

# Interfaces & Namespaces

## Interfaces

The Feat-Standard public interface is covered by the namespaces FeatStd and various subnamespaces of namespace FeatStd. All namespaces containing either **Internal** or **Private** in the name cover private namespaces, which are not recommended to be used in any application, as it is not guaranteed that these interfaces remain stable respectively exist in any future release. Changes in any of these private namespaces can be applied without any (prior) notifications.

## Namespace FeatStd

- [Container](#)
- [Utilities](#)

## Namespace FeatStd::Diagnostics

- [Diagnostics](#)

## Namespace FeatStd::MemoryManagement

- [Memory Management](#)

## Namespace FeatStd::PerfMon

- MonitorPerformanceReporting

## Namespace FeatStd::Platform

This interface defines the access to platform dependent objects. The implementation of this platform dependent code is not part of FeatStd, though there are a few reference platforms, like SimulationPlatform (implemented basically for Win32) or GenericPlatform (providing generic respectively stub implementations), which can be used as templates for the customer.

This platform interface can be seen as an official porting interface of FeatStd to any customer specific platform setup, which is the reason for outlining it in the separate namespace FeatStd::Platform.

- [Platform](#)

---

Revision #4

Created 2023-02-27 05:20:59 UTC by Tsuyoshi.Kato

Updated 2024-11-13 23:30:35 UTC by Tsuyoshi.Kato