

# TechNotes – Azure DevOps Release Pipeline with DBB Artifacts (NLopez IBM/DAT)

## Table of Contents

Overview .....	1
Setup.....	3
CI pipeline scripts .....	4
Define Azure Variables.....	4
CI-Task: Package DBB Artifacts .....	5
CI-Task: Windows Agent SCP.....	6
CI-Task: Publish DBB package.....	7
Release pipeline scripts .....	8
Release-Task: Universal download - DBB Package to Agent .....	8
Release-Task: Init Deploy WorkDir .....	9
Release-Task: Windows Agent SCP: Copy Azure DBB Package to zOS Staging area .....	10
Release-Task: Deploy Package to zOS.....	11
Sample Script Repo (Private).....	12

## Overview

This document builds upon the example described in the "Azure DevOps and IBM Dependency Based Build Integration"<sup>1</sup> document. Some CI tasks will be redefined to support a basic Azure Release Pipeline of DBB generated artifacts.

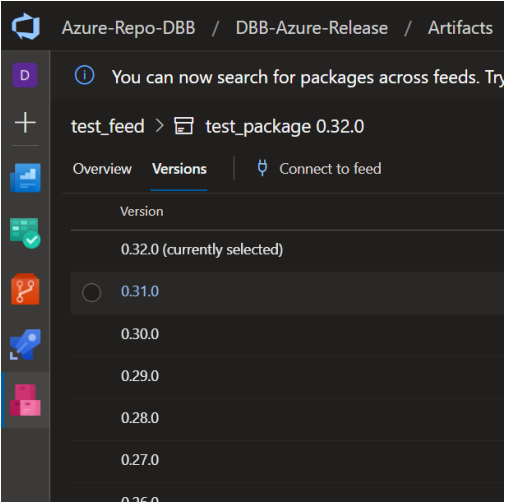
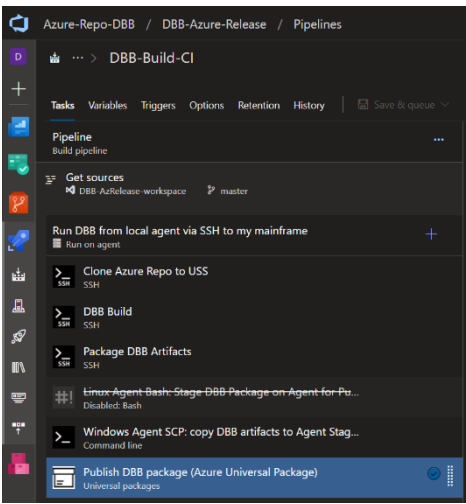
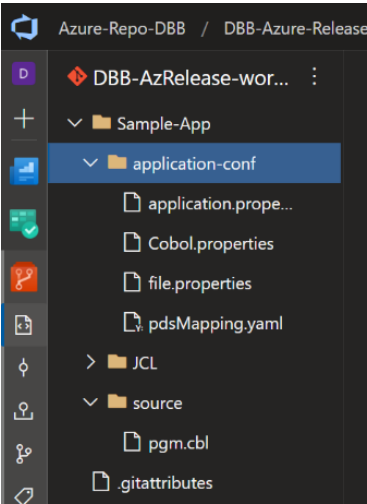
The example uses an Azure Repo with a simple mainframe Cobol batch program. The repo is configured to trigger an Azure CI pipeline (DBB-Build-CI) with the following tasks:

- Clone the repo to the Mainframe.
- Run a DBB build
- Package the DBB artifacts into tar file including a yaml manifest and pdsMapping files.
- SCP the package to the Azure Agent staging area.

---

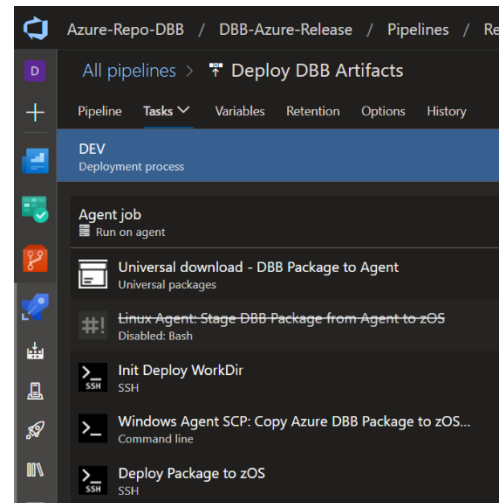
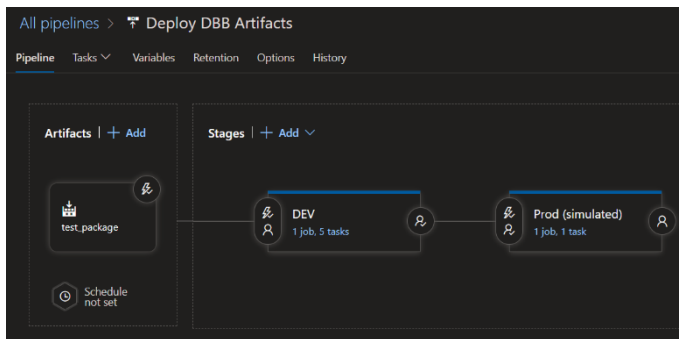
<sup>1</sup> Available under the resource section of the IBM DevOps Acceleration Program's site <https://www.ibm.com/support/pages/node/6422813>

- Invoke Azure's Universal Plugin to store and version the package in an Azure Artifacts Feed (test\_feed)



The example Release pipeline has a DEV stage and a Prod stage to illustrate deployments from one to the other. The DEV stage has these tasks:

- Universal download to pull the latest version of a package and stage it on the agent.
- Initialize the target mainframe working directory.
- SCP the package to the mainframe working directory.
- Deploy - Invokes a sample script to unpack and deploy the artifacts onto the target host.



