Azure Artifacts Workshop

Agenda

- Azure Artifacts introduction
- Create a feed
- Versioning
- Universal packages
- Maven
- NPM
- Nexus vs Azure Artifacts
- Costs
- Q&A

cloud republic

Azure Artifacts

- Private/Public package managers
- Alternative for public package managers
  - Nuget.org
  - Npmjs.com
  - Pypi.org

NuGet

dotnet

NuGet.exe

Visual Studio

npm

npm

Maven

Maven

Gradle

Python

pip

twine

Universal

Universal packages

Create a Feed

- Everyone can create a feed within our organization

- A feed can contain multiple package managers

- It can't be public outside Azure DevOps

Azure Artifacts settings

Who can create feeds

(•) Everyone in this organization can create feeds

( ) Only administrators and these users can create feeds

Who can administer feeds

Project Collection Administrators, the feed's owners, and these users:

Add people

Cancel   Save

Project-scoped feeds vs. Organization-scoped feeds

Until now, all feeds were scoped to an organization, they could be viewed and accessed in the Azure Artifacts hub from any project within an organization. With the introduction of public feeds, we also introduced project-scoped feeds, which live inside the project that they were created in, and can only be seen when accessing the Azure Artifacts hub within that project.

Only project-scoped feeds can be made public, see the following section on public feeds. Learn more about the differences between project-scoped and organization-scoped feeds.

Create new feed

Feeds host your packages and let you control permissions.

Name

This name will appear in the URL for your feed

Project: TS

New: The feed will be scoped to the TS project. Learn more about project-scoped feeds. If this project is made public, its scoped feeds will also become public.

Visibility

- ● Members of your Azure Active Directory
  Any member of your AAD can view the packages in this feed
- ○ Members of ns-topaas
  Any member of your organization can view the packages in this feed
- ○ Specific people
  Only users you grant access to can view the packages in this feed

Upstream sources

☑ Include packages from common public sources

For example: nuget.org, npmjs.com

Cancel   Create

- **Simplicity**: your NuGet.config, .npmrc, or settings.xml contains exactly one feed (your feed).

- **Determinism**: your feed resolves package requests in order, so rebuilding the same codebase at the same commit or changeset uses the same set of packages

- **Provenance**: your feed knows the provenance of packages it saved via upstream sources, so you can verify that you're using the original package, not a custom or malicious copy published to your feed

- **Peace of mind**: packages used via upstream sources are guaranteed to be saved in the feed on first use; if the upstream source is disabled/removed, or the remote feed goes down or deletes a package you depend on, you can continue to develop and build

cloud republic

Versioning

- Semantic versioning - MAJOR.MINOR.PATCH
    - MAJOR version when you make incompatible API changes,
    - MINOR version when you add functionality in a backwards compatible manner, and
    - PATCH version when you make backwards compatible bug fixes
- GitVersion extension in Azure DevOps

cloud republic

UseGitVersion

Universal Packages

Universal Packages store one or more files together in a single unit that has a name and version.

```
az artifacts universal publish --organization https://dev.azure.com/ns-topaas/ --project="TS" --scope
project --feed TOPAAS-Test --name my-first-package --version 0.0.1 --description "Welcome to Universal
Packages" --path .

az artifacts universal download --organization https://dev.azure.com/ns-topaas/ --project="TS" --scope
project --feed TOPAAS-Test --name my-first-package --version 0.0.1 --path .
```

cloud republic

Maven: Create personal access token

Create a new personal access token ✕

Name

Maven Personal Access token

Organization

ns-topaas ⌄

Expiration (UTC)

30 days ⌄   19-6-2020 📅

Scopes

Authorize the scope of access associated with this token

Scopes ◯ Full access
        ⦿ Custom defined

Build
Artifacts, definitions, requests, queue a build, and updated build properties

☐ Read   ☐ Read & execute

Release
Read, update, and delete releases, release pipelines, and stages

☐ Read   ☐ Read, write, & execute   ☐ Read, write, execute, & manage

---

Test Management
Read, create, and updated test plans, cases, and results

☐ Read   ☐ Read & write

Packaging
Create, read, update, and delete feeds and packages

☑ Read   ☑ Read & write   ☑ Read, write, & manage

Show all scopes (27 more)

Create   Cancel

Success! ✕

You have successfully added a new personal access token. Copy the token now!
Maven Personal Access token token

pdeob22ouppqfig3xuuy 📋

⚠ Warning - Make sure you copy the above token now. We don't store it and you will not be able to see it again.

