

中国·北京

.NET Conf China 2023

How .NET runs on Azure App Service

Edi Wang





Edi Wang

Azure & .NET Developer

- 35 y/o Senior Software Engineer
- Microsoft MVP (Azure) since 2017
- Author of project Moonglade, an ASP.NET Core blog system runs on Azure, used by many Microsoft employees
- Creator and former contributor of Windows App “Character Map UWP” , reaching over 40M global users
- Operating the WeChat video account introducing technologies like Azure, committed to no advertisements, no course sales.



PaaS Services to run .NET on Azure

- Azure App Service
- Azure Kubernetes Service (AKS)
- Azure Container Instance
- Azure Container Apps



Azure App Service


Quickly and easily create enterprise-ready web and mobile apps for any platform or device, and deploy them on a scalable and reliable cloud infrastructure



Publish *

Code Docker Container Static Web App

Runtime stack *

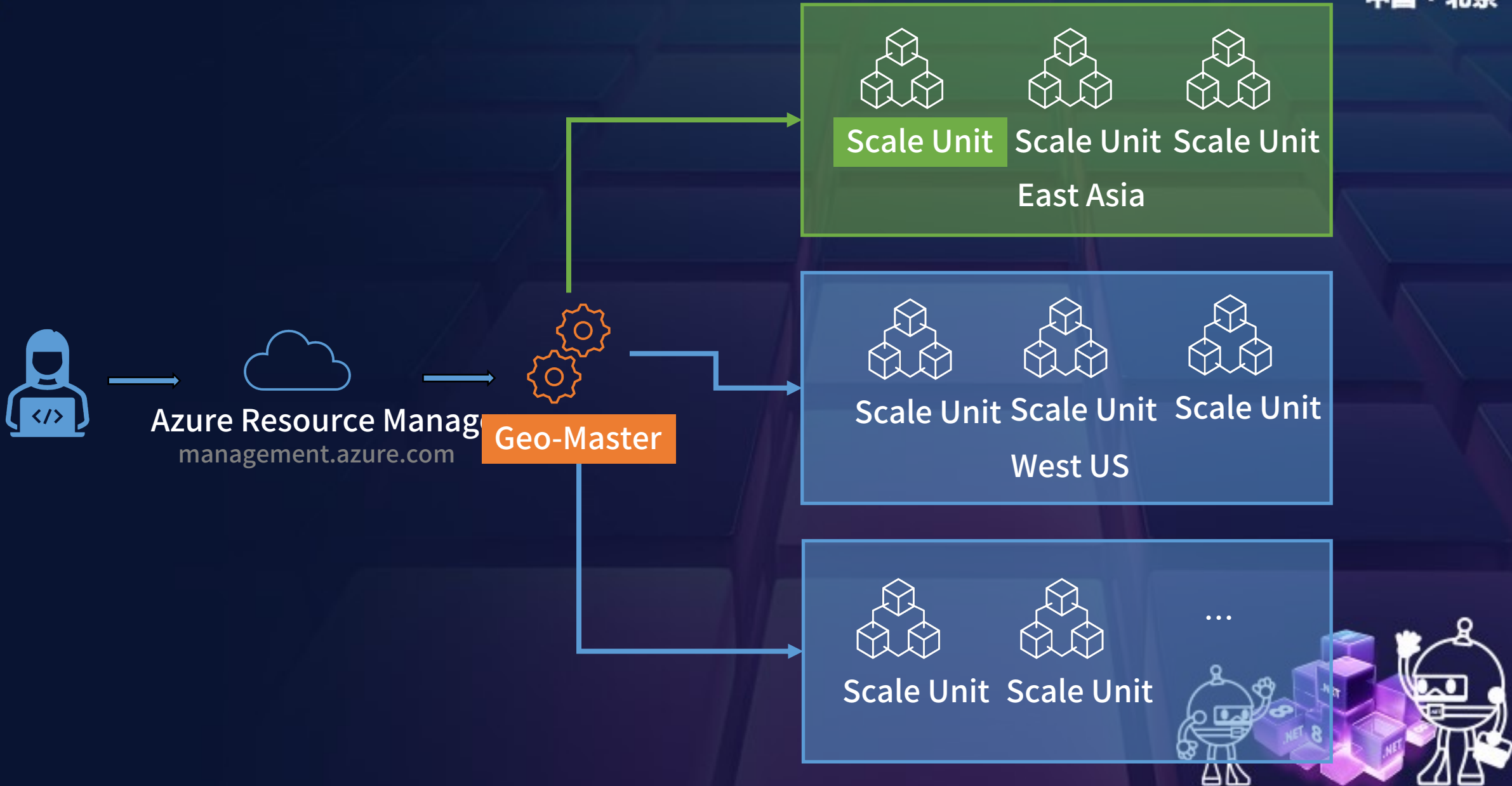
.NET 8 (LTS) (Early Access) 

