pCon.planner Plugin SDK

Version 8.8.1

Tue May 2 2023

Contents

Chapter 1

Main Page

Introduction

You can extent the functionality of pCon.planner by using this plugin system.

The plugin system consists of a set of .NET classes which are used to create a .NET library (dll).

Creating a new plugin requires the following steps: First, create a new .NET 4.0 dll project in VisualStudio.

Reference System.Drawing (from .NET) and X3gPlugin.dll (shipped with the planner).

A class which derives from X3g.Plugin.Plugin must be created.

This class contains the actual plugin code.

There can only be one such class per dll.

For further documentation and how to write a plugin see the documentation of the Plugin class.

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 4

Class Index

4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 5

Namespace Documentation

5.1 X3g Namespace Reference

root namespace

Namespaces

- Plugin

  generic plugin namespace

5.1.1 Detailed Description

root namespace

5.2 X3g::Plugin Namespace Reference

generic plugin namespace

Namespaces

- Articles

  Namespace provides access to article handling.
- IO

  Namespace provides access to file import and export.
- Layout

  Namespace provides access to layout.
- Modeling

  Namespace provides access to modeling classes.
- Room

  Namespace provides access to different room elements.
- Tool

  Namespace provides access to different tools.

Classes

- class IActionManager

  Action Manager.
- interface IAppDialogOverride

  Allows to override application dialogs.
- class DefaultAppDialogOverride

  Implementation of IAppDialogOverride which lets application show all dialogs.
- class IAppManager

  Application Manager.
- class BackgroundParams

  Contains background parameter for Views.
- class Block

  Blocks are containers for entities within a planning.
- struct SaveBlockParams

  Block export parameters.
- class BlockManager

  Block Manager.
- class CannotExplodeEntityException

  CannotExplodeEntityException Thrown if a database entity could not be exploded.
- interface ICamera

  Common camera interface.
- class Camera

  Implementation of a common camera without visual representation in the planning.
- class CameraAnimationNode

  Camera Animation Node.
- struct CameraAnimationInfo

  Camera Animation Information.
- struct CameraAnimationFrame

  Camera Animation Frame.
- class CameraAnimation

  Camera Animation.
- class CameraEntity

  Camera entity.
- class CameraManager

  Camera Manager.
- class DbArrangement

  An arrangement automatic places items with the help of a base geometry.
- class DbCurve

  Base class for all curves.
- class DbCircle

  A simple closed circle.
- class DbEllipse

  An ellipse.
- class DbPolyline

  A line with two or more vertices.
- class DbDictionary

  A dictionary stores arbitrary data as key value pairs.
- class DbDimension

  Base class for all dimension types.
- class DbAlignedDimension

A length dimension with a dimension line parallel to measuring points.

- class DbArcDimension

  A length dimension to measure an arc.
- class DbDiametricDimension

  A length dimension used to measure a diameter.
- class DbRadialDimension

  A length dimension used to measure a radius.
- class DbRotatedDimension

  A length dimension with a dimension line rotated to measuring points.
- class DbThreePointAngularDimension

  An angular dimension based on center point and to mesuring points.
- class DbTwoLineAngularDimension

  An angular dimension between two lines.
- struct EntityColor

  Color of an entity.
- struct Hyperlink

  Interfaces an ACAD Hyperlink.
- class DbEntityCallbacks

  Abstract base class for entity callbacks.
- class DbEntity

  Base class for all database objects having a graphical representation.
- class DbBlockReference

  A block reference is used to place, size, and display an instance of the collection of entities within a DbBlock that it references.
- class DbAttributeDefinition

  Attribute definitions are used to define tags, or labels, for blocks.
- class DbAttribute

  Attributes are tags, or labels, for blocks.
- class DbLight

  A light source which is part of the document.
- class DbMesh

  A DbMesh is a collection of polygons used to define a 3D object.
- class DbSolid

  A DbSolid indicates a geometrical object defined by paramters.
- class DbPoint

  A point in 3D space.
- class DbPointCloud

  A point cloud.
- class DbImage

  A raster image.
- class DbText

  A text entity.
- class DimensionStyle

  A style for dimensions.
- class DimensionStyleManager

  This class manage dimension styles of current document.
- class GeometryUpdateEventArgs

  Event arguments for geometry update of top-level entities.
- class LayersChangedEventArgs

  Event arguments for layer(s) changed.
- class DocumentManager

Document Manager.

- class DocumentSummary

  Document Summary.
- class GeDrawable

  Root class of all drawables.
- class GeGeometry

  Geometry.
- class GePoints

  A set of points.
- class GePolylines

  A set of polylines of the same color.
- class GeMesh

  Mesh data for 3d geometry.
- class GeComponentInstance

  A component instance instanciates and transforms a component.
- class GeComponent

  Container for a set of drawables.
- struct ExportParams

  Params for modelspace export of GeometryManager.
- struct Export3dsParams

  3DS export parameters.
- struct ExportDaeParams

  Dae export parameters.
- struct ExportSkpParams

  Skp export parameters.
- struct ExportFbxParams

  Fbx export parameters.
- struct FbxFormatDescription

  Description of a supported FBX format.
- struct ExportObjParams

  Obj export parameters.
- struct ExportEgmParams

  Egm export parameters.
- struct ExportOffParams

  Off export parameters.
- class GeometryManager

  Geometry Manager.
- class Image

  Holds an image.
- class IImage

  Wraps an object of class Image.
- interface IMessageBox

  Interface for custom message boxes.
- class Layer

  Layers are used to group information in a drawing by function and to enforce linetype, color, and other standards.
- class LayerManager

  Layer Manager.
- class LightManager

  Light Manager.
- class LogEventArgs

  Arguments for log event.

- class Logger

  This class helps to use logging mechanism of pCon.planner.
- struct SoundAbsorption

  Sound absorption values.
- class MaterialUtility

  Provides helper functions for material handling.
- class Material

  Material describes how an object interacts with light.
- class MaterialChangedEventArgs

  Event arguments for change of a material.
- class MaterialManager

  Material Manager.
- class MessageBox

  Static class to group message box functionality.
- class OverlayEntity

  Base class for graphical objects without database representation.
- class OverlayEntity3d

  Base class for 3d overlay entities.
- class OverlayPointSet3d

  A graphical point set without database representation.
- class OverlayPolyline3d

  A graphical polyline without database representation.
- class OverlayMesh3d

  A graphical triangle mesh without database representation.
- class OverlayText3d

  A graphical text without database representation.
- class OverlayImage3d

  An image wich is rendered as 3d overlay.
- class OverlayEntity2d

  Base class for 2d overlay entities.
- class OverlayImage2d

  An image which is rendered as 2d overlay.
- interface IProperty

  Property.
- interface IPropertyDescriptor

  Describes a property.
- class PropertyDescriptor

  Default implementation of IPropertyDescriptor.
- class PropertiesChangedEventArgs

  Provides information about property changes.
- class PropertyProvider

  Property Provider.
- interface IPropertyCallbacks

  Allows to implement your own properties and add them to a provider.
- class PropertyValue

  Base class for various value properties.
- class BoolPropertyValue

  Boolean Property Value.
- class IntPropertyValue

  Integer Property Value.
- class StringPropertyValue

String Property Value.

- class DoublePropertyValue

  Double Property Value.
- class ColorPropertyValue

  Color Property Value.
- class LengthPropertyValue

  Length Property Value.
- class ChoicePropertyValue

  Choice Property Value.
- class TextPropertyValue

  Text (multi-line string) Property Value.
- class CurrencyPropertyValue

  Currency Property Value.
- class FontPropertyValue

  Font Property Value.
- class ImagePropertyValue

  Image Property Value.
- interface IRenderCallback

  Interface for the render callback to supervise render progress.
- class RenderProgressChangedEventArgs

  Event arguments for render progress changes.
- class Renderer

  The base class for all renderer.
- class IOpenGLRenderer

  Realtime OpenGL Renderer.
- class RendererManager

  Manages the Renderer available in the pCon.planner.
- class RenderLineStyle

  Describes a line style of a render style.
- class IRenderStyle

  Render style.
- class RenderStyleManager

  Manager for render styles.
- interface IToolTemplate

  Tool Template.
- interface ITool

  Tool.
- struct GeVec4f

  A four dimensional float vector or point.
- struct GeVec4d

  A four dimensional double vector or point.
- struct GeVec3f

  A three dimensional float vector or point.
- struct GeVec3d

  A three dimensional double vector or point.
- struct GeVec2f

  A two dimensional float vector or point.
- struct GeVec2d

  A two dimensional float vector or point.
- struct GeQuat

  Quaternion.

- class GeMatrix

  Matrix class.
- class GeBoundingBox3d

  Bounding box.
- class GePlane

  Plane.
- class Currency

  Currency class.
- interface ICommand

  Base interface for user defined commands
- interface IUndoManager

  Undo Manager.
- class IUtilities

  Utilities.
- class IVectorImage

  Vector Image.
- class IVectorRenderer

  Vector Image Renderer.
- class View

  A View.
- class DragDropEventArgs

  Provides data associated with drag'n'drop events and allows handlers to reply if they accept the drop.
- class ViewManager

  View manager.
- class Window

  Plugin host window.
- class Widget

  Allows to manage widgets of pCon.planner.
- class WindowManager

  Allows plugins to create and manage dockable windows.
- class SaveEventInfo

  Parameter for the save events.
- class InteractionEventInfo

  Parameter for interaction events.
- class EntityEventInfo

  Parameter interface for events about entities.
- class LicenseException

  License Exception.
- interface IModuleProvider

  Provides access to all major managers of the pCon.planner plugin interface.
- class Plugin

  The plugin interface.
- class IArticleEntityInfo

  Holds additional article information.
- class IArticleEntity

  Article interface for entities.
- class IArticleInstance

  Article Instance.
- class IOfmlArticleInstance

  Legacy class.
- class ICustomTextField

Legacy class.

- class IUserArticleInstance

  Legacy class.
- class IArticleManager

  Article Manager.
- class IBasket

  Basket interface.
- class IBasketArticlePriceComponent

  Basket Article Price Component.
- class IBasketArticleCalculation

  Basket Article Calculation interface.
- class IBasketArticleTextField

  A basket article text field.
- class IBasketArticleFeature

  Basket Article Feature
- class IBasketArticleItem

  Basket Article Item.
- class IBasketCalcItem

  BasketCalcItem.
- class IBasketCalculation

  Basket Calculation.
- class IBasketItem

  Basket Item.
- class IBasketSetArticleItem

  Basket Set Article Item.
- class IFMItem

  Folder Manager Item.
- class IFMFolderItem

  Folder Manager Folder-Item.
- class IFMArticleItem

  Folder Manager Article-Item.
- class IFMSetArticleItem

  Folder Manager SetArticle-Item.
- class IFolderManager

  Folder Manager.
- class IOfmlObject

  Ofml Object.
- class IProjectSettings

  Project Settings.

Typedefs

- typedef System::EventHandler < GeometryUpdateEventArgs^ > GeometryUpdateEventHandler

  Event handler for updated geometry.
- typedef System::EventHandler < LayersChangedEventArgs^ > LayersChangedEventHandler

  Event handler for changed layer.
- typedef System::EventHandler < LogEventArgs^ > LogEventHandler

  Event handler for LogEventArgs.

- typedef System::EventHandler < MaterialChangedEventArgs^ > MaterialChangedEventHandler

  Event handler for changed material.
- typedef System::EventHandler < PropertiesChangedEventArgs^ > PropertiesChangedEventHandler

  Event handler for changed properties.
- typedef System::EventHandler < RenderProgressChangedEventArgs^ > RenderProgressChangedEventHandler

  Event handler for render progress changes.
- typedef System::EventHandler < DragDropEventArgs^ > DragDropEventHandler

  Event handler for a drag n' drop event.

Enumerations

- enum ArticleConversionType { ArticleConversionType::ConvertToUserArticle, ArticleConversionType::ExplodeArticle, ArticleConversionType::ExplodeConfiguration, ArticleConversionType::ExplodePlaceholder, ArticleConversionType::SplitUpArticle, ArticleConversionType::RemoveArticleData }

  Determines a kind of article conversion.
- enum AppTerminateFlags : int { AppTerminateFlags::None = 0x0000, AppTerminateFlags::NoSave = 0x0001, AppTerminateFlags::NoCancel = 0x0002 }

  Flags for application termination.
- enum CameraProjection { CameraProjection::Perspective, CameraProjection::Orthographic, CameraProjection::Architectural, CameraProjection::Spherical }

  Types of camera projection.
- enum CameraAnimationNodeFlags : int { CameraAnimationNodeFlags::InterpolatePositionFirst = 0x0001, CameraAnimationNodeFlags::NormalizeDirection = 0x0002 }

  Camera Animation Node Flags.
- enum CameraInterpolation { CameraInterpolation::Linear, CameraInterpolation::SplineWeak, CameraInterpolation::SplineStrong }

  Determines the interpolation between camera animation nodes.
- enum ArrangementType { ArrangementType::AlongPath, ArrangementType::Area }

  Defines all possible arrangement types.
- enum ArrangementDistributionType { ArrangementDistributionType::ItemsCenter, ArrangementDistributionType::ItemsOuterEdge, ArrangementDistributionType::SymmetricInsideBase, ArrangementDistributionType::OnNodes, ArrangementDistributionType::Manual }

  Defines types how to distribute items of an arrangement.
- enum ArrangementVPlacement { ArrangementVPlacement::Above, ArrangementVPlacement::Center, ArrangementVPlacement::Below, ArrangementVPlacement::ItemBased }

  Defines vertical placement of an arrangement.
- enum EntityColorMethod { EntityColorMethod::ByColor, EntityColorMethod::ByBlock, EntityColorMethod::ByLayer }

  Determines how the color of an entity should be computed.
- enum DbTextureProjection { DbTextureProjection::None, DbTextureProjection::Planar, DbTextureProjection::Box, DbTextureProjection::Cylinder, DbTextureProjection::Sphere }

  Texture Projection.

- enum ArrowHeadType {
  ArrowHeadType::ArchTick, ArrowHeadType::BoxBlank, ArrowHeadType::BoxFilled, ArrowHeadType::ClosedTransparent,
  ArrowHeadType::ClosedBlank, ArrowHeadType::ClosedFilled, ArrowHeadType::DatumBlank, ArrowHeadType::DatumFilled,
  ArrowHeadType::DotBlank, ArrowHeadType::DotFilled, ArrowHeadType::DotSmallFilled, ArrowHeadType::DotSmallBlank,
  ArrowHeadType::Integral, ArrowHeadType::None, ArrowHeadType::Oblique, ArrowHeadType::Open15,
  ArrowHeadType::Open30, ArrowHeadType::Open90, ArrowHeadType::Origin, ArrowHeadType::Origin2 }

  Determines the type of arrows of an dimension line.
- enum DimensionTextHAlignment {
  DimensionTextHAlignment::Center, DimensionTextHAlignment::ByFirstExtension, DimensionTextHAlignment::BySecondExtension, DimensionTextHAlignment::AboveFirstExtension,
  DimensionTextHAlignment::AboveSecondExtension }

  Determines the horizontal alignment of the dimension text.
- enum DimensionTextVAlignment { DimensionTextVAlignment::Top, DimensionTextVAlignment::Online, DimensionTextVAlignment::Bottom }

  Determines the vertical alignment of the dimension text.
- enum DocumentPurgeFlags : int {
  DocumentPurgeFlags::General = 0x0001, DocumentPurgeFlags::Materials = 0x0002, DocumentPurgeFlags::Layers = 0x0004, DocumentPurgeFlags::DimensionStyles = 0x0008,
  DocumentPurgeFlags::TextStyles = 0x0010, DocumentPurgeFlags::LineTypes = 0x0020, DocumentPurgeFlags::SetArticles = 0x0040, DocumentPurgeFlags::All = 0xffff }

  Flags determining which parts of the document will get purged.
- enum DocumentFileFormat {
  DocumentFileFormat::Undefined, DocumentFileFormat::AutoCad2000, DocumentFileFormat::AutoCad2004, DocumentFileFormat::AutoCad2007,
  DocumentFileFormat::AutoCad2010, DocumentFileFormat::AutoCad2013, DocumentFileFormat::AutoCad2018 }

  The document file format version.
- enum DocumentFileType { DocumentFileType::Dwg, DocumentFileType::Dxf }

  The document file type.
- enum Export3DSCompatLevel { Export3DSCompatLevel::MeshInstances, Export3DSCompatLevel::Meshes }

  Compatibility setting for exporting to 3ds.
- enum SkpFormatVersion {
  SkpFormatVersion::Default , SkpFormatVersion::Skp6, SkpFormatVersion::Skp7, SkpFormatVersion::Skp8, SkpFormatVersion::Skp2013, SkpFormatVersion::Skp2014, SkpFormatVersion::Skp2015, SkpFormatVersion::Skp2016,
  SkpFormatVersion::Skp2017, SkpFormatVersion::Skp2018, SkpFormatVersion::Skp2019, SkpFormatVersion::Skp2020,
  SkpFormatVersion::Skp2021 }

  Version of skp format.
- enum ObjOrientation { , ObjOrientation::Y_Up }

  Orientation of the exported coordinates.
- enum GeometryCollectFlags : int {
  GeometryCollectFlags::None = 0x0000, GeometryCollectFlags::Collect2D = 0x0001, GeometryCollectFlags::Collect3D = 0x0002, GeometryCollectFlags::Collect2D3D = 0x0003,
  GeometryCollectFlags::CollectInvisible = 0x0004, GeometryCollectFlags::KeepDoubleSidedFaces = 0x0008 }

  Flags determining which kind of geometries will be collected.
- enum PixelFormat {
  PixelFormat::RGB24, PixelFormat::RGBA32, PixelFormat::GRAY8, PixelFormat::RGB96F,
  PixelFormat::RGBA128F }

  Image pixel formats.

- enum ImageFormat { ImageFormat::JPEG, ImageFormat::PNG, ImageFormat::EXR }

  Image file formats.
- enum LogLevel : int {
  LogLevel::Fatal = 4, LogLevel::Error = 3, LogLevel::Warning = 2, LogLevel::Info = 1,
  LogLevel::Debug = 0 }

  Defines severity of log message.
- enum MaterialType { , MaterialType::Common, MaterialType::Glass , MaterialType::Illuminant }

  Determines the material type.
- enum TextureType { TextureType::BaseColor, TextureType::Metallic, TextureType::Normal, TextureType::-Roughness }

  Determines texture type.
- enum OverlayLevel3d : int { OverlayLevel3d::InScene = 0, OverlayLevel3d::Middle = 4 }

  Overlay has several levels.
- enum OverlayLevel2d : int { OverlayLevel2d::Background = 2, OverlayLevel2d::Foreground = 8 }

  Overlay has several levels.
- enum PropertyType {
  PropertyType::Unknown, PropertyType::Bool, PropertyType::Int, PropertyType::String,
  PropertyType::Double, PropertyType::Color, PropertyType::Length, PropertyType::Choice,
  PropertyType::Text, PropertyType::Currency, PropertyType::Font, PropertyType::Group,
  PropertyType::Image }

  The type of a property
- enum PropertyState : int {
  PropertyState::None = 0x0000, PropertyState::Null = 0x0001, PropertyState::Enabled = 0x0002, Property-State::Hidden = 0x0004 ,
  PropertyState::Editable = 0x0010, PropertyState::ReadOnly = 0x0040 }

  The state of a property.
- enum ChoiceValueType {
  ChoiceValueType::Unknown, ChoiceValueType::String, ChoiceValueType::Int, ChoiceValueType::Double,
  ChoiceValueType::Length }

  The type of a choice value.
- enum RenderStyleBaseMode

  Basic render modes.
- enum MeasureType { MeasureType::Cartesian, MeasureType::Cylindric, MeasureType::Spherical }

  Types of position measure.
- enum Coordinates : int {
  Coordinates::X = 1 << 0, Coordinates::Y = 1 << 1, Coordinates::Z = 1 << 2, Coordinates::Rho = 1 << 3,
  Coordinates::Radius = 1 << 4, Coordinates::Phi = 1 << 5, Coordinates::Theta = 1 << 6, Coordinates::-Cartesian = X | Y | Z,
  Coordinates::Cylindric = Rho | Phi | Z, Coordinates::Spherical = Radius | Phi | Theta, Coordinates::Linear = X | Y | Z | Rho | Radius, Coordinates::Angle = Phi | Theta,
  Coordinates::All = X | Y | Z | Rho | Radius | Phi | Theta }

  Types of Coordinates.
- enum LengthUnit {
  LengthUnit::Undefined, LengthUnit::Millimeter, LengthUnit::Centimeter, LengthUnit::Inch,
  LengthUnit::Feet, LengthUnit::Meter, LengthUnit::Kilometer }

  The length unit of the document.
- enum AcisQuality {
  AcisQuality::Lowest, AcisQuality::Low, AcisQuality::Medium, AcisQuality::High,
  AcisQuality::Highest }

  Quality setting for acis vectorization.
- enum HorizontalTextAlignment

  Horizontal alignment of Text.
- enum VerticalTextAlignment

Vertical alignment of Text.

- enum LightType { LightType::Spot, LightType::Point, LightType::Directional, LightType::Area, LightType::Photometric }

  Determines the type of a light source.

- enum RenderingCategory : int { RenderingCategory::Realtime = 0x0001, RenderingCategory::Offline = 0x0002, RenderingCategory::All = 0x0003 }

  Divides renderers into performance categories.

- enum PhotometricDataFormat { PhotometricDataFormat::EULUMDAT, PhotometricDataFormat::IES }

  Format of photometric data.

- enum VectorImageFormat { VectorImageFormat::Emf, VectorImageFormat::Svg }

  Supported vector image file formats.

- enum ViewLayout { ViewLayout::Single, ViewLayout::OneAndTwo , ViewLayout::FourTiles, ViewLayout::OneAndThree, ViewLayout::HorizontalSplit, ViewLayout::VerticalSplit }

  Determines the layout of the views.

- enum ViewCameraMode { ViewCameraMode::Top, ViewCameraMode::Bottom, ViewCameraMode::Left, ViewCameraMode::Right, ViewCameraMode::Front, ViewCameraMode::Back, ViewCameraMode::IsoNE, ViewCameraMode::IsoNW, ViewCameraMode::IsoSE, ViewCameraMode::IsoSW, ViewCameraMode::Perspective, ViewCameraMode::Orthographic }

  Determines the camera mode of a view.

- enum ViewSpecificLayerVisibility { ViewSpecificLayerVisibility::Global, ViewSpecificLayerVisibility::Visible, ViewSpecificLayerVisibility::Invisible }

  Visability mode of a layer in view.

- enum EventType { EventType::DocumentWillClear, EventType::DocumentOpened, EventType::DocumentNew, EventType::DocumentWillSave, EventType::DocumentSaved, EventType::DocumentSelectionChanged, EventType::DocumentNameChanged, EventType::ApplicationWillTerminate, EventType::ApplicationReady, EventType::InteractionFinished, EventType::DocumentEntitiesAdded }

  The type of a special event.

- enum SaveMode { SaveMode::Save, SaveMode::SaveCopy, SaveMode::SavePart }

  The mode used for saving the document.

- enum ArticleEntityType { ArticleEntityType::UnknownArticle, ArticleEntityType::SingleArticle, ArticleEntityType::Configuration, ArticleEntityType::UserArticle, ArticleEntityType::Placeholder }

  The type of the article entity.

- enum ArticleGeometryMode { ArticleGeometryMode::Undefined, ArticleGeometryMode::AGM_2D, ArticleGeometryMode::AGM_3D, ArticleGeometryMode::AGM_2D3D }

  Geometry mode of an article entity describes if the geometric representation of an article is intended for 2d or 3d planning.

- enum BasketGenerationFlags : int { BasketGenerationFlags::SelectionOnly = 0x0001, BasketGenerationFlags::Summarize = 0x0002, BasketGenerationFlags::DirectMode = 0x0004, BasketGenerationFlags::SplitComposites = 0x0008, BasketGenerationFlags::PrepareCatalogImages = 0x0010, BasketGenerationFlags::PrepareGeneratedImages = 0x0020, BasketGenerationFlags::IgnoreEmptyFolders = 0x0040 }

  Flags determining how the basket is generated.

- enum BasketArtNumType { BasketArtNumType::Default, BasketArtNumType::Base, BasketArtNumType::Final, BasketArtNumType::VariantCode, BasketArtNumType::OFMLVariantCode }

  The type of the article number.

- enum BasketArtDescrType { BasketArtDescrType::Default, BasketArtDescrType::Short, BasketArtDescrType::Long, BasketArtDescrType::Feature }

  The type of the article description.
- enum BasketTextFieldType { BasketTextFieldType::Undefined, BasketTextFieldType::Short, BasketTextFieldType::Long, BasketTextFieldType::Variant, BasketTextFieldType::Application }

  The type of the article text field.
- enum BasketItemType { BasketItemType::UnknownItem, BasketItemType::Folder, BasketItemType::Article, BasketItemType::SetArticle, BasketItemType::UserArticle }

  The type of the basket item

Functions

- public delegate void ValueEvent (System::String^ pKey)

  Delegate to handle Events regarding the change of values.

Variables

- template<typename T >
  where T

  Returns all entities of given type.

5.2.1 Detailed Description

generic plugin namespace

5.2.2 Typedef Documentation

5.2.2.1 typedef System::EventHandler<DragDropEventArgs^> X3g::Plugin::DragDropEventHandler

Event handler for a drag n' drop event.

The event argument has to be of type DragDropEventArgs.

5.2.2.2 typedef System::EventHandler<GeometryUpdateEventArgs^> X3g::Plugin::GeometryUpdateEventHandler

Event handler for updated geometry.

The event argument has to be of type GeometryUpdateEventArgs.

5.2.2.3 typedef System::EventHandler<LayersChangedEventArgs^> X3g::Plugin::LayersChangedEventHandler

Event handler for changed layer.

The event argument has to be of type LayersChangedEventArgs.

5.2.2.4 typedef System::EventHandler<LogEventArgs^> X3g::Plugin::LogEventHandler

Event handler for LogEventArgs.

5.2.2.5 typedef System::EventHandler<MaterialChangedEventArgs^> X3g::Plugin::MaterialChangedEventHandler

Event handler for changed material.

The event argument has to be of type MaterialChangedEventArgs.

5.2.2.6 typedef System::EventHandler<PropertiesChangedEventArgs^> X3g::Plugin::PropertiesChangedEventHandler

Event handler for changed properties.

The event argument has to be of type PropertiesChangedEventArgs.

5.2.2.7 typedef System::EventHandler<RenderProgressChangedEventArgs^> X3g::Plugin::RenderProgressChangedEventHandler

Event handler for render progress changes.

The event argument has to be of type RenderProgressChangedEventArgs.

5.2.3 Enumeration Type Documentation

5.2.3.1 enum X3g::Plugin::AcisQuality [strong]

Quality setting for acis vectorization.

Enumerator

Lowest lowest quality

Low low quality

Medium medium quality

High high quality

Highest highest quality

5.2.3.2 enum X3g::Plugin::AppTerminateFlags : int [strong]

Flags for application termination.

Enumerator

None No flags set.

NoSave The application won't ask to save before termination even if the document has been changed.

NoCancel It won't be possible to cancel the termination.

5.2.3.3 enum X3g::Plugin::ArrangementDistributionType [strong]

Defines types how to distribute items of an arrangement.

Enumerator

ItemsCenter Items will be distributed with offset set to 0 and max possible gap to place all items.

ItemsOuterEdge Items will be distributed that there complete fit inside the base.

SymmetricInsideBase Items will be distributed symmetric inside base geometry. The offset will be half of the gap.

OnNodes Items will be distributed on spezific points of the base geometry (e.g. break-points). Only available on using ArrangementType::AlongPath.

Manual Items will be distributed by fixed offset and gap.

5.2.3.4 enum X3g::Plugin::ArrangementType [strong]

Defines all possible arrangement types.

Enumerator

AlongPath The items will be arranged along the path.

Area The items will be arranged on spanned area.

5.2.3.5 enum X3g::Plugin::ArrangementVPlacement [strong]

Defines vertical placement of an arrangement.

Enumerator

Above Place items above base geometry.

Center Place items through base geometry.

Below Place items below base geometry.

ItemBased Place items based on local origin of item.

5.2.3.6 enum X3g::Plugin::ArrowHeadType [strong]

Determines the type of arrows of an dimension line.

Enumerator

ArchTick A architectural tick.

BoxBlank A blank box.

BoxFilled A filled box.

ClosedTransparent A transparent closed triangle.

ClosedBlank A blank closed triangle

ClosedFilled A filled closed triangle.

DatumBlank A blank datum triangle.

DatumFilled A filled datum triangle.

DotBlank A blank dot.

DotFilled A filled dot.

DotSmallFilled A small filled dot.

DotSmallBlank A small blank dot.

Integral An integrall.

None No arrow.

Oblique A thin oblique.

Open15 A open triangle with 15 degree opening angle.

Open30 A open triangle with 30 degree opening angle.

Open90 A open triangle with 90 degree opening angle

Origin A blank circle.

Origin2 A blank circle with an hole.

5.2.3.7 enum X3g::Plugin::ArticleConversionType [strong]

Determines a kind of article conversion.

Enumerator

ConvertToUserArticle Conversion to user article.

ExplodeArticle Ungroup article.

ExplodeConfiguration Ungroup configuration.

ExplodePlaceholder Ungroup placeholder.

SplitUpArticle Split-up article.

RemoveArticleData Remove article data.

5.2.3.8 enum X3g::Plugin::ArticleEntityType [strong]

The type of the article entity.

Enumerator

UnknownArticle An unknown article.

SingleArticle A single article with optional childs.

Configuration A group/compound of articles.

UserArticle A user/custom article.

Placeholder The entity is only a placeholder which will be replaced with the final article during the basket generation.

5.2.3.9 enum X3g::Plugin::ArticleGeometryMode [strong]

Geometry mode of an article entity describes if the geometric representation of an article is intended for 2d or 3d planning.

Enumerator

Undefined Geometry mode is undefined (e.g. user articles).

AGM_2D Article entity has only two-dimensional geometry.

AGM_3D Article entity has only three-dimensional geometry.

AGM_2D3D Article entity has two- and three-dimensional geometry.

5.2.3.10 enum X3g::Plugin::BasketArtDescrType [strong]

The type of the article description.

Enumerator

Default The short and/or long description, according to the product data's defaults.

Short The short description.

Long The long description.

Feature The feature description.

5.2.3.11 enum X3g::Plugin::BasketArtNumType [strong]

The type of the article number.

Enumerator

Default The base or final article number according to the product data defaults.

Base The base article number.

Final The final article number.

VariantCode The variant code.

OFMLVariantCode The OFML variant code.

5.2.3.12 enum X3g::Plugin::BasketGenerationFlags : int [strong]

Flags determining how the basket is generated.

Enumerator

SelectionOnly If set, only currently selected articles are processed, otherwise all articles in the drawing are used.

Summarize If set, articles with the same article number and configuration are summarized.

DirectMode If set, the basket is generated directly from article data in the document. No placeholder conversion or other rules will be triggered. The summarize flag is ignored in this case.

SplitComposites If set, composite articles will be split up.

PrepareCatalogImages If set, article catalog images will be prepared. See IBasketArticleItem::CatalogImage

PrepareGeneratedImages If set, generated article images will be prepared. See IBasketArticleItem::GeneratedImage

IgnoreEmptyFolders If set, empty folders (and set articles) will be ignored.

5.2.3.13 enum X3g::Plugin::BasketItemType [strong]

The type of the basket item

Enumerator

UnknownItem An unknown item.

Folder A folder item.

Article An article item.

SetArticle A set article item.

UserArticle An user article item.

5.2.3.14 enum X3g::Plugin::BasketTextFieldType [strong]

The type of the article text field.

Enumerator

Undefined Undefined text.

Short Short text.

Long Long text.

Variant Variant/Feature text.

Application Application defined text

5.2.3.15 enum X3g::Plugin::CameraAnimationNodeFlags : int [strong]

Camera Animation Node Flags.

Enumerator

InterpolatePositionFirst If set, the direction/rotation will change after the position of the next node is reached else they will change together.

NormalizeDirection The direction (eye-center distance) will be normalized to create a more uniform rotation.

5.2.3.16 enum X3g::Plugin::CameraInterpolation [strong]

Determines the interpolation between camera animation nodes.

Enumerator

Linear Lines interpolation.

SplineWeak Weak spline interpolation.

SplineStrong Strong spline interpolation.

5.2.3.17 enum X3g::Plugin::CameraProjection [strong]

Types of camera projection.

Enumerator

Perspective perspective camera projection

Orthographic orthographic camera projection

Architectural architectural camera projection

Spherical spherical camera projection

5.2.3.18 enum X3g::Plugin::ChoiceValueType [strong]

The type of a choice value.

Enumerator

Unknown An unknown type.

String A choice of type String.

Int A choice of type int.

Double A choice of type double.

Length A choice which represents a length.

5.2.3.19 enum X3g::Plugin::Coordinates : int [strong]

Types of Coordinates.

This enum is constructed as flags and can be combined.

Enumerator

X X-Direction.

Y Y-Direction.

Z Z-Direction.

Rho Distance on XY-Plane.

Radius Radius (distance to origin).

Phi Rotation around Z-Axis.

Theta Polar angle.

Cartesian Combination of coordinates default used by cartesian measure.

Cylindric Combination of coordinates default used by cylindric measure.

Spherical Combination of coordinates default used by spherical measure.

Linear Combination of all linear coordinates.

Angle Combination of all angle coordinates.

All Combination of all coordinates.

5.2.3.20 enum X3g::Plugin::DbTextureProjection [strong]

Texture Projection.

Enumerator

None No texture mapping.

Planar Maps directly to XY coordinates.

Box Maps to planes perpendicular to major axes.

Cylinder Maps to cylinder aligned with Z-axis.

Sphere Maps to sphere aligned with Z-axis

5.2.3.21 enum X3g::Plugin::DimensionTextHAlignment [strong]

Determines the horizontal alignment of the dimension text.

Enumerator

Center The text appears centered to the dimension line.

ByFirstExtension The text appears beside the first extension line.

BySecondExtension The text appears beside the second extension line.

AboveFirstExtension The text appears above the first extension line.

AboveSecondExtension The text appears above the second extension line.

5.2.3.22 enum X3g::Plugin::DimensionTextVAlignment [strong]

Determines the vertical alignment of the dimension text.

Enumerator

Top The text appears above the dimension line.

Online The text appears about the dimension line.

Bottom The text appears below the dimension line.

5.2.3.23 enum X3g::Plugin::DocumentFileFormat [strong]

The document file format version.

Enumerator

Undefined undefined file format

AutoCad2000 R15 file format.

AutoCad2004 R18 file format.

AutoCad2007 R21 file format.

AutoCad2010 R24 file format.

AutoCad2013 R27 file format.

AutoCad2018 R32 file format.

5.2.3.24 enum X3g::Plugin::DocumentFileType [strong]

The document file type.

Enumerator

Dwg dwg file type

Dxf dxf file type

5.2.3.25 enum X3g::Plugin::DocumentPurgeFlags : int [strong]

Flags determining which parts of the document will get purged.

Enumerator

General general purging (e.g. unuesd blocks)

Materials purge unused materials

Layers purge unused layery

DimensionStyles purge unused dimension styles

TextStyles purge unused text styles

LineTypes purge unused line types*/

SetArticles purge set articles with no children*/

All purge all unused objects*/

5.2.3.26 enum X3g::Plugin::EntityColorMethod [strong]

Determines how the color of an entity should be computed.

Enumerator

ByColor The color is set directly at the entity.

ByBlock The color is derived from the color of the block reference, referencing a block in which the entity resides.

ByLayer The color is derived from the layer of the entity.

5.2.3.27 enum X3g::Plugin::EventType [strong]

The type of a special event.

Enumerator

DocumentWillClear Will be send before the document will be cleared.

DocumentOpened Will be send after a document was opened.

DocumentNew Will be send after a new/empty document was created.

DocumentWillSave Will be send before the document will be saved. The parameter of the event will be a SaveEventInfo instance.

DocumentSaved Will be send after the document was be saved. The parameter of the event will be a SaveEventInfo instance.

DocumentSelectionChanged Will be send if the selection has changed.

DocumentNameChanged Will be send after the name of the document was changed.

ApplicationWillTerminate Will be send once as soon as the gui is about to close/terminate.

ApplicationReady Will be send once after the application was completely intialized and is ready for input.

InteractionFinished Sent after a tool action has been finished.

DocumentEntitiesAdded Sent after entities have been added to the document modelspace. Parameter is of type EntityEventInfo.

5.2.3.28 enum X3g::Plugin::Export3DSCompatLevel [strong]

Compatibility setting for exporting to 3ds.

Enumerator

MeshInstances smaller file size

Meshes higher compatibility with certain applications

5.2.3.29 enum X3g::Plugin::GeometryCollectFlags : int [strong]

Flags determining which kind of geometries will be collected.

Enumerator

None All flags disabled.

Collect2D Collect 2d geometries.

Collect3D Collect 3d geometries.

Collect2D3D Collect 2d and 3d geometries.

CollectInvisible Collect invisible geometries.

KeepDoubleSidedFaces Collected geometries may include double-sided faces. By default all double-sided faces will be split into two single-sided faces, what allows to render all faces with backface culling enabled but with increased vertex, normal and texcoord count.

See Also

GeMesh::DoubleSided

5.2.3.30 enum X3g::Plugin::HorizontalTextAlignment [strong]

Horizontal alignment of Text.

5.2.3.31 enum X3g::Plugin::ImageFormat [strong]

Image file formats.

Enumerator

JPEG Lossy compression format.

PNG Lossless compression format.

EXR High Dynamic Range (HDR) image format.

5.2.3.32 enum X3g::Plugin::LengthUnit [strong]

The length unit of the document.

Enumerator

Undefined undefined length unit

Millimeter millimeter length unit

Centimeter centimeter length unit

Inch inch length unit

Feet feet length unit

Meter meter length unit

Kilometer kilometer length unit

5.2.3.33 enum X3g::Plugin::LightType [strong]

Determines the type of a light source.

Enumerator

Spot A light source which has an opening and hotspot angle.

Point A light source which equally emits light in all directions.

Directional A light source which emits light in one direction without an attenuation (e.g. sun).

Area A light source which is defined by a rectangular area.

Photometric A light source which is defined by photometric data.

5.2.3.34 enum X3g::Plugin::LogLevel : int [strong]

Defines severity of log message.

Enumerator

Fatal Fatal.

Error Error.

Warning Warning.

Info Info.

Debug Debug.

5.2.3.35 enum X3g::Plugin::MaterialType [strong]

Determines the material type.

Enumerator

Common common/simple material

Glass glass material

Illuminant self-illuminating material

5.2.3.36 enum X3g::Plugin::MeasureType [strong]

Types of position measure.

Enumerator

Cartesian Measure position with distance in X, Y and Z direction.

Cylindric Use a cylindric coordinate system with a distance from origin on XY-Plane, a rotation around Z-Axis and a height in Z direction.

Spherical Use a spherical coordinate system with two angles to describe the direction and a radius for the distance.

5.2.3.37 enum X3g::Plugin::ObjOrientation [strong]

Orientation of the exported coordinates.

Enumerator

Y_Up CAD/DWG, default. OpenGL/OFML

5.2.3.38 enum X3g::Plugin::OverlayLevel2d : int [strong]

Overlay has several levels.

Higher levels are rendered over lower levels. This means that entities with higher level may occlude lower level entities.

Enumerator

Background Rendered over scene before 3d overlay.

Foreground Rendered over 3d overlay.

5.2.3.39 enum X3g::Plugin::OverlayLevel3d : int [strong]

Overlay has several levels.

Higher levels are rendered over lower levels. This means that entities with higher level may occlude lower level entities.

Enumerator

InScene Rendered like part of the scene.

Middle Rendered over scene after 2d background.

5.2.3.40 enum X3g::Plugin::PhotometricDataFormat [strong]

Format of photometric data.

Enumerator

EULUMDAT EULUMDAT format.

IES IES format.

5.2.3.41 enum X3g::Plugin::PixelFormat [strong]

Image pixel formats.

Enumerator

RGB24 integer RGB values with 24 bit per pixel

RGBA32 integer RGBA values with 32 bit per pixel

GRAY8 integer gray values with 8 bit per pixel

RGB96F floating point RGB values with 96 bit per pixel

RGBA128F floating point RGBA values with 128 bit per pixel

5.2.3.42 enum X3g::Plugin::PropertyState : int [strong]

The state of a property.

Enumerator

None No flags set.

Null Property value is null or unassigned.

Enabled If not set, the property value is greyed out and read-only.

Hidden Property is hidden.

Editable Property supports user defined values. If not set, only the values which are part of the valueset are valid.

ReadOnly Property value is read-only.

5.2.3.43 enum X3g::Plugin::PropertyType [strong]

The type of a property

Enumerator

Unknown An unknown property.

Bool A property of type bool.

Int A property of type int.

String A property of type String.

Double A property of type double.

Color A property of type Color.

Length A property which represents a length.

Choice A choice property.

Text A multi-line text property.

Currency A property of type Currency.

Font A font property.

Group A property which groups other properties.

Image A property which provides an id and a bitmap stream.

5.2.3.44 enum X3g::Plugin::RenderingCategory : int [strong]

Divides renderers into performance categories.

Enumerator

Realtime realtime rendering
Offline offline rendering
All combines all rendering categories

5.2.3.45 enum X3g::Plugin::RenderStyleBaseMode [strong]

Basic render modes.

5.2.3.46 enum X3g::Plugin::SaveMode [strong]

The mode used for saving the document.

Enumerator

Save The document was saved.
SaveCopy A copy of the document was saved.
SavePart Only a part of the document was saved (e.g. the current selection)

5.2.3.47 enum X3g::Plugin::SkpFormatVersion [strong]

Version of skp format.

Enumerator

Default Default Version.
Skp6 skp version from 2006
Skp7 skp version from 2008
Skp8 skp version from 2010
Skp2013 skp version from 2013
Skp2014 skp version from 2014
Skp2015 skp version from 2015
Skp2016 skp version from 2016
Skp2017 skp version from 2017
Skp2018 skp version from 2018
Skp2019 skp version from 2019
Skp2020 skp version from 2020
Skp2021 skp version from 2021

5.2.3.48 enum X3g::Plugin::TextureType [strong]

Determines texture type.

Enumerator

BaseColor base color texture
Metallic metallic texture
Normal normal texture
Roughness roughness texture

5.2.3.49 enum X3g::Plugin::VectorImageFormat [strong]

Supported vector image file formats.

Enumerator

Emf emf format
Svg svg format

5.2.3.50 enum X3g::Plugin::VerticalTextAlignment [strong]

Vertical alignment of Text.

5.2.3.51 enum X3g::Plugin::ViewCameraMode [strong]

Determines the camera mode of a view.

Enumerator

Top top orthographic camera
Bottom bottom orthographic camera
Left left orthographic camera
Right right orthographic camera
Front front orthographic camera
Back back orthographic camera
IsoNE isometric north-east orthographic camera
IsoNW isometric north-west orthographic camera
IsoSE isometric south-west orthographic camera
IsoSW isometric south-west orthographic camera
Perspective free perspective camera
Orthographic free orthographic camera

5.2.3.52 enum X3g::Plugin::ViewLayout [strong]

Determines the layout of the views.

Enumerator

Single Layout with one main view.
OneAndTwo Layout with one main view and two small views.
FourTiles Layout with four equally sized views.
OneAndThree Layout with one main view and three small views.
HorizontalSplit Layout with two horizontally splitted views.
VerticalSplit Layout with two vertically splitted views.

5.2.3.53 enum X3g::Plugin::ViewSpecificLayerVisibility [strong]

Visability mode of a layer in view.

Enumerator

Global layer is shown in this view, depending on its global visibility
Visible layer is shown in this view, regardless of its global visibility
Invisible layer is not shown in this view, regardless of its global visibility

5.2.4 Function Documentation

5.2.4.1 public delegate void X3g::Plugin::ValueEvent (System::String^ *pKey*)

Delegate to handle Events regarding the change of values.

5.2.5 Variable Documentation

5.2.5.1 template<typename T > where X3g::Plugin::T

Returns all entities of given type.

5.3 X3g::Plugin::Articles Namespace Reference

Namespace provides access to article handling.

Classes

- class ArticleEntityInfo

  Provides additional article information.
- class ArticleEntity

  Article interface for entities.
- class ArticleInstance

  Article instance.
- class OfmlArticleInstance

  OFML article instance.
- class CustomTextField

  Custom text field of user article.
- class UserArticleInstance

  User article instance.
- class ArticleInsertedEventArgs

  Provides information about the inserted article.
- class CatalogItem

  OFML catalog item.
- class Catalog

  OFML product catalog.
- class OfmlObject

  Ofml Object.
- class OfmlEval

  A class covering the evaluation of OFML expressions.
- class Package

  Provides information about registered OFML packages.
- class ProjectSettings

  ProjectSettings.

Typedefs

- typedef System::EventHandler
  < ArticleInsertedEventArgs^ > ArticleInsertedEventHandler

  Event handler for the event that an article has been inserted.

Enumerations

- enum ArticleDescriptionType { ArticleDescriptionType::Default, ArticleDescriptionType::Short, ArticleDescriptionType::Long, ArticleDescriptionType::Features }

  The type of the article description.
- enum ArticleNumberType { ArticleNumberType::Default, ArticleNumberType::Base, ArticleNumberType::Final, ArticleNumberType::VariantCode, ArticleNumberType::OfmlVariantCode }

  The type of the article number.
- enum CatalogItemType { CatalogItemType::Unknown, CatalogItemType::Article, CatalogItemType::Folder, CatalogItemType::Information, CatalogItemType::MethodCall, CatalogItemType::FunctionCall, CatalogItemType::MetaPlanning, CatalogItemType::Graphics }

  Type of catalog item.
- enum CatalogTextPurpose { CatalogTextPurpose::Label, CatalogTextPurpose::Description }

  Predefined purposes for catalog texts.
- enum CatalogImagePurpose { CatalogImagePurpose::Icon, CatalogImagePurpose::Image }

  Predefined purposes for catalog images.
- enum CatalogResourceType { CatalogResourceType::Graphics, CatalogResourceType::Html, CatalogResourceType::Mime, CatalogResourceType::Url }

  Type of catalog resource.
- enum PackageType { PackageType::Unknown, PackageType::Found, PackageType::Product, PackageType::ProductWithCatalog, PackageType::Catalog }

  Type of package.

Functions

- public delegate bool AssignDefaultPropValuesHandler (OfmlObject^ pObject)

  Assign default property values to a OfmlObject handler.

5.3.1 Detailed Description

Namespace provides access to article handling.

5.3.2 Typedef Documentation

5.3.2.1 typedef System::EventHandler<ArticleInsertedEventArgs^> X3g::Plugin::Articles::ArticleInsertedEventHandler

Event handler for the event that an article has been inserted.

The event argument has to be of type ArticleInsertedEventArgs.

5.3.3 Enumeration Type Documentation

5.3.3.1 enum X3g::Plugin::Articles::ArticleDescriptionType [strong]

The type of the article description.

Enumerator

Default The short and/or long description, according to the product data's defaults.
Short The short description.
Long The long description.
Features The features description.

5.3.3.2 enum X3g::Plugin::Articles::ArticleNumberType [strong]

The type of the article number.

Enumerator

Default The base or final article number according to the product data defaults.
Base The base article number.
Final The final article number.
VariantCode The variant code.
OfmlVariantCode The OFML variant code.

5.3.3.3 enum X3g::Plugin::Articles::CatalogImagePurpose [strong]

Predefined purposes for catalog images.

Enumerator

Icon image for presentation in catalog view
Image image for high resolution presentation of catalog item

5.3.3.4 enum X3g::Plugin::Articles::CatalogItemType [strong]

Type of catalog item.

Enumerator

Unknown unknown item type
Article insertable article
Folder catalog folder
Information additional information
MethodCall method call on selected planning element
FunctionCall static class method call
MetaPlanning metaplanning workflow
Graphics insertable graphics file (e.g. dwg, pec, 3ds)

5.3.3.5 enum X3g::Plugin::Articles::CatalogResourceType [strong]

Type of catalog resource.

Enumerator

Graphics insertable graphics file (e.g. dwg, pec, 3ds)
Html file in HTML format
Mime MIME type (e.g. "application/pdf;info.pdf")
Url fully qualified URL to internet resource

5.3.3.6 enum X3g::Plugin::Articles::CatalogTextPurpose [strong]

Predefined purposes for catalog texts.

Enumerator

Label text for presentation in catalog view

Description longer descriptive text, e.g. for tooltips

5.3.3.7 enum X3g::Plugin::Articles::PackageType [strong]

Type of package.

Enumerator

Unknown unknown package type

Found basic OFML library without product data and catalog data

Product OFML library with product data.

ProductWithCatalog OFML library with product data and catalog data.

Catalog catalog data without product data

5.3.4 Function Documentation

5.3.4.1 public delegate bool X3g::Plugin::Articles::AssignDefaultPropValuesHandler (OfmlObject^ *pObject*)

Assign default property values to a OfmlObject handler.

Returns

The function should return true if properties have been modified.

5.4 X3g::Plugin::IO Namespace Reference

Namespace provides access to file import and export.

Classes

- class PecExport

  Allows to export entities to pCon Exchange Container (PEC).
- class GltfExport

  Allows to export entities in glTF format.
- class UsdzExport

  Allows to export entities in Usdz format.
- class RgfxExport

  Allows to export entities in revit exchange format (rgfx).

Enumerations

- enum GltfFormatVersion

  Available glTF format versions.

5.4.1 Detailed Description

Namespace provides access to file import and export.

5.4.2 Enumeration Type Documentation

5.4.2.1 enum X3g::Plugin::IO::GltfFormatVersion [strong]

Available glTF format versions.

5.5 X3g::Plugin::Layout Namespace Reference

Namespace provides access to layout.

Classes

- class DbViewport
  Layout viewport item.
- class DbStamp
  Layout stamp item.
- class PageItem
  Base class for layout page items.
- class ViewportItem
  Layout viewport item.
- class StampItem
  Layout stamp item.
- class Page
  Layout page.
- class LayoutManager
  Layout Manager.

Enumerations

- enum DbViewportBackground { DbViewportBackground::Transparent, DbViewportBackground::Color, DbViewportBackground::Skybox }
  Types of DbViewport background.

5.5.1 Detailed Description

Namespace provides access to layout.

5.5.2 Enumeration Type Documentation

5.5.2.1 enum X3g::Plugin::Layout::DbViewportBackground [strong]

Types of DbViewport background.

Enumerator

Transparent The background is transparent.

Color The background has a solid color.

Skybox Use skybox of renderstyle as background.

5.6 X3g::Plugin::Modeling Namespace Reference

Namespace provides access to modeling classes.

Classes

- class Csg

  Collection of functions to do boolean operations on entities.
- class GeometryAudit

  Allows to check database entities for geometry errors.
- class Projection2d

  Allows to project entities on a plane.

Enumerations

- enum GeometryError { GeometryError::BadVertexNormalDirection, GeometryError::ZeroLengthVertexNormal, GeometryError::NonUnitLengthVertexNormal }

  Types of geometry errors.

5.6.1 Detailed Description

Namespace provides access to modeling classes.

5.6.2 Enumeration Type Documentation

5.6.2.1 enum X3g::Plugin::Modeling::GeometryError [strong]

Types of geometry errors.

Enumerator

BadVertexNormalDirection At least one vertex normal has a deviation of more than 90 degrees from a corresponding face normal.

ZeroLengthVertexNormal At least one vertex normal has zero length.

NonUnitLengthVertexNormal At least one vertex normal is not a unit vector.

5.7 X3g::Plugin::Room Namespace Reference

Namespace provides access to different room elements.

Classes

- class Ceiling

  A ceiling.
- class Door

  A door.
- class Floor

  A Floor.
- class GlassPanel

A glass panel

- class Opening

  An opening.
- class PolyWall

  A PolyWall.
- class RoomSettings

  Room default settings.
- class RoomModule

  Room Manager.
- class RoomSplitter

  Divide rooms conceptually not physically.
- class Wall

  A Wall.
- class WallElement

  A wall element.
- class RoomEntity

  Common base class for room entities.
- class WallManager

  Base class for all walls and wall elements.
- class Window

  A window.

Enumerations

- enum DoorLeafType

  Types of Doors
- enum WallSide { WallSide::Left, WallSide::Right }

  Sides of a wall.

5.7.1 Detailed Description

Namespace provides access to different room elements.

5.7.2 Enumeration Type Documentation

5.7.2.1 enum X3g::Plugin::Room::DoorLeafType [strong]

Types of Doors

5.7.2.2 enum X3g::Plugin::Room::WallSide [strong]

Sides of a wall.

Enumerator

Left left side

Right right side

5.8 X3g::Plugin::Tool Namespace Reference

Namespace provides access to different tools.

Classes

- class CreatePolyline

  This tool let the user create a new polyline.
- class CreateRectangle

  This tool let the user create a new rectangle.
- class ToolCallbackEventArgs

  Event args of ToolBase callback.
- class ToolBase

  Base class of internal tools.

Typedefs

- typedef System::EventHandler
  < ToolCallbackEventArgs^ > ToolCallbackEventHandler

  Event handler when a tool finishes work.

5.8.1 Detailed Description

Namespace provides access to different tools.

5.8.2 Typedef Documentation

5.8.2.1 typedef System::EventHandler<ToolCallbackEventArgs^> X3g::Plugin::Tool::ToolCallbackEventHandler

Event handler when a tool finishes work.

The event argument has to be of type ToolCallbackEventArgs.

Chapter 6

Class Documentation

6.1 X3g::Plugin::Articles::ArticleEntity Class Reference

Article interface for entities.

Public Member Functions

- ArticleEntityInfo^ GetInformation ()

  Receives additional article information.
- bool IsUpToDate ()

  Returns true if the article is based on the currently installed product data (OFML).
- IList< ArticleEntity^ >^ GetChildArticles ()

  Receives a list of child articles.
- Image^ GetCustomArticleImage ()

  Returns custom article image for article list.
- void SetCustomArticleImage (Image^ pImage)

  Sets custom image for article list.

Static Public Member Functions

- static ArticleEntity^ FromEntity (DbEntity^ pEntity)

  Retrieves article interface for the given entity.

Properties

- DbBlockReference^ Entity `[get]`

  The according database entity.
- ArticleEntityType Type `[get]`

  The article type.
- ArticleGeometryMode GeometryMode `[get]`

  Geometry mode of an article entity describes if the geometric representation of an article is intended for 2d or 3d planning.

6.1.1 Detailed Description

Article interface for entities.

6.1.2 Member Function Documentation

6.1.2.1 ArticleEntity X3g::Plugin::Articles::ArticleEntity::FromEntity (DbEntity^ *pEntity*) [static]

Retrieves article interface for the given entity.

Returns null if the given entity doesn't represent an article.

ArticleManager::PrepareArticleEntities() needs to be called once before using this function. Otherwise, the funcion may return null or an article with an invalid BasketId.

6.1.2.2 IList< ArticleEntity^ > X3g::Plugin::Articles::ArticleEntity::GetChildArticles ( )

Receives a list of child articles.

6.1.2.3 Image X3g::Plugin::Articles::ArticleEntity::GetCustomArticleImage ( )

Returns custom article image for article list.

Returns null if no custom image is set.

6.1.2.4 ArticleEntityInfo X3g::Plugin::Articles::ArticleEntity::GetInformation ( )

Receives additional article information.

6.1.2.5 bool X3g::Plugin::Articles::ArticleEntity::IsUpToDate ( )

Returns true if the article is based on the currently installed product data (OFML).

Otherwise it is unknown if the article is up-to-date or not.

6.1.2.6 void X3g::Plugin::Articles::ArticleEntity::SetCustomArticleImage (Image^ *pImage*)

Sets custom image for article list.

Set null to remove current custom article image.

6.1.3 Property Documentation

6.1.3.1 DbBlockReference^ X3g::Plugin::Articles::ArticleEntity::Entity [get]

The according database entity.

6.1.3.2 ArticleGeometryMode X3g::Plugin::Articles::ArticleEntity::GeometryMode [get]

Geometry mode of an article entity describes if the geometric representation of an article is intended for 2d or 3d planning.

6.1.3.3 ArticleEntityType X3g::Plugin::Articles::ArticleEntity::Type [get]

The article type.

6.2 X3g::Plugin::Articles::ArticleEntityInfo Class Reference

Provides additional article information.

Public Member Functions

- String^ GetManufacturerName (String^ pLanguage)

  Returns manufacturer name in given language.
- String^ GetSeriesName (String^ pLanguage)

  Returns series name in given language.
- String^ GetArticleNumber (ArticleNumberType pType)

  Returns article number.
- String^ GetDescription (ArticleDescriptionType pType, String^ pLanguage)

  Returns article description in given language.

Properties

- String^ ManufacturerId `[get]`

  The article's manufacturer id.
- String^ SeriesId `[get]`

  The article's series id.
- String^ Inconsistency `[get]`

  The article's inconsistency information.
- Currency^ PurchasePrice `[get]`

  The article's purchase price.
- Currency^ SalesPrice `[get]`

  The article's sales price.
- System::DateTime PriceDate `[get]`

  The article's price date.
- String^ ReferenceNumber `[get]`

  The article's reference number.
- String^ FolderId `[get]`

  The id of the folder the article belongs to.
- String^ BasketId `[get]`

  A globally unique id for the article.

6.2.1 Detailed Description

Provides additional article information.

6.2.2 Member Function Documentation

6.2.2.1 String X3g::Plugin::Articles::ArticleEntityInfo::GetArticleNumber (ArticleNumberType *pType*)

Returns article number.

Parameters

pType	The type of article number.

6.2.2.2 String X3g::Plugin::Articles::ArticleEntityInfo::GetDescription (ArticleDescriptionType *pType*, String^ *pLanguage*)

Returns article description in given language.

Parameters

pType	The type of description.
pLanguage	The preferred language. (optional)

6.2.2.3 String X3g::Plugin::Articles::ArticleEntityInfo::GetManufacturerName (String^ *pLanguage*)

Returns manufacturer name in given language.

Parameters

pLanguage	The preferred language. (optional)

6.2.2.4 String X3g::Plugin::Articles::ArticleEntityInfo::GetSeriesName (String^ *pLanguage*)

Returns series name in given language.

Parameters

pLanguage	The preferred language. (optional)

6.2.3 Property Documentation

6.2.3.1 String^ X3g::Plugin::Articles::ArticleEntityInfo::BasketId [get]

A globally unique id for the article.

Used to identify the article inside the basket.

6.2.3.2 String^ X3g::Plugin::Articles::ArticleEntityInfo::FolderId [get]

The id of the folder the article belongs to.

6.2.3.3 String^ X3g::Plugin::Articles::ArticleEntityInfo::Inconsistency [get]

The article's inconsistency information.

May be empty.

6.2.3.4 String^ X3g::Plugin::Articles::ArticleEntityInfo::ManufacturerId [get]

The article's manufacturer id.

6.2.3.5 System:: DateTime X3g::Plugin::Articles::ArticleEntityInfo::PriceDate [get]

The article's price date.

6.2.3.6 Currency^ X3g::Plugin::Articles::ArticleEntityInfo::PurchasePrice [get]

The article's purchase price.

6.2.3.7 String^ X3g::Plugin::Articles::ArticleEntityInfo::ReferenceNumber [get]

The article's reference number.

6.2.3.8 Currency^ X3g::Plugin::Articles::ArticleEntityInfo::SalesPrice [get]

The article's sales price.

6.2.3.9 String^ X3g::Plugin::Articles::ArticleEntityInfo::SeriesId [get]

The article's series id.

6.3 X3g::Plugin::Articles::ArticleInsertedEventArgs Class Reference

Provides information about the inserted article.

Inherits EventArgs.

Properties

- ArticleEntity^ Article [get]

  The inserted article.

6.3.1 Detailed Description

Provides information about the inserted article.

6.3.2 Property Documentation

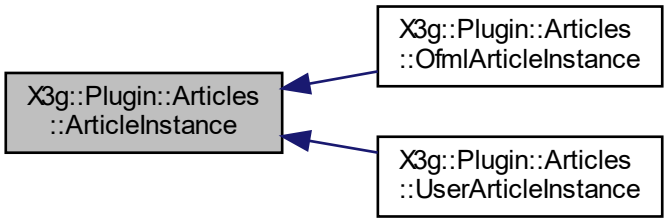
6.3.2.1 ArticleEntity^ X3g::Plugin::Articles::ArticleInsertedEventArgs::Article [get]

The inserted article.

