamcd \$TC_FAMILY 4**
- The TeamConnection family administrator and selected members of the team should monitor the new family, paying particular attention to not running out of disk space. The TeamConnection **monitor** tool can be used to watch the activity in the TeamConnection family. Also, the family **\$TC_DBPATH/audit.log** file can be monitored by using: **tail -f audit.log**
- The family administrator may decide to stop the family for few minutes during the middle of the day and restart it again, just to see if there are any problems with this action.
- At the end of the day, the TeamConnection family will be shut down.
- Perform a backup of the family and then do a restore, to test your backup/restore strategy.

Day: TC (client side for the test migration)

- Every member of the team will spend time interacting with the test TeamConnection family. First with the specific assignments and then later on in an exploratory mode.
- Once there are several workareas and drivers, you should try to do a "normal" build. That is, extract the release and then start builds in all the platforms from the extracted code.
- At the end of the day, you should have a short meeting to discuss the results and if necessary, plan to fix the main problems and if necessary repeat the test (at least, a scale down version of it, such as just for half a day).
- Once you and your team are comfortable with the migration results of the test TeamConnection family, proceed to perform the real migration.

Day: TC + 1 day (server side)

- The test TeamConnection family should be deleted in order to prepare the next test (if needed) or the final migration.
- Perform the final migration for the family by repeating the tasks described in **Day: TC**. Keep in mind that this time the migration is not a test anymore; it is the final migration.
- Start the family servers for the new family.
- Perform the verification of the family.

Migration to VisualAge TeamConnection Version 3

Gathering all the materials

Finding documentation about the migration process

The single largest challenge when performing a migration to VisualAge TeamConnection Enterprise Server Version 3 is sifting through the available documentation in order to perform all of the necessary actions. Here is a brief overview:

1. Read the **README.TXT** and **RELNOTES.PDF** files. This file points to all other installation documents. For English, it is located in the fullpak CD-ROM as follows:
 - Intel:** readme.txt and *nationalLanguage/relnotes.pdf*
 - Unix:** readme.txt and *nationalLanguage/relnotes.pdf*See in the readme.txt for v301, "Migration Notes for Version 3.0.1.
2. Read the migration directions in **Chapter 8, Migrating to TeamConnection version 3** in the **VisualAge TeamConnection Installation Guide** (choose the desired format: **INSTALL.PDF**, **INSTALL.HTM** or **INSTALL.TXT**) from the fullpak CD-ROM:
 - Intel:** nls/doc/*nationalLanguage/install.pdf*
 - Unix:** softpubs/*nationalLanguage/install.pdf*
3. Use this technical report to plan for the actual migration.

Notes:

* The installation and migration sections of the **Administrator's Guide Version 3.0.0** are out-of-date, therefore, do not use them. The version 3.0.1 of this manual does not have anymore the migration instructions, because starting with fixpak 3.0.1, they are now contained in the Installation Guide.

* You need Acrobat Reader to view the documentation in PDF format; you can download it from the VisualAge TeamConnection web site:

<http://www.software.ibm.com/ad/teamcon/downloads>

Migration tools

The migration tools mentioned in this technical report are packaged in a zip file (**migrate.zip**). Because we may update the migrate.zip, we are showing below only some of the tools that are included:

- Unix: **mkmiglst.c**, migVerify.ksh, release_complete.ksh, reassign_work.ksh, release_delete.ksh, file_import.ksh, CmvcUpdateChangeView.ksh, processLevels.ksh, processDefects.ksh
- OS/2: **mkmiglst.c**, find.exe, rm.exe, grep.exe, tail.exe, tee.exe, which.exe
- Windows NT: **mkmiglst.c**, find.exe, grep.exe, rm.exe, tail.exe, tee.exe, which.exe

Notes:

- The shell scripts (suffix .ksh) are described in the section **Migration Shell Scripts** in this document.

Migration to VisualAge TeamConnection Version 3

- The **mkmig1st** program is described in the section **Performing the Migration** chapter of this document.

Checklist of material related to CMVC and TeamConnection

	Document/Media Name	Current Version
	CMVC code, available from the CMVC ftp sites, see note 1)	2.3.1.4
	CMVC Server Administration and Installation (manual)	2.3
	Technical Report: CMVC Frequently Asked Questions: Migration of Families	CMVC ftp site. File trcmfmig.pdf
	VisualAge TeamConnection Enterprise Server V3 CD-ROM	3.0.0
	VisualAge TeamConnection Enterprise Server V3 FixPak	3.0.1
	VisualAge TeamConnection Enterprise Server V3 HotFix(s)	contact IBM
	VisualAge TeamConnection Enterprise Server V3: Administrator's Guide and Installation Guide (softcopy)	3.0.1
	Technical Report 29.3076: Configuration and Administration of DB2 Universal Database V5 by users of VisualAge TeamConnection Enterprise Server V3	TeamConnection ftp site. File trtc3db2.pdf
	Technical Report 29.3094: VisualAge TeamConnection Version 3: how to do routine operating system tasks	TeamConnection ftp site. File trtc3oem.pdf
	Technical Report 29.3096: Comparison between CMVC 2.3.1 and VisualAge TeamConnection Enterprise Server 3.	TeamConnection ftp site. File trtc3-cm.pdf
	Technical Report 29.3032: Making the most of VisualAge TeamConnection user exits.	TeamConnection ftp site. File trtcusrx.pdf
	Migration tools, packaged in the file migrate.zip .	TeamConnection ftp site. File migrate.zip

Notes:

1. To get the latest CMVC 2.3.1.x code, you can access the following sites:
 - Public Internet URL: <ftp://ftp.software.ibm.com/ps/products/cmvc/>
 - IBM intranet URL: <http://tc-cmvc.raleigh.ibm.com/cmvc/>
2. To get the CMVC 2.3 manuals, you can access the following sites:
 - Public Internet URL: <ftp://ftp.software.ibm.com/ps/products/cmvc/doc/>
 - IBM intranet URL: <http://tc-cmvc.raleigh.ibm.com/cmvc/doc/>
3. To get the CMVC 2.3 technical reports, you can access the following sites:
 - Public Internet URL: <ftp://ftp.software.ibm.com/ps/products/cmvc/doc/tr/>
 - IBM intranet URL: <http://tc-cmvc.raleigh.ibm.com/cmvc/doc/tr/>
4. To get the latest VisualAge TeamConnection Version 3 code, you can access the following sites:
 - Public Internet URL: <http://www.software.ibm.com/ad/teamcon/>

Migration to VisualAge TeamConnection Version 3

5. To get the latest VisualAge TeamConnection Version 3 tools, you can access the following sites:

- Public Internet URL: <ftp://ftp.software.ibm.com/ps/products/teamconnection/tools/>
- IBM intranet URL: <http://wwwcmvc.raleigh.ibm.com/tools.html>

6. To get the latest VisualAge TeamConnection Version 3 technical reports, you can access the following sites:

- Public Internet URL: <ftp://ftp.software.ibm.com/ps/products/teamconnection/papers/>
- IBM intranet URL: <http://wwwcmvc.raleigh.ibm.com/papers.html>

Migration to VisualAge TeamConnection Version 3

Prepare the CMVC family for the migration

Checklist for migrating a CMVC family from 2.1, 2.2 or 2.3.0 to the new 2.3.1

The following steps are needed to migrate a CMVC family that is currently using CMVC version 2.1.x, 2.2.x, or 2.3.0.x to the new CMVC 2.3.1.x.

