



# Docker for OpenEdge Revealed

---

Thomas Hansen  
Director, nodeable

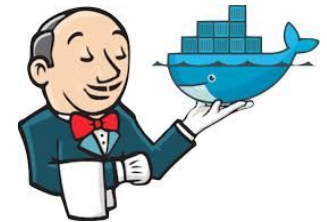
# nodeable?

---

- We do everything we can in containers!
- Worked with containers since 2015
- Worked with OpenEdge and Docker since 2015
  - Internal development, testing and production
  - Tools development and use
  - Customer implementations and PoCs

# Our work with containers

---



# Get Dockering...

---



<https://thomas.workshop.nodeable.io>

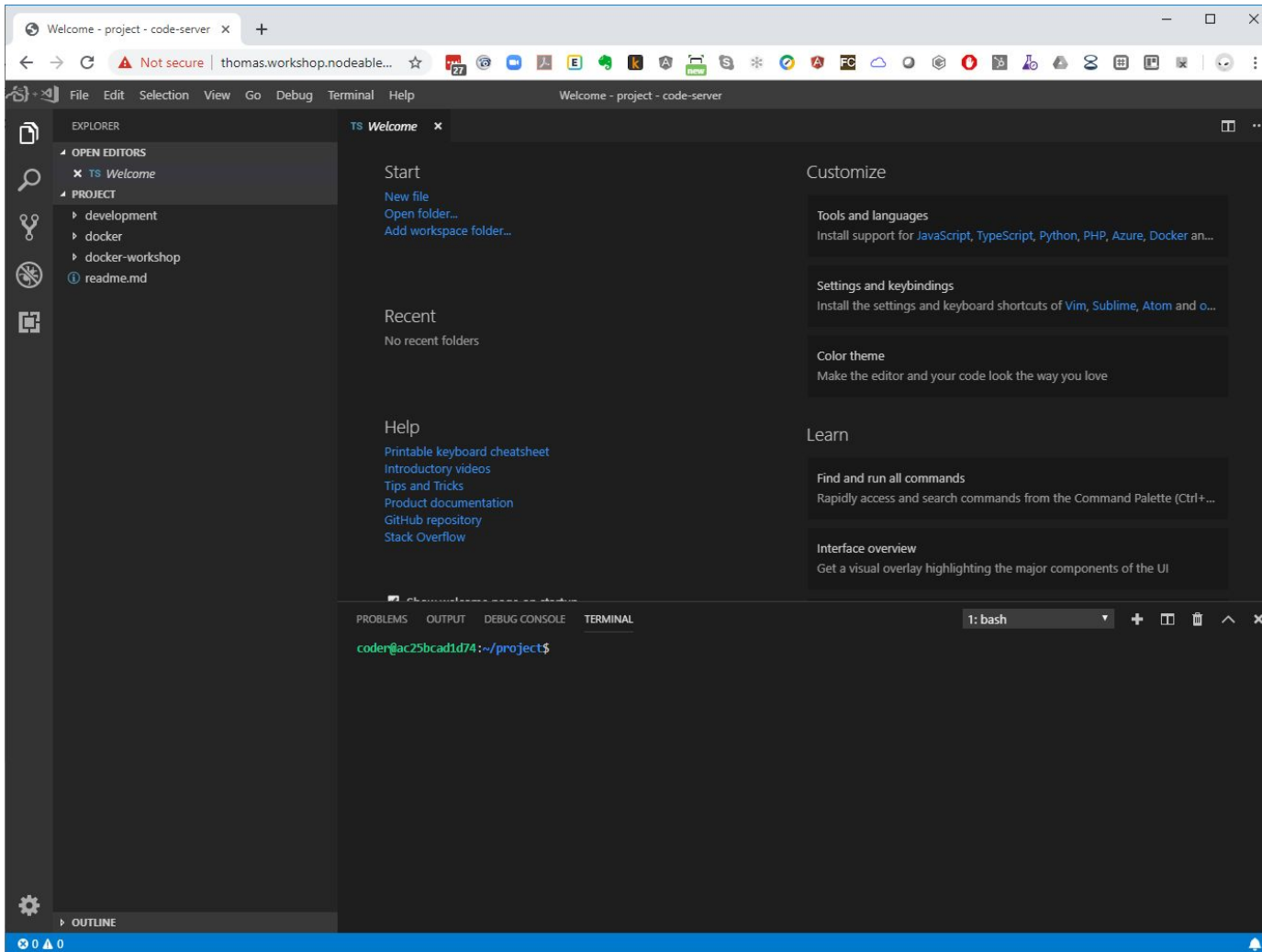
# Getting started

---

## Workshop server access

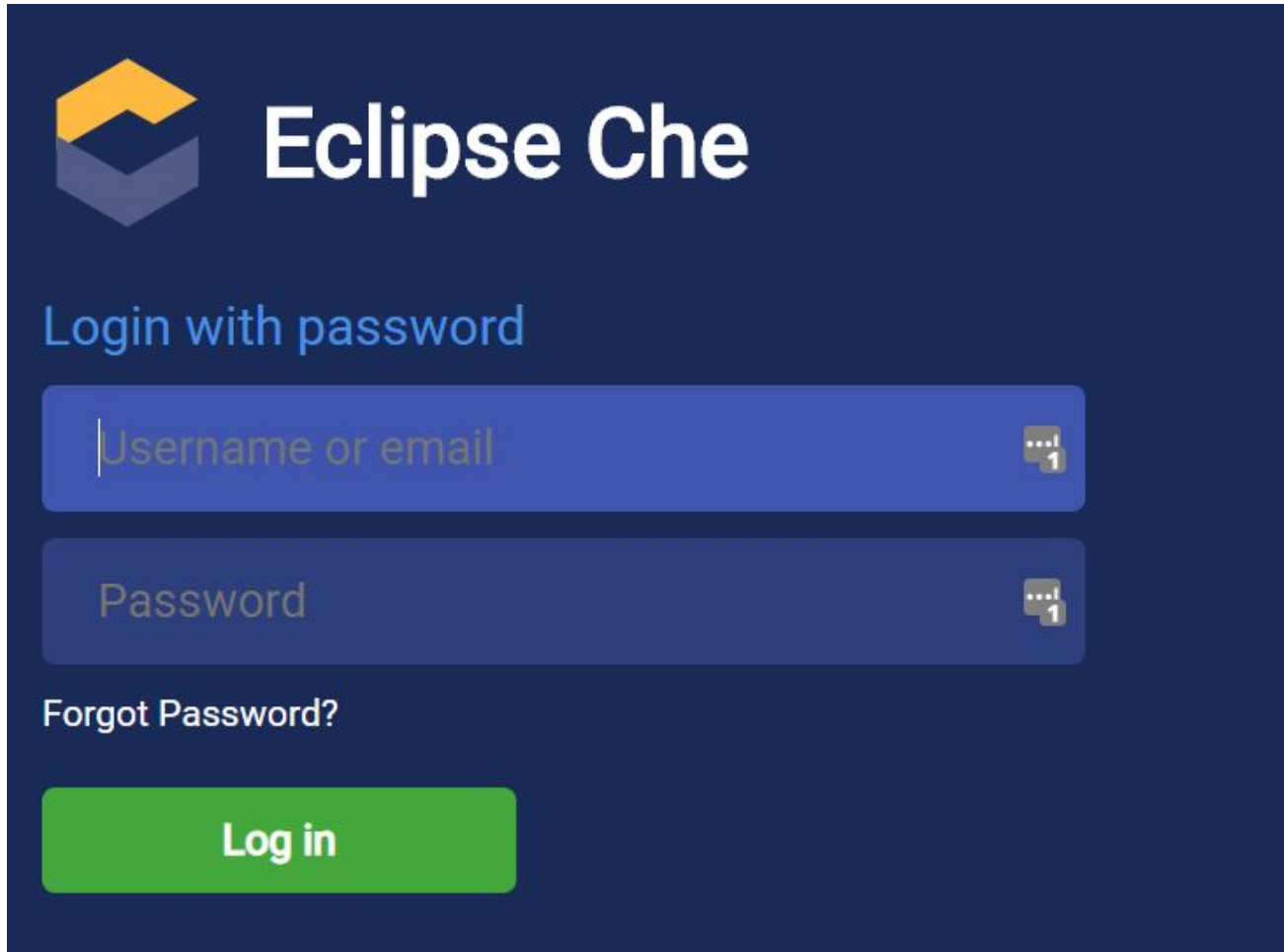
- CentOS hosts
  - Docker
  - Docker Compose
  - Apache Ant
  - Coder IDE (VS Code) in a container - and a browser  
<https://github.com/cdr/code-server>

