F4BMS Namespace

The F4BMS Namespace contains the types and methods required to interact with the F4:BMS file structure. The Namepace contains 3 Namespaces: F4BMS.LOD, F4BMS.Header, and F4BMS.Database.

Namespaces

Namespace	Description
F4BMS.Database	Contains the types required to provide basic interaction with the CT Database.
F4BMS.Header	Contains the types required to provide basic interaction with the Header File (KoreaObj.HDR)
F4BMS.LOD	Contains the types required to provide basic interaction with the LOD Objects (KoreaObj.LOD)
F4BMS.KEYFILE	Contains the types required to provide basic interaction with the Key Files

F4BMS.Database Namespace

Contains the types required to provide basic interaction with the CT Database.

Classes

Class	Description
AircraftData	Represents an Aircraft Object in the Class Table Database.
ClassTable	Represents the collection of Class Table Objects in the CT Database.
ClassTableEntry	A Single Class Table Entry from the Class Table Database.
FeatureData	Represents a Feature Object in the Class Table Database.
FeatureEntryData	Represents a Feature Entry Object in the Class Table Database.
IRSensorData	Represents an IR Sensor Object in the Class Table Database.
ObjectiveData	Represents an Objective Object in the Class Table Database.
PointData	Represents a Point Object in the Class Table Database (Depricated, use PointExData)
PointEXData	Represents a Point Data Object with extended properties in the Class Table Database.
PointHeaderData	Represents a Point Header Object in the Class Table Database.
RWRData	Represents an RWR Object in the Class Table Database.
RadarData	Represents a Radar Object in the Class Table Database.
RocketData	Represents a Rocket Object in the Class Table Database.
SimWeaponsData	Represents the Sim Weapons Data associated with a Weapon Object in the Class Table Database.
SquadronStoresData	Represents a SquadronStores Object in the Class Table Database.
UnitData	Represents a Unit Object in the Class Table Database.
VehicleData	Represents a Vehicle Object in the Class Table Database.
VisualSensorData	Represents a Visual Sensor Object in the Class Table Database.
WeaponData	Represents a Weapon Object in the Class Table Database.
WeaponListData	Represents a Weapon List (Hardpoint) Object in the Class Table Database.
AircraftDataFile	Represents an Aircraft Class Table Database File.
FeatureDataFile	Represents a Feature Class Table Database File.
FeatureEntryDataFile	Represents a Feature Entry Class Table Database File.
IRSensorDataFile	Represents an IR Sensor Class Table Database File.
ObjectiveDataFile	Represents an Objective Class Table Database File.
PointDataFile	Represents a Point Data Class Table Database File.
PointEXDataFile	Represents an Extended Point Data Class Table Database File.
PointHeaderDataFile	Represents a Point Header Class Table Database File.
RWRDataFile	Represents an RWR Class Table Database File.
RadarDataFile	Represents a Radar Class Table Database File.
RocketDataFile	Represents a Rocket Class Table Database File.
SimWeaponsDataFile	Represents a Sim Weapons Class Table Database File.

SquadronStoresDataFile	Represents a Squadron Stores Class Table Database File.
UnitDataFile	Represents a Unit Class Table Database File.
VehicleDataFile	Represents a Vehicle Class Table Database File.
VisualSensorDataFile	Represents a Visual Sensor Class Table Database File.
WeaponDataFile	Represents a Weapon Class Table Database File.
WeaponListDataFile	Represents a Weapon List Class Table Database File.

Enumerations

Enumeration	Description
AirUnitSubType	Enumerates the Subtypes assignable to Air Units.
ClassTableClass	Enumerates the Class of the Object in the Class Table.
ClassTableDomain	Enumerates the Domain of the Object in the Class Table.
CombatClass	Enumerates the Combat Types available to objects in the Class Table.
DamageType	Enumerates the Damage Types available to objects in the Class Table.
GroundUnitSubType	Enumerates the Subtypes available to the Ground Units in the Class Table.
GuidanceType	Enumerates the Guidance Types available to Weapons in the Class Table.
MovementType	Enumerates the types of Movement available to units in the Class Table.
NavalUnitSubType	Enumerates the Subtypes available to Naval Units in the Class Table.
PointType	Enumerates the types of Points assignable to Objectives in the Class Table.
PointListType	Enumerates the types Point Collections available to an Objective in the Class Table.
SensorType	Enumerates the types of Sensors available to units in the Class Table.
TargetingPodType	Enumerates the types of Targeting Pods available to the Aircraft in the Class Table.
UnitRole	Enumerates the Mission Roles available to units in the Class Table.
WeaponClass	Enumerates the class of Weapons available in the Class Table.
WeaponDomain	Enumerates the Domain which a Weapon can operate in.
WeaponType	Enumerates the types of Weapons available in the Class Table.

Structures

Structure	Description
DamageModel	Represents the 11 Byte Damage Model used to identify vulnerability of an object to a specific type of damage.
FeatureFlags	Represents the boolean flags derived from the Int16 Flag in a Feature Object.
FeatureEntryFlags	Represents the boolean flags derived from the Int16 Flag in a Feature Entry Object.
RolesModel	Represents the Collection of Roles a unit can perform and the predominance of each role.
SensorData	Represents the characteristics of a specific Sensor in the Class Table.
StatsModel	Represents the 8 Byte Statistics Model used to identify effectiveness or vulnerability of an object to a given Object Domain.
UnitFlags	Represents the boolean flags derived from the Int16 Flag in a Unit Object.
VehicleFlags	Represents the boolean flags derived from the Int32 Flag in a Vehicle Object.
WeaponFlags	Represents the boolean flags derived from the Int32 Flag in a Weapon Object.

Interfaces

Interface	Description
ICTFile	An interface class to a provide common interface for all CT DB Files.
ICTEntry	An interface class to a provide common interface for all CT DB Types.

F4BMS.Database.AircraftData Class

Represents an Aircraft Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class AircraftData
```

Constructors

Name	Description
AircraftData()	Initializes a new instance of the AircraftData Class.
AircraftData(Stream)	Initializes a new instance of the AircraftData Class using the supplied Stream object to read the data from a file or memory stream.
AircraftData(string, int)	Initializes a new instance of the AircraftData Class using the index of the AircraftData entry in the supplied .ACD File.

Properties

Name	Description
AircraftIndex	Exposes access to the underlying Int32 index of the Aircraft Object in the Aircraft Table.
CombatClass	Exposes access to the underlying CombatClass enumerated object.
Sensors	Exposes access to the underlying List of SensorData Structure objects which define the Sensors assingned to this Aircraft.
SignatureIndex	Exposes access to the underlying Int32 which represents the Aircraft Signature Index.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this AircraftData Object.

DebugPrint()	Returns a formatted string with advanced information about this AircraftData Object.
Save(Stream)	Writes the AircraftData Serializable data to the supplied Stream Object.

F4BMS.Database.AircraftDataFile Class

Represents an Aircraft Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class AircraftDataFile
```

Constructors

Name	Description
AircraftDataFile(string)	Initializes a new instance of the AircraftDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
Aircraft	AircraftData Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.ClassTable Class

Represents a Collection of CT Entry Objects which make up the bulk of the CT Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class ClassTable
```

Constructors

Name	Description
ClassTable()	Initializes a new instance of the ClassTable Class.
ClassTable(string)	Initializes a new instance of the ClassTable Class using the supplied Filename read the data from a file.

Properties

Name	Description
Count	Read-Only Count of the number of CT Entries in the Class Table.
Entries	Exposes access to the underlying List of ClassTableEntry Objects.

Methods

Interface	Description
Save(Stream)	Writes the ClassTable Serializable data to the supplied Stream Object.

F4BMS.Database.ClassTableEntry Class

Represents a CT Entry in the Class Table Database. (FALCON4.CT)

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class ClassTableEntry
```

Constructors

Name	Description
ClassTableEntry()	Initializes a new instance of the ClassTableEntry Class.
ClassTableEntry(Stream)	Initializes a new instance of the ClassTableEntry Class using the supplied Stream object to read the data from a file or memory stream.
ClassTableEntry(string, int)	Initializes a new instance of the ClassTableEntry Class using the index of the Object entry in the supplied .CT File.

Properties

Name	Description
Class	Exposes access to the underlying ClassTableClass Enumeration value.
CollidableFlag	Exposes access to the underlying byte flag identifying if the object is collidable.
CollisionRadius	Exposes access to the underlying Float radius which causes a collision with this object
CreatePriority	Exposes access to the underlying UInt16 used to determine rendering priority.
CTID	Exposes access to the underlying Class Table ID.
DamageSeed	Exposes access to the underlying UInt32 which represents the specific Damage Seed
DamageType	Exposes access to the underlying Int16 which identifies the Damage Type or the
DataType	Exposes access to the underlying File Type used to identify the type of file the
Domain	Exposes access to the underlying ClassTableDomain enumeration of the Environmental Domain this objects belongs to.

FineUpdateMultiplier	The multiplier applied to the Update Rate inside the Bubble.
FineUpdateRange	The Bubble Distance around this object in feet.
FineUpdateForcedRange	Distance in feet the graphics engine will force a fine update.
GlobalFlag	Identifies this Object as a Global Object.
Hitpoints	The number of Hitpoints this Object has.
ManagementDomain	The Domain in which this Object exists.
MajorRev	The Major Revision Number of this Object.
MinorRev	The Minor Revision Number of this Object.
Mode	Denotes the specific Mode of this Object in the Game Engine. Depricated in the File Structure.
NormalLOD	The LOD ID to use when rendering this CT in the Normal State.
RepairedLOD	The LOD ID to use when rendering this CT in the Repaired State.
DamagedLOD	The LOD ID to use when rendering this CT in the Damaged State.
DestroyedLOD	The LOD ID to use when rendering this CT in the Destroyed State.
LeftDestroyedLOD	The LOD ID to use when rendering this CT in the LeftDestroyed State.
RightDestroyedLOD	The LOD ID to use when rendering this CT in the RightDestroyed State.
BothDestroyedLOD	The LOD ID to use when rendering this CT in the BothDestroyed State.
PersistentFlag	Exposes access to the underlying flag which tells the graphics engine if this is a
PrivateFlag	Identifies the Object as private.
RecordID	The CT Record ID of the Object.
Specific	The Specific Type of Object with regards to the Subtype, Type, and Class of the CT Entry.
Subtype	The Subtype of this Object.
TangibleFlag	Identifies if this Object can be interacted with.
TransferrableFlag	Identifies if this Object can be transferred between Parent entities.
Type	The General Type of this Object.
UpdateRate	The time in ms before the graphics engine attempts to update this object.
UpdateTolerance	The maximum time in ms before the Graphics Engine is forced to update this Object.
UNK1	This value is either unused or the use in not currently known.
UNK2	This value is either unused or the use in not currently known.
UNK3	This value is either unused or the use in not currently known.
UNK4	This value is either unused or the use in not currently known.
UNK5	This value is either unused or the use in not currently known.
VehicleDataIndex	The index in the Vehicle Class Table which this CT Object represents.

