# Docker Deployment Guide

Run the FrameworX runtime inside Docker for container-based deployments.

**Reference** **Runtime** **Installation** **Deployment** Docker

**Version 10.1.5+**

---

The 10.1.5 runtime ships on .NET 10 and runs as a long-lived container process. This page covers image preparation, the runtime configuration needed inside a container, and persistent-volume layout.



---

## Runtime Requirements

- .NET 10 runtime. The 10.1.5 runtime targets `net10.0`.
- Linux or Windows container host. Both official .NET 10 base images work.
- Writable volume for the solution folder, log folder, and Historian storage.
- Exposed TCP port for the runtime HTTP endpoint. The conventional port for the runtime service is **3101**. Each service JSON sets its own `portNumber`.
- Network reachability to any connected PLCs, databases, MQTT brokers, and tRPC clients.

### Image Layout

Place the runtime binaries under the container path `/app/FactoryStudio` and the solution under a mounted volume, for example `/solutions/<SolutionName>`. Log files go to `/app/FactoryStudio/Logs`.

---

## Dockerfile Example

```
FROM mcr.microsoft.com/dotnet/runtime:10.0

WORKDIR /app

COPY ./FactoryStudio /app/FactoryStudio
COPY ./Solution /solutions/Plant1

EXPOSE 3101

ENTRYPOINT ["dotnet", "/app/FactoryStudio/bin/net10.0/TStartup.dll", "/solution:/solutions/Plant1.tproj"]
```

Adjust the ENTRYPOINT to match the runtime service bundled in your image.

---

## Kubernetes Manifest

A minimal deployment manifest. A dedicated HTTP probe endpoint is planned for a later release. Until then, rely on Kubernetes process-liveness semantics and the Windows Service state on Windows nodes.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: frameworx-runtime
spec:
  replicas: 1
  template:
    spec:
      containers:
      - name: runtime
        image: tatsoft/frameworx-runtime:10.1.5
        ports:
        - containerPort: 3101
```

## Persistent Volumes

| Path | Purpose | Mount Type |
|---|---|---|
| `/solutions` | Solution files (`.tproj`) and embedded datasets. | Persistent volume. |
| `/app/FactoryStudio/Logs` | Runtime trace logs. | Persistent volume or a log-forwarding sidecar. |
| `/app/FactoryStudio/Historian` | Built-in Historian storage when the solution uses it. | Persistent volume with IOPS matching your retention load. |
| `/app/FactoryStudio/certs` | PFX files for HTTPS. See TLS and SSL Configuration. | Secret-backed volume. |

## Cloud Marketplace Images

Tatsoft plans to publish prebuilt runtime images for Azure, AWS, and GCP marketplaces. Images will ship with environment variables for the solution path, HTTP port, and license activation, a default non-root user, and a minimal base image. Point your orchestrator at the marketplace image tag for your target runtime version, mount a solution volume, and expose the HTTP port. No repackaging is required.

## Troubleshooting

| Symptom | Likely Cause | Next Step |
|---|---|---|
| Container exits immediately. | Solution path wrong, license missing, or entrypoint pointing at the wrong DLL. | Check `docker logs` for the startup trace, and verify the solution volume mount. |
| Runtime cannot reach PLCs or databases. | Network policy or missing service in the pod network. | Confirm the pod network reaches the target, and check outbound firewall rules. |
| Historian writes fail. | Persistent volume read-only, or not mounted. | Verify the Historian volume mount and permissions on `/app/FactoryStudio/Historian`. |

**In this section...**