Close

Maven

```xml
<repositories>
    <repository>
        <id>TOPAAS-Test</id>
        <url>https://pkgs.dev.azure.com/ns-topaas/TS/_packaging/TOPAAS-Test/maven/v1</url>
        <releases>
            <enabled>true</enabled>
        </releases>
        <snapshots>
            <enabled>true</enabled>
        </snapshots>
    </repository>
</repositories>

<distributionManagement>
    <repository>
        <id>TOPAAS-Test</id>
        <url>https://pkgs.dev.azure.com/ns-topaas/TS/_packaging/TOPAAS-Test/maven/v1</url>
        <releases>
            <enabled>true</enabled>
        </releases>
        <snapshots>
            <enabled>true</enabled>
        </snapshots>
    </repository>
</distributionManagement>
```

cloud republic

Settings & Toolchains

```
C:\Users\Dibran\.m2>dir
 Volume in drive C has no label.
 Volume Serial Number is 4618-5F24


 Directory of C:\Users\Dibran\.m2

20-05-2020  14:40    <DIR>          .
20-05-2020  14:40    <DIR>          ..
20-05-2020  14:48    <DIR>          repository
20-05-2020  13:55               488 settings.xml
20-05-2020  14:39               312 toolchains.xml
               2 File(s)            800 bytes
               3 Dir(s)  21.349.195.776 bytes free
```

```xml
1  <settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
2            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3            xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0
4                                https://maven.apache.org/xsd/settings-1.0.0.xsd">
5    <servers>
6      <server>
7        <id>TOPAAS-Test</id>
8        <username>ns-topaas</username>
9        <password>[Personal_Access_Token]</password>
10     </server>
11   </servers>
12 </settings>
```

```xml
1  <?xml version="1.0" encoding="UTF8"?>
2  <toolchains>
3    <toolchain>
4      <type>jdk</type>
5      <provides>
6        <version>11</version>
7        <vendor>sun</vendor>
8      </provides>
9      <configuration>
10       <jdkHome>C:\Program Files\Java\jdk-11.0.6</jdkHome>
11     </configuration>
12   </toolchain>
13 </toolchains>
```

cloud republic

Use Maven package

- mvn install

- mvn compile

- mvn spring-boot:run

- http://localhost:8080/squared?input=6

```xml
<dependencies>
    <dependency>
        <groupId>nl.nsi.iris</groupId>
        <artifactId>azure-demo-library</artifactId>
        <version>1.0.0</version>
    </dependency>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-actuator</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-web</artifactId>
    </dependency>
</dependencies>
```

cloud republic

Update package

- mvn compile

- mvn deploy

- Continuous integration
  - Checkout code
  - Build
  - Test
  - Build artifacts
- Continuous delivery
  - Infrastructure as Code
  - Publish and release artifacts