# Coder in a browser



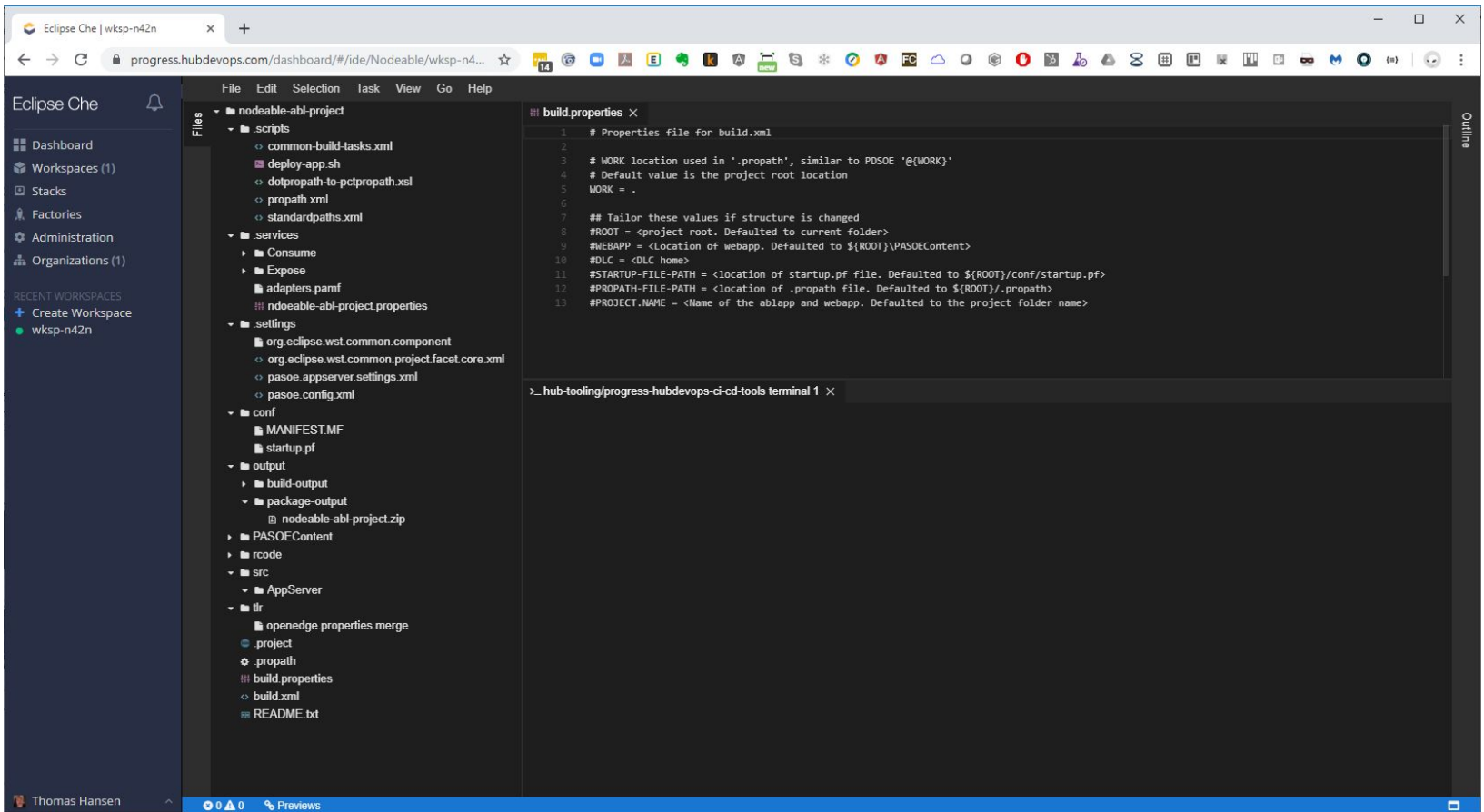
# HubDevops

---



The image shows a login form for Eclipse Che. It features the Eclipse Che logo (a stylized 'C' with orange and blue colors) and the text 'Eclipse Che' in white on a dark blue background. Below the logo, the text 'Login with password' is displayed in a lighter blue color. There are two input fields: the first is labeled 'Username or email' and the second is labeled 'Password'. Both fields have a small icon of a speech bubble with the number '1' next to it. Below the password field, there is a link that says 'Forgot Password?'. At the bottom of the form, there is a green button labeled 'Log in'.

