

Runbook for CI/CD with Azure DevOps for Cora SeQuence projects

For Cora SeQuence V9.x

December 2021



Contents

What is CI/CD with Azure DevOps?4
Want to learn more?5
Application life cycle: from the developer's machine to the production environment5
The CI/CD process at a very high level7
CI/CD for Cora SeQuence7
Setup prerequisites
Update workflows or customize applications9
Update workflows10
Executing the pipelines17
Commit to Azure DevOps18
Deployment methods21
Best practices22
Execute the Unit Package pipeline23
Execute the Processes Release pipeline26
Customize Cora SeQuence



Notice

All rights reserved. No part of this document (including the text, images, graphics or the selection or arrangement of content, which forms an original compilation), may be reproduced or transmitted in any material form or by any means, electronic or mechanical, including photocopying or recording in any medium (whether or not transiently) without the written permission of the copyright holder. Such written permission must also be obtained before any part of this publication is stored in a retrieval system of any nature. The same shall apply to the export of this publication from India, and a violation of this condition will lead to civil and criminal prosecution.



What is CI/CD with Azure DevOps?

CI/CD with Azure DevOps is an automated platform on the Azure cloud that enables you to continuously integrate developed applications (projects) with the Azure cloud and continuously deploy the developed application to any Cora SeQuence environment like UAT or production (Live) seamlessly via pipelines. After all the pipelines are created and set up properly, the whole process can run automatically without any manual intervention.

Continuous Integration (CI) is the practice used by development teams to automate the merging and testing of code. Implementing CI helps to catch bugs early in the development cycle, which makes it less expensive to fix. Automated tests run as part of the CI process to ensure quality. Artifacts are produced from CI systems and fed to release processes to drive frequent deployments. The Build service in the Azure DevOps Server helps you set up and manage CI for your applications.

Continuous Delivery (CD) is a process by which code is built, tested, and deployed to one or more test and production environments. Deploying and testing in multiple environments drives quality. CI systems produce the deployable artifacts including infrastructure and apps. Automated release processes, CD processes, consume these artifacts to release new versions and fixes to existing systems. Monitoring and alerting systems run continually to drive visibility into the entire CD process. The Release service in the Azure DevOps Server helps you set up and manage CD for your applications.

To configure CI and CD, you create pipeline definitions. A pipeline definition is a representation of the automation process that you want to run to build and test your application. The automation process is defined as a collection of tasks. Azure DevOps pipelines have a few inbuilt tasks to help you build and test your application. You can further customize your pipelines by adding your own command lines, PowerShell commands, or Shell scripts in your automation.



Want to learn more?

What is CI/CD?

What is Azure DevOps?

CI/CD with Azure DevOps

Application life cycle: from the developer's machine to the production environment

- 1. The developer (with permissions) creates a project in his/her local machine.
- 2. The CI/CD System Admin creates a repository (repos) in the local machine using GIT commands (Source control system).

A repository, also called repos for short, is a folder with version control tools that you can use to manage your code.

Version control systems are software that help you track changes you make in your code over time. As you edit your code, you tell the version control system to take a snapshot of your files. The version control system saves that snapshot permanently so you can recall it later if you need it.

Use version control to save your work and coordinate code changes across your team.

3. The developer creates an Azure repo in Azure DevOps (cloud server).

Note that the Azure repo in Azure DevOps is the implementation of GIT in the Azure cloud.

4. The developer then creates a CI pipeline in the Azure portal to push the project from the developer's machine repo into the Azure DevOps repos. The result is an artifact that is stored in the Azure cloud.



The pipeline contains build tasks that are composed of tasks like getting the source code, building the solution, running tests, packaging artifacts, and finally publishing the artifacts.

A pipeline is similar to a script with a list of command tasks that are executed in sequence. They could include commands such copy file, delete file, execute a PowerShell command, and others. The artifact is the packaged software that results after the pipeline completely executes all its commands and tasks.

- 5. The CI pipeline (that is, the build job and tasks) is run by an agent machine. Azure usually provides a VM (Microsoft hosted machine) that is used as the agent machine. However, the developer can use his/her own hosted machine as the agent machine.
- The developer creates a CD pipeline (consisting of a list of tasks) that is used for deploying and testing the artifact into multiple environments, like Dev, QA, or Production.

Note that the CD pipeline is also run by the agent machine.

7. The developer can add approvals at certain stages so that projects can be approved by QA. For example, before the project gets deployed into production.

The CI/CD process can be fully or partially automated.

In a fully automated CI/CD process, there's no user intervention. As soon as the user pushes the source code in the GIT repo of the local machine, the whole CD/CD process gets executed and all the environments (UAT, Production, and others) are updated as per the configured CI/CD process.

In a partially automated CI/CD process, some of the processes, like copying the files to the Azure repo, are done manually. Or the pipelines are triggered manually by the user.



The CI/CD process at a very high level

Deploying an application to different environments using the CI/CD process.



CI/CD for Cora SeQuence

In order to set up the pipelines, the **DevOps System Admin** needs relevant project details, such as server name, database username and password, Administration folder name, and SeQuence Administration URLs.

These details are usually provided by the **Tech Lead** or **Project Manager** who created the Cora SeQuence project on the Azure DevOps platform.



Setup prerequisites

The Tech Lead or Project Manager needs to set up the following components:

- **Servers**: The DEV, TEST, and PROD servers need to be setup.
- **Users**: The Tech Lead or Project Manager needs to create users in the Azure DevOps platform for whoever needs to access the Azure portal. Because all users need access to the Repos section, they need to have at least the basic license (not the Stakeholders license).
- **Network**: There needs to be physical network connectivity between the servers and the Azure DevOps application. Several URLs need to be whitelisted to ensure a seamless experience with Azure DevOps.

For more information, see this article: <u>Allowed IP addresses and domain URLs</u>

• **Additional tasks**: The Tech Lead or Project Manager needs to add the following tasks to the Azure DevOps platform through the Marketplace:

•	XDT Transform	Add tasks 🛛 💍 Refresh			
		XDT Transform Apply XDT transforms on XML files	Add		
		by Guillaume Rouchon	① Learn more		
•	Replace Tokens	Add tasks 🛛 💍 Refresh			
		#()# Replace Tokens var-I Replace tokens in files	Add		
		by Guillaume Rouchon	① Learn more		



After the project is configured, the **DevOps System Admin** can set up the pipelines in the Azure DevOps platform. The setup includes installing the CI/CD tool on each server and making the **Commit to Azure DevOps** button available in the Administration site.