[Learn more about Early Access stacks on App Service](#)

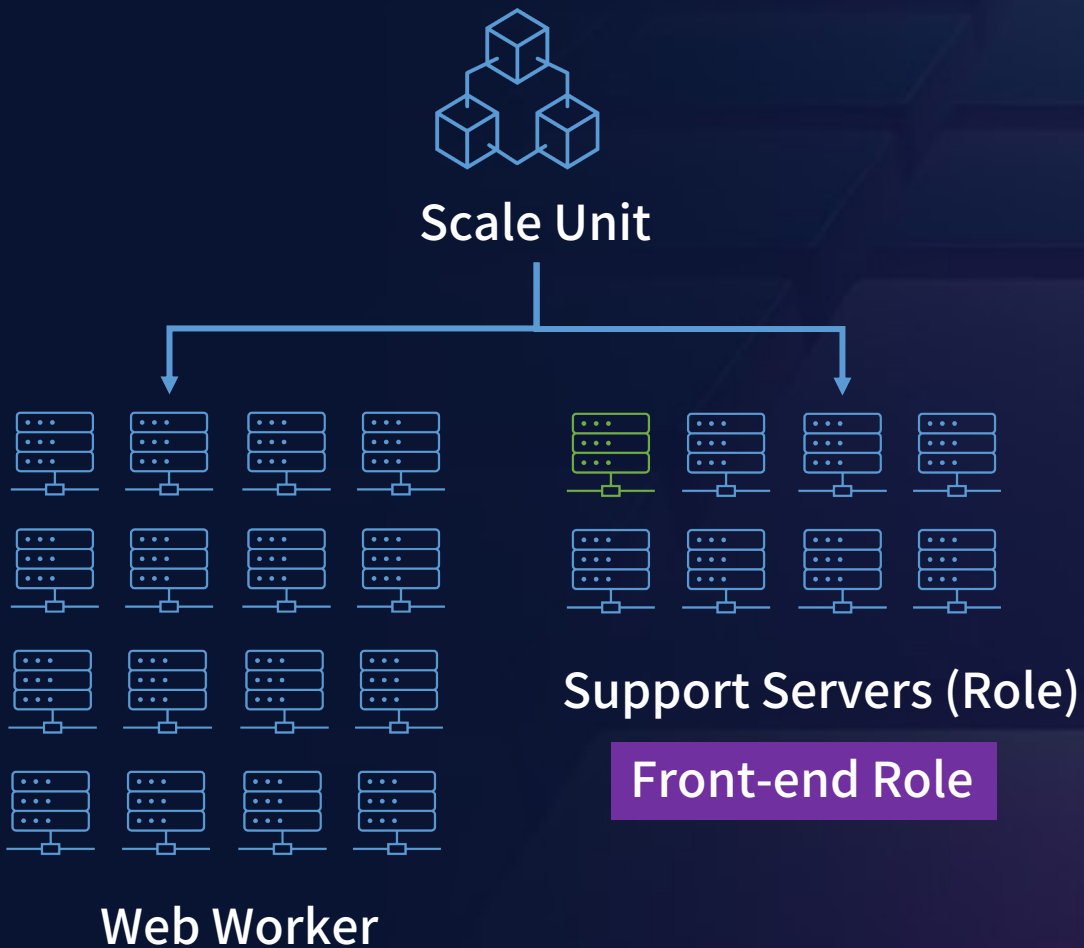
Operating System *

Linux Windows





Front-End Role

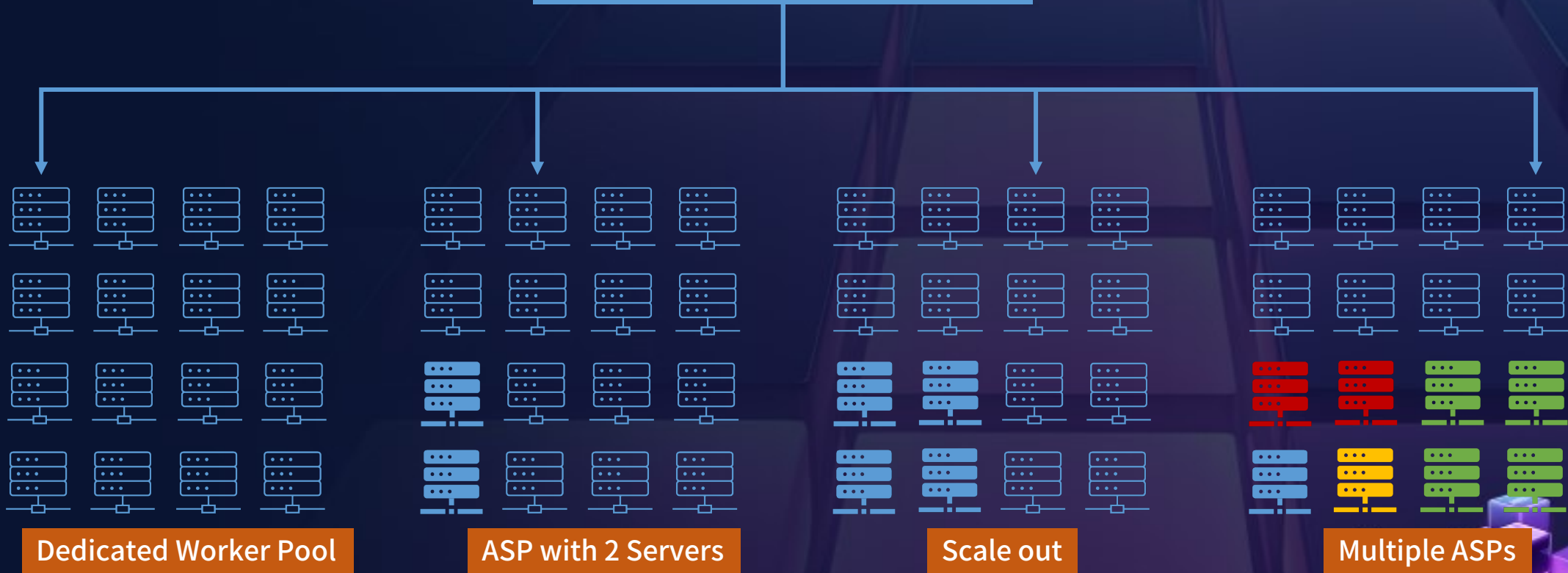


Layer 7 load balancer (IIS -> YARP)
Round-robin (RR) algorithm



Web Worker









App Service Plan



Windows Plan + Code Deployment

Edi Wang

    **Edi Wang** (Edi.Wang@outlook.com) 

-  Posts
-  Pages
-  Categories
-  Comments
-  Tags
-  Menus
-  Friend links
-  Pingbacks