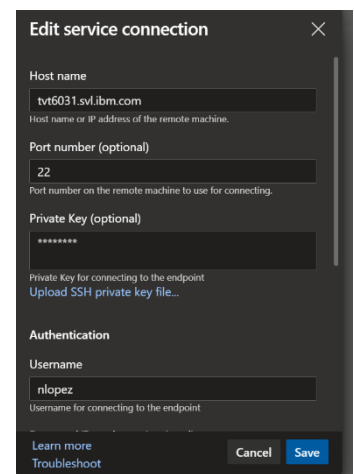
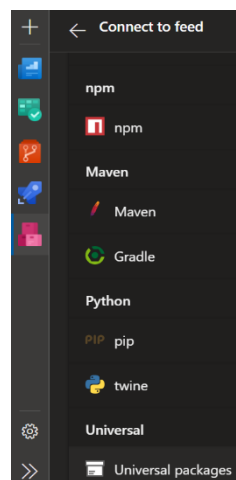
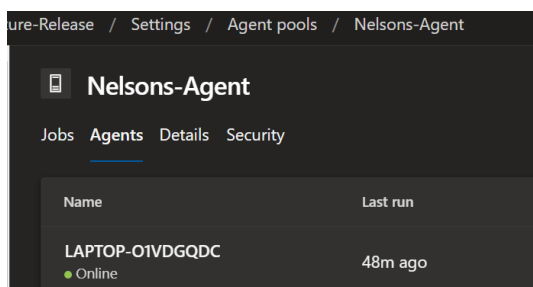
See a demo -> <https://youtu.be/RSBxKkxhxE?&t=90>

## Setup

It is assumed you are familiar with Azure DevOps services, setting up agents, end-points, security networking requirements and Artifacts feeds.

A basic DBB based Azure DevOps CI/CD configuration requires:

- An Azure agent with access to one or more mainframe hosts over SSH (SAF User ID/Password + OMVS Segment)
- Service connection(s) to each host
- Pre-installed and configured zDevOps mainframe stack – IBM DBB, Rocket Git
- A set of Azure pipeline scripts to launch tasks – next section.
- An Azure Artifacts Universal Packages Feed
- A preconfigured Repo (Azure repo is used in this example)



## CI pipeline scripts<sup>2</sup>

As mentioned, these notes are an extension of the "Azure DevOps and IBM Dependency Based Build Integration" doc. It explains how to set up a basic CI pipeline to publishes artifacts using Urban Code Deploy. For an Azure Release pipeline, the publishing tasks will be changed. Instead of UCD/Artifactory we will use Azure's Artifact Repo (Feeds) and a sample groovy deployment scripts

### Define Azure Variables

Azure variables simplify and standardize pipelines across projects. This example suggests the following variables:

- **MyApp** - DBB's main application source folder
- **MyRepo** - the application repo to be cloned as an SSH URL. Azure provides a repo URL system variable but only in HTTPS format. This example uses SSH.
- **MyWorkDir** - a directory where all build output will be stored. This path should have a mount point with enough free space to support your build output needs and will require some log retention processes. This example shows how to use two Azure system variables to create a unique name.
  - o `$buildid` - is automatically assigned at build time with a unique number.
  - o `$teamProject` - is used as a component name for improved traceability.
- **MyWorkSpace** - DBB main working area. This is the last part of the repo URL (a.k.a the repo name)
- **zOS\_Build\_LPAR** – the ssh string to access a Build host
- **zOS\_Deployment\_LPAR** – the ssh string to access the target deployment host (in this example it's the same as the Build host but can be any host with the DBB Toolkit).
- **zOS\_DeploymentDir** – a working directory to stage and deploy the artifacts.

Variables	
Name ↑	Value
MyApp	Sample-App
MyWorkDir	Azure-WorkSpace/\${System.TeamProject}_BUILD_\${Build.buildid}
MyWorkSpace	DBB-AzRelease-workspace
MyRepo	git@ssh.dev.azure.com:v3/Azure-Repo-DBB/DBB-Azure-Release/DBB-AzRelease-workspace
zOS_Build_LPAR	nlopez@tvt6031.svl.ibm.com
zOS_Deployment_LPAR	nlopez@tvt6031.svl.ibm.com
zOS_DeploymentDir	Azure-WorkSpace/\${System.TeamProject}_DEPLOY_\${Release.DeploymentID}

---

<sup>2</sup> All scripts are provided as samples with no user documentation. You are responsible for testing and supporting them. IBM provides no formal support or liability. The IBM DevOps Acceleration team can be consulted on guidance and limited support through the POC/Pilot engagement process.