G Schance Eitan Test	Ø Ø
C 🕐 🖷 🗰 🔏 🐾 + 🛤 📼 🖻 🖉 🧭 Validate 🏦 Delete Workflow 🔗 Export + 🗞 Set Permissions 🔂 Source Control + 🕯	🐞 Process Lab 🕞 Commit to Azure DevOps
Toolbox List View C Find an Activity_ Q	Properties >
Indian Activity Q A KrY Assign Human Suilt-in Command Server Side Empty Server Side Enort Handler Server Side Compty Server Side Enort Handler Server Side Compty Server Side Co	Workflow (Name) Eitan Test Alas Eitan Test Description Eitan Test Workflow ID 7649ac37.6586.4700- 98d8-0da3d9120836 Misc Image Set Statistics Image Set Variables Image Set

Update workflows or customize applications

You use CI/CD pipelines for two main purposes:

- Update workflows or workflow-related applications.
- Customize Cora SeQuence features.

The following diagram provides a very high-level overview of the CI/CD process for Cora SeQuence projects and applications.





Note

The **Golden environment** is where the artifact (software package) is stored after running the CI pipelines. The artifact is used by the CD pipeline to propagate the software to other environments.

Update workflows

You can use two pipeplines to deploy workflows and workflow elements on different environments.

- **Unit Package (CI Pipeline)**: packages workflows that have been committed to Azure and creates an artifact in the project's repo.
- **Processes Release (CD Pipeline)**: takes the artifact from the Azure repo and deploys it to the target environment as specified in the pipeline

We are going to use an **example** to demonstrate how to use the two pipelines.

Assume you already have two Cora SeQuence environments set up:

• Environment 1: CDEVCICD1 (Dev server)

Sequence				
 Administration Analytics 	CICD Test Dashboard			
Analytics Activing Global Settings Copulations Copulations Portal Settings Portal Settings Security S	Number of Instances per Month 0.15 0.22 0.29 0.26 0.20 0.26 0.23 0 0 0 0 0 0 0.25 0.26 0.20 0.20 0.21 0.22 0.23 0 0 0 0 0.27 0.28 0.29 0.20 0.21 0.22 0.23 0.24 0.25 0.26 0.27 0.28 0.29 0.20 0.20 0.20 0.20 0.20 0.20 0.20 0.21 0.22 0.23 0.24 0.25 0.25 0.26 0.27 0.26		Active Version: CLO Test ID: b65b1ea0-066-49de-baf2-cdfe7b43e201 Created At: 21/1/021 Updated By: Eitan Firuzi → Handle Burning Processes → Set Durline Permissions → Set Design Time Permissions Workflow Versions	
CICD Test	Average Workflow Duration (Days)	Open Instances by Interval	Name	
Grappy test Grappy te	0	No data to digilay.	CICD Test	

• Environment 2: CDEVCICD2 (Test server)

G cora SeQuence			🔘 Eitan 🗸
Administration Analytics Archiving Global Settings Lookup Tables Organization Settings Portal Settings Security	Manage Applications Create a New Workflow Import Workflow	Getting Started Control Getting Started Tutorial Building a Workflow Human Activities Defining Form Controls	Control & Monitor Handle Running Processes Open Process Lab
 ♥ Solutions ♣ Workflows All Workflows 	Active Directory Settings Edit Organization Manage Employees Last ASSS Syne: Aug 11, 2021. 01:04 PM Status: Completed	Watch Videos Image: Creating Basic Applications Advanced Applications Creating Dashboards Administration Intro Image: Creating Dashboards	Installation Details Server Name: cdevictd2-admin.demo, Assembly Version: 9.7.1.65 Database Name: CDEVCICD2 Database Version: 9.7.1.24 License Expiration Date: NA Number of Active Users: 122

Let's have a look at the pipelines in the Azure DevOps Portal.

Azure DevOps		
P PNMsoft-TFS	PNMsoft-TFS	
and and a second second	Projects My work items My pull requests	√ Filter projects
New organization	ET Eitan Test D DevOps	• • • • • • • • • • • • • • • • • • •
	AW	

	CT Cloud DevOps IT Team	
Organization settings		

We have an Organization, PNMsoft-TFS, and under that Organization, we have the project Eitan Test.

Azure DevOps) Azure DevOps PNMsoft-TFS / Eitan Test / Pipelines / Releases 🖉 Search 🗉 🗇 🙈 街				
🗉 Eitan Test +	③ Connection with live updates has been term	ninated. Please refresh the page to see the updated view	×		
Overview	\wp Search all pipelines	Unit Package	🖉 Edit 🔗 Create release		
Boards	<u>⊨</u> ⊡ ⊕ + New ∨	Releases Deployments Analytics	E All releases ∨ ∇ :		
😢 Repos	Test Stage 1	Releases Created Stages			
Pipelines	Processes Release	EF 1 7/14/2021, 4:37:01 PM O Dev			
uta Pipelines	Pass QA				
Environments	Stand-alone DevOps Integration				
🖉 Releases	• or				
🕅 Library	Unit Package				
🐨 Task groups					
T Deployment groups					
👗 Test Plans					
Artifacts					

Under the Eitan Test project, there are a few pipelines, including Unit Package (CI Pipeline) and Processes Release (CD Pipeline).

The Unit Package pipeline

Variables available in the Unit Package (CI) pipeline.

Azure DevOps	PNMsoft-TFS / Eitan Test / Pipelines /	Releases / Unit Package		P Search I≣ C	0 % EF
Eitan Test +	All pipelines > 🚏 Unit Package			🗟 Save 💋 Create release 🔳 View rel	eases ···
Overview	Pipeline Tasks Variables Retention	Options History			
E Boards	Pipeline variables	√ Filter by keywords		Scope 🗸 🖂	i≡ List III Grid
😢 Repos	Variable groups Predefined variables @	Name		Value	A Scope
Pipelines		ApprovedWorkItems			Release
🖬 Pipelines		AzureDevOpsIntegration.exe		C:\SequenceDeploymentClient\DevOpsIntegrationApp\AzureDevOpsIn	Release
Environments		DeploymentPackageskepoName		Deployment Packages	Release
🔊 Releases		name		\$(Release.ReleaseName)	Release
Library		ProjectName		\$(System.TeamProject)	Release
🐨 Task groups		Release.Number		1.0	Release
Peployment groups		rootFolderName	Î	Sequence	A Release
👗 Test Plans		1.44			
Artifacts		+ Add			
Project settings		4			

We use the variables to specify the repos of the source project and destination repos of the artifact and other properties needed for the CI process. This configuration is currently done by the PS CI/CD System Admin.