Source system: These activities are done in the host system where the CMVC family is located. This system will be called the “source system”.

	Description of task	Est. Time	Comments
	Make a complete backup of the original CMVC family.	1 hr	See section How to make a complete backup of your CMVC family of the technical report CMVC Frequently Asked Questions: Migration of Families
	Install the CMVC 2.3.1.x code	½ hr	See the installation instructions provided with the 2.3.1 code.
	Update from 2.1, 2.2 or 2.3.0 to the new 2.3.1.	½ day	See Appendix B Migrating a to CMVC Version 2.3 from the manual CMVC Server Administration and Installation Version 2.3
	Update the CMVC family to 2.3.1.x by running: dbSetDate Database Family outFile Example: dbSetDate DB2 cmvcfam setdate.log	1 hr	See the installation instructions provided with the 2.3.1 code. The dates will be changed from 2 digits for the years, to 4 digits. That is, the dates will be Year 2000 ready.
	Make a backup of your upgraded CMVC 2.3.1 family.	1 hr	See appropriate comment above.

Migration to VisualAge TeamConnection Version 3

Checklist for preparing the CMVC 2.3.1.x family for migration

Note: You must have a CMVC family running on 2.3.1.x in order to migrate to VisualAge TeamConnection.

Source system: These activities are done in the host system where the CMVC family is located. This system will be called the **source system**.

Description of task	Est. Time	Comments
As the CMVC superuser, add host list entries for the new host where the TeamConnection family will reside: Host -create login@host Where <i>login</i> is the system login and <i>host</i> is the new host where the TeamConnection family will reside.	1 min	If you do not add these host list entries, you will not be able to access the new family using the default authentication which uses host list entries (HOST_ONLY).
Run CmvcUpdateChangeView.ksh to update ChangeView of the CMVC family.	5 min	This script is available from migrate.zip.
Prepare the releases to be migrated by running: release_complete.ksh	1 hr	Moves all active work to the completed state. This is currently required for migration.
Rename levels that have the same name as existing tracks by running: processLevels.ksh	½ hr	VisualAge TeamConnection does not support identically named drivers/workareas.
Rename Defects that have the same name as an existing release by running: processDefects.ksh	½ hr	VisualAge TeamConnection does not support identically named defects/releases.
Execute Report -view defectView . If it fails, then fix it by running: chfield -object Defect -source SHOME chfield -object Feature -source SHOME	½ hr	The migration utility uses the Report command.
Find out the value of the latest defect number from the Sequence table. See section What to do to CMVC before migration .	5 min	After the migration, this number should be given to the defect number in the Sequence table in TeamConnection.
Backup the database and the VC tree for the CMVC family (\$FAMILY_HOME/vc)	1 hr	See section How to make a complete backup of your

Migration to VisualAge TeamConnection Version 3

		CMVC family of the technical report CMVC Frequently Asked Questions: Migration of Families
Restart the CMVC family in maintenance mode by running: cmvcd -m <i>familyName</i> N Where <i>N</i> is the number of daemons (at least 2).	5 min	The family should be active but in read-only mode, to avoid the data being modified while it is migrated.
Disable the automated shell scripts that might affect the availability of the source CMVC family or the database system. This is to prevent a possible interruption of the migration.	½ hr	For example, if you start the migration in the evening, it may take several hours to migrate the parts. Thus, you want to avoid an automatic daily backup cron job at midnight to interrupt the migration.

Migration to VisualAge TeamConnection Version 3

What to do to CMVC before migration

The TeamConnection migration utility will use CMVC client commands. The only requirement from TeamConnection are:

- The CMVC client be installed on the target system where the TeamConnection server is installed.
- If the CMVC family is using CMVC 2.1 or 2.2 or 2.3.0, the CMVC family must be upgraded to 2.3.1.2, or later. Alternately, the family can be cloned and the clone can be upgraded to CMVC 2.3.1.2 or later.
- For a CMVC family that was recently migrated to 2.3.1, as the CMVC family administrator, run only once the following CMVC tool to modify the dates in all tables in order to expand them from the representation that uses only 2 digits for the years to the representation that uses 4 digits. In other words, this tool makes the data Year 2000 ready.
 - Usage: `dbSetDate Database Family outFile`
 - Example: `dbSetDate DB2 cmvcfam setdate.log`

There are many preparation tasks that can be performed in the CMVC family prior to migrating it to VisualAge TeamConnection.

- The sequence number that CMVC is using for naming defects will be used later on after the migration in order to setup the corresponding sequence number in the TeamConnection family. This will allow a continuous numbering scheme. If the sequence number is not properly updated in the new TeamConnection family then there might be gaps in the numbering of the defects in the new family because may be the canceled defects from CMVC were not migrated and thus, the new TeamConnection family might use their numbers. Do the following to find out the sequence number in CMVC when using DB2:
 - Step 1: Connect to the database: `db2 connect to yourFamily`
 - Step 2: Find out the current number: `db2 "select * from Sequence"`
The output will look like this. The sequence number for defects in this case is 590.

NAME	LASTSERIAL
defect	590
source	435
General	34
- It is important to complete the releases in CMVC because the levels that are in the working and integrate state are not migrated. If you do not want to complete the release, then the levels in the working and integrate state must be manually recreated in TeamConnection. You may use the shell script **release_complete.ksh**.
- If you are cloning the CMVC family, then as the CMVC family administrator you may need to update the database configuration for defects and features:

Migration to VisualAge TeamConnection Version 3

- `chfield -object Defect -source $HOME`
- `chfield -object Feature -source $HOME`
- For a CMVC family 2.3.1, as the CMVC family administrator you must update the current `changeView` view to incorporate additional columns that are needed to migrate files and change history. You can use the following script to assist in this process. It will be available in `$STC_HOME/samples/ksh` starting on fixpak v302.
 - **CmvcUpdateChangeView.ksh** (support for Oracle, Informix, Sybase, DB2)
- Rename any levels that have the same name as a track in releases being migrated, as this is not supported in TeamConnection.

The script **processLevels.ksh** will generate a command file to be executed by the CMVC Administrator. The command file will issue a level rename command where the new name will have an 'L' appended to the original level name.

 - Usage: `processLevels Database`
 - Example: `processLevels DB2`

a) The following report is used by the shell script to determine which Levels are required to be renamed:

```
Report -raw -view levelview -where "name in (select defectName from trackview) order by name, releaseName" | awk -F\| '{ print $1 }' | uniq -c | awk '{ print $2 }' > levels.out
```

b) Execute the generated command file **renameLevelsCMVC.ksh** to update the CMVC database.

c) Review the output file to verify that the update was successful.
- Rename any Defects that have the same name as a Level or a Release being migrated, because this is not supported in TeamConnection.

The script **processDefects.ksh** will generate a command file to be executed by the CMVC Administrator. The command file will issue a defect rename command where the new name will have an 'D' appended to the original defect name.

 - Usage: `processDefects Database`
 - Example: `processDefects DB2`

a) The following report is used by the shell script to determine which Defects are required to be renamed:

```
Report -raw -view TrackView -where "defectName in (select name from releaseView) OR defectName in (select name from levelView)" | awk -F\| '{ print $2 }' > Defects.out
```

b) Execute the generated command file **renameDefectsCMVC.ksh** to update the CMVC database.

c) Review the output file to verify that the update was successful.
- Remove anything from the CMVC family you do not need anymore, such as old host list entries or obsolete entries in the authority list.