## CI-Task: Package DBB Artifacts

This replaces the task "*Define Task 3 - Publish to UCD*" used in the basic CI pipeline. It runs after the DBB Build task. It reads the DBB BuildReport.json file and tars all supported artifacts by deployment type. The sample scripts were stored in \$HOME/Azure-pipeline-scripts. Your location may differ. The sample shell scripts can be obtained by your IBM DAT coordinator.

### Commands:

```
Azure-pipeline-scripts/AzPackage.sh $HOME/$(MyWorkDir) $(MyApp) $(Build.buildid)
```

The screenshot displays the Azure DevOps interface for a pipeline named 'DBB-Build-CI' with build ID 1482. On the left, a list of jobs is shown, including 'Run DBB from local ag...', 'Initialize job', 'Checkout DBB-AzReleas...', 'Clone Azure Repo to USS', 'DBB Build', 'Package DBB Artifacts', 'Windows Agent SCP: cop...', 'Publish DBB package (A...', 'Post-job: Checkout DBB...', 'Finalize Job', and 'Report build status'. The 'Package DBB Artifacts' job is highlighted.

The main panel shows the configuration for the 'Package DBB Artifacts' task. The task type is 'SSH'. The command is set to 'Commands', and the command text is 'Azure-pipeline-scripts/AzPackage.sh \$HOME/\$(MyWorkDir) \$(MyApp) \$(Build.buildid)'. A yellow arrow points to the 'Commands' radio button.

The execution log for the 'Package DBB Artifacts' task is displayed on the right. It shows the task starting, establishing an SSH connection, and executing the command. The log output includes the following information:

- Task : SSH
- Description : Run shell commands or a script on a remote machine using SSH
- Version : 0.180.0
- Author : Microsoft Corporation
- Help : <https://docs.microsoft.com/azure/devops/pipelines/tasks/deploy/ssh>
- Trying to establish an SSH connection to \*\*\*@tvt6031.svl.ibm.com:22
- Successfully connected.
- Azure-pipeline-scripts/AzPackage.sh \$HOME/Azure-Workspace/DBB-Azure-Release\_BUILD\_1482 Sample-App 1482
- .profile executed current shell = /bin/sh
- \*\*\*\*\*
- \*\* Started: Azure Packaging on HOST/USER: z/OS TVT6031 03.00 02 2964 NLOPEZ
- \*\* WorkDir: /u/\*\*\*\*/Azure-Workspace/DBB-Azure-Release\_BUILD\_1482
- \*\* DBB Artifacts: /u/\*\*\*\*/Azure-Workspace/DBB-Azure-Release\_BUILD\_1482/build.20210228.010045.000
- \*\* App: Sample-App
- \*\* Version: 1482
- \*\* Package Script: /u/\*\*\*\*/Azure-pipeline-scripts/package.groovy
- \*\* Packaging using DBB created artifacts defined in:
- \*\* /u/\*\*\*\*/Azure-Workspace/DBB-Azure-Release\_BUILD\_1482/build.20210228.010045.000/BuildReport.json
- \*\* Number of load modules packaged for staging: 1
- \*\* Packaging Done
- \*\* App TAR file =>/u/\*\*\*\*/Azure-Workspace/DBB-Azure-Release\_BUILD\_1482/Sample-App.tar
- \*\* Artifactory CURL =>none.com/TBD
- \*\* Build finished
- Finishing: Package DBB Artifacts

# CI-Task: Windows Agent SCP

## Commands:

```
@echo "Agent Staging Dir ->" $(Build.ArtifactStagingDirectory)
@echo "DBB Build Host ->" $(zOS_Build_LPAR)
@echo "***"
scp $(zOS_Build_LPAR):$(MyWorkDir)/$(MyApp).tar $(Build.ArtifactStagingDirectory)
dir $(Build.ArtifactStagingDirectory)
```

Jobs in run #1482

DBB-Build-CI

Jobs

Run DBB from local ag... 1m 33s

Initialize job <1s

Checkout DBB-AzReleas... <1s

Clone Azure Repo to USS 7s

DBB Build 37s

Package DBB Artifacts 7s

Windows Agent SCP: cop... 5s

Publish DBB package (A... 34s

Post-job: Checkout DBB... <1s

Finalize Job <1s

Windows Agent SCP: copy DBB artifacts to Agent Staging Dir

1 Starting: Windows Agent SCP: copy DBB artifacts to Agent Staging Dir

2 =====

3 Task : Command line

4 Description : Run a command line script using Bash on linux and macOS and cmd.exe on Windows

5 Version : 2.182.0

6 Author : Microsoft Corporation

7 Help : <https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/command-line>

8 =====

9 Generating script.

10 ===== Starting Command Output =====

11 "C:\WINDOWS\system32\cmd.exe" /D /E:ON /V:OFF /S /C "CALL "C:\AzureAgent\vsts-agent-win-x64-2.172.2\\_work\\_temp\3571296b-79e5-4dad-8ffd-59c8b19c9c4a.cmd""

12 "Agent Staging Dir ->" C:\AzureAgent\vsts-agent-win-x64-2.172.2\\_work\7\

13 "DBB Build Host ->" \*\*\*@vt6031.svl.ibm.com

14 \*\*\*\*

15 Volume in drive C is Windows

16 Volume Serial Number is 6804-BC35

17

18 Directory of C:\AzureAgent\vsts-agent-win-x64-2.172.2\\_work\7\

19

20 02/28/2021 08:01 AM <DIR> .