Graphic view of the Unit Package pipeline

Azure DevOps	PNMsoft-TFS / Eltan Test / Pipelines / Releases / Unit Package 🗇 Search 💷 🗇 🗞 😝					
Eitan Test +	All pipelines > 🊏 Unit	All pipelines > 🏋 Unit Package 🗟 Save 💋 Create release 🗮 View releases 🚥				
Overview	Pipeline Tasks - Variables Retention Options History					
🕄 Boards						
😢 Repos	Artifacts + Add	Stages $+$ Add \vee				
Pipelines						
utu Pipelines	+ Add an artifact	A Dev A				
Environments						
🔊 Releases	Schedule					
M Library	not set					
Task groups						
Deployment groups						
👗 Test Plans						
Artifacts						

Unit Package pipeline tasks

Azure DevOps	PNMsoft-TFS / Eitan Test / Pipelines / Releases / Unit Package	🔎 Search 🔠 😨 🙈 🗗
Eitan Test +	All pipelines > 🚏 Unit Package Pipeline Tasks v Variables Retention Options History	🗟 Save 💋 Create release 🔳 View releases \cdots
Boards	Dev	Deployment group job 🕥 🗎 Remove
😢 Repos	Deployment group job	Display name *
Pipelines	Build Command line	Deployment group job
Environments	Add release to package in Github repo	Deployment group * ①
A Releases		Eitan Test-YuvalTest-CI CD Demo I 🗸 🗸 🔞
Task groups		Required tags 🕥
PP Deployment groups		1 matching targets in Eitan Test-YuvalTest-CI CD Demo deployment group
👗 Test Plans		Targets to deploy to in parallel
Artifacts		Multiple One target at a time Maximum number of targets in parallel 100% targets (1) Timeout *
Project settings		0 Job cancel timeout * O

The Processes Release pipeline

Graphic view of the Process Release (CD) pipeline. There are two stages in the pipeline: QA and Pass QA.

Azure DevOps	PNMsoft-TFS / Eitan Test / Pipelines / Releases / Processes Release	III 🗇 🔗 🗗
Eitan Test +	All pipelines > 🕆 Processes Release	🖉 Create release 🛛 🗮 View releases 🛛 …
Overview	Pipeline Tasks v Variables Retention Options History	
🕄 Boards		
Repos	Artifacts + Add Stages + Add \vee	
Pipelines	<i>(b</i>)	
Pipelines	♦ <u>B</u> QA <u>B</u> Pass QA <u>A</u>	
Environments	Bijobs, 11 tasks 2 1 job. 0 task	
SP Releases	Schedule	
Task groups	O not set	
T Deployment groups		
👗 Test Plans		
Artifacts		
Project settings <		

Variables available in the Process Release (CD) pipeline.

Azure DevOps	PNMsoft-TFS / Eitan Test / Pipelines /	Releases / Processes Release	🔎 Search i≡ 🗇	⊘ % E F
🗉 Eitan Test +	All pipelines > ** Processes Rel	ease	🗟 Save 💋 Create release 🗮 View re	eleases ···
Overview	Pipeline Tasks V Variables Retention	Options History		
n Boards	Pipeline variables	√ Filter by keywords	Scope \checkmark \times	i≡ List III Grid
P Repos	Variable groups Predefined variables @	Name	Value	A Scope
Pipelines		AzureDevOpsIntegration.exe	C:\SequenceDeploymentClient\DevOpsIntegrationApp\AzureDevOpsIn	🔒 Release
utu Pipelines		DeploymentFilesBasePath DropFolder	C:\SequenceDeploymentClient	Release
Environments		ExternalResources	C:\inetpub\www.root\Cora SeQuence\Administration\Shared Resources	Release
🔊 Releases		MultiWorkflowsDeploymentApp.exe	C:\SequenceDeploymentClient\MultiWorkflowsDeployment\MultiWork	Release
IIN Library		Release.Number	1.0	Release
📟 Task groups		ShouldCleanPackage	False	Release
T Deployment groups		ShouldStopStartServices	True	Release
👗 Test Plans		WorkingFolder	C:\SequenceDeploymentClient\WorkingFolder	Release
Artifacts		+ Add		
Project settings		a l		•

These variables specify the artifact repos and the destination repos where the applications will be deployed, and additional properties needed for the CD process.

Process Release (CD) pipeline tasks

Azure DevOps	PNMsoft-TFS / Eitan Test / Pipelines / Releases / Processes Release	🔎 Search III 🗇 🗞 😝
Eitan Test -	+ All pipelines > * Processes Release	🗟 Save 💋 Create release 🗮 View releases 🛛 …
Overview	Pipeline Tasks - Variables Retention Options History	
n Boards	QA	Deployment group job ①
😢 Repos	Deployment group job to stop services +	Display name *
Pipelines	Task group: DowerShell Stop RDS	Deployment group job to stop services
Pipelines	Powershell stop Service	Deployment targets A
Environments	Task group: PowerShell Stop JES PowerShell Stop Service	Deployment group *
🖉 Releases	Task group: PowerShell Stop ADSS	Eitan Test-YuvalTest-CD CD Demo (QA 🗸 🗸 🍥
III∿ Library		Required tags 🕠
🐨 Task groups	Prepare Package + A Run on deployment group	
Deployment groups	Task group: _Prepare Release	1 matching targets in Eitan Test-YuvalTest-CD CD Demo QA deployment group
👗 Test Plans	Prepare Release	Targets to deploy to in parallel
Artifacts	Deployment group job for Sequence +	Multiple One target at a time Timeout *
	Task group: Move Deploy package to Deployed packag Move Deploy package to Deployed packages folder	0
	Task group: Deploy Sequence Components	Job cancel timeout *
Project settings <	Components	

This is the repo of the Azure DevOps where the CI pipeline stores the project after processing. That is, the Deployment packages folder.

Azure DevOps	PNMsoft-TFS / Eltan Test / Repos / Files / 🔶 Deployment Pack	ages 🗸	₽ Search	III 🗂 💿 🔗 🗗
🗉 Eitan Test +	Deployment Packages	\mathfrak{P} master \vee \square / Type to find a file or folder		
Overview	> 🖿 Deployment Packages	Files		Succeeded 🖵 Clone
Boards		Contents History		2
P Repos		Name Î	Last change	Commits
Files		Deployment Packages	Jul 14	0f41a7d6 Auto commit by CIC
¢ Commits				
ድ Pushes				
ያያ Branches				
Tags				
82 Pull requests				
Pipelines				
👗 Test Plans				
Artifacts				
ស៊ាំ Project catting				

The Artifact source points to the repo configured in the previous screenshot. That is, the Deployment Packages folder.

