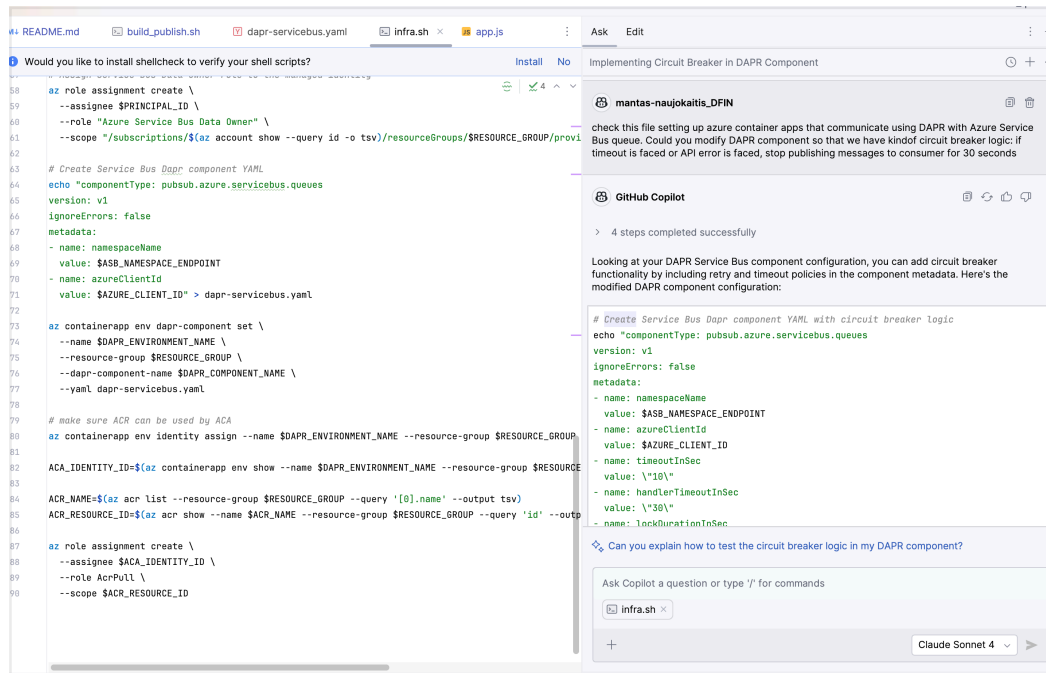


# GenAI usage

I used GitHub coPilot (instead of Azure openAI), as it provides some of the same LLMs, allows setting context by referencing files and setting up context through by providing extensive details in message.

Copilot suggested me how to set up circuit breaker in my infra.sh script:



This is the output:

```
# Create Service Bus Dapr component YAML with circuit breaker logic
echo "componentType: pubsub.azure.servicebus.queues
version: v1
ignoreErrors: false
metadata:
- name: namespaceName
  value: $ASB_NAMESPACE_ENDPOINT
- name: azureClientId
  value: $AZURE_CLIENT_ID
- name: timeoutInSec
  value: \"10\"
- name: handlerTimeoutInSec
  value: \"30\"
- name: lockDurationInSec
  value: \"30\"
- name: maxDeliveryCount
  value: \"3\"
- name: defaultMessageTimeToLiveInSec
  value: \"300\"
- name: autoDeleteOnIdleInSec
  value: \"3600\""
```

- name: disableEntityManagement  
value: \"false\"
- name: maxConcurrentHandlers  
value: \"1\"
- name: prefetchCount  
value: \"0\"
- name: maxRetriableErrorsPerSec  
value: \"10\"
- name: maxActiveMessages  
value: \"1000\" > dapr-servicebus.yaml

Copilot helped to resolve the problem no one was consuming messages as app got scaled down automatically (ACA scales app down if there are no min replicas set and/or if there is not traffic to ingress enabled app):

The screenshot shows the GitHub Copilot chat interface within a code editor. The chat window is titled "Troubleshooting Dapr Communication in Azure Container Apps". It displays a response from Copilot regarding a deployment script issue. The response states that the "order-service-app" likely lacks ingress configuration, causing it to scale to zero and preventing message consumption from the service bus. It provides a list of key issues and solutions, including the need for ingress configuration and scaling rules. A specific solution is provided: "Option 1: Add ingress to order-service (if it needs external access):", followed by a terminal command to create the container app with the necessary environment variables and image. Below the chat window, there is a prompt asking for best practices for configuring Dapr with Azure Container Apps, and a text input field with the placeholder "Ask Copilot a question or type '/' for commands". The input field contains the text "deploy.sh". The interface also shows a "Claude Sonnet 4" model selector and a "Send" button.

Ask Edit

Troubleshooting Dapr Communication in Azure Container Apps

GitHub Copilot

> 4 steps completed successfully

Looking at your deployment script, the issue is likely that your order-service-app doesn't have ingress configured, which means it has no external traffic to keep it scaled up. When Azure Container Apps scale to zero due to inactivity, they won't be able to consume messages from the service bus.

Here are the key issues and solutions:

**Problem**

- The `order-service-app` has no ingress configuration, so it scales to zero when idle
- Scaled-to-zero containers cannot consume Dapr pub/sub messages
- You need to configure scaling rules or ingress to keep the consumer active

**Solutions**

**Option 1: Add ingress to order-service (if it needs external access):**

```
az containerapp create \
  --name order-service-app \
  --resource-group $RESOURCE_GROUP \
  --environment $DAPR_ENVIRONMENT_NAME \
  --registry-server $ACR_LOGIN_SERVER \
  --image $ACR_LOGIN_SERVER/orderservice:latest \
```

What are the best practices for configuring Dapr with Azure Container Apps?

Ask Copilot a question or type '/' for commands

deploy.sh

Claude Sonnet 4