Migration to VisualAge TeamConnection Version 3

- You can take advantage of this transition to reorganize the component hierarchy (if needed) or to make changes to the naming conventions, such as changing release names from very long names with upper and lower case letters, to no more than eight lower-case letters.
- There are things that are not migrated but that should be cleaned up before migration. For example, ensure that there are no Tracks in the **working** state and that all files are unlocked.
- The following additional shell script will help you to prepare the CMVC family for migration
 - **release_delete.ksh**

Significant differences between CMVC and VisualAge TeamConnection

- For more information see the technical report **Comparison between CMVC 2.3.1 and VisualAge TeamConnection Enterprise Server 3**.
- You should learn the differences between CMVC and TeamConnection, and make sure the rest of the user community does too. Although there is a lot common between the products, TeamConnection is different than CMVC.
- If you do not understand the differences you will spend a lot of time fighting with the TeamConnection GUI and client commands, and you will not understand why things do not work the way you expected.

Migration to VisualAge TeamConnection Version 3

Prepare the target VisualAge TeamConnection family

Checklist for creating a new family in VisualAge TeamConnection

The following steps are needed to install VisualAge TeamConnection Enterprise Server Version 3 and to create a new family which will be used as the place to store the data migrated from the CMVC family.

Target system: These activities are done in the host system where the new VisualAge TeamConnection family will be located. This system will be called the **target system**.

Description of task	Est. Time	Comment
Install the CMVC client 2.3.1.	½ hr	The migration utility invokes the CMVC client.
If using VisualAge TeamConnection Version 1 or Version 2, then you need to remove it.	1 hr	It is not recommended to have VA TC Version 3 and a previous version in the same system.
Install the VisualAge TeamConnection Enterprise Server Version 3	1 hr	See Installation Guide .
Install DB2 Universal Database (DB2 UDB) Version 5 and create the DB2 instances.	1 hr	See technical report Configuration and Administration of DB2 Universal Database V5 by users of VisualAge TeamConnection Enterprise Server V3
Install the DB2 UDB fixpak and HotFix.	½ hr	From Unix CD-ROM, see: platform/db2/fixes/readme.db2 From OS/2 CD-ROM, see: dbfiles0/fixes/readmeo.db2 From Windows CD-ROM, see: dbfilesw/fixes/readmew.db2
Install the fixpak 3.0.1 (or later) of VisualAge TeamConnection.	1 hr	See README.INSTALL to install the FixPak. Then see FIXPAK301.README to update the family.
Create the family in TeamConnection. Use the sample profile and update it. Add the following variables which are used for the CMVC part of the migration: <ul style="list-style-type: none"> • CMVC_KEYS • CMVC_FAMILY • CMVC_BECOME • CMVC_HOME • Update PATH with SCMVC_HOME/bin 	½ hr	See in this chapter, the section Environment setup for running mkmiglst

Migration to VisualAge TeamConnection Version 3

	Logout and login again in order to have a fresh environment.	5 min	You should avoid simply re-executing the profile because there are some cumulative variables that may cause problems if they are not properly set.
	Start the TeamConnection family: teamcd <i>yourFamily N</i> Where <i>N</i> is the number of daemons	5 min	The TeamConnection family must be operational in order to migrate the data from the CMVC family.
	Ensure that the notification daemon, notifyd , should not be active. If it is active, then terminate it: <ul style="list-style-type: none"> • find out its process id (pid): ps -ef grep notifyd • terminate it: kill -15 <i>pid</i> 	5 min	During migration some notification messages might be generated; because these messages are not needed, the notification daemon should not be running.
	Ensure that the source CMVC family can be reached: Report -testServer You should see a message saying that the family server is active.	5 min	If the Report command fails now, then migcmvc will fail too. It is better to fix the problem now.
	Make a backup of the new target family. It is faster to restore a newly created family from a backup than to recreate it.	½ hr	See the subsection about DB2, in the section How to make a complete backup of your CMVC family of the technical report CMVC Frequently Asked Questions: Migration of Families .

Check for FixPaks and HotFixes

Determine what version of CMVC and TeamConnection you are using, then go to the IBM software FTP site to see if any FixPaks or HotFixes are available that could make the migration go easier.

To find out the version, run the following command in the **\$TC_HOME/bin** directory:

```
what teamcd | grep teamc
```

Important information about installing the code and creating user ids

After you install the VisualAge TeamConnection Enterprise Server code please adhere to the following guidelines:

Migration to VisualAge TeamConnection Version 3

- Do use different directories for the DB2 code, for the VisualAge TeamConnection code, for the DB2 instance and the TeamConnection family.
- Do not alter the contents of **\$TC_HOME** unless you are going to deinstall the product.
- Do not use **\$TC_HOME** as the home directory for any families or users.
- Do not use the same userid for your DB2 database instance and your TeamConnection family.

It is very important that the directories and the user ids must be kept separate. This will avoid potential loss during maintenance activities.

Create family using tcadmin or dbcreate

You must use the **TeamConnection Family Administrator (tcadmin)** tool or **dbcreate** to create a new VisualAge TeamConnection family before migrating. The **dbcreate** utility will delete any exiting family database prior to creating a new one, saving the effort of manually dropping the database.

Environment setup for running mkmiglst

Before running **mkmiglst** it is necessary to setup the following environment variables (shown here in Korn shell format for Unix):

1. Login to the VisualAge TeamConnection family that will be the target for the migration. In this example is the Unix family "tpec3mig".
2. Update the environment in order to use CMVC:
 - Migrate the expanded keywords:
`export CMVC_KEYS=YES`
 - Establish the location of the CMVC code, by default:
`export CMVC_HOME=/usr/lpp/cmvc`
 - Establish the PATH to look for the CMVC code:
`export PATH=$PATH:$CMVC_HOME/bin`
 - Establish the source CMVC family, for example:
`export CMVC_FAMILY=cmpc3db2@cordoba@3402`
 - Establish the CMVC user, for example:
`export CMVC_BECOME=cmpc3db2`

Migration to VisualAge TeamConnection Version 3

Performing the migration

Checklist for migrating the CMVC family to VisualAge TeamConnection

Target system: These activities are done in the host system where the new VisualAge TeamConnection family will be located. This system will be called the **target system**.

	Description of task	Est. Time	Comment
	Disable the automated shell scripts that might affect the availability of the target TeamConnection family or the DB2 instance. This is to prevent a possible interruption of the migration.	½ hr	For example, if you start the migration in the evening, it may take several hours to migrate the parts. Thus, you want to avoid an automatic daily backup cron job at midnight to interrupt the migration.
	Compile <code>mkmiglst</code> from the C source file <code>mkmiglst.c</code>	5 min	See the compilation instructions in the header of <code>mkmiglst.c</code>
	Run mkmiglst to prepare the migration script.	1/2 hr	If needed, obtain sign-off from management or technical leaders for the migration scripts.
	Run migcmvc with the migration scripts by following the migration instructions.	1 - 12 hrs	These files are generated by <code>mkmiglst</code> .
	Run analyzeReorg on new target family.	½ hr	This will improve performance of the new TeamConnection family.
	Run migVerify to verify the migration.	1-4 hrs	Refer to 'Migration shell Scripts' chapter of this document for usage.
	Generate and load the new (*.ld) files for configurable items: <ul style="list-style-type: none"> • authority groups: authorit.ld • notification groups: interest.ld • component processes: comproc.ld • releases processes: relproc.ld • configurable types: config.ld 	½ hr	See section Preparing to administer your new database in Chapter 8. Migrating to TeamConnection version 3 of the Installation Guide , which is located in the CD-ROM in: Intel: nls/doc/enu/install.pdf Unix: softpubs/en_US/install.pdf
	Setup the configurable fields (*.tbl , *.fmt) in \$HOME/cfgField. These configurable fields files are not migrated. Use TCADMIN to add the configurable field data to TeamConnection.	½ hr	See above comment for instructions.
	Setup the user exits in \$HOME/config/userExit	½ hr	See above comment for instructions.