Azure DevOps	PNMsoft-TFS / Eltan Test / Pipelines / Releases / Processes Release	
Eitan Test +	All pipelines > 👎 Processes Release	🗟 Save 💋 Create release 🗮 View releases 🛛 …
Overview	Pipeline Tasks >> Variables Retention Options History	
🕄 Boards		Artifact III Delete ····
😢 Repos	$ Artifacts + Add \qquad Stages + Add \lor$	GitDropFolder
Pipelines		Project * 🔘
Pipelines	Pass (Eitan Test
L Environments	DropFolder	Source (repository) * ①
🔊 Releases	C anti	Deployment Packages
IIN Library	C Schedule not set	Default branch * ①
Task groups		master 🗸
T Deployment groups		Default version * ()
Lest Plans		Latest from the default branch \sim
Artifacts		Checkout submodules
		Checkout files from LFS ①
		Shallow fetch depth ①
Project settings	<	Source alias * (i)

Executing the pipelines

In this example, we're going to change the Development server, CDEVCICD1, and then propagate the change to the Test server, CDEVCICD2.

Development server CDEVCICD1

G co Seq	uence						Eitan Test				
0		°. °		0 0	⊗ Validate	Delete Workflow	🖻 Export 🔹	Set Permissions	Source Control 👻	Ko Process Lab	Commit to Azure DevOps
Find an Act Find an Act Human	x List View very x ⋅ Y Assign Built-in Comm Empty A Error Handler S Set Stage ⊕ S Storp Sharing S S Stored Proceec S Stored Proceec S Stored Proceec S Stored Proceec	< Q nand			Test DataModel	La Test F La Test F La Test F La Anot	orm 2 tole Form				

Test server CDEVCICD2

G cord SeQuer	Cl nce				Eitan Test						\$ \$
0	1 1 × 5 5	 50	🖉 🧭 Validate	Delete Workflow	🖻 Export +	名 Set Permissions	Source Control 👻	n Process Lab	Comr	nit to Azure DevC	ps
Toolbox	List View 🗸									Properties	>
Find an Activity Human Server Side	y- Q X+Y Assign Sign Built-in Command Empty Common Handler Sold Stage Sold Stage Sold Sharing Sold Stored Procedure Sold Stored Procedure Sold Sub Workflow		Start	En Test F	orm 2 Itale Form					Workflow (Name) Alias Description Workflow ID Misc Display Field Stage Set Statistics Variables	Eitan Test Eitan Test Eitan Test 76494c27.5586.4700- 98d8-0da5d9120836

You need to connect the change to a work item on Azure DevOps. The work item includes detailed instructions about the change and it can also include related documentation. In this example, we use the work item 76095.

Azure DevOps	PNMsoft-TFS / Eltan Test / Boards / Work items	₽ Search	III 🗇 🔗 📭			
ET Eitan Test +	Recently updated 5 Back to Work Items		3 of 10 🔨 🤟			
Overview	B TASK 76095*					
💐 Boards	76095 Eitan Test					
🛱 Work items	C Unassigned C U comments Add tag		Follow Save			
🕅 Boards	State To Do Area Eitan Test		Updated by Yuval Arav: Feb 8			
Backlogs	Reason A New task Heration Eitan Test	Remaining Work	Details () & () Status reporting for locatus in your pipeline's Options menu. Learn more about deployment			
Ď, Sprints		Activity	status reporting			
= Queries	Click to add Definition Of Done Evidence		Development			
Delivery Plans	Discussion	Blocked	+ Add link			
Plans	Add a comment. Use # to link a work item, ! to link a pull request, or @ to mention a person.	Effort	Link a GitHub commit or pull request to see the status of your development. Learn more			
Portfolio plans (Beta)		Actual	Link an Azure Repos commit, pull request or branch to see the status of your development.			
🛅 Calendar		Artefacts	You can also create a branch to get started.			
😰 Repos		Click to add Artefacts	Related Work			
Pinelines			+ Add link ~			
Na ripelines		OneClick actions	Add an existing work item as a parent			
📥 Test Plans		00				
Project settings		No rules found				

Note

If a work item doesn't exist, you can create it in Azure DevOps.

Commit to Azure DevOps

In the following example, we've added a Form activity called **CI/CD Test** to the workflow Eitan Test in the Development server.

After adding the new form, check in the workflow.

	ro uence			0	Eitan Test					
G	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1							• ×	cess Lab	🕞 Commit to Azure DevOps
Toolbo Find an Acti	x List View <	Source Control I	Management Wiza	ard						
ß	🚑 Dynamic Task	Check In	💱 Check Out	Undo Pending Ch	anges					
Human	E Form	📉 Activity Name	Туре	Status	Checked Out To	Last Updated	History			
0	0	Vorkflow		Checked Out	CICD User	8/11/2021 1:17 PM				
Server Side	∠ Message	Another Form	Form	Checked In		2/4/2021 10:05 AM				
9	Qe Task	CI CD Test	Form	Checked Out	CICD User	8/11/2021 1:18 PM				
		Emp Role Form	Form	Checked In		2/4/2021 9:59 AM				
BPMN		End	End	Checked In		1/28/2021 4:10 PM				
		Start	Start Event	Checked In		1/28/2021 4:10 PM				
		Test DataModel	Data Model	Checked In		2/4/2021 9:58 AM				
		Test Form	Form	Checked In		2/2/2021 11:09 AM				
		Test Form 2	Form	Checked In		2/3/2021 12:17 PM				
							Close			

After check-in, commit the workflow to Azure DevOps. Click the **Commit to Azure DevOps** button.

	ra Jence					Eitan Test					$\langle \hat{Q} \rangle$
Toolbo Find an Activ Human	 List View Dynamic Task Form Message Task 	< < < < < < < < < < < < < < < < < < <		C C Validate	Delete Workflow	Form 2	Set Permissions 🖞 So	urce Control +	¥C Process Lab	Commit to Azure DevOps	re DevOps.
					고 Anot	her Form					

After you click **Commit to Azure DevOps**, the following screen displays:

🔓 çora	O Eit	tan Test					\$ (?
Related Work Items							
d Work Item ID 76095 Add Work Item							
Work Item							
Delete 76095 - Eitan Test							
Comment							
							//
Check In Workflow				Deploy	ment Method		
Eitan Test				Standar	d		~
Check in External Files		Check in SQL Scripts					
Include File Update Da	te	Connection	DB	Schema	Name	Туре	
No Files			CDEVCICD1	dbo	GetStorageTime	UserDefinedFunction	
			CDEVCICD1	dbo	USPEXAMPLE	StoredProcedure	
			CDEVCICD1	dbo	USPExampleWithTotal	StoredProcedure	
						Cancel	ĸ

It includes the following main configuration areas:

1. Work Item

a. Click Add Work Item.

We need to connect every change to a work item in Azure DevOps. You can link multiple changes to the same work item. In this example, the work item ID is 76095.

