

# Activity 7 - Ring-based deployment

## Steps

Let's look at how a release pipeline can stage features using ring-based deployments.

When I have a new feature, I might want to release it to a few users first, just in case something goes wrong.

I could do it in authenticated systems by having those users as members of a security group and letting members of that group use the new features.

However, on a public website, I might not have logged-in users. Instead, I might want to direct a small percentage of the traffic to use the new features.

Let's see how that's configured.

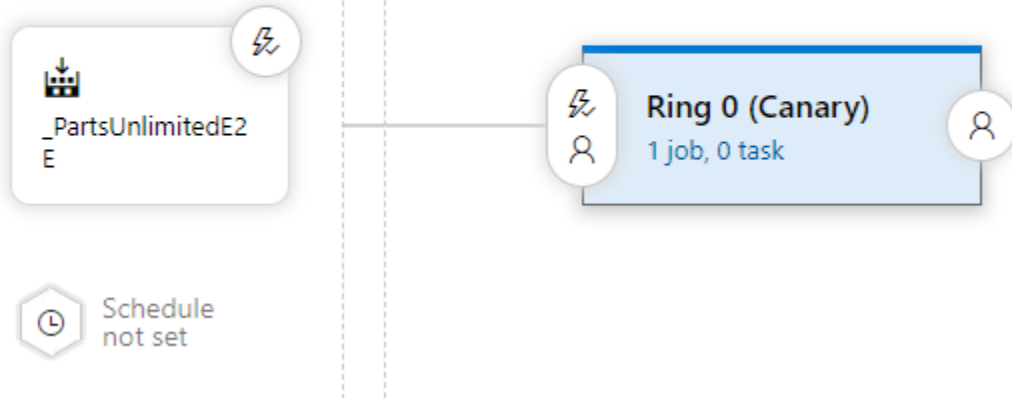
We'll create a new release pipeline that isn't triggered by code changes but manually when we slowly release a new feature.

We start by assuming that a new feature has already been deployed to the Green site (the staging slot).

1. In the main menu for the **Parts Unlimited** project, click **Pipelines**, then click **Release**, click **+New**, then click **New release pipeline**.
2. When prompted to select a template, click **Empty job** from the top of the pane.
3. Click on the **Stage 1** stage and rename it to **Ring 0 (Canary)**.

Artifacts | + Add

Stages | + Add ▾




4. Hover over the **New release pipeline** name at the top of the page, and when a pencil appears, click it, and change the pipeline name to **Ring-based Deployment**.

All pipelines > 📌 Ring-based Deployment

Pipeline Tasks ▾ Variables Retention Options History

5. Select the **Ring 0 (Canary)** stage from the Tasks drop-down list. Click the + to add a new task, and from the list of tasks, hover over **Azure CLI** when the **Add** button appears, click it, then click to select the **Azure CLI** task in the task list for the stage.

Azure CLI ⓘ

 View YAML  Remove

Task version 1.\* ▼

Display name \*

Azure CLI

Azure subscription \* ⓘ | [Manage](#)



ⓘ This setting is required.

Script Location \* ⓘ

Script path



Script Path \* ⓘ




This setting is required.

6. In the **Azure CLI** settings pane, select your **Azure subscription**, set **Script Location** to **Inline script**, set the **Inline Script** to the following, then click **Save** and **OK**.

```
az webapp traffic-routing set --resource-group $(ResourceGroupName) --name $(WebsiteName) --distribution staging=10
```

This distribution will cause 10% of the web traffic to be sent to the new feature Site (currently the staging slot).

7. From the menu above the task list, click **Variables**. Create two new variables as shown. (Make sure to use your correct website name).

Name	Value		Scope
ResourceGroupName	ASPDOTNET		Release
WebsiteName	pule2eb3100890		Release

8. From the menu above the variables, click **Pipeline** to return to editing the pipeline. Hover over the **Ring 0 (Canary)** stage and click the **Clone** icon when it appears. Select the new stage and rename it to **Ring 1 (Early Adopters)**.

Stages | + Add ▾



9. Select the **Ring 1 (Early Adopters)** stage from the Tasks drop-down list and select the **Azure CLI** task. Modify the script by changing the value from **10** to **30** to cause 30% of the traffic to go to the new feature site.