Methods

Interface	**Description**
Print()	Returns a formatted string with basic information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.FeatureData Class

Represents a Feature Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class FeatureData
```

Constructors

Name	Description
FeatureData()	Initializes a new instance of the FeatureData Class.
FeatureData(stream)	Initializes a new instance of the FeatureData Class using the supplied Stream object to read the data from a file or memory stream.
FeatureData(string, int)	Initializes a new instance of the FeatureData Class using the index of the Object entry in the supplied .FCD File.

Properties

Name	Description
Angle	The angle offset from the Objective of this Feature.
CTIndex	The Index of this Object in the CT Database.
DamageModel	Exposes access to the underlying DamageModel Structure for this Object.
DetectionRange	Exposes access to the underlying StatsModel Object representing Detection Ranges for this Object.
Flags	The Flags associated with this Object.
Height	The Height in feet of this Object.
Hitpoints	The amount of damage this Object can withstand.
Name	The Text Name of this Object.
Priority	The Render Priority of this Object.

RadarType	The type of the Radar Object attached to this Feature.
RepairTime	The timein minutes required to repair this Feature.
UNK1	This value is either unused or the use in not currently known.
UNK2	This value is either unused or the use in not currently known.
UNK3	This value is either unused or the use in not currently known.
UNK4	This value is either unused or the use in not currently known.

Methods

Interface	**Description**
Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.FeatureDataFile Class

Represents a Feature Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class FeatureDataFile
```

Constructors

Name	Description
FeatureDataFile(string)	Initializes a new instance of the FeatureDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
Features	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.FeatureEntry Class

Represents a Feature Entry Object assigned to a Feature Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class FeatureEntry
```

Constructors

Name	Description
FeatureEntryData()	Initializes a new instance of the FeatureEntry Class.
FeatureEntry(stream)	Initializes a new instance of the FeatureEntry Class using the supplied Stream object to read the data from a file or memory stream.
FeatureEntry(string, int)	Initializes a new instance of the FeatureEntry Class using the index of the Object entry in the supplied .FED File.

Properties

Name	Description
ClassIDs	Collection of byte objects identifying the Class IDs of the children assigned to this
DamageValue	The amount of damage required to consider this Feature Entry destroyed.
FeatureIndex	The Index of the Feature this Entry belongs to.
Flags	Exposes access to the underlying Flag Object for this Feature Entry.
Heading	The Heading of this Feature Entry.
Offset	Exposes access to the underlying System.Windows.Media.Media3D.Point3D Object
UNK1	This value is either unused or the use in not currently known.
UNK2	This value is either unused or the use in not currently known.

Methods

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.FeatureEntryDataFile Class

Represents a FeatureEntry Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class FeatureEntryDataFile
```

Constructors

Name	Description
FeatureEntryDataFile(string)	Initializes a new instance of the FeatureEntryDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
FeatureEntries	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.IRSensorData Class

Represents an IR Sensor Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class IRSensorData
```

Constructors

Name	Description
IRSensorData()	Initializes a new instance of theIRSensorData Class.
IRSensorData(stream)	Initializes a new instance of the IRSensorData Class using the supplied Stream object to read the data from a file or memory stream.
IRSensorData(string, int)	Initializes a new instance of the IRSensorData Class using the index of the Object entry in the supplied .ICD File.

Properties

Name	Description
FlareEffectiveness	The chance a flare will be effective against this Object.
FOV	The lateral scan limit of this Sensor from Center to Limit in Radians.
Gimbals	The vertical scan limit of this Sensor from Center to Limit in Radians.
GroundFactor	Effectiveness multiplier for ground units with this Sensor.
NominalRange	The effective Range of this Sensor in feet.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Object.

DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.IRSensorDataFile Class

Represents an IRSensor Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class IRSensorDataFile
```

Constructors

Name	Description
IRSensorDataFile(string)	Initializes a new instance of the IRSensorDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
IRSensors	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.ObjectiveData Class

Represents a capturable Objective Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class ObjectiveData
```

Constructors

Name	Description
ObjectiveData()	Initializes a new instance of the ObjectiveData Class.
ObjectiveData(stream)	Initializes a new instance of the ObjectiveData Class using the supplied Stream object to read the data from a file or memory stream.
ObjectiveData(string, int)	Initializes a new instance of the ObjectiveData Class using the index of the Object entry in the supplied .OCD File.

Properties

Name	Description
CTIndex	CT Index of this Objective.
Name	Name of this Objective.
DataRate	Rate at which this Objective Updates.
DeagDistance	The distance which Units will Deaggregate from this Objective.
ParentIndex	The Parent Index associated with this Objective
DetectionRange	The Range at which Different Objects can detect this Objective.
DamageModel	Exposes access to the underlying DamageModel Structure for this Objective.
IconIndex	The Icon Index of this Objective.
FeatureCount	The number of Features associated with this Objective.
RadarVehicle	The Class ID of the Radar Vehicle attached to this Objective.

FirstFeatureIndex	The index of the first Feature assigned to this Objective.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.ObjectiveDataFile Class

Represents an Objective Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class ObjectiveDataFile
```

Constructors

Name	Description
ObjectiveDataFile(string)	Initializes a new instance of the ObjectiveDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
Objectives	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.PointData Class

Represents a Point Data Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class PointData
```

Constructors

Name	Description
PointData	Initializes a new instance of the PointData Class.
PointData(stream)	Initializes a new instance of the PointData Class using the supplied Stream object to read the data from a file or memory stream.
PointData(string, int)	Initializes a new instance of the PointData Class using the index of the Object entry in the supplied .PD File.

Properties

Name	Description
Flags	The Flags associated with this Point Object.
Offset	The 2D Offset of this Point from the Feature it is attached to.
Type	The type of Point this Point Object represents.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

This class is deprecated. PointEXData should be used instead.

F4BMS.Database.PointDataFile Class

Represents a Point Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class PointDataFile
```

Constructors

Name	Description
PointDataFile(string)	Initializes a new instance of the PointDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
Points	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.PointEXData Class

Represents a Point Data Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class PointEXData
```

Constructors

Name	Description
PointEXData()	Initializes a new instance of the PointEXDataData Class.
PointEXData(stream)	Initializes a new instance of the PointEXData Class using the supplied Stream object to read the data from a file or memory stream.
PointEXData(string, int)	Initializes a new instance of the PointEXData Class using the index of the Object entry in the supplied .PDX File.

Properties

Name	Description
Offset	3D Offset point of this Point from the center of the Object it belongs to.
MaxHeight	The Max Height in feet of this Object.
MaxWidth	The Max Width in feet of this Object.
MaxLength	The Max Length in feet of this Object.
PointType	Exposes access to the underlying PointType enumeration for this Point.
Flags	The Flags associated with this Point.
RootPtr	The Root Pointer for this Point. Deprecated in File Structure, should be null.
BranchPtr	The Branch Pointer for this Ppoint. Deprecated in File Structure, should be null.

Methods

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.PointEXDataFile Class

Represents a PointEX Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class PointEXDataFile
```

Constructors

Name	Description
PointEXDataFile(string)	Initializes a new instance of the PointEXDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
ExtendedPoints	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.PointHeaderData Class

Represents a Point Header Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class PointHeaderData
```

Constructors

Name	Description
PointHeaderData()	Initializes a new instance of thePointHeaderData Class.
PointHeaderData(stream)	Initializes a new instance of the PointHeaderData Class using the supplied Stream object to read the data from a file or memory stream.
PointHeaderData(string, int)	Initializes a new instance of the PointHeaderData Class using the index of the Object entry in the supplied .PHD File.

Properties

Name	Description
ObjectiveID	The ID of the Objective this PointHeader belongs to.
Type	Exposes access to the underlying PointListType enumeration.
Count	The number of points in this PointHeader.
Children	The child features associated with this PointHeader.
DataIndex	Unused Data Index of the Point Header.
MiscData	Miscellaneous Data associated with this PointHeader.
SinHeading	The Sin (Y) Component of the Heading for this PointHeader.
CosHeading	The Cos (X) Component of the Heading for this PointHeader.
FirstChild	The index to the first child Point in this PointHeader.
TextureID	The Texture ID used with this PointHeader.

RunwayIndex	The Runway Index relative to the Local Objective.
ApproachType	The type of approach to use if this is a Runway.
NextHeader	The next Point in the Header.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.PointHeaderDataFile Class

Represents a Point Header Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class PointHeaderDataFile
```

Constructors

Name	Description
PointHeaderDataFile(string)	Initializes a new instance of the PointHeaderDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
PointHeaders	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.RadarData Class

Represents a Radar Sensor in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class RadarData
```

Constructors

Name	Description
RadarData()	Initializes a new instance of the RadarData Class.
RadarData(stream)	Initializes a new instance of the RadarData Class using the supplied Stream object to read the data from a file or memory stream.
RadarData(string, int)	Initializes a new instance of the RadarData Class using the index of the Object entry in the supplied .RCD File.

Properties

Name	Description
RWRSound	The index for the Sound to play when the Radar is detected on an RWR.
RWRSymbol	The index of the symbol to display when the Radar is detected by an RWR.
RadarIndex	The index in the Radar Class Table.
HighLethality	The Lethality Value for High Altitude engagements.
LowLethality	The Lethality Value for Low Altitude engagements.
NominalRange	The effective range of the Radar.
BeamAngle	The angle in radians from the center to the limit of the beam.
ScanAngle	The angle in radians from the center to the limit of the lateral scan limit.
SweepRate	The angle in radians the beam sweeps every second.
CoastTime	The number of ms the Radar will allow a track to coast before it drops the track.

LookDownPenalty	The penalty applied when the radar is in a lookdown mode.
JammingPenalty	The effectiveness of a jammer against this radar.
NotchPenalty	The effectiveness of a notch maneuver against this radar.
NotchSpeed	The speed which a notch maneuver will be effective against this radar.
ChaffEffectiveness	The effectiveness of a Chaf against this radar.
Flags	The Flags associated with this radar.

Methods

Interface	**Description**
Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.RadarDataFile Class

Represents a Radar Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class RadarDataFile
```