Moonglade Configuration

Version	14.0.1 (7ba0fe)
Environment	Production
Editor	Html
Forwarded Headers	true: X-Azure-ClientIP
Client IP Address	20.205.37.35

Server Information

System	Windows Server 2016 Datacenter 10.0.14393.6085 (64-bit)
Directory	C:\home\site\wwwroot
Machine Name	pd1sdwk0000LJ
User Name	IIS APPPOOL\moonglade-ediwang-ea
.NET Version	8.0.0
ASP.NET Core Environment	Production
Worker Process	[5672] w3wp (64-bit)
Start Time	11/20/2023 2:54:59 AM (UTC)
Memory	297 MB
Docker Container	False



EXPLORE

- WORKING FILES
 - web.config
- WWWROOT
 - Moonglade.Theme.dll
 - Moonglade.Theme.pdb
 - Moonglade.Utils.dll
 - Moonglade.Utils.pdb
 - Moonglade.Web.dll
 - Moonglade.Web.pdb
 - Moonglade.Web.deps....
 - Moonglade.Web.runti...
 - MySQLConnector.dll
 - Newtonsoft.Json.dll
 - Npgsql.dll
 - Npgsql.EntityFramew...
 - NUglify.dll
 - Pomelo.EntityFramew...
 - SixLabors.Fonts.dll
 - SixLabors.ImageSharp....
 - SixLabors.ImageSharp...
 - Spectre.Console.dll
 - System.Configuration....
 - System.Diagnostics.Pe...

web.config

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <configuration>
3   <location path="." inheritInChildApplications="false">
4     <system.webServer>
5       <handlers>
6         <add name="aspNetCore" path="*" verb="*" modules="AspNetCoreModuleV2" resourceType="Unspecified" />
7       </handlers>
8       <aspNetCore processPath="dotnet" arguments=".\Moonglade.Web.dll" stdoutLogEnabled="false" stdoutLogFile=".\logs\stdout" hostingModel="inprocess" />
9       <staticContent>
10        <remove fileExtension=".woff2"/>
11        <remove fileExtension=".woff"/>
12        <mimeMap fileExtension=".woff2" mimeType="font/woff2"/>
13        <mimeMap fileExtension=".woff" mimeType="font/woff"/>
14      </staticContent>
15      <security>
16        <requestFiltering removeServerHeader="true"/>
17      </security>
18      <httpProtocol>
19        <customHeaders>
20          <remove name="X-Powered-By"/>
21          <add name="x-content-type-options" value="nosniff"/>
22          <add name="x-xss-protection" value="1"/>
23          <add name="content-security-policy"
24            value="frame-ancestors 'self'; form-action 'self'; script-src 'self' 'unsafe-inline' 'unsafe-eval' https://cdnjs.cloudflare.com https://*.vo.ms
25            worker-src blob: data: https://cdnjs.cloudflare.com"/>
26        </customHeaders>
27      </httpProtocol>
28      <rewrite>
29        <rules>
```



- Dashboard >
- moonglade-ediwang-ea Web App
- Search
- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Microsoft Defender for Cloud
- Events (preview)
- Deployment
 - Deployment slots
 - Deployment Center
- Settings
 - Configuration
 - Authentication
 - Application Insights
 - Identity
 - Backups
 - Custom domains

Browse Stop Swap Restart Delete Refresh Download publish profile Reset publish profile Share to mobile Send us your feedback

Click here to access Application Insights for monitoring and profiling for your app.

JSON View

Essentials

Resource group (move)	: ediwang-app-rsg-ea	Default domain	: moonglade-ediwang-ea.azurewebsites.net
Status	: Running	App Service Plan	: ediwang-webplan-ea (P1v2: 1)
Location (move)	: East Asia	Operating System	: Windows
Subscription (move)	: Super Rich Sponsorship	Health Check	: Not Configured
Subscription ID	: 22526f64-4806-4eb9-9ca5-936d798766b2		

Tags (edit) : Add tags

Properties Monitoring Logs Capabilities Notifications Recommendations

Web app

Name	moonglade-ediwang-ea
Publishing model	Code
Runtime Stack	Dotnet - v8.0

Domains

Default domain	moonglade-ediwang-ea.azurewebsites.net
Custom domain	edi.wang

Extra Runtime