6.4 X3g::Plugin::Articles::ArticleInstance Class Reference

Article instance.

Inheritance diagram for X3g::Plugin::Articles::ArticleInstance:



Public Member Functions

- void Close ()

  Closes the instance and applies all changes.

Properties

- DbEntity^ MainEntity `[get]`

  The main entity id of this instance.
- DbEntity^ SelectedEntity `[get]`

  The currently selected entity of this instance.
- PropertyProvider^ Properties `[get]`

  Provides properties for the currently selected entity.
- bool Valid `[get]`

  Returns true if this instance is still valid.

Events

- static System::EventHandler^ AnyInstanceOpened `[add, remove, raise]`

  Fired when an article has been instantiated.
- static System::EventHandler^ AnyInstanceClosing `[add, remove, raise]`

  Fired when current instance is about to be closed.
- static System::EventHandler^ AnyInstanceClosed `[add, remove, raise]`

  Fired when an instance has been closed.

6.4.1 Detailed Description

Article instance.

6.4.2 Member Function Documentation

6.4.2.1 void X3g::Plugin::Articles::ArticleInstance::Close ( )

Closes the instance and applies all changes.

An instance will be invalid after it was closed. The instance gets automatically closed if another article gets instantiated or if it is deselected.

6.4.3 Property Documentation

6.4.3.1 DbEntity^ X3g::Plugin::Articles::ArticleInstance::MainEntity [get]

The main entity id of this instance.

6.4.3.2 PropertyProvider^ X3g::Plugin::Articles::ArticleInstance::Properties [get]

Provides properties for the currently selected entity.

6.4.3.3 DbEntity^ X3g::Plugin::Articles::ArticleInstance::SelectedEntity [get]

The currently selected entity of this instance.

6.4.3.4 bool X3g::Plugin::Articles::ArticleInstance::Valid [get]

Returns true if this instance is still valid.

An instance will be invalid after it was closed. At any given time there can be only one valid instance.

See Also

ArticleInstance::Close

6.4.4 Event Documentation

6.4.4.1 System:: EventHandler^ X3g::Plugin::Articles::ArticleInstance::AnyInstanceClosed [static],[add],[remove],[raise]

Fired when an instance has been closed.

There is no valid instance at this point and no pending geometry updates.

As this event is static you should consider to add a singleton event handler or to implement a weak event pattern.

6.4.4.2 System:: EventHandler^ X3g::Plugin::Articles::ArticleInstance::AnyInstanceClosing [static],[add],[remove],[raise]

Fired when current instance is about to be closed.

Current instance is still valid at this point and there may be pending geometry updates.

As this event is static you should consider to add a singleton event handler or to implement a weak event pattern.

6.4.4.3 System:: EventHandler^ X3g::Plugin::Articles::ArticleInstance::AnyInstanceOpened [static],[add],[remove],[raise]

Fired when an article has been instantiated.

As this event is static you should consider to add a singleton event handler or to implement a weak event pattern.

6.5 X3g::Plugin::Articles::Catalog Class Reference

OFML product catalog.

Public Member Functions

- IList< CatalogItem^ >^ GetAllItems ()

  Returns all items from the catalog.

Static Public Member Functions

- static IList< Catalog^ >^ GetAll ()

  Returns all catalogs from all registered packages.
- static Catalog^ Get (Package^ pPackage)

  Returns catalog from given package.

Properties

- Articles::Package^ Package [get]

  Package which contains the catalog.
- CatalogItem^ RootItem [get]

  Root folder of catalog.

6.5.1 Detailed Description

OFML product catalog.

When a catalog is used, it may keep multiple files open. When you have finished using it, you should dispose of it either directly via Dispose() or indirectly via using().

6.5.2 Member Function Documentation

6.5.2.1 Catalog X3g::Plugin::Articles::Catalog::Get (Articles::Package^ *pPackage*) [static]

Returns catalog from given package.

6.5.2.2 IList< Catalog^ > X3g::Plugin::Articles::Catalog::GetAll ( ) [static]

Returns all catalogs from all registered packages.

6.5.2.3 IList< CatalogItem^ > X3g::Plugin::Articles::Catalog::GetAllItems ( )

Returns all items from the catalog.

Includes invisible items which are not referenced from catalog tree.

6.5.3 Property Documentation

6.5.3.1 Articles:: Package^ X3g::Plugin::Articles::Catalog::Package [get]

Package which contains the catalog.

6.5.3.2 CatalogItem^ X3g::Plugin::Articles::Catalog::RootItem [get]

Root folder of catalog.

6.6 X3g::Plugin::Articles::CatalogItem Class Reference

OFML catalog item.

Public Member Functions

- System::String^ GetText (CatalogTextPurpose pPurpose)

  Returns text with given purpose in current catalog language.
- System::String^ GetImage (CatalogImagePurpose pPurpose)

  Returns absolute path to image with given purpose.
- System::String^ GetResource (CatalogResourceType pResType)

  Returns resource of given type for current catalog language.
- System::String^ GetResourceDir (CatalogResourceType pResType)

  Returns directory which contains resource files of given type.
- IList< CatalogItem^ >^ GetChildren ()

  Returns children of catalog item.

Properties

- CatalogItemType Type [get]

  Type of catalog item.
- System::String^ ArticleNumber [get]

  Article number of catalog item.
- System::String^ VariantCode [get]

  Variant code of catalog item.
- System::String^ PackageName [get]

  Name of package which contains the referenced item.
- bool InCatalogTree [get]

  Returns if catalog item is referenced from catalog tree.

6.6.1 Detailed Description

OFML catalog item.

6.6.2 Member Function Documentation

6.6.2.1 IList< CatalogItem^ > X3g::Plugin::Articles::CatalogItem::GetChildren ( )

Returns children of catalog item.

6.6.2.2 System::String X3g::Plugin::Articles::CatalogItem::GetImage (CatalogImagePurpose *pPurpose*)

Returns absolute path to image with given purpose.

6.6.2.3 System::String X3g::Plugin::Articles::CatalogItem::GetResource (CatalogResourceType *pResType*)

Returns resource of given type for current catalog language.

Filenames are relative.

6.6.2.4 System::String X3g::Plugin::Articles::CatalogItem::GetResourceDir (CatalogResourceType *pResType*)

Returns directory which contains resource files of given type.

6.6.2.5 System::String X3g::Plugin::Articles::CatalogItem::GetText (CatalogTextPurpose *pPurpose*)

Returns text with given purpose in current catalog language.

6.6.3 Property Documentation

6.6.3.1 System:: String^ X3g::Plugin::Articles::CatalogItem::ArticleNumber [get]

Article number of catalog item.

6.6.3.2 bool X3g::Plugin::Articles::CatalogItem::InCatalogTree [get]

Returns if catalog item is referenced from catalog tree.

6.6.3.3 System:: String^ X3g::Plugin::Articles::CatalogItem::PackageName [get]

Name of package which contains the referenced item.

6.6.3.4 CatalogItemType X3g::Plugin::Articles::CatalogItem::Type [get]

Type of catalog item.

6.6.3.5 System:: String^ X3g::Plugin::Articles::CatalogItem::VariantCode [get]

Variant code of catalog item.

6.7 X3g::Plugin::Articles::CustomTextField Class Reference

Custom text field of user article.

Properties

- String^ Key [get]

  The key of the text field.
- String^ Value [get, set]

  Value of the text field.
- bool Editable [get, set]

  Determines if the user should be allowed to edit the text.
- bool Visible [get, set]

  Determines if the text should be visible to the user.

6.7.1 Detailed Description

Custom text field of user article.

6.7.2 Property Documentation

6.7.2.1 bool X3g::Plugin::Articles::CustomTextField::Editable [get],[set]

Determines if the user should be allowed to edit the text.

Exceptions

System::Exception	An exception will be thrown if according article instance is no longer valid.

6.7.2.2 String^ X3g::Plugin::Articles::CustomTextField::Key [get]

The key of the text field.

Exceptions

System::Exception	An exception will be thrown if according article instance is no longer valid.

6.7.2.3 String^ X3g::Plugin::Articles::CustomTextField::Value [get],[set]

Value of the text field.

Exceptions

System::Exception	An exception will be thrown if according article instance is no longer valid.

6.7.2.4 bool X3g::Plugin::Articles::CustomTextField::Visible [get],[set]

Determines if the text should be visible to the user.

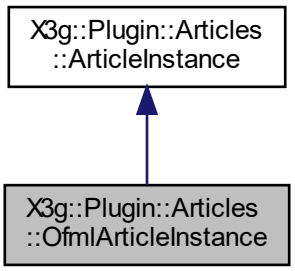
Exceptions

System::Exception	An exception will be thrown if according article instance is no longer valid.

6.8 X3g::Plugin::Articles::OfmlArticleInstance Class Reference

OFML article instance.

Inheritance diagram for X3g::Plugin::Articles::OfmlArticleInstance:



Public Member Functions

- bool SetSelectedObject (OfmlObject^ pObject)

  Changes the selected ofml object of this instance.
- OfmlObject^ GetObject (String^ pObjectName)

  Returns the ofml object with the given name.
- void SetObjectStateModified ()

  Sets Modified flag which will cause an update of database entities at the latest when instance is closed.

Properties

- OfmlObject^ RootObject `[get]`

  The root ofml object of this instance.
- OfmlObject^ SelectedObject `[get]`

  The currently selected ofml object of this instance.

Additional Inherited Members

6.8.1 Detailed Description

OFML article instance.

6.8.2 Member Function Documentation

6.8.2.1 OfmlObject X3g::Plugin::Articles::OfmlArticleInstance::GetObject (String^ *pObjectName*)

Returns the ofml object with the given name.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

6.8.2.2 void X3g::Plugin::Articles::OfmlArticleInstance::SetObjectStateModified ( )

Sets Modified flag which will cause an update of database entities at the latest when instance is closed.

Clients should call this function if they modified ofml objects associated with this instance through ofml method calls.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

See Also

OfmlObject::CallMethod

6.8.2.3 bool X3g::Plugin::Articles::OfmlArticleInstance::SetSelectedObject (OfmlObject^ *pObject*)

Changes the selected ofml object of this instance.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

6.8.3 Property Documentation

6.8.3.1 OfmlObject^ X3g::Plugin::Articles::OfmlArticleInstance::RootObject [get]

The root ofml object of this instance.

6.8.3.2 OfmlObject^ X3g::Plugin::Articles::OfmlArticleInstance::SelectedObject [get]

The currently selected ofml object of this instance.

IArticleInstance::PropertyProvider and IArticleInstance::getArticleInformation() operate on the selected object.

6.9 X3g::Plugin::Articles::OfmlEval Class Reference

A class covering the evaluation of OFML expressions.

Public Member Functions

- String^ Eval (String^ pExpression)

  Evaluate a given String as OFML expression.

6.9.1 Detailed Description

A class covering the evaluation of OFML expressions.

6.9.2 Member Function Documentation

6.9.2.1 String X3g::Plugin::Articles::OfmlEval::Eval (String^ *pExpression*)

Evaluate a given String as OFML expression.

Returns

Processed answer of the given expression.

6.10 X3g::Plugin::Articles::OfmlObject Class Reference

Ofml Object.

Public Member Functions

- bool IsExisting ()

  Returns true if this object still exists.
- bool IsSelectable ()

  Returns true if this object is selectable.
- bool IsArticle ()

  Returns true if this object is an article.
- bool IsA (String^ pTypeName)

  Returns true if this object is an instance of the given type.
- bool HasMember (String^ pMemberName)

  Returns true if this object has a member with the given name (e.g.
- bool CallMethod (String^ pMethodName, String^ pArguments,[Out] String^ %pResult)

  Calls/Executes the given method with the given arguments and returns the results.
- OfmlObject^ GetParent ()

  Returns the parent of this object.
- IList< OfmlObject^ >^ GetChildren (bool pElementsOnly)

  Returns the children of this object.

Properties

- String^ Name `[get]`

  The objects name.
- bool IsValid `[get]`

  Returns true if the IOfmlArticleInstance for this object is valid.

6.10.1 Detailed Description

Ofml Object.

6.10.2 Member Function Documentation

6.10.2.1 bool X3g::Plugin::Articles::OfmlObject::CallMethod (String^ *pMethodName,* String^ *pArguments,* [Out] String^ % *pResult*)

Calls/Executes the given method with the given arguments and returns the results.

Also consider calling if method call modified object state.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.10.2.2 IList< OfmlObject^ > X3g::Plugin::Articles::OfmlObject::GetChildren (bool *pElementsOnly*)

Returns the children of this object.

If pElementsOnly is true only the children which are elements will be returned.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.10.2.3 OfmlObject X3g::Plugin::Articles::OfmlObject::GetParent ( )

Returns the parent of this object.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.10.2.4 bool X3g::Plugin::Articles::OfmlObject::HasMember (String^ *pMemberName*)

Returns true if this object has a member with the given name (e.g.

a method).

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.10.2.5 bool X3g::Plugin::Articles::OfmlObject::IsA (String^ *pTypeName*)

Returns true if this object is an instance of the given type.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.10.2.6 bool X3g::Plugin::Articles::OfmlObject::IsArticle ( )

Returns true if this object is an article.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.10.2.7 bool X3g::Plugin::Articles::OfmlObject::IsExisting ( )

Returns true if this object still exists.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.10.2.8 bool X3g::Plugin::Articles::OfmlObject::IsSelectable ( )

Returns true if this object is selectable.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.10.3 Property Documentation

6.10.3.1 bool X3g::Plugin::Articles::OfmlObject::IsValid [get]

Returns true if the IOfmlArticleInstance for this object is valid.

6.10.3.2 String^ X3g::Plugin::Articles::OfmlObject::Name [get]

The objects name.

6.11 X3g::Plugin::Articles::Package Class Reference

Provides information about registered OFML packages.

Public Member Functions

- IList< System::String^ >^ GetLanguages ()

  Returns languages which are supported by the package.
- System::String^ GetManufacturerName (System::String^ pLanguage)

  Returns name of manufacturer in given language.
- System::String^ GetProgramName (System::String^ pLanguage)

  Returns name of the program in given language.
- System::String^ GetValue (System::String^ pKey, System::String^ pLanguage)

  Returns arbitrary values from package profile (see DSR specification).
- IList< Package^ >^ GetCatalogPackages ()

  Returns other catalog packages which reference products from this package.

Static Public Member Functions

- static IList< Package^ >^ GetAll ()

  Returns all registered packages.
- static Package^ Get (System::String^ pPackageName)

  Returns registered package with given name.

Properties

- System::String^ Manufacturer [get]

  Unique OFML identifier of the manufacturer (e.g.
- System::String^ Program [get]

  Unique OFML identifier of the OFML library (series) within the manufacturer (e.g.
- System::String^ DistributionRegion [get]

  Identifier of the distribution region (e.g.
- System::Version^ Version [get]

  Version number of the package.
- System::String^ ProgId [get]

  OFML program id.
- System::String^ ManufacturerId [get]

  Unique commercial identifier of the manufacturer (e.g.
- System::String^ ProgramIds [get]

Unique commercial identifiers of commercial product lines within the manufacturer which are mapped in this package.

- System::String^ DataDir [get]

  Root dictionary for product data (e.g.
- System::String^ PackageDir [get]

  Root dictionary for current package (e.g.
- System::String^ Path [get]

  Path to package data for actual distribution region and version (e.g.
- PackageType Type [get]

  Type of the package.
- System::String^ VersionedName [get]

  Name of package including version (e.g.
- System::String^ ManufacturerLogoSmall [get]

  Returns absolute path to small manufacturer logo ([1-100]x20 pixel).
- System::String^ ManufacturerLogoLarge [get]

  Returns absolute path to large manufacturer logo ([1-200]x40 pixel).

6.11.1 Detailed Description

Provides information about registered OFML packages.

6.11.2 Member Function Documentation

6.11.2.1 Package X3g::Plugin::Articles::Package::Get (System::String^ *pPackageName*) [static]

Returns registered package with given name.

Returns

Returns null if package was not found.

6.11.2.2 IList< Package^ > X3g::Plugin::Articles::Package::GetAll ( ) [static]

Returns all registered packages.

6.11.2.3 IList< Package^ > X3g::Plugin::Articles::Package::GetCatalogPackages ( )

Returns other catalog packages which reference products from this package.

Returns

Returns null if key was not found.

6.11.2.4 IList< System::String^ > X3g::Plugin::Articles::Package::GetLanguages ( )

Returns languages which are supported by the package.

6.11.2.5 System::String ^ X3g::Plugin::Articles::Package::GetManufacturerName (System::String^ *pLanguage*)

Returns name of manufacturer in given language.

Default manufacturer name is returned if language is not available or null.

6.11.2.6 System::String ^ X3g::Plugin::Articles::Package::GetProgramName (System::String^ *pLanguage*)

Returns name of the program in given language.

Default program name is returned if language is not available or null.

6.11.2.7 System::String X3g::Plugin::Articles::Package::GetValue (System::String^ *pKey,* System::String^ *pLanguage*)

Returns arbitrary values from package profile (see DSR specification).

Parameters

pKey	See DSR specification for possible values.
pLanguage	Set null for values which are not localized.

Returns

Returns null if key was not found.

6.11.3 Property Documentation

6.11.3.1 System:: String^ X3g::Plugin::Articles::Package::DataDir [get]

Root dictionary for product data (e.g.

C:/EasternGraphics).

6.11.3.2 System:: String^ X3g::Plugin::Articles::Package::DistributionRegion [get]

Identifier of the distribution region (e.g.

DE).

6.11.3.3 System:: String^ X3g::Plugin::Articles::Package::Manufacturer [get]

Unique OFML identifier of the manufacturer (e.g.

egr).

6.11.3.4 System:: String^ X3g::Plugin::Articles::Package::ManufacturerId [get]

Unique commercial identifier of the manufacturer (e.g.

EG).

6.11.3.5 System:: String^ X3g::Plugin::Articles::Package::ManufacturerLogoLarge [get]

Returns absolute path to large manufacturer logo ([1-200]x40 pixel).

6.11.3.6 System:: String^ X3g::Plugin::Articles::Package::ManufacturerLogoSmall [get]

Returns absolute path to small manufacturer logo ([1-100]x20 pixel).

6.11.3.7 System:: String^ X3g::Plugin::Articles::Package::PackageDir [get]

Root dictionary for current package (e.g.

C:/EasternGraphics/egr/office2).

6.11.3.8 System:: String^ X3g::Plugin::Articles::Package::Path [get]

Path to package data for actual distribution region and version (e.g.

C:/EasternGraphics/egr/office2/EGR/1).

6.11.3.9 System:: String^ X3g::Plugin::Articles::Package::ProgId [get]

OFML program id.

(e.g.)

6.11.3.10 System:: String^ X3g::Plugin::Articles::Package::Program [get]

Unique OFML identifier of the OFML library (series) within the manufacturer (e.g.

office2).

6.11.3.11 System:: String^ X3g::Plugin::Articles::Package::ProgramIds [get]

Unique commercial identifiers of commercial product lines within the manufacturer which are mapped in this package.

Identifiers are separated by semicolon. (e.g. AX;BX)

6.11.3.12 PackageType X3g::Plugin::Articles::Package::Type [get]

Type of the package.

6.11.3.13 System:: Version^ X3g::Plugin::Articles::Package::Version [get]

Version number of the package.

6.11.3.14 System:: String^ X3g::Plugin::Articles::Package::VersionedName [get]

Name of package including version (e.g.

::egr::office2::1).

6.12 X3g::Plugin::Articles::ProjectSettings Class Reference

ProjectSettings.

Properties

Contact and address settings

- System::String^ CustomerNumber [get, set]

The customer number.

- System::String^ CustomerCompany [get, set]
  The customer's company.
- System::String^ CustomerSalutation [get, set]
  The customer's salutation.
- System::String^ CustomerFirstName [get, set]
  The customer's first name.
- System::String^ CustomerLastName [get, set]
  The customer's last name.
- System::String^ CustomerStreet [get, set]
  The customer's street.
- System::String^ CustomerPostalCode [get, set]
  The customer's postal code.
- System::String^ CustomerCity [get, set]
  The customer's city.
- System::String^ ContactName [get, set]
  The contact's name.
- System::String^ ContactPhone [get, set]
  The contact's phone number.
- System::String^ ContactEmail [get, set]
  The contact's email address.

Calculation settings

- X3g::Plugin::Currency^ VAT [get, set]
  The VAT (value added tax) for this project.
- System::String^ CurrencyUnit [get, set]
  The currency unit for this project.
- X3g::Plugin::Currency^ HeaderAddCharge [get, set]
  The header add charge.
- X3g::Plugin::Currency^ HeaderDiscount [get, set]
  The header discount.
- X3g::Plugin::Currency^ OverallArticleAddCharge [get, set]
  The overall article add charge.
- X3g::Plugin::Currency^ OverallArticleDiscount [get, set]
  The overall article discount.

6.12.1 Detailed Description

ProjectSettings.

6.12.2 Property Documentation

6.12.2.1 System:: String^ X3g::Plugin::Articles::ProjectSettings::ContactEmail [get],[set]

The contact's email address.

6.12.2.2 System:: String^ X3g::Plugin::Articles::ProjectSettings::ContactName [get],[set]

The contact's name.

6.12.2.3 System:: String^ X3g::Plugin::Articles::ProjectSettings::ContactPhone [get],[set]

The contact's phone number.

6.12.2.4 System:: String^ X3g::Plugin::Articles::ProjectSettings::CurrencyUnit [get],[set]

The currency unit for this project.

Contains a 3 letter ISO currency code.

6.12.2.5 System:: String^ X3g::Plugin::Articles::ProjectSettings::CustomerCity [get],[set]

The customer's city.

6.12.2.6 System:: String^ X3g::Plugin::Articles::ProjectSettings::CustomerCompany [get],[set]

The customer's company.

6.12.2.7 System:: String^ X3g::Plugin::Articles::ProjectSettings::CustomerFirstName [get],[set]

The customer's first name.

6.12.2.8 System:: String^ X3g::Plugin::Articles::ProjectSettings::CustomerLastName [get],[set]

The customer's last name.

6.12.2.9 System:: String^ X3g::Plugin::Articles::ProjectSettings::CustomerNumber [get],[set]

The customer number.

6.12.2.10 System:: String^ X3g::Plugin::Articles::ProjectSettings::CustomerPostalCode [get],[set]

The customer's postal code.

6.12.2.11 System:: String^ X3g::Plugin::Articles::ProjectSettings::CustomerSalutation [get],[set]

The customer's salutation.

6.12.2.12 System:: String^ X3g::Plugin::Articles::ProjectSettings::CustomerStreet [get],[set]

The customer's street.

6.12.2.13 X3g:: Plugin:: Currency^ X3g::Plugin::Articles::ProjectSettings::HeaderAddCharge [get],[set]

The header add charge.

Only relative values are supported.

6.12.2.14 X3g:: Plugin:: Currency^ X3g::Plugin::Articles::ProjectSettings::HeaderDiscount [get],[set]

The header discount.

Only relative values are supported.

6.12.2.15 X3g:: Plugin:: Currency^ X3g::Plugin::Articles::ProjectSettings::OverallArticleAddCharge [get],[set]

The overall article add charge.

Only relative values are supported.

6.12.2.16 X3g:: Plugin:: Currency^ X3g::Plugin::Articles::ProjectSettings::OverallArticleDiscount [get],[set]

The overall article discount.

Only relative values are supported.

6.12.2.17 X3g:: Plugin:: Currency^ X3g::Plugin::Articles::ProjectSettings::VAT [get],[set]

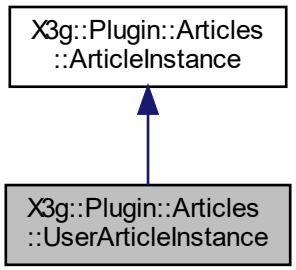
The VAT (value added tax) for this project.

Only relative values are supported.

6.13 X3g::Plugin::Articles::UserArticleInstance Class Reference

User article instance.

Inheritance diagram for X3g::Plugin::Articles::UserArticleInstance:



Public Member Functions

- CustomTextField^ AddCustomTextField (String^ pKey, String^ pValue, bool pEditable)

  Adds a custom text field.
- CustomTextField^ GetCustomTextField (String^ pKey)

  Returns custom text field.
- IList< CustomTextField^ >^ GetAllCustomTextFields ()

  Returns list of existing custom text fields.
- void DeleteCustomTextField (String^ pKey)

  Removes a custom text field.

Properties

- String^ ManufacturerId [get, set]
  The article's manufacturer id.
- String^ SeriesId [get, set]
  The article's series id.
- String^ ArticleNumber [get, set]
  The article's number.
- Currency^ PurchasePrice [get, set]
  The purchase price.
- Currency^ SalesPrice [get, set]
  The sales price.
- String^ ShortText [get, set]
  The article's short description.
- String^ LongText [get, set]
  The article's long description.
- String^ VariantText [get, set]
  The article's variant text.
- String^ AdditionalText [get, set]
  The article's additional text.

Additional Inherited Members

6.13.1 Detailed Description

User article instance.

6.13.2 Member Function Documentation

6.13.2.1 CustomTextField X3g::Plugin::Articles::UserArticleInstance::AddCustomTextField (String^ *pKey*, String^ *pValue*, bool *pEditable*)

Adds a custom text field.

Overwrites existing text field with same key with given values. New field is visible by default.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

6.13.2.2 void X3g::Plugin::Articles::UserArticleInstance::DeleteCustomTextField (String^ *pKey*)

Removes a custom text field.

Parameters

pKey	The key of the text field.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

6.13.2.3 IList< CustomTextField^ > X3g::Plugin::Articles::UserArticleInstance::GetAllCustomTextFields ( )

Returns list of existing custom text fields.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

6.13.2.4 CustomTextField X3g::Plugin::Articles::UserArticleInstance::GetCustomTextField (String^ *pKey*)

Returns custom text field.

Parameters

pKey	The key of the text field.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

6.13.3 Property Documentation

6.13.3.1 String^ X3g::Plugin::Articles::UserArticleInstance::AdditionalText [get],[set]

The article's additional text.

Exceptions

System::Exception	Setter throws exception if this instance is no longer valid.

6.13.3.2 String^ X3g::Plugin::Articles::UserArticleInstance::ArticleNumber [get],[set]

The article's number.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

6.13.3.3 String^ X3g::Plugin::Articles::UserArticleInstance::LongText [get],[set]

The article's long description.

Exceptions

System::Exception	Setter throws exception if this instance is no longer valid.

6.13.3.4 String^ X3g::Plugin::Articles::UserArticleInstance::ManufacturerId [get],[set]

The article's manufacturer id.

Each manufacturer has a unique ID assigned by EasternGraphics. If you set the manufacturer id to an invalid value, articles may not be processed in some applications.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.
System::ArgumentException	An exception will be thrown if set to an invalid ID.

6.13.3.5 Currency^ X3g::Plugin::Articles::UserArticleInstance::PurchasePrice [get],[set]

The purchase price.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

6.13.3.6 Currency^ X3g::Plugin::Articles::UserArticleInstance::SalesPrice [get],[set]

The sales price.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

6.13.3.7 String^ X3g::Plugin::Articles::UserArticleInstance::SeriesId [get],[set]

The article's series id.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.
System::ArgumentException	An exception will be thrown if set to an invalid ID.

6.13.3.8 String^ X3g::Plugin::Articles::UserArticleInstance::ShortText [get],[set]

The article's short description.

Exceptions

System::Exception	Setter throws exception if this instance is no longer valid.

6.13.3.9 String^ X3g::Plugin::Articles::UserArticleInstance::VariantText [get],[set]

The article's variant text.

Exceptions

System::Exception	Setter throws exception if this instance is no longer valid.

6.14 X3g::Plugin::BackgroundParams Class Reference

Contains background parameter for Views.

Properties

- System::Drawing::Color Color [get, set]

  The Background color.
- double Rotation [get, set]

  Rotation of panoramic background image in degrees.
- double Fov [get, set]

  The fov (field of view) which is used for spherical background rendering.
- Image^ SphericalImage [get]

  Spherical background image.

6.14.1 Detailed Description

Contains background parameter for Views.

6.14.2 Property Documentation

6.14.2.1 System:: Drawing:: Color X3g::Plugin::BackgroundParams::Color [get], [set]

The Background color.

Used if no image is set.

6.14.2.2 double X3g::Plugin::BackgroundParams::Fov [get], [set]

The fov (field of view) which is used for spherical background rendering.

Value is in degrees. Values are clamped between 5 and 100.

6.14.2.3 double X3g::Plugin::BackgroundParams::Rotation [get], [set]

Rotation of panoramic background image in degrees.

Background is rotated around the z-axis by given angle.

6.14.2.4 Image^ X3g::Plugin::BackgroundParams::SphericalImage [get]

Spherical background image.

6.15 X3g::Plugin::Block Class Reference

Blocks are containers for entities within a planning.

Public Member Functions

- IList< DbEntity^ >^ CloneEntities (IEnumerable< DbEntity^ >^pEntities)

  Adds clones of given entities to this block.
- IList< DbBlockReference^ >^ GetReferences (bool pDirectOnly)

  Returns all references to this block.
- IList< DbEntity^ >^ GetEntities ()

  Returns all entities which are owned by this block.
- IList< DbEntity^ >^ GetEntities (System::String^ pCustomType)

  Returns all entities which are owned by this block and have the given custom type.

Public Attributes

- template<typename T >
  where T: DbEntity IList<T>^ GetEntities() { List<T>^ entities = gcnew List<T>()

  Returns all entities of given type which are owned by this block.

Properties

- String^ Name [get, set]

  The unique name of the block.
- GeVec3d Origin [get, set]

  Base point of the block in WCS coordinates.
- bool IsModelSpace [get]

  Returns if this block is the modelspace block.

6.15.1 Detailed Description

Blocks are containers for entities within a planning.

They own all entities they contain. There are two special Blocks that are always present in every planning: "*MODEL_SPACE" and "*PAPER_SPACE", where the model space is the block that contains all "top level" entities of the 3D space in the planner.

6.15.2 Member Function Documentation

6.15.2.1 IList< DbEntity^ > X3g::Plugin::Block::CloneEntities (IEnumerable< DbEntity^ >^ *pEntities*)

Adds clones of given entities to this block.

Returns the clones.

6.15.2.2 IList< X3g::Plugin::DbEntity^ > X3g::Plugin::Block::GetEntities ( )

Returns all entities which are owned by this block.

6.15.2.3 IList< X3g::Plugin::DbEntity^ > X3g::Plugin::Block::GetEntities (System::String^ *pCustomType*)

Returns all entities which are owned by this block and have the given custom type.

6.15.2.4 IList< DbBlockReference^ > X3g::Plugin::Block::GetReferences (bool *pDirectOnly*)

Returns all references to this block.

If pDirectOnly is false it also returns references to blocks which own direct or indirect references to this block.

6.15.3 Member Data Documentation

6.15.3.1 template<typename T > where X3g::Plugin::Block::T

Returns all entities of given type which are owned by this block.

6.15.4 Property Documentation

6.15.4.1 bool X3g::Plugin::Block::IsModelSpace [get]

Returns if this block is the modelspace block.

6.15.4.2 String^ X3g::Plugin::Block::Name [get], [set]

The unique name of the block.

6.15.4.3 GeVec3d X3g::Plugin::Block::Origin [get], [set]

Base point of the block in WCS coordinates.

6.16 X3g::Plugin::BlockManager Class Reference

Block Manager.

Public Member Functions

- IList< String^ >^ GetAllBlockNames ()

  Returns a list of all block names.
- Block^ CreateBlock (String^ pBlockName)

  Creates a new block.
- Block^ GetBlock (String^ pBlockName)

  Returns a block with given name.
- void DeleteBlock (String^ pBlockName)

  Deletes a block and each block reference referencing that block.
- void SaveBlock (String^ pBlockName, Stream^ pDwgStream, SaveBlockParams pParams)

  Saves block content to a dwg stream.
- DbBlockReference^ Group (IEnumerable< DbEntity^ >^pEntities, System::String^ pBlockName)

  Groups a collection of entities and returns the according DbBlockReference.
- IList< DbEntity^ >^ Ungroup (DbBlockReference^ pBlockRef)

  Tries to explode a block reference and returns the resulting entities.

Properties

- bool BlockUniquificationEnabled [get, set]

  Global setting which allows to control if a block is automatically made unique when user edits block content, i.e.

6.16.1 Detailed Description

Block Manager.

6.16.2 Member Function Documentation

6.16.2.1 X3g::Plugin::Block X3g::Plugin::BlockManager::CreateBlock (String^ *pBlockName*)

Creates a new block.

Anonymous block naming is supported (e.g. "*E"). Unique block name will be generated if given block name is null or empty.

6.16.2.2 void X3g::Plugin::BlockManager::DeleteBlock (String^ *pBlockName*)

Deletes a block and each block reference referencing that block.

6.16.2.3 IList< System::String^ > X3g::Plugin::BlockManager::GetAllBlockNames ( )

Returns a list of all block names.

6.16.2.4 X3g::Plugin::Block X3g::Plugin::BlockManager::GetBlock (String^ *pBlockName*)

Returns a block with given name.

6.16.2.5 DbBlockReference X3g::Plugin::BlockManager::Group (IEnumerable< DbEntity^ >^ *pEntities*, System::String^ *pBlockName*)

Groups a collection of entities and returns the according DbBlockReference.

Works for modelspace entities only. Block with given name must not exist.

Parameters

pEntities	Enumerable of the entities to be grouped.
pBlockName	The Block name for the group. Block name will be generated if parameter is null or empty.

6.16.2.6 void X3g::Plugin::BlockManager::SaveBlock (String^ *pBlockName*, Stream^ *pDwgStream*, SaveBlockParams *pParams*)

Saves block content to a dwg stream.

Returns false if saving failed.

If pVersion = Undefined the internal/loaded version of the document is used.

6.16.2.7 IList< DbEntity^ > X3g::Plugin::BlockManager::Ungroup (DbBlockReference^ *pBlockRef*)

Tries to explode a block reference and returns the resulting entities.

Block reference must be in modelspace. Returns null if explode failed.

6.16.3 Property Documentation

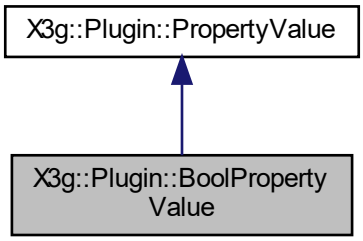
6.16.3.1 bool X3g::Plugin::BlockManager::BlockUniquificationEnabled [get], [set]

Global setting which allows to control if a block is automatically made unique when user edits block content, i.e. when block is opened. Enabled by default.

6.17 X3g::Plugin::BoolPropertyValue Class Reference

Boolean Property Value.

Inheritance diagram for X3g::Plugin::BoolPropertyValue:



Public Member Functions

- BoolPropertyValue (bool pValue)

  Constructor.

Properties

- bool Value `[get, set]`

  The boolean value.

6.17.1 Detailed Description

Boolean Property Value.

6.17.2 Constructor & Destructor Documentation

6.17.2.1 X3g::Plugin::BoolPropertyValue::BoolPropertyValue (bool *pValue*)

Constructor.

Parameters

pValue	Initial boolean value.

6.17.3 Property Documentation

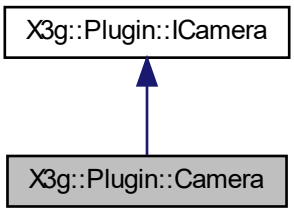
6.17.3.1 bool X3g::Plugin::BoolPropertyValue::Value `[get],[set]`

The boolean value.

6.18 X3g::Plugin::Camera Class Reference

Implementation of a common camera without visual representation in the planning.

Inheritance diagram for X3g::Plugin::Camera:



Public Member Functions

- Camera ()

  Default Constructor.
- Camera (X3g::Plugin::ICamera^ pOther)

  Copy Constructor.
- virtual void GetLookAt ([Out] X3g::Plugin::GeVec3d% pEye,[Out] X3g::Plugin::GeVec3d% pCenter,[Out] X3g::Plugin::GeVec3d% pUp)

  Returns the camera postion and the camera target.
- virtual void SetLookAt (X3g::Plugin::GeVec3d pEye, X3g::Plugin::GeVec3d pCenter, X3g::Plugin::GeVec3d pUp)

  Sets the camera postion and the camera target.
- virtual void ZoomToExtents (IEnumerable< DbEntity^ >^pEntities, double pMargin, double pAspect)

  Fits camera to given entities.
- virtual void CopyFrom (X3g::Plugin::ICamera^ pOther)

  Assigns values of another camera object to this.

Additional Inherited Members

6.18.1 Detailed Description

Implementation of a common camera without visual representation in the planning.

This camera is not persistent.

6.18.2 Constructor & Destructor Documentation

6.18.2.1 X3g::Plugin::Camera::Camera ( )

Default Constructor.

6.18.2.2 X3g::Plugin::Camera::Camera (X3g::Plugin::ICamera^ *pOther*)

Copy Constructor.

6.18.3 Member Function Documentation

6.18.3.1 virtual void X3g::Plugin::Camera::CopyFrom (X3g::Plugin::ICamera^ *pOther*) [virtual]

Assigns values of another camera object to this.

Implements X3g::Plugin::ICamera.

6.18.3.2 virtual void X3g::Plugin::Camera::GetLookAt ([Out] X3g::Plugin::GeVec3d% *pEye*, [Out] X3g::Plugin::GeVec3d% *pCenter*, [Out] X3g::Plugin::GeVec3d% *pUp*) [virtual]

Returns the camera postion and the camera target.

Implements X3g::Plugin::ICamera.

6.18.3.3 virtual void X3g::Plugin::Camera::SetLookAt (X3g::Plugin::GeVec3d *pEye*, X3g::Plugin::GeVec3d *pCenter*, X3g::Plugin::GeVec3d *pUp*) [virtual]

Sets the camera postion and the camera target.

Implements X3g::Plugin::ICamera.

6.18.3.4 void X3g::Plugin::Camera::ZoomToExtents (IEnumerable< DbEntity^ >^ *pEntities*, double *pMargin*, double *pAspect*) [virtual]

Fits camera to given entities.

Implements X3g::Plugin::ICamera.

6.19 X3g::Plugin::CameraAnimation Class Reference

Camera Animation.

Public Member Functions

- CameraAnimationNode^ CreateNode ()

  Creates a new node which is not yet part of the animation.
- IList< CameraAnimationNode^ >^ GetNodes ()

  Returns the nodes of this animation.
- void SetNodes (IEnumerable< CameraAnimationNode^ >^pNodes)

  Sets the nodes of this animation.
- CameraAnimationInfo GetAnimationInfo ()

  Returns additional information about the animation.
- IList< CameraAnimationFrame >^ getAnimationFrames (unsigned int pStartFrame, unsigned int pFPS)

  Returns the interpolated frames of the animation for the given frames per second.

Properties

- System::String^ Name [get, set]

  The name of the animation.
- CameraInterpolation Interpolation [get, set]

  The interpolation of the animation.
- bool ClosedPath [get, set]

  True if the path of the animation is closed.

6.19.1 Detailed Description

Camera Animation.

6.19.2 Member Function Documentation

6.19.2.1 CameraAnimationNode X3g::Plugin::CameraAnimation::CreateNode ( )

Creates a new node which is not yet part of the animation.

6.19.2.2 IList< X3g::Plugin::CameraAnimationFrame > X3g::Plugin::CameraAnimation::getAnimationFrames (unsigned int *pStartFrame*, unsigned int *pFPS*)

Returns the interpolated frames of the animation for the given frames per second.

6.19.2.3 X3g::Plugin::CameraAnimationInfo X3g::Plugin::CameraAnimation::GetAnimationInfo ( )

Returns additional information about the animation.

6.19.2.4 IList< CameraAnimationNode^ > X3g::Plugin::CameraAnimation::GetNodes ( )

Returns the nodes of this animation.

6.19.2.5 void X3g::Plugin::CameraAnimation::SetNodes (IEnumerable< CameraAnimationNode^ >^ *pNodes*)

Sets the nodes of this animation.

6.19.3 Property Documentation

6.19.3.1 bool X3g::Plugin::CameraAnimation::ClosedPath [get],[set]

True if the path of the animation is closed.

6.19.3.2 CameraInterpolation X3g::Plugin::CameraAnimation::Interpolation [get],[set]

The interpolation of the animation.

6.19.3.3 System:: String^ X3g::Plugin::CameraAnimation::Name [get],[set]

The name of the animation.

6.20 X3g::Plugin::CameraAnimationFrame Struct Reference

Camera Animation Frame.

Public Attributes

- GeVec3d Eye

  Camera Postion.
- GeVec3d Center

  Camera Target.

6.20.1 Detailed Description

Camera Animation Frame.

6.20.2 Member Data Documentation

6.20.2.1 GeVec3d X3g::Plugin::CameraAnimationFrame::Center

Camera Target.

6.20.2.2 GeVec3d X3g::Plugin::CameraAnimationFrame::Eye

Camera Postion.

6.21 X3g::Plugin::CameraAnimationInfo Struct Reference

Camera Animation Information.

Public Attributes

- double Duration

  Total animation duration in milliseconds.
- double PathLength

  Total length of the animation path.

6.21.1 Detailed Description

Camera Animation Information.

6.21.2 Member Data Documentation

6.21.2.1 double X3g::Plugin::CameraAnimationInfo::Duration

Total animation duration in milliseconds.

6.21.2.2 double X3g::Plugin::CameraAnimationInfo::PathLength

Total length of the animation path.

6.22 X3g::Plugin::CameraAnimationNode Class Reference

Camera Animation Node.

Inherited by X3g::Plugin::ICameraAnimationNode.

Public Member Functions

- void GetLookAt ([Out] GeVec3d% pEye,[Out] GeVec3d% pCenter)

  Returns the camera position and the camera target.
- void SetLookAt (GeVec3d pEye, GeVec3d pCenter)

  Sets the camera position and the camera target.

Properties

- System::String^ Name `[get, set]`

  The name of the node.
- double Speed `[get, set]`

  Sets the speed (m/s).
- double WaitTime `[get, set]`

  The wait time defines how long the animation stops/waits at the node's camera position.
- CameraAnimationNodeFlags Flags `[get, set]`

  Additional node flags.

6.22.1 Detailed Description

Camera Animation Node.

6.22.2 Member Function Documentation

6.22.2.1 void X3g::Plugin::CameraAnimationNode::GetLookAt ([Out] GeVec3d% *pEye*, [Out] GeVec3d% *pCenter*)

Returns the camera position and the camera target.

6.22.2.2 void X3g::Plugin::CameraAnimationNode::SetLookAt (GeVec3d *pEye*, GeVec3d *pCenter*)

Sets the camera position and the camera target.

6.22.3 Property Documentation

6.22.3.1 CameraAnimationNodeFlags X3g::Plugin::CameraAnimationNode::Flags `[get],[set]`

Additional node flags.

6.22.3.2 System:: String^ X3g::Plugin::CameraAnimationNode::Name `[get],[set]`

The name of the node.

6.22.3.3 double X3g::Plugin::CameraAnimationNode::Speed [get],[set]

Sets the speed (m/s).

The speed defines how long it takes to move from this node to the next node of the animation. Must be greater than 0.

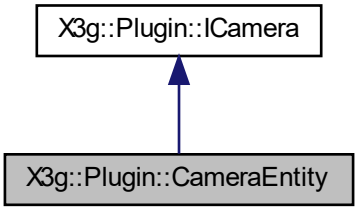
6.22.3.4 double X3g::Plugin::CameraAnimationNode::WaitTime [get],[set]

The wait time defines how long the animation stops/waits at the node's camera position.

6.23 X3g::Plugin::CameraEntity Class Reference

Camera entity.

Inheritance diagram for X3g::Plugin::CameraEntity:



Public Member Functions

- virtual void GetLookAt ([Out] X3g::Plugin::GeVec3d% pEye,[Out] X3g::Plugin::GeVec3d% pCenter,[Out] X3g::Plugin::GeVec3d% pUp)

  Get orientation of the CameraEntity.
- virtual void SetLookAt (X3g::Plugin::GeVec3d pEye, X3g::Plugin::GeVec3d pCenter, X3g::Plugin::GeVec3d pUp)

  Set CameraEntity position and rotation.
- virtual void ZoomToExtents (ReadOnlyCollection< System::String^ >^pEntities, double pMargin, double pAspect)

  Zoom camera in/out to have all the given entities in view.
- virtual void ZoomToExtents (IEnumerable< DbEntity^ >^pEntities, double pMargin, double pAspect)

  Zoom camera in/out to have all the given entities in view.
- virtual void CopyFrom (X3g::Plugin::ICamera^ pOther)

  Copy settings from another camera.

Static Public Member Functions

- static CameraEntity^ FromDbEntity (DbEntity^ pEntity)

  Returns CameraEntity if given DbEntity is a camera.

Properties

- virtual System::String^ Name [get, set]

  The name of the camera.
- virtual X3g::Plugin::CameraProjection Projection [get, set]

  Set or get the projection of the CameraEntity.
- virtual double Fov [get, set]

  Set or get the field of view.
- virtual double YMag [get, set]

  Set or get Y magnification of orthographic camera.
- virtual bool DofEnabled [get, set]

  Enable or disable the depth of field effect.
- virtual double DofDistance [get, set]

  Set or get the depth of field value.
- virtual double DofAperture [get, set]

  Set or get the value of lens aperture / depth of field strength.

6.23.1 Detailed Description

Camera entity.

6.23.2 Member Function Documentation

6.23.2.1 void X3g::Plugin::CameraEntity::CopyFrom (X3g::Plugin::ICamera^ *pOther*) [virtual]

Copy settings from another camera.

Implements X3g::Plugin::ICamera.

6.23.2.2 CameraEntity X3g::Plugin::CameraEntity::FromDbEntity (DbEntity^ *pEntity*) [static]

Returns CameraEntity if given DbEntity is a camera.

Otherwise it returns null.

6.23.2.3 void X3g::Plugin::CameraEntity::GetLookAt ([Out] X3g::Plugin::GeVec3d% *pEye*, [Out] X3g::Plugin::GeVec3d% *pCenter*, [Out] X3g::Plugin::GeVec3d% *pUp*) [virtual]

Get orientation of the CameraEntity.

Implements X3g::Plugin::ICamera.

6.23.2.4 void X3g::Plugin::CameraEntity::SetLookAt (X3g::Plugin::GeVec3d *pEye*, X3g::Plugin::GeVec3d *pCenter*, X3g::Plugin::GeVec3d *pUp*) [virtual]

Set CameraEntity position and rotation.

Parameters

pEye	Position of the camera / eye.
pCenter	The camera / eye will be rotated to have the point pCenter in its center of view.
pUp	Rotate the camera so the pUp vector is aligned with the up-direction of the cameras view.

Implements X3g::Plugin::ICamera.

6.23.2.5 void X3g::Plugin::CameraEntity::ZoomToExtents (ReadOnlyCollection< System::String^ >^ *pEntities*, double *pMargin*, double *pAspect*) [virtual]

Zoom camera in/out to have all the given entities in view.

Parameters

pEntities	Collection of entities to have in view.
pMargin	Margin to keep around the entities.
pAspect	The aspect ratio of the desired view.

6.23.2.6 void X3g::Plugin::CameraEntity::ZoomToExtents (IEnumerable< DbEntity^ >^ *pEntities*, double *pMargin*, double *pAspect*) [virtual]

Zoom camera in/out to have all the given entities in view.

Parameters

pEntities	Enumerable of entities to have in view.
pMargin	Margin to keep around the entities.
pAspect	The aspect ratio of the desired view.

Implements X3g::Plugin::ICamera.

6.23.3 Property Documentation

6.23.3.1 virtual double X3g::Plugin::CameraEntity::DofAperture [get], [set]

Set or get the value of lens aperture / depth of field strength.

6.23.3.2 virtual double X3g::Plugin::CameraEntity::DofDistance [get], [set]

Set or get the depth of field value.

6.23.3.3 virtual bool X3g::Plugin::CameraEntity::DofEnabled [get], [set]

Enable or disable the depth of field effect.

6.23.3.4 virtual double X3g::Plugin::CameraEntity::Fov [get], [set]

Set or get the field of view.

6.23.3.5 virtual System:: String^ X3g::Plugin::CameraEntity::Name [get], [set]

The name of the camera.

6.23.3.6 virtual X3g:: Plugin:: CameraProjection X3g::Plugin::CameraEntity::Projection [get], [set]

Set or get the projection of the CameraEntity.

6.23.3.7 virtual double X3g::Plugin::CameraEntity::YMag [get], [set]

Set or get Y magnification of orthographic camera.

6.24 X3g::Plugin::CameraManager Class Reference

Camera Manager.

Public Member Functions

- IList< CameraEntity^ >^ GetAllCameras ()

  Returns all existing user defined cameras.
- CameraEntity^ GetCameraByName (String^ pName)

  Returns the camera with the given name.
- CameraEntity^ CreateCamera ()

  Creates a new camera.
- void DeleteCamera (CameraEntity^ pCamera)

  Deletes a camera.
- IList< CameraAnimation^ >^ GetAllAnimations ()

  Returns all existing animations.
- CameraAnimation^ GetAnimationByName (String^ pName)

  Returns the animation with the given name.
- CameraAnimation^ CreateAnimation ()

  Creates a new animation.
- void DeleteAnimation (CameraAnimation^ pAnimation)

  Deletes a animation.

6.24.1 Detailed Description

Camera Manager.

6.24.2 Member Function Documentation

6.24.2.1 CameraAnimation X3g::Plugin::CameraManager::CreateAnimation ( )

Creates a new animation.

6.24.2.2 X3g::Plugin::CameraEntity X3g::Plugin::CameraManager::CreateCamera ( )

Creates a new camera.

6.24.2.3 void X3g::Plugin::CameraManager::DeleteAnimation (CameraAnimation^ *pAnimation*)

Deletes a animation.

6.24.2.4 void X3g::Plugin::CameraManager::DeleteCamera (CameraEntity^ *pCamera*)

Deletes a camera.

6.24.2.5 IList< CameraAnimation^ > X3g::Plugin::CameraManager::GetAllAnimations ( )

Returns all existing animations.

6.24.2.6 IList< X3g::Plugin::CameraEntity^ > X3g::Plugin::CameraManager::GetAllCameras ( )

Returns all existing user defined cameras.

6.24.2.7 CameraAnimation X3g::Plugin::CameraManager::GetAnimationByName (String^ *pName*)

Returns the animation with the given name.

6.24.2.8 X3g::Plugin::CameraEntity X3g::Plugin::CameraManager::GetCameraByName (String^ *pName*)

Returns the camera with the given name.

6.25 X3g::Plugin::CannotExplodeEntityException Class Reference

CannotExplodeEntityException Thrown if a database entity could not be exploded.

Inherits Exception.

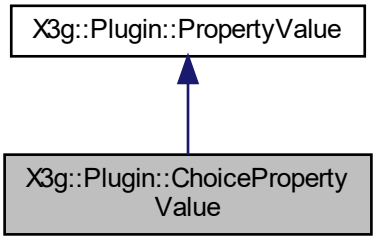
6.25.1 Detailed Description

CannotExplodeEntityException Thrown if a database entity could not be exploded.

6.26 X3g::Plugin::ChoicePropertyValue Class Reference

Choice Property Value.

Inheritance diagram for X3g::Plugin::ChoicePropertyValue:



Public Member Functions

- ChoicePropertyValue (System::String^ pKey)
  Constructor.
- ChoicePropertyValue (System::String^ pKey, System::String^ pText)
  Constructor.
- ChoicePropertyValue (System::String^ pKey, System::String^ pText, System::String^ pSmallImage, System::String^ pLargeImage, ChoiceValueType pValueType)
  Constructor.

Properties

- System::String^ Key [get, set]

  The key of the choice item.
- System::String^ Text [get]

  The text of the choice item.
- System::String^ SmallImage [get]

  Path to a small icon image.
- System::String^ LargeImage [get]

  Path to a large icon image.
- ChoiceValueType ChoiceType [get]

  The type of the choice value.

6.26.1 Detailed Description

Choice Property Value.

6.26.2 Constructor & Destructor Documentation

6.26.2.1 X3g::Plugin::ChoicePropertyValue::ChoicePropertyValue (System::String^ *pKey*)

Constructor.

Parameters

pKey	Set the key of the choice item.

6.26.2.2 X3g::Plugin::ChoicePropertyValue::ChoicePropertyValue (System::String^ *pKey*, System::String^ *pText*)

Constructor.

Parameters

pKey	Set the key of the choice item.
pText	Set the text of the choice item.

6.26.2.3 X3g::Plugin::ChoicePropertyValue::ChoicePropertyValue (System::String^ *pKey*, System::String^ *pText*, System::String^ *pSmallImage*, System::String^ *pLargeImage*, ChoiceValueType *pValueType*)

Constructor.

Parameters

pKey	Set the key of the choice item.
pText	Set the text of the choice item.
pSmallImage	Path to a small icon image.
pLargeImage	Path to a large icon image.
pValueType	The type of the choice value. See ChoiceType.

6.26.3 Property Documentation

6.26.3.1 ChoiceValueType X3g::Plugin::ChoicePropertyValue::ChoiceType [get]

The type of the choice value.

Can be used to generate a formated display value. It has no effect on the internal representation.

6.26.3.2 System:: String^ X3g::Plugin::ChoicePropertyValue::Key [get],[set]

The key of the choice item.

6.26.3.3 System:: String^ X3g::Plugin::ChoicePropertyValue::LargeImage [get]

Path to a large icon image.

6.26.3.4 System:: String^ X3g::Plugin::ChoicePropertyValue::SmallImage [get]

Path to a small icon image.

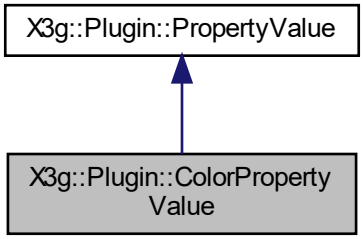
6.26.3.5 System:: String^ X3g::Plugin::ChoicePropertyValue::Text [get]

The text of the choice item.

6.27 X3g::Plugin::ColorPropertyValue Class Reference

Color Property Value.

Inheritance diagram for X3g::Plugin::ColorPropertyValue:



Public Member Functions

- ColorPropertyValue (Color pValue)

  Constructor.

Properties

- Color Value [get, set]

  The color value.

6.27.1 Detailed Description

Color Property Value.

6.27.2 Constructor & Destructor Documentation

6.27.2.1 X3g::Plugin::ColorPropertyValue::ColorPropertyValue (Color *pValue*)

Constructor.

Parameters

pValue	Initial color value.

6.27.3 Property Documentation

6.27.3.1 Color X3g::Plugin::ColorPropertyValue::Value [get],[set]

The color value.

6.28 X3g::Plugin::Currency Class Reference

Currency class.

Public Member Functions

- Currency ()

  Default constructor.
- Currency (String^ pUnit, Decimal pValue)

  Constructor to initialize the unit and the value.
- Currency (Currency^ pOther)

  Copy constructor.
- bool IsRelative ()

  Returns true if the currency represents a relative value.
- void SetRelative ()

  Changes the unit to relative.

Public Attributes

- String^ Unit

  The currency unit.
- Decimal Value

  The currency value.

6.28.1 Detailed Description

Currency class.

6.28.2 Constructor & Destructor Documentation

6.28.2.1 X3g::Plugin::Currency::Currency ( )

Default constructor.

Unit = null, Value = 0

6.28.2.2 X3g::Plugin::Currency::Currency (String^ *pUnit,* Decimal *pValue*)

Constructor to initialize the unit and the value.

6.28.2.3 X3g::Plugin::Currency::Currency (Currency^ *pOther*)

Copy constructor.

6.28.3 Member Function Documentation

6.28.3.1 bool X3g::Plugin::Currency::IsRelative ( )

Returns true if the currency represents a relative value.

(Unit = "%")

6.28.3.2 void X3g::Plugin::Currency::SetRelative ( )

Changes the unit to relative.

(Unit = "%")

6.28.4 Member Data Documentation

6.28.4.1 String ^ X3g::Plugin::Currency::Unit

The currency unit.

Contains a 3 letter ISO currency code. If the Unit is equal to "%" then this currency represents a relative value. (e.g. for discounts)

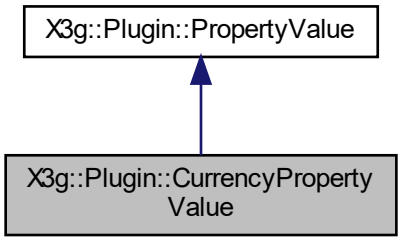
6.28.4.2 Decimal X3g::Plugin::Currency::Value

The currency value.

6.29 X3g::Plugin::CurrencyPropertyValue Class Reference

Currency Property Value.

Inheritance diagram for X3g::Plugin::CurrencyPropertyValue:



Public Member Functions

- CurrencyPropertyValue (Currency^ pValue)

  Constructor.

Properties

- Currency^ Value `[get, set]`

  The currency value.

6.29.1 Detailed Description

Currency Property Value.

6.29.2 Constructor & Destructor Documentation

6.29.2.1 X3g::Plugin::CurrencyPropertyValue::CurrencyPropertyValue (Currency^ *pValue*)

Constructor.

Parameters

pValue	Initial Currency value.

6.29.3 Property Documentation

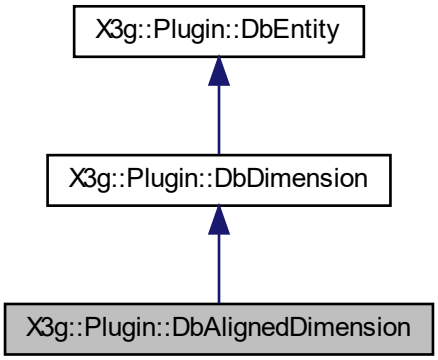
6.29.3.1 Currency^ X3g::Plugin::CurrencyPropertyValue::Value `[get],[set]`

The currency value.

6.30 X3g::Plugin::DbAlignedDimension Class Reference

A length dimension with a dimension line parallel to measuring points.

Inheritance diagram for X3g::Plugin::DbAlignedDimension:



Public Member Functions

- DbAlignedDimension (Layout::Page^ pPage)

  Create a DbAlignedDimension on the given page.
- DbAlignedDimension (Block^ pBlock)

  Create a DbAlignedDimension as child of the given block.
- IList< DbEntity^ >^ GetXLine1Refs ()

  Returns a list of references for the first measuring point.
- void SetXLine1Refs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the first measuring point.
- IList< DbEntity^ >^ GetXLine2Refs ()

  Returns a list of references for the second measuring point.
- void SetXLine2Refs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the second measuring point.

Properties

- GeVec3d XLine1Point `[get, set]`

  First measuring point and start point of the first extension line.
- GeVec3d XLine2Point `[get, set]`

  Second measuring point and start point of the second extension line.
- GeVec3d DimensionLinePoint `[get, set]`

  A point of the dimension line.
- double BaseOffset `[get, set]`

  Specifies the distance of reference points to extension lines.
- double ExtensionLineLength `[get, set]`

  Specifies the length of extension line beyond dimension line.
- double ScaleFactor `[get, set]`

  Specifies the distance multiplier for measurement.
- LengthUnit Unit `[get, set]`

  Specifies the unit for measurement.

Additional Inherited Members

6.30.1 Detailed Description

A length dimension with a dimension line parallel to measuring points.

6.30.2 Constructor & Destructor Documentation

6.30.2.1 X3g::Plugin::DbAlignedDimension::DbAlignedDimension (Layout::Page^ *pPage*)

Create a DbAlignedDimension on the given page.

6.30.2.2 X3g::Plugin::DbAlignedDimension::DbAlignedDimension (Block^ *pBlock*)

Create a DbAlignedDimension as child of the given block.

6.30.3 Member Function Documentation

6.30.3.1 IList< DbEntity^ > X3g::Plugin::DbAlignedDimension::GetXLine1Refs ( )

Returns a list of references for the first measuring point.

6.30.3.2 IList< DbEntity^ > X3g::Plugin::DbAlignedDimension::GetXLine2Refs ( )

Returns a list of references for the second measuring point.

6.30.3.3 void X3g::Plugin::DbAlignedDimension::SetXLine1Refs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the first measuring point.

6.30.3.4 void X3g::Plugin::DbAlignedDimension::SetXLine2Refs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the second measuring point.

6.30.4 Property Documentation

6.30.4.1 double X3g::Plugin::DbAlignedDimension::BaseOffset [get],[set]

Specifies the distance of reference points to extension lines.

6.30.4.2 GeVec3d X3g::Plugin::DbAlignedDimension::DimensionLinePoint [get],[set]

A point of the dimension line.