2. Workflow

- a. Select the **Check In** checkbox.
- b. Verify that the displayed workflow name is the correct one. In this example, it is **Eitan Test**.
- c. Select the relevant deployment method. For more information about deployment methods, see Deployment methods.

3. External files

We need to indicate all the files that are relevant to the change that we made.

a. Under Check In External Files, select the relevant files, if required. Our example doesn't include external files.

4. SQL scripts

We need to indicate all the *database objects* that are relevant to the change that we made.

a. Under Check In SQL Scripts, select the relevant database objects. In our example, all the stored procedures are required.

After the commit is successful, we get a changeset number.

🧮 Apps 🔮 effamininy 😵 Timura 🚯 Ops Managar - Ho 😋 TTS 📒 CorsOps - Flowler	cdigital says	et 😝 Genome Genpart	E feading lat
🔓 çora	Changeset ₹22067 was successfully created.		00
x.	ox		• ×
Fullated Work Rems		1	- 10
Work farm ID 17605 Add Work farm			- 11
Mark San			
Deep NUM - Env for			
Cannell.			

You can also view the changeset in Azure DevOps under the work item number 76095.

Azure DevOps	PNMsoft-TFS / Eitan Test / Boards / Work items		🔎 Search 🔠 🗇 🔗 😝		
Eitan Test +	Recently updated 5 Back to Work Items		1 of 10		
 Overview Boards 	TASK TROPS TASK TROP TASK TR				
🛱 Work items					
Boards	State To Do Area Eitan Test		Updated by Eitan Firuzi: Just now		
Backlogs		Details	Depleyment		
∟, Sprints		Priority	Deployment		
➡ Queries ➡ Delivery Plans	Check to add Description Definition Of Done Evidence	2 Remaining Work	To track releases associated with this work the two go to Releases and turn on deployment status reporting for Boards in your pipeline's Options menu. Learn more about deployment status reporting		
Plans	Click to add Definition Of Done Evidence	Activity	Development		
Portfolio plans (Beta)		Blocked	+ Add link		
🗇 Calendar	Discussion	Effort	Created just now		
😰 Repos	Add a comment. Use # to link a work item, ! to link a pull request, or @ to mention a person.	Actual	Related Work		
Pipelines			+ Add link ~		
Test Olana		Artefacts	Add an existing work item as a parent		
		Click to add Artefacts			
Project settings		OneClick actions	-		

Deployment methods

Related only to the workflow that we are deploying and not to files and database objects.

Method	Description
Standard	Deploys the workflow, including all its activities, permissions, and Cora SeQuence standard tables within the workflows.
	This method doesn't deploy stored procedures, views, reference data, and all the end points definition within the workflow (system objects elements).

Method	Description
End Point	Deploys the workflows and its end points. For example, the web consumer's definition and URLs, without the referenced data within the workflow. Doesn't deploy the permissions.
Data Incremental	Deploys data within the workflow. For example, lookup tables, in an incremental method. Doesn't deploy the permissions.
Data Overwrite	Deploys data within the workflow. For example, lookup tables, in an overwrite method – overwrite the exiting data in the target environment. Doesn't deploy the permissions.

Best practices

Dos	Don'ts
 Have a separate workflow that holds only the end points. Deploy the workflow using the End Point deployment method. Have a separate workflow with the data model that holds only the data objects. For example, tables and stored procedures. Deploy the workflow using the relevant data deployment method (Incremental or Overwrite). 	 Don't deploy the same workflow with several deployment methods to create End Points/Data. If required, create a separate workflow for deploying end points and another one, for deploying data.

Execute the Unit Package pipeline

After the commit has been made, we need to execute the CI pipeline Unit Package in Azure DevOps by clicking on the **Create Release** button.

Azure DevOps	PNMsoft-TFS / Eitan Test / Pipelines / R	leleases	P Search I≣ @ % EF
🗉 Eitan Test +	♀ Search all pipelines	Unit Package	🖉 Edit 🦻 Greate release
Overview	<u>⊨</u> ⊡ û + New ∨	Releases Deployments Analytics	Ⅲ All releases ✓ ♀ ♀ ⋮
n Boards	Test Ø Stage 1	Releases Created Stages	
P Repos	Processes Release	€ 1 7/14/2021, 4:37:01 PM	•
Pipelines	Pass QA		
Pipelines	Stand-alone DevOps Integration Ø QA		
Environments	Unit Package 🙏		
Ø Releases	O Dev		
III Library			
🐨 Task groups			
P Deployment groups			
👗 Test Plans			
Artifacts			

We add the work item number in the **ApprovedWorkItem** text box, 76095, and the Release Number, 1.1 (in this example). And then, click the **Create** button to start the Job.

C Azure DevOps	PNMsoft-TFS / Eitan Test / Pipelines / Re	eleases	Create a new release	\times
Eitan Test +	Search all pipelines	Unit Package	Unit Package	
Overview	≝ ⊡ 🗊 + New ∨	Releases Deployments Analytics	 Pipeline ^ Click on a stage to change its trigger from automated to manual. 	
Boards	Test Stage 1	Releases Created	ي Dev	
P Repos	Processes Release	6F 1 7/14/2021, 4:37:01 P	Stages for a trigger change from automated to manual. ①	
Pipelines	Pass QA			\sim
uiu Pipelines	Stand-alone DevOps Integration QA			
Environments	Unit Package		Variables A	
🖉 Releases	🖉 Dev 🕅 🕅		Edit value for the variables to be overridden during release creation.	
II% Library			Name Value Scope	
🐨 Task groups			ApprovedWorkItems 76095 Release	
T Deployment groups			Release.Number 1.1 Release	
👗 Test Plans			Palace description	
Artifacts			Release description	
Project settings			Create	

In case we want to deploy multiple work items in one go (that is, multiple changesets) in the Unit Package, we can add multiple work items by separating them with a comma without any space (see screenshot below).

 Connection with live updates has been term 	inated. Please refresh the page to :	see the updated view	Unit Package			~
Search all pipelines III □ □ + New ~	Unit Package	nalytics	Pipeline ^ Click on a stage to change its	trigger from automated to manual.		
Processes Release Processes QA	Releases	Created	Stages for a trigger change	ge from automated to manual.	0	
Unit Package	(P) 4	9/9/2021, 12:27:22 PI				\sim
Test Stage 1	€ ³	9/9/2021, 11:22:24 AI	Variables A Edit value for the variables to	be overridden during release creati	on. (i)	
Stand-alone DevOps Integration	(f) 2	8/11/2021, 4:26:13 PI	Name	Value	Scope	
U CA	(7/14/2021, 4:37:01 Pt	ApprovedWorkItems	76197,81266,76439	Release	
			Release.Number	1.0	Release	
			Release description		•	
						h
			Create Cancel			

Deploying several changesets saves time and provides the option to package a few workflows from different work items to the same package.

