FEniCSx Installation

Using Docker Containers on Windows 11 and Conda on macOS

Farhad Kamarei

Abstract

This document provides a general guide for installing **FEniCSx** in a Docker container on Windows 11 and using Conda on macOS. The procedure uses Windows PowerShell/-Command Prompt and Terminal on macOS, along with Visual Studio Code (VS Code) for code editing and container management.

1 Windows 11 Installation

1.1 Step 1: Install Docker Desktop

1. Download Docker Desktop for Windows from the official site: https://www.docker.com/products/docker-desktop/
2. Run the installer and follow the on-screen instructions.
3. Restart your computer if prompted.
4. Verify installation by opening PowerShell or Command Prompt and running:

```
docker --version

```

Listing 1: Verify Docker installation

You should see output similar to `Docker version 28.3.0, build ...`

1.2 Step 2: Install Visual Studio Code (VS Code)

1. Download VS Code from: https://code.visualstudio.com/
2. Run the installer and follow the setup instructions.
3. **Install Essential Extensions:** After installation, install the following extensions for better Docker and container integration:
   - **Docker** (by Microsoft) - Provides Docker management capabilities
   - **Python** (by Microsoft) - Python language support and debugging

   Installation Methods:

   - **Method 1 - Extensions Tab:** Open VS Code, click the Extensions icon (four squares) in the sidebar, search for each extension name, and click "Install"
   - **Method 2 - Command Palette:** Press `Ctrl+Shift+P`, type "Extensions: Install Extensions", then search and install

- **Method 3 - VS Code Terminal:** Open terminal in VS Code (`Ctrl+‘`) and run:

```
code --install-extension ms-azuretools.vscode-docker
code --install-extension ms-python.python

```

Listing 2: Install VS Code extensions via terminal

1.3 Step 3: Prepare Your Working Directory

1. Choose or create a folder where you want to store your FEniCSx project files.
2. Open the folder in VS Code by either:
   - Using `File > Open Folder` in VS Code
   - Right-clicking the folder and selecting "Open with Code" (if available)
   - Using PowerShell to navigate and open:

```
cd "C:\Users\YourUsername\Documents\fenicsx_project"
code .

```

Listing 3: Navigate and open folder in VS Code

1.4 Step 4: Pull the FEniCSx Docker Image

1. Open VS Code terminal by pressing `Ctrl+Shift+‘` (backtick) or going to `Terminal > New Terminal`
2. In the VS Code terminal, pull the latest nightly image:

```
docker pull dolfinx/dolfinx:nightly

```

Listing 4: Pull FEniCSx Docker image via VS Code terminal

3. (Optional) For a stable release, check available tags at https://hub.docker.com/r/dolfinx/dolfinx/tags
4. **Note:** The download may take several minutes depending on your internet connection, as the image is approximately 2-3 GB

1.5 Step 5: Run the FEniCSx Docker Container

1. In the VS Code terminal (or PowerShell), start a new container and mount your working directory:

```
    docker run -it --name dolfinximage -v ${PWD}:/home/shared dolfinx/
dolfinx:nightly

```

Listing 5: Create and run FEniCSx container

2. **Note:** In PowerShell, `${PWD}` automatically resolves to the current directory.
3. Inside the container, your files will be available at `/home/shared`.

1.6 Step 6: Using Python in the Container

1. **Verify installation:** Test that FEniCSx is properly installed:

```
    python3 -c "import dolfinx; print(f'DOLFINx version: {dolfinx.
__version__}')"
```

Listing 6: Verify FEniCSx installation

2. Once inside the container, you can run your Python scripts as follows:

```
    cd /home/shared
    python3 test.py
```

Listing 7: Run Python scripts in container

3. **If you encounter** "python3: command not found" or wish to use the FEniCSx environment explicitly, use:

```
    /dolfinx-env/bin/python test.py
```

Listing 8: Use specific Python environment

1.7 Step 7: Reusing the Container

1. To stop and exit the container, type:

```
    exit
```

Listing 9: Exit container

2. To restart and attach to the container in the future (can be done from VS Code terminal):

```
    docker start -ai dolfiximage
```

Listing 10: Restart existing container

1.8 Windows Tips and Troubleshooting

- **VS Code Integration:** You can edit files in VS Code directly in your project folder; changes are reflected inside the container in real-time.
- **Terminal Preference:** You can use either the integrated VS Code terminal or standalone PowerShell - both work equally well.
- If you need to install additional Python packages, use `pip` inside the container.
- If you get permission issues with file sharing, ensure your drive is shared in Docker Desktop's settings.
- To remove the container:

```
    docker rm dolfiximage
```

Listing 11: Remove container

- To remove the image (if you want to clean up):

```
docker rmi dolfinx/dolfinx:nightly
```

Listing 12: Remove Docker image

2 macOS Installation

2.1 Step 1: Install Anaconda or Miniconda

Note: You do not need to install Python separately. Anaconda/Miniconda comes with Python and conda package manager.