Inline Script \* ⓘ

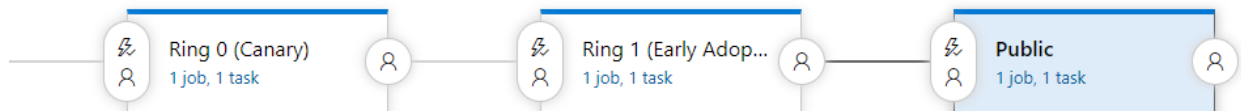
```
az webapp traffic-routing set --resource-group $(ResourceGroupName) --name $(WebsiteName) --distribution staging=30
```

```
az webapp traffic-routing set --resource-group $(ResourceGroupName) --name $(WebsiteName) --distribution staging=30
```

It allows us to move the new feature into broader distribution if it works ok in smaller users.

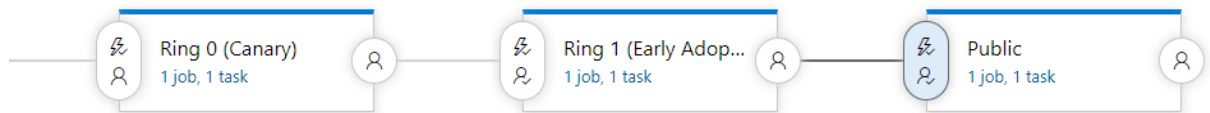
10. From the menu above the tasks, click **Pipeline** to return to editing the release pipeline. Hover over the **Ring 1 (Early Adopters)** stage and when the **Clone** icon appears, click it. Click to select the new stage and rename it to **Public**. Click **Save** and **OK**.

Stages | + Add ▾



11. Click the **Pre-deployment conditions** icon for the **Ring 1 (Early Adopters)** stage and add yourself as a pre-deployment approver. Do the same for the **Public** stage—Click **Save** and **OK**.

Stages | + Add ▾



The first step in releasing the new code to the public is to swap the new feature site (that is, the staging site) with the production so that production is now running the new code.

12. From the **Tasks** drop-down list, select the **Public** stage. Select the **Azure CLI** task, change the **Display name** to **Swap sites** and change the **Inline Script** to the following command:

```
az webapp deployment slot swap -g $(ResourceGroupName) -n $(WebsiteName) --slot staging --target-slot production
```

Inline Script \* ⓘ

```
az webapp deployment slot swap -g $(ResourceGroupName) -n $(WebsiteName) --slot staging --target-slot production
```

```
az webapp deployment slot swap -g $(ResourceGroupName) -n $(WebsiteName) --slot staging --target-slot production
```

Next, we need to remove any traffic from the staging site.

13. Right-click the **Swap sites** task and click **Clone task(s)**. Select the **Swap sites copy** task, change its **Display name** to **Stop staging traffic**, and set the **Inline Script** to the following:

```
az webapp traffic-routing set --resource-group $(ResourceGroupName) --name $(WebsiteName) --distribution staging=0
```

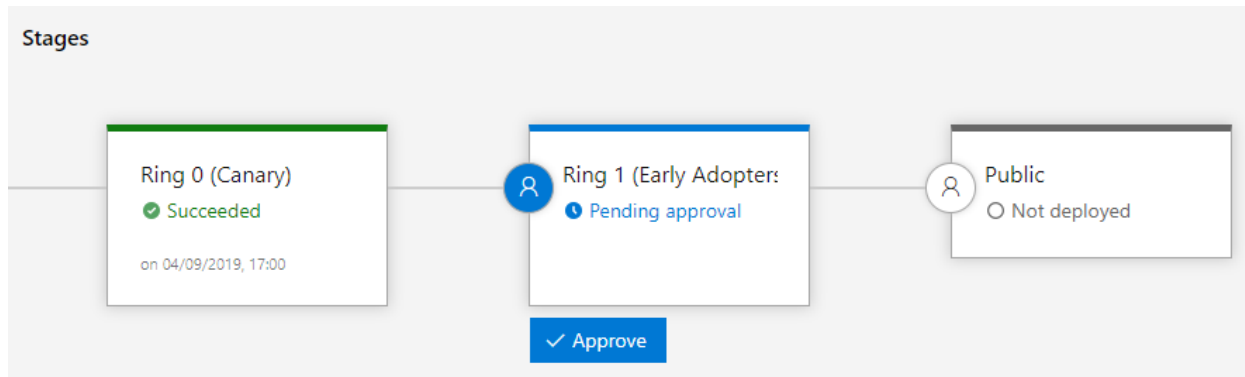
1. Click **Save**, then **OK** to save your work.
2. Click **Create release** and **Create** to start a release. Click the release link to see the progress.

