# Lists

## Classes

* MonoDomain (class)
* MonoAssembly (class)
* MonoAssemblyName (class)
* MonoClass (class)
* MonoMethod (class)
* MonoObject (class)
* MonoString (class)
* MonoBoolean (?)
* MonoArray (class)
* MonoMemoryPool (class)
* MonoCodeManager (class)
* MonoException (class)
* MonoAppDomainSetup (class)
* MonoAppDomain (class)
* MonoAppContext (class)
* MonoVTable (class)
* MonoThreadsSynchronization (class)
* MonoThreadInfoCallbacks (class)
* MonoRuntimeInfo (class)
* MonoCounter (class)
* MonoHost (class)

## Structures

* None

# Class Details

## MonoObject (strcut MonoObject defined in metadata/object.h)

### Members

* MonoVTable\* VTable (struct MonoObject, MonoVTable\* vtable)
* MonoThreadsSynchronization\* ThreadSynchronization (struct MonoObject, MonoThreadsSync\* synchronization)

## MonoVTable (struct MonoVTable defined in metadata/class-internals.h)

### Notes

* Members must be in specific order for vtable to work correctly

### Members

* const MonoClass\* Class (struct MonoVTable, MonoClass\* klass)
* void\* GarbageCollectorDescription (struct MonoVTable, void\* gc\_descr)
* MonoDomain\* Domain (struct MonoVTable, MonoDomain\* domain)
* void\* SystemType (struct MonoVTable, gpointer type)
* uint8\_t\* InterfaceBitmap (struct MonoVTable, guint8\* interface\_bitmap)
* uint16\_t MaxInterfaceId (struct MonoVTable, guint16 max\_interface\_id)
* uint8\_t Rank (struct MonoVTable, guint8 rank)
* uint\_t RemotelyActivated : 1 (struct MonoVTable, guint remote : 1)
* uint\_t Initialized : 1 (struct MonoVTable, guint initialized : 1)
* uint\_t InitializeFailed : 1 (struct MonoVTable, guint init\_failed : 1)
* uint\_t HasStaticFields : 1 (struct MonoVTable, guint has\_static\_fields : 1)
* uint\_t GarbageCollectorBits : 4 (struct MonoVTable, gc\_bits : MONO\_VTABLE\_AVAILABLE\_GC\_BITS)
* uint32\_t IMTCollisionsBitmap (struct MonoVTable, guint32 imt\_collisions\_bitmap)
* MonoRuntimeGenericContext\* RuntimeGenericContext (struct MonoVTable, MonoRuntimeGenericContext\* runtime\_generic\_context)
* void\* vtable[1] (struct MonoVTable, gpointer vtable[MONO\_ZERO\_LEN\_ARRAY])

### Methods

* …

## MonoThreadsSynchronization (formerly MonoThreadsSync, struct \_MonoThreadsSync defined in metadata/monitor.c)

### Members

* …

## MonoDomain (struct \_MonoDomain defined in metadata/domain-internals.h, implemented in metadata/domain.c)

### Members

* ? Lock (struct \_MonoDomain, CRITICAL\_SECTION lock)
* MonoMemPool\* MemoryPool (struct \_MonoDomain, MonoMemPool\* mp)
* MonoCodeManager\* CodeManagerMemoryPool (struct \_MonoDomain, MonoCodeManager\* code\_mp)
* MonoAppDomainSetup\* AppDomainSetup (struct \_MonoDomain, MonoAppDomainSetup\* setup)
* MonoAppDomain\* Domain (struct \_MonoDomain, MonoAppDomain\* domain)
* MonoAppContext\* DefaultAppContext (struct \_MonoDomain, MonoAppContext\* default\_context)
* public const MonoRuntimeInfo\* CurrentRuntime (static const MonoRuntimeInfo\* current\_runtime)
* private static const MonoRuntimeInfo[] SupportedRuntimes (static const MonoRuntimeInfo supported\_runtimes[])