21 02/28/2021 08:01 AM <DIR> ..

22 02/28/2021 08:01 AM 40,960 Sample-App.tar

23 1 File(s) 40,960 bytes

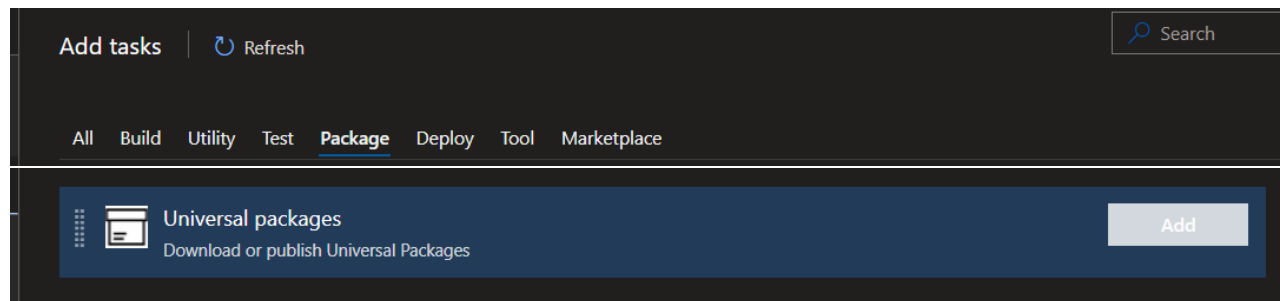
24 2 Dir(s) 160,373,690,368 bytes free

25 Finishing: Windows Agent SCP: copy DBB artifacts to Agent Staging Dir

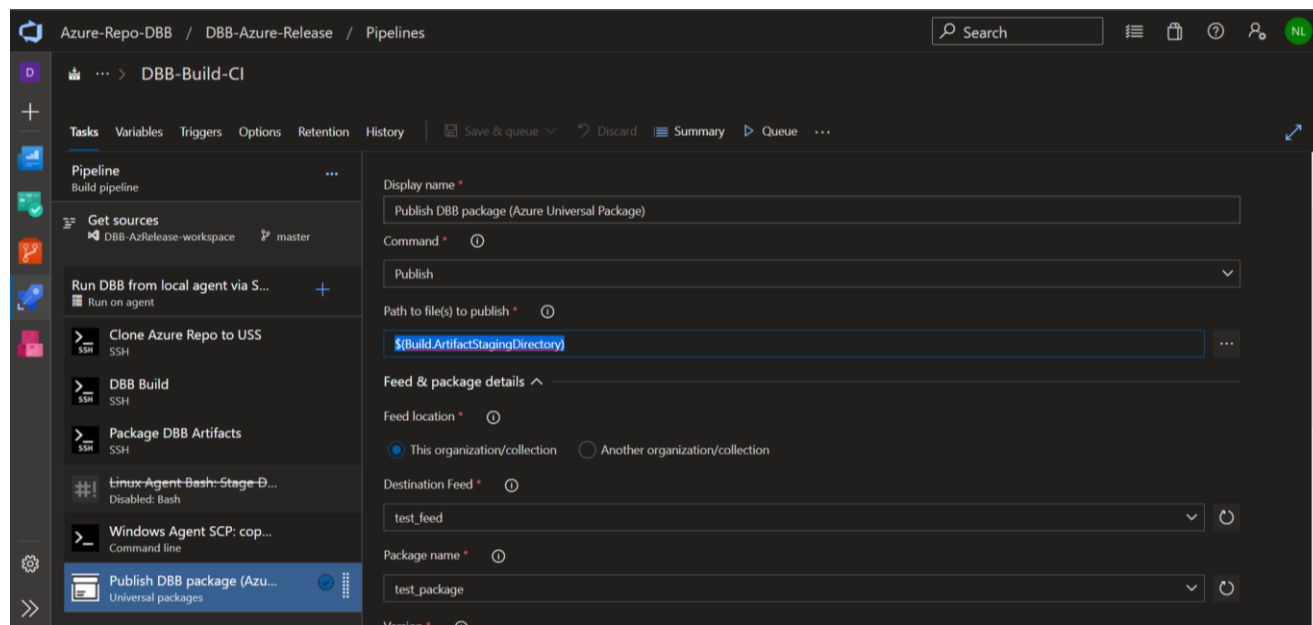
This sample uses a windows agent using pre-defined SSH keys to the host.

## CI-Task: Publish DBB package

This is a standard Azure Universal Package plugin available from Microsoft. Configure it with your Feed and package name.