1. **Choose your installer:**
   - **Anaconda** (recommended for beginners): Full distribution with many pre-installed packages
   - **Miniconda** (minimal): Lighter installation with only essential components
2. **Download and install:**
   - **Anaconda or Miniconda:** Download from https://www.anaconda.com/products/distribution
3. Run the downloaded installer (`.pkg` file) and follow the installation wizard.
4. **Verify installation:** Open Terminal (Applications > Utilities > Terminal) and run:

```
conda --version
```

Listing 13: Verify conda installation

You should see output like `conda 23.x.x`

2.2 Step 2: Install Visual Studio Code (VS Code)

1. Download VS Code for macOS from: https://code.visualstudio.com/
2. Open the downloaded `.zip` file and drag Visual Studio Code to your Applications folder.
3. **Install Essential Extensions:** Install the Python extension for better development experience:
   - **Python** (by Microsoft) - Python language support and debugging

Installation Method via Terminal:

```
code --install-extension ms-python.python
```

Listing 14: Install VS Code extensions via terminal on macOS

2.3 Step 3: Create FEniCSx Environment

1. **Open Terminal** (Applications > Utilities > Terminal)

2. **Create a new conda environment:**

```
conda create -n fenicsx-env
```

Listing 15: Create FEniCSx conda environment

3. **Activate the environment:**

```
conda activate fenicsx-env
```

Listing 16: Activate FEniCSx environment

4. **Install FEniCSx with visualization support:**

```
conda install -c conda-forge fenics-dolfinx mpich pyvista
```

Listing 17: Install FEniCSx and dependencies

Note: This installation may take 10-15 minutes as it downloads and compiles necessary packages.

5. **Verify installation:** Test that FEniCSx is properly installed:

```
python3 -c "import dolfinx; print(f'DOLFINx version: {dolfinx.__version__}')"
```

Listing 18: Verify FEniCSx installation

2.4 Step 4: Prepare Your Working Directory

1. **Create a project directory:**

```
mkdir ~/Documents/fenicsx_project
cd ~/Documents/fenicsx_project
```

Listing 19: Create and navigate to project directory

2. **Open the directory in VS Code:**

```
code .
```

Listing 20: Open project in VS Code

3. **Configure Python interpreter in VS Code:**

   - Open Command Palette (`Cmd+Shift+P`)
   - Type "Python: Select Interpreter"
   - Choose the interpreter from your fenicsx-env (usually located at `/anaconda3/envs/fenicsx-env/bin/python3` or similar)

2.5 Step 5: Test Your Installation

1. **Run a test:** In Terminal (make sure fenicsx-env is activated):

```
# Make sure environment is activated
conda activate fenicsx-env

# Run the test
python3 test.py

```

Listing 21: Run FEniCSx test

2. **Expected output:** You should see success messages confirming the installation works properly.

2.6 macOS Tips and Troubleshooting

- **VS Code Integration:** Ensure you select the correct Python interpreter from your conda environment
- **Package Installation:** Install additional packages using conda when possible: `conda install -c conda-forge package_name`
- **Environment List:** View all environments with: `conda env list`
- **Remove Environment:** If needed, remove with: `conda env remove -n fenicsx-env`

2.7 Alternative: Docker Installation

Note: If you prefer using Docker containers (for consistency with other systems or specific requirements), you can follow the same Docker installation steps described in the Windows section above. The main differences would be:

- Download Docker Desktop for Mac instead of Windows
- Use Terminal instead of PowerShell
- Use `$(pwd)` instead of `${PWD}` in volume mounting commands
- Use `Cmd` instead of `Ctrl` for keyboard shortcuts in VS Code

Author: Farhad Kamarei
Affiliation: University of Illinois Urbana-Champaign
Email: kamarei2@illinois.edu