```
D:\home\site\wwwroot>java
Usage: java [-options] class [args...]
        (to execute a class)
or java [-options] -jar jarfile [args...]
        (to execute a jar file)
where options include:
-d32    use a 32-bit data model if available
-d64    use a 64-bit data model if available
-server to select the "server" VM
        The default VM is server.

-cp <class search path of directories and zip/jar files>
-classpath <class search path of directories and zip/jar files>
        A ; separated list of directories, JAR archives,
        and ZIP archives to search for class files.
-D<name>=<value>
        set a system property
-verbose:[class|gc|jni]
        enable verbose output
```

```
D:\home\site\wwwroot>node -v
v0.10.40
```

```
D:\home\site\wwwroot>python --version
Python 2.7.15
```



Linux Plan + Code Deployment

```
var builder = WebApplication.CreateBuilder(args);  
var app = builder.Build();  
  
var isInDocker = Environment.GetEnvironmentVariable("DOTNET_RUNNING_IN_CONTAINER") ?? false.ToString();  
  
app.MapGet("/", () => $"DOTNET_RUNNING_IN_CONTAINER: {isInDocker}");  
  
app.Run();
```



- Dashboard >
- demo996 Web App
- Search
- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Microsoft Defender for Cloud
- Events (preview)
- Deployment
 - Deployment slots
 - Deployment Center
- Settings
 - Configuration
 - Authentication
 - Application Insights
 - Identity
 - Backups
 - Custom domains

[Browse](#) [Stop](#) [Swap](#) [Restart](#) [Delete](#) [Refresh](#) [Download publish profile](#) [Reset publish profile](#) [Share to mobile](#) [Send us your feedback](#)

Essentials

Resource group (move) : [netconfdemo-group](#) Default domain : [demo996.azurewebsites.net](#)

Status : Running App Service Plan : [netconf-linux-plan \(B2: 1\)](#)

Location (move) : East Asia Operating System : Linux

Subscription (move) : [DevTest](#) Health Check : [Not Configured](#)

Subscription ID : [afc3200d-f6b3-46a9-94e8-88181e62fd49](#)

Tags (edit) : [Add tags](#)

[JSON View](#)

Properties Monitoring Logs Capabilities Notifications Recommendations

Web app

Name	demo996
Publishing model	Code
Runtime Stack	Dotnetcore - 8.0

Domains

Default domain	demo996.azurewebsites.net
Custom domain	Add custom domain

Hosting

Download Docker Logs

Azure App Service Environment SSH Bash Log stream

Environment

Build	1.0.0.7 (e59ed50ca2)
Site up time	00.00:04:02
Site folder	/home
Temp folder	/tmp/

REST API

(works best when using a JSON viewer extension)

- App Settings
- Deployments
- Source control info
- Files
- Current Docker logs (Download as zip)**

Browse Directory

- Deployment Logs
- Site wwwroot

edi.wang

demo996 | Advanced Tools ☆ ...
Web App

Search

Change App Service plan

Development Tools

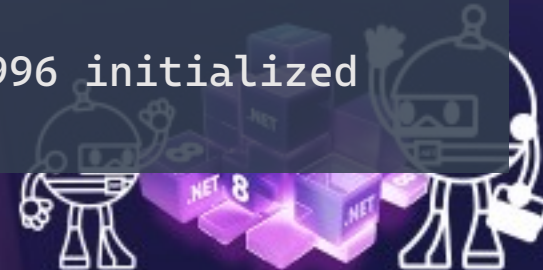
- SSH
- Advanced Tools**

Advanced Tools provides a collect
Go →



The Container Image