Migration to VisualAge TeamConnection Version 3

Background on mkmiglst

The TeamConnection utility **migcmvc** is used to migrate a CMVC family to TeamConnection. It is a front end to a generalized SQL query and update mechanism. This tool should be used only during migration.

The **migcmvc** utility provides a command line environment from which you can issue the migration commands. Because the order of the commands for migration is very important, it is recommended to use an input file that contains migrate commands that can be piped to the **migcmvc** utility to perform a generic migration. Each of the migrate commands can be customized using SQL syntax to fine tune the selection of data to be migrated.

In order to make migration easier, this technical report describes a tool **mkmiglst** that generates a customized input file for the **migcmvc** utility. This input file is much more useful than the generic **migcmvc.lst** file shipped with TeamConnection.

Note: On OS/2 and Windows NT, **mkmiglst** uses **rm.exe** (to delete files and directories) and **tail.exe** (to show the last lines in a text file) that are provided in **migrate.zip**.

Compiling mkmiglst

The tool **mkmiglst** is provided in a C source code file. You need to compile it for your platform:

- Unix:
 - `unset LIBPATH`
 - `cc -D__UNIX__ mkmiglst.c -o mkmiglst`
- Intel: `icc -D__INTEL__ mkmiglst.c`

Usage statement of mkmiglst

If you invoke **mkmiglst** without arguments, then the usage statement is displayed:

```
mkmiglst usage:
mkmiglst -T|-C|-h|-? -f|-c [-o <outfile>] [-a|-r <release>...]
mkmiglst                - This help message
mkmiglst [-h|-?]        - Extended help
Notes: Parameters must be used in order above,
      Run command from family account home directory.
```

Help statement of mkmiglst

If you invoke **mkmiglst** with the **-h** or **-?** Arguments, then the help text is displayed into different sections:

Description:

```
This utility creates a migration script for use by the
VisualAge TeamConnection Version 3 migration utilities:
- migtc:  Migrate from TeamConnection Version 2
- migcmvc: Migrate from CMVC Version 2.3.1
```

```
The generated migration script is input to these utilities.
Also generated is a file containing instructions for performing
```


Migration to VisualAge TeamConnection Version 3

the migration.

Example: Migrate all data in a TeamConnection V2 family to a TeamConnection V3 family (generate script, list instructions, migrate):

```
$ mkmiglst -T -c -o migtc -a
$ more migtc.txt
$ migtc < migtc1.lst > migtc.out 2>&1
$ migtc < migtc2.lst >> migtc.out 2>&1
```

Example: CMVC 2.3.1.3 to TeamConnection V3 migration, migrating snapshots of selected releases from CMVC 2.3.1.3 family to a TeamConnection V3 family (generate script, list instructions, migrate):

```
$ mkmiglst -C -f -o migcmvc -r v209 -r v210
$ more migcmvc.txt
$ migcmvc < migcmvc1.lst 2>&1 | tee migcmvc1.out
$ migcmvc < migcmvc2.lst 2>&1 | tee migcmvc2.out
```

Example: CMVC 2.3.1.3 to TeamConnection V3 migration, migrating snapshots of selected releases inputted from a file:

```
$ mkmiglst -C -f -o migcmvc < release.list
$ more migcmvc.txt
$ migcmvc < migcmvc1.lst 2>&1 | tee migcmvc1.out
$ migcmvc < migcmvc2.lst 2>&1 | tee migcmvc2.out
```

Details:

Executing `mkmiglst` generates the following outputs (assume "-o XXX" used):

- XXX.txt: Instructions for performing migration, customized based on the parameters used in `mkmiglst`.
- XXX1.lst: First file containing actual migration commands, customized based on the parameters used in `mkmiglst`. Commands in this file migrate all non-part data.
- XXX2.lst: Second file containing actual migration commands, customized based on the parameters used in `mkmiglst`. Commands in this file migrate all files/parts (snapshot of release or all change history).

These scripts do the following:

- Create an extract directory.
- Create a subdirectory for each release.
- Provide a commented entry that will allow for delete of directory after the release is migrated.

Options:

Migrate from CMVC, TeamConnection, or provide help:

- C: To migrate from CMVC Version 2.3.1.3 (upgrade to CMVC 2.3.1.3 required)
- T: To migrate from VisualAge TeamConnection Version 2

-h or -?: For this detailed help

Migrate snapshot of files or complete change history:

- f: Migrate FileView (snapshot of committed release)
- c: Migrate ChangeView (complete history of release)

Output file option:

- o: Specify filename for migration listfile. Default=migrate.lst.

Migrate selected release or all releases:

- r ReleaseName: The release to be migrated. Use -r for each release.
- a: To migrate the entire family (all releases)

Note: If no releases are specified, and -a is not used, then the user is prompted to enter each release.

Notes:

Migration to VisualAge TeamConnection Version 3

1. This tool and associated technical report is a supplement to the VisualAge TeamConnection Version 3 Administrator's Guide. Read the chapter on Migration before using this tool.
2. The output of this tool is determined by the experiences of the VisualAge TeamConnection development, test and services teams. Editing of the output may be desired, but should not be necessary.
3. Required environment variables
For CMVC 2.3.1.3:
To migrate from CMVC 2.3.1:
CMVC_FAMILY, CMVC_USER, CMVC_BECOME, CMVC_KEYS, CMVC_ALLCOMMON
TC_FAMILY, TC_USER, TC_BECOME, TC_DBPATH, HOSTNAME
To migrate from TeamConnection Version 2:
TC_FAMILY, TC_FAMILY_CLIENT, TC_USER, TC_BECOME, HOSTNAME

Example of using mkmiglst to generate the migration script

In this example, the mkmiglst tool was used to provide the migration scripts for the following example in which we wish to migrate the releases **Release1** and **Release2** from the CMVC family.

```
mkmiglst -C -f -o myfile -r Release1 -r Release2
```

Where:

- -C indicates that the migration will be from CMVC.
- -f indicates to migrate the tip of the versions (not all the history of file changes).
- -o indicates the name to be used for the resulting migration scripts.
- -r indicates one release to be migrated.

The following files will be generated by mkmiglst:

- **myfile1.lst**: The first of 2 migration steps.
- **myfile2.lst**: The second of 2 migration steps.
- **myfile.txt**: The migration instructions.

When mkmiglst is executed with the options shown above, these are the runtime messages:

```
Migrating from CMVC Version 2.3.1
Migrating snapshot of committed version of release(s)
Source family: cmvc3db2@cordoba@3402
Target family: tcpc3mig
Migration file 1: myfile1.lst
Migration file 2: myfile2.lst
Instruction file: myfile.txt
Migrating list of releases
Migrating release: Release1
Migrating release: Release2
Total number of releases in list: 2
Writing to root directory: /home/tcpc3mig
Writing first set of instructions for CMVC to VATC3
Writing second set of instructions for CMVC to VATC3
Writing user instructions
Processing complete: Output files generated.
```

Example of migration file 1 generated by mkmiglst: myfile1.lst

The following file is the migration file 1 (myfile1.lst) generated by mkmiglst.