# HubDevops





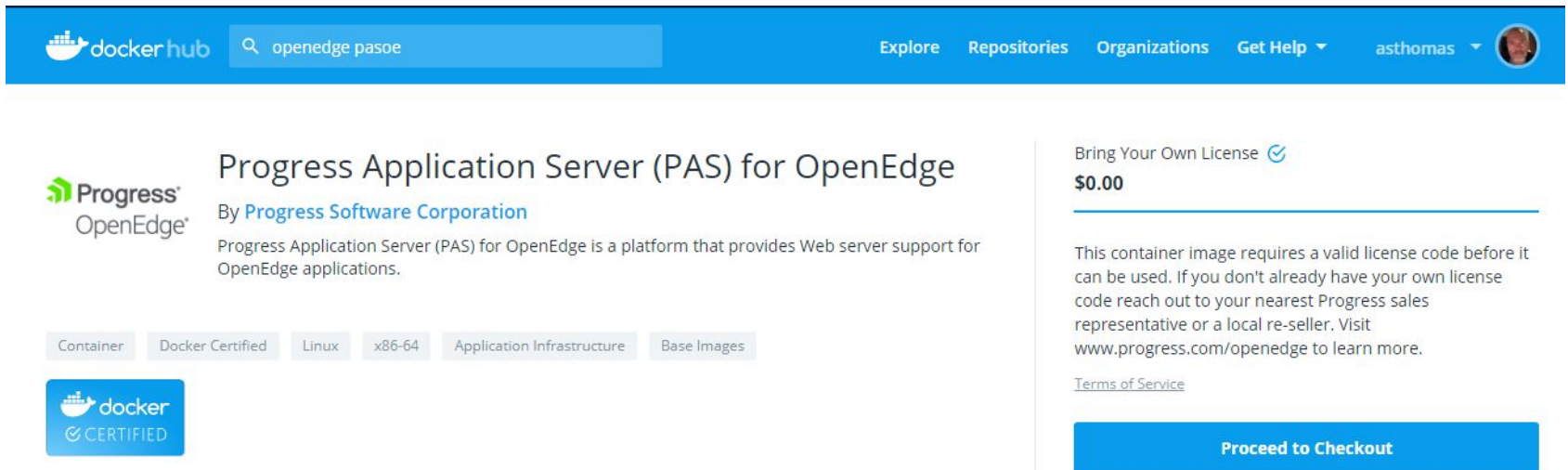
# PASOE in Docker

---

# Official Docker images

## Official images available from Progress Software:

- ESD
  - As exported Docker image in .tar.gz format
- Docker Hub
  - For download for free (login to Docker required)



The screenshot shows the Docker Hub interface for the 'Progress Application Server (PAS) for OpenEdge' image. The search bar contains 'openedge pasoe'. The page features a blue header with navigation links: Explore, Repositories, Organizations, Get Help, and a user profile for 'asthomas'. The main content area displays the image name, 'By Progress Software Corporation', and a description: 'Progress Application Server (PAS) for OpenEdge is a platform that provides Web server support for OpenEdge applications.' Below the description are tags for 'Container', 'Docker Certified', 'Linux', 'x86-64', 'Application Infrastructure', and 'Base Images'. A 'docker CERTIFIED' badge is visible. On the right, the pricing section shows 'Bring Your Own License' with a checkmark and '\$0.00'. A note states: 'This container image requires a valid license code before it can be used. If you don't already have your own license code reach out to your nearest Progress sales representative or a local re-seller. Visit www.progress.com/openedge to learn more.' A 'Terms of Service' link is present. At the bottom right, there is a blue button labeled 'Proceed to Checkout'.

# PASOE Project packaging

---

- Standard PASOE development in PDSOE
- Packaging for Docker deployment with extra files
  - build.properties file
  - ant scripts to build and package