## All pipelines > Ring-based Deployment

Release [Release-1](#) has been created

Pipeline Tasks Variables Retention Options History

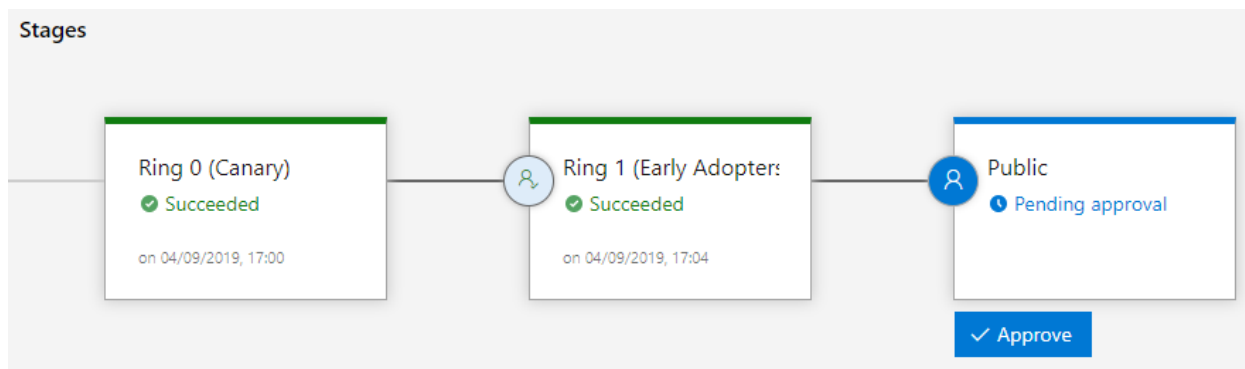
Wait until Ring 0 (Canary) has succeeded.



Currently, 10% of the traffic will go to the new feature site.

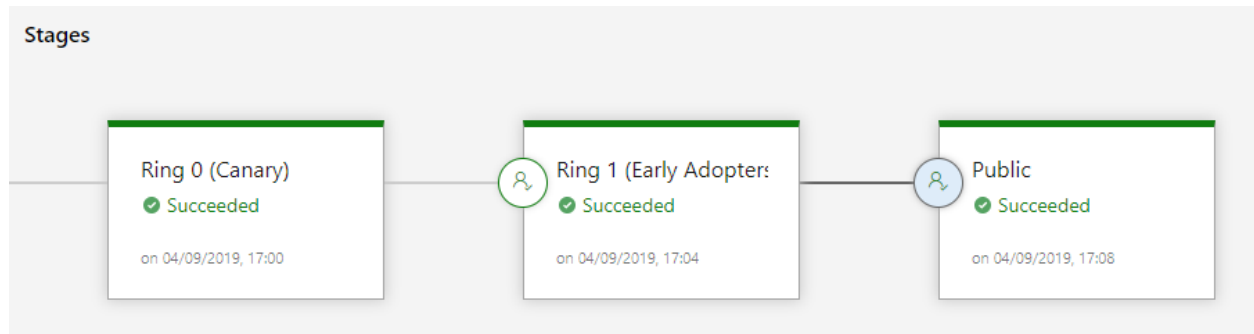
- Click **Approve** on the **Ring 1 (Early Adopters)** stage, and then **Approve**.

When this stage completes, 30% of the traffic will go to the early adopters in ring 1.



- Click **Approve** on the **Public** stage, and then **Approve**.

When this stage completes, all the traffic will go to the swapped production site, running the new code.



The new feature has been fully deployed.