Migration to VisualAge TeamConnection Version 3

```
# Writing first set of instructions for CMVC to VATC3
# If you use release_delete.ksh set maxerrors to 500
# (allowing for duplicate records)
set maxerrors 5000
# This is the recommended batchsize (a smaller number reduces the
# rollback on failure)
set batchsize 100
# Migrating individual tables
migrate Users
migrate HostView
migrate Authority
migrate Interest
migrate Cfgcomproc
migrate Cfgrelproc
migrate Config
migrate CompView where l=1 order by addDate
migrate bCompView where name = 'root'
# Recommend migrating all Release records, due to references
# from other objects
migrate ReleaseView
migrate EnvView
migrate AccessView
migrate NotifyView
# Recommend migrating all defects and features, due to references
# from other objects
migrate DefectView
migrate FeatureView
migrate LevelView where state in ('commit','complete') order by commitDate
migrate TrackView where state in ('commit','complete')
migrate LevelMemberView
migrate FixView
migrate ApproverView
migrate ApprovalView
migrate SizeView
migrate TestView
migrate VerifyView
migrate CoreqView
migrate NoteView
quit
```

Example of migration file 2 generated by mkmiglst: myfile2.lst

The following file is the migration file 2 (myfile2.lst) generated by mkmiglst.

Notice that because some lines are larger than 80 characters, we needed to show them in multiple lines in this document; however, in reality they take only one line in the actual script.

```
# Writing second set of instructions for CMVC to VATC3
# Few warnings should be reported migrating part data
set maxerrors 100
# You can increase decache to increase performance if you do not approach
# your paging space limit during tests
set decache 1
# This is the recommended batchsize (a smaller number reduces
# the rollback on failure)
set batchsize 100
# Migrating File data
# Extract directory must be exported for CMVC to mount
set top /home/tcpc3mig/extract/Release1
! mkdir -p /home/tcpc3mig/extract/Release1
# Extract without expanding keywords
! Release -extract Release1 -committed -nokeys -node $(hostname) \
```

Migration to VisualAge TeamConnection Version 3

```
-root /home/tcpc3mig/extract/Release1 -family cmpc3db2@cordoba@3402
migrate FileView where releaseName = 'Release1'
# rm -rf /home/tcpc3mig/extract/Release1
# Extract directory must be exported for CMVC to mount
set top /home/tcpc3mig/extract/Release2
! mkdir -p /home/tcpc3mig/extract/Release2
# Extract without expanding keywords
! Release -extract Release2 -committed -nokeys -node $(hostname) \
  -root /home/tcpc3mig/extract/Release2 -family cmpc3db2@cordoba@3402
migrate FileView where releaseName = 'Release2'
# rm -rf /home/tcpc3mig/extract/Release2
quit
```

Example of file with migration instructions generated by mkmiglst: myfile.txt

The following file is the file that contains the migration instructions (myfile.txt) generated by mkmiglst.

PRE-MIGRATION RECOMMENDATIONS

1. Upgrade CMVC family to CMVC 2.3.1.n, using latest FixPak and HotFix. Technical reports addressing CMVC and TeamConnection migration issues are available at <http://www.software.ibm.com/ad/teamcon> (select Library).
2. Read the README.TXT file provided with the installation media for any notes on migration.
3. Create a superuser login and a host list entry for the TeamConnection family account in the CMVC family to be migrated.
4. Backup the CMVC family and database.
5. Follow directions for migration in the VisualAge TeamConnection Enterprise Server Version 3 Installation. Use the on-line copy delivered on CD-ROM, as the directions are more current.

MIGRATION COMMANDS

Use the following commands, where the migration instructions call for executing migcmvc:

1. migcmvc < myfile1.out 2>&1 | tee myfile1.out
2. For status: tail -f myfile1.out
 - * tail.exe for OS/2 and Windows NT provided in migrate.zip available from the IBM VisualAge TeamConnection ftp site: <ftp://ftp.boulder.ibm.com/ps/products/teamconnection/tools>
3. Repeat for myfile2.lst
4. Return to migration instructions for the following:
 - Update starting Defect/Feature number in fhcsequence table
 - Performance Tuning
 - Backup of Database

POST-MIGRATION RECOMMENDATIONS

1. Remove authorities from CMVC family to block any further work on the migrated CMVC release.

OPTIONAL ACTIONS FOR CMVC MIGRATION

1. Cleanup the work in progress in the target release(s) using the following shell scripts.
 - release_complete.ksh
 - reassign_work.ksh
 - release_delete.ksh

VERIFICATION ACTIONS FOR CMVC MIGRATION

Migration to VisualAge TeamConnection Version 3

1. Change to directory with myfile2.lst
2. migVerify.ksh -C myfile2.lst
migVerify.ksh will temporarily create \$HOME/cmvc/extract
and \$HOME/TCv300/extract directory structures.
3. Review output file: summary.file.

end of instructions

Migration to VisualAge TeamConnection Version 3

Hints and tips for migrating the data from CMVC

You are likely to perform the migration task several times, and thus, we suggest several tasks to make each migration easier and more reliable:

Maintenance mode

Make sure the CMVC family is in maintenance mode. In that way, the users of the CMVC family will not be able to create or modify objects in the CMVC family, which could cause a conflict during the migration to TeamConnection. To run the CMVC in maintenance mode execute:

```
cmvcd -m $CMVC_FAMILY N
```

Where N is the number of daemons (at least 2).

Notes:

- The migration script generated by `mkmiglst` includes a CMVC Release `-extract`. Currently, this command does not run when CMVC is in maintenance mode. The **Release -extract** command is included mostly for documentation. We suggest performing the extract before running `migcmvc`. However, if there is a way for you to prevent user access to the CMVC family during migration (with CMVC in normal mode), the Release `-extract` can be run from the migration script.
- If CMVC is not in maintenance mode and users have access to the family, then you risk migrating inconsistent data and corrupting your TeamConnection family

Take note of the time that it took to perform the test migrations

Take note of how long it takes you to perform the practice migration so you know for how long CMVC will need to be in maintenance mode.

Qualify appropriate migration steps

All of the migration steps support SQL for limiting the amount of data that can be migrated. However, when data is referenced by other tables and is not migrated, some warning messages will be displayed and they may be numerous and disconcerting. The CMVC objects that are not qualified by `mkmiglst` should be allowed to migrate all data.

If it is critical that the data be limited (that is, cannot migrate all data), we recommend contacting IBM VisualAge TeamConnection Services to assist with the migration.

Redirect output during migration

When running the migration step, we recommend redirecting your output to a file, in that way you can review the output at leisure. The example below redirects both normal output and error messages to the same file, in the correct order:

```
migcmvc < myfile1.lst 2>&1 | tee myfile1.out
```

Migration to VisualAge TeamConnection Version 3

If this is a practice run, we also recommend timing your test:

```
timex migcmvc < myfile1.lst 2>&1 | tee myfile1.out
```

Note for OS/2:

In OS/2 we have seen cases where the migration does not start properly with the commands above; repeating the message “Restarting...”. In this case we recommend the alternative syntax:

```
c:\> migcmvc 2>&1 | tee myfile1.out
```

...

Press <Enter> once

```
cmvcfam> exec myfile1.lst
```

In a separate window run:

```
c:\> tail -f myfile1.out
```

Executing the migration scripts

The actual migration is covered thoroughly in the VisualAge TeamConnection Installation Guide. However, this document adds steps. Therefore, it is helpful to tie the two processes together.