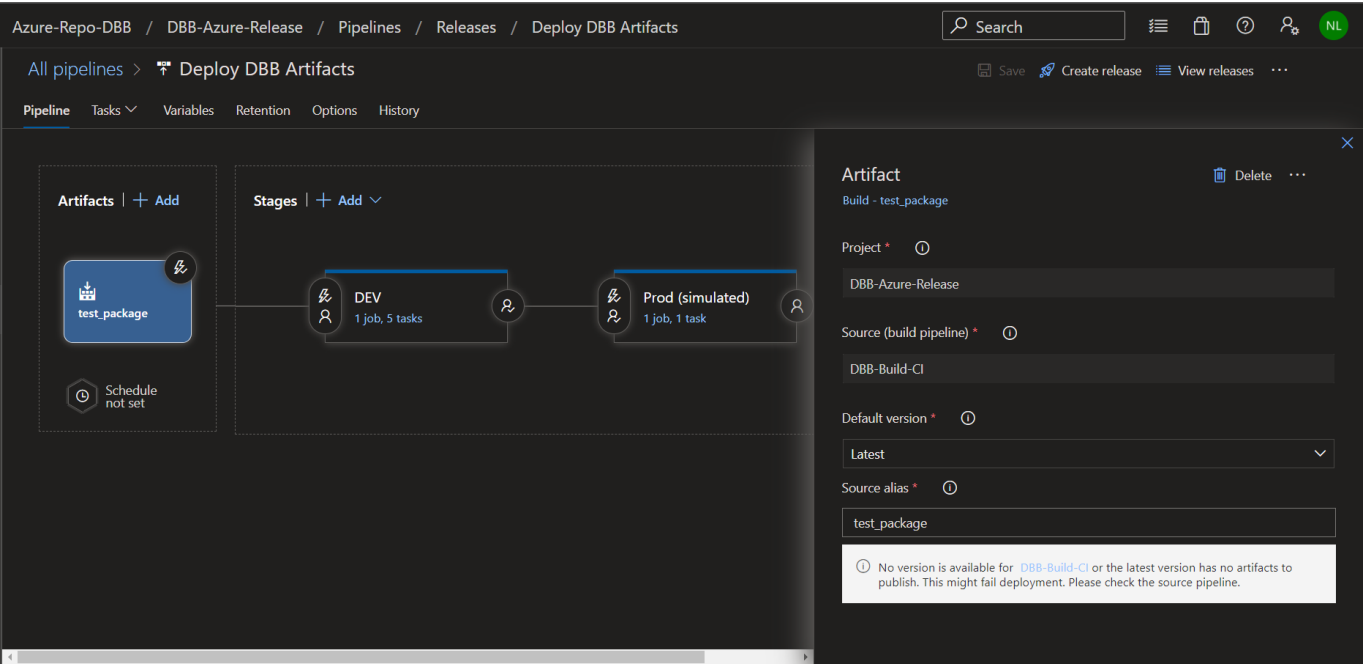


Use this path for publishing `$(Build.ArtifactStagingDirectory)`



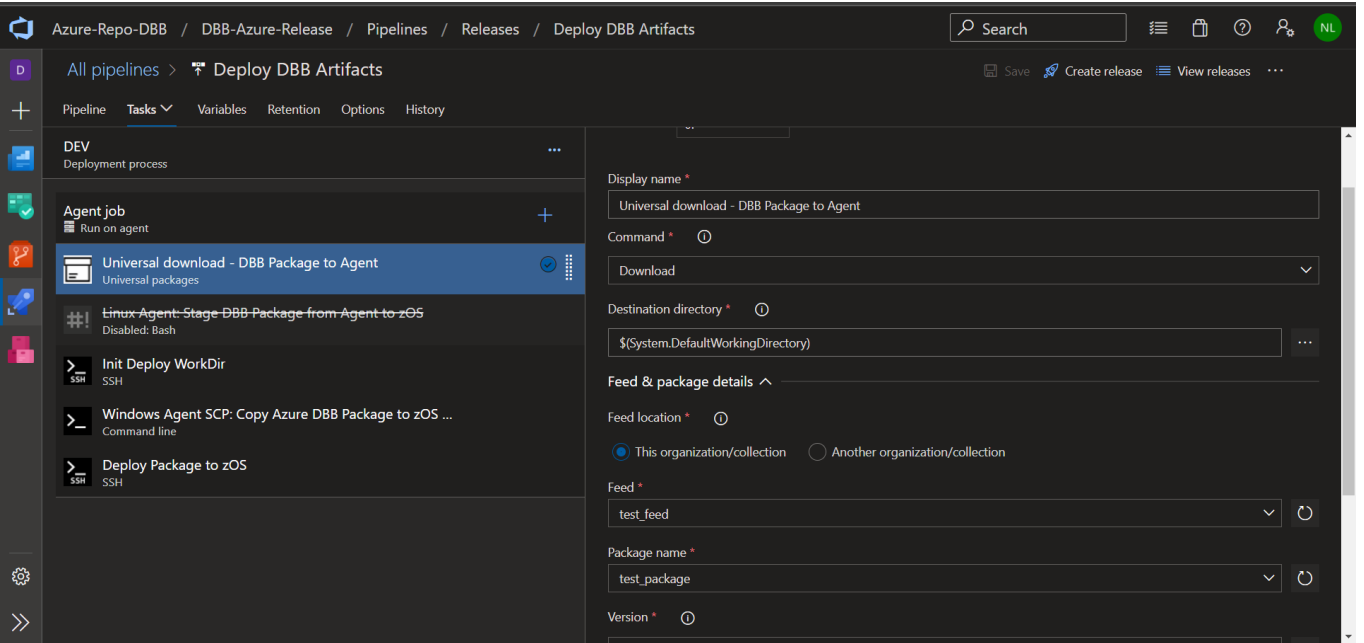
# Release pipeline scripts

Design your release pipeline to use your Feed with the following tasks for a stage called DEV as an example. Refer to Microsoft's documentation on setting up a feed and release pipelines.