Azure DevOps PNMsoft	/ Eitan Test / Pipelines / Releases / Unit Package / 2	EF
Eitan Test + ↑ Ur	ackage > 2 > Dev < 💿 In progress	
✓ Pipe	Tasks Variables Logs Tests 🕜 Deploy 🛇 Cancel 💟 Refresh 🛓 Download all logs 🗡 Edit 🗸 …	2
Boards Deploy	process Deployment group job Started: 8/11/2021, 4:26:2	28 PM
P Repos 💿 Dep	ent group job Deployment group: Eitan Test-YuvalTest-Cl CD De	23s
Pipelines		
utu Pipelines	0 PENDING	
Environments		
🔊 Releases	0 0	
II% Library	SUCCEEDED FAILED	
😇 Task groups	1 IN-PROGRESS	
Deployment groups		
👗 Test Plans		
Artifacts		

After the CI pipeline job has started, we can track its progress on the following screen:

After the CI pipeline finishes running, the **Package.zip** is created as an artifact in the deployment package folder.

Execute the Processes Release pipeline

After the artifact Package.zip has been created, we can now execute the CD pipeline (Processes Release pipeline).

To view a list of the pipelines, in the Azure DevOps portal, in the left-panel menu, click **Releases**, and then select **Processes Release**.

Graphical view of the Processes Release CD pipeline

Ċ	Azure DevOps	PNMsoft-TFS / Eltan Test / Pipelines / Releases / Processes Release	
ET	Eitan Test $+$	All pipelines > T Processes Release	🗟 Save 🖋 Create release 🗮 View releases \cdots
2	Overview	Pipeline Tasks	
	Boards		
2	Repos	Artifacts + Add Stages + Add ∑	
	Pipelines	<u> </u>	
utu	Pipelines	Prostolikar & QA & & Pass QA	8
1	Environments	Loopi ouer X 6jobs. 11tasks X 1job. 0 task	
2	Releases	() Schedule	
08	Library	O not set	
	Task groups		
T	Deployment groups		
4	Test Plans		
A	Artifacts		

Azure DevOps	PNMsoft-TFS / Eltan Test / Pipelines / I	Releases		P Search	III 🗇 🖓 🗲
Eitan Test +	 Search all pipelines 	Processes Release			C Edit Create release
Overview	≝ ⊵ ŵ + New ~	Releases Deployments Analytics			🗄 All releases 🗸 🤘 🗄
Noards	Unit Package Ø Dev	Releases	Created	Stages	
Repos	Test	Package Release-122 (***) 0f41a7d6 \$? master	7/22/2021, 1:21:34 PM	O QA O Pass QA	
Pipelines	Processes Release	Package Release-121	7/22/2021, 1:17:42 PM	QA O Pass QA	
Environments	Stand-alone DevOps Integration	Package Release-120	7/15/2021, 12:48:52 PM	QA O Pass QA	
Releases	© QA	Package Release-119 (*) 0f41a7d6 % master	7/15/2021, 12:46:50 PM	O Pass QA	
Task groups Deployment groups		Package Release-118	7/14/2021, 9:55:06 PM	QA O Pass QA	
Test Plans		Package Release-117 ③ 0f41a7d6 多 master	7/14/2021, 6:01:23 PM	O Pass QA	
Artifacts		Package Release-115	7/14/2021, 5:20:15 PM	O Pass QA	
		Package Release-114 (************************************	7/14/2021, 4:59:04 PM	O Pass QA	
Project settings	<	Package Release-113	7/14/2021 4/65-45 044		

To execute the pipeline, click the **Create release** button.

Enter values for the **ReleaseNumber** (give a version number to the release) and **ShouldStopStartServices** fields (specify if IIS should be stopped and restarted after the packages are deployed).

To begin execution, click **Create**.

Azure DevOps PNMsoft-TFS / Eitan Test / Pipelines	/ Releases	Create a new release ×
Eitan Test + C Search all pipelines	Processes Release	Processes Release Stages for a trigger change from automated to manual.
Cverview	Releases Deployments Analytics	v
Boards Unit Package	Releases Crea	ite.
Repos Test	Package Release-122 (************************************	Artifacts Artifacts Select the version for the artifact sources for this release
Pipelines Stage 1	Package Release-121	Source alias Version
ta Pipelines Processes Release ☆ Pass QA	€F (≫)0f41a7d6 \$P master 7/2.	_DropFolder 6d1849ed (Auto commit by CICD Ad V
Environments Stand-alone DevOps Integration	Package Release-120 (************************************	5/2
Ø Releases ◎ QA	Package Release-119	Variables ^ Edit value for the variables to be overridden during release creation. ()
II's Library	◆ 0f41a7d6	Name Value Scone
Task groups	Package Release-118 7/1	4/2 Release Number 11 Release
T Deployment groups	- Working of master	ShouldStonStartServices True Release
👗 Test Plans	Package Release-117 Pockage Release-117 W0f41a7d6 §P master	
Artifacts	Package Release-115 So 0f41a7d6 \$° master 7/1.	4/3 Release description
	Package Release-114 (*) 0f41a7d6 % master 7/1	4/3
Project settings	Package Release-113	Creste Cancel

You can track the progress of the execution (see below).

lease	Stages					
Manually triggered by Eitan Firuzi 8/11/2021, 4:30 PM		QA In progress Job 1/6	8-	Pass QA O Not deploye	ed	
Artifacts		1 Target				

You can also follow the process in the logs view.

		SI	atus 🗸
Start time \checkmark DEVCICD2	CDEVCICD2 Target: CDEVCICD2	Started: 8/11/2021, 4	I:30:27 PM 30s
progress	Initialize job · succeeded		<1s
	PowerShell Stop Service · succeeded		9s
	PowerShell Stop Service Usucceeded		7s
	PowerShell Stop Service		12s

In the above screenshot, the process is in the last stage.

The Processes release pipeline has two stages: QA and Pass QA.

¢	Azure DevOps	PNMsoft-TFS / Eitan Test / Pipelin	nes / Release	s / Processes Release / Package Release-129	& Search	¥	Ô	0
ET	Eitan Test +	Processes Release > Packa	age Release-1	29 ~				
	Overview	Pipeline Variables History	$+$ Deploy \vee	S Cancel A Approve multiple ♥ Refresh ✓ Edit ✓ ·	••			
-	Boards	Release	Stanes					
2	Repos	herease	Stages					
2	Pipelines	Manually triggered		DA Pass DA				
utu	Pipelines	by Eitan Firuzi 8/11/2021, 4:30 PM		Pending approval O Not deployed				
1	Environments			On Eitan Firuzi				
2	Releases	Artifacts		✓ Approve				
0%	Library	_DropFolder						
	Task groups	8 master						
eņe †	Deployment groups							
4	Test Plans							
	Artifacts			La				

In the QA stage, the administrator needs to check and confirm if the deployment was done correctly and then select **Approve**.