6.30.4.3 double X3g::Plugin::DbAlignedDimension::ExtensionLineLength [get],[set]

Specifies the length of extension line beyond dimension line.

6.30.4.4 double X3g::Plugin::DbAlignedDimension::ScaleFactor [get], [set]

Specifies the distance multiplier for measurement.

6.30.4.5 LengthUnit X3g::Plugin::DbAlignedDimension::Unit [get], [set]

Specifies the unit for measurement.

If set to 'Undefined' the scale factor will be used.

6.30.4.6 GeVec3d X3g::Plugin::DbAlignedDimension::XLine1Point [get], [set]

First measuring point and start point of the first extension line.

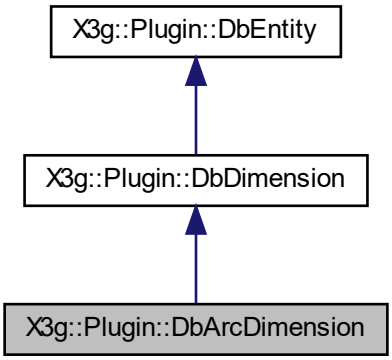
6.30.4.7 GeVec3d X3g::Plugin::DbAlignedDimension::XLine2Point [get], [set]

Second measuring point and start point of the second extension line.

6.31 X3g::Plugin::DbArcDimension Class Reference

A length dimension to measure an arc.

Inheritance diagram for X3g::Plugin::DbArcDimension:



Public Member Functions

- DbArcDimension (Layout::Page^ pPage)

  Create a DbArcDimension on the given page.
- DbArcDimension (Block^ pOwner)

  Create a DbArcDimension as child of the given block.
- IList< DbEntity^ >^ GetXLine1Refs ()

  Returns a list of references for the first measuring point.
- void SetXLine1Refs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the first measuring point.

- IList< DbEntity^ >^ GetXLine2Refs ()

  Returns a list of references for the second measuring point.
- void SetXLine2Refs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the second extension line.

Properties

- GeVec3d XLine1Point `[get, set]`

  First measuring point and start point of the first extension line.
- GeVec3d CenterPoint `[get, set]`

  The center point of the arc being dimensioned.
- GeVec3d XLine2Point `[get, set]`

  Second measuring point and start point of the second extension line.
- GeVec3d DimensionLinePoint `[get, set]`

  A point of the dimension arc.
- double BaseOffset `[get, set]`

  Specifies the distance of reference points to extension lines.
- double ExtensionLineLength `[get, set]`

  Specifies the length of extension line beyond dimension line.
- double ScaleFactor `[get, set]`

  Specifies the distance multiplier for measurement.
- LengthUnit Unit `[get, set]`

  Specifies the unit for measurement.

Additional Inherited Members

6.31.1 Detailed Description

A length dimension to measure an arc.

6.31.2 Constructor & Destructor Documentation

6.31.2.1 X3g::Plugin::DbArcDimension::DbArcDimension (Layout::Page^ *pPage*)

Create a DbArcDimension on the given page.

6.31.2.2 X3g::Plugin::DbArcDimension::DbArcDimension (Block^ *pOwner*)

Create a DbArcDimension as child of the given block.

6.31.3 Member Function Documentation

6.31.3.1 IList< DbEntity^ > X3g::Plugin::DbArcDimension::GetXLine1Refs ( )

Returns a list of references for the first measuring point.

6.31.3.2 IList< DbEntity^ > X3g::Plugin::DbArcDimension::GetXLine2Refs ( )

Returns a list of references for the second measuring point.

6.31.3.3 void X3g::Plugin::DbArcDimension::SetXLine1Refs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the first measuring point.

6.31.3.4 void X3g::Plugin::DbArcDimension::SetXLine2Refs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the second extension line.

6.31.4 Property Documentation

6.31.4.1 double X3g::Plugin::DbArcDimension::BaseOffset [get],[set]

Specifies the distance of reference points to extension lines.

6.31.4.2 GeVec3d X3g::Plugin::DbArcDimension::CenterPoint [get],[set]

The center point of the arc being dimensioned.

6.31.4.3 GeVec3d X3g::Plugin::DbArcDimension::DimensionLinePoint [get],[set]

A point of the dimension arc.

6.31.4.4 double X3g::Plugin::DbArcDimension::ExtensionLineLength [get],[set]

Specifies the length of extension line beyond dimension line.

6.31.4.5 double X3g::Plugin::DbArcDimension::ScaleFactor [get],[set]

Specifies the distance multiplier for measurement.

6.31.4.6 LengthUnit X3g::Plugin::DbArcDimension::Unit [get],[set]

Specifies the unit for measurement.
If set to 'Undefined' the scale factor will be used.

6.31.4.7 GeVec3d X3g::Plugin::DbArcDimension::XLine1Point [get],[set]

First measuring point and start point of the first extension line.

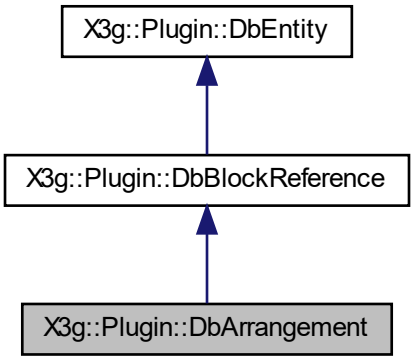
6.31.4.8 GeVec3d X3g::Plugin::DbArcDimension::XLine2Point [get],[set]

Second measuring point and start point of the second extension line.

6.32 X3g::Plugin::DbArrangement Class Reference

An arrangement automatic places items with the help of a base geometry.

Inheritance diagram for X3g::Plugin::DbArrangement:



Public Member Functions

- DbArrangement (DbCurve^ pBaseGeometry, DbEntity^ pItemToArrange)

  Creates an arrangement out of a geometry and an item to arrange.
- void StartEdit ()

  Saves the current state of this arrangement.
- void CancelEdit ()

  Cancels all changes since StartEdit was called.
- void EndEdit ()

  End edit mode and creates an undo step for all changes.
- void ResetAvailableItems ()

  Reset position and rotation of all available items.
- void RefillDeletedItems ()

  Create new items for all deleted items.

Properties

- ArrangementType Type `[get, set]`

  Basic type of this arrangement.
- bool IsAreaTypeAllowed `[get]`

  Returns true if DbArrangement::Type can be set to ArrangementType::Area.
- ArrangementVPlacement VerticalPlacement `[get, set]`

  Vertical item placement of this arrangement.
- bool AlignToBase `[get, set]`

  If true items will be rotated with the help of the base geometry.
- double ItemsXRotation `[get, set]`

  Additional rotation of all items around x-axis.
- double ItemsYRotation `[get, set]`

  Additional rotation of all items around y-axis.
- double ItemsZRotation `[get, set]`

  Additional rotation of all items around z-axis.

- ArrangementDistributionType DistributionType [get, set]

  Item distribution of this arrangement.
- double ColumnOffset [get, set]

  Offset from base point to first column.
- double RowOffset [get, set]

  Offset from base point to first row.
- double ColumnGap [get, set]

  Gap between two columns.
- double RowGap [get, set]

  Gap between two rows.
- int ColumnCount [get, set]

  Number of columns.
- int RowCount [get, set]

  Number of rows.
- bool ItemsCompleteInside [get, set]

  If true only items that are complete inside the base geometry will be displayed.

Additional Inherited Members

6.32.1 Detailed Description

An arrangement automatic places items with the help of a base geometry.

6.32.2 Constructor & Destructor Documentation

6.32.2.1 X3g::Plugin::DbArrangement::DbArrangement (DbCurve^ *pBaseGeometry,* DbEntity^ *pItemToArrange*)

Creates an arrangement out of a geometry and an item to arrange.

Both parameters must not be null.

6.32.3 Member Function Documentation

6.32.3.1 void X3g::Plugin::DbArrangement::CancelEdit ( )

Cancels all changes since StartEdit was called.

6.32.3.2 void X3g::Plugin::DbArrangement::EndEdit ( )

End edit mode and creates an undo step for all changes.

6.32.3.3 void X3g::Plugin::DbArrangement::RefillDeletedItems ( )

Create new items for all deleted items.

6.32.3.4 void X3g::Plugin::DbArrangement::ResetAvailableItems ( )

Reset position and rotation of all available items.

6.32.3.5 void X3g::Plugin::DbArrangement::StartEdit ( )

Saves the current state of this arrangement.

Everytime this function is call DbArrangement::CancelEdit() or DbArrangement::EndEdit() should be called.

6.32.4 Property Documentation

6.32.4.1 bool X3g::Plugin::DbArrangement::AlignToBase [get], [set]

If true items will be rotated with the help of the base geometry.

6.32.4.2 int X3g::Plugin::DbArrangement::ColumnCount [get], [set]

Number of columns.

This property will only be used if DbArrangement::DistributionType is not set to ArrangementDistributionType::-Manual.

6.32.4.3 double X3g::Plugin::DbArrangement::ColumnGap [get], [set]

Gap between two columns.

This property will only be used if DbArrangement::DistributionType is set to ArrangementDistributionType::Manual.

6.32.4.4 double X3g::Plugin::DbArrangement::ColumnOffset [get], [set]

Offset from base point to first column.

This property will only be used if DbArrangement::DistributionType is set to ArrangementDistributionType::Manual.

6.32.4.5 ArrangementDistributionType X3g::Plugin::DbArrangement::DistributionType [get], [set]

Item distribution of this arrangement.

6.32.4.6 bool X3g::Plugin::DbArrangement::IsAreaTypeAllowed [get]

Returns true if DbArrangement::Type can be set to ArrangementType::Area.

6.32.4.7 bool X3g::Plugin::DbArrangement::ItemsCompleteInside [get], [set]

If true only items that are complete inside the base geometry will be displayed.

This property will only be used if DbArrangement::Type is set to ArrangementType::Area.

6.32.4.8 double X3g::Plugin::DbArrangement::ItemsXRotation [get], [set]

Additional rotation of all items around x-axis.

6.32.4.9 double X3g::Plugin::DbArrangement::ItemsYRotation [get], [set]

Additional rotation of all items around y-axis.

6.32.4.10 double X3g::Plugin::DbArrangement::ItemsZRotation [get], [set]

Additional rotation of all items around z-axis.

6.32.4.11 int X3g::Plugin::DbArrangement::RowCount [get], [set]

Number of rows.

This property will only be used if DbArrangement::Type is set to ArrangementType::Area and DbArrangement::-DistributionType is not set to ArrangementDistributionType::Manual.

6.32.4.12 double X3g::Plugin::DbArrangement::RowGap [get], [set]

Gap between two rows.

This property will only be used if DbArrangement::Type is set to ArrangementType::Area and DbArrangement::-DistributionType is set to ArrangementDistributionType::Manual.

6.32.4.13 double X3g::Plugin::DbArrangement::RowOffset [get], [set]

Offset from base point to first row.

This property will only be used if DbArrangement::Type is set to ArrangementType::Area and DbArrangement::-DistributionType is set to ArrangementDistributionType::Manual.

6.32.4.14 ArrangementType X3g::Plugin::DbArrangement::Type [get], [set]

Basic type of this arrangement.

The type can only be set to ArrangementType::Area if the area type is allowed. Otherwise this property always returns ArrangementType::AlongPath.

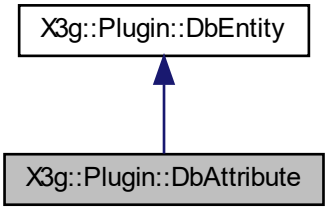
6.32.4.15 ArrangementVPlacement X3g::Plugin::DbArrangement::VerticalPlacement [get], [set]

Vertical item placement of this arrangement.

6.33 X3g::Plugin::DbAttribute Class Reference

Attributes are tags, or labels, for blocks.

Inheritance diagram for X3g::Plugin::DbAttribute:



Public Member Functions

- DbAttribute (DbBlockReference^ pOwner, DbAttributeDefinition^ pDef)

  Creates an attribute from an attribute definition.

Properties

- String^ Tag `[get, set]`

  Tag used to identify the attribute.
- String^ Text `[get, set]`

  Displayed text of this attribute.
- bool Visible `[get, set]`

  Visibility.

Additional Inherited Members

6.33.1 Detailed Description

Attributes are tags, or labels, for blocks.

Attributes are attached to block references which reference a block which contains an attribute definition with the same tag as the attribute. A DbAttribute cannot be owned by a block.

6.33.2 Constructor & Destructor Documentation

6.33.2.1 X3g::Plugin::DbAttribute::DbAttribute (DbBlockReference^ *pOwner,* DbAttributeDefinition^ *pDef*)

Creates an attribute from an attribute definition.

Tag, Text and Visibility are taken from the definition. This attribute may be added to a block reference for which the according block has the given attribute definition.

6.33.3 Property Documentation

6.33.3.1 String^ X3g::Plugin::DbAttribute::Tag `[get],[set]`

Tag used to identify the attribute.

6.33.3.2 String^ X3g::Plugin::DbAttribute::Text `[get],[set]`

Displayed text of this attribute.

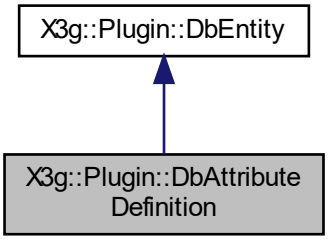
6.33.3.3 bool X3g::Plugin::DbAttribute::Visible `[get],[set]`

Visibility.

6.34 X3g::Plugin::DbAttributeDefinition Class Reference

Attribute definitions are used to define tags, or labels, for blocks.

Inheritance diagram for X3g::Plugin::DbAttributeDefinition:



Public Member Functions

- DbAttributeDefinition (Block^ pOwner)

  Create a DbAttributeDefinition as child of a given block.

Properties

- String^ Tag `[get, set]`

  Tag used to identify the attribute definition.
- String^ DefaultText `[get, set]`

  The default text.
- bool Visible `[get, set]`

  Visibility.

Additional Inherited Members

6.34.1 Detailed Description

Attribute definitions are used to define tags, or labels, for blocks.

They are owned by blocks and can be referenced by DbAttribute objects, which are owned by DbBlockReference objects.

6.34.2 Constructor & Destructor Documentation

6.34.2.1 X3g::Plugin::DbAttributeDefinition::DbAttributeDefinition (Block^ *pOwner*)

Create a DbAttributeDefinition as child of a given block.

6.34.3 Property Documentation

6.34.3.1 String^ X3g::Plugin::DbAttributeDefinition::DefaultText `[get],[set]`

The default text.

6.34.3.2 String^ X3g::Plugin::DbAttributeDefinition::Tag [get], [set]

Tag used to identify the attribute definition.

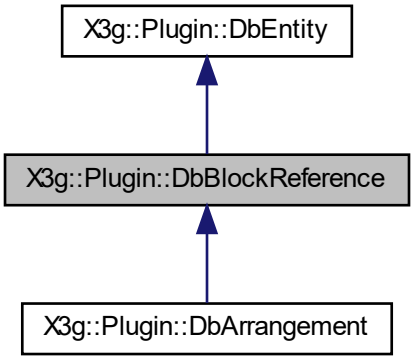
6.34.3.3 bool X3g::Plugin::DbAttributeDefinition::Visible [get], [set]

Visibility.

6.35 X3g::Plugin::DbBlockReference Class Reference

A block reference is used to place, size, and display an instance of the collection of entities within a DbBlock that it references.

Inheritance diagram for X3g::Plugin::DbBlockReference:



Public Member Functions

- DbBlockReference (Block^ pOwner, String^ pBlock)

  Creates a block reference to the given block.
- DbBlockReference (Block^ pOwner, Block^ pBlock)

  Creates a block reference to the given block.
- DbBlockReference (Block^ pOwner, Block^ pBlock, String^ pCustomType, Object^ pCustomObject)

  Creates a block reference to the given block.
- GeMatrix^ GetTransform ()

  Returns transformation of the block reference.
- void SetTransform (GeMatrix^ pTransform)

  Sets transformation of the block reference.
- array< String^ >^ GetAttributes ()

  Returns a list of all attributes.
- DbBlockReference^ CloneLinked ()

  Clones the block reference and links it with the copy.
- bool OpenBlock ()

  Opens the referenced block to work with child entities (e.g.
- void CloseBlock ()

  Close the referenced block and all child entities.

Properties

- String^ Block [get, set]

  The referenced block, identified by the block name.
- GeVec3d Position [get, set]

  Position of this block reference.
- GeQuat Rotation [get, set]

  Rotation of this block reference.
- GeVec3d Scale [get, set]

  Scale of this block reference.
- bool IsBlockOpen [get]

  Returns true if the referenced block is open for edit.

Events

- System::EventHandler^ IsBlockOpenChanged

  Occurs when DbBlockReference::IsBlockOpen changed.

Additional Inherited Members

6.35.1 Detailed Description

A block reference is used to place, size, and display an instance of the collection of entities within a DbBlock that it references.

In addition, block references can be the owner of DbAttribute entities.

6.35.2 Constructor & Destructor Documentation

6.35.2.1 X3g::Plugin::DbBlockReference::DbBlockReference (Block^ *pOwner,* String^ *pBlock*)

Creates a block reference to the given block.

6.35.2.2 X3g::Plugin::DbBlockReference::DbBlockReference (Block^ *pOwner,* Block^ *pBlock*)

Creates a block reference to the given block.

6.35.2.3 X3g::Plugin::DbBlockReference::DbBlockReference (Block^ *pOwner,* Block^ *pBlock,* String^ *pCustomType,* Object^ *pCustomObject*)

Creates a block reference to the given block.

Allows to set custom type and custom object on construction.

6.35.3 Member Function Documentation

6.35.3.1 DbBlockReference X3g::Plugin::DbBlockReference::CloneLinked ( )

Clones the block reference and links it with the copy.

When user changes block content of one reference the linked references will also be changed. If block references are not linked the blocks would be made unique in this case. Linked block references are only supported in modelspace.

Exceptions

System::NotSupported-Exception	Block reference not in modelspace.

6.35.3.2 void X3g::Plugin::DbBlockReference::CloseBlock ( )

Close the referenced block and all child entities.

6.35.3.3 array< String^ > X3g::Plugin::DbBlockReference::GetAttributes ( )

Returns a list of all attributes.

Use DbEntity::FromId() to retrieve the DbAttribute behind it's id.

6.35.3.4 GeMatrix X3g::Plugin::DbBlockReference::GetTransform ( )

Returns transformation of the block reference.

6.35.3.5 bool X3g::Plugin::DbBlockReference::OpenBlock ( )

Opens the referenced block to work with child entities (e.g.

select, edit). Returns true on success.

The parent block of this entity must be the modelspace or already open. Other DbBlockReferences at the same level will be closed. Attention: The block of this reference maybe cloned on open to avoid editing multiple references at one.

6.35.3.6 void X3g::Plugin::DbBlockReference::SetTransform (GeMatrix^ *pTransform*)

Sets transformation of the block reference.

6.35.4 Property Documentation

6.35.4.1 String^ X3g::Plugin::DbBlockReference::Block [get], [set]

The referenced block, identified by the block name.

Use IBlockManager::GetBlock() to retrieve the block interface.

6.35.4.2 bool X3g::Plugin::DbBlockReference::IsBlockOpen [get]

Returns true if the referenced block is open for edit.

6.35.4.3 GeVec3d X3g::Plugin::DbBlockReference::Position [get], [set]

Position of this block reference.

6.35.4.4 GeQuat X3g::Plugin::DbBlockReference::Rotation [get], [set]

Rotation of this block reference.

6.35.4.5 GeVec3d X3g::Plugin::DbBlockReference::Scale [get],[set]

Scale of this block reference.

6.35.5 Event Documentation

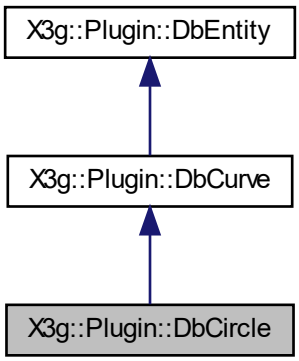
6.35.5.1 System:: EventHandler^ X3g::Plugin::DbBlockReference::IsBlockOpenChanged

Occurs when DbBlockReference::IsBlockOpen changed.

6.36 X3g::Plugin::DbCircle Class Reference

A simple closed circle.

Inheritance diagram for X3g::Plugin::DbCircle:



Public Member Functions

- DbCircle (Layout::Page^ pPage)

  Create a DbCircle on the given Page.
- DbCircle (Block^ pOwner)

  Create a DbCircle as child of the given Block.

Properties

- GeVec3d Center [get, set]

  Center point of this circle.
- double Radius [get, set]

  Radius of this circle.
- GeVec3d Normal [get, set]

  Normal of this circle.

Additional Inherited Members

6.36.1 Detailed Description

A simple closed circle.

6.36.2 Constructor & Destructor Documentation

6.36.2.1 X3g::Plugin::DbCircle::DbCircle (Layout::Page^ *pPage*)

Create a DbCircle on the given Page.

6.36.2.2 X3g::Plugin::DbCircle::DbCircle (Block^ *pOwner*)

Create a DbCircle as child of the given Block.

6.36.3 Property Documentation

6.36.3.1 GeVec3d X3g::Plugin::DbCircle::Center [get],[set]

Center point of this circle.

6.36.3.2 GeVec3d X3g::Plugin::DbCircle::Normal [get],[set]

Normal of this circle.

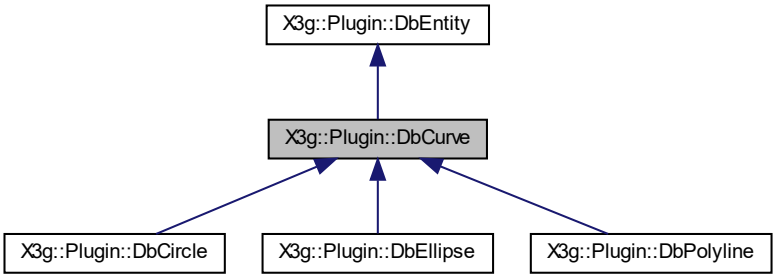
6.36.3.3 double X3g::Plugin::DbCircle::Radius [get],[set]

Radius of this circle.

6.37 X3g::Plugin::DbCurve Class Reference

Base class for all curves.

Inheritance diagram for X3g::Plugin::DbCurve:



Public Member Functions

- GeVec3d GetPointAtDistance (double pDistance)

  Returns the point on this DbCurve corresponding to the given distance.

Properties

- bool IsClosed [get]

  Returns true if the DbCurve is closed.
- bool IsPlanar [get]

  Returns true if the DbCurve is planar.
- double Length [get]

  Returns the length of this DbCurve.

Additional Inherited Members

6.37.1 Detailed Description

Base class for all curves.

6.37.2 Member Function Documentation

6.37.2.1 GeVec3d X3g::Plugin::DbCurve::GetPointAtDistance (double *pDistance*)

Returns the point on this DbCurve corresponding to the given distance.

6.37.3 Property Documentation

6.37.3.1 bool X3g::Plugin::DbCurve::IsClosed [get]

Returns true if the DbCurve is closed.

6.37.3.2 bool X3g::Plugin::DbCurve::IsPlanar [get]

Returns true if the DbCurve is planar.

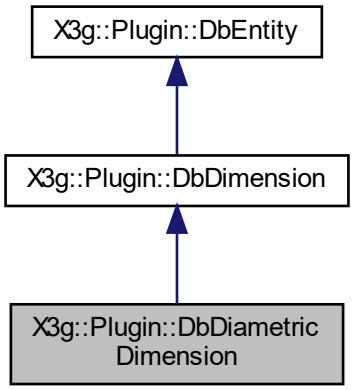
6.37.3.3 double X3g::Plugin::DbCurve::Length [get]

Returns the length of this DbCurve.

6.38 X3g::Plugin::DbDiametricDimension Class Reference

A length dimension used to measure a diameter.

Inheritance diagram for X3g::Plugin::DbDiametricDimension:



Public Member Functions

- DbDiametricDimension (Layout::Page^ pPage)

  Create a DbDiametricDimension on the given page.
- DbDiametricDimension (Block^ pOwner)

  Create a DbDiametricDimension as child of the given block.
- IList< DbEntity^ >^ GetFirstChordRefs ()

  Returns a list of references for the first chord point.
- void SetFirstChordRefs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the first chord point.
- IList< DbEntity^ >^ GetSecondChordRefs ()

  Returns a list of references for the second chord point.
- void SetSecondChordRefs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the second chord point.

Properties

- GeVec3d FirstChordPoint `[get, set]`

  The first chord point of the arc being dimensioned.
- GeVec3d SecondChordPoint `[get, set]`

  The second chord point of the arc being dimensioned.
- double ScaleFactor `[get, set]`

  Specifies the distance multiplier for measurement.
- LengthUnit Unit `[get, set]`

  Specifies the unit for measurement.

Additional Inherited Members

6.38.1 Detailed Description

A length dimension used to measure a diameter.

6.38.2 Constructor & Destructor Documentation

6.38.2.1 X3g::Plugin::DbDiametricDimension::DbDiametricDimension (Layout::Page^ *pPage*)

Create a DbDiametricDimension on the given page.

6.38.2.2 X3g::Plugin::DbDiametricDimension::DbDiametricDimension (Block^ *pOwner*)

Create a DbDiametricDimension as child of the given block.

6.38.3 Member Function Documentation

6.38.3.1 IList< DbEntity^ > X3g::Plugin::DbDiametricDimension::GetFirstChordRefs ( )

Returns a list of references for the first chord point.

6.38.3.2 IList< DbEntity^ > X3g::Plugin::DbDiametricDimension::GetSecondChordRefs ( )

Returns a list of references for the second chord point.

6.38.3.3 void X3g::Plugin::DbDiametricDimension::SetFirstChordRefs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the first chord point.

6.38.3.4 void X3g::Plugin::DbDiametricDimension::SetSecondChordRefs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the second chord point.

6.38.4 Property Documentation

6.38.4.1 GeVec3d X3g::Plugin::DbDiametricDimension::FirstChordPoint [get],[set]

The first chord point of the arc being dimensioned.

6.38.4.2 double X3g::Plugin::DbDiametricDimension::ScaleFactor [get],[set]

Specifies the distance multiplier for measurement.

6.38.4.3 GeVec3d X3g::Plugin::DbDiametricDimension::SecondChordPoint [get],[set]

The second chord point of the arc being dimensioned.

6.38.4.4 LengthUnit X3g::Plugin::DbDiametricDimension::Unit [get],[set]

Specifies the unit for measurement.

If set to 'Undefined' the scale factor will be used.

6.39 X3g::Plugin::DbDictionary Class Reference

A dictionary stores arbitrary data as key value pairs.

Public Types

- enum ValueType {
  ValueType::Undefined, ValueType::Boolean, ValueType::Integer, ValueType::Double, ValueType::String, ValueType::Vec2d, ValueType::Vec3d, ValueType::ByteArray, ValueType::Dictionary, ValueType::EntityRef }

  Determines the type of a value in a dictionary.

Public Member Functions

- bool Has (System::String^ pKey)

  Returns if the dictionary contains a value with the given key.
- void Delete (System::String^ pKey)

  Deletes entry with given key from dictionary.
- void Clear ()

  Deletes all entries from dictionary.
- bool Get (System::String^ pKey,[Out] System::String^ %pValue)

  Get string with given key.
- void Set (System::String^ pKey, System::String^ pValue)

  Set string with given key.
- bool Get (System::String^ pKey,[Out] array< unsigned char >^%pData)

  Get binary data with given key.
- void Set (System::String^ pKey, array< unsigned char >^pData)

  Set binary data with given key.
- bool Get (System::String^ pKey,[Out] bool% pValue)

  Get boolean value with given key.
- void Set (System::String^ pKey, bool pValue)

  Set boolean value with given key.
- bool Get (System::String^ pKey,[Out] int% pValue)

  Get integer value with given key.
- void Set (System::String^ pKey, int pValue)

  Set integer value with given key.
- bool Get (System::String^ pKey,[Out] System::Drawing::Color% pValue)

  Get color with given key.
- void Set (System::String^ pKey, System::Drawing::Color pValue)

  Set color with given key.
- bool Get (System::String^ pKey,[Out] double% pValue)

  Get double value with given key.
- void Set (System::String^ pKey, double pValue)

  Set double value with given key.
- bool Get (System::String^ pKey,[Out] GeVec2d% pValue)

  Get 2d vector with given key.
- void Set (System::String^ pKey, GeVec2d pValue)

  Set 2d vector with given key.
- bool Get (System::String^ pKey,[Out] GeVec3d% pValue)

  Get 3d vector with given key.
- void Set (System::String^ pKey, GeVec3d pValue)

Set 3d vector with given key.

- bool Get (System::String^ pKey,[Out] DbEntity^ %pEntityRef)

  Returns entity reference with given key.
- void Set (System::String^ pKey, DbEntity^ pEntityRef)

  Sets entity reference with given key.
- bool Get (System::String^ pKey,[Out] IList< DbEntity^ >^%pEntityRefs)

  Returns an array of entity references with given key.
- void Set (System::String^ pKey, IEnumerable< DbEntity^ >^pEntityRefs)

  Sets a collection of entity references with given key.
- DbDictionary^ GetDictionary (System::String^ pKey, bool pCreate)

  Returns subsidiary dictionary with given key.
- IList< System::String^ >^ GetAllKeys ()

  Returns keys of all entries in this dictionary.
- ValueType GetValueType (System::String^ pKey)

  Returns the value type of an entry.
- void CopyFrom (DbDictionary^ pOther, bool pOverwrite)

  Copies all entries from other dictionary to this dictionary.

6.39.1 Detailed Description

A dictionary stores arbitrary data as key value pairs.

Keys are unique within a dictionary.

6.39.2 Member Enumeration Documentation

6.39.2.1 enum X3g::Plugin::DbDictionary::ValueType [strong]

Determines the type of a value in a dictionary.

Enumerator

Undefined value type is undefined

Boolean value is of type boolean

Integer value is a 32bit integer

Double value is of type double

String value is of type string

Vec2d value is of type GeVec2d

Vec3d value is of type GeVec3d

ByteArray value is a byte array

Dictionary value is a subsidiary dictionary

EntityRef value is of type DbEntity

6.39.3 Member Function Documentation

6.39.3.1 void X3g::Plugin::DbDictionary::Clear ( )

Deletes all entries from dictionary.

6.39.3.2 void X3g::Plugin::DbDictionary::CopyFrom (DbDictionary^ *pOther,* bool *pOverwrite*)

Copies all entries from other dictionary to this dictionary.

pOverwrite determines if existing values should be overwritten.

Note that CopyFrom doesn't delete any entries. So if you want to replace the whole content of a dictionary you may have to clear it first. Entries from subsidiary dictionaries are also copied by the same rules.

6.39.3.3 void X3g::Plugin::DbDictionary::Delete (System::String^ *pKey*)

Deletes entry with given key from dictionary.

6.39.3.4 bool X3g::Plugin::DbDictionary::Get (System::String^ *pKey,* [Out] System::String^ % *pValue*)

Get string with given key.

Returns false if the value doesn't exist.

6.39.3.5 bool X3g::Plugin::DbDictionary::Get (System::String^ *pKey,* [Out] array< unsigned char >^% *pData*)

Get binary data with given key.

Returns false if the value doesn't exist.

6.39.3.6 bool X3g::Plugin::DbDictionary::Get (System::String^ *pKey,* [Out] bool% *pValue*)

Get boolean value with given key.

Returns false if the value doesn't exist.

6.39.3.7 bool X3g::Plugin::DbDictionary::Get (System::String^ *pKey,* [Out] int% *pValue*)

Get integer value with given key.

Returns false if the value doesn't exist.

6.39.3.8 bool X3g::Plugin::DbDictionary::Get (System::String^ *pKey,* [Out] System::Drawing::Color% *pValue*)

Get color with given key.

Returns false if the value doesn't exist.

6.39.3.9 bool X3g::Plugin::DbDictionary::Get (System::String^ *pKey,* [Out] double% *pValue*)

Get double value with given key.

Returns false if the value doesn't exist.

6.39.3.10 bool X3g::Plugin::DbDictionary::Get (System::String^ *pKey,* [Out] GeVec2d% *pValue*)

Get 2d vector with given key.

Returns false if the value doesn't exist.

6.39.3.11 bool X3g::Plugin::DbDictionary::Get (System::String^ *pKey,* [Out] GeVec3d% *pValue*)

Get 3d vector with given key.

Returns false if the value doesn't exist.

6.39.3.12 bool X3g::Plugin::DbDictionary::Get (System::String^ *pKey,* [Out] DbEntity^ % *pEntityRef*)

Returns entity reference with given key.

Null if entity no longer exists.

6.39.3.13 bool X3g::Plugin::DbDictionary::Get (System::String^ *pKey,* [Out] IList< DbEntity^ >^% *pEntityRefs*)

Returns an array of entity references with given key.

6.39.3.14 IList< System::String^ > X3g::Plugin::DbDictionary::GetAllKeys ( )

Returns keys of all entries in this dictionary.

6.39.3.15 DbDictionary X3g::Plugin::DbDictionary::GetDictionary (System::String^ *pKey,* bool *pCreate*)

Returns subsidiary dictionary with given key.

Allows to create dictionary if it doesn't exist.

6.39.3.16 DbDictionary::ValueType X3g::Plugin::DbDictionary::GetValueType (System::String^ *pKey*)

Returns the value type of an entry.

Returns ValueType::Undefined if value doesn't exist.

6.39.3.17 bool X3g::Plugin::DbDictionary::Has (System::String^ *pKey*)

Returns if the dictionary contains a value with the given key.

6.39.3.18 void X3g::Plugin::DbDictionary::Set (System::String^ *pKey,* System::String^ *pValue*)

Set string with given key.

Overwrites existing values.

6.39.3.19 void X3g::Plugin::DbDictionary::Set (System::String^ *pKey,* array< unsigned char >^ *pData*)

Set binary data with given key.

Overwrites existing value.

6.39.3.20 void X3g::Plugin::DbDictionary::Set (System::String^ *pKey,* bool *pValue*)

Set boolean value with given key.

Overwrites existing value.

6.39.3.21 void X3g::Plugin::DbDictionary::Set (System::String^ *pKey*, int *pValue*)

Set integer value with given key.

Overwrites existing value.

6.39.3.22 void X3g::Plugin::DbDictionary::Set (System::String^ *pKey*, System::Drawing::Color *pValue*)

Set color with given key.

Color is stored as 32bit integer. Overwrites existing value.

6.39.3.23 void X3g::Plugin::DbDictionary::Set (System::String^ *pKey*, double *pValue*)

Set double value with given key.

Overwrites existing value.

6.39.3.24 void X3g::Plugin::DbDictionary::Set (System::String^ *pKey*, GeVec2d *pValue*)

Set 2d vector with given key.

Overwrites existing value.

6.39.3.25 void X3g::Plugin::DbDictionary::Set (System::String^ *pKey*, GeVec3d *pValue*)

Set 3d vector with given key.

Overwrites existing value.

6.39.3.26 void X3g::Plugin::DbDictionary::Set (System::String^ *pKey*, DbEntity^ *pEntityRef*)

Sets entity reference with given key.

Overwrites existing reference.

The dictionary doesn't take ownership of the entity but keeps a reference only. The stored reference will automatically be nulled if referenced entity is erased.

6.39.3.27 void X3g::Plugin::DbDictionary::Set (System::String^ *pKey*, IEnumerable< DbEntity^ >^ *pEntityRefs*)

Sets a collection of entity references with given key.

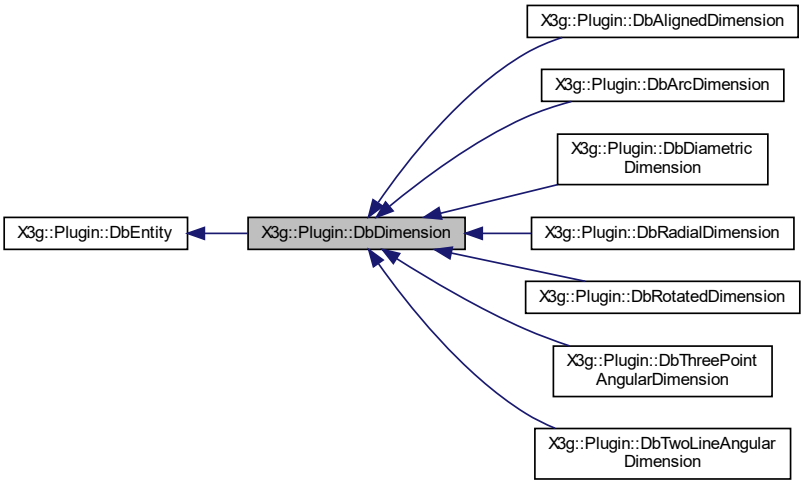
Overwrites existing references.

The dictionary doesn't take ownership of the entities but keeps references only. The stored references will automatically be nulled if referenced entities are erased.

6.40 X3g::Plugin::DbDimension Class Reference

Base class for all dimension types.

Inheritance diagram for X3g::Plugin::DbDimension:



Public Member Functions

- void ResetProperties ()

  Reset all overridden properties to the values of the dimension style.

Properties

- GeVec3d Normal `[get, set]`

  Normal of this Dimension.
- double Elevation `[get, set]`

  Distance from WCS origin to the plane of this dimension in normal direction.
- double Scale `[get, set]`

  Scale factor applied to all sizes, offset and distance variables of this dimension.
- DimensionStyle^ Style `[get, set]`

  Specifies the style of this dimension.
- bool AlwaysShowDimensionLine `[get, set]`

  If true a dimension line between extension lines will also be drawn if the text is outside the dimension lines.
- ArrowHeadType ArrowHead `[get, set]`

  Specifies the type of arrow head at the end of dimension lines.
- double ArrowSize `[get, set]`

  Specifies the size of arrow heads.
- DimensionTextHAlignment TextHorizontalAlignment `[get, set]`

  Specifies the dimension text horizontal position.
- double TextOffset `[get, set]`

  Specifies the gap between dimension text and dimension line.
- String^ TextOverride `[get, set]`

  Specifies the text creation description.
- GeVec3d TextPosition `[get, set]`

  Specifies the text position.
- double TextSize `[get, set]`

Specifies the size of the dimension text.

- bool UseDefaultTextPosition [get, set]

  If true the text position is automatically computed.

- double TextRotation [get, set]

  Specifies the text rotation to local X-Axis in radians.

- DimensionTextVAlignment TextVerticalAlignment [get, set]

  Specifies the dimension text vertical position.

- int Precision [get, set]

  Specifies the number of decimal places.

Additional Inherited Members

6.40.1 Detailed Description

Base class for all dimension types.

6.40.2 Member Function Documentation

6.40.2.1 void X3g::Plugin::DbDimension::ResetProperties ( )

Reset all overridden properties to the values of the dimension style.

6.40.3 Property Documentation

6.40.3.1 bool X3g::Plugin::DbDimension::AlwaysShowDimensionLine [get],[set]

If true a dimension line between extension lines will also be drawn if the text is outside the dimension lines.

6.40.3.2 ArrowHeadType X3g::Plugin::DbDimension::ArrowHead [get],[set]

Specifies the type of arrow head at the end of dimension lines.

6.40.3.3 double X3g::Plugin::DbDimension::ArrowSize [get],[set]

Specifies the size of arrow heads.

6.40.3.4 double X3g::Plugin::DbDimension::Elevation [get],[set]

Distance from WCS origin to the plane of this dimension in normal direction.

6.40.3.5 GeVec3d X3g::Plugin::DbDimension::Normal [get],[set]

Normal of this Dimension.

6.40.3.6 int X3g::Plugin::DbDimension::Precision [get],[set]

Specifies the number of decimal places.

6.40.3.7 double X3g::Plugin::DbDimension::Scale [get], [set]

Scale factor applied to all sizes, offset and distance variables of this dimension.

6.40.3.8 DimensionStyle^ X3g::Plugin::DbDimension::Style [get], [set]

Specifies the style of this dimension.

6.40.3.9 DimensionTextHAlignment X3g::Plugin::DbDimension::TextHorizontalAlignment [get], [set]

Specifies the dimension text horizontal position.

6.40.3.10 double X3g::Plugin::DbDimension::TextOffset [get], [set]

Specifies the gap between dimension text and dimension line.

6.40.3.11 String^ X3g::Plugin::DbDimension::TextOverride [get], [set]

Specifies the text creation description.

"" means only the default text will appear. "." means no text will appear. "<>" insert the default text into text line.

6.40.3.12 GeVec3d X3g::Plugin::DbDimension::TextPosition [get], [set]

Specifies the text position.

6.40.3.13 double X3g::Plugin::DbDimension::TextRotation [get], [set]

Specifies the text rotation to local X-Axis in radians.

0.0 means autoalignment to dimension line.

6.40.3.14 double X3g::Plugin::DbDimension::TextSize [get], [set]

Specifies the size of the dimension text.

6.40.3.15 DimensionTextVAlignment X3g::Plugin::DbDimension::TextVerticalAlignment [get], [set]

Specifies the dimension text vertical position.

6.40.3.16 bool X3g::Plugin::DbDimension::UseDefaultTextPosition [get], [set]

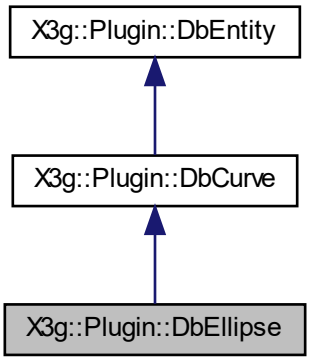
If true the text position is automatically computed.

If false the DbDimension::TextPosition will be used.

6.41 X3g::Plugin::DbEllipse Class Reference

An ellipse.

Inheritance diagram for X3g::Plugin::DbEllipse:



Public Member Functions

- DbEllipse (Layout::Page^ pPage)

  Create a DbEllipse on the given Page.
- DbEllipse (Block^ pOwner)

  Create a DbEllipse as child of the given Block.

Properties

- GeVec3d Center `[get, set]`

  Center point of this ellipse.
- GeVec3d Normal `[get, set]`

  Normal of this ellipse.
- GeVec3d MajorAxis `[get, set]`

  Major axis of this ellipse.
- GeVec3d MinorAxis `[get]`

  Minor axis of this ellipse.
- double RadiusRatio `[get, set]`

  Ratio between major and minor Axis.
- double StartAngle `[get, set]`

  Start angle of this ellipse in radians.
- double EndAngle `[get, set]`

  Start angle of this ellipse in radians.

Additional Inherited Members

6.41.1 Detailed Description

An ellipse.

6.41.2 Constructor & Destructor Documentation

6.41.2.1 X3g::Plugin::DbEllipse::DbEllipse (Layout::Page^ *pPage*)

Create a DbEllipse on the given Page.

6.41.2.2 X3g::Plugin::DbEllipse::DbEllipse (Block^ *pOwner*)

Create a DbEllipse as child of the given Block.

6.41.3 Property Documentation

6.41.3.1 GeVec3d X3g::Plugin::DbEllipse::Center [get], [set]

Center point of this ellipse.

6.41.3.2 double X3g::Plugin::DbEllipse::EndAngle [get], [set]

Start angle of this ellipse in radians.

It's measured counterclockwise from the major axis. For a closed ellipse set StartAngle to 0 and EndAngle to 2 * Pi.

6.41.3.3 GeVec3d X3g::Plugin::DbEllipse::MajorAxis [get], [set]

Major axis of this ellipse.

6.41.3.4 GeVec3d X3g::Plugin::DbEllipse::MinorAxis [get]

Minor axis of this ellipse.

6.41.3.5 GeVec3d X3g::Plugin::DbEllipse::Normal [get], [set]

Normal of this ellipse.

6.41.3.6 double X3g::Plugin::DbEllipse::RadiusRatio [get], [set]

Ratio between major and minor Axis.

Must be a value between 1e-6 and 1.0

6.41.3.7 double X3g::Plugin::DbEllipse::StartAngle [get], [set]

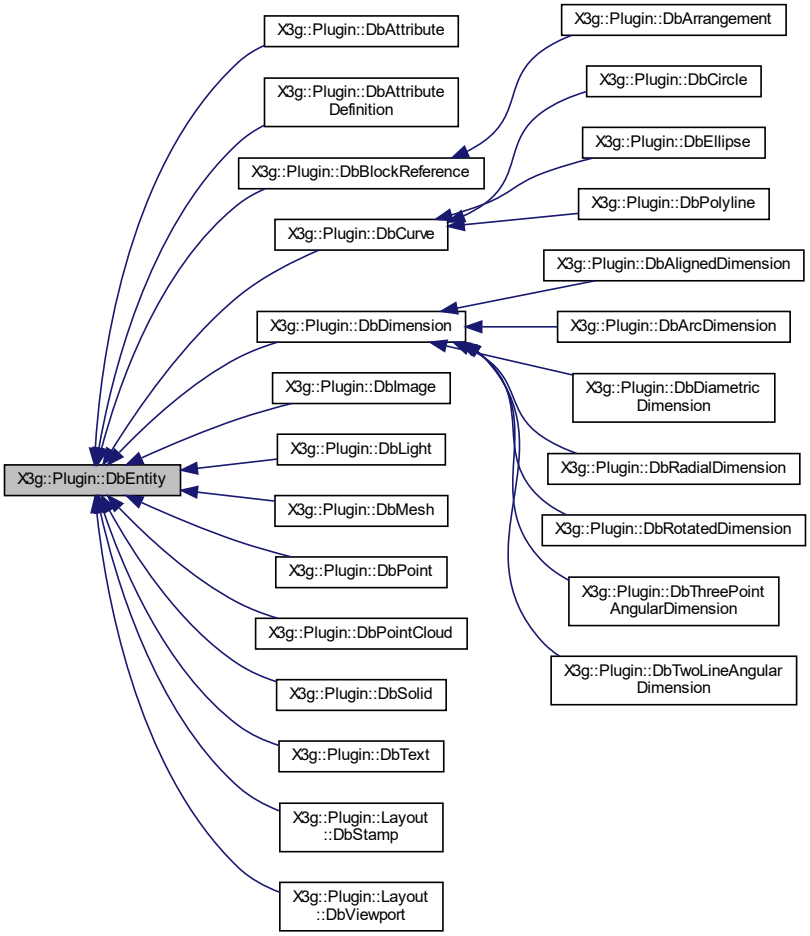
Start angle of this ellipse in radians.

It's measured counterclockwise from the major axis. For a closed ellipse set StartAngle to 0 and EndAngle to 2 * Pi.

6.42 X3g::Plugin::DbEntity Class Reference

Base class for all database objects having a graphical representation.

Inheritance diagram for X3g::Plugin::DbEntity:



Public Member Functions

- GeMatrix^ GetTextureMatrix ()

  The texture matrix determines how to map the material texture to the entity.
- void SetTextureMatrix (GeMatrix^ pMatrix)

  Sets the texture matrix.
- bool TransformBy (GeMatrix^ pTransform)

  Transforms the entity.
- GeMatrix^ GetTransientTransform ()

  Returns transient transformation of this entity.
- void SetTransientTransform (GeMatrix^ pTransform)

  Sets the transient transformation of this entity.
- void UpdateGeometry ()

  Updates the geometry of the entity.
- void Delete ()

  Deletes the entity from the document.
- GeBoundingBox3d^ GetBoundingBox (GeMatrix^ pTransform)

  Returns bounding box of entity within the owning blocks local coordinate system.

- IPropertyProvider^ GetPropertyProvider ()

  Returns property provider of the entity.
- DbEntity^ Clone (Block^ pOwner)

  Creates a clone of the entity.
- DbEntity^ Cut2 (DbEntity^ pEntity)

  Subtracts pEntity from this object.
- DbEntity^ Cut2 (DbEntity^ pEntity, Block^ pToOwner)

  Subtracts pEntity from this object.
- DbEntity^ Common2 (DbEntity^ pEntity)

  Computes an object which represents all common geometry of this entity and pEntity.
- DbEntity^ Common2 (DbEntity^ pEntity, Block^ pToOwner)

  Computes an object which represents all common geometry of this entity and pEntity.
- DbEntity^ Fuse2 (DbEntity^ pEntity)

  Computes an object by merging all geometry of this entity and pEntity.
- DbEntity^ Fuse2 (DbEntity^ pEntity, Block^ pToOwner)

  Computes an object by merging all geometry of this entity and pEntity.
- List< DbEntity^ >^ Slice (GePlane^ pPlane, bool pKeepBothHalves)

  Slice the DbEntity with the plane.
- DbEntity^ PartialMove (GePlane^ pPlane, double pLength)

  Strech or shrink the DbEntity at the given plane.
- void AddCallbacks (DbEntityCallbacks^ pCallbacks)

  Add callbacks to the entity.
- void RemoveCallbacks (DbEntityCallbacks^ pCallbacks)

  Remove callbacks from entity.

Static Public Member Functions

- static DbEntity^ FromId (String^ pEntityId)

  Creates DbEntity to given entity id.

Package Functions

- DbEntity (Block^ pOwner, EntityImpl *pImpl)

  This constructor is used to create new OdDbEntities and add to a block.
- DbEntity (Layout::Page^ pPage, EntityImpl *pImpl)

  This constructor is used to create new OdDbEntities and add to a page.

Properties

- String^ Id `[get]`

  The unqiue id of the entity.
- String^ Name `[get, set]`

  User-defined name of the entity.
- String^ Material `[get, set]`

  Material of this entity.
- EntityColor Color `[get, set]`

  Color of this entity.
- String^ Layer `[get, set]`

  Layer name of this entity.
- String^ OwningBlock `[get]`

Name of the block that owns this entity.

- bool TopLevel [get]

  Returns if entity is owned by modelspace.
- bool Valid [get]

  Returns if entity is valid and not erased.
- DbTextureProjection TextureProjection [get, set]

  The AutoCAD texture projection.
- bool IsModelingSupported [get]

  If true the entity can be used with modeling functions.
- bool Locked [get, set]

  Lock status of this entity.
- DbDictionary^ CustomData [get]

  Allows to attach custom data to the entity.
- String^ CustomType [get, set]

  Allows plugins to identify their entities.
- Object^ CustomObject

  Allows to attach an object which extents the entity (e.g.
- bool IsVisible [get, set]

  Visibility of this entity.
- Hyperlink Link [get, set]

  Get or Set the Hyperlink attached to the DbEntity.

Events

- System::EventHandler^ VisibilityChanged

  Occurs when DbEntity::IsVisible changed.

6.42.1 Detailed Description

Base class for all database objects having a graphical representation.

6.42.2 Constructor & Destructor Documentation

6.42.2.1 X3g::Plugin::DbEntity::DbEntity (Block^ *pOwner*, EntityImpl * *pImpl*) [package]

This constructor is used to create new OdDbEntities and add to a block.

If pOwner is null, an exception is thorwn.

6.42.2.2 X3g::Plugin::DbEntity::DbEntity (Layout::Page^ *pPage*, EntityImpl * *pImpl*) [package]

This constructor is used to create new OdDbEntities and add to a page.

If pPage is null, an exception is thorwn.

6.42.3 Member Function Documentation

6.42.3.1 void X3g::Plugin::DbEntity::AddCallbacks (DbEntityCallbacks^ *pCallbacks*)

Add callbacks to the entity.

6.42.3.2 DbEntity X3g::Plugin::DbEntity::Clone (Block^ *pOwner*)

Creates a clone of the entity.

The owner of the clone can be defined with pOwner. If it is null the owner of the cloned entity will be used.

6.42.3.3 DbEntity X3g::Plugin::DbEntity::Common2 (DbEntity^ *pEntity*)

Computes an object which represents all common geometry of this entity and pEntity.

On success a new entity added to the same block is returned.

Both entities have to return true on IsModelingSupported.

6.42.3.4 DbEntity X3g::Plugin::DbEntity::Common2 (DbEntity^ *pEntity,* Block^ *pToOwner*)

Computes an object which represents all common geometry of this entity and pEntity.

On success a new entity added to pToOwner is returned.

Both entities have to return true on IsModelingSupported.

6.42.3.5 DbEntity X3g::Plugin::DbEntity::Cut2 (DbEntity^ *pEntity*)

Subtracts pEntity from this object.

On success a new entity added to the same block is returned.

Both entities have to return true on IsModelingSupported.

6.42.3.6 DbEntity X3g::Plugin::DbEntity::Cut2 (DbEntity^ *pEntity,* Block^ *pToOwner*)

Subtracts pEntity from this object.

On success a new entity added to pToOwner is returned.

Both entities have to return true on IsModelingSupported.

6.42.3.7 void X3g::Plugin::DbEntity::Delete ( )

Deletes the entity from the document.

This invalidates the entity and it must no longer be used.

6.42.3.8 DbEntity X3g::Plugin::DbEntity::FromId (String^ *pEntityId*) [static]

Creates DbEntity to given entity id.

Returns null if the entity doesn't exist in the current document or if it has been erased.

6.42.3.9 DbEntity X3g::Plugin::DbEntity::Fuse2 (DbEntity^ *pEntity*)

Computes an object by merging all geometry of this entity and pEntity.

On success a new entity added to the same block is returned.

Both entities have to return true on IsModelingSupported.

6.42.3.10 DbEntity X3g::Plugin::DbEntity::Fuse2 (DbEntity^ *pEntity,* Block^ *pToOwner*)

Computes an object by merging all geometry of this entity and pEntity.

On success a new entity added to pToOwner is returned.

Both entities have to return true on IsModelingSupported.

6.42.3.11 GeBoundingBox3d X3g::Plugin::DbEntity::GetBoundingBox (GeMatrix^ *pTransform*)

Returns bounding box of entity within the owning blocks local coordinate system.

Entity geometry may be transformed for bounds computation. Entity must be owned by a block.

Exceptions

System::NotSupported-Exception	Throws exception if entity is not owned by a block.

6.42.3.12 IPropertyProvider X3g::Plugin::DbEntity::GetPropertyProvider ( )

Returns property provider of the entity.

The entity must be part of the current document.

6.42.3.13 GeMatrix X3g::Plugin::DbEntity::GetTextureMatrix ( )

The texture matrix determines how to map the material texture to the entity.

If custom texture coordinates are used, this projection has no effect.

6.42.3.14 GeMatrix X3g::Plugin::DbEntity::GetTransientTransform ( )

Returns transient transformation of this entity.

Transient transforms are not propagated to the database. This fast transformation is used while user interaction is in progress. Otherwise it is always an identity matrix.

6.42.3.15 DbEntity X3g::Plugin::DbEntity::PartialMove (GePlane^ *pPlane,* double *pLength*)

Strech or shrink the DbEntity at the given plane.

Parameters

pPlane	The plane defines where to operate.
pLength	If this value is positive strech the entity. If this value is negative shrinks the entity.

Returns

The modified DbEntity or null.

6.42.3.16 void X3g::Plugin::DbEntity::RemoveCallbacks (DbEntityCallbacks^ *pCallbacks*)

Remove callbacks from entity.

6.42.3.17 void X3g::Plugin::DbEntity::SetTextureMatrix (GeMatrix^ *pMatrix*)

Sets the texture matrix.

6.42.3.18 void X3g::Plugin::DbEntity::SetTransientTransform (GeMatrix^ *pTransform*)

Sets the transient transformation of this entity.

Transient transforms are not propagated to the database. This fast transformation is used while user interaction is in progress. Otherwise it is always an identity matrix.

6.42.3.19 List< DbEntity^ > X3g::Plugin::DbEntity::Slice (GePlane^ *pPlane*, bool *pKeepBothHalves*)

Slice the DbEntity with the plane.

Either returns the half to which the plane is facing or return both halves.

Parameters

pPlane	The plane to slice with.
pKeepBoth-Halves	Keep the half to which the plane is facing only or keep both halves.

Returns

A list of sliced entities. In case of an error, this list might be empty.

6.42.3.20 bool X3g::Plugin::DbEntity::TransformBy (GeMatrix^ *pTransform*)

Transforms the entity.

Returns false if the transformation is not applicable.

6.42.3.21 void X3g::Plugin::DbEntity::UpdateGeometry ( )

Updates the geometry of the entity.

6.42.4 Property Documentation

6.42.4.1 EntityColor X3g::Plugin::DbEntity::Color [get],[set]

Color of this entity.

6.42.4.2 DbDictionary^ X3g::Plugin::DbEntity::CustomData [get]

Allows to attach custom data to the entity.

6.42.4.3 Object^ X3g::Plugin::DbEntity::CustomObject

Allows to attach an object which extents the entity (e.g.

adds properties).

The concrete type of the custom object should depend on DbEntity::CustomType. Plugins are responsible to create and set the custom object when an entity with a particular custom type is added or loaded. This also means that the custom object is not persistent.

6.42.4.4 String^ X3g::Plugin::DbEntity::CustomType [get], [set]

Allows plugins to identify their entities.

The type name should be set once when the entity is created. It's strongly recommended to use plugin name as prefix to prevent naming conflicts between different plugins. Returns an empty string if no custom type was set.

6.42.4.5 String^ X3g::Plugin::DbEntity::Id [get]

The unqiue id of the entity.

Returns an empty string if the entity is not valid.

6.42.4.6 bool X3g::Plugin::DbEntity::IsModelingSupported [get]

If true the entity can be used with modeling functions.

6.42.4.7 bool X3g::Plugin::DbEntity::IsVisible [get], [set]

Visibility of this entity.

Attention: This property is not persistent

6.42.4.8 String^ X3g::Plugin::DbEntity::Layer [get], [set]

Layer name of this entity.

6.42.4.9 Hyperlink X3g::Plugin::DbEntity::Link [get], [set]

Get or Set the Hyperlink attached to the DbEntity.

6.42.4.10 bool X3g::Plugin::DbEntity::Locked [get], [set]

Lock status of this entity.

6.42.4.11 String^ X3g::Plugin::DbEntity::Material [get], [set]

Material of this entity.

6.42.4.12 String^ X3g::Plugin::DbEntity::Name [get], [set]

User-defined name of the entity.

Name is not unique. Empty by default.

6.42.4.13 String^ X3g::Plugin::DbEntity::OwningBlock [get]

Name of the block that owns this entity.

6.42.4.14 DbTextureProjection X3g::Plugin::DbEntity::TextureProjection [get], [set]

The AutoCAD texture projection.

This projection determines (together with the TextureMatrix) how texture coordinates are generated for this entity. If custom texture coordinates are used, this projection has no effect.

6.42.4.15 bool X3g::Plugin::DbEntity::TopLevel [get]

Returns if entity is owned by modelspace.

6.42.4.16 bool X3g::Plugin::DbEntity::Valid [get]

Returns if entity is valid and not erased.

6.42.5 Event Documentation

6.42.5.1 System:: EventHandler^ X3g::Plugin::DbEntity::VisibilityChanged

Occurs when DbEntity::IsVisible changed.

6.43 X3g::Plugin::DbEntityCallbacks Class Reference

Abstract base class for entity callbacks.

Public Member Functions

- virtual void NotifyTransform (String^ pEntityId, GeMatrix^ pTransform, bool pTransient)

  Called when the entity is transformed.
- virtual void NotifyErase (String^ pEntityId)

  Called when the entity is erased.

6.43.1 Detailed Description

Abstract base class for entity callbacks.

6.43.2 Member Function Documentation

6.43.2.1 virtual void X3g::Plugin::DbEntityCallbacks::NotifyErase (String^ *pEntityId*) [virtual]

Called when the entity is erased.

6.43.2.2 virtual void X3g::Plugin::DbEntityCallbacks::NotifyTransform (String^ *pEntityId,* GeMatrix^ *pTransform,* bool *pTransient*) [virtual]

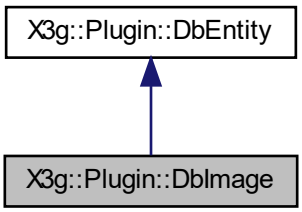
Called when the entity is transformed.

The passed transform matrix is relative to the previous transform. Transforms are transient while user interaction is in progress.

6.44 X3g::Plugin::DbImage Class Reference

A raster image.

Inheritance diagram for X3g::Plugin::DbImage:



Public Member Functions

- DbImage (Block^ pOwner, Image^ pImage)

  Create a DbImage as child of the given block.
- DbImage (Layout::Page^ pPage, Image^ pImage)

  Create a DbImage on the given page.
- void SetImage (Image^ pImage)

  Set image from Image object.
- Image^ GetImage ()

  Returns assigned image.