## Release-Task: Universal download - DBB Package to Agent

Add the 'Universal packages' task and configure it using your Feed and package names. Use this as the Destination directory `$(System.DefaultWorkingDirectory)`





## Release-Task: Init Deploy WorkDir

### Commands:

```
echo "Building zOS staging area"
mkdir -p $(zOS_DeploymentDir)
```

The screenshot displays the Azure DevOps pipeline interface. On the left, the 'Deployment process' section shows the following status:

- Deploy DBB Artifacts >
- Pipeline Tasks Variables >
- Deployment process: Failed
- Agent job: Succeeded
- Post-deployment gates: Succeeded
- Post-deployment approval: Rejected

The main panel shows the 'Init Deploy WorkDir' task, which is currently running. The task log is as follows:

```
1 |021-02-28T13:11:30.1834648Z ##[section]Starting: Init Deploy WorkDir
2
3 2021-02-28T13:11:30.1993936Z =====
4 2021-02-28T13:11:30.1994163Z Task : SSH
5 2021-02-28T13:11:30.1994355Z Description : Run shell commands or a script on a remote machine using SSH
6 2021-02-28T13:11:30.1994541Z Version : 0.180.0
7 2021-02-28T13:11:30.1994691Z Author : Microsoft Corporation
8 2021-02-28T13:11:30.1994983Z Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/deploy/ssh
9 2021-02-28T13:11:30.1995146Z =====
10 2021-02-28T13:11:31.6806702Z Trying to establish an SSH connection to ***@tvt6031.svl.ibm.com:22
11 2021-02-28T13:11:32.9913161Z Successfully connected.
12 2021-02-28T13:11:32.9914691Z echo "Building zOS staging area"
13 2021-02-28T13:11:33.3956134Z Building zOS staging area
14 2021-02-28T13:11:33.3956442Z
15 2021-02-28T13:11:33.3999565Z pwd
16 2021-02-28T13:11:33.7879604Z /u/**
17 2021-02-28T13:11:33.7880316Z
18 2021-02-28T13:11:33.7893282Z mkdir -p Azure-Workspace/DBB-Azure-Release_DEPLOY_175
19 2021-02-28T13:11:34.3626128Z ##[section]Finishing: Init Deploy WorkDir
```

## Release-Task: Windows Agent SCP: Copy Azure DBB Package to zOS Staging area

### Commands:

```
@echo "Agent Staging Dir:"
dir $(System.DefaultWorkingDirectory)
@echo "Stage Package on Target Host      ->" $(zOS_Deployment_LPAR) " Dir->" $(zOS_DeploymentDir)
@echo "***"
scp $(System.DefaultWorkingDirectory)/* $(zOS_Deployment_LPAR):$(zOS_DeploymentDir)
```

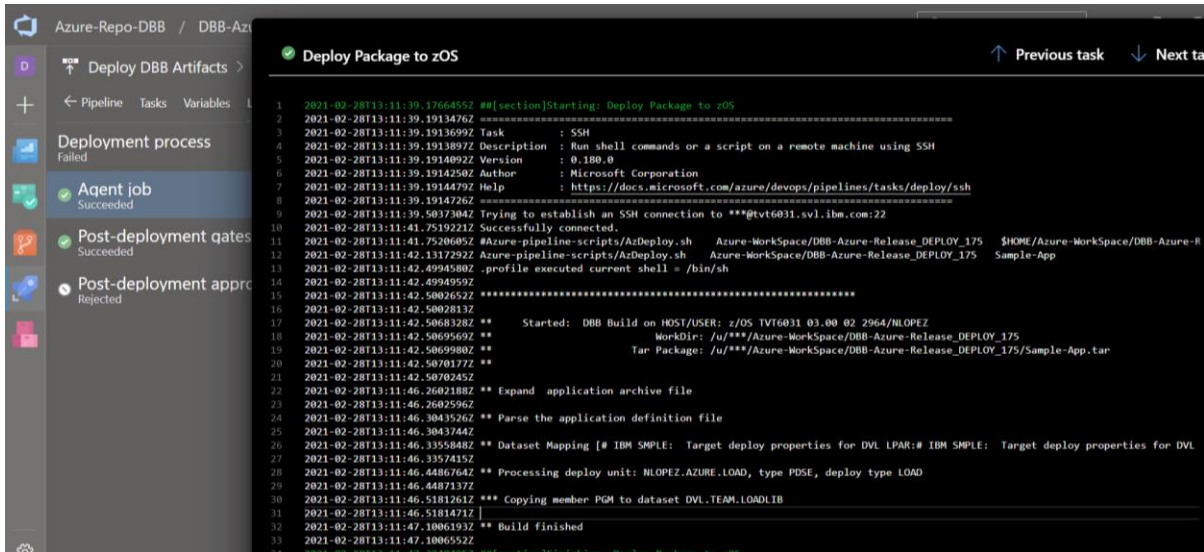
**Windows Agent SCP: Copy Azure DBB Package to zOS Staging area**