Constructors

Name	Description
RadarDataFile(string)	Initializes a new instance of the RadarDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
Radars	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.RocketData Class

Represents a Rocket System Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class RocketData
```

Constructors

Name	Description
RocketData()	Initializes a new instance of the RocketData Class.
RocketData(stream)	Initializes a new instance of the RocketData Class using the supplied Stream object to read the data from a file or memory stream.
RocketData(string, int)	Initializes a new instance of the RocketData Class using the index of the Object entry in the supplied .RKT File.

Properties

Name	Description
RocketPodID	The Weapon Index of the Rocket Pod Component.
RocketWeaponID	The Weapon Index of the Rocket Weapon Component.
RocketCount	The number of Rockets in the Rocket Pod.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.RocketDataFile Class

Represents a Rocket Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class RocketDataFile
```

Constructors

Name	Description
RocketDataFile(string)	Initializes a new instance of the RocketDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
Rockets	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.RWRData Class

Represents a Radar Warning Receiver Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class RWRData
```

Constructors

Name	Description
RWRData()	Initializes a new instance of the RWRData Class.
RWRData(stream)	Initializes a new instance of the RWRData Class using the supplied Stream object to read the data from a file or memory stream.
RWRData(string, int)	Initializes a new instance of the RWRData Class using the index of the Object entry in the supplied .RWD File.

Properties

Name	Description
NominalRange	The effective range of this RWR Object.
TopScanAngle	The receive angle of this RWR from center to the Top Limit.
BottomScanAngle	The receive angle of this RWR from center to the Bottom Limit.
LeftScanAngle	The receive angle of this RWR from center to the Left Limit.
RightScanAngle	The receive angle of this RWR from center to the Right Limit.
Flags	The flags associated with this RWR.

Methods

Interface	Description

Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.RWRDataFile Class

Represents an RWR Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class RWRDataFile
```

Constructors

Name	Description
RWRDataFile(string)	Initializes a new instance of the RWRDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
RWRs	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.SimWeaponsData Class

Represents a Sim Weapon Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Implements

F4BMS.Database.ICTEntry

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class SimWeaponsData
```

Constructors

Name	Description
SimWeaponsData()	Initializes a new instance of the SimWeaponsData Class.
SimWeaponsData(stream)	Initializes a new instance of the SimWeaponsData Class using the supplied Stream object to read the data from a file or memory stream.
SimWeaponsData(string, int)	Initializes a new instance of the SimWeaponsData Class using the index of the Object entry in the supplied .SWD File.

Properties

Name	Description
Flags	The flags associated with this Weapon.
DragCoefficient	The Drag Coefficient used in AFM calculations.
Weight	The weight of the Weapon used in AFM calculations
Area	The surface area of the weapon used in AFM calculations.
EjectionXVector	The X Speed Vector when the Weapon is ejected.
EjectionYVector	The Y Speed Vector when the Weapon is ejected.
EjectionZVector	The Z Speed Vector when the Weapon is ejected.
Mnemonic	The text displayed for the weapon in the SMS.
WeaponClass	The General Class of the Weapon.
Domain	The employment domain of the Weapon.

WeaponType	The General Type of the Weapon.
WeaponDataIndex	The Index of the Weapon in the Weapon Class Table.

Methods

Interface	**Description**
Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.SimWeaponsDataFile Class

Represents a SimWeapons Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class SimWeaponsDataFile
```

Constructors

Name	Description
SimWeaponsDataFile(string	Initializes a new instance of the SimWeaponsDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
SimWeapons	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.SquadronStoresData Class

Represents a Squadron Stores Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class SquadronStoresData
```

Constructors

Name	Description
SquadronStoresData()	Initializes a new instance of theSquadronStoresData Class.
SquadronStoresData(stream)	Initializes a new instance of the SquadronStoresData Class using the supplied Stream object to read the data from a file or memory stream.
SquadronStoresData(string, int)	Initializes a new instance of the SquadronStoresData Class using the index of the Object entry in the supplied .SSD File.

Properties

Name	Description
WeaponList	Collection of bytes to indicate the number of the corresponding WeaponIndex in this
UnlimitedGunID	The WeaponID of a Gun Ammunition type which will always be available to the Squadron.
UnlimitedAG	A WeaponID of an AG Munition that will always be available to the Squadron.
UnlimitedAA	A WeaponID of an AA Munition that will always be available to the Squadron.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Object.

DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.SquadronStoresDataFile Clas

Represents a SquadronStores Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class SquadronStoresDataFile
```

Constructors

Name	Description
SquadronStoresDataFile(str	Initializes a new instance of the SquadronStoresDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
SquadronStores	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.UnitData Class

Represents a Unit Object in the Class Table Database. Units can be Squadrons, Flights, Packages, Battalions, or Naval Units.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class UnitData
```

Constructors

Name	Description
UnitData()	Initializes a new instance of the UnitData Class.
UnitData(Stream)	Initializes a new instance of the UnitData Class using the supplied Stream object to read the data from a file or memory stream.
UnitData(string, int)	Initializes a new instance of the UnitData Class using the index of the Object entry in the supplied .UCD File.

Properties

Name	Description
CTIndex	The Class Table Index of this Unit.
Elements	A Collection of Class IDs identifying the child units of this Unit Object.
Flags	The Unit Flags associated with this Unit.
Name	The Name of this Unit.
MovementType	The Movement Domain of this Unit.
Speed	The movement speed of this unit used in ATO planning.
MaxRange	The max range of this Unit used in ATO planning.
FuelCapacity	The max Fuel Capacity of this Unit.
FuelBurn	The Fuel Burn Rate of this Unit.
Roles	Exposes the underlying RolesModel structure used to identify the Roles this Unit can

MainRole	The Main Role this unit is tasked to perform.
HitChance	Exposes the underlying StatsModel Structure to identify the HitChance of this object
Strength	Exposes the underlying StatsModel Structure to identify the Strength of this object
Range	Exposes the underlying StatsModel Structure to identify the Range this object against specific Domains.
DetectionRange	Exposes the underlying StatsModel Structure to identify the Range which this Object is detected by specific Domains.
Damage	Exposes the underlying DamageModel Structure.
RadarUnit	The Class Table Index of the Radar Unit assigned to this Unit.
SquadronStores	The SquadronStores ID used by this Unit.
Icon	The Icon used on the Campaign Map by this Unit.

Methods

Interface	**Description**
Print()	Returns a formatted string with basic information about this AircraftData Object.
DebugPrint()	Returns a formatted string with advanced information about this AircraftData Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.UnitDataFile Class

Represents a Unit Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class UnitDataFile
```

Constructors

Name	Description
UnitDataFile(string)	Initializes a new instance of the UnitDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
Units	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.VisualSensorData Class

Represents a Visual Sensor Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class VisualSensorData
```

Constructors

Name	Description
VisualSensorData()	Initializes a new instance of the VisualSensorData Class.
VisualSensorData(stream)	Initializes a new instance of the VisualSensorData Class using the supplied Stream object to read the data from a file or memory stream.
VisualSensorData(string, int)	Initializes a new instance of the VisualSensorData Class using the index of the Object entry in the supplied .VSD File.

Properties

Name	Description
NominalRange	The effective range of this Sensor Object.
TopScanAngle	The receive angle of this Sensor from center to the Top Limit.
BottomScanAngle	The receive angle of this Sensor from center to the Bottom Limit.
LeftScanAngle	The receive angle of this Sensor from center to the Left Limit.
RightScanAngle	The receive angle of this Sensor from center to the Right Limit.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Object.

DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.VisualSensorDataFile Class

Represents a VisualSensor Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class VisualSensorDataFile
```

Constructors

Name	Description
VisualSensorDataFile(string	Initializes a new instance of the VisualSensorsDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
VisualSensors	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.VehicleData Class

Represents an Aircraft Object in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class VehicleData
```

Constructors

Name	Description
VehicleData()	Initializes a new instance of the VehicleData Class.
VehicleData(Stream)	Initializes a new instance of the VehicleData Class using the supplied Stream object to read the data from a file or memory stream.
VehicleData(string, int)	Initializes a new instance of the VehicleData Class using the index of the Object entry in the supplied .VCD File.

Properties

Name	Description
CTIndex	The Class Table Index of this Vehicle.
HitPoints	The amount of damage required to destroy this vehicle.
Flags	The flags associated with this vehicle.
Name	The name of this vehicle.
Cockpit	The Cockpit associated with this Vehicle.
RCS	The Radar Cross Signature of this Vehicle.
MaxWeight	The Max Operating Weight of this Vehicle.
EmptyWeight	The Empty Operating Weight of this Vehicle.
FuelWeight	The Max Fuel Weight of this Vehicle.
FuelFlow	The Fuel Flow in lbs/min of this Vehicle.

EngineSound	The index of the Sound to use for the Engine.
HighAlt	The Max Operating Altitude of this Vehicle.
LowAlt	The Min Operating Altitude of this Vehicle.
CruiseAlt	The Standard Cruise Operating Altitude of this Vehicle.
MaxSpeed	The Max Speed of this Vehicle.
RadarType	The Radar ID used by this Vehicle.
CrewMembers	The number of crew members assigned to this Vehicle.
RackFlags	The byte flags that indicate if a Hardpoint is a Rack or not.
RackVisibleFlags	The byte flags that indicate if a Hardpoint is visible or not.
CallsignIndex	The Callsign Index.
HitChance	Exposes the underlying StatsModel Structure used during combat calculations.
Strength	Exposes the underlying StatsModel Structure used during combat calculations.
Range	Exposes the underlying StatsModel Structure used during combat calculations.
DetectionRange	Exposes the underlying StatsModel Structure used during combat calculations.
WeaponIndex	The Weapon ID of the Weapon attached to Hardpoint.
WeaponCount	The Max number of Weapons attached to a Hardpoint.
DamageModel	Exposes the underlying DamageModel Structure used in damage calculations.
IRSignature	The IR Signature of this Vehicle.
VISSignature	The Visual Signature of this Vehicle.

Methods

Interface	**Description**
Print()	Returns a formatted string with basic information about this AircraftData Object.
DebugPrint()	Returns a formatted string with advanced information about this AircraftData Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.VehicleDataFile Class

Represents a Vehicle Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class VehicleDataFile
```