Properties

- GeVec3d Origin `[get, set]`

  Origin of the image.
- GeVec3d U `[get, set]`

  U vector is the x-Axis of the image.
- GeVec3d V `[get, set]`

  V vector is the y-Axis of the image.
- bool Transparent `[get, set]`

  If true the alpha channel of the image sould be used.
- bool ShowBorder `[get, set]`

  If true a border is visible around the image.
- bool BilinearFiltering `[get, set]`

  Allows to control if image should be rendered with bilinear filtering.

Additional Inherited Members

6.44.1 Detailed Description

A raster image.

6.44.2 Constructor & Destructor Documentation

6.44.2.1 X3g::Plugin::DbImage::DbImage (Block^ *pOwner,* Image^ *pImage*)

Create a DbImage as child of the given block.

6.44.2.2 X3g::Plugin::DbImage::DbImage (Layout::Page^ *pPage,* Image^ *pImage*)

Create a DbImage on the given page.

6.44.3 Member Function Documentation

6.44.3.1 Image X3g::Plugin::DbImage::GetImage ( )

Returns assigned image.

6.44.3.2 void X3g::Plugin::DbImage::SetImage (Image^ *pImage*)

Set image from Image object.

6.44.4 Property Documentation

6.44.4.1 bool X3g::Plugin::DbImage::BilinearFiltering [get],[set]

Allows to control if image should be rendered with bilinear filtering.

Enabled by default.

6.44.4.2 GeVec3d X3g::Plugin::DbImage::Origin [get],[set]

Origin of the image.

6.44.4.3 bool X3g::Plugin::DbImage::ShowBorder [get],[set]

If true a border is visible around the image.

6.44.4.4 bool X3g::Plugin::DbImage::Transparent [get],[set]

If true the alpha channel of the image sould be used.

6.44.4.5 GeVec3d X3g::Plugin::DbImage::U [get],[set]

U vector is the x-Axis of the image.

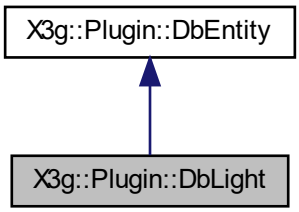
6.44.4.6 GeVec3d X3g::Plugin::DbImage::V [get],[set]

V vector is the y-Axis of the image.

6.45 X3g::Plugin::DbLight Class Reference

A light source which is part of the document.

Inheritance diagram for X3g::Plugin::DbLight:



Classes

- class Instance

  DbLights may be owned by blocks which are referenced multiple times.

Public Member Functions

- bool GetLightEnabled (RenderingCategory pRenderingCategory)

  Returns if light is enabled for given rendering category.
- void SetLightEnabled (RenderingCategory pRenderingCategory, bool pEnabled)

  Enables light for given rendering category.
- bool GetGeometryEnabled (RenderingCategory pRenderingCategory)

  Returns if light geometry is enabled for given rendering category.
- void SetGeometryEnabled (RenderingCategory pRenderingCategory, bool pEnabled)

  Enables light geometry for given rendering category.
- bool GetShadowEnabled (RenderingCategory pRenderingCategory)

  Returns if shadow casting is enabled for given rendering category.
- void SetShadowEnabled (RenderingCategory pRenderingCategory, bool pEnabled)

  Enables shadow casting for given rendering category.
- bool GetPhotometricData (System::IO::Stream^ pStream,[Out] PhotometricDataFormat% pFormat)

  Allows to obtain photometric data.
- void SetPhotometricData (System::IO::Stream^ pStream, PhotometricDataFormat pFormat)

  Sets photometric data in given format.
- array< float >^ GetIntensityDistribution ([Out] int% pWidth,[Out] int% pHeight)

  Returns intensity distribution of photometric lights.
- void UpdateLight ()

  After changes are made to the light this function should be called to update the realtime visualization.

Static Public Member Functions

- static IList< Instance^ >^ GetAllActiveInstances (RenderingCategory pRenderingCategory)

  Returns all active light instances in current document for given rendering category.

Properties

- X3g::Plugin::LightType LightType [get, set]

  The type of the light source.
- System::Drawing::Color LightColor [get, set]

  The color of the light.
- double Intensity [get, set]

  The intensity of the light in candela.
- double OpeningAngle [get, set]

  The opening angle of a spot light in degrees;
- double HotspotAngle [get, set]

  The hotspot angle of a spot light in degrees;
- GeVec3d Position [get, set]

  The position of the light.
- GeVec3d Target [get, set]

  The target of the light.
- GeVec3d Up [get, set]

  Up vector of rectangular area light.
- GeVec3d Size [get, set]

  The size of the light.

Additional Inherited Members

6.45.1 Detailed Description

A light source which is part of the document.

6.45.2 Member Function Documentation

6.45.2.1 IList< DbLight::Instance^ > X3g::Plugin::DbLight::GetAllActiveInstances (RenderingCategory *pRenderingCategory*) [static]

Returns all active light instances in current document for given rendering category.

See Also

DbLight::Instance

6.45.2.2 bool X3g::Plugin::DbLight::GetGeometryEnabled (RenderingCategory *pRenderingCategory*)

Returns if light geometry is enabled for given rendering category.

6.45.2.3 array< float > X3g::Plugin::DbLight::GetIntensityDistribution ([Out] int% *pWidth*, [Out] int% *pHeight*)

Returns intensity distribution of photometric lights.

Returns null if data is not available.

6.45.2.4 bool X3g::Plugin::DbLight::GetLightEnabled (RenderingCategory *pRenderingCategory*)

Returns if light is enabled for given rendering category.

6.45.2.5 bool X3g::Plugin::DbLight::GetPhotometricData (System::IO::Stream^ *pStream,* [Out] PhotometricDataFormat% *pFormat*)

Allows to obtain photometric data.

Returns

True if photometric data is available.

6.45.2.6 bool X3g::Plugin::DbLight::GetShadowEnabled (RenderingCategory *pRenderingCategory*)

Returns if shadow casting is enabled for given rendering category.

6.45.2.7 void X3g::Plugin::DbLight::SetGeometryEnabled (RenderingCategory *pRenderingCategory,* bool *pEnabled*)

Enables light geometry for given rendering category.

6.45.2.8 void X3g::Plugin::DbLight::SetLightEnabled (RenderingCategory *pRenderingCategory,* bool *pEnabled*)

Enables light for given rendering category.

6.45.2.9 void X3g::Plugin::DbLight::SetPhotometricData (System::IO::Stream^ *pStream,* PhotometricDataFormat *pFormat*)

Sets photometric data in given format.

6.45.2.10 void X3g::Plugin::DbLight::SetShadowEnabled (RenderingCategory *pRenderingCategory,* bool *pEnabled*)

Enables shadow casting for given rendering category.

6.45.2.11 void X3g::Plugin::DbLight::UpdateLight ( )

After changes are made to the light this function should be called to update the realtime visualization.

6.45.3 Property Documentation

6.45.3.1 double X3g::Plugin::DbLight::HotspotAngle [get],[set]

The hotspot angle of a spot light in degrees;

6.45.3.2 double X3g::Plugin::DbLight::Intensity [get],[set]

The intensity of the light in candela.

6.45.3.3 System:: Drawing:: Color X3g::Plugin::DbLight::LightColor [get],[set]

The color of the light.

6.45.3.4 X3g:: Plugin:: LightType X3g::Plugin::DbLight::LightType [get],[set]

The type of the light source.

6.45.3.5 double X3g::Plugin::DbLight::OpeningAngle [get],[set]

The opening angle of a spot light in degrees;

6.45.3.6 GeVec3d X3g::Plugin::DbLight::Position [get],[set]

The position of the light.

No effect for directional lights.

6.45.3.7 GeVec3d X3g::Plugin::DbLight::Size [get],[set]

The size of the light.

The bigger the light the softer the shadow. For lights other then area lights the size is equal in all dimensions. Returns angular size in case of directional light.

6.45.3.8 GeVec3d X3g::Plugin::DbLight::Target [get],[set]

The target of the light.

6.45.3.9 GeVec3d X3g::Plugin::DbLight::Up [get],[set]

Up vector of rectangular area light.

Vector has unit length and is perpendicular to light direction.

6.46 X3g::Plugin::DbLight::Instance Class Reference

DbLights may be owned by blocks which are referenced multiple times.

Properties

- DbLight^ Light [get]

  The instanced light.
- GeMatrix^ Transform [get]

  The instance transform.

6.46.1 Detailed Description

DbLights may be owned by blocks which are referenced multiple times.

In this case there are multiple instances of the according DbLight, each instance with it's own world transform.

See Also

DbLight::GetAllActiveInstances

6.46.2 Property Documentation

6.46.2.1 DbLight^ X3g::Plugin::DbLight::Instance::Light [get]

The instanced light.

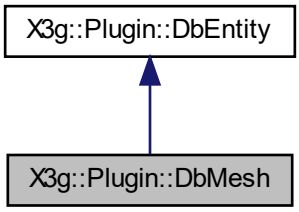
6.46.2.2 GeMatrix^ X3g::Plugin::DbLight::Instance::Transform [get]

The instance transform.

6.47 X3g::Plugin::DbMesh Class Reference

A DbMesh is a collection of polygons used to define a 3D object.

Inheritance diagram for X3g::Plugin::DbMesh:



Public Member Functions

- DbMesh (Block^ pOwner)

  Create a DbMesh as child of the given block.
- void SetMesh (array< GeVec3d >^pVertices, array< int >^pFaces)

  Sets the mesh vertices and faces.
- array< GeVec3d >^ GetVertices ()

  Returns a copy of the mesh vertices.
- array< int >^ GetFaces ()

  Returns a copy of the mesh faces.
- array< GeVec3d >^ GetNormals ()

  Returns a copy of the vertex normals.
- void SetNormals (array< GeVec3d >^pNormals)

  Sets the vertex normals.
- array< GeVec2d >^ GetTexCoords ()

  Returns a copy of the vertex texture coordinates.
- void SetTexCoords (array< GeVec2d >^pTexCoords)

  Sets vertex texture coordinates.

Properties

- bool SingleSided [get, set]

  If enabled the triangles will be rendered single sided.
- double EdgeSoftening [get, set]

  Control the visibility of edges within the Mesh.

Additional Inherited Members

6.47.1 Detailed Description

A DbMesh is a collection of polygons used to define a 3D object.

6.47.2 Constructor & Destructor Documentation

6.47.2.1 X3g::Plugin::DbMesh::DbMesh (Block^ *pOwner*)

Create a DbMesh as child of the given block.

6.47.3 Member Function Documentation

6.47.3.1 array< int > X3g::Plugin::DbMesh::GetFaces ( )

Returns a copy of the mesh faces.

6.47.3.2 array< GeVec3d > X3g::Plugin::DbMesh::GetNormals ( )

Returns a copy of the vertex normals.

6.47.3.3 array< GeVec2d > X3g::Plugin::DbMesh::GetTexCoords ( )

Returns a copy of the vertex texture coordinates.

6.47.3.4 array< GeVec3d > X3g::Plugin::DbMesh::GetVertices ( )

Returns a copy of the mesh vertices.

6.47.3.5 void X3g::Plugin::DbMesh::SetMesh (array< GeVec3d >^ *pVertices,* array< int >^ *pFaces*)

Sets the mesh vertices and faces.

Faces are defined by vertex indices as follows: [num indices], i1, i2, i3, ... [num indices], i1, i2, i3,.... Example: Triangle and Rectangle ([3], 0, 1, 2, [4], 3, 4, 2, 1)

6.47.3.6 void X3g::Plugin::DbMesh::SetNormals (array< GeVec3d >^ *pNormals*)

Sets the vertex normals.

The number of normals must be zero or match the number of vertices.

6.47.3.7 void X3g::Plugin::DbMesh::SetTexCoords (array< GeVec2d >^ *pTexCoords*)

Sets vertex texture coordinates.

The number of texture coordinates must be zero or match the number of vertices. This is a planner extension and has no effect in other DWG viewers. TextureMatrix and TextureProjection have no effect (except for other DWG viewers) when this field is used.

6.47.4 Property Documentation

6.47.4.1 double X3g::Plugin::DbMesh::EdgeSoftening [get],[set]

Control the visibility of edges within the Mesh.

The value is in degree between adjacent faces. Value of 0 shows all edges.

6.47.4.2 bool X3g::Plugin::DbMesh::SingleSided [get],[set]

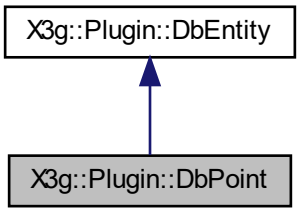
If enabled the triangles will be rendered single sided.

Single sided rendering improves memory consumption and rendering performance but in general it is suitable for closed meshes only. Disabled by default.

6.48 X3g::Plugin::DbPoint Class Reference

A point in 3D space.

Inheritance diagram for X3g::Plugin::DbPoint:



Public Member Functions

- DbPoint (Block^ pOwner)

  Create a DbPoint as child of the given block.

Properties

- GeVec3d Position [get, set]

  The positon of the point.
- double Size [get, set]

  Diameter of rasterized point in pixels.

Additional Inherited Members

6.48.1 Detailed Description

A point in 3D space.

6.48.2 Constructor & Destructor Documentation

6.48.2.1 X3g::Plugin::DbPoint::DbPoint (Block^ *pOwner*)

Create a DbPoint as child of the given block.

6.48.3 Property Documentation

6.48.3.1 GeVec3d X3g::Plugin::DbPoint::Position [get],[set]

The positon of the point.

6.48.3.2 double X3g::Plugin::DbPoint::Size [get],[set]

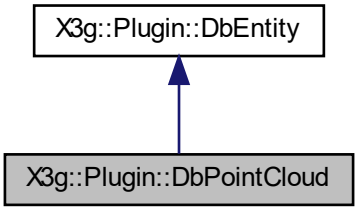
Diameter of rasterized point in pixels.

Value is zero for global point size.

6.49 X3g::Plugin::DbPointCloud Class Reference

A point cloud.

Inheritance diagram for X3g::Plugin::DbPointCloud:



Public Member Functions

- DbPointCloud (Block^ pOwner, System::String^ pFileName)

  Create a DbPointCloud as child of the given block.
- void GetAllPoints ([Out] array< GeVec3f >^%pPositions,[Out] array< System::Drawing::Color >^%pColors)

  Returns all points ignoring density and clip bounds.
- void GetVisiblePoints ([Out] array< GeVec3f >^%pPositions,[Out] array< System::Drawing::Color >^%pColors)

  Returns points with density and clip bounds applied.

Properties

- System::String^ FileName [get]

  Source file of point cloud.

- GeMatrix^ Transform [get, set]

  Transformation of point cloud.
- double Density [get, set]

  Density of point cloud in range [0.0, 1.0].
- GeBoundingBox3d^ ClipBounds [get, set]

  Points outside of bounds are clipped.

Additional Inherited Members

6.49.1 Detailed Description

A point cloud.

6.49.2 Constructor & Destructor Documentation

6.49.2.1 X3g::Plugin::DbPointCloud::DbPointCloud (Block^ *pOwner,* System::String^ *pFileName*)

Create a DbPointCloud as child of the given block.

6.49.3 Member Function Documentation

6.49.3.1 void X3g::Plugin::DbPointCloud::GetAllPoints ([Out] array< GeVec3f >^% *pPositions,* [Out] array< System::Drawing::Color >^% *pColors*)

Returns all points ignoring density and clip bounds.

6.49.3.2 void X3g::Plugin::DbPointCloud::GetVisiblePoints ([Out] array< GeVec3f >^% *pPositions,* [Out] array< System::Drawing::Color >^% *pColors*)

Returns points with density and clip bounds applied.

6.49.4 Property Documentation

6.49.4.1 GeBoundingBox3d^ X3g::Plugin::DbPointCloud::ClipBounds [get],[set]

Points outside of bounds are clipped.

6.49.4.2 double X3g::Plugin::DbPointCloud::Density [get],[set]

Density of point cloud in range [0.0, 1.0].

6.49.4.3 System:: String^ X3g::Plugin::DbPointCloud::FileName [get]

Source file of point cloud.

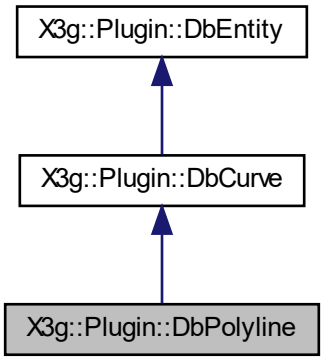
6.49.4.4 GeMatrix^ X3g::Plugin::DbPointCloud::Transform [get],[set]

Transformation of point cloud.

6.50 X3g::Plugin::DbPolyline Class Reference

A line with two or more vertices.

Inheritance diagram for X3g::Plugin::DbPolyline:



Public Member Functions

- DbPolyline (Layout::Page^ pPage)

  Create a DbPolyline on the given Page.
- DbPolyline (Block^ pOwner)

  Create a DbPolyline as child of the given Block.
- void SetClosed (bool pClosed)

  If set to true there is a line between last and first point.
- array< GeVec3d >^ GetVertices ()

  Returns a copy of the vertices this polyline is made of.
- void SetVertices (array< GeVec3d >^pVertices)

  Sets the vertices this polyline is made of.

Additional Inherited Members

6.50.1 Detailed Description

A line with two or more vertices.

6.50.2 Constructor & Destructor Documentation

6.50.2.1 X3g::Plugin::DbPolyline::DbPolyline (Layout::Page^ *pPage*)

Create a DbPolyline on the given Page.

6.50.2.2 X3g::Plugin::DbPolyline::DbPolyline (Block^ *pOwner*)

Create a DbPolyline as child of the given Block.

6.50.3 Member Function Documentation

6.50.3.1 array< GeVec3d > X3g::Plugin::DbPolyline::GetVertices ( )

Returns a copy of the vertices this polyline is made of.

6.50.3.2 void X3g::Plugin::DbPolyline::SetClosed (bool *pClosed*)

If set to true there is a line between last and first point.

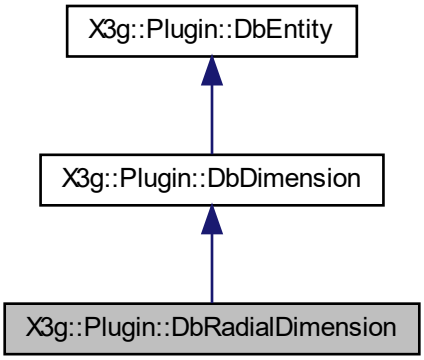
6.50.3.3 void X3g::Plugin::DbPolyline::SetVertices (array< GeVec3d >^ *pVertices*)

Sets the vertices this polyline is made of.

6.51 X3g::Plugin::DbRadialDimension Class Reference

A length dimension used to measure a radius.

Inheritance diagram for X3g::Plugin::DbRadialDimension:



Public Member Functions

- DbRadialDimension (Layout::Page^ pPage)

  Create a DbRadialDimension on the given page.
- DbRadialDimension (Block^ pOwner)

  Create a DbRadialDimension as child of the given block.
- IList< DbEntity^ >^ GetChordRefs ()

  Returns a list of references for the chord point.
- void SetChordRefs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the chord point.
- IList< DbEntity^ >^ GetCenterRefs ()

  Returns a list of references for the center point.
- void SetCenterRefs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the center point.

Properties

- GeVec3d ChordPoint [get, set]

  The chord point of the arc being dimensioned.
- GeVec3d CenterPoint [get, set]

  The center point of the arc being dimensioned.
- double ScaleFactor [get, set]

  Specifies the distance multiplier for measurement.
- LengthUnit Unit [get, set]

  Specifies the unit for measurement.

Additional Inherited Members

6.51.1 Detailed Description

A length dimension used to measure a radius.

6.51.2 Constructor & Destructor Documentation

6.51.2.1 X3g::Plugin::DbRadialDimension::DbRadialDimension (Layout::Page^ *pPage*)

Create a DbRadialDimension on the given page.

6.51.2.2 X3g::Plugin::DbRadialDimension::DbRadialDimension (Block^ *pOwner*)

Create a DbRadialDimension as child of the given block.

6.51.3 Member Function Documentation

6.51.3.1 IList< DbEntity^ > X3g::Plugin::DbRadialDimension::GetCenterRefs ( )

Returns a list of references for the center point.

6.51.3.2 IList< DbEntity^ > X3g::Plugin::DbRadialDimension::GetChordRefs ( )

Returns a list of references for the chord point.

6.51.3.3 void X3g::Plugin::DbRadialDimension::SetCenterRefs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the center point.

6.51.3.4 void X3g::Plugin::DbRadialDimension::SetChordRefs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the chord point.

6.51.4 Property Documentation

6.51.4.1 GeVec3d X3g::Plugin::DbRadialDimension::CenterPoint [get],[set]

The center point of the arc being dimensioned.

6.51.4.2 GeVec3d X3g::Plugin::DbRadialDimension::ChordPoint [get], [set]

The chord point of the arc being dimensioned.

6.51.4.3 double X3g::Plugin::DbRadialDimension::ScaleFactor [get], [set]

Specifies the distance multiplier for measurement.

6.51.4.4 LengthUnit X3g::Plugin::DbRadialDimension::Unit [get], [set]

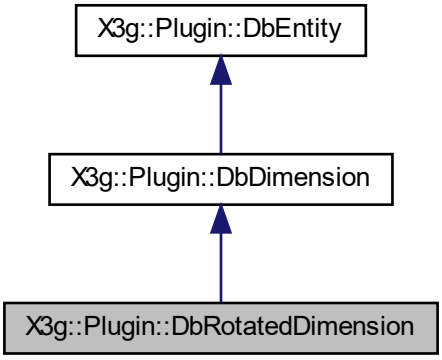
Specifies the unit for measurement.

If set to 'Undefined' the scale factor will be used.

6.52 X3g::Plugin::DbRotatedDimension Class Reference

A length dimension with a dimension line rotated to measuring points.

Inheritance diagram for X3g::Plugin::DbRotatedDimension:



Public Member Functions

- DbRotatedDimension (Layout::Page^ pPage)

  Create a DbRotatedDimension on the given page.
- DbRotatedDimension (Block^ pOwner)

  Create a DbRotatedDimension as child of the given block.
- IList< DbEntity^ >^ GetXLine1Refs ()

  Returns a list of references for the first measuring point.
- void SetXLine1Refs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the first measuring points.
- IList< DbEntity^ >^ GetXLine2Refs ()

  Returns a list of references for the second measuring point.
- void SetXLine2Refs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the second measuring point.

Properties

- GeVec3d XLine1Point [get, set]

  First measuring point and start point of the first extension line.
- GeVec3d XLine2Point [get, set]

  Second measuring point and start point of the second extension line.
- GeVec3d DimensionLinePoint [get, set]

  A point of the dimension line.
- double DimensionLineRotation [get, set]

  Specifies the dimension line rotation to local X-Axis in radians.
- double BaseOffset [get, set]

  Specifies the distance of reference points to extension lines.
- double ExtensionLineLength [get, set]

  Specifies the length of extension line beyond dimension line.
- double ScaleFactor [get, set]

  Specifies the distance multiplier for measurement.
- LengthUnit Unit [get, set]

  Specifies the unit for measurement.

Additional Inherited Members

6.52.1 Detailed Description

A length dimension with a dimension line rotated to measuring points.

6.52.2 Constructor & Destructor Documentation

6.52.2.1 X3g::Plugin::DbRotatedDimension::DbRotatedDimension (Layout::Page^ *pPage*)

Create a DbRotatedDimension on the given page.

6.52.2.2 X3g::Plugin::DbRotatedDimension::DbRotatedDimension (Block^ *pOwner*)

Create a DbRotatedDimension as child of the given block.

6.52.3 Member Function Documentation

6.52.3.1 IList< DbEntity^ > X3g::Plugin::DbRotatedDimension::GetXLine1Refs ( )

Returns a list of references for the first measuring point.

6.52.3.2 IList< DbEntity^ > X3g::Plugin::DbRotatedDimension::GetXLine2Refs ( )

Returns a list of references for the second measuring point.

6.52.3.3 void X3g::Plugin::DbRotatedDimension::SetXLine1Refs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the first measuring points.

6.52.3.4 void X3g::Plugin::DbRotatedDimension::SetXLine2Refs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the second measuring point.

6.52.4 Property Documentation

6.52.4.1 double X3g::Plugin::DbRotatedDimension::BaseOffset [get], [set]

Specifies the distance of reference points to extension lines.

6.52.4.2 GeVec3d X3g::Plugin::DbRotatedDimension::DimensionLinePoint [get], [set]

A point of the dimension line.

6.52.4.3 double X3g::Plugin::DbRotatedDimension::DimensionLineRotation [get], [set]

Specifies the dimension line rotation to local X-Axis in radians.

6.52.4.4 double X3g::Plugin::DbRotatedDimension::ExtensionLineLength [get], [set]

Specifies the length of extension line beyond dimension line.

6.52.4.5 double X3g::Plugin::DbRotatedDimension::ScaleFactor [get], [set]

Specifies the distance multiplier for measurement.

6.52.4.6 LengthUnit X3g::Plugin::DbRotatedDimension::Unit [get], [set]

Specifies the unit for measurement.
If set to 'Undefined' the scale factor will be used.

6.52.4.7 GeVec3d X3g::Plugin::DbRotatedDimension::XLine1Point [get], [set]

First measuring point and start point of the first extension line.

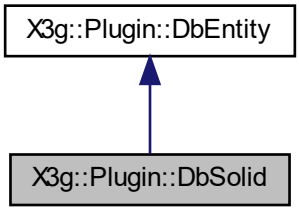
6.52.4.8 GeVec3d X3g::Plugin::DbRotatedDimension::XLine2Point [get], [set]

Second measuring point and start point of the second extension line.

6.53 X3g::Plugin::DbSolid Class Reference

A DbSolid indicates a geometrical object defined by paramters.

Inheritance diagram for X3g::Plugin::DbSolid:



Additional Inherited Members

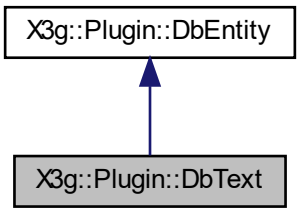
6.53.1 Detailed Description

A DbSolid indicates a geometrical object defined by paramters.

6.54 X3g::Plugin::DbText Class Reference

A text entity.

Inheritance diagram for X3g::Plugin::DbText:



Public Member Functions

- DbText (Block^ pOwner)

  Create a DbImage as child of the given block.
- DbText (Layout::Page^ pPage)

  Create a DbImage on the given page.

Properties

- String^ Text `[get, set]`

The text of this entity.

- GeVec3d Position [get, set]

  Position of the text.
- GeVec3d Direction [get, set]

  Direction of the text.
- GeVec3d Normal [get, set]

  Normal of the text.
- double CharacterSize [get, set]

  Size of text characters.
- HorizontalTextAlignment HorizontalAlignment [get, set]

  Gets or sets horizontal alignment of the text.
- VerticalTextAlignment VerticalAlignment [get, set]

  Gets or sets vertical alignment of the text.

Additional Inherited Members

6.54.1 Detailed Description

A text entity.

6.54.2 Constructor & Destructor Documentation

6.54.2.1 X3g::Plugin::DbText::DbText (Block^ *pOwner*)

Create a DbImage as child of the given block.

6.54.2.2 X3g::Plugin::DbText::DbText (Layout::Page^ *pPage*)

Create a DbImage on the given page.

6.54.3 Property Documentation

6.54.3.1 double X3g::Plugin::DbText::CharacterSize [get],[set]

Size of text characters.

6.54.3.2 GeVec3d X3g::Plugin::DbText::Direction [get],[set]

Direction of the text.

6.54.3.3 HorizontalTextAlignment X3g::Plugin::DbText::HorizontalAlignment [get],[set]

Gets or sets horizontal alignment of the text.

6.54.3.4 GeVec3d X3g::Plugin::DbText::Normal [get],[set]

Normal of the text.

6.54.3.5 GeVec3d X3g::Plugin::DbText::Position [get], [set]

Position of the text.

6.54.3.6 String^ X3g::Plugin::DbText::Text [get], [set]

The text of this entity.

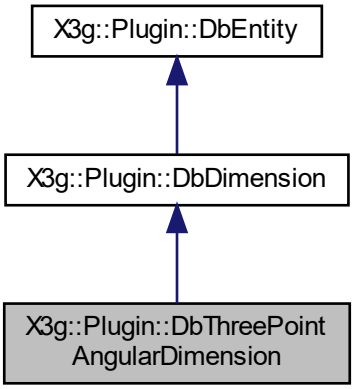
6.54.3.7 VerticalTextAlignment X3g::Plugin::DbText::VerticalAlignment [get], [set]

Gets or sets vertical alignment of the text.

6.55 X3g::Plugin::DbThreePointAngularDimension Class Reference

An angular dimension based on center point and to mesuring points.

Inheritance diagram for X3g::Plugin::DbThreePointAngularDimension:



Public Member Functions

- DbThreePointAngularDimension (Layout::Page^ pPage)

  Create a DbThreePointAngularDimension on the given page.
- DbThreePointAngularDimension (Block^ pOwner)

  Create a DbThreePointAngularDimension as child of the given block.
- IList< DbEntity^ >^ GetXLine1Refs ()

  Returns a list of references for the first measuring point.
- void SetXLine1Refs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the first measuring point.
- IList< DbEntity^ >^ GetXLine2Refs ()

  Returns a list of references for the second measuring point.
- void SetXLine2Refs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the second measuring point.

Properties

- GeVec3d XLine1Point [get, set]

  First measuring point and start point of the first extension line.
- GeVec3d XLine2Point [get, set]

  Second measuring point and start point of the second extension line.
- GeVec3d CenterPoint [get, set]

  The center point of arc being dimensioned.
- GeVec3d DimensionLinePoint [get, set]

  A point of the dimension arc.
- double BaseOffset [get, set]

  Specifies the distance of reference points to extension lines.
- double ExtensionLineLength [get, set]

  Specifies the length of extension line beyond dimension line.
- int Precision [get, set]

  Specifies the number of decimal places.

Additional Inherited Members

6.55.1 Detailed Description

An angular dimension based on center point and to mesuring points.

6.55.2 Constructor & Destructor Documentation

6.55.2.1 X3g::Plugin::DbThreePointAngularDimension::DbThreePointAngularDimension (Layout::Page^ *pPage*)

Create a DbThreePointAngularDimension on the given page.

6.55.2.2 X3g::Plugin::DbThreePointAngularDimension::DbThreePointAngularDimension (Block^ *pOwner*)

Create a DbThreePointAngularDimension as child of the given block.

6.55.3 Member Function Documentation

6.55.3.1 IList< DbEntity^ > X3g::Plugin::DbThreePointAngularDimension::GetXLine1Refs ( )

Returns a list of references for the first measuring point.

6.55.3.2 IList< DbEntity^ > X3g::Plugin::DbThreePointAngularDimension::GetXLine2Refs ( )

Returns a list of references for the second measuring point.

6.55.3.3 void X3g::Plugin::DbThreePointAngularDimension::SetXLine1Refs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the first measuring point.

6.55.3.4 void X3g::Plugin::DbThreePointAngularDimension::SetXLine2Refs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the second measuring point.

6.55.4 Property Documentation

6.55.4.1 double X3g::Plugin::DbThreePointAngularDimension::BaseOffset [get],[set]

Specifies the distance of reference points to extension lines.

6.55.4.2 GeVec3d X3g::Plugin::DbThreePointAngularDimension::CenterPoint [get],[set]

The center point of arc being dimensioned.

6.55.4.3 GeVec3d X3g::Plugin::DbThreePointAngularDimension::DimensionLinePoint [get],[set]

A point of the dimension arc.

6.55.4.4 double X3g::Plugin::DbThreePointAngularDimension::ExtensionLineLength [get],[set]

Specifies the length of extension line beyond dimension line.

6.55.4.5 int X3g::Plugin::DbThreePointAngularDimension::Precision [get],[set]

Specifies the number of decimal places.

6.55.4.6 GeVec3d X3g::Plugin::DbThreePointAngularDimension::XLine1Point [get],[set]

First measuring point and start point of the first extension line.

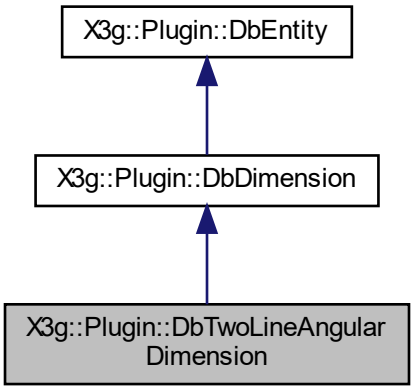
6.55.4.7 GeVec3d X3g::Plugin::DbThreePointAngularDimension::XLine2Point [get],[set]

Second measuring point and start point of the second extension line.

6.56 X3g::Plugin::DbTwoLineAngularDimension Class Reference

An angular dimension between two lines.

Inheritance diagram for X3g::Plugin::DbTwoLineAngularDimension:



Public Member Functions

- DbTwoLineAngularDimension (Layout::Page^ pPage)

  Create a DbTwoLineAngularDimension on the given page.
- DbTwoLineAngularDimension (Block^ pOwner)

  Create a DbTwoLineAngularDimension as child of the given block.
- IList< DbEntity^ >^ GetXLine1StartRefs ()

  Returns a list of references for the first line start point.
- void SetXLine1StartRefs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the first line start point.
- IList< DbEntity^ >^ GetXLine1EndRefs ()

  Returns a list of references for the first line end point.
- void SetXLine1EndRefs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the first line end point.
- IList< DbEntity^ >^ GetXLine2StartRefs ()

  Returns a list of references for the second line start point.
- void SetXLine2StartRefs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the second line start point.
- IList< DbEntity^ >^ GetXLine2EndRefs ()

  Returns a list of references for the second line end point.
- void SetXLine2EndRefs (IEnumerable< DbEntity^ >^pRefs)

  Set references for the second line end point.

Properties

- GeVec3d XLine1StartPoint `[get, set]`

  The start point of the first line.
- GeVec3d XLine1EndPoint `[get, set]`

  The end point of the first line.
- GeVec3d XLine2StartPoint `[get, set]`

The start point of the second line.

- GeVec3d XLine2EndPoint `[get, set]`

  The end point of the second line.
- GeVec3d DimensionLinePoint `[get, set]`

  A point of the dimension line.
- double BaseOffset `[get, set]`

  Specifies the distance of reference points to extension lines.
- double ExtensionLineLength `[get, set]`

  Specifies the length of extension line beyond dimension line.
- int Precision `[get, set]`

  Specifies the number of decimal places.

Additional Inherited Members

6.56.1 Detailed Description

An angular dimension between two lines.

6.56.2 Constructor & Destructor Documentation

6.56.2.1 X3g::Plugin::DbTwoLineAngularDimension::DbTwoLineAngularDimension (Layout::Page^ *pPage*)

Create a DbTwoLineAngularDimension on the given page.

6.56.2.2 X3g::Plugin::DbTwoLineAngularDimension::DbTwoLineAngularDimension (Block^ *pOwner*)

Create a DbTwoLineAngularDimension as child of the given block.

6.56.3 Member Function Documentation

6.56.3.1 IList< DbEntity^ > X3g::Plugin::DbTwoLineAngularDimension::GetXLine1EndRefs ( )

Returns a list of references for the first line end point.

6.56.3.2 IList< DbEntity^ > X3g::Plugin::DbTwoLineAngularDimension::GetXLine1StartRefs ( )

Returns a list of references for the first line start point.

6.56.3.3 IList< DbEntity^ > X3g::Plugin::DbTwoLineAngularDimension::GetXLine2EndRefs ( )

Returns a list of references for the second line end point.

6.56.3.4 IList< DbEntity^ > X3g::Plugin::DbTwoLineAngularDimension::GetXLine2StartRefs ( )

Returns a list of references for the second line start point.

6.56.3.5 void X3g::Plugin::DbTwoLineAngularDimension::SetXLine1EndRefs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the first line end point.

6.56.3.6 void X3g::Plugin::DbTwoLineAngularDimension::SetXLine1StartRefs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the first line start point.

6.56.3.7 void X3g::Plugin::DbTwoLineAngularDimension::SetXLine2EndRefs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the second line end point.

6.56.3.8 void X3g::Plugin::DbTwoLineAngularDimension::SetXLine2StartRefs (IEnumerable< DbEntity^ >^ *pRefs*)

Set references for the second line start point.

6.56.4 Property Documentation

6.56.4.1 double X3g::Plugin::DbTwoLineAngularDimension::BaseOffset [get],[set]

Specifies the distance of reference points to extension lines.

6.56.4.2 GeVec3d X3g::Plugin::DbTwoLineAngularDimension::DimensionLinePoint [get],[set]

A point of the dimension line.

6.56.4.3 double X3g::Plugin::DbTwoLineAngularDimension::ExtensionLineLength [get],[set]

Specifies the length of extension line beyond dimension line.

6.56.4.4 int X3g::Plugin::DbTwoLineAngularDimension::Precision [get],[set]

Specifies the number of decimal places.

6.56.4.5 GeVec3d X3g::Plugin::DbTwoLineAngularDimension::XLine1EndPoint [get],[set]

The end point of the first line.

6.56.4.6 GeVec3d X3g::Plugin::DbTwoLineAngularDimension::XLine1StartPoint [get],[set]

The start point of the first line.

6.56.4.7 GeVec3d X3g::Plugin::DbTwoLineAngularDimension::XLine2EndPoint [get],[set]

The end point of the second line.

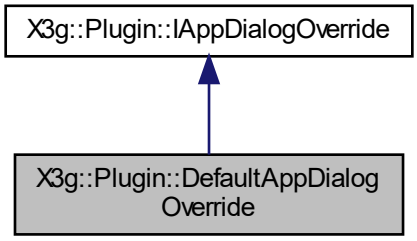
6.56.4.8 GeVec3d X3g::Plugin::DbTwoLineAngularDimension::XLine2StartPoint [get],[set]

The start point of the second line.

6.57 X3g::Plugin::DefaultAppDialogOverride Class Reference

Implementation of IAppDialogOverride which lets application show all dialogs.

Inheritance diagram for X3g::Plugin::DefaultAppDialogOverride:



Public Member Functions

- virtual bool GetInsertScale (const GeBoundingBox3d^ pBounds,[In, Out] LengthUnit% pUnit,[In, Out] double% pCustomScale)

  Called if planning unit is undefined on import.
- virtual bool RequestArticleConversion (ArticleConversionType pType,[Out] bool% pAllowConversion)

  Called if an action would imply the conversion of an article entity.
- virtual bool ContinueImportWithLayerVisibilityChange ([Out] bool% pContinue)

  Called on import if all imported entities are on invisible layers.

6.57.1 Detailed Description

Implementation of IAppDialogOverride which lets application show all dialogs.

6.57.2 Member Function Documentation

6.57.2.1 virtual bool X3g::Plugin::DefaultAppDialogOverride::ContinueImportWithLayerVisibilityChange ([Out] bool% *pContinue*) [virtual]

Called on import if all imported entities are on invisible layers.

Parameters

pContinue	Returns true if import should be continued. Layers will be enabled in this case.

Returns

Return true to override application dialog.

Implements X3g::Plugin::IAppDialogOverride.

6.57.2.2 virtual bool X3g::Plugin::DefaultAppDialogOverride::GetInsertScale (const GeBoundingBox3d^ *pBounds*, [In, Out] LengthUnit% *pUnit*, [In, Out] double% *pCustomScale*) [virtual]

Called if planning unit is undefined on import.

Parameters

pBounds	Bounding box of inserted planning.
pUnit	Length unit of inserted planning.
pCustomScale	Scale in addition to unit scale.

Returns

Return true to override application dialog.

Implements X3g::Plugin::IAppDialogOverride.

6.57.2.3 virtual bool X3g::Plugin::DefaultAppDialogOverride::RequestArticleConversion (ArticleConversionType *pType*, [Out] bool% *pAllowConversion*) [virtual]

Called if an action would imply the conversion of an article entity.

Parameters

pType	Kind of conversion.
pAllow-Conversion	Returns true if conversion is allowed. Action is aborted otherwise.

Returns

Return true to override application dialog.

Implements X3g::Plugin::IAppDialogOverride.

6.58 X3g::Plugin::DimensionStyle Class Reference

A style for dimensions.

Public Member Functions

- void UpdateDimensions ()

  Force a redraw of all connected dimension.

Properties

- bool AlwaysShowDimensionLine [get, set]

  If true a dimension line between extension lines will also be drawn if the text is outside the dimension lines.
- int AngularPrecision [get, set]

  Specifies the number of decimal places for angular dimensions.
- bool AutoUpdateEnabled [get, set]

  If set to true a property change will redraw all connected dimension.
- ArrowHeadType ArrowHead [get, set]

  Specifies the type of arrow head at the end of dimension lines.
- double ArrowSize [get, set]

  Specifies the size of arrow heads.
- double BaseOffset [get, set]

  Specifies the distance of reference points to extension lines.
- double ExtensionLineLength [get, set]

Specifies the length of extension line beyond dimension line.

- String^ Font [get, set]

  Specifies the font (e.g.

- int LinearPrecision [get, set]

  Specifies the number of decimal places for linear dimensions.

- bool IsUse [get]

  Returns true if one or more dimension use this style.

- String^ Name [get, set]

  Specifies a unique name of the style.

- double ScaleFactor [get, set]

  Specifies the distance multiplier for measurement.

- DimensionTextHAlignment TextHorizontalAlignment [get, set]

  Specifies the dimension text horizontal position.

- double TextOffset [get, set]

  Specifies the gap between dimension text and dimension line.

- double TextSize [get, set]

  Specifies the size of the dimension text.

- DimensionTextVAlignment TextVerticalAlignment [get, set]

  Specifies the dimension text vertical position.

- LengthUnit Unit [get, set]

  Specifies the unit for measurement.

6.58.1 Detailed Description

A style for dimensions.

6.58.2 Member Function Documentation

6.58.2.1 void X3g::Plugin::DimensionStyle::UpdateDimensions ( )

Force a redraw of all connected dimension.

6.58.3 Property Documentation

6.58.3.1 bool X3g::Plugin::DimensionStyle::AlwaysShowDimensionLine [get],[set]

If true a dimension line between extension lines will also be drawn if the text is outside the dimension lines.

6.58.3.2 int X3g::Plugin::DimensionStyle::AngularPrecision [get],[set]

Specifies the number of decimal places for angular dimensions.

6.58.3.3 ArrowHeadType X3g::Plugin::DimensionStyle::ArrowHead [get],[set]

Specifies the type of arrow head at the end of dimension lines.

6.58.3.4 double X3g::Plugin::DimensionStyle::ArrowSize [get],[set]

Specifies the size of arrow heads.

6.58.3.5 bool X3g::Plugin::DimensionStyle::AutoUpdateEnabled [get],[set]

If set to true a property change will redraw all connected dimension.

6.58.3.6 double X3g::Plugin::DimensionStyle::BaseOffset [get],[set]

Specifies the distance of reference points to extension lines.

6.58.3.7 double X3g::Plugin::DimensionStyle::ExtensionLineLength [get],[set]

Specifies the length of extension line beyond dimension line.

6.58.3.8 String^ X3g::Plugin::DimensionStyle::Font [get],[set]

Specifies the font (e.g.
Arial) of the dimension text.

6.58.3.9 bool X3g::Plugin::DimensionStyle::IsUse [get]

Returns true if one or more dimension use this style.

6.58.3.10 int X3g::Plugin::DimensionStyle::LinearPrecision [get],[set]

Specifies the number of decimal places for linear dimensions.

6.58.3.11 String^ X3g::Plugin::DimensionStyle::Name [get],[set]

Specifies a unique name of the style.

6.58.3.12 double X3g::Plugin::DimensionStyle::ScaleFactor [get],[set]

Specifies the distance multiplier for measurement.

6.58.3.13 DimensionTextHAlignment X3g::Plugin::DimensionStyle::TextHorizontalAlignment [get],[set]

Specifies the dimension text horizontal position.

6.58.3.14 double X3g::Plugin::DimensionStyle::TextOffset [get],[set]

Specifies the gap between dimension text and dimension line.

6.58.3.15 double X3g::Plugin::DimensionStyle::TextSize [get],[set]

Specifies the size of the dimension text.

6.58.3.16 DimensionTextVAlignment X3g::Plugin::DimensionStyle::TextVerticalAlignment [get],[set]

Specifies the dimension text vertical position.

6.58.3.17 LengthUnit X3g::Plugin::DimensionStyle::Unit [get],[set]

Specifies the unit for measurement.

If set to 'Undefined' the scale factor will be used.

6.59 X3g::Plugin::DimensionStyleManager Class Reference

This class manage dimension styles of current document.

Public Member Functions

- DimensionStyle^ GetStyle (String^ pName)

  Returns a existing style with given name.
- IList< DimensionStyle^ >^ GetStyles ()

  Returns a list of all dimension styles in the document.
- DimensionStyle^ CreateStyle (String^ pName)

  Creates a new style with given name.
- void RemoveStyle (DimensionStyle^ pStyle)

  Removes given style from the document.

Properties

- DimensionStyle^ CurrentBaseStyle [get, set]

  Specifies the style to use when creating new dimensions in model space.
- DimensionStyle^ CurrentLayoutStyle [get, set]

  Specifies the style to use when creating new dimensions in layout mode.

6.59.1 Detailed Description

This class manage dimension styles of current document.

6.59.2 Member Function Documentation

6.59.2.1 DimensionStyle X3g::Plugin::DimensionStyleManager::CreateStyle (String^ *pName*)

Creates a new style with given name.

The name must be unique for this document otherwise creation fails.

6.59.2.2 DimensionStyle X3g::Plugin::DimensionStyleManager::GetStyle (String^ *pName*)

Returns a existing style with given name.

6.59.2.3 IList< DimensionStyle^ > X3g::Plugin::DimensionStyleManager::GetStyles ( )

Returns a list of all dimension styles in the document.

6.59.2.4 void X3g::Plugin::DimensionStyleManager::RemoveStyle (DimensionStyle^ *pStyle*)

Removes given style from the document.

Fails if style is in use.

6.59.3 Property Documentation

6.59.3.1 DimensionStyle^ X3g::Plugin::DimensionStyleManager::CurrentBaseStyle [get],[set]

Specifies the style to use when creating new dimensions in model space.

6.59.3.2 DimensionStyle^ X3g::Plugin::DimensionStyleManager::CurrentLayoutStyle [get],[set]

Specifies the style to use when creating new dimensions in layout mode.

6.60 X3g::Plugin::DocumentManager Class Reference

Document Manager.

Public Member Functions

- IList< DbEntity^ >^ GetSelection ()

  Retrieves all currently selected entities.
- void SetSelection (IEnumerable< DbEntity^ >^pSelection)

  Sets selected entities.
- bool NewPlanning ()

  Clears the document to start a new planning.
- bool LoadPlanning (String^ pFileName)

  Loads a planning.
- bool SavePlanning (String^ pFileName, DocumentFileFormat pVersion, bool pCopy)

  Saves the current planning.
- bool SaveEntities (String^ pFileName, IEnumerable< DbEntity^ >^pEntities, DocumentFileType pType, DocumentFileFormat pVersion)

  Saves given entities as dwg/dxf.
- IList< DbEntity^ >^ InsertPlanning (String^ pFileName, bool pInteractive)

  Inserts a planning into the current planning.
- IList< DbEntity^ >^ InsertPlanning (System::IO::Stream^ pStream, String^ pFileNameHint, bool pInteractive)

  Inserts the content of a stream into the current planning.
- unsigned int Purge (DocumentPurgeFlags pFlags)

  Purges the document.
- void Audit ([Out] unsigned int% pNumErrors,[Out] unsigned int% pNumFixed)

  Audits/repairs the document.

Properties

- String^ DocumentName [get, set]

  The name of the document.
- LengthUnit DocumentLengthUnit [get, set]

The document length unit.

- bool ReadOnly [get]

  True if the document file was opened in read-only/write-protected mode.
- bool Modified [get]

  True if the document was modified.
- AcisQuality ActiveAcisQuality [get]

  The currently active acis quality.
- BlockManager^ Blocks [get]

  Returns block manager.
- CameraManager^ Cameras [get]

  Returns camera manager.
- DimensionStyleManager^ DimStyles [get]

  Returns dimension style manager.
- LayerManager^ Layers [get]

  Returns layer manager.
- LightManager^ Lights [get]

  Returns light manager.
- MaterialManager^ Materials [get]

  Returns material manager.
- Block^ ModelSpace [get]

  Returns the modelspace block of current document.
- DocumentSummary^ Summary [get]

  Retrieves the document summary, which holds additional document information.
- DbDictionary^ CustomData [get]

  Allows to attach custom data to the document.

Events

- GeometryUpdateEventHandler^ GeometryUpdated [add, remove, raise]

  Occurs when geometry of top level entity has been updated.
- LayersChangedEventHandler^ LayersChanged

  Occurs when layers have been changed (add/remove/rename/visibility changed).

6.60.1 Detailed Description

Document Manager.

6.60.2 Member Function Documentation

6.60.2.1 void X3g::Plugin::DocumentManager::Audit ([Out] unsigned int% *pNumErrors*, [Out] unsigned int% *pNumFixed*)

Audits/repairs the document.

Returns the number of found and fixed errors.

6.60.2.2 IList< DbEntity^ > X3g::Plugin::DocumentManager::GetSelection ( )

Retrieves all currently selected entities.

6.60.2.3 IList<DbEntity^> ^ X3g::Plugin::DocumentManager::InsertPlanning (String^ *pFileName,* bool *pInteractive*)

Inserts a planning into the current planning.

If 'pInteractive' is true, the user can interactively choose the position of the inserted objects. Otherwise the objects are placed at the document origin. The function returns a list of created modelspace entities if 'pInteractive' is false. Otherwise an empty list is returned. If 'pInteractive' is true you may react on EventType::DocumentEntitiesAdded to retrieve the inserted entities. Note that EventType::DocumentEntitiesAdded is not fired if 'pInteractive' is false.

Supported file formats: dwg, dxf, 3ds, skp, ifc, sat, sab

See Also

IAppDialogOverride::GetInsertScale

6.60.2.4 IList< DbEntity^ > X3g::Plugin::DocumentManager::InsertPlanning (System::IO::Stream^ *pStream,* String^ *pFileNameHint,* bool *pInteractive*)

Inserts the content of a stream into the current planning.

Parameters

pStream	Any type of input stream that contains a single file to import.
pFileNameHint	If possible, the name of the original file. This is mostly important for the suffix like ".dwg" or ".glb".
pInteractive	If 'pInteractive' is true you may react on EventType::DocumentEntitiesAdded to retrieve the inserted entities. Note that EventType::DocumentEntitiesAdded is not fired if 'pInteractive' is false.

Returns

Depending on the imported file, the return value might be a List of one or more DbEntity.

Currently supported file streams: DWG, binary glTF v2.0 (glb).

6.60.2.5 bool X3g::Plugin::DocumentManager::LoadPlanning (String^ *pFileName*)

Loads a planning.

Returns false if loading failed.

Supported file formats: dwg, dxf, 3ds, skp, ifc, sat, sab

See Also

IAppDialogOverride::GetInsertScale

6.60.2.6 bool X3g::Plugin::DocumentManager::NewPlanning ( )

Clears the document to start a new planning.

Returns false on fail.

6.60.2.7 unsigned int X3g::Plugin::DocumentManager::Purge (DocumentPurgeFlags *pFlags*)

Purges the document.

All unused objects will be removed. Returns the number of purged objects.

6.60.2.8 bool X3g::Plugin::DocumentManager::SaveEntities (String^ *pFileName,* IEnumerable< DbEntity^ >^ *pEntities,* DocumentFileType *pType,* DocumentFileFormat *pVersion*)

Saves given entities as dwg/dxf.

Returns false if saving failed.

If pVersion = Undefined the internal/loaded version of the document is used.

6.60.2.9 bool X3g::Plugin::DocumentManager::SavePlanning (String^ *pFileName,* DocumentFileFormat *pVersion,* bool *pCopy*)

Saves the current planning.

Returns false if saving failed.

The planning will be saved as dxf if the file extension is .dxf, otherwise as dwg. If pCopy is true only a copy of the current document will be saved. If pVersion = Undefined the internal/loaded version of the document is used.

6.60.2.10 void X3g::Plugin::DocumentManager::SetSelection (IEnumerable< DbEntity^ >^ *pSelection*)

Sets selected entities.

6.60.3 Property Documentation

6.60.3.1 AcisQuality X3g::Plugin::DocumentManager::ActiveAcisQuality [get]

The currently active acis quality.

6.60.3.2 BlockManager^ X3g::Plugin::DocumentManager::Blocks [get]

Returns block manager.

6.60.3.3 CameraManager^ X3g::Plugin::DocumentManager::Cameras [get]

Returns camera manager.

6.60.3.4 DbDictionary^ X3g::Plugin::DocumentManager::CustomData [get]

Allows to attach custom data to the document.

6.60.3.5 DimensionStyleManager^ X3g::Plugin::DocumentManager::DimStyles [get]

Returns dimension style manager.

6.60.3.6 LengthUnit X3g::Plugin::DocumentManager::DocumentLengthUnit [get],[set]

The document length unit.

6.60.3.7 String^ X3g::Plugin::DocumentManager::DocumentName [get],[set]

The name of the document.

(Usally the path to the file the document was opend from).

6.60.3.8 LayerManager^ X3g::Plugin::DocumentManager::Layers [get]

Returns layer manager.

6.60.3.9 LightManager^ X3g::Plugin::DocumentManager::Lights [get]

Returns light manager.

6.60.3.10 MaterialManager^ X3g::Plugin::DocumentManager::Materials [get]

Returns material manager.

6.60.3.11 Block^ X3g::Plugin::DocumentManager::ModelSpace [get]

Returns the modelspace block of current document.

The modelspace block is the block which contains all toplevel 3d space elements.

6.60.3.12 bool X3g::Plugin::DocumentManager::Modified [get]

True if the document was modified.

6.60.3.13 bool X3g::Plugin::DocumentManager::ReadOnly [get]

True if the document file was opened in read-only/write-protected mode.

6.60.3.14 DocumentSummary^ X3g::Plugin::DocumentManager::Summary [get]

Retrieves the document summary, which holds additional document information.

6.60.4 Event Documentation

6.60.4.1 GeometryUpdateEventHandler^ X3g::Plugin::DocumentManager::GeometryUpdated [add], [remove], [raise]

Occurs when geometry of top level entity has been updated.

This event may also be raised when DbEntity::UpdateGeometry is called.

6.60.4.2 LayersChangedEventHandler^ X3g::Plugin::DocumentManager::LayersChanged

Occurs when layers have been changed (add/remove/rename/visibility changed).

6.61 X3g::Plugin::DocumentSummary Class Reference

Document Summary.

Inherited by X3g::Plugin::IDocumentSummery.

Public Member Functions

- String^ GetCustomInfo (String^ pInfoKey)

  Retrieves custom information by key.
- String^ GetCustomInfo (int pInfoIndex,[Out] String^ %pInfoKey)

  Retrieves custom information by index.
- void SetCustomInfo (String^ pInfoKey, String^ pValue)

  Sets custom information by key.
- void SetCustomInfo (int pInfoIndex, String^ pInfoKey, String^ pValue)

  Sets custom information by index.
- void DeleteCustomInfo (String^ pInfoKey)

  Deletes custom information by key.
- void DeleteCustomInfo (int pInfoIndex)

  Deletes custom information by index.

Properties

- String^ Title `[get, set]`

  The document's title.
- String^ Author `[get, set]`

  The document's author.
- String^ Subject `[get, set]`

  The document's subject.
- String^ RevisionNumber `[get, set]`

  The document's revision number.
- String^ Keywords `[get, set]`

  The document's keywords.
- String^ LastSavedBy `[get, set]`

  The document's LastSavedBy field.
- String^ Hyperlink `[get, set]`

  The document's hyperlink field.
- String^ Comments `[get, set]`

  Additional comments.
- int CustomInfoCount `[get]`

  Number of custom information fields in the document.

6.61.1 Detailed Description

Document Summary.

6.61.2 Member Function Documentation

6.61.2.1 void X3g::Plugin::DocumentSummary::DeleteCustomInfo (String^ *pInfoKey*)

Deletes custom information by key.

6.61.2.2 void X3g::Plugin::DocumentSummary::DeleteCustomInfo (int *pInfoIndex*)

Deletes custom information by index.

6.61.2.3 String X3g::Plugin::DocumentSummary::GetCustomInfo (String^ *pInfoKey*)

Retrieves custom information by key.

6.61.2.4 String X3g::Plugin::DocumentSummary::GetCustomInfo (int *pInfoIndex,* [Out] String^ % *pInfoKey*)

Retrieves custom information by index.

6.61.2.5 void X3g::Plugin::DocumentSummary::SetCustomInfo (String^ *pInfoKey,* String^ *pValue*)

Sets custom information by key.

6.61.2.6 void X3g::Plugin::DocumentSummary::SetCustomInfo (int *pInfoIndex,* String^ *pInfoKey,* String^ *pValue*)

Sets custom information by index.

6.61.3 Property Documentation

6.61.3.1 String^ X3g::Plugin::DocumentSummary::Author [get],[set]

The document's author.

6.61.3.2 String^ X3g::Plugin::DocumentSummary::Comments [get],[set]

Additional comments.

6.61.3.3 int X3g::Plugin::DocumentSummary::CustomInfoCount [get]

Number of custom information fields in the document.

6.61.3.4 String^ X3g::Plugin::DocumentSummary::Hyperlink [get],[set]

The document's hyperlink field.

6.61.3.5 String^ X3g::Plugin::DocumentSummary::Keywords [get],[set]

The document's keywords.

6.61.3.6 String^ X3g::Plugin::DocumentSummary::LastSavedBy [get],[set]

The document's LastSavedBy field.

6.61.3.7 String^ X3g::Plugin::DocumentSummary::RevisionNumber [get],[set]

The document's revision number.

6.61.3.8 String^ X3g::Plugin::DocumentSummary::Subject [get],[set]

The document's subject.

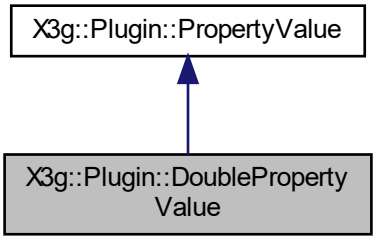
6.61.3.9 String^ X3g::Plugin::DocumentSummary::Title [get],[set]

The document's title.

6.62 X3g::Plugin::DoublePropertyValue Class Reference

Double Property Value.

Inheritance diagram for X3g::Plugin::DoublePropertyValue:



Public Member Functions

- DoublePropertyValue (double pValue)

  Constructor.

Properties

- double Value [get, set]

  The double value.

6.62.1 Detailed Description

Double Property Value.

6.62.2 Constructor & Destructor Documentation

6.62.2.1 X3g::Plugin::DoublePropertyValue::DoublePropertyValue (double *pValue*)

Constructor.

Parameters

pValue	Initial double value.

6.62.3 Property Documentation

6.62.3.1 double X3g::Plugin::DoublePropertyValue::Value [get], [set]

The double value.

6.63 X3g::Plugin::DragDropEventArgs Class Reference

Provides data associated with drag'n'drop events and allows handlers to reply if they accept the drop.

Inherits EventArgs.

Public Member Functions

- void AcceptDrop ()

  Listeners of DragOver event call this method if they want to handle drop of data.

Properties

- System::Windows::IDataObject^ Data [get]

  Returns the data object which shall be dropped.

6.63.1 Detailed Description

Provides data associated with drag'n'drop events and allows handlers to reply if they accept the drop.

6.63.2 Member Function Documentation

6.63.2.1 void X3g::Plugin::DragDropEventArgs::AcceptDrop ( )

Listeners of DragOver event call this method if they want to handle drop of data.

This will change the cursor. Listeners of Drop event call this method when drop was successful.

6.63.3 Property Documentation

6.63.3.1 System:: Windows:: IDataObject^ X3g::Plugin::DragDropEventArgs::Data [get]

Returns the data object which shall be dropped.

Data is currently always in file drop format.

6.64 X3g::Plugin::EntityColor Struct Reference

Color of an entity.

Public Member Functions

- EntityColor (Color pColor)

  Create EntityColor from Color.

Public Attributes

- EntityColorMethod Method

  The color method.
- Color Color

  The color, if the color method is 'ByColor'.

6.64.1 Detailed Description

Color of an entity.

6.64.2 Constructor & Destructor Documentation

6.64.2.1 X3g::Plugin::EntityColor::EntityColor (Color *pColor*)

Create EntityColor from Color.

Method is initialized with EntityColorMethod::ByColor.

6.64.3 Member Data Documentation

6.64.3.1 Color X3g::Plugin::EntityColor::Color

The color, if the color method is 'ByColor'.