1 | 2021-02-28T13:11:34.3758717Z ##[section]Starting: Windows Agent SCP: Copy Azure DBB Package to zOS Staging area  
2 | 2021-02-28T13:11:34.3963274Z =====  
3 | 2021-02-28T13:11:34.3963988Z Task : Command line  
4 | 2021-02-28T13:11:34.3964554Z Description : Run a command line script using Bash on Linux and macOS and cmd.exe on Windows  
5 | 2021-02-28T13:11:34.3965154Z Version : 2.182.0  
6 | 2021-02-28T13:11:34.3965598Z Author : Microsoft Corporation  
7 | 2021-02-28T13:11:34.3966264Z Help : <https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/command-line>  
8 | 2021-02-28T13:11:34.3967036Z =====  
9 | 2021-02-28T13:11:35.7578629Z Generating script.  
10 | 2021-02-28T13:11:35.7987780Z ===== Starting Command Output =====  
11 | 2021-02-28T13:11:35.8294075Z ##[command]"C:\WINDOWS\system32\cmd.exe" /D /E:ON /V:OFF /S /C "CALL "C:\AzureAgent\vsst-agent-win-x64-2.172.2\\_work\\_t  
12 | 2021-02-28T13:11:35.8490418Z "Agent Staging Dir:"  
13 | 2021-02-28T13:11:35.8502400Z Volume in drive C is Windows  
14 | 2021-02-28T13:11:35.8502617Z Volume Serial Number is 6004-BC35  
15 | 2021-02-28T13:11:35.8503941Z  
16 | 2021-02-28T13:11:35.8504338Z Directory of C:\AzureAgent\vsst-agent-win-x64-2.172.2\\_work\r1\A  
17 | 2021-02-28T13:11:35.8504709Z  
18 | 2021-02-28T13:11:35.8504955Z 02/28/2021 08:11 AM <DIR> .  
19 | 2021-02-28T13:11:35.8505385Z 02/28/2021 08:11 AM <DIR> ..  
20 | 2021-02-28T13:11:35.8506178Z 02/28/2021 08:11 AM 40,960 Sample-App.tar  
21 | 2021-02-28T13:11:35.8506741Z 1 File(s) 40,960 bytes  
22 | 2021-02-28T13:11:35.8507216Z 2 Dir(s) 160,353,705,984 bytes free  
23 | 2021-02-28T13:11:35.8513353Z "Stage Package on Target Host ->" \*\*\*@tvt6031.svl.ibm.com " Dir->" Azure-WorkSpace/DBB-Azure-Release\_DEPLOY\_175  
24 | 2021-02-28T13:11:35.8517478Z \*\*\*  
25 | 2021-02-28T13:11:39.1661207Z ##[section]Finishing: Windows Agent SCP: Copy Azure DBB Package to zOS Staging area  
26 |

## Release-Task: Deploy Package to zOS

### Commands:

Azure-pipeline-scripts/AzDeploy.sh     \$(zOS\_DeploymentDir)     \$(MyApp)



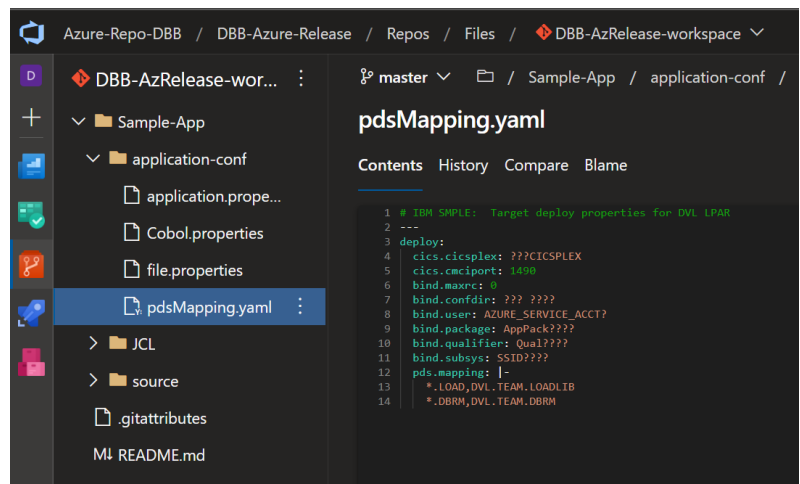
```
1 2021-02-28T13:11:39.1766455Z ##[section]Starting: Deploy Package to zOS
2 2021-02-28T13:11:39.1913476Z *****
3 2021-02-28T13:11:39.1913699Z Task : SSH
4 2021-02-28T13:11:39.1913892Z Description : Run shell commands or a script on a remote machine using SSH
5 2021-02-28T13:11:39.1914092Z Version : 0.180.0
6 2021-02-28T13:11:39.1914250Z Author : Microsoft Corporation
7 2021-02-28T13:11:39.1914479Z Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/deploy/ssh
8 2021-02-28T13:11:39.1914726Z *****
9 2021-02-28T13:11:39.5037304Z Trying to establish an SSH connection to ***@vt6031.svl.ibm.com:22
10 2021-02-28T13:11:41.7519221Z Successfully connected.
11 2021-02-28T13:11:41.7520605Z Azure-pipeline-scripts/AzDeploy.sh Azure-WorkSpace/DBB-Azure-Release_DEPLOY_175 $HOME/Azure-WorkSpace/DBB-Azure-R
12 2021-02-28T13:11:42.1317292Z .profile executed current shell = /bin/sh
13 2021-02-28T13:11:42.4994580Z
14 2021-02-28T13:11:42.4994959Z *****
15 2021-02-28T13:11:42.5002652Z *****
16 2021-02-28T13:11:42.5002813Z Started: DBB Build on HOST/USER: z/OS TVT6031 03.00 02 2964/NLOPEZ
17 2021-02-28T13:11:42.5006320Z WorkDir: /u/***Azure-WorkSpace/DBB-Azure-Release_DEPLOY_175
18 2021-02-28T13:11:42.5009563Z Tar Package: /u/***Azure-WorkSpace/DBB-Azure-Release_DEPLOY_175/Sample-App.tar
19 2021-02-28T13:11:42.5070177Z
20 2021-02-28T13:11:42.5070245Z
21 2021-02-28T13:11:42.5070245Z
22 2021-02-28T13:11:46.2602188Z ** Expand application archive file
23 2021-02-28T13:11:46.2602596Z
24 2021-02-28T13:11:46.3043526Z ** Parse the application definition file
25 2021-02-28T13:11:46.3043744Z
26 2021-02-28T13:11:46.3355848Z ** Dataset Mapping [# IBM SMPLE: Target deploy properties for DVL LPAR:# IBM SMPLE: Target deploy properties for DVL
27 2021-02-28T13:11:46.3357415Z
28 2021-02-28T13:11:46.4486764Z ** Processing deploy unit: NLOPEZ.AZURE.LOAD, type PDSE, deploy type LOAD
29 2021-02-28T13:11:46.4487137Z
30 2021-02-28T13:11:46.5181261Z *** Copying member PGM to dataset DVL.TEAM.LOADLIB
31 2021-02-28T13:11:46.5181471Z
32 2021-02-28T13:11:47.1006193Z ** Build finished
33 2021-02-28T13:11:47.1006552Z
34 2021-02-28T13:11:47.1006552Z
```