Constructors

Name	Description
VehicleDataFile(string)	Initializes a new instance of the VehicleDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
Vehicles	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.WeaponData Class

Represents the specific characteristics and statistics of a Weapon in the Class Table.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class WeaponData
```

Constructors

Name	Description
WeaponData()	Initializes a new instance of theWeaponData Class.
WeaponData(stream)	Initializes a new instance of the WeaponData Class using the supplied Stream object to read the data from a file or memory stream.
WeaponData(string, int)	Initializes a new instance of the WeaponData Class using the index of the AircraftData entry in the supplied .WCD File.

Properties

Name	Description
CTIndex	The Class Table Index of the Weapon Object.
Strength	Amount of damage done by this weapon. Also indicates effectiveness of non-kinetic weapons such as ECM.
DamageType	The type of damage done by this Weapon.
Range	The effective range of this weapon.
Flags	The Flags associated with this Weapon.
Name	The Name of this Weapon.
HitChance	Exposes the underlying StatsModel for Hit Chance used in combat calculations.
FireRate	The rate of fire for this Weapon.
Rarety	The rarety of this weapon in the campaign engine.

GuidanceType	Exposes the GuidanceType enumeration of this Weapon.
Collective	Identifies if this Weapon is part of a collective.
BulletTTL	The time a Bullet will last after it is fired.
RackGroup	The RackGroup ID this Weapon uses.
Weight	The weight of this Weapon.
DragIndex	The drag index of this Weapon used in AFM calculations.
BlastRadius	The blast radius of this Weapon. Also used to identify the number of flares in a Flare Ejector.
RadarType	The Radar Type used by this Weapon.
SimDataIndex	The SimWeaponDatawhich references this Weapon.
BulletRounds	The number of Bullets this Weapon can hold.

Methods

Interface	**Description**
Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.WeaponDataFile Class

Represents a Weapon Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class WeaponDataFile
```

Constructors

Name	Description
WeaponDataFile(string)	Initializes a new instance of the WeaponDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
Weapons	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.WeaponListData Class

Represents a Weapon List Object in the Class Table. Weapon List Objects are Hardpoints.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTEntry

Syntax

```
[ComVisibleAttribute(true)]
public class WeaponListData
```

Constructors

Name	Description
WeaponListData()	Initializes a new instance of the WeaponListData Class.
WeaponListData(stream)	Initializes a new instance of the WeaponListData Class using the supplied Stream object to read the data from a file or memory stream.
WeaponListData(string, int)	Initializes a new instance of the WeaponListData Class using the index of the AircraftData entry in the supplied .WLD File.

Properties

Name	Description
Name	The name of the Hardpoint.
WeaponIndex	The WeaponID attached to this Hardpoint.
WeaponCount	The Max number of Weapons attached to this Hardpoint.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Object.
DebugPrint()	Returns a formatted string with advanced information about this Object.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Database.WeaponListDataFile Class

Represents a WeaponList Data File in the Class Table Database.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Implements

F4BMS.Database.ICTFile

Syntax

```
[ComVisibleAttribute(true)]
public class WeaponListDataFile
```

Constructors

Name	Description
WeaponListDataFile(string)	Initializes a new instance of the WeaponListDataFile Class.

Properties

Name	Description
Count	Number of entries in the Database File
Name	File Name
FullName	File Name and Folder
LastModified	Time the underlying file was last modified
WeaponLists	Underlying CT Object Collection

Methods

Interface	Description
Create(string)	Creates an empty database file with the specified file name
ToXML(string)	Writes the data in XML format to the specified file name
Save(Stream)	Writes the data to the supplied stream object
Save()	Saves the data to the currently loaded file

SaveAs()	Saves the data to a new file with the specified file name
GetData()	Returns a collection of ICTEntry objects for use with the Interface

F4BMS.Database.ICTFile Interface

Interface to provide a common interface for CT DB Files.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public interface ICTFile
```

Properties

Name	Description
Count	Number of CT Data Objects in the File
Name	File Name
FullName	File Name and Folder
LastModified	Last Modified Time of File

Methods

Interface	Description
ToXML()	Save the data to an XML File
Save(Stream)	Save the Data to the supplied stream
Save()	Save the data to the currently loaded file
SaveAs(filename)	Save the data to the supplied file name
GetData()	Return a List of ICTEntry objects for use with the Interface

F4BMS.Database.ICTEntry Interface

Interface to provide a common interface for all CT DB Types.

Namespace: F4BMS.Database
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public interface ICTEntry
```

Methods

Interface	Description
Print()	Returns a formatted string with basic information about the underlying CT Object.
DebugPrint()	Returns a formatted string with advanced information about the underlying CT Object
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Header Namespace

Contains the types required to provide basic interaction with the Header File (KoreaObj.HDR). Individual classes are exposed to provide functionality, however it is highly reccommended to only interact with Header Data as a whole Header File (See F4BMS.Header.HeaderFile).

Classes

Class	Description
HeaderFile	Represents an entire Header File (KoreaObj.HDR).
LODHeader	Represents a LOD Header Entry. LOD Headers point to LOD Structures in the .LOD File.
Palette	Represents a 256 Color Palette used in the Graphics Engine.
ParentHeader	Represents a Parent Header Object. Parent Headers contain generic 3D space data metrics used to render and interact with an object in the 3D world.
Texture	A custom Texture Header object used by the Parent Header to reference Textures during Render.

F4BMS.Header.HeaderFile Class

Container Class to represent the entire collections of Objects in the KoreaObj.HDR file.

Namespace: F4BMS.Header
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class HeaderFile
```

Constructors

Name	Description
HeaderFile()	Initializes a new instance of the HeaderFile Class.
HeaderFile(Stream)	Initializes a new instance of the HeaderFile Class using the supplied Stream object to read the data from a file or memory stream.
HeaderFile(string)	Initializes a new instance of the HeaderFile Class using the index of the Object entry in the supplied .HDR File.

Properties

Name	Description
FileName	The name of the File Used to load the header data.
FileVersion	The File Version of the file the data was read from.
ColorCount	Read-Only Count of the Colors in the Header File.
DarkColorCount	Read-Only Count of the Dark Colors in the Header File.
PaletteCount	Read-Only Count of the Palettes in the Header File.
TextureCount	Read-Only Count of the Textures in the Header File.
LODHeaderCount	Read-Only Count of the LOD Headers in the Header File.
ParentCount	Read-Only Count of the Parent Headers in the Header File.
FileSize	Read-Only size of the Header Data.
Headers	Collection of Parent Headers.
Colors	Collection of Color Objects.
Palettes	Collection of Palettes.
Textures	Collection of Texture Headers.
LODHeaders	Collection of LOD Headers.

Methods

Interface	Description
GetUnusedLODs()	Returns a List of Int32 objects identifying unused LODs in the Header.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.Header.LODHeader Class

LODHeader Class to hold LOD Header Data. LOD Headers contain the File Offset of the LOD Tree in the LOD File.

Namespace: F4BMS.Header
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class LODHeader
```

Constructors

Name	Description
LODHeader()	Initializes a new instance of the LODHeader Class.
LODHeader(Stream)	Initializes a new instance of the LODHeader Class using the supplied Stream object to read the data from a file or memory stream.
LODHeader(int, short, short, int, int, int)	Initializes a new instance of the LODHeader Class. This will initialize to the 4.33 and below File Version.
LODHeader(int, short, short, int, long, long)	Initializes a new instance of the LODHeader Class.This will iniitalize to the 4.34 and above File Version.
LODHeader(LODHeader)	Initializes a new instance of the LODHeader Class.

Properties

Name	Description
ReferenceCount	Internal count of Graphics Engine references to this LOD.
OnOrder	Indicates if the LOD is ready or waiting to be drawn.
Flags	Flags associated with this LOD Header.
Pnode	The internal Pointer to a LOD structure after the Model is loaded into memory.
FileOffset	The offset to the start of the model in the LOD File.
FileSize	The Size of the model in the LOD File.

Methods

Interface	Description
GetLod()	Returns an F4BMS.LOD.LOD object from the File Offset of this LOD Header.

Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Header Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the HeaderFile Class.

F4BMS.Header.Palette Class

Custom Palette Class to represent the Palette Data in BMS.

Namespace: F4BMS.Header
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class Palette
```

Constructors

Name	Description
Palette()	Initializes a new instance of the Palette Class.
Palette(int[], int, int, int, int)	Initializes a new instance of the Palette Class.
Palette(stream, int)	Initializes a new instance of the Palette Class using the index of the Object entry.

Properties

Name	Description
RefCount	Internal count of the references to the Palette in memory. Unused in File Structure.
PaletteHandle	Internal reference to the Palette Handle. Unusued in File Structure.
PaletteColors	Collection of Int32 objects to represent the colors of the palette in numeric format.
Colors	Collection of Color Objects used to represent the colors of the Palette.

Methods

Interface	Description
Export(string)	Exports the Palette to a Microsoft Palette Format (PAL) file.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Header Namespace are provided for functionality. It is highly recommended to only use the

types in this Namespace through the HeaderFile Class.

F4BMS.Database.ParentHeader Class

Generic Parent Object data and LOD File references.

Namespace: F4BMS.Header
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class ParentHeader
```

Constructors

Name	Description
ParentHeader()	Initializes a new instance of the ParentHeader Class.
ParentHeader(stream)	Initializes a new instance of the ParentHeader Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
Radius	The Hit Box Radius of the Parent Object.
MinX	The Minimum X Cooridnate of the Parent Object.
MaxX	The Maximum X Cooridnate of the Parent Object.
MinY	The Minimum Y Cooridnate of the Parent Object.
MaxY	The Maximum Y Cooridnate of the Parent Object.
MinZ	The Minimum Z Cooridnate of the Parent Object.
MaxZ	The Maximum Z Cooridnate of the Parent Object.
LODRecordPointer	The file offset where this Parent Headers LOD Records exist.
SlotPointer	The file offset where this Parent Headers Slot Data exists.
TextureCount	The number of Textures this Parent Header uses.
DynamicCoordinateCount	The number of Dynamic Coordinates this Parent Header uses.
SwitchCount	The number of Slots this Parent Header uses.
DOFCount	The number of DOFs this Parent Header uses.
SlotCount	The number of Slots this Parent Header uses.
LODCount	The number of LODs this Parent Header uses.
ParentID	The index of this Parent Header.