6.64.3.2 EntityColorMethod X3g::Plugin::EntityColor::Method

The color method.

Determines how the color of an entity should be computed. Possible values are 'ByColor', 'ByBlock' or 'ByLayer'.

6.65 X3g::Plugin::EntityEventInfo Class Reference

Parameter interface for events about entities.

Public Member Functions

- virtual ReadOnlyCollection
  < DbEntity^ >^ GetEntities ()=0

  Returns all affected entities.
- virtual ReadOnlyCollection
  < DbEntity^ >^ GetEntities (System::String^ pCustomType)=0

  Returns all entities with given custom type.

Public Attributes

- template<typename T >
  where T: DbEntity ReadOnlyCollection<T>^ GetEntities()

  Returns all entities of given type.

6.65.1 Detailed Description

Parameter interface for events about entities.

6.65.2 Member Function Documentation

6.65.2.1 virtual ReadOnlyCollection<DbEntity^> ^ X3g::Plugin::EntityEventInfo::GetEntities ( ) [pure virtual]

Returns all affected entities.

6.65.2.2 virtual ReadOnlyCollection<DbEntity^> ^ X3g::Plugin::EntityEventInfo::GetEntities (System::String^ *pCustomType*) [pure virtual]

Returns all entities with given custom type.

6.65.3 Member Data Documentation

6.65.3.1 template<typename T > where X3g::Plugin::EntityEventInfo::T

Returns all entities of given type.

6.66 X3g::Plugin::Export3dsParams Struct Reference

3DS export parameters.

Public Attributes

- AcisQuality AcisQuality

  ACIS vectorization quality setting.
- double DefaultEdgeSmoothness

  Default edge smoothness.
- double DefaultNormalSmoothness

  Default normal smoothness.
- ReadOnlyCollection< String^ >^ Entities

  List of model space entities which should be exported.
- Export3DSCompatLevel CompatLevel

  Compatibility level.
- bool SingleSided

  Single or double sided.

6.66.1 Detailed Description

3DS export parameters.

An example initialisation would look like:

```
var exportParams = new Export3dsParams {
    AcisQuality = AcisQuality.Medium,
    DefaultEdgeSmoothness = 19.8,
    DefaultNormalSmoothness = 19.8,
    Entities = null,
```

```
    CompatLevel = Export3DSCompatLevel.MeshInstances,
    SingleSided = false
};
```

6.66.2 Member Data Documentation

6.66.2.1 AcisQuality X3g::Plugin::Export3dsParams::AcisQuality

ACIS vectorization quality setting.

6.66.2.2 Export3DSCompatLevel X3g::Plugin::Export3dsParams::CompatLevel

Compatibility level.

6.66.2.3 double X3g::Plugin::Export3dsParams::DefaultEdgeSmoothness

Default edge smoothness.

6.66.2.4 double X3g::Plugin::Export3dsParams::DefaultNormalSmoothness

Default normal smoothness.

6.66.2.5 ReadOnlyCollection<String^> ^ X3g::Plugin::Export3dsParams::Entities

List of model space entities which should be exported.

If Entities is not set, then the whole planning is exported.

6.66.2.6 bool X3g::Plugin::Export3dsParams::SingleSided

Single or double sided.

6.67 X3g::Plugin::ExportDaeParams Struct Reference

Dae export parameters.

Public Attributes

- AcisQuality AcisQuality

  ACIS vectorization quality setting.
- double DefaultEdgeSmoothness

  Default edge smoothness.
- double DefaultNormalSmoothness

  Default normal smoothness.
- ReadOnlyCollection< String^ >^ Entities

  List of model space entities which should be exported.
- bool Export2d

  True if 2D geometry should be exported.
- bool Export3d

  True if 3D geometry should be exported.

- double Scale

  Scaling factor.
- LengthUnit Unit

  Length unit.
- bool DuplicateDoubleSided

  Duplicate double sided faces.
- int MaxTextureSize

  Define max texture resolution.
- bool Edges

  Export with edges.
- bool FlipMirroredFaces

  Flip faces for entities with mirror transform.

6.67.1 Detailed Description

Dae export parameters.

An example initialisation would look like:

```
var exportParams = new ExportDaeParams {
    AcisQuality = AcisQuality.Medium,
    DefaultEdgeSmoothness = 19.8,
    DefaultNormalSmoothness = 19.8,
    Entities = null,
    Export2d = false,
    Export3d = true,
    Scale = 1.0,
    Unit = LengthUnit.Meter,
    DuplicateDoubleSided = false,
    MaxTextureSize = 8192,
    Edges = false,
    FlipMirroredFaces = true
};
```

6.67.2 Member Data Documentation

6.67.2.1 AcisQuality X3g::Plugin::ExportDaeParams::AcisQuality

ACIS vectorization quality setting.

6.67.2.2 double X3g::Plugin::ExportDaeParams::DefaultEdgeSmoothness

Default edge smoothness.

6.67.2.3 double X3g::Plugin::ExportDaeParams::DefaultNormalSmoothness

Default normal smoothness.

6.67.2.4 bool X3g::Plugin::ExportDaeParams::DuplicateDoubleSided

Duplicate double sided faces.

6.67.2.5 bool X3g::Plugin::ExportDaeParams::Edges

Export with edges.

6.67.2.6 ReadOnlyCollection<String^> ^ X3g::Plugin::ExportDaeParams::Entities

List of model space entities which should be exported.

If Entities is not set, then the whole planning is exported.

6.67.2.7 bool X3g::Plugin::ExportDaeParams::Export2d

True if 2D geometry should be exported.

6.67.2.8 bool X3g::Plugin::ExportDaeParams::Export3d

True if 3D geometry should be exported.

6.67.2.9 bool X3g::Plugin::ExportDaeParams::FlipMirroredFaces

Flip faces for entities with mirror transform.

6.67.2.10 int X3g::Plugin::ExportDaeParams::MaxTextureSize

Define max texture resolution.

6.67.2.11 double X3g::Plugin::ExportDaeParams::Scale

Scaling factor.

6.67.2.12 LengthUnit X3g::Plugin::ExportDaeParams::Unit

Length unit.

6.68 X3g::Plugin::ExportEgmParams Struct Reference

Egm export parameters.

Public Attributes

- String^ Block

  Block which should be exported.
- Dictionary< String^ , String^ >^ LayerMapping

  Rename layers in EGM.

6.68.1 Detailed Description

Egm export parameters.

6.68.2 Member Data Documentation

6.68.2.1 String ^ X3g::Plugin::ExportEgmParams::Block

Block which should be exported.

If block is not set, modelspace is exported.

6.68.2.2 Dictionary<String^, String^> ^ X3g::Plugin::ExportEgmParams::LayerMapping

Rename layers in EGM.

Keys are DWG layer names, values are EGM layer names. Layers which are not in this map are retained.

6.69 X3g::Plugin::ExportFbxParams Struct Reference

Fbx export parameters.

Public Attributes

- AcisQuality AcisQuality

  ACIS vectorization quality setting.
- double DefaultEdgeSmoothness

  Default edge smoothness.
- double DefaultNormalSmoothness

  Default normal smoothness.
- ReadOnlyCollection< String^ >^ Entities

  List of model space entities which should be exported.
- int Format

  Format id.
- bool SingleSided

  Single or double sided faces.
- bool EmbedMedia

  Embedded media.
- bool WithLights

  Export of lights.
- bool FlipMirroredFaces

  Flip faces for entities with mirror transform.

6.69.1 Detailed Description

Fbx export parameters.

An example initialisation would look like:

```
var exportParams = new ExportFbxParams {
    AcisQuality = AcisQuality.Medium,
    DefaultEdgeSmoothness = 19.8,
    DefaultNormalSmoothness = 19.8,
    Entities = null,
    Format = 0,
    SingleSided = true,
    EmbedMedia = true,
```

```
    WithLights = false,
    FlipMirroredFaces = true
};
```

6.69.2 Member Data Documentation

6.69.2.1 AcisQuality X3g::Plugin::ExportFbxParams::AcisQuality

ACIS vectorization quality setting.

6.69.2.2 double X3g::Plugin::ExportFbxParams::DefaultEdgeSmoothness

Default edge smoothness.

6.69.2.3 double X3g::Plugin::ExportFbxParams::DefaultNormalSmoothness

Default normal smoothness.

6.69.2.4 bool X3g::Plugin::ExportFbxParams::EmbedMedia

Embedded media.

6.69.2.5 ReadOnlyCollection<String^> ^ X3g::Plugin::ExportFbxParams::Entities

List of model space entities which should be exported.

If Entities is not set, then the whole planning is exported.

6.69.2.6 bool X3g::Plugin::ExportFbxParams::FlipMirroredFaces

Flip faces for entities with mirror transform.

6.69.2.7 int X3g::Plugin::ExportFbxParams::Format

Format id.

Use -1 to autodetect, otherwise 0.

6.69.2.8 bool X3g::Plugin::ExportFbxParams::SingleSided

Single or double sided faces.

6.69.2.9 bool X3g::Plugin::ExportFbxParams::WithLights

Export of lights.

6.70 X3g::Plugin::ExportObjParams Struct Reference

Obj export parameters.

Public Attributes

- AcisQuality AcisQuality

  ACIS vectorization quality setting.
- double DefaultNormalSmoothness

  Default normal smoothness.
- ReadOnlyCollection< String^ >^ Entities

  List of model space entities which should be exported.
- bool DuplicateDoubleSided

  Duplicate double sided faces.
- bool AlwaysExportUVs

  If true, export UV coordinates even if there's no textured material.
- bool NoMaterials

  If true, don't export materials.
- ObjOrientation Orientation

  Orientation of the exported coordinates.
- String^ Block

  Name of the Block which should be exported.

6.70.1 Detailed Description

Obj export parameters.

An example initialisation would look like:

```
var exportParams = new ExportObjParams {
    AcisQuality = AcisQuality.Medium,
    DefaultNormalSmoothness = 19.8,
    Entities = null,
    DuplicateDoubleSided = false,
    AlwaysExportUVs = true,
    NoMaterials = false,
    Orientation = ObjOrientation.Y_Up,
    Block = null
};
```

6.70.2 Member Data Documentation

6.70.2.1 AcisQuality X3g::Plugin::ExportObjParams::AcisQuality

ACIS vectorization quality setting.

6.70.2.2 bool X3g::Plugin::ExportObjParams::AlwaysExportUVs

If true, export UV coordinates even if there's no textured material.

6.70.2.3 String ^ X3g::Plugin::ExportObjParams::Block

Name of the Block which should be exported.

If Entities is set, then Entitiies is preferred. If Entities and Block are not set, then the whole planning is exported.

6.70.2.4 double X3g::Plugin::ExportObjParams::DefaultNormalSmoothness

Default normal smoothness.

6.70.2.5 bool X3g::Plugin::ExportObjParams::DuplicateDoubleSided

Duplicate double sided faces.

6.70.2.6 ReadOnlyCollection<String^> ^ X3g::Plugin::ExportObjParams::Entities

List of model space entities which should be exported.

If Entities is not set, then the Block is exported. If Entities and Block are not set, then the whole planning is exported.

6.70.2.7 bool X3g::Plugin::ExportObjParams::NoMaterials

If true, don't export materials.

6.70.2.8 ObjOrientation X3g::Plugin::ExportObjParams::Orientation

Orientation of the exported coordinates.

6.71 X3g::Plugin::ExportOffParams Struct Reference

Off export parameters.

Public Attributes

- String^ Block

  Block which should be exported.
- AcisQuality AcisQuality

  ACIS vectorization quality setting.
- bool DuplicateDoubleSided

  Duplicate double sided faces.

6.71.1 Detailed Description

Off export parameters.

An example initialisation would look like:

```
var exportParams = new ExportOffParams {
    Block = null,
    AcisQuality = AcisQuality.Medium,
    DuplicateDoubleSided = false
};
```

6.71.2 Member Data Documentation

6.71.2.1 AcisQuality X3g::Plugin::ExportOffParams::AcisQuality

ACIS vectorization quality setting.

6.71.2.2 String ^ X3g::Plugin::ExportOffParams::Block

Block which should be exported.

If block is not set, modelspace is exported.

6.71.2.3 bool X3g::Plugin::ExportOffParams::DuplicateDoubleSided

Duplicate double sided faces.

6.72 X3g::Plugin::ExportParams Struct Reference

Params for modelspace export of GeometryManager.

Public Attributes

- AcisQuality AcisQuality

  ACIS vectorization quality setting.
- double DefaultEdgeSmoothness

  Default edge smoothness.
- double DefaultNormalSmoothness

  Default normal smoothness.
- ReadOnlyCollection< String^ >^ Entities

  List of model space entities which should be exported.
- bool SingleSided

  Single or double sided faces.
- View^ View

  If View is not null, the layer visibilities of this view will be used.

6.72.1 Detailed Description

Params for modelspace export of GeometryManager.

An example initialisation would look like:

```
var exportParams = new ExportParams {
    AcisQuality = AcisQuality.Medium,
    DefaultEdgeSmoothness = 19.8,
    DefaultNormalSmoothness = 19.8,
    Entities = null,
    SingleSided = false,
    View = null
};
```

6.72.2 Member Data Documentation

6.72.2.1 AcisQuality X3g::Plugin::ExportParams::AcisQuality

ACIS vectorization quality setting.

6.72.2.2 double X3g::Plugin::ExportParams::DefaultEdgeSmoothness

Default edge smoothness.

6.72.2.3 double X3g::Plugin::ExportParams::DefaultNormalSmoothness

Default normal smoothness.

6.72.2.4 ReadOnlyCollection<String^> ^ X3g::Plugin::ExportParams::Entities

List of model space entities which should be exported.

If Entities is not set, then the whole planning is exported.

6.72.2.5 bool X3g::Plugin::ExportParams::SingleSided

Single or double sided faces.

6.72.2.6 View ^ X3g::Plugin::ExportParams::View

If View is not null, the layer visibilities of this view will be used.

6.73 X3g::Plugin::ExportSkpParams Struct Reference

Skp export parameters.

Public Attributes

- AcisQuality AcisQuality

  ACIS vectorization quality setting.
- double DefaultEdgeSmoothness

  Default edge smoothness.
- double DefaultNormalSmoothness

  Default normal smoothness.
- ReadOnlyCollection< String^ >^ Entities

  List of model space entities which should be exported.
- View^ View

  The camera from the given view will be used as the default camera in the skp file.
- bool ExportMaterials

  True if materials should be exported.
- SkpFormatVersion FormatVersion

  Version of skp format.

6.73.1 Detailed Description

Skp export parameters.

An example initialisation would look like:

```
var exportParams = new ExportSkpParams {
    AcisQuality = AcisQuality.Medium,
    DefaultEdgeSmoothness = 19.8,
    DefaultNormalSmoothness = 19.8,
    Entities = null,
    View = null,
    ExportMaterials = true,
    FormatVersion = SkpFormatVersion.Skp2020
};
```

6.73.2 Member Data Documentation

6.73.2.1 AcisQuality X3g::Plugin::ExportSkpParams::AcisQuality

ACIS vectorization quality setting.

6.73.2.2 double X3g::Plugin::ExportSkpParams::DefaultEdgeSmoothness

Default edge smoothness.

6.73.2.3 double X3g::Plugin::ExportSkpParams::DefaultNormalSmoothness

Default normal smoothness.

6.73.2.4 ReadOnlyCollection<String^> ^ X3g::Plugin::ExportSkpParams::Entities

List of model space entities which should be exported.

If Entities is not set, then the whole planning is exported.

6.73.2.5 bool X3g::Plugin::ExportSkpParams::ExportMaterials

True if materials should be exported.

6.73.2.6 SkpFormatVersion X3g::Plugin::ExportSkpParams::FormatVersion

Version of skp format.

6.73.2.7 View ^ X3g::Plugin::ExportSkpParams::View

The camera from the given view will be used as the default camera in the skp file.

6.74 X3g::Plugin::FbxFormatDescription Struct Reference

Description of a supported FBX format.

Public Attributes

- String^ Extension

  Extension.
- String^ Description

  Description.

6.74.1 Detailed Description

Description of a supported FBX format.

6.74.2 Member Data Documentation

6.74.2.1 String ^ X3g::Plugin::FbxFormatDescription::Description

Description.

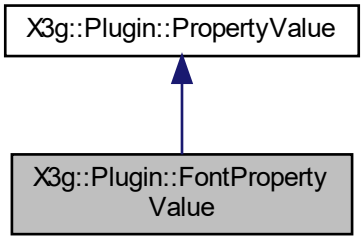
6.74.2.2 String ^ X3g::Plugin::FbxFormatDescription::Extension

Extension.

6.75 X3g::Plugin::FontPropertyValue Class Reference

Font Property Value.

Inheritance diagram for X3g::Plugin::FontPropertyValue:



Public Member Functions

- FontPropertyValue (System::String^ pValue)

  Constructor.

Properties

- System::String^ Value `[get, set]`

  The font value (font name).

6.75.1 Detailed Description

Font Property Value.

6.75.2 Constructor & Destructor Documentation

6.75.2.1 X3g::Plugin::FontPropertyValue::FontPropertyValue (System::String^ *pValue*)

Constructor.

Parameters

pValue	Initial font (name).

6.75.3 Property Documentation

6.75.3.1 System:: String^ X3g::Plugin::FontPropertyValue::Value [get],[set]

The font value (font name).

6.76 X3g::Plugin::GeBoundingBox3d Class Reference

Bounding box.

Public Member Functions

- GeBoundingBox3d ()

  Constructor.
- bool IsEmpty ()

  Returns if bounding box is empty.
- bool Contains (GeVec3d pPoint)

  Returns if bounding box contains the given point.
- void Extend (GeBoundingBox3d^ pOther)

  Extends bounding box so that it contains the other bounding box.
- void Extend (GeVec3d Point)

  Extends bounding box so that it contains the given point.

Public Attributes

- GeVec3d Min

  Minimum.
- GeVec3d Max

  Maximum.

Properties

- GeVec3d Center [get]

  Center of bounding box.

6.76.1 Detailed Description

Bounding box.

6.76.2 Constructor & Destructor Documentation

6.76.2.1 X3g::Plugin::GeBoundingBox3d::GeBoundingBox3d ( )

Constructor.

6.76.3 Member Function Documentation

6.76.3.1 bool X3g::Plugin::GeBoundingBox3d::Contains (GeVec3d *pPoint*)

Returns if bounding box contains the given point.

6.76.3.2 void X3g::Plugin::GeBoundingBox3d::Extend (GeBoundingBox3d^ *pOther*)

Extends bounding box so that it contains the other bounding box.

6.76.3.3 void X3g::Plugin::GeBoundingBox3d::Extend (GeVec3d *Point*)

Extends bounding box so that it contains the given point.

6.76.3.4 bool X3g::Plugin::GeBoundingBox3d::IsEmpty ( )

Returns if bounding box is empty.

6.76.4 Member Data Documentation

6.76.4.1 GeVec3d X3g::Plugin::GeBoundingBox3d::Max

Maximum.

6.76.4.2 GeVec3d X3g::Plugin::GeBoundingBox3d::Min

Minimum.

6.76.5 Property Documentation

6.76.5.1 GeVec3d X3g::Plugin::GeBoundingBox3d::Center [get]

Center of bounding box.

6.77 X3g::Plugin::GeComponent Class Reference

Container for a set of drawables.

Public Attributes

- String^ Name

  Component name which is the name of the according DbBlock.
- List< GeMesh^ >^ Meshes

  Mesh data.
- List< GePolylines^ >^ Polylines

  Polylines.
- List< GePoints^ >^ Points

  Points.
- List< GeComponentInstance^ >^ Instances

  Component instances.

6.77.1 Detailed Description

Container for a set of drawables.

6.77.2 Member Data Documentation

6.77.2.1 List<GeComponentInstance^> ^ X3g::Plugin::GeComponent::Instances

Component instances.

6.77.2.2 List<GeMesh^> ^ X3g::Plugin::GeComponent::Meshes

Mesh data.

6.77.2.3 String ^ X3g::Plugin::GeComponent::Name

Component name which is the name of the according DbBlock.

6.77.2.4 List<GePoints^> ^ X3g::Plugin::GeComponent::Points

Points.

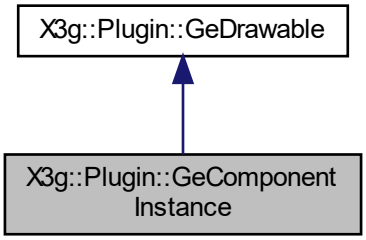
6.77.2.5 List<GePolylines^> ^ X3g::Plugin::GeComponent::Polylines

Polylines.

6.78 X3g::Plugin::GeComponentInstance Class Reference

A component instance instanciates and transforms a component.

Inheritance diagram for X3g::Plugin::GeComponentInstance:



Public Attributes

- GeMatrix^ Transform

  Transformation matrix.

- GeComponent^ Component

  Referenced component.

6.78.1 Detailed Description

A component instance instanciates and transforms a component.

6.78.2 Member Data Documentation

6.78.2.1 GeComponent ^ X3g::Plugin::GeComponentInstance::Component

Referenced component.

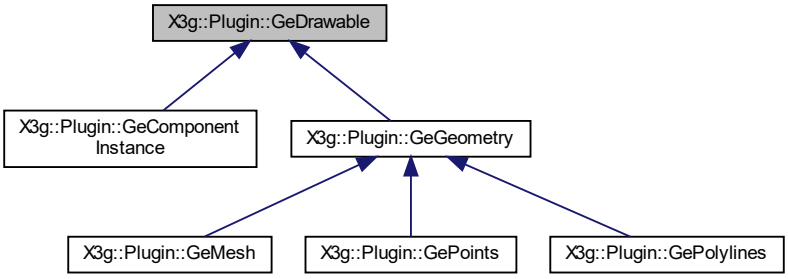
6.78.2.2 GeMatrix ^ X3g::Plugin::GeComponentInstance::Transform

Transformation matrix.

6.79 X3g::Plugin::GeDrawable Class Reference

Root class of all drawables.

Inheritance diagram for X3g::Plugin::GeDrawable:



Public Attributes

- String^ Layer

  Layer name.

6.79.1 Detailed Description

Root class of all drawables.

6.79.2 Member Data Documentation

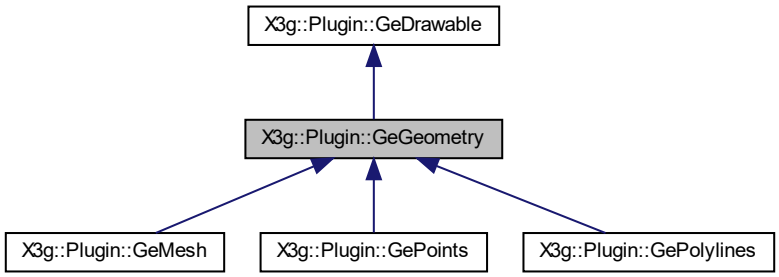
6.79.2.1 String ^ X3g::Plugin::GeDrawable::Layer

Layer name.

6.80 X3g::Plugin::GeGeometry Class Reference

Geometry.

Inheritance diagram for X3g::Plugin::GeGeometry:



Public Attributes

- Color Color

  Simple color.

6.80.1 Detailed Description

Geometry.

6.80.2 Member Data Documentation

6.80.2.1 Color X3g::Plugin::GeGeometry::Color

Simple color.

6.81 X3g::Plugin::GeMatrix Class Reference

Matrix class.

Public Member Functions

- GeMatrix ()

  Creates an identity matrix.
- GeMatrix (double a00, double a01, double a02, double a03, double a10, double a11, double a12, double a13, double a20, double a21, double a22, double a23, double a30, double a31, double a32, double a33)

  Creates a matrix from each component.
- GeMatrix (array< double >^pValues)

  Creates matrix from array with 16 elements.
- GeMatrix (GeMatrix^ other)

  Duplicates a matrix.
- GeMatrix (GeVec3d xAxis, GeVec3d yAxis, GeVec3d zAxis, GeVec3d origin)

  Creates a transformation, based on a coordinate system.
- virtual String^ ToString () override

  Serializes the matrix.
- array< double >^ Get ()

  Return the components of this matrix as double array with 16 elements.
- void Decompose ([Out] GeVec3d% pTranslation,[Out] GeQuat% pRotation,[Out] GeVec3d% pScale,[Out] GeQuat% pScaleOrientation)

  Decomposes this matrix into translation, rotation, scale and scale orientation.
- GeMatrix^ PreMult (GeMatrix^ pOther)

  Performs matrix multiplication pOther∗this.
- GeMatrix^ PostMult (GeMatrix^ pOther)

  Performs matrix multiplication this∗other.
- GeVec2d Transform (GeVec2d v)

  Transforms a vector.
- GeVec2f Transform (GeVec2f v)

  Transforms a vector.
- GeVec3d Transform (GeVec3d v)

  Transforms a vector.
- GeVec3f Transform (GeVec3f v)

Transforms a vector.

- GeVec4d Transform (GeVec4d v)

  Transforms a vector.
- GeVec4f Transform (GeVec4f v)

  Transforms a vector.
- GeVec3d TransformDirection (GeVec3d v)

  Transforms a vector with this matrix by transforming its rotation and scale only.
- GeVec3f TransformDirection (GeVec3f v)

  Transforms a vector with this matrix by transforming its rotation and scale only.
- bool IsIdentity ()

  Returns true if this matrix is the identity matrix.
- virtual bool Equals (System::Object^ pOther) override

  Returns true if this matrix equals another matrix.

Static Public Member Functions

- static GeMatrix^ Translate (double x, double y, double z)

  Creates a translation transformation.
- static GeMatrix^ Translate (GeVec3d v)

  Creates a translation transformation.
- static GeMatrix^ Translate (GeVec3f v)

  Creates a translation transformation.
- static GeMatrix^ Scale (double x, double y, double z)

  Creates a scaling transformation.
- static GeMatrix^ Scale (double v)

  Creates a uniform scaling transformation.
- static GeMatrix^ Scale (GeVec3d v)

  Creates a scaling transformation.
- static GeMatrix^ Scale (GeVec3f v)

  Creates a scaling transformation.
- static GeMatrix^ Rotate (double angle, double x, double y, double z)

  Creates a rotation transformation.
- static GeMatrix^ Rotate (double angle, GeVec3d axis)

  Creates a rotation transformation.
- static GeMatrix^ Rotate (double angle, GeVec3f axis)

  Creates a rotation transformation.
- static GeMatrix^ Rotate (GeQuat q)

  Creates rotation matrix from quaternion.
- static GeMatrix^ Invert (GeMatrix^ pMatrix)

  Invertes a matrix.
- static GeMatrix^ Transpose (GeMatrix^ pMatrix)

  Transposes a matrix.
- static GeMatrix^ NormalMatrix (GeMatrix^ pMatrix)

  Returns the transposed inverse of the linear part.

Static Public Attributes

- static GeMatrix^ Identity = gcnew GeMatrix()

  Identity matrix.

6.81.1 Detailed Description

Matrix class.

6.81.2 Constructor & Destructor Documentation

6.81.2.1 X3g::Plugin::GeMatrix::GeMatrix ( )

Creates an identity matrix.

6.81.2.2 X3g::Plugin::GeMatrix::GeMatrix (double *a00,* double *a01,* double *a02,* double *a03,* double *a10,* double *a11,* double *a12,* double *a13,* double *a20,* double *a21,* double *a22,* double *a23,* double *a30,* double *a31,* double *a32,* double *a33*)

Creates a matrix from each component.

6.81.2.3 X3g::Plugin::GeMatrix::GeMatrix (array< double >^ *pValues*)

Creates matrix from array with 16 elements.

6.81.2.4 X3g::Plugin::GeMatrix::GeMatrix (GeMatrix^ *other*)

Duplicates a matrix.

6.81.2.5 X3g::Plugin::GeMatrix::GeMatrix (GeVec3d *xAxis,* GeVec3d *yAxis,* GeVec3d *zAxis,* GeVec3d *origin*)

Creates a transformation, based on a coordinate system.

6.81.3 Member Function Documentation

6.81.3.1 void X3g::Plugin::GeMatrix::Decompose ([Out] GeVec3d% *pTranslation,* [Out] GeQuat% *pRotation,* [Out] GeVec3d% *pScale,* [Out] GeQuat% *pScaleOrientation*)

Decomposes this matrix into translation, rotation, scale and scale orientation.

6.81.3.2 bool X3g::Plugin::GeMatrix::Equals (System::Object^ *pOther*) [override],[virtual]

Returns true if this matrix equals another matrix.

6.81.3.3 array< double > X3g::Plugin::GeMatrix::Get ( )

Return the components of this matrix as double array with 16 elements.

6.81.3.4 GeMatrix X3g::Plugin::GeMatrix::Invert (GeMatrix^ *pMatrix*) [static]

Invertes a matrix.

6.81.3.5 bool X3g::Plugin::GeMatrix::IsIdentity ( )

Returns true if this matrix is the identity matrix.

6.81.3.6 GeMatrix X3g::Plugin::GeMatrix::NormalMatrix (GeMatrix^ *pMatrix*) [static]

Returns the transposed inverse of the linear part.

Normal matrix may be used to transform normals.

6.81.3.7 GeMatrix X3g::Plugin::GeMatrix::PostMult (GeMatrix^ *pOther*)

Performs matrix multiplication this∗other.

If result is used to transform a vector the order is vector∗this∗other;

6.81.3.8 GeMatrix X3g::Plugin::GeMatrix::PreMult (GeMatrix^ *pOther*)

Performs matrix multiplication pOther∗this.

If result is used to transform a vector the order is vector∗other∗this;

6.81.3.9 GeMatrix X3g::Plugin::GeMatrix::Rotate (double *angle,* double *x,* double *y,* double *z*) [static]

Creates a rotation transformation.

Angle in radians.

6.81.3.10 GeMatrix X3g::Plugin::GeMatrix::Rotate (double *angle,* GeVec3d *axis*) [static]

Creates a rotation transformation.

Angle in radians.

6.81.3.11 GeMatrix X3g::Plugin::GeMatrix::Rotate (double *angle,* GeVec3f *axis*) [static]

Creates a rotation transformation.

Angle in radians.

6.81.3.12 GeMatrix X3g::Plugin::GeMatrix::Rotate (GeQuat *q*) [static]

Creates rotation matrix from quaternion.

6.81.3.13 GeMatrix X3g::Plugin::GeMatrix::Scale (double *x,* double *y,* double *z*) [static]

Creates a scaling transformation.

6.81.3.14 GeMatrix X3g::Plugin::GeMatrix::Scale (double *v*) [static]

Creates a uniform scaling transformation.

6.81.3.15 GeMatrix X3g::Plugin::GeMatrix::Scale (GeVec3d *v*) [static]

Creates a scaling transformation.

6.81.3.16 GeMatrix X3g::Plugin::GeMatrix::Scale (GeVec3f *v*) [static]

Creates a scaling transformation.

6.81.3.17 String X3g::Plugin::GeMatrix::ToString ( ) [override],[virtual]

Serializes the matrix.

6.81.3.18 GeVec2d X3g::Plugin::GeMatrix::Transform (GeVec2d *v*)

Transforms a vector.

6.81.3.19 GeVec2f X3g::Plugin::GeMatrix::Transform (GeVec2f *v*)

Transforms a vector.

6.81.3.20 GeVec3d X3g::Plugin::GeMatrix::Transform (GeVec3d *v*)

Transforms a vector.

6.81.3.21 GeVec3f X3g::Plugin::GeMatrix::Transform (GeVec3f *v*)

Transforms a vector.

6.81.3.22 GeVec4d X3g::Plugin::GeMatrix::Transform (GeVec4d *v*)

Transforms a vector.

6.81.3.23 GeVec4f X3g::Plugin::GeMatrix::Transform (GeVec4f *v*)

Transforms a vector.

6.81.3.24 GeVec3d X3g::Plugin::GeMatrix::TransformDirection (GeVec3d *v*)

Transforms a vector with this matrix by transforming its rotation and scale only.

6.81.3.25 GeVec3f X3g::Plugin::GeMatrix::TransformDirection (GeVec3f *v*)

Transforms a vector with this matrix by transforming its rotation and scale only.

6.81.3.26 GeMatrix X3g::Plugin::GeMatrix::Translate (double *x*, double *y*, double *z*) [static]

Creates a translation transformation.

6.81.3.27 GeMatrix X3g::Plugin::GeMatrix::Translate (GeVec3d *v*) [static]

Creates a translation transformation.

6.81.3.28 GeMatrix X3g::Plugin::GeMatrix::Translate (GeVec3f *v*) [static]

Creates a translation transformation.

6.81.3.29 GeMatrix X3g::Plugin::GeMatrix::Transpose (GeMatrix^ *pMatrix*) [static]

Transposes a matrix.

6.81.4 Member Data Documentation

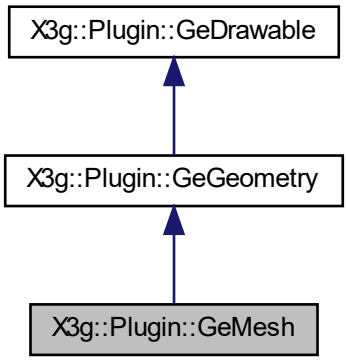
6.81.4.1 GeMatrix ^ X3g::Plugin::GeMatrix::Identity = gcnew GeMatrix() [static]

Identity matrix.

6.82 X3g::Plugin::GeMesh Class Reference

Mesh data for 3d geometry.

Inheritance diagram for X3g::Plugin::GeMesh:



Public Member Functions

- GeBoundingBox3d^ GetBoundingBox (GeMatrix^ pTransform)

  Returns the bounding box of this mesh.

Public Attributes

- array< GeVec3f >^ Vertices

  Vertices.
- array< GeVec3f >^ Normals

  Normals.
- array< GeVec2f >^ TexCoords

  Texture Coordinates.

- array< int >^ EdgeVisibilities

  Edge Visibilities.
- String^ Material

  Optional complex material.
- bool DoubleSided = false

  Mesh faces shall be visible from both sides.

6.82.1 Detailed Description

Mesh data for 3d geometry.

6.82.2 Member Function Documentation

6.82.2.1 GeBoundingBox3d X3g::Plugin::GeMesh::GetBoundingBox (GeMatrix^ *pTransform*)

Returns the bounding box of this mesh.

6.82.3 Member Data Documentation

6.82.3.1 bool X3g::Plugin::GeMesh::DoubleSided = false

Mesh faces shall be visible from both sides.

In rendering this may be achieved by deactivation of backface culling or by face duplication.

6.82.3.2 array<int> ^ X3g::Plugin::GeMesh::EdgeVisibilities

Edge Visibilities.

6.82.3.3 String ^ X3g::Plugin::GeMesh::Material

Optional complex material.

Use MaterialManager::GetMaterial2() to retrieve a material object.

6.82.3.4 array<GeVec3f> ^ X3g::Plugin::GeMesh::Normals

Normals.

6.82.3.5 array<GeVec2f> ^ X3g::Plugin::GeMesh::TexCoords

Texture Coordinates.

6.82.3.6 array<GeVec3f> ^ X3g::Plugin::GeMesh::Vertices

Vertices.

6.83 X3g::Plugin::GeometryManager Class Reference

Geometry Manager.

Public Member Functions

- GeComponent^ GetModelspaceGeometry (GeometryCollectFlags pFlags, View^ pView)

  Retrieves modelspace geometry.
- GeComponent^ GetSeparatedModelspaceGeometry (GeometryCollectFlags pFlags, View^ pView, bool pOnlySelected)

  Retrieves modelspace geometry.
- GeComponent^ GetGeometryOnLayers (IEnumerable< String^ >^pLayerNames, GeometryCollectFlags pFlags, View^ pView)

  Returns the geometries which are on the specified layers.
- IList< DbEntity^ >^ InsertGeometry (GeComponent^ pComponent, bool pInteractive)

  Inserts geometry.
- GeComponent^ GetGeometry (DbEntity^ pEntity, GeometryCollectFlags pFlags)

  Retrieves the geometry of an entity.
- GeComponent^ GetGeometry (DbEntity^ pEntity)

  Retrieves the geometry of an entity.
- IList< GeComponent^ >^ GetGeometrySeparated (IEnumerable< DbEntity^ >^pEntities, GeometryCollectFlags pFlags)

  Retrieves geometry for a collection of modelspace entities.

Exports

- void Export3ds (String^ pFileName, Export3dsParams pParameters)

  Exports the modelspace to a 3ds file.
- void ExportDae (String^ pFileName, ExportDaeParams pParameters)

  Exports the modelspace to a dae file.
- void ExportSkp (String^ pFileName, ExportSkpParams pParameters)

  Exports the modelspace to a skp file.
- void ExportFbx (String^ pFileName, ExportFbxParams pParameters)

  Exports the modelspace to fbx format.
- void ExportObj (String^ pFileName, ExportObjParams pParameters)

  Exports the modelspace to obj format.
- void ExportEgm (Stream^ pStream, ExportEgmParams pParameters)

  Export geometry in egm format.
- void ExportOff (Stream^ pOffStream, Stream^ pVnmStream, ExportOffParams pParameters)

  Export geometry in off format.
- GeComponent^ Export (ExportParams pParameters)

  Exports the modelspace.
- IList< FbxFormatDescription >^ GetFbxFormats ()

  Retrieves a list of available formats supported by X3gExportFbx()

Properties

- double DefaultEdgeSmoothness `[get]`

  Current default edge smoothness of the application.
- double DefaultNormalSmoothness `[get]`

  Current default normal smoothness of the application.

6.83.1 Detailed Description

Geometry Manager.

6.83.2 Member Function Documentation

6.83.2.1 X3g::Plugin::GeComponent X3g::Plugin::GeometryManager::Export (ExportParams *pParameters*)

Exports the modelspace.

6.83.2.2 void X3g::Plugin::GeometryManager::Export3ds (String^ *pFileName,* Export3dsParams *pParameters*)

Exports the modelspace to a 3ds file.

6.83.2.3 void X3g::Plugin::GeometryManager::ExportDae (String^ *pFileName,* ExportDaeParams *pParameters*)

Exports the modelspace to a dae file.

6.83.2.4 void X3g::Plugin::GeometryManager::ExportEgm (Stream^ *pStream,* ExportEgmParams *pParameters*)

Export geometry in egm format.

6.83.2.5 void X3g::Plugin::GeometryManager::ExportFbx (String^ *pFileName,* ExportFbxParams *pParameters*)

Exports the modelspace to fbx format.

6.83.2.6 void X3g::Plugin::GeometryManager::ExportObj (String^ *pFileName,* ExportObjParams *pParameters*)

Exports the modelspace to obj format.

6.83.2.7 void X3g::Plugin::GeometryManager::ExportOff (Stream^ *pOffStream,* Stream^ *pVnmStream,* ExportOffParams *pParameters*)

Export geometry in off format.

6.83.2.8 void X3g::Plugin::GeometryManager::ExportSkp (String^ *pFileName,* ExportSkpParams *pParameters*)

Exports the modelspace to a skp file.

6.83.2.9 IList< X3g::Plugin::FbxFormatDescription > X3g::Plugin::GeometryManager::GetFbxFormats ( )

Retrieves a list of available formats supported by X3gExportFbx()

6.83.2.10 GeComponent X3g::Plugin::GeometryManager::GetGeometry (DbEntity^ *pEntity,* GeometryCollectFlags *pFlags*)

Retrieves the geometry of an entity.

The entity must be a top level entity, i.e. the owning DbBlock must be the modelspace block.

6.83.2.11 GeComponent ^ X3g::Plugin::GeometryManager::GetGeometry (DbEntity^ *pEntity*)

Retrieves the geometry of an entity.

The entity must be a top level entity, i.e. the owning DbBlock must be the modelspace block.

6.83.2.12 X3g::Plugin::GeComponent X3g::Plugin::GeometryManager::GetGeometryOnLayers (IEnumerable< String^ >^ *pLayerNames*, GeometryCollectFlags *pFlags*, View^ *pView*)

Returns the geometries which are on the specified layers.

If a view is given, the layer visibilities for this view will be used.

6.83.2.13 IList< GeComponent^ > X3g::Plugin::GeometryManager::GetGeometrySeparated (IEnumerable< DbEntity^ >^ *pEntities*, GeometryCollectFlags *pFlags*)

Retrieves geometry for a collection of modelspace entities.

The entities must be top level, i.e. the owning block must be the modelspace block. Using this function is more efficient than calling GetGeometry() for each entity separately as identical components in the result are shared.

Returns

Returns list with a geometry component for each entity.

6.83.2.14 X3g::Plugin::GeComponent X3g::Plugin::GeometryManager::GetModelspaceGeometry (GeometryCollectFlags *pFlags*, View^ *pView*)

Retrieves modelspace geometry.

If a view is given, the layer visibilities for this view will be used.

6.83.2.15 X3g::Plugin::GeComponent X3g::Plugin::GeometryManager::GetSeparatedModelspaceGeometry (GeometryCollectFlags *pFlags*, View^ *pView*, bool *pOnlySelected*)

Retrieves modelspace geometry.

In contrast to GetModelspaceGeometry() this function delivers the geometry for each top level entity separately. If a view is given, the layer visibilities for this view will be used.

6.83.2.16 IList< DbEntity^ > X3g::Plugin::GeometryManager::InsertGeometry (GeComponent^ *pComponent*, bool *pInteractive*)

Inserts geometry.

Each geometry is inserted into a component which can be passed to this function. The names of all components are unused. Inserted geometry of components is not shared between different calls to this function. This function returns a list of created modelspace entities.

6.83.3 Property Documentation

6.83.3.1 double X3g::Plugin::GeometryManager::DefaultEdgeSmoothness [get]

Current default edge smoothness of the application.

Minimum angle between two triangles, for an edge to become permanently visible.

6.83.3.2 double X3g::Plugin::GeometryManager::DefaultNormalSmoothness [get]

Current default normal smoothness of the application.

Maximum angle between reference normal and adjacent normal before it is ignored.

6.84 X3g::Plugin::GeometryUpdateEventArgs Class Reference

Event arguments for geometry update of top-level entities.

Inherits EventArgs.

Properties

- DbEntity^ Entity [get]

  Geometry of this entity is updated.
- bool ComponentUpdate [get]

  True if components have been updated.

6.84.1 Detailed Description

Event arguments for geometry update of top-level entities.

6.84.2 Property Documentation

6.84.2.1 bool X3g::Plugin::GeometryUpdateEventArgs::ComponentUpdate [get]

True if components have been updated.

If the entity geometry includes component instances, an event handler may skip the according components, if their state is already known. Components are identified by their name.

See Also

GeComponent

<seealso cref="GeComponentInstance" />

6.84.2.2 DbEntity^ X3g::Plugin::GeometryUpdateEventArgs::Entity [get]

Geometry of this entity is updated.

6.85 X3g::Plugin::GePlane Class Reference

Plane.

Public Member Functions

- GePlane ()

  Constructor.
- GePlane (GeVec3d pNormal, double pDistance)

  Constructor.
- GePlane (GeVec3d pNormal, GeVec3d pPosition)

  Constructor.

Public Attributes

- GeVec3d Normal

  Normal of the plane.

- double Distance

  Distance to origin.

6.85.1 Detailed Description

Plane.

6.85.2 Constructor & Destructor Documentation

6.85.2.1 X3g::Plugin::GePlane::GePlane ( )

Constructor.

6.85.2.2 X3g::Plugin::GePlane::GePlane (GeVec3d *pNormal*, double *pDistance*)

Constructor.

6.85.2.3 X3g::Plugin::GePlane::GePlane (GeVec3d *pNormal*, GeVec3d *pPosition*)

Constructor.

6.85.3 Member Data Documentation

6.85.3.1 double X3g::Plugin::GePlane::Distance

Distance to origin.

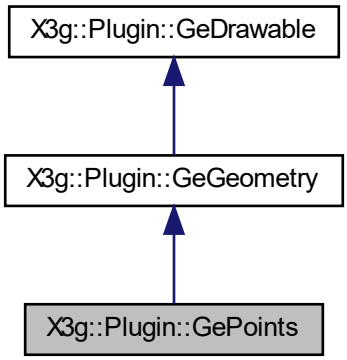
6.85.3.2 GeVec3d X3g::Plugin::GePlane::Normal

Normal of the plane.

6.86 X3g::Plugin::GePoints Class Reference

A set of points.

Inheritance diagram for X3g::Plugin::GePoints:



Public Attributes

- array< GeVec3f >^ Vertices

  Point Coordinates.

- array< GeVec3f >^ Colors

  Point Colors (optional).

6.86.1 Detailed Description

A set of points.

6.86.2 Member Data Documentation

6.86.2.1 array<GeVec3f> ^ X3g::Plugin::GePoints::Colors

Point Colors (optional).

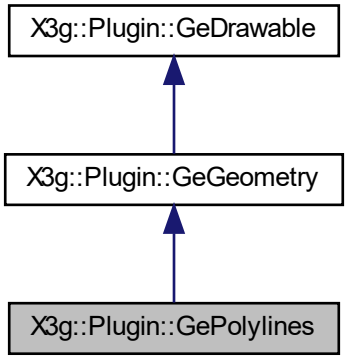
6.86.2.2 array<GeVec3f> ^ X3g::Plugin::GePoints::Vertices

Point Coordinates.

6.87 X3g::Plugin::GePolylines Class Reference

A set of polylines of the same color.

Inheritance diagram for X3g::Plugin::GePolylines:



Public Attributes

- array< GeVec3f >^ Vertices

  Vertices of all polylines.
- array< int >^ Lengths

  Length of each polyline.
- double LineWidth

  Width of all polylines.

6.87.1 Detailed Description

A set of polylines of the same color.

6.87.2 Member Data Documentation

6.87.2.1 array<int> ^ X3g::Plugin::GePolylines::Lengths

Length of each polyline.

6.87.2.2 double X3g::Plugin::GePolylines::LineWidth

Width of all polylines.

6.87.2.3 array<GeVec3f> ^ X3g::Plugin::GePolylines::Vertices

Vertices of all polylines.

6.88 X3g::Plugin::GeQuat Struct Reference

Quaternion.

Public Member Functions

- GeQuat (double x, double y, double z, double w)

  Constructor.
- void GetRotate ([Out] double% angle,[Out] double% x,[Out] double% y,[Out] double% z)

  Returns the rotation.
- GeVec3d GetEuler ()

  Returns rotation as Euler angles in radians.
- GeVec3d GetEulerDeg ()

  Returns rotation as Euler angles in degrees.

Static Public Member Functions

- static GeQuat Rotate (double angle, double x, double y, double z)

  Creates a rotation quaternion from angle and axis.
- static GeQuat Rotate (double angle, GeVec3d axis)

  Creates a rotation quaternion.
- static GeQuat Rotate (double angle, GeVec3f axis)

  Creates a rotation quaternion.
- static GeQuat Euler (double angleX, double angleY, double angleZ)

  Creates a rotation quaternion from Euler angles in radians.
- static GeQuat Euler (GeVec3d angles)

  Creates a rotation quaternion from Euler angles in radians.
- static GeQuat EulerDeg (double angleX, double angleY, double angleZ)

  Creates a rotation quaternion from Euler angles in degrees.
- static GeQuat EulerDeg (GeVec3d angles)

  Creates a rotation quaternion from Euler angles in degrees.

Public Attributes

- double x

  X-component.
- double y

  Y-component.
- double z

  Z-component.
- double w

  W-component.

Static Public Attributes

- static GeQuat Identity = GeQuat(0, 0, 0, 1)

  Identity quaternion.

6.88.1 Detailed Description

Quaternion.

Please note that the default constructor produces a null quaternion which doesn't describe a valid rotation. If you need a quaternion with zero rotation you may use GeQuat::Identity.

6.88.2 Constructor & Destructor Documentation

6.88.2.1 X3g::Plugin::GeQuat::GeQuat (double *x*, double *y*, double *z*, double *w*)

Constructor.

6.88.3 Member Function Documentation

6.88.3.1 GeQuat X3g::Plugin::GeQuat::Euler (double *angleX*, double *angleY*, double *angleZ*) [static]

Creates a rotation quaternion from Euler angles in radians.

Rotation order is XYZ.

6.88.3.2 GeQuat X3g::Plugin::GeQuat::Euler (GeVec3d *angles*) [static]

Creates a rotation quaternion from Euler angles in radians.

Rotation order is XYZ.

6.88.3.3 GeQuat X3g::Plugin::GeQuat::EulerDeg (double *angleX*, double *angleY*, double *angleZ*) [static]

Creates a rotation quaternion from Euler angles in degrees.

Rotation order is XYZ.

6.88.3.4 GeQuat X3g::Plugin::GeQuat::EulerDeg (GeVec3d *angles*) [static]

Creates a rotation quaternion from Euler angles in degrees.

Rotation order is XYZ.

6.88.3.5 GeVec3d X3g::Plugin::GeQuat::GetEuler ( )

Returns rotation as Euler angles in radians.

Rotation order is XYZ.

6.88.3.6 GeVec3d X3g::Plugin::GeQuat::GetEulerDeg ( )

Returns rotation as Euler angles in degrees.

Rotation order is XYZ.

6.88.3.7 void X3g::Plugin::GeQuat::GetRotate ([Out] double% *angle*, [Out] double% *x*, [Out] double% *y*, [Out] double% *z*)

Returns the rotation.

Angle in radians.

6.88.3.8 GeQuat X3g::Plugin::GeQuat::Rotate (double *angle*, double *x*, double *y*, double *z*) [static]

Creates a rotation quaternion from angle and axis.

Angle in radians.

6.88.3.9 GeQuat X3g::Plugin::GeQuat::Rotate (double *angle*, GeVec3d *axis*) [static]

Creates a rotation quaternion.

Angle in radians.

6.88.3.10 GeQuat X3g::Plugin::GeQuat::Rotate (double *angle*, GeVec3f *axis*) [static]

Creates a rotation quaternion.

Angle in radians.

6.88.4 Member Data Documentation

6.88.4.1 GeQuat X3g::Plugin::GeQuat::Identity = GeQuat(0, 0, 0, 1) [static]

Identity quaternion.

Describes zero rotation.

6.88.4.2 double X3g::Plugin::GeQuat::w

W-component.

6.88.4.3 double X3g::Plugin::GeQuat::x

X-component.

6.88.4.4 double X3g::Plugin::GeQuat::y

Y-component.

6.88.4.5 double X3g::Plugin::GeQuat::z

Z-component.

6.89 X3g::Plugin::GeVec2d Struct Reference

A two dimensional float vector or point.

Public Member Functions

- GeVec2d (double x, double y)

  Constructor.
- GeVec2d Normalized ()

  Returns a new vector with normalized length.
- bool IsSame (GeVec2d other, double tol)

  Returns true if this vector equals another vector using a tolerance.
- GeVec2f ToVec2f ()

  Converts this vector to GeVec2f.

Public Attributes

- double x

  X coordinate.
- double y

  Y coordinate.

Properties

- double Length `[get]`

  Returns the length of the vector.
- double Length2 `[get]`

  Returns the squared length of the vector.

6.89.1 Detailed Description

A two dimensional float vector or point.

6.89.2 Constructor & Destructor Documentation

6.89.2.1 X3g::Plugin::GeVec2d::GeVec2d (double *x*, double *y*)

Constructor.

6.89.3 Member Function Documentation

6.89.3.1 bool X3g::Plugin::GeVec2d::IsSame (GeVec2d *other*, double *tol*)

Returns true if this vector equals another vector using a tolerance.

6.89.3.2 GeVec2d X3g::Plugin::GeVec2d::Normalized ( )

Returns a new vector with normalized length.

6.89.3.3 GeVec2f X3g::Plugin::GeVec2d::ToVec2f ( )

Converts this vector to GeVec2f.

6.89.4 Member Data Documentation

6.89.4.1 double X3g::Plugin::GeVec2d::x

X coordinate.

6.89.4.2 double X3g::Plugin::GeVec2d::y

Y coordinate.

6.89.5 Property Documentation

6.89.5.1 double X3g::Plugin::GeVec2d::Length [get]

Returns the length of the vector.

6.89.5.2 double X3g::Plugin::GeVec2d::Length2 [get]

Returns the squared length of the vector.

6.90 X3g::Plugin::GeVec2f Struct Reference

A two dimensional float vector or point.

Public Member Functions

- GeVec2f (float x, float y)

  Constructor.
- GeVec2f Normalized ()

  Returns a new vector with normalized length.
- bool IsSame (GeVec2f other, float tol)

  Returns true if this vector equals another vector using a tolerance.
- GeVec2d ToVec2d ()

  Converts this vector to GeVec2d.

Public Attributes

- float x

  X coordinate.
- float y

  Y coordinate.

Properties

- float Length [get]

  Returns the length of the vector.
- float Length2 [get]

  Returns the squared length of the vector.

6.90.1 Detailed Description

A two dimensional float vector or point.

6.90.2 Constructor & Destructor Documentation

6.90.2.1 X3g::Plugin::GeVec2f::GeVec2f (float *x*, float *y*)

Constructor.

6.90.3 Member Function Documentation

6.90.3.1 bool X3g::Plugin::GeVec2f::IsSame (GeVec2f *other*, float *tol*)

Returns true if this vector equals another vector using a tolerance.

6.90.3.2 GeVec2f X3g::Plugin::GeVec2f::Normalized ( )

Returns a new vector with normalized length.

6.90.3.3 GeVec2d X3g::Plugin::GeVec2f::ToVec2d ( )

Converts this vector to GeVec2d.

6.90.4 Member Data Documentation

6.90.4.1 float X3g::Plugin::GeVec2f::x

X coordinate.

6.90.4.2 float X3g::Plugin::GeVec2f::y

Y coordinate.

6.90.5 Property Documentation

6.90.5.1 float X3g::Plugin::GeVec2f::Length [get]

Returns the length of the vector.

6.90.5.2 float X3g::Plugin::GeVec2f::Length2 [get]

Returns the squared length of the vector.

6.91 X3g::Plugin::GeVec3d Struct Reference

A three dimensional double vector or point.

Public Member Functions

- GeVec3d (double x, double y, double z)

  Constructor.
- GeVec3d (double v)

  Initializes x, y and z with same value.
- GeVec3d Normalized ()

  Returns a new vector with normalized length.
- double DotProduct (GeVec3d v)

  Returns the dot product of this vector with another vector.
- GeVec3d CrossProduct (GeVec3d v)

Returns the cross product of this vector with another vector.

- bool IsSame (GeVec3d other, double tol)

  Returns true if this vector equals another vector using a tolerance.
- GeVec3f ToVec3f ()

  Converts this vector to GeVec3f.

Public Attributes

- double x

  X coordinate.
- double y

  Y coordinate.
- double z

  Z coordinate.

Properties

- double Length [get]

  Returns the length of the vector.
- double Length2 [get]

  Returns the squared length of the vector.

6.91.1 Detailed Description

A three dimensional double vector or point.

6.91.2 Constructor & Destructor Documentation

6.91.2.1 X3g::Plugin::GeVec3d::GeVec3d (double *x*, double *y*, double *z*)

Constructor.

6.91.2.2 X3g::Plugin::GeVec3d::GeVec3d (double *v*)

Initializes x, y and z with same value.

6.91.3 Member Function Documentation

6.91.3.1 GeVec3d X3g::Plugin::GeVec3d::CrossProduct (GeVec3d *v*)

Returns the cross product of this vector with another vector.

6.91.3.2 double X3g::Plugin::GeVec3d::DotProduct (GeVec3d *v*)

Returns the dot product of this vector with another vector.

6.91.3.3 bool X3g::Plugin::GeVec3d::IsSame (GeVec3d *other*, double *tol*)

Returns true if this vector equals another vector using a tolerance.

6.91.3.4 GeVec3d X3g::Plugin::GeVec3d::Normalized ( )

Returns a new vector with normalized length.

6.91.3.5 GeVec3f X3g::Plugin::GeVec3d::ToVec3f ( )

Converts this vector to GeVec3f.

6.91.4 Member Data Documentation

6.91.4.1 double X3g::Plugin::GeVec3d::x

X coordinate.

6.91.4.2 double X3g::Plugin::GeVec3d::y

Y coordinate.

6.91.4.3 double X3g::Plugin::GeVec3d::z

Z coordinate.

6.91.5 Property Documentation

6.91.5.1 double X3g::Plugin::GeVec3d::Length [get]

Returns the length of the vector.

6.91.5.2 double X3g::Plugin::GeVec3d::Length2 [get]

Returns the squared length of the vector.

6.92 X3g::Plugin::GeVec3f Struct Reference

A three dimensional float vector or point.

Public Member Functions

- GeVec3f (float x, float y, float z)

  Constructor.
- GeVec3f (float v)

  Initializes x, y and z with same value.
- GeVec3f Normalized ()

  Returns a new vector with normalized length.
- float DotProduct (GeVec3f v)

  Returns the dot product of this vector with another vector.
- GeVec3f CrossProduct (GeVec3f v)

  Returns the cross product of this vector with another vector.
- bool IsSame (GeVec3f other, float tol)

Returns true if this vector equals another vector using a tolerance.

- GeVec3d ToVec3d ()

  Converts this vector to GeVec3d.

Public Attributes

- float x

  X coordinate.
- float y

  Y coordinate.
- float z

  Z coordinate.

Properties

- float Length `[get]`

  Returns the length of the vector.
- float Length2 `[get]`

  Returns the squared length of the vector.

6.92.1 Detailed Description

A three dimensional float vector or point.

6.92.2 Constructor & Destructor Documentation

6.92.2.1 X3g::Plugin::GeVec3f::GeVec3f (float *x*, float *y*, float *z*)

Constructor.

6.92.2.2 X3g::Plugin::GeVec3f::GeVec3f (float *v*)

Initializes x, y and z with same value.

6.92.3 Member Function Documentation

6.92.3.1 GeVec3f X3g::Plugin::GeVec3f::CrossProduct (GeVec3f *v*)

Returns the cross product of this vector with another vector.

6.92.3.2 float X3g::Plugin::GeVec3f::DotProduct (GeVec3f *v*)

Returns the dot product of this vector with another vector.

6.92.3.3 bool X3g::Plugin::GeVec3f::IsSame (GeVec3f *other*, float *tol*)

Returns true if this vector equals another vector using a tolerance.

6.92.3.4 GeVec3f X3g::Plugin::GeVec3f::Normalized ( )

Returns a new vector with normalized length.

6.92.3.5 GeVec3d X3g::Plugin::GeVec3f::ToVec3d ( )

Converts this vector to GeVec3d.

6.92.4 Member Data Documentation

6.92.4.1 float X3g::Plugin::GeVec3f::x

X coordinate.

6.92.4.2 float X3g::Plugin::GeVec3f::y

Y coordinate.

6.92.4.3 float X3g::Plugin::GeVec3f::z

Z coordinate.

6.92.5 Property Documentation

6.92.5.1 float X3g::Plugin::GeVec3f::Length [get]

Returns the length of the vector.

6.92.5.2 float X3g::Plugin::GeVec3f::Length2 [get]

Returns the squared length of the vector.

6.93 X3g::Plugin::GeVec4d Struct Reference

A four dimensional double vector or point.

Public Member Functions

- GeVec4d (double x, double y, double z, double w)

  Constructor.
- bool IsSame (GeVec4d other, double tol)

  Returns true if this vector equals another vector using a tolerance.
- GeVec4f ToVec4f ()

  Converts this vector to GeVec4f.

Public Attributes

- double x

  X coordinate.
- double y

  Y coordinate.
- double z

  Z coordinate.
- double w

  W coordinate.

6.93.1 Detailed Description

A four dimensional double vector or point.

6.93.2 Constructor & Destructor Documentation

6.93.2.1 X3g::Plugin::GeVec4d::GeVec4d (double *x*, double *y*, double *z*, double *w*)

Constructor.

6.93.3 Member Function Documentation

6.93.3.1 bool X3g::Plugin::GeVec4d::IsSame (GeVec4d *other*, double *tol*)

Returns true if this vector equals another vector using a tolerance.

6.93.3.2 GeVec4f X3g::Plugin::GeVec4d::ToVec4f ( )

Converts this vector to GeVec4f.

6.93.4 Member Data Documentation

6.93.4.1 double X3g::Plugin::GeVec4d::w

W coordinate.

6.93.4.2 double X3g::Plugin::GeVec4d::x

X coordinate.

6.93.4.3 double X3g::Plugin::GeVec4d::y

Y coordinate.

6.93.4.4 double X3g::Plugin::GeVec4d::z

Z coordinate.

6.94 X3g::Plugin::GeVec4f Struct Reference

A four dimensional float vector or point.

Public Member Functions

- GeVec4f (float x, float y, float z, float w)

  Constructor.
- bool IsSame (GeVec4f other, float tol)

  Returns true if this vector equals another vector using a tolerance.
- GeVec4d ToVec4d ()

  Converts this vector to GeVec4d.

