FeatStd V3.0.0

- FeatStd CMake Switches
- FeatStd Diagnostics
- FeatStd Memory Management
- FeatStd Memory Pool
- FeatStd Monitor
- FeatStd Util

FeatStd CMake Switches

Note:

For detailed information on FeatStd configuration please see Application Development > FeatStd > FeatStd Features

V2.10.0	V3.0.0
CGIFEATURE_ENABLE_LOG has been unified with FEATSTD_ENABLE_LOG	FEATSTD_LOG_ENABLED
CGIFEATURE_ENABLE_MEMORYPOOL has been unified with FEATSTD_ENABLE_MEMORYPOOL	FEATSTD_MEMORYPOOL_ENABLED
CGIFEATURE_ENABLE_MEMORY_STATISTIC has been unified with FEATSTD_ENABLE_SYSTEM_MEMORY_STATISTIC	FEATSTD_SYSTEM_MEMORY_STATISTIC_ENABLED
CGIFEATURE_SYSTEM_MEMORY_STATISTIC_FILE_AND_LINE_TRACKING has been unified with FEATSTD_ENABLE_SYSTEM_MEMORY_STATISTIC_FILE_AND_LINE_TRACKING	FEATSTD_SYSTEM_MEMORY_STATISTIC_FILE_AND_LINE_TRACKING_ENABLED
CGIFEATURE_FRACTIONAL_NUMBER_TYPE has been unified with FEATSTD_FRACTIONAL_NUMBER_TYPE	FEATSTD_FRACTIONAL_NUMBER_TYPE
CGIFEATURE_ENABLE_MONITOR has been unified with FEATSTD_ENABLE_MONITOR	FEATSTD_MONITOR_ENABLED
COURIER_ENABLE_IPC has been unified with FEATSTD_ENABLE_IPC	FEATSTD_IPC_ENABLED
FEATSTD_ENABLE_THREADSAFETY	FEATSTD_THREADSAFETY_ENABLED
FEATSTD_ENABLE_UNITTEST_FRAMEWORK	FEATSTD_UNITTEST_FRAMEWORK_ENABLED
FEATSTD_ENABLE_DLT	FEATSTD_DLT_ENABLED

Note:

Refer to the Candera 3.0.0 CMake Changelog to learn about Candera V3.0.0 CMake switch modifications.

FeatStd Diagnostics

MemoryAppender added

MemoryAppender is added which allows logging events to a buffer in memory.

FeatStd Memory Management

Removed Sharing Object

The outdated and orphaned file SharingObject.h has been removed.

Retain Count atomic and uncopyable

The retain count of shared objects - enabled by macro "FEATSTD_SHARED_POINTER_DECLARATION" - has been made atomic to make increments and decrements thread-safe. Additionally the retain count has been made uncopyable to avoid copying the retain count when cloning shared objects.

FeatStd Memory Pool

Enforce Initialization of BackingHeap

Initialization of BackingHeap (MallocBackingHeap) is enforced. BackingHeap initializations are reference counted and missing initialization will assert. This is to detect missing initialization early, even if MallocBackingHeap does not require Init (but switching to other BackingHeap will cause crashes then). A new parameter in BackingHeap::Destroy() / MallocBackingHeap::Destroy() forces the destruction of the heap (even if init counter is larger 1).

FeatStd Monitor

Monitor module migrated from Candera

The Candera module CanderaMonitor has been migrated to FeatStd.

See also:

FeatStd CMake Switches for the changed CMake feature flag related to Monitor functionality.

FeatStd Util

String Ids for translatable Text

A new class TextId has been added for accessing translatable texts from a string.

See also:

String::String(const TextId& id)