Slots	Collection of 3D Point Objects to represent the Slot Data.
DynamicCoordinates	Colleciton of 3D Point Objects to represent the Dynamic Coordinate Data.
LODReferences	The Collection of LODReference Objects contained in this Parent Header.

Methods

Interface	Description
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
SaveSlots(stream)	Writes the Serializable Slot data to the supplied Stream Object.
SaveDynamics(stream)	Writes the Serializable Dynamic Coordinate data to the supplied Stream Object.
SaveLODs(stream)	Writes the Serializable LOD Reference data to the supplied Stream Object.

Structures

Structures	Description
LODReference	Represents the LOD Reference used by the Parent Header to reference a LOD Index in the Header File.

Remarks

Individual classes in the Header Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the HeaderFile Class.

F4BMS.Header.Texture Class

Container Class to hold the Texture Data.

Namespace: F4BMS.Header
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class Texture
```

Constructors

Name	Description
Texture()	Initializes a new instance of the Texture Class.
Texture(stream)	Initializes a new instance of the Texture Class using the supplied Stream object to read the data from a file or memory stream.
Texture(int, int, int, int, int, int, int, int, int, int)	Initializes a new instance of the Texture Class.
Texture(long, long, int, int, int, int, int, int, int, int)	Initializes a new instance of the Texture Class.

Properties

Name	Description
FileOffset	The File offset of the Texture data.
FileSize	The size of the Texture data.
Dimensions	The dimensions of the Image represented by the Texture.
ImageData	The Image pixel data.
PaletteData	The Palette used by the Texture.
Flags	The Flags associated with this Texture.
ChromaKey	The Chroma Key used by this Texture.
TextureHandle	The internal Handle used to locate the Texture in Memory.
PaletteID	The Palette ID used by the Texture.
ReferenceCount	Internal Reference count used to track the Texture in Memory.

Methods

Methods

Interface	Description
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Header Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the HeaderFile Class.

F4BMS.LOD Namespace

Contains the types required to provide basic interaction with the LOD Database (KoreaObj.LOD). The F4BMS.LOD Namespace contains two Namespaces; F4BMS.LOD.Nodes and F4BMS.LOD.Polys to provide interaction with the BSP Node and Polygon objects in the LOD File.

Namespaces

Namespaces	Description
F4BMS.LOD.Nodes	Contains the types required to provide basic interaction with BSP Nodes in the LOD Database.
F4BMS.LOD.Polys	Contains the types required to provide basic interaction with the Primitive and Polygon objects in the LOD Database.

Classes

Namespaces	Description
LOD	A LOD Structure used to render a 3D Model in F4:BMS.
LODFile	A collection of LOD Objects which represent an entire KoreaObj.LOD file.
Normal	A Custom wrapper of a 3D Point used to represent a 3D Normal Vector.

Enumerations

Enumeration	Description
NodeType	Enumerates the BSP Node types used in the LOD Structures.

F4BMS.LOD.LOD Class

Represents a complete BSP Tree LOD Model required for the 3D renderer in F4BMS.

Namespace: F4BMS.LOD
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class LOD
```

Constructors

Name	Description
LOD()	Initializes a new instance of the LOD Class.
LOD(string, long)	Initializes a new instance of the LOD Class using the supplied file name and a file offset to start reading from.
LOD(stream, string)	Initializes a new instance of the LOD Class using the supplied Stream object and a file name to read the data from a file or memory stream.
LOD(stream)	Initializes a new instance of the LOD Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
NodeCount	Read-Only count of the Nodes in this LOD Model.
Size	The size of this Model in the LOD File.
RootPointer	The Root Pointer Offset of the LOD Model.
Nodes	Collection of IBSPNode Objects that make up this Model.
NodeTypes	List of Int32 Objects which denote the Node Types in this Model.
Filename	The File Name this LOD Model was read from.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this LOD Object.
DebugPrint()	Returns a formatted string with advanced information about this LOD Object.

Save(Stream)	Writes the Serializable data to the supplied Stream Object.

F4BMS.LOD.LODFile Class

Container Class to hold an Entire LOD File.

Namespace: F4BMS.LOD
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class LODFile
```

Constructors

Name	Description
LODFile()	Initializes a new instance of the LODFile Class.
LODFIle(string, string)	Initializes a new instance of the LODFIle Class using the supplied LOD and HDR File Names.

Properties

Name	Description
LODCount	Read-Only count of the number of LOD Models.
NodeCount	Read-Only count of the Total Number of Nodes.
FileSize	The Size of the LOD Model Collection.
FileName	The File Name used to load the LOD Models.
LODS	Collection of F4BMS.LOD.LOD Objects.

Remarks

The LOD file with no overhead is approximately 2GB in size. This function should only be used on 64-Bit Operating Systems with at least 8GB of RAM installed. This function builds the Node Objects, checks for data errors, and reorganises the Nodes into the most efficient order during the load. This process is time consuming, and loading an entire file may take more than 5 minutes when done through this method.

F4BMS.LOD.Normal Class

Represents an I-, J-, K- Normal vector for 3D light mapping.

Namespace: F4BMS.LOD
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class Normal
```

Constructors

Name	Description
Normal()	Initializes a new instance of the Normal Class.
Normal(double, double, double)	Initializes a new instance of the Normal Class using the supplied values.
Normal(System.Windows.Media.Media3d.Point3d)	Initializes a new instance of the Normal Class using the supplied Point3D Object.

Properties

Name	Description
I	Normal I Vector
J	Normal J Vector
K	Normal K vector

F4BMS.LOD.Nodes Namespace

Contains the types and methods required to interact with the BSP Node trees used to render models in F4:BMS. The Nodes are exposed individually to provide interaction with specific node elements, however it is highly recommended to start all Node interactions from a LOD Class (See F4BMS.LOD.LOD).

Classes

Namespaces	Description
BNode	A Basic Node Structure with file structure navigation data.
BSubTreeNode	A Node containing a Subtree of Nodes, and the Coordinate data associated with rendering the child Nodes.
BRootNode	The Root Node used to mark the beginning of a LOD Structure.
BSlotNode	A Container Node used to identify attachment and hard point positions on a model.
BDOFNode	A Node that identifies an animation or moveable part of a model.
BSwitchNode	A Node that identifies a selectable group of child nodes to render based on a condition.
BSplitterNode	A Node used to identify a split in the LOD Structure. May be used in culling, but distance rendering functions are depricated in F4:BMS.
BPrimitiveNode	A Node containing a Primitive Object which can be sent to the Graphics Engine.
BLitPrimitiveNode	A Node containing a set of Primitive Objects which can be combined with lighting effects to generate a Primitive Object with Lighting data..
BCulledPrimitiveNode	A Node containing a Primitive Object with culling data used in rendering.
BSpecialXFormNode	A Node containing a subtree of Nodes which can be transformed in some way during rendering.
BLightStringNode	A Node containing a series of lights or lit primitives to provide lighting information to the renderer.
BTransNode	A Node that identifies a moveable part or animation which can also have a Transformation applied to it.
BScaleNode	A Node containing scaling data which can be applied to a collection of Nodes.
BXDofNode	An enhanced DOF Node with additional parameters and options.
BXSwitchNode	An enhanced Switch Node with additional parameters and options.
BRenderControlNode	A Node used to control render characteristics within the Graphics Engine.
BCullNode	A Culling Node used in distance calculations.

Abstract Classes

Abstract Class	Description
BSPNode	An abstract base class for all BSP Nodes to inherit.

Interfaces

Interface	Description
IBSPNode	An interface class to a provide common interface for all the BSP Node Types.

Enumerations

Enumeration	Description
TransformType	Enumerates the Transformation Types used in Transform Nodes.
VFT	Enumerates the VFT Entries found in BSP Nodes. VFTs identify Node Types.
DOFID	Enumerates the DOF Type IDs
SimpleDOFID	Enumerates the Simple Model DOF Type IDs
SwitchID	Enumerates the Switch IDs
RenderControlType	Enumerates the types of RenderControl Operations available
RenderDOFMathType	Enumerates the Math Operations a RenderControlNode can perform

F4BMS.LOD.Nodes.IBSPNode Interface

BSP Node Interface to provide a common interface for all BSP Node Types.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public interface IBSPNode
```

Properties

Name	Description
NodeType	The NodeType enumeration of this BSPNode.
Sibling	The Sibling Pointer of this BSP Node.
NodeAddress	The File Offset of this BSP Node.
FileSize	The size of the generic data of this BSP Node.
VFT	The VFT Enumeration of this BSP Node used to identify the Node Type from File.
TotalSize	The Total Size of this BSP Node including any Sub Trees, Polygons, Coordinates, or other referenced items.
NodeCount	Read-Only count of the number of Nodes contained in this Node including any Sub

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

F4BMS.LOD.Nodes.BSPNode Class

Abstract base class for all BSP Nodes to extend.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public abstract class BSPNode
```

Properties

Name	Description
NodeType	Exposes the underlying NodeType enumeration.
FileSize	The size of the generic data of this BSP Node.
TotalSize	The Total Size of this BSP Node including any Sub Trees, Polygons, Coordinates, or
NodeAddress	The File Offset of this BSP Node.
NodeCount	Read-Only count of the number of Nodes contained in this Node including any Sub