Public Attributes

- float x

  X coordinate.
- float y

  Y coordinate.
- float z

  Z coordinate.
- float w

  W coordinate.

6.94.1 Detailed Description

A four dimensional float vector or point.

6.94.2 Constructor & Destructor Documentation

6.94.2.1 X3g::Plugin::GeVec4f::GeVec4f (float *x*, float *y*, float *z*, float *w*)

Constructor.

6.94.3 Member Function Documentation

6.94.3.1 bool X3g::Plugin::GeVec4f::IsSame (GeVec4f *other*, float *tol*)

Returns true if this vector equals another vector using a tolerance.

6.94.3.2 GeVec4d X3g::Plugin::GeVec4f::ToVec4d ( )

Converts this vector to GeVec4d.

6.94.4 Member Data Documentation

6.94.4.1 float X3g::Plugin::GeVec4f::w

W coordinate.

6.94.4.2 float X3g::Plugin::GeVec4f::x

X coordinate.

6.94.4.3 float X3g::Plugin::GeVec4f::y

Y coordinate.

6.94.4.4 float X3g::Plugin::GeVec4f::z

Z coordinate.

6.95 X3g::Plugin::Hyperlink Struct Reference

Interfaces an ACAD Hyperlink.

Public Attributes

- String^ Name

  A filename or URL.
- String^ Description

  A display name for the Hyperlink.
- String^ Sublocation

  A named view, range of cells, etc.

6.95.1 Detailed Description

Interfaces an ACAD Hyperlink.

A Hyperlink in ACAD consists of:

Hyperlink Name: A filename or URL.

Sublocation: A named view, range of cells, etc.

Description: A display name for the Hyperlink.

Example:

Name: `http://foo.com`, Sublocation: bar -> `http://foo.com#bar`

6.95.2 Member Data Documentation

6.95.2.1 String ^ X3g::Plugin::Hyperlink::Description

A display name for the Hyperlink.

6.95.2.2 String ^ X3g::Plugin::Hyperlink::Name

A filename or URL.

6.95.2.3 String ^ X3g::Plugin::Hyperlink::Sublocation

A named view, range of cells, etc.

6.96 X3g::Plugin::IActionManager Class Reference

Action Manager.

Public Member Functions

- virtual void RegisterAction (String^ pActionKey)=0

  Registers an action with given action key.
- virtual void UnregisterAction (String^ pActionKey)=0

  Unregisters an action via action key.
- virtual void RegisterTool (String^ pToolKey, ITool^ pTool)=0

  Registers a tool.
- virtual void UnregisterTool (String^ pToolKey)=0

  Unregisters a tool.
- bool ExecuteAction (String^ pKey, String^ pArguments)

  Executes an action or a tool with given optional arguments.
- void PlaceEntities (IEnumerable< DbEntity^ >^pEntities, bool pRotate)

  Starts tool which allows user to place given top-level entities.

6.96.1 Detailed Description

Action Manager.

6.96.2 Member Function Documentation

6.96.2.1 bool X3g::Plugin::IActionManager::ExecuteAction (String^ *pKey,* String^ *pArguments*)

Executes an action or a tool with given optional arguments.

6.96.2.2 void X3g::Plugin::IActionManager::PlaceEntities (IEnumerable< DbEntity^ >^ *pEntities,* bool *pRotate*)

Starts tool which allows user to place given top-level entities.

Entities are erased if user aborts the placement.

Parameters

pRotate	Enables rotation when positioning is done.

6.96.2.3 virtual void X3g::Plugin::IActionManager::RegisterAction (String^ *pActionKey*) [pure virtual]

Registers an action with given action key.

An action results in behavior provided by the given plugin. The key must be unique within all registered plugin actions. Each time the action is executed, X3gAction() gets called. A registered action can also be triggered by a gui element provided in the manifest.

6.96.2.4 virtual void X3g::Plugin::IActionManager::RegisterTool (String^ *pToolKey,* ITool^ *pTool*) [pure virtual]

Registers a tool.

6.96.2.5 virtual void X3g::Plugin::IActionManager::UnregisterAction (String^ *pActionKey*) [pure virtual]

Unregisters an action via action key.

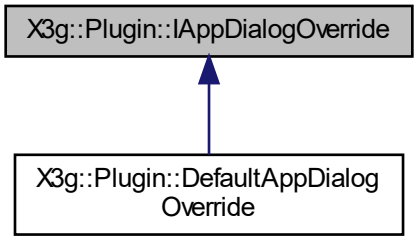
6.96.2.6 virtual void X3g::Plugin::IActionManager::UnregisterTool (String^ *pToolKey*) [pure virtual]

Unregisters a tool.

6.97 X3g::Plugin::IAppDialogOverride Interface Reference

Allows to override application dialogs.

Inheritance diagram for X3g::Plugin::IAppDialogOverride:



Public Member Functions

- bool GetInsertScale (const GeBoundingBox3d^ pBounds,[In, Out] LengthUnit% pUnit,[Out] double% pCustomScale)

  Called if planning unit is undefined on import.
- bool RequestArticleConversion (ArticleConversionType pType,[Out] bool% pAllowConversion)

  Called if an action would imply the conversion of an article entity.
- bool ContinueImportWithLayerVisibilityChange ([Out] bool% pContinue)

  Called on import if all imported entities are on invisible layers.

6.97.1 Detailed Description

Allows to override application dialogs.

6.97.2 Member Function Documentation

6.97.2.1 bool X3g::Plugin::IAppDialogOverride::ContinueImportWithLayerVisibilityChange ([Out] bool% *pContinue*)

Called on import if all imported entities are on invisible layers.

Parameters

pContinue	Returns true if import should be continued. Layers will be enabled in this case.

Returns

Return true to override application dialog.

Implemented in X3g::Plugin::DefaultAppDialogOverride.

6.97.2.2 bool X3g::Plugin::IAppDialogOverride::GetInsertScale (const GeBoundingBox3d^ *pBounds*, [In, Out] LengthUnit% *pUnit*, [Out] double% *pCustomScale*)

Called if planning unit is undefined on import.

Parameters

pBounds	Bounding box of inserted planning.
pUnit	Length unit of inserted planning.
pCustomScale	Scale in addition to unit scale.

Returns

Return true to override application dialog.

Implemented in X3g::Plugin::DefaultAppDialogOverride.

6.97.2.3 bool X3g::Plugin::IAppDialogOverride::RequestArticleConversion (ArticleConversionType *pType*, [Out] bool% *pAllowConversion*)

Called if an action would imply the conversion of an article entity.

Parameters

pType	Kind of conversion.
pAllow-Conversion	Returns true if conversion is allowed. Action is aborted otherwise.

Returns

Return true to override application dialog.

Implemented in X3g::Plugin::DefaultAppDialogOverride.

6.98 X3g::Plugin::IAppManager Class Reference

Application Manager.

Public Member Functions

- void SetStatusMessage (String^ pStatusMessage)

  Sets the applications status message.
- void PopUpNotification (String^ pMessage)

  Shows a pop-up notification in lower right corner.
- unsigned int StartProgress (String^ pMessage, int pLimit, bool pAbortable)

Starts a new progess and returns its id.

- bool SetProgress (unsigned int pProgressId, int pProgress)

  Sets the progess for the given progress id.
- void EndProgress (unsigned int pProgressId)

  Ends a progess.
- void BeginShowOperationDelay ()

  Notifies the application about the beginning of a longer-lasting operation.
- void EndShowOperationDelay ()

  Notifies the application about the end of a longer-lasting operation.
- bool IsFeatureEnabled (String^ pFeatureKey)

  Returns true if a feature is available or a license can be locked (is already locked).
- String^ GetFeatureInfo (String^ pFeatureKey)

  Returns a JSON string containing more information to the specified feature/license (pCon.login licensing system).
- Windows::Forms::IWin32Window^ GetMainWindow ()

  Returns the applications main window which can be used as a parent for other windows.
- bool Terminate (AppTerminateFlags pFlags)

  Request to terminate the application.
- void BringWindowToTop ()

  Brings application window to the top.

Properties

- IAppDialogOverride^ DialogOverride

  Replace the default handler to override application dialogs.
- Windows::Forms::FormWindowState WindowState `[get, set]`

  Window state of application main window.
- bool WindowVisibility `[get, set]`

  Visibility of application main window.

Events

- System::EventHandler^ FeaturesChanged `[add, remove, raise]`

  Raised when feature states changed and when features were added or removed.

6.98.1 Detailed Description

Application Manager.

6.98.2 Member Function Documentation

6.98.2.1 void X3g::Plugin::IAppManager::BeginShowOperationDelay ( )

Notifies the application about the beginning of a longer-lasting operation.

While it takes place the user has to wait. The application may lock the user interface and show some kind of wait-cursor.

6.98.2.2 void X3g::Plugin::IAppManager::BringWindowToTop ( )

Brings application window to the top.

6.98.2.3 void X3g::Plugin::IAppManager::EndProgress (unsigned int *pProgressId*)

Ends a progess.

6.98.2.4 void X3g::Plugin::IAppManager::EndShowOperationDelay ( )

Notifies the application about the end of a longer-lasting operation.

6.98.2.5 String X3g::Plugin::IAppManager::GetFeatureInfo (String^ *pFeatureKey*)

Returns a JSON string containing more information to the specified feature/license (pCon.login licensing system).

The license has to be obtained before (see IAppManager::IsFeatureEnabled).

Parameters

pFeatureKey	Formatted feature key. Supported schemas are: Query application or aplication feature: "license.el.<AppID>[:<FeatureID>]" Query pCon.planner feature: "license.el:<FeatureID>"

<AppID> - pCon.login Application ID

<FeatureID> - pCon.login Feature ID

6.98.2.6 System::Windows::Forms::IWin32Window X3g::Plugin::IAppManager::GetMainWindow ( )

Returns the applications main window which can be used as a parent for other windows.

6.98.2.7 bool X3g::Plugin::IAppManager::IsFeatureEnabled (String^ *pFeatureKey*)

Returns true if a feature is available or a license can be locked (is already locked).

A license will be locked if necessary. It is released when the application is shut down.

Parameters

pFeatureKey	Formatted feature key. Supported schemas are: DLM system: "license.dlm:<Key>:<Value>" Safenet licensing: "license.safenet:<ProductNumber>:<FeatureNumber>" pCon.login licensing: "license.el<AppID>[:<FeatureID>]"

A pCon.login license should first be locked without a FeatureID. A query can fail if not logged in. Just a query with no FeatureID prompts the user to sign if needed.

Multiple features of a license should queried using IAppManager::GetFeatureInfo.

IsFeatureEnabled() is very dynamic - a feature is not guaranteed to always be available or remain available once it has been. (see FeaturesChanged event). Do not rely on IsFeatureEnabled during X3gInitialize!

6.98.2.8 void X3g::Plugin::IAppManager::PopUpNotification (String^ *pMessage*)

Shows a pop-up notification in lower right corner.

6.98.2.9 bool X3g::Plugin::IAppManager::SetProgress (unsigned int *pProgressId*, int *pProgress*)

Sets the progess for the given progress id.

This function returns false if the operation belonging to the current progress shall be aborted.

6.98.2.10 void X3g::Plugin::IAppManager::SetStatusMessage (String^ *pStatusMessage*)

Sets the applications status message.

6.98.2.11 unsigned int X3g::Plugin::IAppManager::StartProgress (String^ *pMessage,* int *pLimit,* bool *pAbortable*)

Starts a new progess and returns its id.

Parameters

pMessage	A description for the progress.
pLimit	Maximum number of progress steps. Set to -1 if the number of steps is undetermined. SetProgress() should not be called in this case.
pAbortable	Tells if the progress is abortable.

6.98.2.12 bool X3g::Plugin::IAppManager::Terminate (AppTerminateFlags *pFlags*)

Request to terminate the application.

Returns true if the application will terminate.

6.98.3 Property Documentation

6.98.3.1 IAppDialogOverride^ X3g::Plugin::IAppManager::DialogOverride

Replace the default handler to override application dialogs.

6.98.3.2 Windows:: Forms:: FormWindowState X3g::Plugin::IAppManager::WindowState [get],[set]

Window state of application main window.

6.98.3.3 bool X3g::Plugin::IAppManager::WindowVisibility [get],[set]

Visibility of application main window.

6.98.4 Event Documentation

6.98.4.1 System:: EventHandler^ X3g::Plugin::IAppManager::FeaturesChanged [add],[remove],[raise]

Raised when feature states changed and when features were added or removed.

6.99 X3g::Plugin::IArticleEntity Class Reference

Article interface for entities.

Public Member Functions

- IArticleEntityInfo^ GetInformation ()

Receives additional article information.

- bool IsUpToDate ()

  Returns true if the article is based on the currently installed product data (OFML).
- ReadOnlyCollection< String^ >^ GetChildArticles ()

  Receives a list of child article ids.

Properties

- String^ Id [get]

  The unqiue id of the entity.
- ArticleEntityType Type [get]

  The article type
- ArticleGeometryMode GeometryMode [get]

  Geometry mode of an article entity describes if the geometric representation of an article is intended for 2d or 3d planning.

6.99.1 Detailed Description

Article interface for entities.

6.99.2 Member Function Documentation

6.99.2.1 ReadOnlyCollection<String^> ^ X3g::Plugin::IArticleEntity::GetChildArticles ( )

Receives a list of child article ids.

6.99.2.2 IArticleEntityInfo ^ X3g::Plugin::IArticleEntity::GetInformation ( )

Receives additional article information.

6.99.2.3 bool X3g::Plugin::IArticleEntity::IsUpToDate ( )

Returns true if the article is based on the currently installed product data (OFML).

Otherwise it is unknown if the article is up-to-date or not.

6.99.3 Property Documentation

6.99.3.1 ArticleGeometryMode X3g::Plugin::IArticleEntity::GeometryMode [get]

Geometry mode of an article entity describes if the geometric representation of an article is intended for 2d or 3d planning.

6.99.3.2 String^ X3g::Plugin::IArticleEntity::Id [get]

The unqiue id of the entity.

Relates to DbEntity::Id; The DbEntity returned by DbEntity::FromId() for this id can be cast to DbBlockReference.

6.99.3.3 ArticleEntityType X3g::Plugin::IArticleEntity::Type [get]

The article type

6.100 X3g::Plugin::IArticleEntityInfo Class Reference

Holds additional article information.

Properties

- String^ ManufacturerId [get]

  The article's manufacturer id.
- String^ SeriesId [get]

  The article's series id.
- String^ ArticleNumber [get]

  The article's number.
- String^ VariantCode [get]

  The article's variant code.
- String^ ShortText [get]

  The article's short text.
- String^ ReferenceNumber [get]

  The article's reference number.
- String^ FolderId [get]

  The id of the folder the article belongs to.
- String^ BasketId [get]

  A globally unique id for the article.

6.100.1 Detailed Description

Holds additional article information.

6.100.2 Property Documentation

6.100.2.1 String^ X3g::Plugin::IArticleEntityInfo::ArticleNumber [get]

The article's number.

6.100.2.2 String^ X3g::Plugin::IArticleEntityInfo::BasketId [get]

A globally unique id for the article.

Used to identify the article inside the basket.

6.100.2.3 String^ X3g::Plugin::IArticleEntityInfo::FolderId [get]

The id of the folder the article belongs to.

6.100.2.4 String^ X3g::Plugin::IArticleEntityInfo::ManufacturerId [get]

The article's manufacturer id.

6.100.2.5 String^ X3g::Plugin::IArticleEntityInfo::ReferenceNumber [get]

The article's reference number.

6.100.2.6 String^ X3g::Plugin::IArticleEntityInfo::SeriesId [get]

The article's series id.

6.100.2.7 String^ X3g::Plugin::IArticleEntityInfo::ShortText [get]

The article's short text.

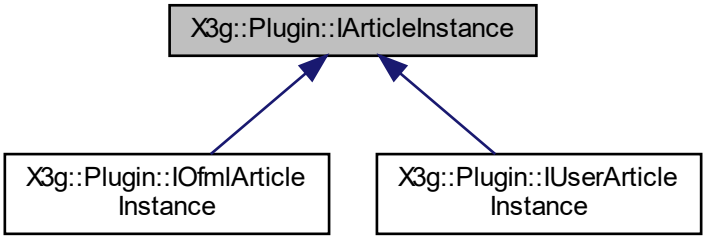
6.100.2.8 String^ X3g::Plugin::IArticleEntityInfo::VariantCode [get]

The article's variant code.

6.101 X3g::Plugin::IArticleInstance Class Reference

Article Instance.

Inheritance diagram for X3g::Plugin::IArticleInstance:



Public Member Functions

- IBasketItem^ getArticleInformation (bool pPrepareArticleImages)

  Returns a copy of the article information of the currently selected entity.
- bool SetArticleTextField (String^ pTextFieldId, String^ pLang, String^ pText)

  Sets a text field value of the currently selected article.
- void Close ()

  Closes the instance and applies all changes.

Properties

- bool IsValid [get]

  Returns true if this instance is still valid.
- String^ MainEntityId [get]

  The main entity id of this instance (see DbEntity::Id).
- String^ EntityId [get]

  The currently selected entity id of this instance (see DbEntity::Id).
- IPropertyProvider^ PropertyProvider [get]

The property provider of the currently selected entity.

- ReadOnlyCollection < System::String^ >^ AvailableLanguages [get]

  A list of potentially available languages (two letter ISO codes) for this instance.

- bool IsAlternatePosition [get, set]

  Allows to tag the article as an alternate position.

6.101.1 Detailed Description

Article Instance.

6.101.2 Member Function Documentation

6.101.2.1 void X3g::Plugin::IArticleInstance::Close ( )

Closes the instance and applies all changes.

An instance will be invalid after it was closed. The instance gets automatically closed if another article gets instantiated or if it is deselected.

6.101.2.2 X3g::Plugin::IBasketItem X3g::Plugin::IArticleInstance::getArticleInformation (bool *pPrepareArticleImages*)

Returns a copy of the article information of the currently selected entity.

If the article has changed, this function needs to be called again.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

6.101.2.3 bool X3g::Plugin::IArticleInstance::SetArticleTextField (String^ *pTextFieldId,* String^ *pLang,* String^ *pText*)

Sets a text field value of the currently selected article.

Use IBasketArticleItem::TextFields to get the available text fields.

Parameters

pTextFieldId	The id of the text field (see IBasketArticleTextField::Id).
pLang	The language of the new text value. If it is null or empty the application language will be used.
pText	The new text value.

Exceptions

System::Exception	An exception will be thrown if this instance is no longer valid.

6.101.3 Property Documentation

6.101.3.1 ReadOnlyCollection< System:: String^>^ X3g::Plugin::IArticleInstance::AvailableLanguages [get]

A list of potentially available languages (two letter ISO codes) for this instance.

6.101.3.2 String^ X3g::Plugin::IArticleInstance::EntityId [get]

The currently selected entity id of this instance (see DbEntity::Id).

6.101.3.3 bool X3g::Plugin::IArticleInstance::IsAlternatePosition [get],[set]

Allows to tag the article as an alternate position.

Alternate positions are excluded from total price calculation.

6.101.3.4 bool X3g::Plugin::IArticleInstance::IsValid [get]

Returns true if this instance is still valid.

An instance will be invalid after it was closed. (see IArticleInstance::Close()) At any given time there can be only one valid instance.

6.101.3.5 String^ X3g::Plugin::IArticleInstance::MainEntityId [get]

The main entity id of this instance (see DbEntity::Id).

6.101.3.6 IPropertyProvider^ X3g::Plugin::IArticleInstance::PropertyProvider [get]

The property provider of the currently selected entity.

6.102 X3g::Plugin::IArticleManager Class Reference

Article Manager.

Public Member Functions

- void PrepareArticleEntities ()

  Prepares all article entities in the document.
- bool IsArticle (String^ pEntityId)

  Checks if the given id belongs to an article.
- IArticleEntity^ GetArticleEntity (String^ pEntityId)

  Retrieves the article behind id 'pEntityId' (see DbEntity::Id).
- IArticleEntity^ InsertArticle (String^ pManufacturerId, String^ pSeriesId, String^ pBaseArticleNumber, String^ pVariantCode, bool pKeepInstanced)

  Inserts a new article.
- IArticleEntity^ InsertArticle (String^ pManufacturerId, String^ pSeriesId, String^ pBaseArticleNumber, String^ pVariantCode, bool pKeepInstanced, ArticleGeometryMode pGeoMode)

  Inserts a new article.
- Articles::ArticleEntity^ InsertArticleByPackage (String^ pCatalogPkgName, String^ pArticlePkgName, String^ pArticleNumber, String^ pVariantCode, bool pKeepInstanced, ArticleGeometryMode pGeoMode)

  Inserts an article from given package.
- IArticleEntity^ CreateUserArticle (String^ pEntityId)

  Creates an user article from the given entity/geometry.
- IArticleEntity^ CreateUserArticle (DbBlockReference^ pEntity)

  Creates an user article from a given block reference.
- bool ConvertToUserArticle (IArticleEntity^ pArticle)

  Converts the given article to a user article.
- ReadOnlyCollection< String^ >^ SplitUpArticle (IArticleEntity^ pArticle)

  Splits-up a configuration or an article with childs into separate articles.
- bool UpdateArticleToCurrentPD (IArticleEntity^ pArticle)

Updates the given article and all its sub-articles to the currently installed product data (OFML).

- bool UpdateArticleToCurrentPD (IArticleEntity^ pArticle, DateTime pPriceDate)

  Updates the given article and all its sub-articles to the currently installed product data (OFML).
- IArticleInstance^ InstantiateArticle (IArticleEntity^ pArticle, ReadOnlyCollection< String^ >^pAdditionalLang)

  Intantiates the given article and selects it to give access to its properties.
- IFolderManager^ GetFolderManager ()

  Returns the folder manager.
- IProjectSettings^ GetProjectSettings ()

  Returns the project settings.
- IList< String^ >^ GetEffectiveArticleLanguages ()

  Returns effective product data languages as two-letter country codes.
- IList< String^ >^ GetEffectiveCatalogLanguages ()

  Returns effective catalog languages as two-letter country codes.
- IBasket^ GenerateBasket (BasketGenerationFlags pFlags, ReadOnlyCollection< String^ >^pAdditionalLang)

  Generates the basket view.
- virtual bool ExportBasketToObx (BasketGenerationFlags pFlags, System::String^ pFilename)

  Generates the basket view and exports it to an obx file.
- bool ExportBasketToObk (BasketGenerationFlags pFlags, System::String^ pFilename)

  Generates the basket view and exports it to an obk file.
- bool ExportBasketToCsv (BasketGenerationFlags pFlags, System::String^ pFilename)

  Generates the basket view and exports it to an csv file.

Properties

- IArticleInstance^ CurrentInstance `[get]`

  Current instanced article.

Events

- Articles::ArticleInsertedEventHandler^ ArticleInserted `[add, remove, raise]`

  Occurs when article was inserted from catalog.
- Articles::AssignDefaultPropValuesHandler^ AssignDefaultPropValues `[add, remove, raise]`

  Allows to initialize ofml articles with default property values before insertion.

6.102.1 Detailed Description

Article Manager.

6.102.2 Member Function Documentation

6.102.2.1 bool X3g::Plugin::IArticleManager::ConvertToUserArticle (IArticleEntity^ *pArticle*)

Converts the given article to a user article.

Returns true if the conversion was successful.

The article must not be a configuration or has child articles. IArticleManager::SplitUpArticle() can be used to split-up those articles.

6.102.2.2 IArticleEntity ^ X3g::Plugin::IArticleManager::CreateUserArticle (String^ *pEntityId*)

Creates an user article from the given entity/geometry.

The entity must not be part of an article or include articles. The entity will be cloned to a block and a new block reference is created for the user article. The original entity is removed.

6.102.2.3 IArticleEntity X3g::Plugin::IArticleManager::CreateUserArticle (DbBlockReference^ *pEntity*)

Creates an user article from a given block reference.

The block reference must not be part of an article or include articles. The returned article entity will attach it's data directly to the given block reference. No additional block or block reference is created. The given block reference stays valid.

6.102.2.4 bool X3g::Plugin::IArticleManager::ExportBasketToCsv (BasketGenerationFlags *pFlags*, System::String^ *pFilename*)

Generates the basket view and exports it to an csv file.

Returns true if the export was successful.

The flags PrepareCatalogImages and PrepareGeneratedImages will be ignored.

6.102.2.5 bool X3g::Plugin::IArticleManager::ExportBasketToObk (BasketGenerationFlags *pFlags*, System::String^ *pFilename*)

Generates the basket view and exports it to an obk file.

Returns true if the export was successful.

The flags PrepareCatalogImages and PrepareGeneratedImages will be ignored.

6.102.2.6 bool X3g::Plugin::IArticleManager::ExportBasketToObx (BasketGenerationFlags *pFlags*, System::String^ *pFilename*) [virtual]

Generates the basket view and exports it to an obx file.

Returns true if the export was successful.

The flags PrepareCatalogImages and PrepareGeneratedImages will be ignored.

6.102.2.7 X3g::Plugin::IBasket X3g::Plugin::IArticleManager::GenerateBasket (BasketGenerationFlags *pFlags*, ReadOnlyCollection< String^ >^ *pAdditionalLang*)

Generates the basket view.

The generated basket represents a static article list. If changes are made to the document, the basket needs to be generated again. IBasket provides the article information for different languages. The availalbe languages depend on the current application settings. The pAdditionalLang parameter can be used to request additional languages. Any open ModelSpace block will be closed and set to be the current selection.

Parameters

pAdditionalLang	List of additional two letter ISO language keys.

6.102.2.8 X3g::Plugin::IArticleEntity X3g::Plugin::IArticleManager::GetArticleEntity (String^ *pEntityId*)

Retrieves the article behind id 'pEntityId' (see DbEntity::Id).

PrepareArticleEntities() needs to be called once before using this function. Otherwise, the funcion may return null or an article with an invalid BasketId.

6.102.2.9 IList< String^ > X3g::Plugin::IArticleManager::GetEffectiveArticleLanguages ( )

Returns effective product data languages as two-letter country codes.

Primary language at index 0.

6.102.2.10 IList< String^ > X3g::Plugin::IArticleManager::GetEffectiveCatalogLanguages ( )

Returns effective catalog languages as two-letter country codes.

Primary language at index 0.

6.102.2.11 X3g::Plugin::IFolderManager X3g::Plugin::IArticleManager::GetFolderManager ( )

Returns the folder manager.

6.102.2.12 X3g::Plugin::IProjectSettings X3g::Plugin::IArticleManager::GetProjectSettings ( )

Returns the project settings.

6.102.2.13 IArticleEntity ^ X3g::Plugin::IArticleManager::InsertArticle (String^ *pManufacturerId,* String^ *pSeriesId,* String^ *pBaseArticleNumber,* String^ *pVariantCode,* bool *pKeepInstanced*)

Inserts a new article.

If pKeepInstanced is true the article will stay instantiated. This will speed-up a subsequent call to IArticleManager::InstantiateArticle() for the inserted article. Geometry mode depends on user setting in pCon.planner.

6.102.2.14 IArticleEntity X3g::Plugin::IArticleManager::InsertArticle (String^ *pManufacturerId,* String^ *pSeriesId,* String^ *pBaseArticleNumber,* String^ *pVariantCode,* bool *pKeepInstanced,* ArticleGeometryMode *pGeoMode*)

Inserts a new article.

If pKeepInstanced is true the article will stay instantiated. This will speed-up a subsequent call to IArticleManager::InstantiateArticle() for the inserted article. Allows to specify the geometry mode.

6.102.2.15 Articles::ArticleEntity X3g::Plugin::IArticleManager::InsertArticleByPackage (String^ *pCatalogPkgName,* String^ *pArticlePkgName,* String^ *pArticleNumber,* String^ *pVariantCode,* bool *pKeepInstanced,* ArticleGeometryMode *pGeoMode*)

Inserts an article from given package.

Parameters

pCatalogPkg-Name	Name of catalog package which references the article.
pArticlePkg-Name	Name of package which contains the article data.

pArticleNumber	The article number.
pVariantCode	The variant code.
pKeepInstanced	Article stays instanced for configuration.
pGeoMode	Set ArticleGeometryMode::Undefined to apply user setting.

6.102.2.16 X3g::Plugin::IArticleInstance X3g::Plugin::IArticleManager::InstantiateArticle (IArticleEntity^ *pArticle,* ReadOnlyCollection< String^ >^ *pAdditionalLang*)

Intantiates the given article and selects it to give access to its properties.

IArticleInstance provides the article information for different languages. The availalbe languages depend on the current application settings. The pAdditionalLang parameter can be used to request additional languages.

Parameters

pAdditionalLang	List of additional two letter ISO language keys.

Exceptions

System::Exception	An exception will be thrown if the instantiation failed.

6.102.2.17 bool X3g::Plugin::IArticleManager::IsArticle (String^ *pEntityId*)

Checks if the given id belongs to an article.

6.102.2.18 void X3g::Plugin::IArticleManager::PrepareArticleEntities ( )

Prepares all article entities in the document.

This function ensures that each article resides in an unique block hirarchie and has a unique BasketId. (IArticle-Entity::ArticleInformation::BasketId)

This function needs to be called everytime before using GetArticleEntity() if the block structure of the document may have changed. (e.g. an article was grouped or grouped articles were copied)

The block structure may have changed after calling this function. Therefore, it should be called before iterating the hirarchie (DbBlock) of the document.

6.102.2.19 ReadOnlyCollection< System::String^ > X3g::Plugin::IArticleManager::SplitUpArticle (IArticleEntity^ *pArticle*)

Splits-up a configuration or an article with childs into separate articles.

Returns the ids of the separated articles.

6.102.2.20 bool X3g::Plugin::IArticleManager::UpdateArticleToCurrentPD (IArticleEntity^ *pArticle*)

Updates the given article and all its sub-articles to the currently installed product data (OFML).

Keeps the saved price date. Returns true if the update was successful.

6.102.2.21 bool X3g::Plugin::IArticleManager::UpdateArticleToCurrentPD (IArticleEntity^ *pArticle,* DateTime *pPriceDate*)

Updates the given article and all its sub-articles to the currently installed product data (OFML).

Updates prices to given price date. Returns true if the update was successful.

6.102.3 Property Documentation

6.102.3.1 IArticleInstance^ X3g::Plugin::IArticleManager::CurrentInstance [get]

Current instanced article.

6.102.4 Event Documentation

6.102.4.1 Articles:: ArticleInsertedEventHandler^ X3g::Plugin::IArticleManager::ArticleInserted [add], [remove], [raise]

Occurs when article was inserted from catalog.

6.102.4.2 Articles:: AssignDefaultPropValuesHandler^ X3g::Plugin::IArticleManager::AssignDefaultPropValues [add], [remove], [raise]

Allows to initialize ofml articles with default property values before insertion.

Handlers should return true if they assigned any property values.

6.103 X3g::Plugin::IBasket Class Reference

Basket interface.

Properties

- IBasketItem^ RootFolder [get]

  The root folder.
- IBasketCalculation^ Calculation [get]

  The basket's calculation.
- ReadOnlyCollection< String^ >^ AvailableLanguages [get]

  A list of potentially available languages (two letter ISO codes) for this basket.

6.103.1 Detailed Description

Basket interface.

6.103.2 Property Documentation

6.103.2.1 ReadOnlyCollection< String^ >^ X3g::Plugin::IBasket::AvailableLanguages [get]

A list of potentially available languages (two letter ISO codes) for this basket.

Use IArticleManager::GenerateBasket to request additional languages. Not all items are supporting all languages. For article items the supported language depends on the product data. e.g. IBasketItem::GetLabel() may return null even if the requested language is returned here.

6.103.2.2 IBasketCalculation^ X3g::Plugin::IBasket::Calculation [get]

The basket's calculation.

6.103.2.3 IBasketItem^ X3g::Plugin::IBasket::RootFolder [get]

The root folder.

6.104 X3g::Plugin::IBasketArticleCalculation Class Reference

Basket Article Calculation interface.

Properties

- System::Nullable< System::Decimal > Quantity [get]

  The article's quantity.
- IBasketCalcItem^ VAT [get]

  The VAT (value added tax).
- IBasketCalcItem^ ArticleAddCharge [get]

  The article add charge.
- IBasketCalcItem^ ArticleDiscount [get]

  The article discount.
- IBasketCalcItem^ PurchasePricePD [get]

  The purchase prices defined in the product data.
- IBasketCalcItem^ PurchasePrice [get]

  The purchase prices.
- IBasketCalcItem^ SalesPricePD [get]

  The sales prices defined in the product data.
- IBasketCalcItem^ SalesPrice [get]

  The sales prices.
- IBasketCalcItem^ NetPrice [get]

  The net price.
- IBasketCalcItem^ GrossPrice [get]

  The gross price.
- IBasketCalcItem^ Margin [get]

  The margin.
- ReadOnlyCollection
  < IBasketArticlePriceComponent^ >^ SalesPriceComponents [get]

  A list of sales price components.

6.104.1 Detailed Description

Basket Article Calculation interface.

6.104.2 Property Documentation

6.104.2.1 IBasketCalcItem^ X3g::Plugin::IBasketArticleCalculation::ArticleAddCharge [get]

The article add charge.

Relates to IProjectSettings::OverallArticleAddCharge

6.104.2.2 IBasketCalcItem^ X3g::Plugin::IBasketArticleCalculation::ArticleDiscount [get]

The article discount.

Relates to IProjectSettings::OverallArticleDiscount

6.104.2.3 IBasketCalcItem^ X3g::Plugin::IBasketArticleCalculation::GrossPrice [get]

The gross price.

This is the net price plus VAT.

6.104.2.4 IBasketCalcItem^ X3g::Plugin::IBasketArticleCalculation::Margin [get]

The margin.

This is the difference between the NetPrice and the PurchasePrice.

6.104.2.5 IBasketCalcItem^ X3g::Plugin::IBasketArticleCalculation::NetPrice [get]

The net price.

6.104.2.6 IBasketCalcItem^ X3g::Plugin::IBasketArticleCalculation::PurchasePrice [get]

The purchase prices.

This is the purchase prices obtained after applying possibly available price profiles to the purchase price stored in the product data.

6.104.2.7 IBasketCalcItem^ X3g::Plugin::IBasketArticleCalculation::PurchasePricePD [get]

The purchase prices defined in the product data.

6.104.2.8 System:: Nullable< System:: Decimal> X3g::Plugin::IBasketArticleCalculation::Quantity [get]

The article's quantity.

6.104.2.9 IBasketCalcItem^ X3g::Plugin::IBasketArticleCalculation::SalesPrice [get]

The sales prices.

This is the sales prices obtained after applying possibly available price profiles to the sales price stored in the product data.

6.104.2.10 ReadOnlyCollection< IBasketArticlePriceComponent^>^ X3g::Plugin::IBasketArticleCalculation::SalesPriceComponents [get]

A list of sales price components.

6.104.2.11 IBasketCalcItem^ X3g::Plugin::IBasketArticleCalculation::SalesPricePD [get]

The sales prices defined in the product data.

6.104.2.12 IBasketCalcItem^ X3g::Plugin::IBasketArticleCalculation::VAT [get]

The VAT (value added tax).

Relates to IProjectSettings::VAT

6.105 X3g::Plugin::IBasketArticleFeature Class Reference

Basket Article Feature

Properties

- String^ NameKey [get]

  Key of the feature name.
- String^ NameText [get]

  Text of the feature name.
- String^ ValueKey [get]

  Key of the feature value.
- String^ ValueText [get]

  Text of the feature value.
- bool Visible [get]

  Defines if the feature is visible or not.

6.105.1 Detailed Description

Basket Article Feature

6.105.2 Property Documentation

6.105.2.1 String^ X3g::Plugin::IBasketArticleFeature::NameKey [get]

Key of the feature name.

6.105.2.2 String^ X3g::Plugin::IBasketArticleFeature::NameText [get]

Text of the feature name.

6.105.2.3 String^ X3g::Plugin::IBasketArticleFeature::ValueKey [get]

Key of the feature value.

6.105.2.4 String^ X3g::Plugin::IBasketArticleFeature::ValueText [get]

Text of the feature value.

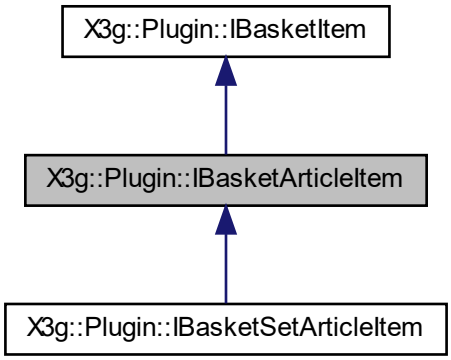
6.105.2.5 bool X3g::Plugin::IBasketArticleFeature::Visible [get]

Defines if the feature is visible or not.

6.106 X3g::Plugin::IBasketArticleItem Class Reference

Basket Article Item.

Inheritance diagram for X3g::Plugin::IBasketArticleItem:



Public Member Functions

- System::String^ GetManufacturerName (System::String^ pLanguage)

  Returns the manufacturer name in the given language.
- System::String^ GetSeriesName (System::String^ pLanguage)

  Returns the series name in the given language.
- System::String^ GetArticleNumber (BasketArtNumType pType)

  Returns the article number.
- System::String^ GetDescription (BasketArtDescrType pType, System::String^ pLanguage)

  Returns the article description in the given language.
- ReadOnlyCollection < IBasketArticleFeature^ >^ GetFeatures (System::String^ pLanguage)

  Returns a list of article features in the given language.

Properties

- System::String^ ManufacturerId `[get]`

  The article's manufacturer id.
- System::String^ SeriesId `[get]`

  The article's series id.
- System::String^ CatalogImage `[get]`

  URI to the article's catalog image.
- System::String^ GeneratedImage `[get]`

  URI to the article's generated image.
- System::String^ ArticleImage `[get]`

  Returnes an URI to the default article image.
- ReadOnlyCollection < IBasketArticleTextField^ >^ TextFields `[get]`

A list of optional text fields.

- IBasketArticleCalculation^ Calculation [get]

  The article's calculation.

- ReadOnlyCollection< IBasketItem^ >^ SubArticles [get]

  The sub-articles belonging to this article.

- bool IsAlternatePosition [get]

  Returns if article item is an alternate position.

6.106.1 Detailed Description

Basket Article Item.

6.106.2 Member Function Documentation

6.106.2.1 System::String X3g::Plugin::IBasketArticleItem::GetArticleNumber (BasketArtNumType *pType*)

Returns the article number.

6.106.2.2 System::String X3g::Plugin::IBasketArticleItem::GetDescription (BasketArtDescrType *pType*, System::String^ *pLanguage*)

Returns the article description in the given language.

"" can be used as a fallback language. A fallback description is returned in this case.

6.106.2.3 ReadOnlyCollection< X3g::Plugin::IBasketArticleFeature^ > X3g::Plugin::IBasketArticleItem::GetFeatures (System::String^ *pLanguage*)

Returns a list of article features in the given language.

"" can be used as a fallback language.

6.106.2.4 System::String X3g::Plugin::IBasketArticleItem::GetManufacturerName (System::String^ *pLanguage*)

Returns the manufacturer name in the given language.

"" can be used as a fallback language. A fallback name is returned in this case.

6.106.2.5 System::String X3g::Plugin::IBasketArticleItem::GetSeriesName (System::String^ *pLanguage*)

Returns the series name in the given language.

"" can be used as a fallback language. A fallback name is returned in this case.

6.106.3 Property Documentation

6.106.3.1 System:: String^ X3g::Plugin::IBasketArticleItem::ArticleImage [get]

Returnes an URI to the default article image.

Depending on the product data settings and the used BasketGenerationFlags the CatalogImage or the GeneratedImage will be returned.

6.106.3.2 IBasketArticleCalculation^ X3g::Plugin::IBasketArticleItem::Calculation [get]

The article's calculation.

6.106.3.3 System:: String^ X3g::Plugin::IBasketArticleItem::CatalogImage [get]

URI to the article's catalog image.

6.106.3.4 System:: String^ X3g::Plugin::IBasketArticleItem::GeneratedImage [get]

URI to the article's generated image.

6.106.3.5 bool X3g::Plugin::IBasketArticleItem::IsAlternatePosition [get]

Returns if article item is an alternate position.

Alternate positions are excluded from total price calculation.

6.106.3.6 System:: String^ X3g::Plugin::IBasketArticleItem::ManufacturerId [get]

The article's manufacturer id.

6.106.3.7 System:: String^ X3g::Plugin::IBasketArticleItem::SeriesId [get]

The article's series id.

6.106.3.8 ReadOnlyCollection< IBasketItem^>^ X3g::Plugin::IBasketArticleItem::SubArticles [get]

The sub-articles belonging to this article.

The returned articles may not be real childs (hierarchically) of this article.

6.106.3.9 ReadOnlyCollection< IBasketArticleTextField^>^ X3g::Plugin::IBasketArticleItem::TextFields [get]

A list of optional text fields.

6.107 X3g::Plugin::IBasketArticlePriceComponent Class Reference

Basket Article Price Component.

Properties

- IBasketCalcItem^ Price

  The price.
- System::String^ Name

  The name of the price component.

6.107.1 Detailed Description

Basket Article Price Component.

6.107.2 Property Documentation

6.107.2.1 System:: String^ X3g::Plugin::IBasketArticlePriceComponent::Name

The name of the price component.

6.107.2.2 IBasketCalcItem^ X3g::Plugin::IBasketArticlePriceComponent::Price

The price.

6.108 X3g::Plugin::IBasketArticleTextField Class Reference

A basket article text field.

Public Member Functions

- String^ GetValue (String^ pLanguage)

  Returns the value in the given language.

Properties

- String^ Id [get]

  The textfield's id.
- String^ Name [get]

  The textfield's name.
- BasketTextFieldType Type [get]

  The textfield's type.
- bool ReadOnly [get]

  True if this text field is read-only.

6.108.1 Detailed Description

A basket article text field.

6.108.2 Member Function Documentation

6.108.2.1 String ^ X3g::Plugin::IBasketArticleTextField::GetValue (String^ *pLanguage*)

Returns the value in the given language.

"" can be used as a fallback language.

6.108.3 Property Documentation

6.108.3.1 String^ X3g::Plugin::IBasketArticleTextField::Id [get]

The textfield's id.

6.108.3.2 String^ X3g::Plugin::IBasketArticleTextField::Name [get]

The textfield's name.

6.108.3.3 bool X3g::Plugin::IBasketArticleTextField::ReadOnly [get]

True if this text field is read-only.

Use IArticleInstance::SetArticleTextField() to change non-readonly text fields.

6.108.3.4 BasketTextFieldType X3g::Plugin::IBasketArticleTextField::Type [get]

The textfield's type.

6.109 X3g::Plugin::IBasketCalcItem Class Reference

BasketCalcItem.

Properties

- Currency^ Relative

  The item's relative value.
- Currency^ Single

  The item's single value.
- Currency^ Total

  The item's total value.

6.109.1 Detailed Description

BasketCalcItem.

6.109.2 Property Documentation

6.109.2.1 Currency^ X3g::Plugin::IBasketCalcItem::Relative

The item's relative value.

This may be null if the item has no relative value;

6.109.2.2 Currency^ X3g::Plugin::IBasketCalcItem::Single

The item's single value.

This may be null if the item has no single value;

6.109.2.3 Currency^ X3g::Plugin::IBasketCalcItem::Total

The item's total value.

This may be null if the item has no Total value;

6.110 X3g::Plugin::IBasketCalculation Class Reference

Basket Calculation.

Properties

- System::String^ CurrencyUnit [get]

  The active currency unit.
- IBasketCalcItem^ VAT [get]

  The VAT (value added tax).
- IBasketCalcItem^ HeaderAddCharge [get]

  The header add charge.
- IBasketCalcItem^ HeaderDiscount [get]

  The header discount.
- IBasketCalcItem^ NetDiscount [get]

  The net discount.
- Currency^ NetArticleTotal [get]

  The total article net price.
- Currency^ PurchasePrice [get]

  The sum of the purchase prices of all basket articles.
- Currency^ SalesPrice [get]

  The sum of the sales prices of all basket articles.
- Currency^ NetPrice [get]

  The net price.
- Currency^ GrossPrice [get]

  The gross price.
- IBasketCalcItem^ Margin [get]

  The margin.

6.110.1 Detailed Description

Basket Calculation.

6.110.2 Property Documentation

6.110.2.1 System:: String^ X3g::Plugin::IBasketCalculation::CurrencyUnit [get]

The active currency unit.

Relates to IProjectSettings::CurrencyUnit

6.110.2.2 Currency^ X3g::Plugin::IBasketCalculation::GrossPrice [get]

The gross price.

This is the net price plus VAT.

6.110.2.3 IBasketCalcItem^ X3g::Plugin::IBasketCalculation::HeaderAddCharge [get]

The header add charge.

Relates to IProjectSettings::HeaderAddCharge

6.110.2.4 IBasketCalcItem^ X3g::Plugin::IBasketCalculation::HeaderDiscount [get]

The header discount.

Relates to IProjectSettings::HeaderDiscount

6.110.2.5 IBasketCalcItem^ X3g::Plugin::IBasketCalculation::Margin [get]

The margin.

This is the difference between the NetPrice and the PurchasePrice.

6.110.2.6 Currency^ X3g::Plugin::IBasketCalculation::NetArticleTotal [get]

The total article net price.

This is the net price sum of all positions.

6.110.2.7 IBasketCalcItem^ X3g::Plugin::IBasketCalculation::NetDiscount [get]

The net discount.

This is the discount on the net price sum of all positions.

6.110.2.8 Currency^ X3g::Plugin::IBasketCalculation::NetPrice [get]

The net price.

This is the net price sum of all articles minus the basket net discount.

6.110.2.9 Currency^ X3g::Plugin::IBasketCalculation::PurchasePrice [get]

The sum of the purchase prices of all basket articles.

This is the purchase prices obtained after applying possibly available price profiles to the purchase price stored in the product data.

6.110.2.10 Currency^ X3g::Plugin::IBasketCalculation::SalesPrice [get]

The sum of the sales prices of all basket articles.

This is the sales prices obtained after applying possibly available price profiles to the sales price stored in the product data.

6.110.2.11 IBasketCalcItem^ X3g::Plugin::IBasketCalculation::VAT [get]

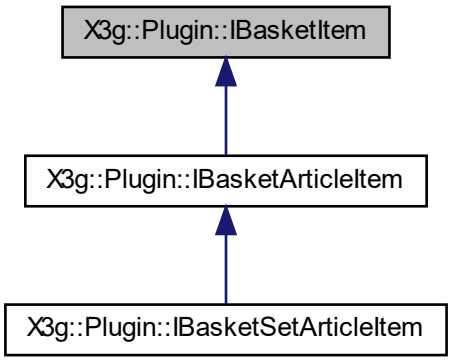
The VAT (value added tax).

Relates to IProjectSettings::VAT

6.111 X3g::Plugin::IBasketItem Class Reference

Basket Item.

Inheritance diagram for X3g::Plugin::IBasketItem:



Public Member Functions

- System::String^ GetLabel (System::String^ pLanguage)

  Returns the item's label in the given language.

Properties

- System::String^ Id `[get]`

  The item id.
- BasketItemType Type `[get]`

  The item type
- System::String^ ReferenceNumber `[get]`

  The item's reference number.
- System::String^ Position `[get]`

  The item's position inside the basket.
- IBasketItem^ Parent `[get]`

  The item's parent.
- ReadOnlyCollection< IBasketItem^ >^ Children `[get]`

  The item's children.

6.111.1 Detailed Description

Basket Item.

6.111.2 Member Function Documentation

6.111.2.1 System::String X3g::Plugin::IBasketItem::GetLabel (System::String^ *pLanguage*)

Returns the item's label in the given language.

"" can be used as a fallback language. A fallback name is returned in this case.

6.111.3 Property Documentation

6.111.3.1 ReadOnlyCollection< IBasketItem^>^ X3g::Plugin::IBasketItem::Children [get]

The item's children.

6.111.3.2 System:: String^ X3g::Plugin::IBasketItem::Id [get]

The item id.

If the item is an article the id relates to IArticleEntityInfo::BasketId. If the item is a folder the id relates to IFMItem::Id.

6.111.3.3 IBasketItem^ X3g::Plugin::IBasketItem::Parent [get]

The item's parent.

6.111.3.4 System:: String^ X3g::Plugin::IBasketItem::Position [get]

The item's position inside the basket.

6.111.3.5 System:: String^ X3g::Plugin::IBasketItem::ReferenceNumber [get]

The item's reference number.

If the item is an article the number relates to IArticleEntityInfo::ReferenceNumber. If the item is a folder the number relates to IFMItem::ReferenceNumber.

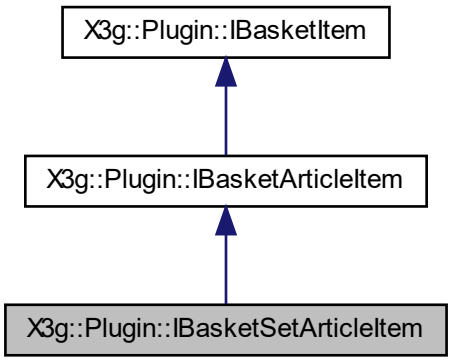
6.111.3.6 BasketItemType X3g::Plugin::IBasketItem::Type [get]

The item type

6.112 X3g::Plugin::IBasketSetArticleItem Class Reference

Basket Set Article Item.

Inheritance diagram for X3g::Plugin::IBasketSetArticleItem:



Properties

- bool IsCollapsed [get]

  Returns if set article item should list it's children.

Additional Inherited Members

6.112.1 Detailed Description

Basket Set Article Item.

6.112.2 Property Documentation

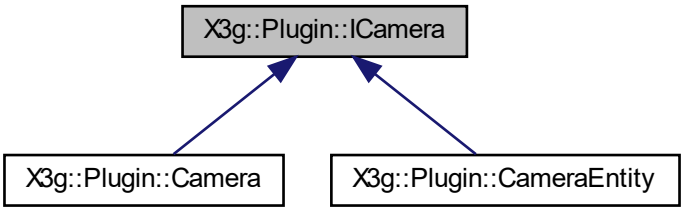
6.112.2.1 bool X3g::Plugin::IBasketSetArticleItem::IsCollapsed [get]

Returns if set article item should list it's children.

6.113 X3g::Plugin::ICamera Interface Reference

Common camera interface.

Inheritance diagram for X3g::Plugin::ICamera:



Public Member Functions

- void GetLookAt ([Out] GeVec3d% pEye,[Out] GeVec3d% pCenter,[Out] GeVec3d% pUp)

  Returns the camera postion and the camera target.
- void SetLookAt (GeVec3d pEye, GeVec3d pCenter, GeVec3d pUp)

  Sets the camera postion and the camera target.
- void ZoomToExtents (ReadOnlyCollection< String^ >^pEntities, double pMargin, double pAspect)

  Fits camera to given entities.
- void ZoomToExtents (IEnumerable< DbEntity^ >^pEntities, double pMargin, double pAspect)

  Fits camera to given entities.
- void CopyFrom (ICamera^ pOther)

  Assigns values of another camera object to this.

Properties

- System::String^ Name

  Optional name of the camera.
- CameraProjection Projection

  The type of camera projection.
- double Fov

  The fov (field of view) of perspective camera in degrees.
- double YMag

  Y magnification of orthographic camera.
- bool DofEnabled

  True if depth of field is enabled.
- double DofDistance

  The depth of field distance.
- double DofAperture

  The depth of field aperture/strength.

6.113.1 Detailed Description

Common camera interface.

6.113.2 Member Function Documentation

6.113.2.1 void X3g::Plugin::ICamera::CopyFrom (ICamera^ *pOther*)

Assigns values of another camera object to this.

Implemented in X3g::Plugin::Camera, and X3g::Plugin::CameraEntity.

6.113.2.2 void X3g::Plugin::ICamera::GetLookAt ([Out] GeVec3d% *pEye,* [Out] GeVec3d% *pCenter,* [Out] GeVec3d% *pUp*)

Returns the camera postion and the camera target.

Implemented in X3g::Plugin::Camera, and X3g::Plugin::CameraEntity.

6.113.2.3 void X3g::Plugin::ICamera::SetLookAt (GeVec3d *pEye,* GeVec3d *pCenter,* GeVec3d *pUp*)

Sets the camera postion and the camera target.

Implemented in X3g::Plugin::Camera, and X3g::Plugin::CameraEntity.

6.113.2.4 void X3g::Plugin::ICamera::ZoomToExtents (ReadOnlyCollection< String^ >^ *pEntities,* double *pMargin,* double *pAspect*)

Fits camera to given entities.

6.113.2.5 void X3g::Plugin::ICamera::ZoomToExtents (IEnumerable< DbEntity^ >^ *pEntities,* double *pMargin,* double *pAspect*)

Fits camera to given entities.

Implemented in X3g::Plugin::Camera, and X3g::Plugin::CameraEntity.

6.113.3 Property Documentation

6.113.3.1 double X3g::Plugin::ICamera::DofAperture

The depth of field aperture/strength.

6.113.3.2 double X3g::Plugin::ICamera::DofDistance

The depth of field distance.

6.113.3.3 bool X3g::Plugin::ICamera::DofEnabled

True if depth of field is enabled.

6.113.3.4 double X3g::Plugin::ICamera::Fov

The fov (field of view) of perspective camera in degrees.

6.113.3.5 System:: String^ X3g::Plugin::ICamera::Name

Optional name of the camera.

May be empty and not unique.

6.113.3.6 CameraProjection X3g::Plugin::ICamera::Projection

The type of camera projection.

6.113.3.7 double X3g::Plugin::ICamera::YMag

Y magnification of orthographic camera.

6.114 X3g::Plugin::ICommand Interface Reference

Base interface for user defined commands

Public Member Functions

- void Undo ()

  Undo operation.
- void Redo ()

  Redo operation.

6.114.1 Detailed Description

Base interface for user defined commands

6.114.2 Member Function Documentation

6.114.2.1 void X3g::Plugin::ICommand::Redo ( )

Redo operation.

6.114.2.2 void X3g::Plugin::ICommand::Undo ( )

Undo operation.

6.115 X3g::Plugin::ICustomTextField Class Reference

Legacy class.

6.115.1 Detailed Description

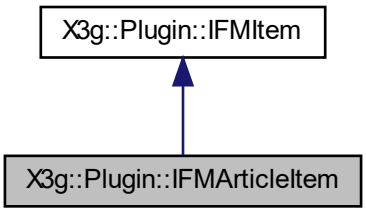
Legacy class.

See Articles::CustomTextField.

6.116 X3g::Plugin::IFMArticleItem Class Reference

Folder Manager Article-Item.

Inheritance diagram for X3g::Plugin::IFMArticleItem:



Properties

- System::String^ ArticleNumber [get]

  The article's number.
- System::Nullable< Decimal > Quantity [get]

  The article's quantity.

Additional Inherited Members

6.116.1 Detailed Description

Folder Manager Article-Item.

6.116.2 Property Documentation

6.116.2.1 System:: String^ X3g::Plugin::IFMArticleItem::ArticleNumber [get]

The article's number.

Relates to IArticleEntityInfo::ArticleNumber.

6.116.2.2 System:: Nullable< Decimal> X3g::Plugin::IFMArticleItem::Quantity [get]

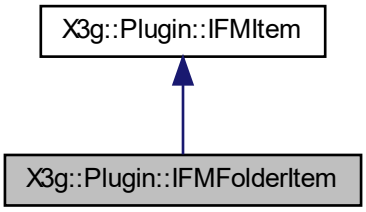
The article's quantity.

One article item may represent multiple articles (e.g. in the case of reference copies).

6.117 X3g::Plugin::IFMFolderItem Class Reference

Folder Manager Folder-Item.

Inheritance diagram for X3g::Plugin::IFMFolderItem:



Public Member Functions

- void SetLabel (System::String^ pLabel)
  Assigns a new name/label to the folder.

Additional Inherited Members

6.117.1 Detailed Description

Folder Manager Folder-Item.

6.117.2 Member Function Documentation

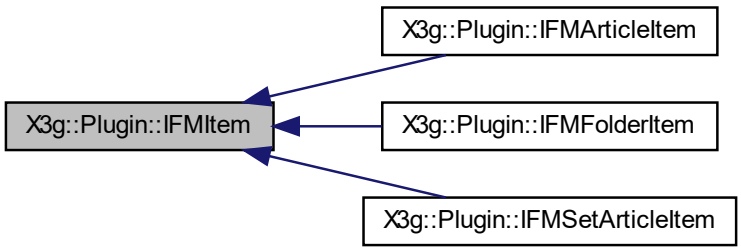
6.117.2.1 void X3g::Plugin::IFMFolderItem::SetLabel (System::String^ *pLabel*)

Assigns a new name/label to the folder.

6.118 X3g::Plugin::IFMItem Class Reference

Folder Manager Item.

Inheritance diagram for X3g::Plugin::IFMItem:



Public Member Functions

- void SetReferenceNumber (System::String^ pNumber)

  Assignes a new reference number to the item.
- IFMItem^ GetParent ()

  Returns the current parent of the item.
- ReadOnlyCollection< IFMItem^ >^ GetChildren ()

  Returns the current children of the item.

Properties

- System::String^ Id [get]

  The item id.
- System::String^ Label [get]

  The label of the item.
- System::String^ Description [get]

  The description of the item.
- System::String^ ReferenceNumber [get]

  The item's reference number.

6.118.1 Detailed Description

Folder Manager Item.

6.118.2 Member Function Documentation

6.118.2.1 ReadOnlyCollection< X3g::Plugin::IFMItem^ > X3g::Plugin::IFMItem::GetChildren ( )

Returns the current children of the item.

6.118.2.2 X3g::Plugin::IFMItem X3g::Plugin::IFMItem::GetParent ( )

Returns the current parent of the item.

6.118.2.3 void X3g::Plugin::IFMItem::SetReferenceNumber (System::String^ *pNumber*)

Assignes a new reference number to the item.

6.118.3 Property Documentation

6.118.3.1 System:: String^ X3g::Plugin::IFMItem::Description [get]

The description of the item.

For items supporting long text (currently sets and articles) this is the long text, otherwise the label is used.

6.118.3.2 System:: String^ X3g::Plugin::IFMItem::Id [get]

The item id.

If the item is an article the id relates to IArticleEntity::Id. If the item is a folder the id relates to IArticleEntityInfo::-FolderId.

6.118.3.3 System:: String^ X3g::Plugin::IFMItem::Label [get]

The label of the item.

For articles, if the short text of the article is not empty, it is used as label, otherwise the article number is used.

6.118.3.4 System:: String^ X3g::Plugin::IFMItem::ReferenceNumber [get]

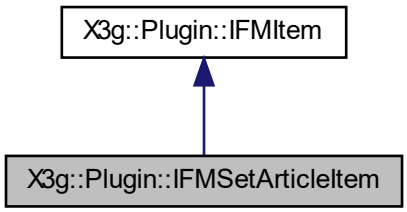
The item's reference number.

If the item is an article the number relates to IArticleEntityInfo::ReferenceNumber.

6.119 X3g::Plugin::IFMSetArticleItem Class Reference

Folder Manager SetArticle-Item.

Inheritance diagram for X3g::Plugin::IFMSetArticleItem:



Properties

- System::String^ ManufacturerId [get, set]

  The article's manufacturer id.
- System::String^ SeriesId [get, set]

  The article's series id.
- System::String^ ArticleNumber [get, set]

  The article's number.
- System::String^ ShortText [get, set]

  The short text.
- System::String^ LongText [get, set]

  The long text.
- System::String^ VariantText [get, set]

  The variant text.
- System::String^ AdditionalText [get, set]

  The additional text.
- bool ShowContent [get, set]

  Show contained articles in report.

Additional Inherited Members

6.119.1 Detailed Description

Folder Manager SetArticle-Item.

6.119.2 Property Documentation

6.119.2.1 System:: String^ X3g::Plugin::IFMSetArticleItem::AdditionalText [get],[set]

The additional text.

6.119.2.2 System:: String^ X3g::Plugin::IFMSetArticleItem::ArticleNumber [get],[set]

The article's number.

6.119.2.3 System:: String^ X3g::Plugin::IFMSetArticleItem::LongText [get],[set]

The long text.

6.119.2.4 System:: String^ X3g::Plugin::IFMSetArticleItem::ManufacturerId [get],[set]

The article's manufacturer id.

Exceptions

System::ArgumentException	An exception will be thrown if set to an invalid ID.