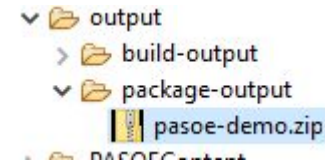
# ant package

```
Console Problems Tasks Servers Progress OpenEdge Server Monitor Log Viewer SonarLint Report Search
<terminated> pasoe-demo build.xml (package full) [Ant Build] C:\java-1.8.0-openjdk-1.8.0.191-1.b12.redhat.windows.x86_64\bin\javaw.exe (7 Oct 2019, 14:57:00)
package:
  [get] Destination already exists (skipping): D:\Development\pasoe-demo\.scripts\standardpaths.xml
clean:
  [delete] Deleting directory D:\Development\pasoe-demo\PASOEContent\output
  [mkdir] Created dir: D:\Development\pasoe-demo\PASOEContent\output
cleanBuildOutput:
  [mkdir] Created dir: D:\Development\pasoe-demo\PASOEContent\output\build-output
compile:
  [echo] OpenEdge Version: 12.1.0
[PCTVersion] PCT Version : pct-212-master-bf2b096f45e37bd8e52fffa32e83d8ecb830b8cb
[PCTCompile] PCTCompile - Progress Code Compiler
[PCTCompile] 16 file(s) compiled
package:
  [mkdir] Created dir: D:\Development\pasoe-demo\PASOEContent\output\package-output
[ABLWebAppPackage] Started executing ABLWebAppPackage
[ABLWebAppPackage] Validating Task Properties
[ABLWebAppPackage] Executing ABLWebApp packaging task
[ABLWebAppPackage] Getting Rest Services
[ABLWebAppPackage] Getting WebSpeed Services
[ABLWebAppPackage] Using Temporary Location: C:\Users\thomas\AppData\Local\Temp\Temp19616843515416318461570474625664
[ABLWebAppPackage] Copying Rest Content to temporary folder.
[ABLWebAppPackage] Copying JSON files
[ABLWebAppPackage] Generating Handler file
[ABLWebAppPackage] Generating PAAR files
[ABLWebAppPackage] Creating War file
[ABLWebAppPackage] War Generation is successful, Check file at D:\Development\pasoe-demo\PASOEContent\output\package-output\pasoe-demo_tmp.war
[ABLWebAppPackage] Cleaning up temporary files...
[ABLWebAppPackage] War file generation task completed successful.
  [zip] Building zip: D:\Development\pasoe-demo\PASOEContent\output\package-output\pasoe-demo.war
  [delete] Deleting: D:\Development\pasoe-demo\PASOEContent\output\package-output\pasoe-demo_tmp.war
  [zip] Building zip: D:\Development\pasoe-demo\output\package-output\pasoe-demo.zip
BUILD SUCCESSFUL
Total time: 8 seconds
```

# Packaged PASOE app

---

- Zip file created with
  - .war file
  - extra files for PASOE deployment
- This can now be copied to PASOE Docker deployment setup



# PASOE Docker Image

---

- Docker image built on CentOS
- Runtime version of OpenEdge for PASOE
- BYOL - Bring Your Own License
- Zip file contains
  - PROGRESS\_PASOE\_CONTAINER\_IMAGE\_12.1.0\_LNX\_64.tar.gz
  - deploy directory structure for creating application images and deploying to Docker

# Demo

---

# OpenEdge Database Docker Image

---

- Docker image built on CentOS
- Runtime version of OpenEdge for databases
- BYOL - Bring Your Own License



# OpenEdge Database Docker Image

---

- Intended for development and testing purposes only
- Requires the following licenses:
  - Docker Enterprise Edition (testing will work with Docker CE)
  - Enterprise DB licenses
- Can handle various database options
  - Create database from .df, .st and .d files
  - Create database from backup
  - Create database from sample database e.g. sports2020
  - Run database from host file system

# Demo

---

# OpenEdge and Docker Use Cases

---

- PASOE
- Databases
- CI/CD - compile, build .pl files, unit testing
- CI/CD - automated application testing
- Application development environment spin-ups
- QA of OE application stacks
- ...

# Need more OpenEdge in Docker?

---

Need more OpenEdge in Docker than the official images?

- Create your own images for internal use - linux only
- response.ini file from Linux installation
- Silent install of OpenEdge with Dockerfile
- Customize or extend images to different purposes
  - Standard OE - use for compiling, tooling,
  - WebSpeed (OE + Apache configured for CGI)
  - AppServer (OR + Tomcat + AIA)

# Technology integration example

---

## Bridgeable:

- Combined node.js & OpenEdge container
- Bridge between node.js and OE
  - node.js server receives request with JSON payload
  - DYNAMIC-NEW of OOABL class
  - Execute class to call AppServer backend to execute API and receive JSON payload
  - return JSON payload to node.js
  - return JSON from node.js



[thomas@nodeable.io](mailto:thomas@nodeable.io)