F4BMS.LOD.Nodes.BNode Class

Generic Base class Node which every Node inherits.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNod

Implements

F4BMS.LOD.Nodes.IBSPNod

Syntax

```
[ComVisibleAttribute(true)]
public class BNode : BSPNode, IBSPNode
```

Constructors

Name	Description
BNode()	Initializes a new instance of the BNode Class.
BNode(int, int)	Initializes a new instance of the BNode Class.
BNode(stream, long)	Initializes a new instance of the BNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
VFT	The VFT Enemuration of this Node.
Sibling	A Pointer to the Next node to be processed when reading or rendering this BSP Model.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BSubTreeNode Class

Container Node which holds a collection of child Nodes, Coordinate and Dynamic Coordinate Data, and Normal information for a series of Nodes in the BSP Model.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BSubTreeNode : BNode, IBSPNode
```

Constructors

Name	Description
BSubTreNode()	Initializes a new instance of the BSubTreeNode Class.
BSubTreeNode(stream, long)	Initializes a new instance of the BSubTreeNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
Coordinates	Collection of 3D Point Objects representing the 3D Vertex Data for SubTree Nodes.
DynamicCoordinates	Collection of 3D Point Objects representing the 3D Vertex Data for Movable SubTree
Normals	Collection of Normal Objects used for lighting data on the SubTree Nodes.
SubTreePointer	A Pointer to the collection of SubTree Nodes which make up this portion of the Model.
SubTree	Collecion of BSP Nodes which make up this portion of the BSP Model.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.

DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BRootNode Class

Container Node which holds a collection of child Nodes, Coordinate and Dynamic Coordinate Data, and Normal information for a series of Nodes in the BSP Model. Root Nodes identify the start of a BSP Tree.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode
F4BMS.LOD.Nodes.BSubTreeNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BRootNode : BSubTreeNode, IBSPNode
```

Constructors

Name	Description
BRootNode()	Initializes a new instance of the BRootNode Class.
BRootNode(stream, long)	Initializes a new instance of the BRootNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
ScriptID	Animation Script ID.
TextureCount	Read-Only count of Textures assigned to this Model.
Textures	Collection of the Texture Indices for this Root Node.
TexturePointer	The File Offset hwere the Texture Indices exist in the File.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.

DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BSlotNode Class

Node which indicates an attachment or Hard Point location of a Model.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BSlotNode : BNode, IBSPNode
```

Constructors

Name	Description
BSlotNode()	Initializes a new instance of the BSlotNode Class.
BSlotNode(stream, long)	Initializes a new instance of the BSlotNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
RotationMatrix	The Rotation and Transformation matrix used in 3D Model transforms.
RotationPoint	The Point where the Attachment Point or Hardpoint attaches to the Model.
SlotID	The Slot Identifier of this Slot.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BDOFNode Class

Node which indicates an animation in a LOD Model.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode
F4BMS.LOD.Nodes.BSubTreeNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BDOFNode : BSubTreeNode, IBSPNode
```

Constructors

Name	Description
BDOFNode()	Initializes a new instance of the BDOFNode Class.
BDOFNode(stream, long)	Initializes a new instance of the BDOFNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
RotationMatrix	The Rotation and Transformation matrix used in 3D Model transforms.
RotationPoint	The Point where the Attachment Point or Hardpoint attaches to the Model.
DOFID	The DOF Identifier of this Animation.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.

Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BSwitchNode Class

Node which indicates a series of potential SubTrees in a LOD Model selected by a Switch.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BSwitchNode : BNode, IBSPNode
```

Constructors

Name	Description
BSwitchNode()	Initializes a new instance of the BSwitchNode Class.
BSwitchNode(stream, long)	Initializes a new instance of the BSwitchNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
ChildCount	Read-Only number of Child SubTrees in this Switch.
SubTreePointer	File offset of the list of Child Subtrees
SwitchID	The Switch Identifier the trees in this Node comprise.
ChildrenPointers	Collection of Pointers that identify the SubTrees of this SwitchNode.
SubTree	Collecion of BSP Nodes which make up this portion of the BSP Model.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.

DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BSplitterNode Class

Node which indicates a Split in the BSP Node Processing Tree. Contains child trees available to be drawn. Can be used to determine Culling Order in DirectX. Distance Culling is deprecated in the F4:BMS implementation of BSP Nodes.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BSplitterNode : BNode, IBSPNode
```

Constructors

Name	Description
BSplitterNode()	Initializes a new instance of the BPlitterNode Class.
BSplitterNode(stream, long)	Initializes a new instance of the BSplitterNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
A	I- Component of a Face Angle Vector.
B	J- Component of a Face Angle Vector.
C	K- Component of a Face Angle Vector.
D	Distance from the Center of the Model used for Culling Calculations.
FrontNodePointer	The File Offset of the Front Node SubTree.
BackNodePointer	The File Offset of the Back Node SubTree
FrontNode	Collection of IBSPNode Objects representing the Front SubTree.
BackNode	Collection of IBSPNode Objects representing the Back SubTree.

Methods

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BPrimitiveNode Class

Container Node for an unlit Primitive Object in the BSP Tree.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BPrimitiveNode : BNode, IBSPNode
```

Constructors

Name	Description
BPrimitiveNode()	Initializes a new instance of the BPrimitiveNode Class.
BPrimitiveNode(stream, long)	Initializes a new instance of the BPrimitiveNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
PolygonType	Exposes the underlying PolyType enumeration of the Polygon Object.
Primitive	The F4BMS.LOD.Polys.IPrimitive Object referenced in this Node.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and

UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BLitPrimitiveNode Class

Container Node for a lit Primitive Object in the BSP Tree.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BLitPrimitiveNode : BNode, IBSPNode
```

Constructors

Name	Description
BLitPrimitiveNode()	Initializes a new instance of the BLitPrimitiveNode Class.
BLitPrimitiveNode(stream, long)	Initializes a new instance of the BLitPrimitiveNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
PolygonType	Exposes the underlying PolyType enumeration of the Polygon Object.
FrontPrimitive	The F4BMS.LOD.Polys.IPrimitive Object referenced as the Front Side of the Lit Polygon
BackPrimitive	The F4BMS.LOD.Polys.IPrimitive Object referenced as the Back Side of the Lit Polygon

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BCulledPrimitiveNode Class

Container Node for a Cullable Primitive Object in the BSP Tree. Cullable Objects can be used to limit rendering to screen space, Z-Bias, and View Port calculations.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BCulledPrimitiveNode : BNode, IBSPNode
```

Constructors

Name	Description
BCulledPrimitiveNode()	Initializes a new instance of the BCulledPrimitiveNode Class.
BCulledPrimitiveNode(stream, long)	Initializes a new instance of the BCulledPrimitiveNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
PolygonType	Exposes the underlying PolyType enumeration of the Polygon Object.
Primitive	The F4BMS.LOD.Polys.IPrimitive Object referenced as the Polygon in this Node.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and

UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BSpecialXFormNode Class

Used as a container to apply a Special type of 3D transformation to a SubTree of Nodes.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BSpecialXFormNode : BNode, IBSPNode
```

Constructors

Name	Description
BSpecialXFormNode()	Initializes a new instance of the BSpecialXFormNode Class.
BSpecialXFormNode(stream, long)	Initializes a new instance of the BSpecialXFormNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
Coordinates	The 3D Coordinates associated with the SubTree Nodes contained in this Node.
CoordinateCount	Read-Only count of the number of Coordinates found in this Node.
CoordinatePointer	The File Offset to the Coordinates for this Node.
TransformType	Exposes access to the underlying TransformType enumeration of this
SubTreePointer	The File Offset where the SubTree Nodes exist.
SubTree	Collection of IBSPNodes that make up the SubTree of this Node.

Methods

Interface	Description

Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BLightStringNode Class

Used as a container to apply a specific lighting effecs to a series of Primitive Objects.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode
F4BMS.LOD.Nodes.BPrimitiveNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

[ComVisibleAttribute(true)]
public class BLightStringNodeNode : BPrimitiveNode, IBSPNode

Constructors

Name	Description
BLightStringNode()	Initializes a new instance of the BLightStringNode Class.
BLightStringNode(stream, long)	Initializes a new instance of the BLightStringNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
A	I- Component of a Normal Vector.
B	J- Component of a Normal Vector.
C	K- Component of a Normal Vector.
D	Light Intensity Component.
FrontRGBA	The Color Index of the Front RGBA Color.
BackRGBA	The Color Index of the Back RGBA Color.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BTransNode Class

Container Node to performa specific Transformation on a SubTree of Nodes.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode
F4BMS.LOD.Nodes.BSubTreeNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BTransNode : BSubTreeNode, IBSPNode
```

Constructors

Name	Description
BTransNode()	Initializes a new instance of the BScaleNode Class.
BTransNode(stream, long)	Initializes a new instance of the BScaleNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
DOFID	The DOF ID associated with this Transformation.
Min	The Minimum transformation scaling applied to the child SubTree.
Max	The Maximum transformation scaling applied to the child SubTree.
Multiplier	The Scaling Factor applied to the Child SubTree.
Future	The desired Scaling Factor to be applied to the SubTree.
Flags	Node Specific Flags applied during the Transformation.
TranslationPoint	The 3D Point where the SubTree transformation is centered.

Methods

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BScaleNode Class

Container Node to performa specific Scaling Transformation on a SubTree of Nodes.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode
F4BMS.LOD.Nodes.BSubTreeNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

[ComVisibleAttribute(true)]
public class BScaleNode : BSubTreeNode, IBSPNode

Constructors

Name	Description
BScaleNode()	Initializes a new instance of the BScaleNode Class.
BScaleNode(stream, long)	Initializes a new instance of the BScaleNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
DOFID	The DOF ID associated with this Transformation.
Min	The Minimum transformation scaling applied to the child SubTree.
Max	The Maximum transformation scaling applied to the child SubTree.
Multiplier	The Scaling Factor applied to the Child SubTree.
Future	The desired Scaling Factor to be applied to the SubTree.
Flags	Node Specific Flags applied during the Transformation.
TranslationPoint	The 3D Point where the SubTree transformation is centered.
ScalePoint	The 3D Point where the SubTree scaling transformation is centered.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BXDofNode Class

Extended DOF Node incorporating Flags to effect the DOF animation.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode
F4BMS.LOD.Nodes.BSubTreeNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BXDofNode : BSubTreeNode, IBSPNode
```