6.119.2.5 System:: String^ X3g::Plugin::IFMSetArticleItem::SeriesId [get],[set]

The article's series id.

Exceptions

System::ArgumentException	An exception will be thrown if set to an invalid ID.

6.119.2.6 System:: String^ X3g::Plugin::IFMSetArticleItem::ShortText [get],[set]

The short text.

6.119.2.7 bool X3g::Plugin::IFMSetArticleItem::ShowContent [get],[set]

Show contained articles in report.

6.119.2.8 System:: String^ X3g::Plugin::IFMSetArticleItem::VariantText [get],[set]

The variant text.

6.120 X3g::Plugin::IFolderManager Class Reference

Folder Manager.

Public Member Functions

- IFMItem^ GetItem (System::String^ pId)

  Get an Item by id.
- IFMFolderItem^ CreateFolder (IFMFolderItem^ pFather)

  Create a new folder, which will be the last child of it's father.
- bool DeleteFolder (IFMFolderItem^ pFolder)

  Delete a given folder.
- IFMSetArticleItem^ CreateSetArticle (IFMFolderItem^ pFather)

  Create a new set article, which will be the last child of it's father.
- bool DeleteSetArticle (IFMSetArticleItem^ pSetArticle)

  Delete a given folder.
- IFMSetArticleItem^ ConvertToSetArticle (IFMFolderItem^ pFather)

  Convert the given folder into a set article.
- IFMFolderItem^ ConvertToFolder (IFMSetArticleItem^ pFather)

  Convert the given set article into a folder.
- bool MoveItems (ReadOnlyCollection< IFMItem^ >^pItems, IFMItem^ pFather, IFMItem^ pPrev)

  Move items.
- bool StartTransaction ()

  Start Transaction.
- bool FinishTransaction ()

  Finish Transaction.

Properties

- IFMFolderItem^ CurrentFolder `[get, set]`

  The folder into which new articles will be inserted.
- IFMFolderItem^ RootFolder `[get]`

  The root folder.

6.120.1 Detailed Description

Folder Manager.

6.120.2 Member Function Documentation

6.120.2.1 X3g::Plugin::IFMFolderItem X3g::Plugin::IFolderManager::ConvertToFolder (IFMSetArticleItem^ *pFather*)

Convert the given set article into a folder.

6.120.2.2 X3g::Plugin::IFMSetArticleItem X3g::Plugin::IFolderManager::ConvertToSetArticle (IFMFolderItem^ *pFather*)

Convert the given folder into a set article.

Converting a folder which contains sub-folders or set-articles is not allowed.

6.120.2.3 X3g::Plugin::IFMFolderItem X3g::Plugin::IFolderManager::CreateFolder (IFMFolderItem^ *pFather*)

Create a new folder, which will be the last child of it's father.

6.120.2.4 X3g::Plugin::IFMSetArticleItem X3g::Plugin::IFolderManager::CreateSetArticle (IFMFolderItem^ *pFather*)

Create a new set article, which will be the last child of it's father.

6.120.2.5 bool X3g::Plugin::IFolderManager::DeleteFolder (IFMFolderItem^ *pFolder*)

Delete a given folder.

The folder and it's sub-folders must not contain articles, otherwise the folder can't be deleted. The function returns true if the folder was deleted.

6.120.2.6 bool X3g::Plugin::IFolderManager::DeleteSetArticle (IFMSetArticleItem^ *pSetArticle*)

Delete a given folder.

The set and it's sub-folders must not contain articles, otherwise the set can't be deleted. The function returns true if the set was deleted.

6.120.2.7 bool X3g::Plugin::IFolderManager::FinishTransaction ( )

Finish Transaction.

This method finishes the transaction started before with StartTransaction(). It carries out any operation whis was deferred since the call to StartTransaction()

6.120.2.8 X3g::Plugin::IFMItem X3g::Plugin::IFolderManager::GetItem (System::String^ *pId*)

Get an Item by id.

Parameters

pId	The id of the item. Relates to IFMItem::Id.

6.120.2.9 bool X3g::Plugin::IFolderManager::MoveItems (ReadOnlyCollection< IFMItem^ >^ *pItems*, IFMItem^ *pFather*, IFMItem^ *pPrev*)

Move items.

Move a set of Items either from one folder/set to another or insided the folder/set. Moving a folder/set into itself or one of it's sub-folders is not allowed. If items are moved inside the same folder/set, the list must be consecutive. The function returns true if the items could be moved.

Parameters

pItems	The items to move.
pFather	The new father (Folder or SetArticle).
pPrev	The new previous sibling. NULL means to insert at the first position in the folder/set.

6.120.2.10 bool X3g::Plugin::IFolderManager::StartTransaction ( )

Start Transaction.

This method starts an transaction. Updates to the CAD system (including) display are deferred until the transaction is finished.

6.120.3 Property Documentation

6.120.3.1 IFMFolderItem^ X3g::Plugin::IFolderManager::CurrentFolder [get],[set]

The folder into which new articles will be inserted.

6.120.3.2 IFMFolderItem^ X3g::Plugin::IFolderManager::RootFolder [get]

The root folder.

6.121 X3g::Plugin::IImage Class Reference

Wraps an object of class Image.

Public Member Functions

- virtual bool Equals (System::Object^ pOther) override

  Returns only true if the given Object is of Image type and delivers the same hash.
- virtual int GetHashCode () override

  Returns the 32 bit hash of the image.
- System::Int64 GetHashCode64 ()

  Returns the 64 bit hash of the image.
- bool IsTransparent ()

  Returns true if the image has transparent pixels.
- bool WriteToFile (System::String^ pFileName)

  Writes the image to file.
- Bitmap^ ToBitmap ()

  Converts image to a GDI+ bitmap.
- array< unsigned char >^ GetData ()

  Returns raw image data.

Properties

- String^ FileName [get]

  Returns the source filename of the image.
- int Width [get]

  Width of the image in pixels.
- int Height [get]

  Height of the image in pixels.
- PixelFormat Format [get]

  Pixel format of the image.
- int BitsPerPixel [get]

  Number of bits per pixel.
- Image^ ActualImage [get]

  Returns the actual image which is wrapped by this.

6.121.1 Detailed Description

Wraps an object of class Image.

Needed for legacy reasons.

6.121.2 Member Function Documentation

6.121.2.1 bool X3g::Plugin::IImage::Equals (System::Object^ *pOther*) [override],[virtual]

Returns only true if the given Object is of Image type and delivers the same hash.

Otherwise the result is false.

6.121.2.2 array< unsigned char > X3g::Plugin::IImage::GetData ( )

Returns raw image data.

6.121.2.3 int X3g::Plugin::IImage::GetHashCode ( ) [override],[virtual]

Returns the 32 bit hash of the image.

6.121.2.4 System::Int64 X3g::Plugin::IImage::GetHashCode64 ( )

Returns the 64 bit hash of the image.

6.121.2.5 bool X3g::Plugin::IImage::IsTransparent ( )

Returns true if the image has transparent pixels.

6.121.2.6 System::Drawing::Bitmap X3g::Plugin::IImage::ToBitmap ( )

Converts image to a GDI+ bitmap.

6.121.2.7 bool X3g::Plugin::IImage::WriteToFile (System::String^ *pFileName*)

Writes the image to file.

6.121.3 Property Documentation

6.121.3.1 Image^ X3g::Plugin::IImage::ActualImage [get]

Returns the actual image which is wrapped by this.

6.121.3.2 int X3g::Plugin::IImage::BitsPerPixel [get]

Number of bits per pixel.

6.121.3.3 String^ X3g::Plugin::IImage::FileName [get]

Returns the source filename of the image.

6.121.3.4 PixelFormat X3g::Plugin::IImage::Format [get]

Pixel format of the image.

6.121.3.5 int X3g::Plugin::IImage::Height [get]

Height of the image in pixels.

6.121.3.6 int X3g::Plugin::IImage::Width [get]

Width of the image in pixels.

6.122 X3g::Plugin::Image Class Reference

Holds an image.

Public Member Functions

- Image (int pWidth, int pHeight, PixelFormat pFormat, array< unsigned char >^pData)

  Constructor for class Image with custom image data.
- virtual bool Equals (System::Object^ pOther) override

  Returns only true if the given Object is of Image type and delivers the same hash.
- virtual int GetHashCode () override

  Returns the 32 bit hash of the image.
- System::Int64 GetHashCode64 ()

  Returns the 64 bit hash of the image.
- bool IsTransparent ()

  Returns true if the image has transparent pixels.
- Image^ Resize (int pWidth, int pHeight)

  Returns resized version of image.
- void ToStream (System::IO::Stream^ pStream, ImageFormat pFormat)

  Writes the image to stream.
- Bitmap^ ToBitmap ()

  Converts image to a GDI+ bitmap.
- array< unsigned char >^ GetData ()

  Returns copy of uncompressed image data.
- GeVec4f Sample4f (float u, float v, bool pBilinear)

  Allows image resampling with bilinear interpolation.
- IntPtr GetDataPtr ()

  Returns pointer to uncompressed image data.

Static Public Member Functions

- static Image^ LoadFromFile (String^ pFileName)

  Loads image from file.
- static Image^ LoadFromStream (System::IO::Stream^ pStream)

  Loads image from stream.

Properties

- String^ FileName [get]

  Returns the source filename of the image.
- int Width [get]

  Width of the image in pixels.
- int Height [get]

  Height of the image in pixels.
- PixelFormat Format [get]

  Pixel format of the image.
- int BitsPerPixel [get]

  Number of bits per pixel.

6.122.1 Detailed Description

Holds an image.

6.122.2 Constructor & Destructor Documentation

6.122.2.1 X3g::Plugin::Image::Image (int *pWidth*, int *pHeight*, PixelFormat *pFormat*, array< unsigned char >^ *pData*)

Constructor for class Image with custom image data.

Parameters

pWidth	The width of the image. Expects a value greater then 0.
pHeight	The height of the image. Expects a value greater then 0.
pFormat	The pixel format. See PixelFormat for the bit width.
pData	The desired image data. The size of this array has to be: `width * height * (bit_width / 8)`

6.122.3 Member Function Documentation

6.122.3.1 bool X3g::Plugin::Image::Equals (System::Object^ *pOther*) [override],[virtual]

Returns only true if the given Object is of Image type and delivers the same hash.

Otherwise the result is false.

6.122.3.2 array< unsigned char > X3g::Plugin::Image::GetData ( )

Returns copy of uncompressed image data.

6.122.3.3 IntPtr X3g::Plugin::Image::GetDataPtr ( )

Returns pointer to uncompressed image data.

6.122.3.4 int X3g::Plugin::Image::GetHashCode ( ) [override],[virtual]

Returns the 32 bit hash of the image.

6.122.3.5 System::Int64 X3g::Plugin::Image::GetHashCode64 ( )

Returns the 64 bit hash of the image.

6.122.3.6 bool X3g::Plugin::Image::IsTransparent ( )

Returns true if the image has transparent pixels.

6.122.3.7 Image X3g::Plugin::Image::LoadFromFile (String^ *pFileName*) [static]

Loads image from file.

Returns null if image couldn't be loaded.

6.122.3.8 Image X3g::Plugin::Image::LoadFromStream (System::IO::Stream^ *pStream*) [static]

Loads image from stream.

Returns null if image couldn't be loaded. Image format is detected automatically.

6.122.3.9 Image X3g::Plugin::Image::Resize (int *pWidth*, int *pHeight*)

Returns resized version of image.

6.122.3.10 GeVec4f X3g::Plugin::Image::Sample4f (float *u*, float *v*, bool *pBilinear*)

Allows image resampling with bilinear interpolation.

6.122.3.11 Bitmap X3g::Plugin::Image::ToBitmap ( )

Converts image to a GDI+ bitmap.

6.122.3.12 void X3g::Plugin::Image::ToStream (System::IO::Stream^ *pStream*, ImageFormat *pFormat*)

Writes the image to stream.

6.122.4 Property Documentation

6.122.4.1 int X3g::Plugin::Image::BitsPerPixel [get]

Number of bits per pixel.

6.122.4.2 String^ X3g::Plugin::Image::FileName [get]

Returns the source filename of the image.

6.122.4.3 PixelFormat X3g::Plugin::Image::Format [get]

Pixel format of the image.

6.122.4.4 int X3g::Plugin::Image::Height [get]

Height of the image in pixels.

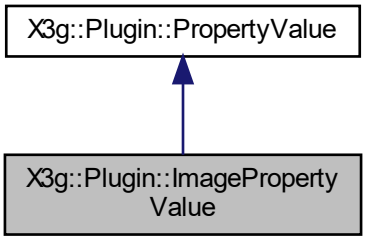
6.122.4.5 int X3g::Plugin::Image::Width [get]

Width of the image in pixels.

6.123 X3g::Plugin::ImagePropertyValue Class Reference

Image Property Value.

Inheritance diagram for X3g::Plugin::ImagePropertyValue:



Public Member Functions

- ImagePropertyValue (System::Windows::Media::Imaging::BitmapSource^ pValue, eom::Boolean pStatic)
  Constructor.

Properties

- System::Windows::Media::Imaging::BitmapSource^ Value [get, set]
  The image value.

6.123.1 Detailed Description

Image Property Value.

6.123.2 Constructor & Destructor Documentation

6.123.2.1 X3g::Plugin::ImagePropertyValue::ImagePropertyValue (System::Windows::Media::Imaging::BitmapSource^ *pValue*, eom::Boolean *pStatic*)

Constructor.

Parameters

pValue	Initial image.
pStatic	Indicates that the image source will receive updates.

6.123.3 Property Documentation

6.123.3.1 System:: Windows:: Media:: Imaging:: BitmapSource^ X3g::Plugin::ImagePropertyValue::Value [get], [set]

The image value.

6.124 X3g::Plugin::IMessageBox Interface Reference

Interface for custom message boxes.

Public Member Functions

- MessageBox::Button Show (MessageBox::Icon pIcon, System::String^ pText, MessageBox::Button pButton0, MessageBox::Button pButton1, MessageBox::Button pButton2, int pDefaultButtonIdx, int pEscapeButtonIdx)

  Show a message box with predefined button types.
- int Show (MessageBox::Icon pIcon, System::String^ pText, System::String^ pButton0, System::String^ pButton1, System::String^ pButton2, int pDefaultButtonIdx, int pEscapeButtonIdx)

  Show a message box.

6.124.1 Detailed Description

Interface for custom message boxes.

6.124.2 Member Function Documentation

6.124.2.1 MessageBox::Button X3g::Plugin::IMessageBox::Show (MessageBox::Icon *pIcon*, System::String^ *pText*, MessageBox::Button *pButton0*, MessageBox::Button *pButton1*, MessageBox::Button *pButton2*, int *pDefaultButtonIdx*, int *pEscapeButtonIdx*)

Show a message box with predefined button types.

Parameters

pIcon	Dictates the Icon and title of the message box.
pText	The central text for the message box.
pButton0	Request one MessageBox::Button to be part of the message box.
pButton1	Request one MessageBox::Button to be part of the message box.
pButton2	Request one MessageBox::Button to be part of the message box.
pDefaultButton-Idx	Specify which button is focused by default based on their index.
pEscapeButton-Idx	Specify which MessageBox::Button to return, when the user closes the message box.

Returns

The MessageBox::Button enum pressed by the user.

6.124.2.2 int X3g::Plugin::IMessageBox::Show (MessageBox::Icon *pIcon*, System::String^ *pText*, System::String^ *pButton0*, System::String^ *pButton1*, System::String^ *pButton2*, int *pDefaultButtonIdx*, int *pEscapeButtonIdx*)

Show a message box.

This method allows custom text for the message and each Button.

Parameters

pIcon	Set the icon and title of the message box.
pText	The central text for the message box.
pButton0	Text for the left button of the message box. This parameter is not allowed to be null or an empty String.
pButton1	Text for the centered button of the message box. This parameter is not allowed to be null or an empty String.
pButton2	Text for the right button of the message box. This parameter is not allowed to be null or an empty String.
pDefaultButton-Idx	Specify which button is focused by default based on their index.
pEscapeButton-Idx	Specify which button index to return, when the user closes the message box.

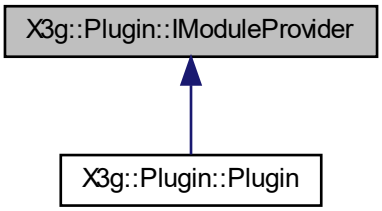
Returns

The index of the clicked button.

6.125 X3g::Plugin::IModuleProvider Interface Reference

Provides access to all major managers of the pCon.planner plugin interface.

Inheritance diagram for X3g::Plugin::IModuleProvider:



Properties

- DocumentManager^ Document [get]

  Modify or change the currently open document.
- GeometryManager^ Geometry [get]

  Covers various export formats and can be used for custom geometry exports.
- Layout::LayoutManager^ Layouts [get]

  Manage Layout.
- Logger^ Log [get]

Logging functionality.

- RendererManager^ Renderers [get]

  Manage Renderer available in the pCon.planner.
- Room::RoomModule^ Rooms [get]

  Manage Rooms.
- ViewManager^ Viewer [get]

  Manage the primary views and their arrangement.
- WindowManager^ Windows [get]

  Create and manage dockable windows.

6.125.1 Detailed Description

Provides access to all major managers of the pCon.planner plugin interface.

6.125.2 Property Documentation

6.125.2.1 DocumentManager^ X3g::Plugin::IModuleProvider::Document [get]

Modify or change the currently open document.

Contains other manager related to the current document.

See Class reference for more information.

6.125.2.2 GeometryManager^ X3g::Plugin::IModuleProvider::Geometry [get]

Covers various export formats and can be used for custom geometry exports.

6.125.2.3 Layout:: LayoutManager^ X3g::Plugin::IModuleProvider::Layouts [get]

Manage Layout.

See Class reference for more information.

6.125.2.4 Logger^ X3g::Plugin::IModuleProvider::Log [get]

Logging functionality.

See Class reference for more information.

6.125.2.5 RendererManager^ X3g::Plugin::IModuleProvider::Renderers [get]

Manage Renderer available in the pCon.planner.

See Class reference for more information.

6.125.2.6 Room:: RoomModule^ X3g::Plugin::IModuleProvider::Rooms [get]

Manage Rooms.

See Class reference for more information.

6.125.2.7 ViewManager^ X3g::Plugin::IModuleProvider::Viewer [get]

Manage the primary views and their arrangement.

See Class reference for more information.

6.125.2.8 WindowManager^ X3g::Plugin::IModuleProvider::Windows [get]

Create and manage dockable windows.

See Class reference for more information.

6.126 X3g::Plugin::InteractionEventInfo Class Reference

Parameter for interaction events.

Public Member Functions

- InteractionEventInfo (System::String^ pActionKey, bool pCanceled)

  Constructor to initialize the object.

Public Attributes

- System::String^ ActionKey

  The action key.
- bool Canceled

  True if interaction has been canceled.

6.126.1 Detailed Description

Parameter for interaction events.

6.126.2 Constructor & Destructor Documentation

6.126.2.1 X3g::Plugin::InteractionEventInfo::InteractionEventInfo (System::String^ *pActionKey,* bool *pCanceled*)

Constructor to initialize the object.

6.126.3 Member Data Documentation

6.126.3.1 System::String ^ X3g::Plugin::InteractionEventInfo::ActionKey

The action key.

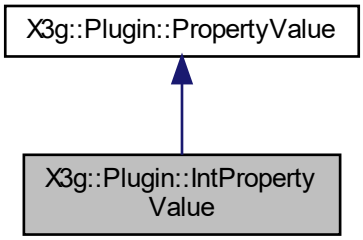
6.126.3.2 bool X3g::Plugin::InteractionEventInfo::Canceled

True if interaction has been canceled.

6.127 X3g::Plugin::IntPropertyValue Class Reference

Integer Property Value.

Inheritance diagram for X3g::Plugin::IntPropertyValue:



Public Member Functions

- IntPropertyValue (int pValue)

  Constructor.

Properties

- int Value `[get, set]`

  The integer value.

6.127.1 Detailed Description

Integer Property Value.

6.127.2 Constructor & Destructor Documentation

6.127.2.1 X3g::Plugin::IntPropertyValue::IntPropertyValue (int *pValue*)

Constructor.

Parameters

pValue	Initial integer value.

6.127.3 Property Documentation

6.127.3.1 int X3g::Plugin::IntPropertyValue::Value `[get],[set]`

The integer value.

6.128 X3g::Plugin::IO::GltfExport Class Reference

Allows to export entities in glTF format.

Public Member Functions

- GltfExport ()

  Initializes a new instance of GltfExport class.
- void ExportDocument (String^ pFileName)

  Exports document modelspace to file in glTF format.
- void ExportEntities (String^ pFileName, IEnumerable< DbEntity^ >^pEntities)

  Exports given entities to file in glTF format.

Properties

- AcisQuality AcisQuality

  ACIS vectorization quality setting.
- bool DuplicateDoubleSided

  Duplicate double sided faces.
- GltfFormatVersion FormatVersion

  glTF format version.
- int MaxTextureSize

  Defines max texture resolution.

6.128.1 Detailed Description

Allows to export entities in glTF format.

6.128.2 Constructor & Destructor Documentation

6.128.2.1 X3g::Plugin::IO::GltfExport::GltfExport ( )

Initializes a new instance of GltfExport class.

6.128.3 Member Function Documentation

6.128.3.1 void X3g::Plugin::IO::GltfExport::ExportDocument (String^ *pFileName*)

Exports document modelspace to file in glTF format.

6.128.3.2 void X3g::Plugin::IO::GltfExport::ExportEntities (String^ *pFileName*, IEnumerable< DbEntity^ >^ *pEntities*)

Exports given entities to file in glTF format.

6.128.4 Property Documentation

6.128.4.1 AcisQuality X3g::Plugin::IO::GltfExport::AcisQuality

ACIS vectorization quality setting.

6.128.4.2 bool X3g::Plugin::IO::GltfExport::DuplicateDoubleSided

Duplicate double sided faces.

6.128.4.3 GltfFormatVersion X3g::Plugin::IO::GltfExport::FormatVersion

glTF format version.

6.128.4.4 int X3g::Plugin::IO::GltfExport::MaxTextureSize

Defines max texture resolution.

6.129 X3g::Plugin::IO::PecExport Class Reference

Allows to export entities to pCon Exchange Container (PEC).

Public Member Functions

- PecExport ()

  Initializes a new instance of PecExport class.
- void ExportDocument (Stream^ pStream)

  Exports document modelspace to PEC format.
- void ExportEntities (Stream^ pStream, IEnumerable< DbEntity^ >^pEntities)

  Exports given entities to PEC format.

Properties

- AcisQuality AcisQuality

  ACIS vectorization quality setting.
- int MaxTextureSize

  Defines max texture resolution.
- bool ExportDwg

  true if dwg file is to be exported.

6.129.1 Detailed Description

Allows to export entities to pCon Exchange Container (PEC).

6.129.2 Constructor & Destructor Documentation

6.129.2.1 X3g::Plugin::IO::PecExport::PecExport ( )

Initializes a new instance of PecExport class.

6.129.3 Member Function Documentation

6.129.3.1 void X3g::Plugin::IO::PecExport::ExportDocument (Stream^ *pStream*)

Exports document modelspace to PEC format.

6.129.3.2 void X3g::Plugin::IO::PecExport::ExportEntities (Stream^ *pStream,* IEnumerable< DbEntity^ >^ *pEntities*)

Exports given entities to PEC format.

6.129.4 Property Documentation

6.129.4.1 AcisQuality X3g::Plugin::IO::PecExport::AcisQuality

ACIS vectorization quality setting.

6.129.4.2 bool X3g::Plugin::IO::PecExport::ExportDwg

true if dwg file is to be exported.

6.129.4.3 int X3g::Plugin::IO::PecExport::MaxTextureSize

Defines max texture resolution.

6.130 X3g::Plugin::IO::RgfxExport Class Reference

Allows to export entities in revit exchange format (rgfx).

Public Member Functions

- RgfxExport ()
  Initializes a new instance of RgfxExport class.
- int ExportDocument (String^ pFileName)
  Exports document modelspace to file in rgfx format.
- int ExportEntities (String^ pFileName, IEnumerable< DbEntity^ >^pEntities)
  Exports given entities to file in rgfx format.

6.130.1 Detailed Description

Allows to export entities in revit exchange format (rgfx).

6.130.2 Constructor & Destructor Documentation

6.130.2.1 X3g::Plugin::IO::RgfxExport::RgfxExport ( )

Initializes a new instance of RgfxExport class.

6.130.3 Member Function Documentation

6.130.3.1 int X3g::Plugin::IO::RgfxExport::ExportDocument (String^ *pFileName*)

Exports document modelspace to file in rgfx format.

Returns

Returns number of exported main articles.

6.130.3.2 int X3g::Plugin::IO::RgfxExport::ExportEntities (String^ *pFileName,* IEnumerable< DbEntity^ >^ *pEntities*)

Exports given entities to file in rgfx format.

Returns

Returns number of exported main articles.

6.131 X3g::Plugin::IO::UsdzExport Class Reference

Allows to export entities in Usdz format.

Public Member Functions

- UsdzExport ()

  Initializes a new instance of UsdzExport class.
- void ExportDocument (String^ pFileName)

  Exports document modelspace to file in Usdz format.
- void ExportEntities (String^ pFileName, IEnumerable< DbEntity^ >^pEntities)

  Exports given entities to file in Usdz format.

Properties

- AcisQuality AcisQuality

  ACIS vectorization quality setting.
- double Scale

  Defines overall scale factor.

6.131.1 Detailed Description

Allows to export entities in Usdz format.

6.131.2 Constructor & Destructor Documentation

6.131.2.1 X3g::Plugin::IO::UsdzExport::UsdzExport ( )

Initializes a new instance of UsdzExport class.

6.131.3 Member Function Documentation

6.131.3.1 void X3g::Plugin::IO::UsdzExport::ExportDocument (String^ *pFileName*)

Exports document modelspace to file in Usdz format.

6.131.3.2 void X3g::Plugin::IO::UsdzExport::ExportEntities (String^ *pFileName,* IEnumerable< DbEntity^ >^ *pEntities*)

Exports given entities to file in Usdz format.

6.131.4 Property Documentation

6.131.4.1 AcisQuality X3g::Plugin::IO::UsdzExport::AcisQuality

ACIS vectorization quality setting.

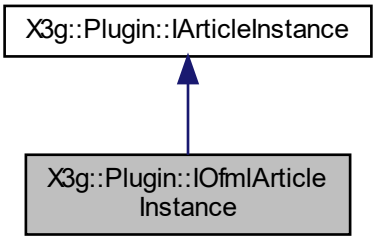
6.131.4.2 double X3g::Plugin::IO::UsdzExport::Scale

Defines overall scale factor.

6.132 X3g::Plugin::IOfmlArticleInstance Class Reference

Legacy class.

Inheritance diagram for X3g::Plugin::IOfmlArticleInstance:



Additional Inherited Members

6.132.1 Detailed Description

Legacy class.

See Articles::OfmlArticleInstance.

6.133 X3g::Plugin::IOfmlObject Class Reference

Ofml Object.

Public Member Functions

- bool IsExisting ()

  Returns true if this object still exists.
- bool IsSelectable ()

  Returns true if this object is selectable.
- bool IsArticle ()

  Returns true if this object is an article.

- bool IsA (String^ pTypeName)

  Returns true if this object is an instance of the given type.
- bool HasMember (String^ pMemberName)

  Returns true if this object has a member with the given name (e.g.
- bool CallMethod (String^ pMethodName, String^ pArguments,[Out] String^ %pResult)

  Calls/Executes the given method with the given arguments and returns the results.
- IOfmlObject^ GetParent ()

  Returns the parent of this object.
- ReadOnlyCollection< IOfmlObject^ >^ GetChildren (bool pElementsOnly)

  Returns the children of this object.

Properties

- String^ Name [get]

  The objects name.
- bool IsValid [get]

  Returns true if the IOfmlArticleInstance for this object is valid.

6.133.1 Detailed Description

Ofml Object.

6.133.2 Member Function Documentation

6.133.2.1 bool X3g::Plugin::IOfmlObject::CallMethod (String^ *pMethodName,* String^ *pArguments,* [Out] String^ % *pResult*)

Calls/Executes the given method with the given arguments and returns the results.

Also consider calling if method call modified object state.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.133.2.2 ReadOnlyCollection<IOfmlObject^> ^ X3g::Plugin::IOfmlObject::GetChildren (bool *pElementsOnly*)

Returns the children of this object.

If pElementsOnly is true only the children which are elements will be returned.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.133.2.3 IOfmlObject ^ X3g::Plugin::IOfmlObject::GetParent ( )

Returns the parent of this object.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.133.2.4 bool X3g::Plugin::IOfmlObject::HasMember (String^ *pMemberName*)

Returns true if this object has a member with the given name (e.g.

a method).

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.133.2.5 bool X3g::Plugin::IOfmlObject::IsA (String^ *pTypeName*)

Returns true if this object is an instance of the given type.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.133.2.6 bool X3g::Plugin::IOfmlObject::IsArticle ( )

Returns true if this object is an article.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.133.2.7 bool X3g::Plugin::IOfmlObject::IsExisting ( )

Returns true if this object still exists.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.133.2.8 bool X3g::Plugin::IOfmlObject::IsSelectable ( )

Returns true if this object is selectable.

Exceptions

System::Exception	An exception will be thrown if this object is no longer valid.

6.133.3 Property Documentation

6.133.3.1 bool X3g::Plugin::IOfmlObject::IsValid [get]

Returns true if the IOfmlArticleInstance for this object is valid.

6.133.3.2 String^ X3g::Plugin::IOfmlObject::Name [get]

The objects name.

6.134 X3g::Plugin::IOpenGLRenderer Class Reference

Realtime OpenGL Renderer.

Public Member Functions

- IImage^ RenderImage (int pWidth, int pHeight)

  Renders image with given width and height.
- Image^ RenderImage (int pWidth, int pHeight, IEnumerable< DbEntity^ >^pEntities)

  Renders image of given modelspace entities with given width and height.

Properties

- ICamera^ Camera [get]

  Camera parameters.
- IRenderStyle^ Style [get, set]

  Rendering style.
- System::Drawing::Color BackgroundColor

  Sets custom background color.

6.134.1 Detailed Description

Realtime OpenGL Renderer.

6.134.2 Member Function Documentation

6.134.2.1 IImage X3g::Plugin::IOpenGLRenderer::RenderImage (int *pWidth*, int *pHeight*)

Renders image with given width and height.

Returned image has pre-multiplied alpha in case of transparent background.

6.134.2.2 Image X3g::Plugin::IOpenGLRenderer::RenderImage (int *pWidth*, int *pHeight*, IEnumerable< DbEntity^ >^ *pEntities*)

Renders image of given modelspace entities with given width and height.

Returned image has pre-multiplied alpha in case of transparent background.

6.134.3 Property Documentation

6.134.3.1 System:: Drawing:: Color X3g::Plugin::IOpenGLRenderer::BackgroundColor

Sets custom background color.

White by default.

6.134.3.2 ICamera^ X3g::Plugin::IOpenGLRenderer::Camera [get]

Camera parameters.

It's not allowed to assign another camera reference. Use ICamera::CopyFrom() instead.

6.134.3.3 IRenderStyle^ X3g::Plugin::IOpenGLRenderer::Style [get], [set]

Rendering style.

6.135 X3g::Plugin::IProjectSettings Class Reference

Project Settings.

Properties

Contact and address settings

- String^ CustomerNumber [get, set]
  The customer number.
- String^ CustomerCompany [get, set]
  The customer's company.
- String^ CustomerSalutation [get, set]
  The customer's salutation.
- String^ CustomerFirstName [get, set]
  The customer's first name.
- String^ CustomerLastName [get, set]
  The customer's last name.
- String^ CustomerStreet [get, set]
  The customer's street.
- String^ CustomerPostalCode [get, set]
  The customer's postal code.
- String^ CustomerCity [get, set]
  The customer's city.
- String^ ContactName [get, set]
  The contact's name.
- String^ ContactPhone [get, set]
  The contact's phone number.
- String^ ContactEmail [get, set]
  The contact's email address.

Calculation settings

- Currency^ VAT [get, set]
  The VAT (value added tax) for this project.
- String^ CurrencyUnit [get, set]
  The currency unit for this project.
- Currency^ HeaderAddCharge [get, set]
  The header add charge.
- Currency^ HeaderDiscount [get, set]
  The header discount.
- Currency^ OverallArticleAddCharge [get, set]
  The overall article add charge.
- Currency^ OverallArticleDiscount [get, set]
  The overall article discount.

6.135.1 Detailed Description

Project Settings.

6.135.2 Property Documentation

6.135.2.1 String^ X3g::Plugin::IProjectSettings::ContactEmail [get],[set]

The contact's email address.

6.135.2.2 String^ X3g::Plugin::IProjectSettings::ContactName [get],[set]

The contact's name.

6.135.2.3 String^ X3g::Plugin::IProjectSettings::ContactPhone [get],[set]

The contact's phone number.

6.135.2.4 String^ X3g::Plugin::IProjectSettings::CurrencyUnit [get],[set]

The currency unit for this project.

Contains a 3 letter ISO currency code.

6.135.2.5 String^ X3g::Plugin::IProjectSettings::CustomerCity [get],[set]

The customer's city.

6.135.2.6 String^ X3g::Plugin::IProjectSettings::CustomerCompany [get],[set]

The customer's company.

6.135.2.7 String^ X3g::Plugin::IProjectSettings::CustomerFirstName [get],[set]

The customer's first name.

6.135.2.8 String^ X3g::Plugin::IProjectSettings::CustomerLastName [get],[set]

The customer's last name.

6.135.2.9 String^ X3g::Plugin::IProjectSettings::CustomerNumber [get],[set]

The customer number.

6.135.2.10 String^ X3g::Plugin::IProjectSettings::CustomerPostalCode [get],[set]

The customer's postal code.

6.135.2.11 String^ X3g::Plugin::IProjectSettings::CustomerSalutation [get],[set]

The customer's salutation.

6.135.2.12 String^ X3g::Plugin::IProjectSettings::CustomerStreet [get], [set]

The customer's street.

6.135.2.13 Currency^ X3g::Plugin::IProjectSettings::HeaderAddCharge [get], [set]

The header add charge.

Only relative values are supported.

6.135.2.14 Currency^ X3g::Plugin::IProjectSettings::HeaderDiscount [get], [set]

The header discount.

Only relative values are supported.

6.135.2.15 Currency^ X3g::Plugin::IProjectSettings::OverallArticleAddCharge [get], [set]

The overall article add charge.

Only relative values are supported.

6.135.2.16 Currency^ X3g::Plugin::IProjectSettings::OverallArticleDiscount [get], [set]

The overall article discount.

Only relative values are supported.

6.135.2.17 Currency^ X3g::Plugin::IProjectSettings::VAT [get], [set]

The VAT (value added tax) for this project.

Only relative values are supported.

6.136 X3g::Plugin::IProperty Interface Reference

Property.

Inherited by X3g::Plugin::PropertyImpl.

Public Member Functions

- bool SetValue (PropertyValue^ pValue, bool pUndoable)

  Assigns a new value to the property.
- void Hide ()

  Hide this property at property editor.
- void Unhide ()

  Unhide this property at property editor if hidden by Hide.

Properties

- String^ Key [get]

  The key of the property.

- PropertyType Type [get]

  The type of the property.
- String^ Name [get]

  The name of the property.
- String^ Description [get]

  The description of the property.
- ReadOnlyCollection < PropertyValue^ >^ ValueSet [get]

  Returns an optional list of valid values for this property.
- PropertyValue^ Value [get]

  The current value of the property.
- PropertyState State [get]

  The state of the property.

6.136.1 Detailed Description

Property.

6.136.2 Member Function Documentation

6.136.2.1 void X3g::Plugin::IProperty::Hide ( )

Hide this property at property editor.

6.136.2.2 bool X3g::Plugin::IProperty::SetValue (PropertyValue^ *pValue,* bool *pUndoable*)

Assigns a new value to the property.

If pUndoable is true, an undo step will be created.

Changing the value of a property may also change the state and the value of other properterties if they depend on each other. Changing the value may even add new properties. Therefore, IPropertyProvider::GetProperties() should be called again to get an up-to-date property list.

6.136.2.3 void X3g::Plugin::IProperty::Unhide ( )

Unhide this property at property editor if hidden by Hide.

6.136.3 Property Documentation

6.136.3.1 String^ X3g::Plugin::IProperty::Description [get]

The description of the property.

6.136.3.2 String^ X3g::Plugin::IProperty::Key [get]

The key of the property.

6.136.3.3 String^ X3g::Plugin::IProperty::Name [get]

The name of the property.

6.136.3.4 PropertyState X3g::Plugin::IProperty::State [get]

The state of the property.

6.136.3.5 PropertyType X3g::Plugin::IProperty::Type [get]

The type of the property.

6.136.3.6 PropertyValue^ X3g::Plugin::IProperty::Value [get]

The current value of the property.

6.136.3.7 ReadOnlyCollection< PropertyValue^ >^ X3g::Plugin::IProperty::ValueSet [get]

Returns an optional list of valid values for this property.

6.137 X3g::Plugin::IPropertyCallbacks Interface Reference

Allows to implement your own properties and add them to a provider.

Public Member Functions

- IList< IPropertyDescriptor^ >^ GetProperties ()

  Returns collection of property descriptions which are implemented by these callbacks.
- PropertyValue^ GetPropertyValue (System::String^ pKey)

  Returns value of property with given key.
- void SetPropertyValue (System::String^ pKey, PropertyValue^ pValue)

  Sets value of property with given key.
- PropertyState GetPropertyState (System::String^ pKey)

  Returns state of property with given key.

6.137.1 Detailed Description

Allows to implement your own properties and add them to a provider.

A plugin should always use a group property as root for its specific properties. In a property layout with tabs, the top group represents the tab and thus the name of the tab. Without such a group, all properties are placed in a generic "Plugin" group - this should be avoided.

6.137.2 Member Function Documentation

6.137.2.1 IList<IPropertyDescriptor^> ^ X3g::Plugin::IPropertyCallbacks::GetProperties ( )

Returns collection of property descriptions which are implemented by these callbacks.

6.137.2.2 PropertyState X3g::Plugin::IPropertyCallbacks::GetPropertyState (System::String^ *pKey*)

Returns state of property with given key.

6.137.2.3 PropertyValue ^ X3g::Plugin::IPropertyCallbacks::GetPropertyValue (System::String^ *pKey*)

Returns value of property with given key.

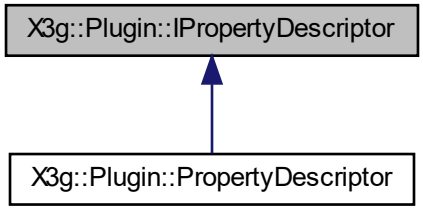
6.137.2.4 void X3g::Plugin::IPropertyCallbacks::SetPropertyValue (System::String^ *pKey*, PropertyValue^ *pValue*)

Sets value of property with given key.

6.138 X3g::Plugin::IPropertyDescriptor Interface Reference

Describes a property.

Inheritance diagram for X3g::Plugin::IPropertyDescriptor:



Properties

- System::String^ Key [get]

  The key of the property.
- PropertyType Type [get]

  The type of the property.
- System::String^ Name [get]

  The name of the property.
- System::String^ Description [get]

  The description of the property.
- ReadOnlyCollection < PropertyValue^ >^ ValueSet [get]

  Returns an optional list of valid values for this property.
- ReadOnlyCollection < IPropertyDescriptor^ >^ SubProperties [get]

  Returns an optional list of grouped properties.

6.138.1 Detailed Description

Describes a property.

6.138.2 Property Documentation

6.138.2.1 System:: String^ X3g::Plugin::IPropertyDescriptor::Description [get]

The description of the property.

6.138.2.2 System:: String^ X3g::Plugin::IPropertyDescriptor::Key [get]

The key of the property.

6.138.2.3 System:: String^ X3g::Plugin::IPropertyDescriptor::Name [get]

The name of the property.

6.138.2.4 ReadOnlyCollection< IPropertyDescriptor^ >^ X3g::Plugin::IPropertyDescriptor::SubProperties [get]

Returns an optional list of grouped properties.

6.138.2.5 PropertyType X3g::Plugin::IPropertyDescriptor::Type [get]

The type of the property.

6.138.2.6 ReadOnlyCollection< PropertyValue^ >^ X3g::Plugin::IPropertyDescriptor::ValueSet [get]

Returns an optional list of valid values for this property.

6.139 X3g::Plugin::IRenderCallback Interface Reference

Interface for the render callback to supervise render progress.

Inherited by X3g::Plugin::ManagedRenderCallbackAdapter.

Public Member Functions

- void SetFramebuffer (array< unsigned char >^pBuffer, int pWidth, int pHeight, PixelFormat pFormat)

  Notifies about new framebuffer.
- void UpdateFrame ()

  Notifies that an entire frame has been updated.
- void FinishedRendering (bool pError)

  Notifies that the rendering is finished.
- void StartRenderTask (int pNumSteps, int pTask, int pTaskCount)

  Notifies about a new Render task and sets the limits for the UpdateTaskProgress() progress.
- void UpdateTaskProgress (int long pStep)

  Notifies about the current progress of the rendering.
- bool RequestRestart ()

  Restart was requested.

6.139.1 Detailed Description

Interface for the render callback to supervise render progress.

Create your own class that inherits from IRenderCallback and register it to the property called Callback of your desired Renderer.

6.139.2 Member Function Documentation

6.139.2.1 void X3g::Plugin::IRenderCallback::FinishedRendering (bool *pError*)

Notifies that the rendering is finished.

Parameters

pError	Indicates that an error happened.

6.139.2.2 bool X3g::Plugin::IRenderCallback::RequestRestart ( )

Restart was requested.

This function will be polled before changing render settings.

Returns

A return value of true allows rendering restarts and the change of renderer settings.
A return value of false keeps previous settings.
The default return value should be true.

6.139.2.3 void X3g::Plugin::IRenderCallback::SetFramebuffer (array< unsigned char >^ *pBuffer*, int *pWidth*, int *pHeight*, PixelFormat *pFormat*)

Notifies about new framebuffer.

Pixel formats RGBA32 and RGBA128F have pre-multiplied alpha.

6.139.2.4 void X3g::Plugin::IRenderCallback::StartRenderTask (int *pNumSteps*, int *pTask*, int *pTaskCount*)

Notifies about a new Render task and sets the limits for the UpdateTaskProgress() progress.

Parameters

pNumSteps	The amount of steps for one task to complete.
pTask	The number of the started task.
pTaskCount	The amount of tasks in queue.

6.139.2.5 void X3g::Plugin::IRenderCallback::UpdateFrame ( )

Notifies that an entire frame has been updated.

A frame is one iteration over the entire image.

6.139.2.6 void X3g::Plugin::IRenderCallback::UpdateTaskProgress (int long *pStep*)

Notifies about the current progress of the rendering.

Parameters

pStep	Indicates the progress on a scale from 0 to NumSteps representing 0 to 100% progress. The value of NumSteps has been set in StartRenderTask() with the pNumSteps parameter.

6.140 X3g::Plugin::IRenderStyle Class Reference

Render style.

Public Member Functions

- void ApplyChanges ()

  Applies changes to render styles which are currently in use.

Static Public Member Functions

- static bool IsBaseModeSupported (RenderStyleBaseMode pBaseMode)

  Returns if base mode is supported by current OpenGL implementation.

Properties

- System::String^ Name `[get]`

  Name of render style.
- RenderStyleBaseMode BaseMode `[get, set]`

  Basic render mode.
- RenderLineStyle^ LineStyle `[get]`

  Render style of line entities.
- RenderLineStyle^ OutlineStyle `[get]`

  Render style of outlines.
- bool ShadowPlaneEnabled `[get, set]`

  Enables plane below scene showing shadows.
- static bool DebugRenderingEnabled `[get, set]`

  Allows users to enable debug render mode, which shows underlying wireframe and normals.

6.140.1 Detailed Description

Render style.

6.140.2 Member Function Documentation

6.140.2.1 void X3g::Plugin::IRenderStyle::ApplyChanges ( )

Applies changes to render styles which are currently in use.

6.140.2.2 bool X3g::Plugin::IRenderStyle::IsBaseModeSupported (RenderStyleBaseMode *pBaseMode*) `[static]`

Returns if base mode is supported by current OpenGL implementation.

6.140.3 Property Documentation

6.140.3.1 RenderStyleBaseMode X3g::Plugin::IRenderStyle::BaseMode [get], [set]

Basic render mode.

6.140.3.2 bool X3g::Plugin::IRenderStyle::DebugRenderingEnabled [static], [get], [set]

Allows users to enable debug render mode, which shows underlying wireframe and normals.

6.140.3.3 RenderLineStyle^ X3g::Plugin::IRenderStyle::LineStyle [get]

Render style of line entities.

6.140.3.4 System:: String^ X3g::Plugin::IRenderStyle::Name [get]

Name of render style.

6.140.3.5 RenderLineStyle^ X3g::Plugin::IRenderStyle::OutlineStyle [get]

Render style of outlines.

6.140.3.6 bool X3g::Plugin::IRenderStyle::ShadowPlaneEnabled [get], [set]

Enables plane below scene showing shadows.

6.141 X3g::Plugin::ITool Interface Reference

Tool.

Public Member Functions

- bool IsAvailable ()

  Will be called before the tool gets started to check if the tool is currently available.
- bool Start (String^ pArguments, IToolTemplate^ pTemplate)

  Will be called if the tool was started.
- void Abort ()

  Will be called if the tool was aborted.
- void Finish ()

  Will be called if the tool was finished.
- void StartPoint ()

  Will be called if the next tool step has started.
- void FinishPoint ()

  Will be called if the current tool step was finished.
- void PointUpdated ()

  Will be called every time the current point has changed (e.g.

6.141.1 Detailed Description

Tool.

6.141.2 Member Function Documentation

6.141.2.1 void X3g::Plugin::ITool::Abort ( )

Will be called if the tool was aborted.

6.141.2.2 void X3g::Plugin::ITool::Finish ( )

Will be called if the tool was finished.

6.141.2.3 void X3g::Plugin::ITool::FinishPoint ( )

Will be called if the current tool step was finished.

6.141.2.4 bool X3g::Plugin::ITool::IsAvailable ( )

Will be called before the tool gets started to check if the tool is currently available.

6.141.2.5 void X3g::Plugin::ITool::PointUpdated ( )

Will be called every time the current point has changed (e.g.

the mouse was moved).

6.141.2.6 bool X3g::Plugin::ITool::Start (String^ *pArguments,* IToolTemplate^ *pTemplate*)

Will be called if the tool was started.

If this function returns false then the tool start will be canceled.

6.141.2.7 void X3g::Plugin::ITool::StartPoint ( )

Will be called if the next tool step has started.

6.142 X3g::Plugin::IToolTemplate Interface Reference

Tool Template.

Public Member Functions

- GeVec3d GetPoint ()

  Returns the currently picked point in world coordinates.
- void AbortTool ()

  Abort the tool.
- void FinishTool ()

  Finish the tool.

Properties

- X3g::Plugin::GeVec3d^ UserCoordSysOrigin [get, set]

  Represents the origin the tool is relative to.
- MeasureType Measurement [get, set]

  Defines current measure type of this tool.
- Coordinates EnabledCoordinates [get, set]

  Defines the visible and modifiable coordinates of this tool.

6.142.1 Detailed Description

Tool Template.

6.142.2 Member Function Documentation

6.142.2.1 void X3g::Plugin::IToolTemplate::AbortTool ( )

Abort the tool.

6.142.2.2 void X3g::Plugin::IToolTemplate::FinishTool ( )

Finish the tool.

6.142.2.3 GeVec3d X3g::Plugin::IToolTemplate::GetPoint ( )

Returns the currently picked point in world coordinates.

6.142.3 Property Documentation

6.142.3.1 Coordinates X3g::Plugin::IToolTemplate::EnabledCoordinates [get],[set]

Defines the visible and modifiable coordinates of this tool.

6.142.3.2 MeasureType X3g::Plugin::IToolTemplate::Measurement [get],[set]

Defines current measure type of this tool.

6.142.3.3 X3g:: Plugin:: GeVec3d^ X3g::Plugin::IToolTemplate::UserCoordSysOrigin [get],[set]

Represents the origin the tool is relative to.

6.143 X3g::Plugin::IUndoManager Interface Reference

Undo Manager.

Public Member Functions

- void SetUndoRecordingEnabled (bool pEnabled)

  Enabled or disables undo recording.
- void BeginTransaction ()

  Starts a new transaction.
- void EndTransaction ()

  Ends a transaction.
- void AbortTransaction ()

  Aborts a transaction.
- void Undo ()

  Performs an undo operation.
- void Redo ()

  Performs a redo operation.
- void AppendCommand (ICommand^ pCommand)

  Adds a custom command.

6.143.1 Detailed Description

Undo Manager.

6.143.2 Member Function Documentation

6.143.2.1 void X3g::Plugin::IUndoManager::AbortTransaction ( )

Aborts a transaction.

Undo recording must be enabled to use this function.

6.143.2.2 void X3g::Plugin::IUndoManager::AppendCommand (ICommand^ *pCommand*)

Adds a custom command.

6.143.2.3 void X3g::Plugin::IUndoManager::BeginTransaction ( )

Starts a new transaction.

Undo recording must be enabled to use this function. All changes between begin and end of a transaction creates a single undo step. Nested transactions are supported.

6.143.2.4 void X3g::Plugin::IUndoManager::EndTransaction ( )

Ends a transaction.

Undo recording must be enabled to use this function. All changes between begin and end of a transaction creates a single undo step. Nested transactions are supported.

6.143.2.5 void X3g::Plugin::IUndoManager::Redo ( )

Performs a redo operation.

6.143.2.6 void X3g::Plugin::IUndoManager::SetUndoRecordingEnabled (bool *pEnabled*)

Enabled or disables undo recording.

Undo recording is enabled by default.

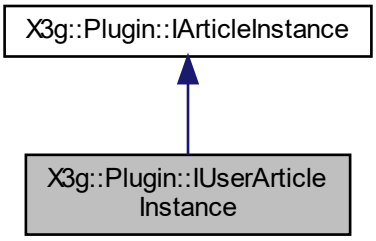
6.143.2.7 void X3g::Plugin::IUndoManager::Undo ( )

Performs an undo operation.

6.144 X3g::Plugin::IUserArticleInstance Class Reference

Legacy class.

Inheritance diagram for X3g::Plugin::IUserArticleInstance:



Additional Inherited Members

6.144.1 Detailed Description

Legacy class.

See Articles::UserArticleInstance.

6.145 X3g::Plugin::IUtilities Class Reference

Utilities.

Public Member Functions

- IImage^ LoadImageFromFile (System::String^ pFileName)

  Loads an image from a file.
- double convertLengthUnit (double pValue, LengthUnit pFrom, LengthUnit pTo)

  Converts the given value from one length unit into another length unit.

6.145.1 Detailed Description

Utilities.

6.145.2 Member Function Documentation

6.145.2.1 double X3g::Plugin::IUtilities::convertLengthUnit (double *pValue,* LengthUnit *pFrom,* LengthUnit *pTo*)

Converts the given value from one length unit into another length unit.

6.145.2.2 X3g::Plugin::IImage X3g::Plugin::IUtilities::LoadImageFromFile (System::String^ *pFileName*)

Loads an image from a file.

6.146 X3g::Plugin::IVectorImage Class Reference

Vector Image.

Public Member Functions

- void Save (System::IO::Stream^ pStream, VectorImageFormat pFormat)

  Saves the vector image to a stream in the specified format.
- void Save (String^ pFileName, VectorImageFormat pFormat)

  Saves the vector image to a file in the specified format.
- void SaveDwg (String^ pFileName, DocumentFileType pType, DocumentFileFormat pVersion)

  Saves the vector image to a file in dwg format.

Properties

- Color DefaultLineColor

  Default line color is used for lines without individual line style.
- double DefaultLineWidth

  Default line width is used for lines without individual line style.
- bool OverrideLineColor

  Override individual line color with default.
- bool OverrideLineWidth

  Override individual line width with default.

6.146.1 Detailed Description

Vector Image.

6.146.2 Member Function Documentation

6.146.2.1 void X3g::Plugin::IVectorImage::Save (System::IO::Stream^ *pStream,* VectorImageFormat *pFormat*)

Saves the vector image to a stream in the specified format.

6.146.2.2 void X3g::Plugin::IVectorImage::Save (String^ *pFileName,* VectorImageFormat *pFormat*)

Saves the vector image to a file in the specified format.

6.146.2.3 void X3g::Plugin::IVectorImage::SaveDwg (String^ *pFileName,* DocumentFileType *pType,* DocumentFileFormat *pVersion*)

Saves the vector image to a file in dwg format.

6.146.3 Property Documentation

6.146.3.1 Color X3g::Plugin::IVectorImage::DefaultLineColor

Default line color is used for lines without individual line style.

6.146.3.2 double X3g::Plugin::IVectorImage::DefaultLineWidth

Default line width is used for lines without individual line style.

6.146.3.3 bool X3g::Plugin::IVectorImage::OverrideLineColor

Override individual line color with default.

6.146.3.4 bool X3g::Plugin::IVectorImage::OverrideLineWidth

Override individual line width with default.

6.147 X3g::Plugin::IVectorRenderer Class Reference

Vector Image Renderer.

Public Member Functions

- IVectorImage^ RenderImage (int pWidth, int pHeight)

  Renders vector image of modelspace with given width and height.
- IVectorImage^ RenderImage (int pWidth, int pHeight, ReadOnlyCollection< System::String^ >^pEntities)

  Renders vector image of modelspace entities with given width and height.

Properties

- ICamera^ Camera `[get]`

  Camera parameters.
- bool LineStyles

  Enables export of individual line styles.
- bool HiddenLineRemoval

  Enables removal of hidden lines.

6.147.1 Detailed Description

Vector Image Renderer.

6.147.2 Member Function Documentation

6.147.2.1 X3g::Plugin::IVectorImage X3g::Plugin::IVectorRenderer::RenderImage (int *pWidth,* int *pHeight*)

Renders vector image of modelspace with given width and height.

6.147.2.2 X3g::Plugin::IVectorImage X3g::Plugin::IVectorRenderer::RenderImage (int *pWidth,* int *pHeight,* ReadOnlyCollection< System::String^ >^ *pEntities*)

Renders vector image of modelspace entities with given width and width and height.

6.147.3 Property Documentation

6.147.3.1 ICamera^ X3g::Plugin::IVectorRenderer::Camera [get]

Camera parameters.

It's not allowed to assign another camera reference. Use ICamera::CopyFrom() instead.

6.147.3.2 bool X3g::Plugin::IVectorRenderer::HiddenLineRemoval

Enables removal of hidden lines.

6.147.3.3 bool X3g::Plugin::IVectorRenderer::LineStyles

Enables export of individual line styles.

Vector image will only have a default line style if property is disabled.

6.148 X3g::Plugin::Layer Class Reference

Layers are used to group information in a drawing by function and to enforce linetype, color, and other standards.

Inherited by X3g::Plugin::ILayer.

Public Member Functions

- void SetVisible (bool pVisible)
  Sets the layer's visibility.

Properties

- String^ Name [get, set]
  The unique name of the layer.
- bool Visible [get]
  Determines if objects on this layer are visible or not.
- System::Drawing::Color Color [get, set]
  The color of the layer.

6.148.1 Detailed Description

Layers are used to group information in a drawing by function and to enforce linetype, color, and other standards.

6.148.2 Member Function Documentation

6.148.2.1 void X3g::Plugin::Layer::SetColor (System::Drawing::Color *pColor*) [property]

6.148.2.2 void X3g::Plugin::Layer::SetVisible (bool *pVisible*)

Sets the layer's visibility.

Depending on the number of objects on this layer this operation may be slow.

6.148.3 Property Documentation

6.148.3.1 System:: Drawing:: Color X3g::Plugin::Layer::Color [get],[set]

The color of the layer.

6.148.3.2 String^ X3g::Plugin::Layer::Name [get],[set]

The unique name of the layer.

6.148.3.3 bool X3g::Plugin::Layer::Visible [get]

Determines if objects on this layer are visible or not.

The default value is 'true'.

6.149 X3g::Plugin::LayerManager Class Reference

Layer Manager.

Public Member Functions

- Layer^ GetLayer (String^ pName)

  Retrieves a layer by name.
- IList< String^ >^ GetAllLayerNames ()

  Returns all layers of the document.
- Layer^ CreateLayer (String^ pName)

  Creates a new layer with the given name.
- void DeleteLayer (String^ pName)

  Deletes a layer.

6.149.1 Detailed Description

Layer Manager.

6.149.2 Member Function Documentation

6.149.2.1 X3g::Plugin::Layer X3g::Plugin::LayerManager::CreateLayer (String^ *pName*)

Creates a new layer with the given name.

6.149.2.2 void X3g::Plugin::LayerManager::DeleteLayer (String^ *pName*)

Deletes a layer.

If there are entities on that layer, they will be erased.

6.149.2.3 IList< System::String^ > X3g::Plugin::LayerManager::GetAllLayerNames ( )

Returns all layers of the document.

6.149.2.4 X3g::Plugin::Layer X3g::Plugin::LayerManager::GetLayer (String^ *pName*)

Retrieves a layer by name.

The layer name is not case sensitive.

6.150 X3g::Plugin::LayersChangedEventArgs Class Reference

Event arguments for layer(s) changed.

Inherits EventArgs.

Properties

- X3g::Plugin::Layer^ Layer `[get]`

  Set if name or visibility of a single layer has changed.
- bool LayerListChanged `[get]`

  True if Layers have been added or removed.

6.150.1 Detailed Description

Event arguments for layer(s) changed.

6.150.2 Property Documentation

6.150.2.1 X3g:: Plugin:: Layer^ X3g::Plugin::LayersChangedEventArgs::Layer `[get]`

Set if name or visibility of a single layer has changed.

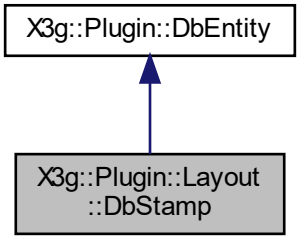
6.150.2.2 bool X3g::Plugin::LayersChangedEventArgs::LayerListChanged `[get]`

True if Layers have been added or removed.

6.151 X3g::Plugin::Layout::DbStamp Class Reference

Layout stamp item.

Inheritance diagram for X3g::Plugin::Layout::DbStamp:



Public Member Functions

- DbStamp (Page^ pPage, System::String^ pFileName)

  Create a DbStamp.
- IEnumerable< String^ >^ GetAttributes ()

  Returns all attribute keys of this stamp item.
- String^ GetAttributeValue (String^ pKey)

  Returns value of attribute with given key.
- void SetAttributeValue (String^ pKey, String^ pValue)

  Sets value of attribute with given key.
- void SetAttributeName (String^ pKey, String^ pName)

  Sets name of attribute with given key.
- String^ GetAttributeName (String^ pKey)

  Returns name of attribute with given key.

Properties

- GeVec2d Position `[get, set]`

  Position of this item.

Additional Inherited Members

6.151.1 Detailed Description

Layout stamp item.

6.151.2 Constructor & Destructor Documentation

6.151.2.1 X3g::Plugin::Layout::DbStamp::DbStamp (Page^ *pPage*, System::String^ *pFileName*)

Create a DbStamp.

Parameters

pPage	The page to place your DbStamp on.
pFileName	Filename of the DWG to load. The file needs to be suitable for stamp import.

6.151.3 Member Function Documentation

6.151.3.1 String X3g::Plugin::Layout::DbStamp::GetAttributeName (String^ *pKey*)

Returns name of attribute with given key.

6.151.3.2 IEnumerable< String^ > X3g::Plugin::Layout::DbStamp::GetAttributes ( )

Returns all attribute keys of this stamp item.

6.151.3.3 String X3g::Plugin::Layout::DbStamp::GetAttributeValue (String^ *pKey*)

Returns value of attribute with given key.

6.151.3.4 void X3g::Plugin::Layout::DbStamp::SetAttributeName (String^ *pKey*, String^ *pName*)

Sets name of attribute with given key.

6.151.3.5 void X3g::Plugin::Layout::DbStamp::SetAttributeValue (String^ *pKey*, String^ *pValue*)

Sets value of attribute with given key.

6.151.4 Property Documentation

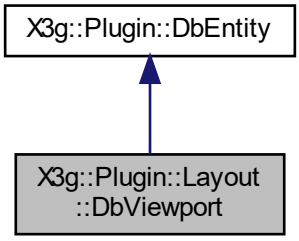
6.151.4.1 GeVec2d X3g::Plugin::Layout::DbStamp::Position [get],[set]

Position of this item.