```
2023-11-23T13:24:42.649Z INFO - Initiating warmup request to container demo996_2_2bb3ad40 for
site demo996
2023-11-23T13:24:59.646Z INFO - Waiting for response to warmup request for container
demo996_2_2bb3ad40. Elapsed time = 16.996715 sec
2023-11-23T13:25:23.846Z INFO - Container demo996_2_2bb3ad40 for site demo996 initialized
successfully and is ready to serve requests.
2023-11-23T13:25:24.225Z INFO - Starting container for site
2023-11-23T13:25:24.232Z INFO - docker run -d --expose=8080 --name demo996_3_063c727e -e
WEBSITE_SITE_NAME=demo996 -e WEBSITE_AUTH_ENABLED=False -e WEBSITE_ROLE_INSTANCE_ID=0 -e
WEBSITE_HOSTNAME=demo996.azurewebsites.net -e
WEBSITE_INSTANCE_ID=dc7a138dfcb63bce17254cc4321cf14c9178fc6ae4c339b927686049d314ddf6 -e
WEBSITE_USE_DIAGNOSTIC_SERVER=True appsvc/dotnetcore:8.0_20231114.3.tuxprod dotnet demo996.dll
2023-11-23T13:25:24.234Z INFO - Logging is not enabled for this container.
Please use https://aka.ms/linux-diagnostics to enable logging to see container logs here.
2023-11-23T13:25:27.249Z INFO - Initiating warmup request to container demo996_3_063c727e for
site demo996
2023-11-23T13:25:45.593Z INFO - Container demo996_3_063c727e for site demo996 initialized
successfully and is ready to serve requests.
```



edi@curryfish: ~\$

Exploring the Container Image

```
docker inspect mcr.microsoft.com/appsvc/dotnetcore:8.0_20231114.3.tuxprod
```

```
"WorkingDir": "/home/site/wwwroot",  
"Entrypoint": [  
  "/bin/init_container.sh"  
],
```

```
"Env": [  
  "PATH=/opt/dotnetcore-  
tools:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/home/site/wwwroot",  
  "ASPNETCORE_URLS=",  
  "DOTNET_RUNNING_IN_CONTAINER=true",  
  "USER_DOTNET_AI_VERSION=2.8.42",  
  "DOTNET_VERSION=6.0",  
  "ASPNETCORE_LOGGING__CONSOLE__DISABLECOLORS=true",  
  "PATH_CA_CERTIFICATE=/etc/ssl/certs/ca-certificate.crt",  
  "PORT=8080",  
  "SSH_PORT=2222",  
  "ASPNETCORE_FORWARDEDHEADERS_ENABLED=true",  
  "HOME=/home"  
]
```



edi@curryfish:~\$

Oryx: Generate startup.sh

```
docker run -it --rm --entrypoint "bash"  
mcr.microsoft.com/appsvc/dotnetcore:8.0_20231114.3.tuxprod
```

```
cat /bin/init_container.sh
```

```
startupCommandPath="/opt/startup/startup.sh"
```

```
source /bin/generate_and_execute_startup_script.sh
```

```
#!/bin/bash
```

```
oryxArgs="create-script -appPath $appPath -output $startupCommandPath -  
defaultAppFilePath $defaultAppPath \  
-bindPort $PORT -bindPort2 '$HTTP20_ONLY_PORT' -userStartupCommand  
'$userStartupCommand' $runFromPathArg"
```