The Deploy script uses a hardcoded yaml file named pdsMapping.yaml. It maps the LLQ of a file being deployed to a target PDS. For example, the LLQ in PDS NLOPEZ.TEST.LOAD is mapped to the PDS **DVL.TEAM.LOADLIB**. Ideally these target PDS(s) are existing files used to store Dev, QA or PROD versions of the binaries. This sample also illustrates, but does not implement, DB2 and CICS deployment properties. This could be extended to describe properties across any target environment as needed.

# IBM SMPLE: Target deploy properties for DVL LPAR  
deploy:

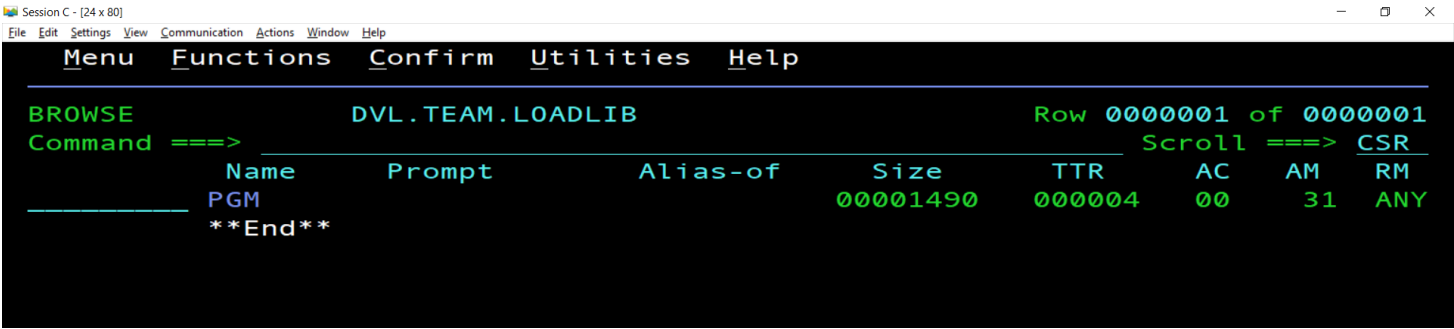
```
cics.cicsplex: ???CICSPLEX
cics.cmciport: 1490
bind.maxrc: 0
bind.confdir: ??? ???
bind.user: AZURE_SERVICE_ACCT?
bind.package: AppPack????
bind.qualifier: Qual????
bind.subsys: SSID????
```

```
pds.mapping: |-
  *.LOAD,DVL.TEAM.LOADLIB
  *.DBRM,DVL.TEAM.DBRM
```



```
1 # IBM SMPLE: Target deploy properties for DVL LPAR
2 ---
3 deploy:
4   cics.cicsplex: ???CICSPLEX
5   cics.cmciport: 1490
6   bind.maxrc: 0
7   bind.confdir: ??? ???
8   bind.user: AZURE_SERVICE_ACCT?
9   bind.package: AppPack????
10  bind.qualifier: Qual????
11  bind.subsys: SSID????
12  pds.mapping: |-
13    *.LOAD,DVL.TEAM.LOADLIB
14    *.DBRM,DVL.TEAM.DBRM
```

The result, in this example, was to deploy the sample 'PGM' built by DBB to the DVL.TEAM.LOADLIB in a DEV LPAR.



## Sample Script Repo (Private)

<https://github.com/nlopez59/Azure-pipeline-scripts>