Let's check the workflow in the Test server to see if the deployment has been successful.

G cor SeQu	ra uence		Eitan Test			(2)
o 🗹	🗈 🗴 🔧 °+ 🗭 📼 🖻	🗇 🥝 🔗 Validate 🇊 Delete Wo	orkflow 🔗 Export 🔹 육 Set Permissio	ons 🔒 Source Control 👻 🕷 Proc	cess Lab 🕞 Commit to Azure Dev	Ops
Toolbox Find an Activ	x List View <	Start			Properties Workflow	
Human Server Side	X+Y Assign Built-in Command Empty Compared for the set of the s	End	Test Form		(Name) Alias Description Workflow ID Misc Dipploy Field	Eitan Test Eitan Test Eitan Test 7649dc27-6586-4700 98d8-0da5d9120836
	ev [©] _o sharing ev [©] _o store Sharing SP _☉ stored Procedure	ے ا	Emp Role Form		Stage Set Statistics Variables	
	_G S _∰ Sub Workflow	2	jg CiCD Test			

The deployment was successful. The CICD form has been added to the workflow.

If we check the Source control history, we can see that the last user who checked in the workflow was the CI/CD user.

G co	ro vence	Eitan Test	© @
G		🖉 🎯 Validate 🌐 Delete Workflow 🖄 Export * 🤹 Set Permissions 🔂 Source Control * 🐞 Process Lab 🕞 C	Commit to Azure DevOps
Toolbo	List View <		Properties >
Find an Acti	vityQ. X+Y Assign	History List	Workflow (Name) Eitan Test Alias Eitan Test
Human	Built-in Command	ID Check-In Date Checked-In By Label Name Description	Description Eltan Test
Server Side	 Empty 	6 11/08/2021 14:31 CCD User 202108111331 28002 Deployment t	Verdifier ID 7649427 6586-0700 9668-964569120816 Misc Display Feld Stage Set Statistics
	D™ and horizon.	Restore Version Close	

The first approval is done by the user who executes the pipeline. This user needs to verify that the implementation completed successfully, based on its purpose.

Azure DevOps	PNMsoft-TFS / Eitan Test	/ Pipelines / Releases / Processes R	kelease / Package Release-129	III 🗇 💿 🖧 🗗
Eitan Test +	↑ Processes Release	e $>$ Package Release-129 \vee		Help
Cverview	Pipeline Variables Hist	tory + Deploy \vee \otimes Cancel \otimes	Approve multiple \circlearrowright Refresh \checkmark Edit \lor \cdots	2
Boards		Channel	OA.	×
😰 Repos		stages	Post-deployment conditions * Pending approval	
Pipelines	ually triggered		Approvers C View logs	
utu Pipelines	Eitan Firuzi	• Pending approval	Approval pending for 1 minute	S Timeout in 30d
Environments	2021, 4:30 PM	On 🔮 Eitan Firuzi for 1 minute	waiting for an approvers to approve in sequence.	
\$₽ Releases	cts	✓ Approve	EF Eltan Firuzi S Pending for 1 minute	Reassign
II% Library	pFolder 49ed		Comment	
T Deployment groups	aster			
Lans Test Plans			Approve Reject	
Artifacts				
Project settings <	4		,	

After approval, the pipeline moves to the Pass QA stage.

After this stage is completed, the QA person needs to approve the changes to complete the process. This approval cycle is not mandatory, but it's recommended. The QA user should make sure that the workflows have been deployed and that they are working properly.

elease	Stages				
Manually triggered by Eitan Firuzi 8/11/2021, 4:30 PM		QA Succeeded on 8/11/2021, 4:34 PM		QA nding approval Eitan Firuzi	
Artifacts			→ App	rove 🛇 穿	
DropFolder					

Pass QA stage approval screen.

Processes Release > Package Release Pipeline Variables History + Deploy	xe-129 ∽ ∽ 🛇 Cancel 🕺 Approve multiple 🖑 Refresh 🖌 Edit ∽ ···	ී Help දී
Stages	Pass QA Pre-deployment conditions • • • Pending approval Approvers View logs	>
QA Succeeded on 8/11/2021, 4:34 PM	Approval pending Waiting for all approvers to approve in sequence.	Timeout in 30d
	Eitan Firuzi Pending Comment	Reassign
	Defer deployment for later Approve Reject	<i>A</i>

After Pass QA is approved, the CD process is completed.

C Azure DevOps	PNMsoft-TFS / Eltan Test / Pipelin	es / Releases / Processes Release / Package Release-129	₽ Search	1≣ €	1 0	₽ , EF
ET Eitan Test +	↑ Processes Release > Packa	ge Release-129 \vee				③ Help
Overview	Pipeline Variables History	- Deploy 🗸 🚫 Cancel 🖏 Refresh 🖌 Edit 🗸				2
n Boards	Release	Stanos				
😢 Repos	Release	Stages				
Pipelines	Manually triggered					
Pipelines	by Eitan Firuzi 8/11/2021, 4:30 PM	Succeeded	4			
Environments		on 8/11/2021, 4:34 PM on 8/11/2021, 4:35 PM				
,∯ Releases	Artifacts	🕐 Redeploy 🖉 Lo	pgs			
II∿ Library	_DropFolder					
🐨 Task groups	6d1849ed P master					
T Deployment groups						
📥 Test Plans						
Artifacts						

After QA is approved, the deployment continues to the next environment, usually Production, but it depends on the specific project.

Important

When deploying to Production, or Production-like environments, make sure that the package that you deploy is approved and final.

Customize Cora SeQuence

The pipepline available for Cora SeQuence customizations is the **Release Shared Resources** pipeline.

The Release Shared Resources pipeline reads files from a specified repo and propagates them in the required Cora SeQuence folders of the target server.

We need to create a repo to be used by the customization pipeline:

- Repo type: GIT
- Repo name: Same name as the Azure DevOps project name.