To actually migrate the CMVC family to TeamConnection, follow the instructions provided with the output file generated from **mkmiglst** which has a file extension of **.txt** (see previous sections).

Updating the configuration files

The configuration files should be manually regenerated on the file system following the migration. The migration utility will migrate the configuration data as stored in the database, however the actual files will not be replaced on the file system. Refer to the TeamConnection Installation Guide in the section **Preparing your new Database** for complete details on regenerating these configuration (*.ld) files.

Notice that in TeamConnection, the names of some of the *.ld files have changed from CMVC:

CMVC Name	TeamConnection Name
authority.ld	authorit.ld
cfgrelproc.ld	relproc.ld
cfgcomproc.ld	comproc.ld
config.ld	config.ld
interest.ld	interest.ld

Migration to VisualAge TeamConnection Version 3

Updating the configurable fields

After creating a TeamConnection V3 Family, a subdirectory (cfgField) is created in the Family \$HOME directory. This directory contains *.tbl files for each object that has configurable fields, such as Defect.tbl.

The configurable fields files in \$HOME/cfgField (*.tbl, *.fmt) must be manually updated after the migration. The migration utility DOES NOT migrate these configurable fields files:

- Use TCADMIN to modify them. TCADMIN is completely documented in the TeamConnection Administrator's Guide.
- Refer to the TeamConnection Installation Guide in the section **Preparing your new Database** for complete details on updating the configurable fields files.
- By default, there should be two *.tbl files for Defect and Feature. You will need to create any additional *.tbl files that your current CMVC family may contain, such as for Files and Users.

Updating the user exits

User Exits from CMVC are not migrated to Team Connection. Refer to the TeamConnection Administrator's Guide for details on creating user exits. You should use TCADMIN to generate the required User Exits.

Be aware of the difference in user exits between TeamConnection and CMVC

There are subtle differences in user exit behavior in VisualAge TeamConnection with respect to CMVC. The main differences are:

- The order of parameters have changed.
- The names of some exits have changed.
- There is a new feature that allows for easier handling of parameters using the names of the parameter.

There is a technical report covering TeamConnection user exits and it is available from the TeamConnection home page or ftp site (file **trtcusrx.pdf**):

TR 29.3032 Making the most of VisualAge TeamConnection User Exits

Migration to VisualAge TeamConnection Version 3

Verification of the new TeamConnection family

Checklist for verifying the new family in VisualAge TeamConnection

Target system: These activities are done in the host system where the new VisualAge TeamConnection family will be located. This system will be called the **target system**.

Description of task	Est. Time	Comment
Review output of migcmvc. There might be warnings or errors that you need to be aware of.	½ hr	See section Migration Warnings and Messages at the end of this document.
Verify the migrated data by running the teamc Report command on various objects in the target TeamConnection family. Perform a comparison with the Report command on the source CMVC family.	1 hr	Ensure that the number of reported items by TeamConnection matches the ones reported by CMVC.
Run the migVerify tool.	1-4 hrs	Usage described in the section Migration shell scripts of this document.
Explore some objects in the new target family and verify that they look fine.	½ hr	Manual observation.

After the migration has been performed, it is necessary to verify that the new TeamConnection family has valid data. You can perform the following verifications:

1. First of all, you will need to review the output file from migcmvc, which has a suffix of **.out**, to see if there are any errors or warnings reported by the migration utility. If there are errors, see section **Migration Warnings and Messages** at the end of this document.
2. Run CMVC **Report** and TeamConnection **teamc report** commands to ensure that the number of records match for different objects, such as users, components. You will need to use the **-where** clause to limit the number of records found by CMVC to match TeamConnection (as defined by the *.lst file generated by the tool). You may create your own shell scripts that have pairs of commands, the first item in the pair for CMVC and the second for TeamConnection.
3. Manually explore some objects, such as view a defect, show the host lists for a user, see the history of a file (-view -long) if the history was migrated, etc.
4. Execute the **migVerify** script to verify the following:
 - verify that the correct number of parts were migrated.
 - verify that the correct number of change history versions were migrated.
 - verify that the part content is correct for all parts in the releases migrated.

Migration to VisualAge TeamConnection Version 3

Post-migration tasks for the new TeamConnection family

Checklist for performing post-migration tasks for the new TeamConnection family

Target system: These activities are done in the host system where the new VisualAge TeamConnection family will be located. This system will be called the **target system**.

	Description of task	Est. Time	Comment
	Update the sequence number for defects and features.	5 min	Allow new defects/features to be numbered appropriately
	Update the scripts that used CMVC commands and replace these commands by TeamConnection.	1 hr	Add teamc before the CMVC command. For example: CMVC: Report TeamConnection: teamc Report
	For both the source CMVC system and the target TeamConnection system, update the administration system scripts or cron jobs to include the new family. You may need to enable any cron jobs that were disabled before the migration to avoid an interruption of the migration process.	1 hr	For example, to include the new TeamConnection family with the daily backup scripts.
	Remove files for unnecessary messages: cd \$TC_HOME/queue rm *	5 min	Some messages might have been generated and placed in the queue during migration. These messages should not be sent because they might confuse users.
	Make a complete backup of the family	1 hr	
	Run analyzeReorg in order to update the statistics that will be used by the query optimizer from DB2.	1 hr	Consider scheduling this along with off shift backups.
	Start the notification daemon to send new messages to users: notifyd familyName mailexit	5 min	This ensures that the users will get notification from the TeamConnection family.
	Notify the users that the new TeamConnection family is ready to be used.	5 min	

Migration to VisualAge TeamConnection Version 3

Updating the sequence number for defects and features

The sequence number that CMVC is using for naming defects will be used later on after the migration in order to setup the corresponding sequence number in the TeamConnection family. This will allow a continuous numbering scheme. If the sequence number is not properly updated in the new TeamConnection family then there might be gaps in the numbering of the defects in the new family because may be the canceled defects from CMVC were not migrated and thus, the new TeamConnection family might use their numbers. Do the following to find out the sequence number:

- Connect to the database:
db2 connect to \$TC_FAMILY
- Find out the current number:
db2 "select * from Sequence"

The output will look like this. The sequence number for defects in this example is 253.

NAME	LASTSERIAL
defect	253
source	435
General	34

In our example, the same DB2 commands used on the CMVC family showed a sequence number for defects of 590. Thus, we need to update this sequence number in the TeamConnection family by performing the following:

- Update the value for the defect counter:
db2 "update Sequence set lastSerial=590 where name='defect' "
- Commit the change:
db2 commit
- Verify it:
db2 "select * from Sequence"

The output should look now like this:

NAME	LASTSERIAL
defect	590
source	435
General	34

- Terminate the DB2 session:
db2 terminate
- The next defect that is opened will use the sequence number 591.

Migration to VisualAge TeamConnection Version 3

Our experiences migrating to TeamConnection Version 3

Migration decisions

When we prepared the plan to migrate our original development CMVC family to VisualAge TeamConnection Version 2, and then later on to migrate to VisualAge TeamConnection Enterprise Server Version 3, we had to address the following questions:

Do we migrate all of our releases or just the active ones?

We decided to migrate only the active releases. There was too much data to migrate over a weekend (the period when there would be no development activity) if we wanted to migrate all the releases. We wanted to leave the releases related to VisualAge TeamConnection Version 2 in a Version 2 family. Likewise, we still maintain the CMVC code in a CMVC family.