Constructors

Name	Description
BXDofNode()	Initializes a new instance of the BXDofNode Class.
BXDofNode(stream, long)	Initializes a new instance of the BXDofNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
DOFID	The DOF ID associated with this Transformation.
Min	The Minimum transformation scaling applied to the child SubTree.
Max	The Maximum transformation scaling applied to the child SubTree.
Multiplier	The Scaling Factor applied to the Child SubTree.
Future	The desired Scaling Factor to be applied to the SubTree.
Flags	Node Specific Flags applied during the Transformation.
TranslationPoint	The 3D Point where the SubTree transformation is centered.
RotationMatrix	The 3D Vector Math Matrix used to apply to the DOF animation.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BXSwitchNode Class

Extended Switch Node incorporating Flags to effect the Switch behavior.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BXSwitchNode : BNode, IBSPNode
```

Constructors

Name	Description
BXSwitchNode()	Initializes a new instance of the BXSwitchNode Class.
BXSwitchNode(stream, long)	Initializes a new instance of the BXSwitchNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
SwitchID	The Switch ID associated with this Transformation.
SubTreePointer	The File Offset where the List of SubTree Pointers exists.
ChildCount	Read-Only count of the number of SubTrees attached to this Switch.
ChildrenPointers	List of Int64 Objects indicated the File Offset of each of the SubTrees.
Flags	The Flags used in the processing of this Switch.
SubTrees	Collection of IBSPNodes representign the SubTrees attached to this Switch.

Methods

Interface	Description

Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Nodes.BRenderControlNode Class

Controller Node designed to effect the way a Tree or Model is rendered in DirectX.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BRenderControlNode : BNode, IBSPNode
```

Constructors

Name	Description
BRenderControlNode()	Initializes a new instance of the BRenderControlNode Class.
BRenderControlNode(stream, long)	Initializes a new instance of the BRenderControlNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
RenderControlType	Determines the type of Render data contained in the RenderControlNode
DataArgs	Data arguments available to RenderControl type 0.
Zbias	Zbias value used in RenderControl type 1.
MathMode	Type of Math calculations performed on the target Node. Available in RenderControl
Args	Math Mode arguments used in RenderControl type 2.
ResultType	Result type of Math Mode calculations avaialble in RenderControl type 2.
ResultID	The result type ID used in calculations for RenderControl type 2
MathIDs	The identifiers used in the calculations in RenderControl type 0 or 2.
MathValues	Math values used in calculations in RenderControl types 1 or 2.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

The BRenderControlNode functionality is not completely mapped. Functionality and availability of the Properties is based on the assumption of the RenderControlType property.

F4BMS.LOD.Nodes.BCullNode Class

Contains specific Culling data which can be applied to a SubTree or set of SubTrees in the BSP Render process.

Namespace: F4BMS.LOD.Nodes
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Nodes.BSPNode
F4BMS.LOD.Nodes.BNode

Implements

F4BMS.LOD.Nodes.IBSPNode

Syntax

```
[ComVisibleAttribute(true)]
public class BCullNode : BNode, IBSPNode
```

Constructors

Name	Description
BCullNode()	Initializes a new instance of the BCullNode Class.
BCullNode(stream, long)	Initializes a new instance of the BCullNode Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
A	I- Component of a Face Angle Vector.
B	J- Component of a Face Angle Vector.
C	K- Component of a Face Angle Vector.
D	Distance from the Center of the Model used for Culling Calculations.
FrontNodePointer	The File Offset of the Front Node SubTree.
BackNodePointer	The File Offset of the Back Node SubTree
FrontTree	Collection of IBSPNode Objects representing the Front SubTree.
BackTree	Collection of IBSPNode Objects representing the Back SubTree.

Methods

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
GetNodeTypes()	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Node used as a reference for all other Pointers.

Remarks

Individual classes in the Nodes Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys Namespace

Contains the types required to provide basic interaction with the Primitive and Polygon Objects in the LOD File Structure. Individual types are exposed to provide functionality, however it is highly recommended to only interact with Primitive and Polygon Objects through a LOD Class Structure (See F4BMS.LOD.LOD)

Classes

Namespaces	Description
PrimitivePoint	A Single Primitive Vertex point.
PrimitiveLine	A Series of Vertex Points to create a Primitive Line.
PrimitiveLightString	A collection of lighted Vertex Points to forma string of lights. This class is depricated and should not be used.
PolyFC	Flat Polygon Object.
PolyVC	Flat Gouraud Polygon Object.
PolyFCN	Flat Polygon Object with Lighting Information.
PovVCN	Flat Gouraud Polygon Object with Lighting Information.
PolyTexFC	Flat Textured Polygon Object.
PolyTexVC	Flat Textured Gouraud Polygon Object.
PolyTexFCN	Flat Textured Polygon Object with Lighting Information.
PolyTexVCN	Flat Textured Gouraud Polygon Object with Lighting Information.

Abstract Classes

Abstract Class	Description
Primitive	An abstract base class for all Primitive and Polygon Objects to inherit.
Polygon	An abstract base class for all Polygon Objects to inherit.

Interfaces

Interface	Description
IPrimitive	An interface class to a provide common interface for all the Primitive and Polygon Types.

Enumerations

Enumeration	Description
PrimType	Enumerates the Primitive Types available in the F4BMS LOD File.
PolyType	Enumerates the Polygon Types available in the F4BMS LOD File.

F4BMS.LOD.Polys.IPrimitive Interface

Interface for Primitive and Polygon Objects. Provides a common functionality to all Primitive and Polygon Objects.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public interface IPrimitive
```

Properties

Name	Description
PolygonType	Exposes access to the underlying PolyType enumeration to determine the type of Primitive or Polygon this Object represents.
Size	The size on disk of this Primitive Object.
PrimitiveAddress	The base File Offset of the Primitive Object.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
UpdateRoot(long)	Updates the Root Pointer of this Polygon used as a reference for all other Pointers.

F4BMS.LOD.Polys.Primitive Class

Abstract Base Primitive class with generic Primitive control and render data.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

[ComVisibleAttribute(true)]
public abstract class Primitive

Properties

Name	Description
PolygonType	Exposes the underlying PolyType enumeration to identify the type of Primitive or Polygon this Object represents.
VertexCount	Read-Only count of the Vertex Points associated with this Primitive.
VertexPointer	File Offset to the location where the Vertex Data exists.
Vertices	Collection of Int32 Objects which indicate the Vertex data as an index into the SubTree
Size	The size on disk of this Primitive Object.
PrimitiveAddress	The base File Offset of this Primitive Object used to reference other Pointers.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.
SaveVerts(Stream)	Returns a List of Int32 Objects denoting the Node Types contained in this Node and
UpdateRoot(long)	Updates the Root Pointer of this Primitive used as a reference for all other Pointers.

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.PrimitivePoint Class

A simple Primitive Object with a single vertex.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive

Implements

F4BMS.LOD.Polys.IPrimitive

Syntax

```
[ComVisibleAttribute(true)]
public class PrimitivePoint : Primitve, IPrimitive
```

Constructors

Name	Description
PrimitivePoint()	Initializes a new instance of the PrimitivePoint Class.
PrimitivePoint(Stream, long)	Initializes a new instance of the PrimitivePoint Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
RGBA	Color index of the RGBA color used by this Point Object.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.PrimitiveLine Class

A simple Primitive Object with a series of vertex points.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive

Implements

F4BMS.LOD.Polys.IPrimitive

Syntax

[ComVisibleAttribute(true)]
public class PrimitiveLine : Primitve, IPrimitive

Constructors

Name	Description
PrimitiveLine()	Initializes a new instance of the PrimitiveLine Class.
PrimitiveLine(Stream, long)	Initializes a new instance of the PrimitiveLine Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
RGBA	Color index of the RGBA color used by this Point Object.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.PrimitiveLightString Class

A simple Primitive Object with a series of lighted vertex points. Currently unused in the F4:BMS program.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive

Implements

F4BMS.LOD.Polys.IPrimitive

Syntax

```
[ComVisibleAttribute(true)]
public class PrimitiveLightString : Primitve, IPrimitive
```

Constructors

Name	Description
PrimitiveLightString()	Initializes a new instance of the PrimitiveLightString Class.
PrimitiveLightString(Stream, long)	Initializes a new instance of the PrimitiveLightString Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
FrontRGBA	Color index of the Front RGBA color used by this Point Object.
BackRGBA	Color index of the Back RGBA color used by this Point Object.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.Poly Class

Abstract base class for Polygon Objects to extend.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive

Syntax

[ComVisibleAttribute(true)]
public abstract class Poly : Primitve

Properties

Name	Description
A	I- Vector Component
B	J- Vector Component
C	K- Vector Component
D	Distance Vector Component.
PrimitiveAddress	Base address of this Polygon Object Object used as a reference for other Pointers.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.PolyFC Class

A Flat Polygon Object.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive
F4BMS.LOD.Polys.Poly

Implements

F4BMS.LOD.Polys.IPrimitive

Syntax

```
[ComVisibleAttribute(true)]
public class PolyFC : Poly, IPrimitive
```

Constructors

Name	Description
PolyFC()	Initializes a new instance of the PolyFC Class.
PolyFC(Stream, long)	Initializes a new instance of the PolyFC Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
RGBA	Color index of the RGBA color used by this Polygon.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.PolyVC Class

A Flat Gouraud Polygon Object.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive
F4BMS.LOD.Polys.Poly

Implements

F4BMS.LOD.Polys.IPrimitive

Syntax

```
[ComVisibleAttribute(true)]
public class PolyVC : Poly, IPrimitive
```

Constructors

Name	Description
PolyVC()	Initializes a new instance of the PolyVC Class.
PolyVC(Stream, long)	Initializes a new instance of the PolyVC Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
RGBAPointer	Pointer to the Vertex Color data.
VertexColors	Collection of the Color Indices used for the Verex Points.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.PolyFCN Class

A Flat Polygon Object with Normal Lighting data..

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive
F4BMS.LOD.Polys.Poly
F4BMS.LOD.Polys.PolyFC

Implements

F4BMS.LOD.Polys.IPrimitive

Syntax