### Methods

* static MonoDomain\* InitializeInternal(const string& filename, const string& executableFilename, const string& runtimeVersion) – (mono\_init\_internal)

## MonoMemoryPool (formerly MonoMemPool, struct \_MonoMemPool defined in metadata/mempool.c)

### Notes

* Definition depends on MALLOC\_ALLOCATION

### Members

* …

## MonoCodeManager (struct \_MonoCodeManager defined in utils/mono-codeman.c)

### Members

* …

## MonoAppDomainSetup (struct MonoAppDomainSetup defined in metadata/domain-internals.h)

### Notes

* Copy of System.AppDomainSetup

### Members

* …

## MonoAppDomain (struct \_MonoAppDomain defined in metadata/object-internals.h)

### Notes

* Copy of System.AppDomain

### Members

* …

## MonoAppContext (struct \_MonoAppContext defined in metadata/domain-internals.h)

### Members

* …

## MonoAssembly (struct \_MonoAssembly defined in metadata/metadata-internals.h)

### Members

* …

## MonoAssemblyName (struct \_MonoAssemblyName defined in metadata/metadata-internals.h)

### Members

* …

## MonoThreadInfoCallbacks

### Members

* …

## MonoThreadInfo

### Members

* …

## MonoGarbageCollector (defined in metadata/mono-gc.h, metadata/gc-internal.h)

### Members

* …

### Methods

* public void Collect(int generation) – (mono\_gc\_collect)
* public int GetMaxGeneration() – (mono\_gc\_max\_generation)
* public int GetGeneration(MonoObject\* object) – (mono\_gc\_get\_generation)
* public int GetCollectionCount(int generation) – (mono\_gc\_collection\_count)
* public int64\_t GetUsedSize() – (mono\_gc\_get\_used\_size)
* public int64\_t GetHeapSize() – (mono\_gc\_get\_heap\_size)
* public int InvokeFinalizers() – (mono\_gc\_invoke\_finalizers)
* public int WalkHeap(int flags, MonoGCReferences callback, void\* data) – (mono\_gc\_walk\_heap)

## MonoRuntimeInfo (defined in metadata/domain-internals.h)

### Members

* const char[12] RuntimeVersion – (struct MonoRuntimeInfo, char runtime\_version[12])
* const char[4] FrameworkVersion – (struct MonoRuntimeInfo, char framework\_version[4])
* const AssemblyVersionSet[4] VersionSets – (struct MonoRuntimeInfo, AssemblyVersionSet version\_sets[4])

## AssemblyVersionSet (defined in metadata/domain-internals.h)

### Members

* uint16\_t Major – (struct AssemblyVersionSet, guint16 major)
* uint16\_t Minor – (struct AssemblyVersionSet, guint16 minor)
* unit16\_t Build – (struct AssemblyVersionSet, guint16 build)
* uint16\_t Revision – (struct AssemblyVersionSet, guint16 revision)

## MonoHost

### Notes

* Threads
* TLS
* Semaphores
* Critical Sections
* Anything else “Native”

# Notes

* Look into USE\_MALLOC\_FOR\_MEMPOOLS, MALLOC\_ALLOCATION
* MONO\_INTERNAL defined in utils/mono-compiler.h

# Type Replacement

## eglib

* gboolean = bool
* gint32 = int32\_t (#include <cstdint>)
* guint32 = uint32\_t (#include <cstdint>)
* guint16 = uint16\_t (#include <cstdint>)
* guint8 = uint8\_t (#include <cstdint>)
* gpointer = void\*
* gconstpointer = const void\*
* gfloat = float
* gdouble = double
* GSList = std::forward\_list (#include <forward\_list>)
* GList = std::list (#include <list>)
* GPtrArray = std::vector<void\*> (#include <vector>)
* GHashTable = std::unordered\_map (#include <unordered\_map>)
* gchar = char
* gunichar = uint32\_t (#include <cstdint>)
* gunichar = uint16\_t (#include <cstdint>)

## Other

* void\*
* int (size different on different platforms?)