¢	Azure DevOps		PNMsoft-TFS / Eitan Test / Repos / Files / 🔶 Eitan Test 🗸			₽ Search			Ô	0
ET	Eitan Test	+	🔶 Eitan Test	:						
	Overview		V 🖿 AppGlobalResources		Files		🔛 Se	t up build	(🖵 Ck
	Boards		Delete Me Message Message Delete Me Del		Contents History					
8	Repos		Delete Me		① Committed ϕ e31c1c36: Deleted azure-pipeline	es.yml		Crea	ite a pu	ıll requ
٦	Files									
¢	Commits				Name 1	Last change	Commit	s		
ድ	Pushes				AppGlobalResources	Jun 2	<u>ca5d4</u> @	35c Adde	d Delete	e Me E
s	Branches				Shared Resources	Jun 2	1cfeca	a2f Adde	d Delete	e Me E
0	Tags									
ເນ	Pull requests									
2	Pipelines									
Å	Test Plans									
	Artifacts									

Note

It is recommended to create a separate repo for each customer. The naming convention is **<CustomerName+CICD>**.

The developers upload their files directly into the repos that have been created for them in the appropriate folder.

For example, if developers have made changes to javascript files that are used by forms, all they need to do is to upload the updated files to the appropriate folder in the repo. In our example, it's the folder **shared resources\js**. See screenshot below.

Example of repo (the customer name has been hidden).

Azure DevOps	PNMsoft-TFS / Becle / Repos / Files /				1	ů 0	٩,
a an b +	• beix	:					
🥶 Overview	> AppGlobalResources		Files		succeeded	🖵 Clone	÷
n Boards	✓ ■ Shared Resources	÷	Contents History				2
😢 Repos	Analytics Ecomponents		Name Î	Last change	Commits		
Files	✓ 🖿 js		AppGlobalResources	Thursday	d6770696 Update	ed	
¢ Commits	JS Custom .js		Shared Resources	Wednesday	8c960d95 Update	ed Business	
හ Pushes	> Resources		M4 README.md	Jun 2	e1786a5f Added	README	
🖗 Branches	> Styles						
Tags	> 🖿 Themes		Introduction	is section evaluin the ob-	viactives or the motiv	ation behind t	thic
입 Pull requests	> sample.xml		project.	is section explain the ob	Jecuves of the motiva	ation bening	
Pipelines	🗅 test.txt		Getting Started				
👗 Test Plans	MI README.md		TODO: Guide users through getting your code up and about:	running on their own sy	stem. In this section y	you can talk	
Artifacts			Installation process Software dependencies Latest releases				

The **Release Shared Resources** pipeline is used for customizations. It's the main pipeline used for customer deployments.

Azure DevOps PNMsoft-TFS / Pipelines /	Releases	∠ Search	1≡ 🗂	0 %
Search all pipelines	Release Shared Resources		🖉 Edit 🔗 Create	e release
🔁 Overview 🗮 🖻 🗊 🕂 New 🛇	Releases Deployments Analytics		E All releases ~	- 7 :
Boards Processes Release PRD REAL	Releases Created	Stages		
Repos Stand alone DevOps Integration	EZ Release-316 ⊗ d67706 \$ ^p master 12/9/2021, 12:07:33 PM	O Dev OLD O QA OLD	0 0	O Re-
Pipelines PRD Release Shared Resources	Release-315 0	O Dev OLD	0) O	O Re
Pipelines	Watera412 § ⁹ master			
Environments Unit Package Paleacer Down BEAL	EZ Release-314 ⊗ 8c960d [®] master 12/8/2021, 2:03:07 PM	O Dev OLD	0	O Re.
II Library	€ Release-313	O Dev OLD	• · · ·	O Re
 Task groups Deployment groups 	Release-312 12/8/2021, 1:19:39 PM ⊗ 51d3be \$ ^p master	O Dev OLD	0	O Re
Last Plans	Release-311 12/8/2021, 12:39:01 PM	O Dev OLD	Ø 0	O Re.
Artifacts	Release-310 12/8/2021, 12:28:04 PM Sega6b3 \$P\$ master	O Dev OLD	• · · •	O Re.

Important

The DevOps platform includes additional pipelines, which are used for the initial Cora SeQuence installation. These pipelines are not covered as they are used only by CI/CD System Admins.

The Release Shared Resources pipeline takes files from the repo (artifacts) and updates the Dev and QA environments.

¢	Azure DevOps	PNMsoft-TFS / / Pipelines / Releases / Release Shared Resources	P Search I≣ ⊕ ⊘ Pe
	nn +	All pipelines > ** Release Shared Resources	🖶 Save 🚀 Create release 🗮 View releases 🛛 🚥
6	Overview	Pipeline Tasks Variables Retention Options History	
	Boards		
8	Repos	Artifacts + Add Stages + Add	
2	Pipelines	()	
	Pipelines	♦ <u>&</u> DEV <u>B</u> & TST <u>S</u>	R & PRD R
l	Environments	R 1 job, 7 tasks R 1 job, 7 tasks	R 1 job, 7 tasks
5	Releases		
01/	Library	Schedule not set	
•	Task groups		
• D • †	Deployment groups		
a	Test Plans		
8	Artifacts		

Most of the tasks in the Release Shared Resources pipeline are to **delete** files and **copy** files, which replace them with the ones from the repo.

C Azure DevOps	PNMsoft-TFS / Pipelines / Releases / Release Shared Resources		
a n +	All pipelines > 👎 Release Shared Resources		Save
Overview	Pipeline Tasks - Variables Retention Options History		
Noards	DEV Deployment process		Copy files 🛈 🏠 View YAML 📋 Remove
😢 Repos	Deployment group job	+	Task version 2.* V
Pipelines	Delete files from \$(System.ArtifactsDirectory)_ \Shar		Display name *
🟥 Pipelines	Delete files		Copy JES Shared Resources files
L Environments	Delete files from \$(System.ArtifactsDirectory)\\App		Source Folder ①
🔊 Releases	Copy JES Shared Resources files	⊘	\$(System.ArtifactsDirectory)_ \Shared Resources
II\ Library	Copy mes		Contents * ①
🐨 Task groups	Copy BRS Shared Resources files		**
Deployment groups	Copy Administration Shared Resources files		Target Folder * 🕜
📥 Test Plans	Copy Flowtime Shared Resources files		C:\Program Files\Genpact Digital\Cora SeQuence\Job Execution Service 1\Shared Resources
Artifacts	Copy App GlobalResources to Flowtime		Advanced V
	40 copy files		Control Options V
			Output Variables 🗸

The pipeline executes each task in the list in the specified order.

The successful completion of the pipeline indicates that the customization has been deployed to all the configured environments (commonly in this order DEV > TEST > PROD).

elease	Stages			
Manually triggered by S 11/29/2021, 10:19 AM		Dev OLD DO NOT CO O Not deployed		
Artifacts 25dfeaed P master		QA OLD DO NOT CON O Not deployed		
		DEV	TCT	PRD

 $\stackrel{\bullet\bullet\bullet}{\uparrow}$ Release Shared Resources > Release-299 \sim