```
echo "Running oryx $oryxArgs"  
eval oryx $oryxArgs  
exec $startupCommandPath
```



Dashboard > demo996 Web App

Search

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Microsoft Defender for Cloud
- Events (preview)

Deployment

- Deployment slots
- Deployment Center

Settings

- Configuration
- Authentication
- Application Insights
- Identity
- Backups
- Custom domains

[Browse](#) [Stop](#) [Swap](#) [Restart](#) [Delete](#) [Refresh](#) [Download publish profile](#) [Reset publish profile](#) [Share to mobile](#) [Send us your feedback](#)

Essentials

Resource group (move) : [netconfdemo-group](#) Default domain : [demo996.azurewebsites.net](#)

Status : Running App Service Plan : [netconf-linux-plan](#)

Location (move) : East Asia Operating System : Linux

Subscription (move) : [DevTest](#) Health Check : [Not Configured](#)

Subscription ID : [afc3200d-f6b3-46a9-94e8-88181e62fd49](#)

Tags (edit) : [Add tags](#)

[JSON View](#)

Properties Monitoring Logs Capabilities Notifications Recommendations

Web app

Name	demo996
Publishing model	Code
Runtime Stack	Dotnetcore - 8.0

Domains

Default domain	demo996.azurewebsites.net
Custom domain	Add custom domain

Startup.sh

```
root@51f93592d614:/opt/startup# cat startup.sh
#!/bin/bash
set -e

export PORT=8080

export ASPNETCORE_URLS=http://*:$PORT

if [ ! -z "$PORT2" ]; then
    export Kestrel__Endpoints__Http2__Url=http://*:$PORT2
    export Kestrel__Endpoints__Http2__Protocols=Http2
    export Kestrel__Endpoints__Http1__Url=http://*:$PORT
    export Kestrel__Endpoints__Http1__Protocols=Http1
fi

echo Running user provided startup command...
cd "/home/site/wwwroot"
dotnet demo996.dll
```



Last login: Fri Dec 8 12:27:25 2023 from 169.254.130.3



Documentation: <http://aka.ms/webapp-linux>
Dotnet quickstart: <https://aka.ms/dotnet-qs>
ASP .NETCore Version: 8.0.0
Note: Any data outside '/home' is not persisted
root@57edfbab84ba:~/site/wwwroot#

Application Files

```
Documentation: http://aka.ms/webapp-linux
Dotnet quickstart: https://aka.ms/dotnet-qs
ASP .NETCore Version: 8.0.0
Note: Any data outside '/home' is not persisted
root@51f93592d614:~/site/wwwroot# ls
root@51f93592d614:~/site/wwwroot# cd ~
root@51f93592d614:~# cd site/wwwroot
root@51f93592d614:~/site/wwwroot# ls -l
total 102
-rwxrwxrwx 1 nobody nogroup 127 Nov 23 21:22 appsettings.Development.json
-rwxrwxrwx 1 nobody nogroup 151 Nov 23 21:22 appsettings.json
-rwxrwxrwx 1 nobody nogroup 72520 Nov 23 21:23 demo996
-rwxrwxrwx 1 nobody nogroup 470 Nov 23 21:23 demo996.deps.json
-rwxrwxrwx 1 nobody nogroup 4608 Nov 23 21:23 demo996.dll
-rwxrwxrwx 1 nobody nogroup 20456 Nov 23 21:23 demo996.pdb
-rwxrwxrwx 1 nobody nogroup 488 Nov 23 21:23 demo996.runtimeconfig.json
root@51f93592d614:~/site/wwwroot#
```



.NET Tools and Runtime

```
root@55a2f2c9e303:/opt/dotnetcore-tools# ls -la
total 856
drwxr-xr-x 3 root root 4096 Jul 21 01:34 .
drwxr-xr-x 1 root root 4096 Jul 26 11:53 ..
drwxr-xr-x 9 root root 4096 Jul 21 01:34 .store
-rwxr-xr-x 1 root root 142840 Jul 21 01:34 dotnet-counters
-rwxr-xr-x 1 root root 142840 Jul 21 01:33 dotnet-dump
-rwxr-xr-x 1 root root 142840 Jul 21 01:34 dotnet-gcdump
-rwxr--r-- 1 root root 142840 Jul 6 20:17 dotnet-monitor
-rwxr-xr-x 1 root root 142840 Jul 21 01:33 dotnet-sos
-rwxr-xr-x 1 root root 142840 Jul 21 01:33 dotnet-trace
```