Do we migrate the change history or just the currently committed version of each file?

When we migrated from CMVC to TeamConnection Version 2, we choose to keep our CMVC family around running in maintenance mode, and only migrate the most recently committed files. This minimized the amount of time we were down for migration and preserved in CMVC the historical information.

We did the same thing when we migrated from TeamConnection Version 2 to Version 3: we choose to keep the Version 2 family around in maintenance mode and only migrate the most recently committed files.

Do we clean up the family before the migration?

We decided to do the following for our appropriate family (running on AIX):

- Run **release_complete.ksh** on the releases to be migrated
- Review output of **release_complete.ksh** carefully, then verified that all workareas and drivers were in complete state.
- Prune workareas and drivers not used to save space so that we could keep the old family around for a while, just in case we needed to migrate something else.

For the migration from Version 2 to Version 3, we did not run **reassign_work.ksh**, since we had migrated our Version 2 family from CMVC only 18 months prior to this new migration. Also, we did not use **release_delete.ksh**, since it is not applicable to TeamConnection.

Results of migration decisions

As a result of the above decisions, here is how our migration from Version 2 to Version 3 went:

- 7000+ files (mostly text source files)
- 2 releases (most recent changes, with no change history)
- Migrated from RS/6000 c20
- Migrated to RS/6000 f30
- It took 23 hours to migrate

Migration to VisualAge TeamConnection Version 3

- 1 hour to migrate base objects
- 4 hours to migrate tip versions
- 18 hours to migrate notes

We did many practice runs, wrote the tools referenced in this document, and tuned what we were migrating along the way, timing the migration to make sure it would fit in a weekend.

Migration to VisualAge TeamConnection Version 3

Migration from CMVC95

This chapter gives an overview on how to migrate from CMVC95 (available only for IBM internal users and similar to CMVC 2.3.1) to VisualAge TeamConnection V3. This chapter is intended for CMVC95 family administrators. The procedures in this document will focus on the AIX platform. These instructions will work for all database platforms except where **DB2 only** is indicated.

The general methodology for migration a CMVC 2.3.1 family is very similar to the migration of a CMVC95 family. This chapter focuses only on the differences:

- VisualAge TeamConnection V3 does not support data driven user exits.
- In the *.ld files for TeamConnection, the new action names cannot be over 15 characters. If the name does exceed 15 characters, you will need to truncate the name:

WorkAreaIntegrate	WorkAreaIntegra
WorkAreaComplete	WorkAreaComple

- * The following interests must be removed from the interest.ld file: DefectTransfer, FeatureTransfer, DefectRecall, WorkareaChange, FamilyCreate, FamilyDelete, WorkAreaComple, FamilyRecreate, FamilyModify, PackageCreate, PackageDelete, PackageExclude, PackageExtract, PackageGet, PackageInclude, PackageModify, PackageRecreate, PackageUpdate, PackageView

AIX Environment Variables

Before running migcmvc, the following environment variables must be set for CMVC95:

Required

- CMVC_FAMILY set to the CMVC95 family being migrated.
- CMVC_HOME set to the CMVC95 client directory on the target machine.
- CMVC_TYPE set to 95.

Optional

- CMVC_BECOME: if set, then **-become** will be added to Report commands.
- CMVC_KEYS: if set to YES, then the SCCS keywords will be migrated to TeamConnection keywords.
- CMVC_ALLCOMMON: if set, then all common file links will be maintained; if not set, then only those file links in binding releases will be maintained.

If all file versions are being migrated and if common file relationships are being maintained in TeamConnection, issue the following command from migcmvc: **migrate commonparts**

Decisions that will greatly increase the migration times

Migration to VisualAge TeamConnection Version 3

- CMVC_ALLCOMMON=YES
- Migrating all file versions.
- Migrating SCCS keywords.

Migrating of CMVC95 Notes

The migration of CMVC95 Notes is done according to the database.

Migrating from Oracle, Informix or Sybase

If migrating from a Oracle, Informix or Sybase database, run the following migrate command:
migrate noteview

The migcmvc migration tool moves the Defect history as a note and the version remarks are not migrated. In most cases, this is not an issue. However, if it is necessary to migrate the version remarks, follow the instructions below for DB2 to download the required tools.

Migrating from DB2

- You will need to get the following migration tools as shown below:
 1. `ftp orbitadm.austin.ibm.com`
 2. `login guest` (Get the password from Bob Pietrasik TL: 678-1952)
 3. `cd /home/guest/migrate/v2.01/ga`
- You will need to download the following set of files to migrate the Notes, Version remarks and History data for DB2: `updt_note_hist`, `update_oid`, `update_oid.bnd`, `update_ver`, `update_ver.bnd`, `export_history.db2`, `export_notes.db2`, `export_versions.db2`
- On the CMVC95 machine export the Notes, Versions, and History data from CMVC95 family. Three separate commands will need to be run to complete the migration:
 - `db2 -f export_notes.db2 -t`
 - `db2 -f export_versions.db2 -t`
 - `db2 -f export_history.db2 -t`

These commands will dump the notes, history and versions tables into the directory where the commands are run.

- Three files will be created: `history.ixf`, `notes.ixf` and `versions.ixf`. Transfer them to the target TeamConnection machine. Because these files can be very large, make sure there is plenty of space in target directory.
- On the TeamConnection Machine run **updt_note_hist** to load the data into TeamConnection.

Migration to VisualAge TeamConnection Version 3

Migration shell scripts

All the tools used here are written in Korn shell and are available via the public Internet or through IBM's intranet. The tools are updated periodically, so we are not providing the source in this document. The tools are zipped into a single file called **migrate.zip**. Because we may update the migrate.zip, we are showing below only some of the tools that are included:

- Unix: **mkmiglst.c**, migVerify.ksh, release_complete.ksh, reassign_work.ksh, release_delete.ksh, file_import.ksh, CmvcUpdateChangeView.ksh, processLevels.ksh, processDefects.ksh
- OS/2: **mkmiglst.c**, find.exe, rm.exe, grep.exe, tail.exe, tee.exe, which.exe
- Windows NT: **mkmiglst.c**, find.exe, grep.exe, rm.exe, tail.exe, tee.exe, which.exe

You can request help for each tool by entering the "-?" parameter, for example:

```
release_delete.ksh -?
```

The shells scripts are listed in alphabetical order.

Updating the CMVC ChangeView (DB2, Oracle, Sybase, Informix): CmvcUpdateChangeView.ksh

Help contents:

```
USAGE: CmvcUpdateChangeView.ksh <DATABASE> <FAMILY>
WHERE: DATABASE      is ORACLE|DB2|SYBASE|INFORMIX
WHERE: FAMILY       is the CMVC family to process
NOTE : script must be run by a SUPERUSER
```

Import a release into CMVC before migration: file_import.ksh

Help contents:

```
USAGE: file_import.ksh -f <FAMILY> -a <RELPATH> -m <MAPFILE> -r <RELEASE>
WHERE: FAMILY       is the CMVC family to process
      RELPATH       is the absolute path to the file tree
      MAPFILE       is the pathname component mapfile
      RELEASE       is the is the active CMVC release
NOTE : script must be run by a SUPERUSER
      MAPFILE format pathname|compname
      Assuming empty RELEASE
```

This utility will load files into a CMVC release from a given root directory. It requires that the user should provide a mapfile that maps the pathnames of the files to the components where the files are to be loaded into.