cloud republic

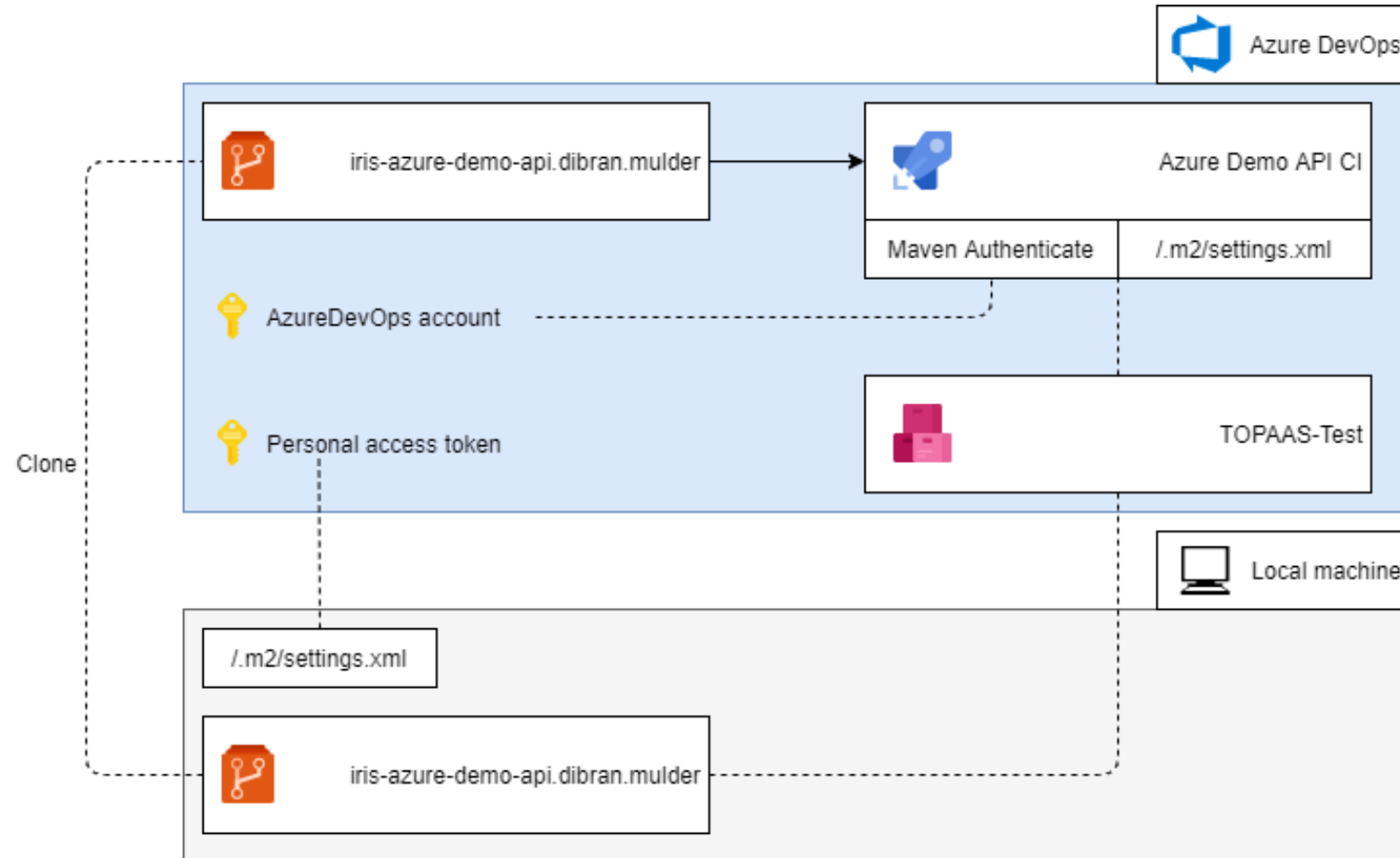## Use Maven in Azure Pipelines

- https://github.com/actions/virtual-environments

cloud republic

NPM

Get the tools

To use npm packages, get npm (which comes with Node.js) and the vsts-npm-auth helper.

Step 1

Download Node.js and npm

Step 2

Run this command

```
npm install -g vsts-npm-auth --registry
https://registry.npmjs.com --always-auth false
```

Can't use the credential helper?
Learn how to use a personal access token (PAT)

Close

---

npm

Command line reference

Get the tools

ⓘ   First time using Azure Artifacts with npm on this machine?   Get the tools   ✕

Project setup

Windows   Other

Add a .npmrc to your project, in the same directory as your package.json

```
registry=https://pkgs.dev.azure.com/ns-topaas/TS/_packaging/TOPAAS-Test/npm/registry/

always-auth=true
```

Then, run vsts-npm-auth to get an Azure Artifacts token added to your user-level .npmrc file

```
vsts-npm-auth -config .npmrc
```

Note: You don't need to do this every time. npm will give you a 401 Unauthorized error when you need to run it again.

Having issues? See the instructions for using a Personal Access Token to authenticate.

Restore packages

Run this command in your project directory

```
npm install
```

Publish packages

Run this command in your project directory

```
npm publish
```

cloud republic

NPM

- npm publish

Docker containers

- Azure Artifacts is **not** a container registry

- Azure Container Registry
    - Azure Managed service
- Amazon ECR
    - AWS Managed service
- Docker hub
    - Public container registry

cloud republic