```
root@55a2f2c9e303:/usr/share/dotnet# ls -la
total 244
drwxrwxr-x 4 root root 4096 Jun 20 19:39 .
drwxr-xr-x 1 root root 4096 Jul 26 11:53 ..
-rw-rw-r-- 1 root root 1116 Jun 20 19:31 LICENSE.txt
-rw-rw-r-- 1 root root 78479 Jun 20 19:31 ThirdPartyNotices.txt
-rwxr-xr-x 1 root root 141760 Jun 20 19:33 dotnet
drwxrwxr-x 3 root root 4096 Jun 20 19:39 host
drwxrwxr-x 4 root root 4096 Jul 21 01:34 shared
```



Linux Plan + Docker

```
2023-11-26T07:02:58.648Z INFO - Starting container for site
2023-11-26T07:02:58.648Z INFO - docker run -d --expose=8080 --name moonglade996_3_606b8d73 -e
WEBSITE_USE_DIAGNOSTIC_SERVER=false -e WEBSITES_ENABLE_APP_SERVICE_STORAGE=false -e
WEBSITE_SITE_NAME=moonglade996 -e WEBSITE_AUTH_ENABLED=False -e PORT=8080 -e
WEBSITE_ROLE_INSTANCE_ID=0 -e WEBSITE_HOSTNAME=moonglade996.azurewebsites.net -e
WEBSITE_INSTANCE_ID=54377a4fbcca99061eb4bfef866d1be5a35314cee3d940f62291ce669db1da6a
ediwang/moonglade
2023-11-26T07:02:58.648Z INFO - Logging is not enabled for this container.
Please use https://aka.ms/linux-diagnostics to enable logging to see container logs here.
2023-11-26T07:03:00.577Z INFO - Initiating warmup request to container moonglade996_3_606b8d73
for site moonglade996
2023-11-26T07:03:17.392Z INFO - Container moonglade996_3_606b8d73 for site moonglade996
initialized successfully and is ready to serve requests.
```

PORT

80

8080

443





```
FROM mcr.microsoft.com/dotnet/aspnet:8.0 AS base

USER app

WORKDIR /app
EXPOSE 8080
EXPOSE 8081

FROM mcr.microsoft.com/dotnet/sdk:8.0 AS build
ARG BUILD_CONFIGURATION=Release
WORKDIR /src

# Auto copy to prevent 996
COPY ./src/**/*.csproj ./
RUN for file in $(ls *.csproj); do mkdir -p ./${file%.*}/ && mv $file ./${file%.*}/; done

RUN dotnet restore "Moonglade.Web/Moonglade.Web.csproj"
COPY ./src .
WORKDIR "/src/Moonglade.Web"
RUN dotnet build "Moonglade.Web.csproj" -c $BUILD_CONFIGURATION -o /app/build

FROM build AS publish
ARG BUILD_CONFIGURATION=Release
RUN dotnet publish "Moonglade.Web.csproj" -c $BUILD_CONFIGURATION -o /app/publish

FROM base AS final
WORKDIR /app
COPY --from=publish /app/publish .
ENTRYPOINT ["dotnet", "Moonglade.Web.dll"]
```



Windows Plan + Code Deployment

- No Docker Containers
- All language runtimes are pre-installed
- Uses IIS in-process mode

Linux Plan + Code Deployment

- Use a special Docker Container - **appsvc/dotnetcore**
- Only .NET runtime are pre-installed
- IIS/YARP is only on Web Role

Linux Plan + Docker

- Use your own container
- Use your own .NET runtime
- IIS/YARP is only on Web Role



Thank You && Have a great .NET Conf