Automated verification tool: migVerify.ksh

Migration to VisualAge TeamConnection Version 3

Help contents:

```
USAGE: migVerify.ksh -C|-T MIGRATION_SCRIPT
WHERE: -C   Migration from CMVC to TC
        -T   Migration from TC - TC
        MIGRATION_SCRIPT is the name of the migration script
        that was used to issue the "migrate FileView" or the
        "migrate PartView" commands.
NOTE : script must be run by a SUPERUSER
```

This shell script helps you to verify that the correct number of parts were migrated, that the correct number of change history versions were migrated and that the part content is correct for all parts in the releases migrated.

Renaming Defects That have the same name as existing Releases in CMVC: processDefects.ksh

Help contents:

```
USAGE: processDefects.ksh <DATABASE>
        Where DATABASE is: ORACLE|DB2|INFORMIX|SYBASE
NOTE : script must be run by a SUPERUSER
```

Renaming Levels That have the same name as existing Tracks in CMVC: processLevels.ksh

Help contents:

```
USAGE: processLevels.ksh
        Where DATABASE is: ORACLE|DB2|INFORMIX|SYBASE
NOTE : script must be run by a SUPERUSER
```

Reassign user's work and delete users: reassign_work.ksh

Help contents:

```
USAGE: reassign_work.ksh -f <FAMILY> -s <FROMUSER> -t <TOUSER>
WHERE: FAMILY      is the CMVC family to process
        FROMUSER   is the CMVC login to be deleted
        TOUSER     is the CMVC login to assign work to
NOTE : script must be run by a SUPERUSER
```

This utility will reassign incomplete work from one active CMVC login to another active CMVC login.

Incomplete work can be defined as belonging to the following set: defects owned/originated, features owned/originated, verification, components, releases, levels, environments, test records, size records, approval, approver, tracks, fix records, notifications deleted, access deleted, login deleted

Migration to VisualAge TeamConnection Version 3

Completing a release in CMVC before migration: release_complete.ksh

Help contents:

```
USAGE: release_complete.ksh -f <FAMILY> -r <RELEASE>
WHERE: FAMILY      is the CMVC family to process
        RELEASE    is the is the active CMVC release
NOTE : script must be run by a SUPERUSER
```

This utility is used to commit and complete all Tracks and Drivers in the specified release prior to migration.

Delete a release from a CMVC family: release_delete.ksh

Help contents:

```
USAGE: release_delete.ksh -f <FAMILY> -r <RELEASENAME> -d <DEFECT> -w
WHERE: FAMILY      is the CMVC family to process
        RELEASENAME is the active CMVC release
        DEFECT      is the CMVC defect for the track
NOTE : script must be run by the family account SUPERUSER
        w option define what sccs and mapfiles can be deleted
```

This utility will delete work in a given release. It will also identify a list of vc (SCCS/PVCS) files that are not related to any other release that can be removed from the file system to save disk space.

Migration to VisualAge TeamConnection Version 3

Warning and Error Messages

These are some warning and error messages from the migcmvc:

- **migrate users:** You may get the following messages because the users are already in the database. They were added when the database was created.

0010-098 The login name InheritedAccess already exists. Specify a different login name.

0010-098 The login name migrate already exists. Specify a different login name.

- **migrate HostView:** The host list entry for the initial superuser was created when the database was created.

0010-055 migrate is already a host list member for user
zorin.raleigh.ibm.com.

- **migrate Authority:** You will get MANY of the following messages because the authority tables were loaded when the database was created

6026-009 *** authority general, "CompView", already loaded.

6026-009 *** authority general, "DefectView", already loaded.

- **migrate Interest:** You will get MANY of the following messages because the Interest tables were loaded when the database was created.

6026-009 *** interest high, "VerifyAssign", already loaded.

6026-009 *** interest high, "VerifyReject", already loaded.

- **migrate Cfgcomproc:** You will get MANY of the following messages because the tables were loaded when the database was created.

6026-009 *** comProc preship, "dsrDefect", already loaded.

6026-009 *** comProc default, "dsrFeature", already loaded.

- **migrate Cfgrelproc:** You will get MANY of the following messages because the tables were loaded when the database was created.

6026-009 *** relProc preship, "approval", already loaded.

6026-009 *** relProc track_approval, "approval", already loaded.

Migration to VisualAge TeamConnection Version 3

Obtaining the Tools

The tools described in this technical report can be downloaded as follows:

- From the IBM intranet (only for IBM employees).
- From the Internet (open to everyone).

Notes:

- The tools referenced in this document are available from ftp sites instead of being included in this document so that we can update the tools as necessary.
- VisualAge TeamConnection Enterprise Server Version 3 is available to IBM users through our internal web site.
- The fixpaks for VisualAge TeamConnection are available from the external home page:
<http://www.software.ibm.com/ad/teamcon/downloads>

IBM Intranet

Web Home Page for TeamConnection

You can access the VisualAge TeamConnection Home Page at:

<http://tc-cmvc.raleigh.ibm.com/>

From the index at the top of the page, select **Tools to migrate from CMVC to TeamConnection**.

FTP site for TeamConnection

You can download the code from our internal FTP site (an OS/2 host) for TeamConnection, by doing:

1. ftp tc-cmvc.raleigh.ibm.com
2. login as **anonymous** and for password give your email address.
3. cd papers
4. binary
5. get migrate.zip # You can get also PDF files
6. quit

Public Internet

Web Home Page for TeamConnection

You can access the VisualAge TeamConnection Home Page at:

<http://www.software.ibm.com/ad/teamcon>

FTP site for TeamConnection

You can download the code from our external FTP site for TeamConnection, by doing:

1. ftp ftp.software.ibm.com
2. login as **anonymous** and for password give your email address.
3. cd ps/products/teamconnection/papers
4. binary
5. get migrate.zip # You can get also PDF files
6. quit

Migration to VisualAge TeamConnection Version 3

Obtaining Info-ZIP

The VisualAge TeamConnection team uses the Info-Zip **zip** and **unzip** tools to package compressed files (in which the files to be packaged are compressed first). The migration tools mentioned in this technical report are packaged into a single zip file.

The main advantages of Info-ZIP are:

Compatibility: these tools are compatible with other ZIP programs.

Portability: they are available in ALL the platforms that are supported by VisualAge TeamConnection.

Cross-platform: A zip file prepared in Unix can be unzipped in the correct format in Windows NT and vice versa.

Info-ZIP's software is free and can be obtained for the desired platforms from various anonymous ftp sites, including the URL:

`ftp://ftp.uu.net:/pub/archiving/zip/`

How to unzip files

- To only view the contents of the zip file (without actually unpackaging and uncompressing the files) do: **unzip -l migrate.zip**
- To unpackage and uncompress the zip file do: **unzip migrate.zip**

Migration to VisualAge TeamConnection Version 3

Copyrights, Trademarks and Service marks

The following terms used in this technical report, are trademarks or service marks of the indicated companies:

TRADEMARK, REGISTERED TRADEMARK, OR SERVICE MARK	COMPANY
AIX, IBM, CMVC, VisualAge, TeamConnection, RS/6000, DB2 Universal Database, OS/2	IBM Corporation
Unix	Unix System Laboratories, Inc.
Acrobat	Adobe Systems Incorporated
Windows, Windows NT	Microsoft Corporation
Intel	Intel Corporation
Info-ZIP	Info-ZIP Group
NFS	Sun Microsystems Inc.

*** END OF DOCUMENT ***