```
[ComVisibleAttribute(true)]
public class PolyFCN : PolyFC, IPrimitive
```

Constructors

Name	Description
PolyFCN()	Initializes a new instance of the PolyFCN Class.
PolyFCN(Stream, long)	Initializes a new instance of the PolyFCN Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
Intensity	Normal Lighting Intensity value.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.PolyVCN Class

A Flat Gouraud Polygon Object with Normal Lighting data..

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive
F4BMS.LOD.Polys.Poly
F4BMS.LOD.Polys.PolyVC

Implements

F4BMS.LOD.Polys.IPrimitive

Syntax

```
[ComVisibleAttribute(true)]
public class PolyVCN : PolyVC, IPrimitive
```

Constructors

Name	Description
PolyVCN()	Initializes a new instance of the PolyVCN Class.
PolyVCN(Stream, long)	Initializes a new instance of the PolyVCN Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
IntensityPointer	Pointer to the Vertex Intensity Data.
VertexIntensities	Collection of the Normal Intensity Values for the vertex data.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.PolyTexFC Class

A Textured Flat Polygon Object.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive
F4BMS.LOD.Polys.Poly
F4BMS.LOD.Polys.PolyFC

Implements

F4BMS.LOD.Polys.IPrimitive

Syntax

```
[ComVisibleAttribute(true)]
public class PolyTexFC : PolyFC, IPrimitive
```

Constructors

Name	Description
PolyTexFC()	Initializes a new instance of the PolyTexFC Class.
PolyTexFC(Stream, long)	Initializes a new instance of the PolyTexFC Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
Texture	The index for the Texture to use with this Polygon.
UVAddress	The Pointer for the UV Data used in Texture Mapping.
UV	The colleciton of 2D Point Objects to represent the Texture Map.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.

Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.PolyTexVC Class

A Textured Flat Gouraud Polygon Object.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive
F4BMS.LOD.Polys.Poly
F4BMS.LOD.Polys.PolyVC

Implements

F4BMS.LOD.Polys.IPrimitive

Syntax

```
[ComVisibleAttribute(true)]
public class PolyTexVC : PolyVC, IPrimitive
```

Constructors

Name	Description
PolyTexVC()	Initializes a new instance of the PolyTexVC Class.
PolyTexVC(Stream, long)	Initializes a new instance of the PolyTexVC Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
Texture	The index for the Texture to use with this Polygon.
UVAddress	The Pointer for the UV Data used in Texture Mapping.
UV	The colleciton of 2D Point Objects to represent the Texture Map.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.
DebugPrint()	Returns a formatted string with advanced information about this Node.

Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.PolyTexFCN Class

A Textured Flat Polygon Object with Lighting Data.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive
F4BMS.LOD.Polys.Poly
F4BMS.LOD.Polys.PolyFC
F4BMS.LOD.Polys.PolyFCN

Implements

F4BMS.LOD.Polys.IPrimitive

Syntax

```
[ComVisibleAttribute(true)]
public class PolyTexFCN : PolyFCN, IPrimitive
```

Constructors

Name	Description
PolyTexFCN()	Initializes a new instance of the PolyTexFCN Class.
PolyTexFCN(Stream, long)	Initializes a new instance of the PolyTexFCN Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
Texture	The index for the Texture to use with this Polygon.
UVAddress	The Pointer for the UV Data used in Texture Mapping.
UV	The colleciton of 2D Point Objects to represent the Texture Map.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.

DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.LOD.Polys.PolyTexVCN Class

A Textured Flat Gouraud Polygon Object with Lighting Data.

Namespace: F4BMS.LOD.Polys
Assembly: F4BMS.dll

Inheritance

System.Object
F4BMS.LOD.Polys.Primitive
F4BMS.LOD.Polys.Poly
F4BMS.LOD.Polys.PolyVC
F4BMS.LOD.Polys.PolyVCN

Implements

F4BMS.LOD.Polys.IPrimitive

Syntax

```
[ComVisibleAttribute(true)]
public class PolyTexVCN : PolyVCN, IPrimitive
```

Constructors

Name	Description
PolyTexVCN()	Initializes a new instance of the PolyTexVCN Class.
PolyTexVCN(Stream, long)	Initializes a new instance of the PolyTexVCN Class using the supplied Stream object to read the data from a file or memory stream.

Properties

Name	Description
Texture	The index for the Texture to use with this Polygon.
UVAddress	The Pointer for the UV Data used in Texture Mapping.
UV	The colleciton of 2D Point Objects to represent the Texture Map.

Methods

Interface	Description
Print()	Returns a formatted string with basic information about this Node.

DebugPrint()	Returns a formatted string with advanced information about this Node.
Save(Stream)	Writes the Serializable data to the supplied Stream Object.

Remarks

Individual classes in the Polys Namespace are provided for functionality. It is highly recommended to only use the types in this Namespace through the LOD Class.

F4BMS.KEYFILE Namespace

Contains the types required to provide basic interaction with the BMS Key Files.

Classes

Namespaces	Description
KeyFile	Provides interaction with the BMS Key Files

F4BMS.KEYFILE.KeyFile Class

Represents the BMS Key File stucture

Namespace: F4BMS.KeyFile
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class KeyFile
```

Constructors

Name	Description
KeyFile(string)	Initializes a new instance of the KeyFile Class using the supplied file.
KeyFile(string, string)	Initializes a new instance of the KeyFile class using the supplied file and the supplied Falcon BMS.cfg file.

Properties

Name	Description
KeyBindings	Collection of bindings used for Keyboard Key mapping
ButtonBindings	Collection of bindings used for DX Button mapping
POVBindings	Collection of bindings used for POV and Hat Switches
FileName	The name of the current file
DXShiftOffset	The offset applied to DX Button IDs when shifting is used

Methods

Interface	Description
Print()	Returns a formatted string representation of the Key File.
Save()	Writes the Key File to the currently loaded file.
Save(Stream)	Writes the data to the supplied stream
Save(string)	Writes the data to the supplied file name

Classes

Class	Description
KeyBinding	Logic used to build or manipulate a Keyboard Key Binding.
ButtonBinding	Logic used to build or manipulate a DirectX Button Binding
POVBinding	Logic used to build or manipulate a POV or HAT Switch Binding

Enumerations

Enumeration	Description
KeyModifier	Enumerates the possible Key Modifiers
AccessDescriptor	Enumerates the Visibility and Access of a Command Binding in the UI
ButtonBehavior	Enumerates the ways a command is executed when pressing a button.

F4BMS.KEYFILE.KeyFile.KeyBinding Class

Represents the BMS Key Binding stucture used inside the file

Namespace: F4BMS.KeyFile
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class KeyBinding
```

Constructors

Name	Description
KeyBinding(string)	Initializes a new instance of the KeyBinding Class from formatted text.
KeyBinding(string, int, Key, KeyModifier, Key, KeyModifier, AccessDescriptor, string)	Initializes a new instance of the KeyBinding Class from the supplied initial values

Properties

Name	Description
Callback	BMS Code callback to execute with this binding
SoundID	Sound Index from f4sndtbl.txt to use when the command is executed
Key	The Key the command is bound to
KeyModifier	Modifiers used with the Key Binding
ComboKey	Key Combo Initiator
ComboModifier	Modifiers used with the Combo Key
Access	The Visibility and Access of the Command in the UI
Description	Description of the binding which appears in the UI. Limited to 45 characters

Methods

Interface	Description
Print()	Returns a formatted string representation of the Key Binding
Save(Stream)	Writes the data to the supplied stream

F4BMS.KEYFILE.KeyFile.ButtonBinding Class

Represents the BMS DirectX Button Binding stucture used inside the file

Namespace: F4BMS.KeyFile
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class ButtonBinding
```

Constructors

Name	Description
ButtonBinding(string)	Initializes a new instance of the ButtonBinding Class from formatted text.
ButtonBinding(string, int, ButtonBehavior, int, int, string)	Initializes a new instance of the ButtonBinding Class from the supplied initial values
ButtonBinding(string, int, ButtonBehavior, int, int)	Initializes a new instance of the ButtonBinding Class from the supplied initial values

Properties

Name	Description
Callback	BMS Code callback to execute with this binding
RawButtonID	The Button ID used by BMS. Linked to DirectXButtonID
DirectXButtonID	The DirectX Button ID. Linked to RawButtonID
Behavior	Determines if the button responds to Up, Down, or Both events
CommandMappingType	Read-Only Property indicating this is a Button Command and not a POV or Hat
ActivationType	Identifies if the Command is executed when Pressed or Released
SoundID	Sound Index from f4sndtbl.txt to use when the command is executed
Description	Optional: Description of the Binding. Not visible in UI

Methods

Interface	Description
Print()	Returns a formatted string representation of the Button Binding
Save(Stream)	Writes the data to the supplied stream

F4BMS.KEYFILE.KeyFile.POVBinding Class

Represents the BMS POV and Hat Binding stucture used inside the file

Namespace: F4BMS.KeyFile
Assembly: F4BMS.dll

Inheritance

System.Object

Syntax

```
[ComVisibleAttribute(true)]
public class POVBinding
```

Constructors

Name	Description
POVBinding(string)	Initializes a new instance of the POVBinding Class from formatted text.
POVBinding(string, int, int, int, string)	Initializes a new instance of the POVBinding Class from the supplied initial values
POVBinding(string, int, int, int)	Initializes a new instance of the POVBinding Class from the supplied initial values

Properties

Name	Description
Callback	BMS Code callback to execute with this binding
ID	The Button ID used by BMS. Linked to DirectXButtonID
Direction	The Direction the POV or Hat must move to execute the binding
CommandMappingType	Read-Only Property indicating this is a POV or Hat and not a Button
SoundID	Sound Index from f4sndtbl.txt to use when the command is executed
Description	Optional: Description of the Binding. Not visible in UI

Methods

Interface	Description
Print()	Returns a formatted string representation of the Button Binding
Save(Stream)	Writes the data to the supplied stream