6.152 X3g::Plugin::Layout::DbViewport Class Reference

Layout viewport item.

Inheritance diagram for X3g::Plugin::Layout::DbViewport:



Public Member Functions

- void GetDimensions ([Out] double% pWidth,[Out] double% pHeight)

  Get dimensions of the DbViewport.
- void SetDimensions (double pWidth, double pHeight)

  Set dimensions of the DbViewport.

Properties

- GeVec2d Position `[get, set]`

  Position of this item.
- ICamera^ Camera `[get]`

  Interface to camera parameters.
- IRenderStyle^ RenderStyle `[get, set]`

  Rendering style.
- System::String^ Title `[get, set]`

  Titel of this viewport.
- bool IsScaleLocked `[get, set]`

  If true the user can not change the scale by accident.
- DbViewportBackground Background `[get, set]`

  Background type of this viewport.

Additional Inherited Members

6.152.1 Detailed Description

Layout viewport item.

6.152.2 Member Function Documentation

6.152.2.1 void X3g::Plugin::Layout::DbViewport::GetDimensions ([Out] double% *pWidth*, [Out] double% *pHeight*)

Get dimensions of the DbViewport.

6.152.2.2 void X3g::Plugin::Layout::DbViewport::SetDimensions (double *pWidth,* double *pHeight*)

Set dimensions of the DbViewport.

6.152.3 Property Documentation

6.152.3.1 DbViewportBackground X3g::Plugin::Layout::DbViewport::Background [get],[set]

Background type of this viewport.

6.152.3.2 ICamera^ X3g::Plugin::Layout::DbViewport::Camera [get]

Interface to camera parameters.

It's not allowed to assign another camera reference. Use ICamera::CopyFrom() instead.

6.152.3.3 bool X3g::Plugin::Layout::DbViewport::IsScaleLocked [get],[set]

If true the user can not change the scale by accident.

This feature is only available if ICamera::Projection of this DbViewport::Camera is set to CameraProjection::-Orthographic.

6.152.3.4 GeVec2d X3g::Plugin::Layout::DbViewport::Position [get],[set]

Position of this item.

6.152.3.5 IRenderStyle^ X3g::Plugin::Layout::DbViewport::RenderStyle [get],[set]

Rendering style.

6.152.3.6 System:: String^ X3g::Plugin::Layout::DbViewport::Title [get],[set]

Titel of this viewport.

6.153 X3g::Plugin::Layout::LayoutManager Class Reference

Layout Manager.

Public Member Functions

- Page^ GetPage (int pIndex)

  Returns page with given index.
- Page^ GetPage (System::String^ pName)

  Returns page with given name.
- Page^ CreatePage (System::String^ pName)

  Creates page with given name.
- bool DeletePage (System::String^ pName)

  Deletes page with given name.

Properties

- bool Active [get, set]

  Allows to activate or deactivate layout mode.
- Page^ CurrentPage [get, set]

  The current layout page.
- int PageCount [get]

  The number of pages.

6.153.1 Detailed Description

Layout Manager.

6.153.2 Member Function Documentation

6.153.2.1 Page X3g::Plugin::Layout::LayoutManager::CreatePage (System::String^ *pName*)

Creates page with given name.

6.153.2.2 bool X3g::Plugin::Layout::LayoutManager::DeletePage (System::String^ *pName*)

Deletes page with given name.

6.153.2.3 Page X3g::Plugin::Layout::LayoutManager::GetPage (int *pIndex*)

Returns page with given index.

Returns

Returns null if page doesn't exist.

6.153.2.4 Page X3g::Plugin::Layout::LayoutManager::GetPage (System::String^ *pName*)

Returns page with given name.

Returns

Returns null if page doesn't exist.

6.153.3 Property Documentation

6.153.3.1 bool X3g::Plugin::Layout::LayoutManager::Active [get],[set]

Allows to activate or deactivate layout mode.

6.153.3.2 Page^ X3g::Plugin::Layout::LayoutManager::CurrentPage [get],[set]

The current layout page.

May be null.

6.153.3.3 int X3g::Plugin::Layout::LayoutManager::PageCount [get]

The number of pages.

6.154 X3g::Plugin::Layout::Page Class Reference

Layout page.

Public Member Functions

- void GetDimensions ([Out] double% pWidth,[Out] double% pHeight)

  Get the dimensions of the Layout page.
- void SetDimensions (double pWidth, double pHeight)

  Set the dimensions of the Layout page.
- void GetMargins ([Out] double% pLeft,[Out] double% pRight,[Out] double% pTop,[Out] double% pBottom)

  Get the margins of the Layout page.
- void SetMargins (double pLeft, double pRight, double pTop, double pBottom)

  Set the margins of the Layout page.
- void DeleteEntity (DbEntity^ pEntity)

  Delete the given Entity.
- void DeleteEntities (IEnumerable< DbEntity^ >^pEntities)

  Delete a given Enumerable of Entities.
- IList< DbEntity^ >^ GetAllEntities ()

  Get a list of all Entities on the Page.
- void SetSelection (IEnumerable< DbEntity^ >^pSelection)

  Set the Selection to the given Entities
- IList< DbEntity^ >^ GetSelection ()

  Retrieves all currently selected entities of this page.

Properties

- System::String^ Name [get, set]

  Get/Set the name of the Layout page.

6.154.1 Detailed Description

Layout page.

6.154.2 Member Function Documentation

6.154.2.1 void X3g::Plugin::Layout::Page::DeleteEntities (IEnumerable< DbEntity^ >^ *pEntities*)

Delete a given Enumerable of Entities.

6.154.2.2 void X3g::Plugin::Layout::Page::DeleteEntity (DbEntity^ *pEntity*)

Delete the given Entity.

6.154.2.3 IList< DbEntity^ > X3g::Plugin::Layout::Page::GetAllEntities ( )

Get a list of all Entities on the Page.

6.154.2.4 void X3g::Plugin::Layout::Page::GetDimensions ([Out] double% *pWidth,* [Out] double% *pHeight*)

Get the dimensions of the Layout page.

6.154.2.5 void X3g::Plugin::Layout::Page::GetMargins ([Out] double% *pLeft,* [Out] double% *pRight,* [Out] double% *pTop,* [Out] double% *pBottom*)

Get the margins of the Layout page.

6.154.2.6 IList< DbEntity^ > X3g::Plugin::Layout::Page::GetSelection ( )

Retrieves all currently selected entities of this page.

6.154.2.7 void X3g::Plugin::Layout::Page::SetDimensions (double *pWidth,* double *pHeight*)

Set the dimensions of the Layout page.

6.154.2.8 void X3g::Plugin::Layout::Page::SetMargins (double *pLeft,* double *pRight,* double *pTop,* double *pBottom*)

Set the margins of the Layout page.

6.154.2.9 void X3g::Plugin::Layout::Page::SetSelection (IEnumerable< DbEntity^ >^ *pSelection*)

Set the Selection to the given Entities

6.154.3 Property Documentation

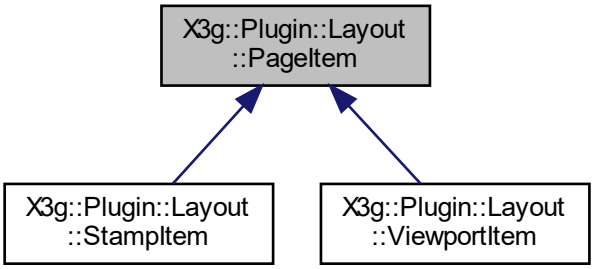
6.154.3.1 System:: String^ X3g::Plugin::Layout::Page::Name [get],[set]

Get/Set the name of the Layout page.

6.155 X3g::Plugin::Layout::PageItem Class Reference

Base class for layout page items.

Inheritance diagram for X3g::Plugin::Layout::PageItem:



Properties

- GeVec2d Position `[get, set]`

  Position of this item.

6.155.1 Detailed Description

Base class for layout page items.

6.155.2 Property Documentation

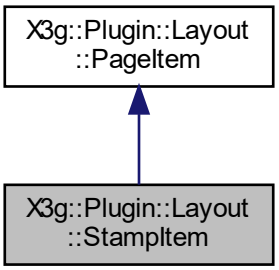
6.155.2.1 GeVec2d X3g::Plugin::Layout::PageItem::Position `[get],[set]`

Position of this item.

6.156 X3g::Plugin::Layout::StampItem Class Reference

Layout stamp item.

Inheritance diagram for X3g::Plugin::Layout::StampItem:



Public Member Functions

- array< String^ >^ GetAttributes ()

  Returns all attribute keys of this stamp item.
- String^ GetAttributeValue (String^ pKey)

  Returns value of attribute with given key.
- void SetAttributeValue (String^ pKey, String^ pValue)

  Sets value of attribute with given key.

Additional Inherited Members

6.156.1 Detailed Description

Layout stamp item.

6.156.2 Member Function Documentation

6.156.2.1 array< String^ > X3g::Plugin::Layout::StampItem::GetAttributes ( )

Returns all attribute keys of this stamp item.

6.156.2.2 String X3g::Plugin::Layout::StampItem::GetAttributeValue (String^ *pKey*)

Returns value of attribute with given key.

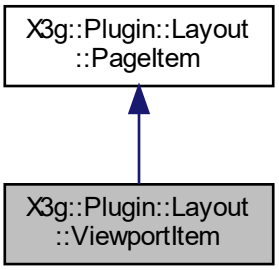
6.156.2.3 void X3g::Plugin::Layout::StampItem::SetAttributeValue (String^ *pKey*, String^ *pValue*)

Sets value of attribute with given key.

6.157 X3g::Plugin::Layout::ViewportItem Class Reference

Layout viewport item.

Inheritance diagram for X3g::Plugin::Layout::ViewportItem:



Public Member Functions

- void GetDimensions ([Out] double% pWidth,[Out] double% pHeight)

  Get dimensions of the ViewportItem.
- void SetDimensions (double pWidth, double pHeight)

  Set dimensions of the ViewportItem.

Properties

- ICamera^ Camera `[get]`

  Interface to camera parameters.
- IRenderStyle^ RenderStyle `[get, set]`

  Rendering style.

6.157.1 Detailed Description

Layout viewport item.

6.157.2 Member Function Documentation

6.157.2.1 void X3g::Plugin::Layout::ViewportItem::GetDimensions ([Out] double% *pWidth*, [Out] double% *pHeight*)

Get dimensions of the ViewportItem.

6.157.2.2 void X3g::Plugin::Layout::ViewportItem::SetDimensions (double *pWidth*, double *pHeight*)

Set dimensions of the ViewportItem.

6.157.3 Property Documentation

6.157.3.1 ICamera^ X3g::Plugin::Layout::ViewportItem::Camera `[get]`

Interface to camera parameters.

It's not allowed to assign another camera reference. Use ICamera::CopyFrom() instead.

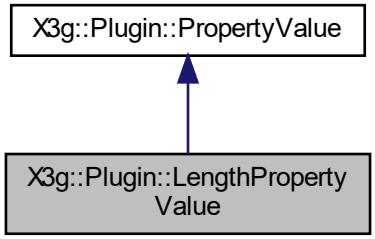
6.157.3.2 IRenderStyle^ X3g::Plugin::Layout::ViewportItem::RenderStyle [get],[set]

Rendering style.

6.158 X3g::Plugin::LengthPropertyValue Class Reference

Length Property Value.

Inheritance diagram for X3g::Plugin::LengthPropertyValue:



Public Member Functions

- LengthPropertyValue (double pValue)

  Constructor.

Properties

- double Value [get, set]

  The Length value (always in meter length unit).

6.158.1 Detailed Description

Length Property Value.

6.158.2 Constructor & Destructor Documentation

6.158.2.1 X3g::Plugin::LengthPropertyValue::LengthPropertyValue (double *pValue*)

Constructor.

Parameters

pValue	Initial length value.

6.158.3 Property Documentation

6.158.3.1 double X3g::Plugin::LengthPropertyValue::Value [get],[set]

The Length value (always in meter length unit).

6.159 X3g::Plugin::LicenseException Class Reference

License Exception.

Inherits Exception.

6.159.1 Detailed Description

License Exception.

Thrown if a license could not be obtained.

Should not longer used, if then only in combination with Safenet licensing.

6.160 X3g::Plugin::LightManager Class Reference

Light Manager.

Properties

- Image^ AmbientImage [get]

  Spherical image used for image-based lighting.
- double AmbientIntensity [get, set]

  Intensity of ambient light setup;
- double AmbientRotation [get, set]

  Rotation of image-based ambient light in degrees.

6.160.1 Detailed Description

Light Manager.

6.160.2 Property Documentation

6.160.2.1 Image^ X3g::Plugin::LightManager::AmbientImage [get]

Spherical image used for image-based lighting.

This is usually an HDR image, but users may also set LDR images. In either case the color values should be interpreted as linear.

6.160.2.2 double X3g::Plugin::LightManager::AmbientIntensity [get], [set]

Intensity of ambient light setup;

6.160.2.3 double X3g::Plugin::LightManager::AmbientRotation [get], [set]

Rotation of image-based ambient light in degrees.

Image is rotated around the z-axis by given angle.

6.161 X3g::Plugin::LogEventArgs Class Reference

Arguments for log event.

Inherits EventArgs.

Public Member Functions

- void SetSilent ()

  Allows to suppress application message dialog for all messages except fatal.

Properties

- LogLevel Level [get]

  Severity of log message.
- System::String^ Message [get]

  The message to be logged.
- System::String^ Scope [get]

  Context in which log message was raised.

6.161.1 Detailed Description

Arguments for log event.

6.161.2 Member Function Documentation

6.161.2.1 void X3g::Plugin::LogEventArgs::SetSilent ( )

Allows to suppress application message dialog for all messages except fatal.

6.161.3 Property Documentation

6.161.3.1 LogLevel X3g::Plugin::LogEventArgs::Level [get]

Severity of log message.

6.161.3.2 System:: String^ X3g::Plugin::LogEventArgs::Message [get]

The message to be logged.

6.161.3.3 System:: String^ X3g::Plugin::LogEventArgs::Scope [get]

Context in which log message was raised.

6.162 X3g::Plugin::Logger Class Reference

This class helps to use logging mechanism of pCon.planner.

Public Member Functions

- void WriteToLog (System::String^ pText)

  Write a debug info into the pCon.planner log file.
- void AddFileToCrashReport (System::String^ pFileName)

  Add a file to the crash report.

Events

- LogEventHandler^ Logging [add, remove, raise]

  Fired before message is handled by application logger.

6.162.1 Detailed Description

This class helps to use logging mechanism of pCon.planner.

6.162.2 Member Function Documentation

6.162.2.1 void X3g::Plugin::Logger::AddFileToCrashReport (System::String^ *pFileName*)

Add a file to the crash report.

6.162.2.2 void X3g::Plugin::Logger::WriteToLog (System::String^ *pText*)

Write a debug info into the pCon.planner log file.

6.162.3 Event Documentation

6.162.3.1 LogEventHandler^ X3g::Plugin::Logger::Logging [add],[remove],[raise]

Fired before message is handled by application logger.

6.163 X3g::Plugin::Material Class Reference

Material describes how an object interacts with light.

Public Member Functions

- Material (System::String^ pName, MaterialType pType)

  Creates a new material within the current document.
- virtual bool Equals (System::Object^ pOther) override

  Compares two Materials.
- virtual int GetHashCode () override

  Hash the Material and get the hash value.
- void BeginEdit ()

  Enables editing of material.
- void EndEdit ()

  Finishes editing of material.
- Image^ GetTextureImage (TextureType pType)

  Returns texture image of given type.
- void SetTextureImage (TextureType pType, Image^ pImage)

  Sets texture image of given type.
- IList< SoundAbsorption >^ GetSoundAbsorption ()

  Returns an optional list of sound absorption values.
- void SetSoundAbsorption (IEnumerable< SoundAbsorption >^pValues)

  Sets an optional collection of sound absorption values.

Static Public Member Functions

- static Material^ LoadMatz (array< unsigned char >^pData)

  Load a Material by byte array.
- static Material^ LoadMatz (System::String^ pFileName)

  Load a Material by file.

Properties

- System::String^ Name `[get]`

  The unique name of the material.
- System::String^ Label `[get, set]`

  Optional label.
- MaterialType Type `[get, set]`

  The material type.
- Color BaseColor `[get, set]`

  Base color.
- bool UseBaseColorFromEntity `[get]`

  If this flag is set, DbEntity::Color is used instead of BaseColor.
- System::Nullable< float > Albedo `[get, set]`

  Albedo equals the luminance of the base color or the average luminance of the base color texture in linear color space.
- float Metallic `[get, set]`

  Metallness of the surface.
- float Roughness `[get, set]`

  The roughness of the surface.
- float Transparency `[get, set]`

  Transparency.
- float RefractiveIndex `[get, set]`

  Index of refraction.

- Color EmissiveColor [get, set]

  Emissive color.
- float Luminance [get, set]

  Luminance.
- GeMatrix^ BaseTextureMatrix [get, set]

  Transformation of all textures except normal texture.
- GeMatrix^ NormalTextureMatrix [get, set]

  Transformation of normal texture.
- bool ReadOnly [get]

  Returns if material is read-only.

6.163.1 Detailed Description

Material describes how an object interacts with light.

6.163.2 Constructor & Destructor Documentation

6.163.2.1 X3g::Plugin::Material::Material (System::String^ *pName*, MaterialType *pType*)

Creates a new material within the current document.

Exceptions

System::NotSupported-Exception	Thrown if a material with this name already exists.

6.163.3 Member Function Documentation

6.163.3.1 void X3g::Plugin::Material::BeginEdit ( )

Enables editing of material.

This method has to be called before any setters.

6.163.3.2 void X3g::Plugin::Material::EndEdit ( )

Finishes editing of material.

Material is written to database when this function is called.

6.163.3.3 bool X3g::Plugin::Material::Equals (System::Object^ *pOther*) [override],[virtual]

Compares two Materials.

Returns

true if both are equal.
false otherwise.

6.163.3.4 int X3g::Plugin::Material::GetHashCode ( ) [override],[virtual]

Hash the Material and get the hash value.

6.163.3.5 IList< SoundAbsorption > X3g::Plugin::Material::GetSoundAbsorption ( )

Returns an optional list of sound absorption values.

6.163.3.6 Image X3g::Plugin::Material::GetTextureImage (TextureType *pType*)

Returns texture image of given type.

The base color texture is gamma-corrected (SRGB). Metallic, normal and roughness textures are linear.

6.163.3.7 Material X3g::Plugin::Material::LoadMatz (array< unsigned char >^ *pData*) [static]

Load a Material by byte array.

The byte array should have the content of a matz file.

6.163.3.8 static Material ^ X3g::Plugin::Material::LoadMatz (System::String^ *pFileName*) [static]

Load a Material by file.

6.163.3.9 void X3g::Plugin::Material::SetSoundAbsorption (IEnumerable< SoundAbsorption >^ *pValues*)

Sets an optional collection of sound absorption values.

6.163.3.10 void X3g::Plugin::Material::SetTextureImage (TextureType *pType*, Image^ *pImage*)

Sets texture image of given type.

The base color texture is expected to be in SRGB. Metallic, normal and roughness textures should be linear.

Setting base color texture resets Albedo and changes BaseColor to average texture color.

6.163.4 Property Documentation

6.163.4.1 System:: Nullable< float> X3g::Plugin::Material::Albedo [get], [set]

Albedo equals the luminance of the base color or the average luminance of the base color texture in linear color space.

If albedo is null, the base color or base color texture is used for rendering as it is. If albedo is set, the base color or texture is blended with black or white accordingly for rendering. MaterialUtility provides helper functions to apply albedo to base color.

6.163.4.2 Color X3g::Plugin::Material::BaseColor [get], [set]

Base color.

Base color is gamma-corrected (SRGB).

Setting base color resets Albedo.

6.163.4.3 GeMatrix^ X3g::Plugin::Material::BaseTextureMatrix [get], [set]

Transformation of all textures except normal texture.

6.163.4.4 Color X3g::Plugin::Material::EmissiveColor [get],[set]

Emissive color.

Color of light originating from an object. Emissive color is gamma-corrected (SRGB).

6.163.4.5 System:: String^ X3g::Plugin::Material::Label [get],[set]

Optional label.

6.163.4.6 float X3g::Plugin::Material::Luminance [get],[set]

Luminance.

The luminance indicates how much luminous power will be detected by an eye looking at the surface from a particular angle of view.

6.163.4.7 float X3g::Plugin::Material::Metallic [get],[set]

Metallness of the surface.

6.163.4.8 System:: String^ X3g::Plugin::Material::Name [get]

The unique name of the material.

6.163.4.9 GeMatrix^ X3g::Plugin::Material::NormalTextureMatrix [get],[set]

Transformation of normal texture.

6.163.4.10 bool X3g::Plugin::Material::ReadOnly [get]

Returns if material is read-only.

It's not allowed to change properties of read-only materials.

6.163.4.11 float X3g::Plugin::Material::RefractiveIndex [get],[set]

Index of refraction.

Can be changed for MaterialType::Glass only.

6.163.4.12 float X3g::Plugin::Material::Roughness [get],[set]

The roughness of the surface.

6.163.4.13 float X3g::Plugin::Material::Transparency [get],[set]

Transparency.

Transparency is the percentage of light which passes through the material without being scattered.

6.163.4.14 MaterialType X3g::Plugin::Material::Type [get],[set]

The material type.

6.163.4.15 bool X3g::Plugin::Material::UseBaseColorFromEntity [get]

If this flag is set, DbEntity::Color is used instead of BaseColor.

6.164 X3g::Plugin::MaterialChangedEventArgs Class Reference

Event arguments for change of a material.

Inherits EventArgs.

Properties

- String^ Name [get]

  Name of the material which has changed.

6.164.1 Detailed Description

Event arguments for change of a material.

6.164.2 Property Documentation

6.164.2.1 String^ X3g::Plugin::MaterialChangedEventArgs::Name [get]

Name of the material which has changed.

6.165 X3g::Plugin::MaterialManager Class Reference

Material Manager.

Public Member Functions

- Material^ GetMaterial2 (System::String^ pName)

  Retrieves a material by name.
- IList< System::String^ >^ GetAllMaterialNames ()

  Retrieves all material names of the planning.
- void DeleteMaterial (System::String^ pName)

  Deletes a material.
- void ExportMatz (String^ pName, System::IO::Stream^ pStream)

  Writes MATZ description of given material to stream.

Events

- MaterialChangedEventHandler^ MaterialChanged [add, remove, raise]

  Raised when parameters of a material have changed.

6.165.1 Detailed Description

Material Manager.

6.165.2 Member Function Documentation

6.165.2.1 void X3g::Plugin::MaterialManager::DeleteMaterial (System::String^ *pName*)

Deletes a material.

Each entity of that material will have the global material afterwards.

6.165.2.2 void X3g::Plugin::MaterialManager::ExportMatz (String^ *pName*, System::IO::Stream^ *pStream*)

Writes MATZ description of given material to stream.

6.165.2.3 IList< System::String^ > X3g::Plugin::MaterialManager::GetAllMaterialNames ( )

Retrieves all material names of the planning.

6.165.2.4 Material X3g::Plugin::MaterialManager::GetMaterial2 (System::String^ *pName*)

Retrieves a material by name.

Returns null if material doesn't exist.

6.165.3 Event Documentation

6.165.3.1 MaterialChangedEventHandler^ X3g::Plugin::MaterialManager::MaterialChanged [add], [remove], [raise]

Raised when parameters of a material have changed.

6.166 X3g::Plugin::MaterialUtility Class Reference

Provides helper functions for material handling.

Static Public Member Functions

- static Color ApplyAlbedoToBaseColor (Color pBaseColor, float pAlbedo)

  Adopts luminance of base color to the given albedo value.
- static Image^ ApplyAlbedoToBaseColor (Image^ pBaseColorTexture, float pAlbedo)

  Adopts average luminance of base color texture to the given albedo value.

6.166.1 Detailed Description

Provides helper functions for material handling.

6.166.2 Member Function Documentation

6.166.2.1 Color X3g::Plugin::MaterialUtility::ApplyAlbedoToBaseColor (Color *pBaseColor,* float *pAlbedo*) [static]

Adopts luminance of base color to the given albedo value.

The given base color has to be in srgb. Returned color is also in srgb.

6.166.2.2 Image X3g::Plugin::MaterialUtility::ApplyAlbedoToBaseColor (Image^ *pBaseColorTexture,* float *pAlbedo*) [static]

Adopts average luminance of base color texture to the given albedo value.

The given base color texture has to be in srgb. Returned texture image is also in srgb.

6.167 X3g::Plugin::MessageBox Class Reference

Static class to group message box functionality.

Public Types

- enum Button {
  Button::NoButton, Button::Ok, Button::Cancel, Button::Yes,
  Button::No, Button::Abort, Button::Retry, Button::Ignore }

  Default button types.
- enum Icon {
  Icon::NoIcon, Icon::Information, Icon::Warning, Icon::Error,
  Icon::Question }

  Icon types and titles available to show for a message box.

Static Public Member Functions

- static Button Show (Icon pIcon, System::String^ pText, Button pButton0, Button pButton1, Button pButton2, int pDefaultButtonIdx, int pEscapeButtonIdx)

  Show a message box with predefined button types.
- static int Show (Icon pIcon, System::String^ pText, System::String^ pButton0, System::String^ pButton1, System::String^ pButton2, int pDefaultButtonIdx, int pEscapeButtonIdx)

  Show a message box.
- static void Show (System::String^ pText)

  Show a message box with "Ok" button.
- static void Show (Icon pIcon, System::String^ pText)

  Show a message box with "Ok" button.

Properties

- static IMessageBox^ CustomMessageBox [get, set]

  Set/Get a custom handler for GUI messages.

6.167.1 Detailed Description

Static class to group message box functionality.

6.167.2 Member Enumeration Documentation

6.167.2.1 enum X3g::Plugin::MessageBox::Button [strong]

Default button types.

Enumerator

NoButton No button request.

Ok Ok button.

Cancel Cancel button.

Yes Yes button.

No No button.

Abort Abort button.

Retry Retry button.

Ignore Ignore button.

6.167.2.2 enum X3g::Plugin::MessageBox::Icon [strong]

Icon types and titles available to show for a message box.

Enumerator

NoIcon No icon.

Information Give the user some information.

Warning Show a warning.

Error Display an error.

Question Formulate a question.

6.167.3 Member Function Documentation

6.167.3.1 MessageBox::Button X3g::Plugin::MessageBox::Show (MessageBox::Icon *pIcon*, System::String^ *pText*, MessageBox::Button *pButton0*, MessageBox::Button *pButton1*, MessageBox::Button *pButton2*, int *pDefaultButtonIdx*, int *pEscapeButtonIdx*) [static]

Show a message box with predefined button types.

Parameters

pIcon	Dictates the Icon and title of the message box.
pText	The central text for the message box.
pButton0	Request one Button to be part of the message box.
pButton1	Request one Button to be part of the message box.
pButton2	Request one Button to be part of the message box.
pDefaultButton-Idx	Specify which button is focused by default based on their index.
pEscapeButton-Idx	Specify which Button to return, when the user closes the message box.

Returns

The Button enum pressed by the user.

6.167.3.2 int X3g::Plugin::MessageBox::Show (MessageBox::Icon *pIcon,* System::String^ *pText,* System::String^ *pButton0,* System::String^ *pButton1,* System::String^ *pButton2,* int *pDefaultButtonIdx,* int *pEscapeButtonIdx*) [static]

Show a message box.

This method allows custom text for the message and each Button.

Parameters

pIcon	Set the icon and title of the message box.
pText	The central text for the message box.
pButton0	Text for the left button of the message box. This parameter is not allowed to be null or an empty String.
pButton1	Text for the centered button of the message box. This parameter is not allowed to be null or an empty String.
pButton2	Text for the right button of the message box. This parameter is not allowed to be null or an empty String.
pDefaultButton-Idx	Specify which button is focused by default based on their index.
pEscapeButton-Idx	Specify which button index to return, when the user closes the message box.

Returns

The index of the clicked button.

6.167.3.3 void X3g::Plugin::MessageBox::Show (System::String^ *pText*) [static]

Show a message box with "Ok" button.

Parameters

pText	The central text for the message box.

6.167.3.4 void X3g::Plugin::MessageBox::Show (MessageBox::Icon *pIcon,* System::String^ *pText*) [static]

Show a message box with "Ok" button.

Parameters

pIcon	Set the icon and title of the message box.
pText	The central text for the message box.

6.167.4 Property Documentation

6.167.4.1 IMessageBox^ X3g::Plugin::MessageBox::CustomMessageBox [static],[get],[set]

Set/Get a custom handler for GUI messages.

6.168 X3g::Plugin::Modeling::Csg Class Reference

Collection of functions to do boolean operations on entities.

Static Public Member Functions

- static DbEntity^ CreateBox (GeVec3d pPoint1, GeVec3d pPoint2, Block^ pOwner)

  Create a new axis aligned box with given corner points.
- static DbEntity^ CreateSphere (GeVec3d pCenter, double pRadius, Block^ pOwner)

  Create a new sphere with center and radius.
- static DbEntity^ CreateCone (GeVec3d pBottom, double pBottomRadius, GeVec3d pTop, double pTopRadius, Block^ pOwner)

  Create a new cone out of two points and two radii.
- static DbEntity^ Extrude (IEnumerable< DbCurve^ >^pEntities, GeVec3d pVector, Block^ pOwner)

  Extrudes a number of curves along given vector.

6.168.1 Detailed Description

Collection of functions to do boolean operations on entities.

6.168.2 Member Function Documentation

6.168.2.1 DbEntity X3g::Plugin::Modeling::Csg::CreateBox (GeVec3d *pPoint1,* GeVec3d *pPoint2,* Block^ *pOwner*) [static]

Create a new axis aligned box with given corner points.

6.168.2.2 DbEntity X3g::Plugin::Modeling::Csg::CreateCone (GeVec3d *pBottom,* double *pBottomRadius,* GeVec3d *pTop,* double *pTopRadius,* Block^ *pOwner*) [static]

Create a new cone out of two points and two radii.

6.168.2.3 DbEntity X3g::Plugin::Modeling::Csg::CreateSphere (GeVec3d *pCenter,* double *pRadius,* Block^ *pOwner*) [static]

Create a new sphere with center and radius.

6.168.2.4 DbEntity X3g::Plugin::Modeling::Csg::Extrude (IEnumerable< DbCurve^ >^ *pEntities,* GeVec3d *pVector,* Block^ *pOwner*) [static]

Extrudes a number of curves along given vector.

6.169 X3g::Plugin::Modeling::GeometryAudit Class Reference

Allows to check database entities for geometry errors.

Static Public Member Functions

- static SortedSet< GeometryError >^ AuditEntities (IEnumerable< DbEntity^ >^pEntities)

  Checks given entities for geometry errors.

6.169.1 Detailed Description

Allows to check database entities for geometry errors.

6.169.2 Member Function Documentation

6.169.2.1 SortedSet< GeometryError > X3g::Plugin::Modeling::GeometryAudit::AuditEntities (IEnumerable< DbEntity^ >^ *pEntities*) [static]

Checks given entities for geometry errors.

Blocks are traversed recursively.

Parameters

pEntities	A collection of database entities.

Returns

Set of found geometry errors.

6.170 X3g::Plugin::Modeling::Projection2d Class Reference

Allows to project entities on a plane.

Public Member Functions

- Projection2d ()

  Initializes a new instance of Projection2d class.
- IEnumerable< DbEntity^ >^ ProjectOnPlane (DbEntity^ pEntity, Block^ pOwner)

  Project pEntity on plane and add the result to pOwner.
- IEnumerable< DbEntity^ >^ ProjectOnPlane (IEnumerable< DbEntity^ >^pEntities, Block^ pOwner)

  Project pEntities on plane and add the result to pOwner.

Properties

- GePlane^ ProjectionPlane [get, set]

  Plane to be project on.

6.170.1 Detailed Description

Allows to project entities on a plane.

6.170.2 Constructor & Destructor Documentation

6.170.2.1 X3g::Plugin::Modeling::Projection2d::Projection2d ( )

Initializes a new instance of Projection2d class.

6.170.3 Member Function Documentation

6.170.3.1 IEnumerable< DbEntity^ > X3g::Plugin::Modeling::Projection2d::ProjectOnPlane (DbEntity^ *pEntity,* Block^ *pOwner*)

Project pEntity on plane and add the result to pOwner.

6.170.3.2 IEnumerable< DbEntity^ > X3g::Plugin::Modeling::Projection2d::ProjectOnPlane (IEnumerable< DbEntity^ >^ *pEntities*, Block^ *pOwner*)

Project pEntities on plane and add the result to pOwner.

6.170.4 Property Documentation

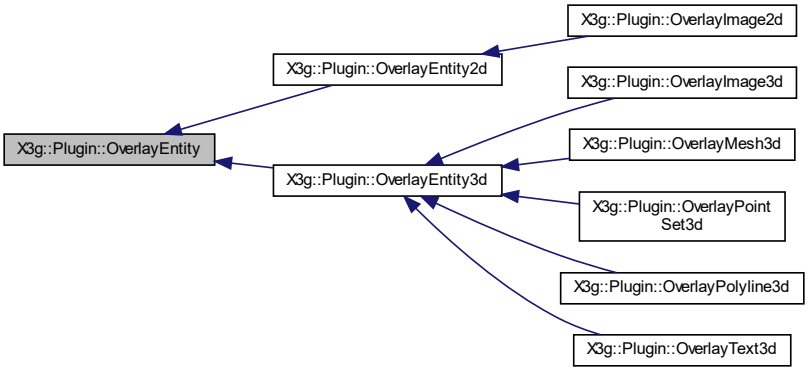
6.170.4.1 GePlane^ X3g::Plugin::Modeling::Projection2d::ProjectionPlane [get],[set]

Plane to be project on.

6.171 X3g::Plugin::OverlayEntity Class Reference

Base class for graphical objects without database representation.

Inheritance diagram for X3g::Plugin::OverlayEntity:



Public Member Functions

- void Delete ()

  Deletes the overlay entity.

Properties

- bool Visible [get, set]

  Allows to change visibility.
- bool Printable [get, set]

  If set to true the overlay element will be included during printing.
- System::String^ ToolTip [get, set]

  Text shown near mouse pointer when it hovers over this entity.
- System::String^ StatusMessage [get, set]

  Text shown in status bar when mouse pointer hovers over this entity.

Events

- System::EventHandler^ Click [add, remove, raise]

  Occurs when a OverlayEntity is clicked.
- System::EventHandler^ MouseEnter [add, remove, raise]

  Occurs when the mouse pointer enters this entity.
- System::EventHandler^ MouseLeave [add, remove, raise]

  Occurs when the mouse pointer leaves this entity.

6.171.1 Detailed Description

Base class for graphical objects without database representation.

This kind of objects are rendered as overlay on the actual scene. They are intended to be used for temporary graphical feedback and interactive manipulation (e.g. bounding boxes and buttons). For the latter purpose it's possible to register according event handlers. The objects become visible on instantiation. Property OverlayEntity::Visible allows temporary hiding. Overlay entities which are no longer needed must be released by calling OverlayEntity::Delete. Otherwise the graphical representation would stay in memory.

6.171.2 Member Function Documentation

6.171.2.1 void X3g::Plugin::OverlayEntity::Delete ( )

Deletes the overlay entity.

This invalidates the entity and it must no longer be used. For OverlayEntity this is the same as calling Dispose().

6.171.3 Property Documentation

6.171.3.1 bool X3g::Plugin::OverlayEntity::Printable [get],[set]

If set to true the overlay element will be included during printing.

6.171.3.2 System:: String^ X3g::Plugin::OverlayEntity::StatusMessage [get],[set]

Text shown in status bar when mouse pointer hovers over this entity.

6.171.3.3 System:: String^ X3g::Plugin::OverlayEntity::ToolTip [get],[set]

Text shown near mouse pointer when it hovers over this entity.

6.171.3.4 bool X3g::Plugin::OverlayEntity::Visible [get],[set]

Allows to change visibility.

Invisible entities don't receive mouse events.

6.171.4 Event Documentation

6.171.4.1 System:: EventHandler^ X3g::Plugin::OverlayEntity::Click [add],[remove],[raise]

Occurs when a OverlayEntity is clicked.

6.171.4.2 System:: EventHandler^ X3g::Plugin::OverlayEntity::MouseEnter [add],[remove],[raise]

Occurs when the mouse pointer enters this entity.

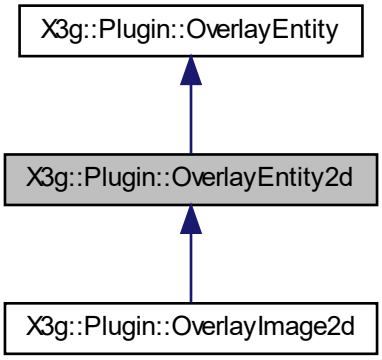
6.171.4.3 System:: EventHandler^ X3g::Plugin::OverlayEntity::MouseLeave [add],[remove],[raise]

Occurs when the mouse pointer leaves this entity.

6.172 X3g::Plugin::OverlayEntity2d Class Reference

Base class for 2d overlay entities.

Inheritance diagram for X3g::Plugin::OverlayEntity2d:



Properties

- OverlayLevel2d Level [get, set]
  Overlay has several levels.

Additional Inherited Members

6.172.1 Detailed Description

Base class for 2d overlay entities.

Each 2d overlay entity ist just visible within a certain view. The entities are placed in projection space.

6.172.2 Property Documentation

6.172.2.1 OverlayLevel2d X3g::Plugin::OverlayEntity2d::Level [get],[set]

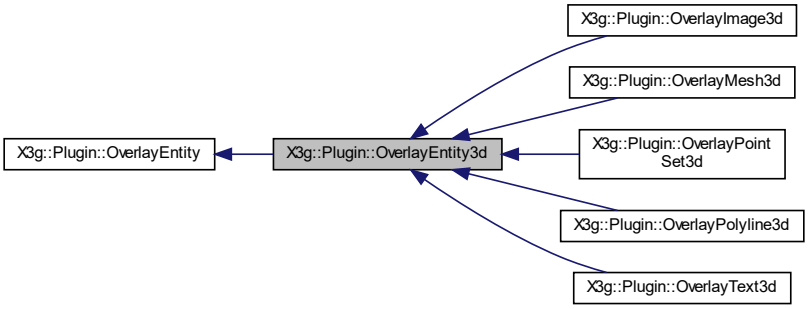
Overlay has several levels.

Higher levels are rendered over lower levels. This means that entities with higher level may occlude lower level entities. Default is Foreground.

6.173 X3g::Plugin::OverlayEntity3d Class Reference

Base class for 3d overlay entities.

Inheritance diagram for X3g::Plugin::OverlayEntity3d:



Properties

- System::Drawing::Color Color [get, set]

  The entity color.
- GeMatrix^ Transform [get, set]

  Transformation of the entity.
- OverlayLevel3d Level [get, set]

  Overlay has several levels.

Additional Inherited Members

6.173.1 Detailed Description

Base class for 3d overlay entities.

6.173.2 Property Documentation

6.173.2.1 System:: Drawing:: Color X3g::Plugin::OverlayEntity3d::Color [get],[set]

The entity color.

6.173.2.2 OverlayLevel3d X3g::Plugin::OverlayEntity3d::Level [get],[set]

Overlay has several levels.

Higher levels are rendered over lower levels. This means that entities with higher level may occlude lower level entities. Default is Middle.

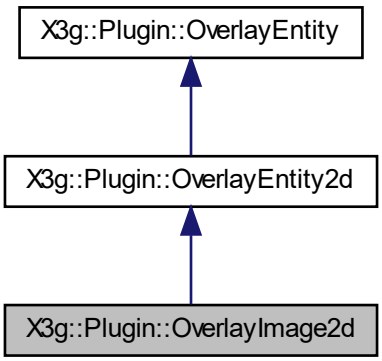
6.173.2.3 GeMatrix^ X3g::Plugin::OverlayEntity3d::Transform [get],[set]

Transformation of the entity.

6.174 X3g::Plugin::OverlayImage2d Class Reference

An image which is rendered as 2d overlay.

Inheritance diagram for X3g::Plugin::OverlayImage2d:



Public Member Functions

- OverlayImage2d (View^ pView)

  Creates an image in view overlay.
- void SetImage (array< unsigned char >^pData, int pWidth, int pHeight, PixelFormat pFormat)

  Sets image data.

Properties

- GeVec2d Position `[get, set]`

  Position of top left image corner in view projection space.
- GeVec2d Offset `[get, set]`

  Image offset from it's origin position in pixels.
- bool DisplayScaling `[get, set]`

  Enables automatic scaling of offset and size for high DPI display devices.

Additional Inherited Members

6.174.1 Detailed Description

An image which is rendered as 2d overlay.

6.174.2 Constructor & Destructor Documentation

6.174.2.1 X3g::Plugin::OverlayImage2d::OverlayImage2d (View^ *pView*)

Creates an image in view overlay.

6.174.3 Member Function Documentation

6.174.3.1 void X3g::Plugin::OverlayImage2d::SetImage (array< unsigned char >^ *pData*, int *pWidth*, int *pHeight*, PixelFormat *pFormat*)

Sets image data.

Width and height in pixels.

6.174.4 Property Documentation

6.174.4.1 bool X3g::Plugin::OverlayImage2d::DisplayScaling [get],[set]

Enables automatic scaling of offset and size for high DPI display devices.

Enabled by default.

6.174.4.2 GeVec2d X3g::Plugin::OverlayImage2d::Offset [get],[set]

Image offset from it's origin position in pixels.

Offset is affected by display scaling.

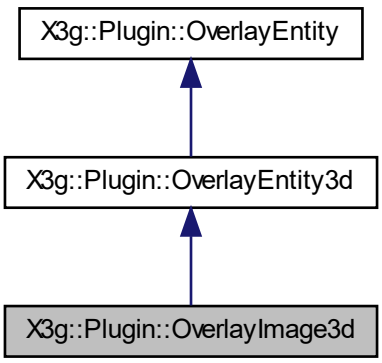
6.174.4.3 GeVec2d X3g::Plugin::OverlayImage2d::Position [get],[set]

Position of top left image corner in view projection space.

6.175 X3g::Plugin::OverlayImage3d Class Reference

An image wich is rendered as 3d overlay.

Inheritance diagram for X3g::Plugin::OverlayImage3d:



Public Member Functions

- OverlayImage3d ()

Creates an image.

- void SetImage (array< unsigned char >^pData, int pWidth, int pHeight, PixelFormat pFormat)

  Sets image data.

Properties

- GeVec3d Origin [get, set]

  Origin in world space.
- bool FixedSize [get, set]

  Image rendered with original pixel size regardless of camera position and zoom.
- bool HideOccluded [get, set]

  Hides overlay image if it is occluded by a regular entity.
- bool ScreenAligned [get, set]

  Image facing towards viewer (billboard).
- double ViewingAngle [get, set]

  Hides overlay image if observation point is outside of viewing angle.

Additional Inherited Members

6.175.1 Detailed Description

An image wich is rendered as 3d overlay.

Untransformed image has center in origin and is facing positive z-direction. Width and height are one unit. The image is without database representation.

6.175.2 Constructor & Destructor Documentation

6.175.2.1 X3g::Plugin::OverlayImage3d::OverlayImage3d ( )

Creates an image.

6.175.3 Member Function Documentation

6.175.3.1 void X3g::Plugin::OverlayImage3d::SetImage (array< unsigned char >^ *pData*, int *pWidth*, int *pHeight*, PixelFormat *pFormat*)

Sets image data.

Width and height in pixels.

6.175.4 Property Documentation

6.175.4.1 bool X3g::Plugin::OverlayImage3d::FixedSize [get],[set]

Image rendered with original pixel size regardless of camera position and zoom.

6.175.4.2 bool X3g::Plugin::OverlayImage3d::HideOccluded [get],[set]

Hides overlay image if it is occluded by a regular entity.

Works for fixed size images only. See OverlayImage3d::FixedSize.

6.175.4.3 GeVec3d X3g::Plugin::OverlayImage3d::Origin [get], [set]

Origin in world space.

6.175.4.4 bool X3g::Plugin::OverlayImage3d::ScreenAligned [get], [set]

Image facing towards viewer (billboard).

6.175.4.5 double X3g::Plugin::OverlayImage3d::ViewingAngle [get], [set]

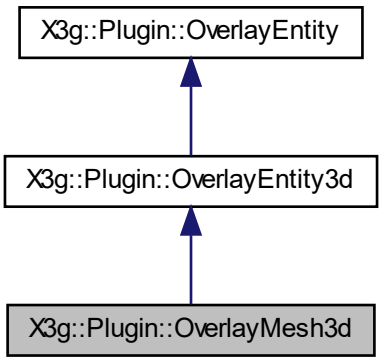
Hides overlay image if observation point is outside of viewing angle.

Angle in [0,180]. Works for fixed size images only. See OverlayImage3d::FixedSize.

6.176 X3g::Plugin::OverlayMesh3d Class Reference

A graphical triangle mesh without database representation.

Inheritance diagram for X3g::Plugin::OverlayMesh3d:



Public Member Functions

- OverlayMesh3d (array< GeVec3d >^pVertices, array< int >^pIndices)

  Creates a triangle mesh with given vertices and vertex indices.
- void SetVertices (array< GeVec3d >^pVertices)

  Sets the vertices of the mesh.
- void SetIndices (array< int >^pIndices)

  Sets the vertex indices of the mesh.

Additional Inherited Members

6.176.1 Detailed Description

A graphical triangle mesh without database representation.

6.176.2 Constructor & Destructor Documentation

6.176.2.1 X3g::Plugin::OverlayMesh3d::OverlayMesh3d (array< GeVec3d >^ *pVertices*, array< int >^ *pIndices*)

Creates a triangle mesh with given vertices and vertex indices.

6.176.3 Member Function Documentation

6.176.3.1 void X3g::Plugin::OverlayMesh3d::SetIndices (array< int >^ *pIndices*)

Sets the vertex indices of the mesh.

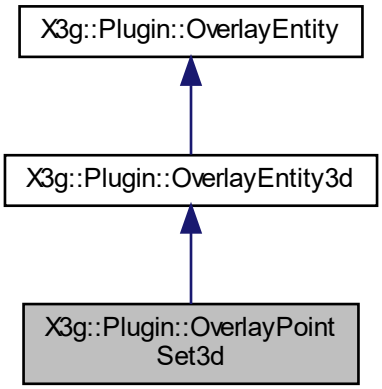
6.176.3.2 void X3g::Plugin::OverlayMesh3d::SetVertices (array< GeVec3d >^ *pVertices*)

Sets the vertices of the mesh.

6.177 X3g::Plugin::OverlayPointSet3d Class Reference

A graphical point set without database representation.

Inheritance diagram for X3g::Plugin::OverlayPointSet3d:



Public Member Functions

- OverlayPointSet3d (array< GeVec3d >^pPositions, array< System::Drawing::Color >^pColors)

  Creates a point set with given positions and colors.
- array< GeVec3d >^ GetPositions ()

  Returns a copy of the point set positions.
- void SetPositions (array< GeVec3d >^pPositions)

  Sets the positions of the points.
- array< System::Drawing::Color >^ GetColors ()

  Returns a copy of the point set colors.
- void SetColors (array< System::Drawing::Color >^pColors)

Sets the colors of the points.

Properties

- double PointSize [get, set]

  Point size.

Additional Inherited Members

6.177.1 Detailed Description

A graphical point set without database representation.

6.177.2 Constructor & Destructor Documentation

6.177.2.1 X3g::Plugin::OverlayPointSet3d::OverlayPointSet3d (array< GeVec3d >^ *pPositions*, array< System::Drawing::Color >^ *pColors*)

Creates a point set with given positions and colors.

pColors may be null. If pColors is not null it must be of the same size as pPositions.

6.177.3 Member Function Documentation

6.177.3.1 array< System::Drawing::Color > X3g::Plugin::OverlayPointSet3d::GetColors ( )

Returns a copy of the point set colors.

6.177.3.2 array< GeVec3d > X3g::Plugin::OverlayPointSet3d::GetPositions ( )

Returns a copy of the point set positions.

6.177.3.3 void X3g::Plugin::OverlayPointSet3d::SetColors (array< System::Drawing::Color >^ *pColors*)

Sets the colors of the points.

6.177.3.4 void X3g::Plugin::OverlayPointSet3d::SetPositions (array< GeVec3d >^ *pPositions*)

Sets the positions of the points.

6.177.4 Property Documentation

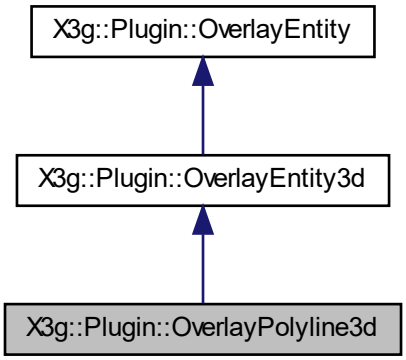
6.177.4.1 double X3g::Plugin::OverlayPointSet3d::PointSize [get],[set]

Point size.

6.178 X3g::Plugin::OverlayPolyline3d Class Reference

A graphical polyline without database representation.

Inheritance diagram for X3g::Plugin::OverlayPolyline3d:



Public Member Functions

- OverlayPolyline3d (array< GeVec3d >^pVertices)

  Creates a polyline with given vertices.
- array< GeVec3d >^ GetVertices ()

  Returns a copy of the point set positions.
- void SetVertices (array< GeVec3d >^pVertices)

  Sets the vertices of the polyline.
- void SetLineStyle (bool pDashed, long pFactor, int pPattern)

  If enabled the polyline is rendered with given dash pattern.

Properties

- double LineWidth `[get, set]`

  Width of line in pixels.

Additional Inherited Members

6.178.1 Detailed Description

A graphical polyline without database representation.

6.178.2 Constructor & Destructor Documentation

6.178.2.1 X3g::Plugin::OverlayPolyline3d::OverlayPolyline3d (array< GeVec3d >^ *pVertices*)

Creates a polyline with given vertices.

6.178.3 Member Function Documentation

6.178.3.1 array< GeVec3d > X3g::Plugin::OverlayPolyline3d::GetVertices ( )

Returns a copy of the point set positions.

6.178.3.2 void X3g::Plugin::OverlayPolyline3d::SetLineStyle (bool *pDashed,* long *pFactor,* int *pPattern*)

If enabled the polyline is rendered with given dash pattern.

Otherwise the polyline is rendered solid (default).

6.178.3.3 void X3g::Plugin::OverlayPolyline3d::SetVertices (array< GeVec3d >^ *pVertices*)

Sets the vertices of the polyline.

6.178.4 Property Documentation

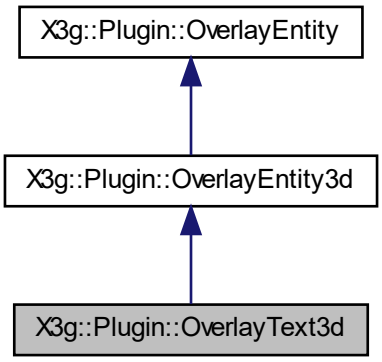
6.178.4.1 double X3g::Plugin::OverlayPolyline3d::LineWidth [get],[set]

Width of line in pixels.

6.179 X3g::Plugin::OverlayText3d Class Reference

A graphical text without database representation.

Inheritance diagram for X3g::Plugin::OverlayText3d:



Public Member Functions

- OverlayText3d (String^ pText)

  Creates a text.
- OverlayText3d (String^ pText, bool pAlignToScreen)

  Creates a text.

Properties

- String^ Text [get, set]

  The text.
- double CharacterSize [get, set]

  Size of text characters.
- HorizontalTextAlignment HorizontalAlignment [get, set]

  Gets or sets horizontal alignment of the text.
- VerticalTextAlignment VerticalAlignment [get, set]

  Gets or sets vertical alignment of the text.

Additional Inherited Members

6.179.1 Detailed Description

A graphical text without database representation.

6.179.2 Constructor & Destructor Documentation

6.179.2.1 X3g::Plugin::OverlayText3d::OverlayText3d (String^ *pText*)

Creates a text.

6.179.2.2 X3g::Plugin::OverlayText3d::OverlayText3d (String^ *pText,* bool *pAlignToScreen*)

Creates a text.

6.179.3 Property Documentation

6.179.3.1 double X3g::Plugin::OverlayText3d::CharacterSize [get],[set]

Size of text characters.

6.179.3.2 HorizontalTextAlignment X3g::Plugin::OverlayText3d::HorizontalAlignment [get],[set]

Gets or sets horizontal alignment of the text.

6.179.3.3 String^ X3g::Plugin::OverlayText3d::Text [get],[set]

The text.

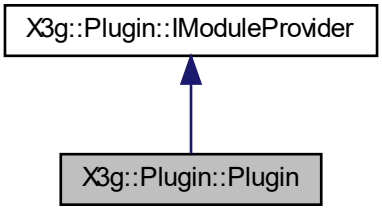
6.179.3.4 VerticalTextAlignment X3g::Plugin::OverlayText3d::VerticalAlignment [get],[set]

Gets or sets vertical alignment of the text.

6.180 X3g::Plugin::Plugin Class Reference

The plugin interface.

Inheritance diagram for X3g::Plugin::Plugin:



Public Member Functions

Required Interfaces

- virtual System::String^ X3gGetGlobalIdentifier () override
  Must return a globally unique plugin identifier
- virtual System::String^ X3gGetName () override
  Must return the plugin's name.
- virtual System::String^ X3gGetVendor () override
  Must return the vendor the the plugin.
- virtual System::String^ X3gGetDescription () override
  Must return the detailed plugin description.
- virtual System::String^ X3gGetVersion () override
  Must return the version of the plugin.
- virtual System::IntPtr **X3gGetCustomData** ()

Optional Notifications

- virtual bool X3gInitialize (System::String^ pPluginPath)
  Notification, called once when the plugin is loaded.
- virtual void X3gFinalize ()
  Notification, called once when the plugin is unloaded.
- virtual void X3gActivate ()
  Notification, called when the plugin must reactivate itself.
- virtual void X3gDeactivate ()
  Notification, called when the plugin must deactivate itself.
- virtual void X3gEvent (EventType pEventType, Object^ pParameter)
  Called when the plugin receives an event.

Actions

- virtual void X3gAction (System::String^ pActionKey, System::String^ pArguments)
  Executes an action with given optional arguments.
- virtual bool X3gIsActionAvailable (System::String^ pActionKey)
  Checks if the given action is currently available.
- virtual bool X3gSetWindowContent (Window^ pWindow)
  Requests plugin to set the window content.

- virtual IEnumerable
  < System::String^ >^ X3gGetContextMenuUIKeys ()
  Requests plugin to return UI keys for context menu items.

Threading

- void Invoke (SendOrPostCallback^ pCallback, Object^ pParam, bool pWait)
  Invoke a function call from another thread.

Protected Attributes

- IActionManager^ X3gActions
  Actions.
- IAppManager^ X3gApp
  Application Services.
- IArticleManager^ X3gArticles
  Article handling.
- IUndoManager^ X3gUndo
  Undo/Redo.
- IUtilities^ X3gUtils
  Utilities.

Properties

- DocumentManager^ Document `[get]`
  Modify or change the currently open document.
- GeometryManager^ Geometry `[get]`
  Covers various export formats and can be used for custom geometry exports.
- Layout::LayoutManager^ Layouts `[get]`
  Manage Layout.
- Logger^ Log `[get]`
  Logging functionality.
- RendererManager^ Renderers `[get]`
  Manage Renderer available in the pCon.planner.
- Room::RoomModule^ Rooms `[get]`
  Manage Rooms.
- ViewManager^ Viewer `[get]`
  Manage the primary views and their arrangement.
- WindowManager^ Windows `[get]`
  Create and manage dockable windows.

6.180.1 Detailed Description

The plugin interface.

Services from the kernel are divided into several manager classes.

Each communication with the planner itself is done using one of the X3gXXX() functions of the plugin class or calling member functions

of the manager classes like DocumentManager.

A plugin processes data as a result of notification functions or events. Notification functions can be:

- Notifications about the plugin itself (e.g. X3gInitialize())
- A special event, e.g. EventType.DocumentNew
- An action

All geometry data classes have the Ge prefix while all database classes have a Db prefix.

There are also general geometry helper classes with Ge prefix like GeMatrix or GeVec3f.

For example, if you want to write some custom geometry export and are therefore only interested

in processing 3d geometry data regardless of the underlying database objects, use GeometryManager::GetSelectedGeometry().

On the other hand, if you are interested in the database entities, use e.g. DocumentManager::GetSelection().

You can also insert raw geometry from which database objects are created using GeometryManager::InsertGeometry().

Floating point values of database entities are stored with double precision.

The vectorization of database entities into OpenGL data uses single precision.

Therefore all GeDrawable objects are using `float` values, while DbEntity objects use `double` values.

6.180.2 Member Function Documentation

6.180.2.1 void X3g::Plugin::Plugin::Invoke (SendOrPostCallback^ *pCallback,* Object^ *pParam,* bool *pWait*)

Invoke a function call from another thread.

The plugin interface is not threadsafe. Direct calls are only allowed from the plugins main thread, otherwise Invoke() needs to be used.

Parameters

pCallback	The function to be called in form of a SendOrPostCallback object.
pParam	The pParam parameter will be used as a parameter for the pCallback function.
pWait	If pWait is true, the calling thread will wait/block until pCallback was executed.

6.180.2.2 virtual void X3g::Plugin::Plugin::X3gAction (System::String^ *pActionKey,* System::String^ *pArguments*) `[virtual]`

Executes an action with given optional arguments.

6.180.2.3 virtual void X3g::Plugin::Plugin::X3gActivate ( ) `[virtual]`

Notification, called when the plugin must reactivate itself.

6.180.2.4 virtual void X3g::Plugin::Plugin::X3gDeactivate ( ) `[virtual]`

Notification, called when the plugin must deactivate itself.

6.180.2.5 virtual void X3g::Plugin::Plugin::X3gEvent (EventType *pEventType,* Object^ *pParameter*) `[virtual]`

Called when the plugin receives an event.

6.180.2.6 virtual void X3g::Plugin::Plugin::X3gFinalize ( ) [virtual]

Notification, called once when the plugin is unloaded.

6.180.2.7 virtual IEnumerable<System::String^> ^ X3g::Plugin::Plugin::X3gGetContextMenuUIKeys ( ) [virtual]

Requests plugin to return UI keys for context menu items.

This method is called before the context menu is opened. If the plugin wants to add items to the context menu it has to return the according UI keys here. UI keys may be action keys and widget keys as defined in the plugins .ui-file.

6.180.2.8 virtual System::String ^ X3g::Plugin::Plugin::X3gGetDescription ( ) [pure virtual]

Must return the detailed plugin description.

6.180.2.9 virtual System::String ^ X3g::Plugin::Plugin::X3gGetGlobalIdentifier ( ) [pure virtual]

Must return a globally unique plugin identifier

6.180.2.10 virtual System::String ^ X3g::Plugin::Plugin::X3gGetName ( ) [pure virtual]

Must return the plugin's name.

6.180.2.11 virtual System::String ^ X3g::Plugin::Plugin::X3gGetVendor ( ) [pure virtual]

Must return the vendor the the plugin.

6.180.2.12 virtual System::String ^ X3g::Plugin::Plugin::X3gGetVersion ( ) [pure virtual]

Must return the version of the plugin.

6.180.2.13 virtual bool X3g::Plugin::Plugin::X3gInitialize (System::String^ *pPluginPath*) [virtual]

Notification, called once when the plugin is loaded.

Should be used to register actions and do basic initialization. If the return value is false, the plugin will not be loaded.

No major processing should be done here. The application initialization is not finished at this point. Processing can be done after the event EventType::ApplicationReady was triggered.

6.180.2.14 virtual bool X3g::Plugin::Plugin::X3gIsActionAvailable (System::String^ *pActionKey*) [virtual]

Checks if the given action is currently available.

6.180.2.15 virtual bool X3g::Plugin::Plugin::X3gSetWindowContent (Window^ *pWindow*) [virtual]

Requests plugin to set the window content.

This method is called when windows are created through WindowManager and when windows are restored on application startup. Window creation will be canceled if this method returns false. This may be used in case of restore when the plugin has changed and particluar windows are no longer available.

6.180.3 Property Documentation

6.180.3.1 DocumentManager^ X3g::Plugin::Plugin::Document [get]

Modify or change the currently open document.

Contains other manager related to the current document.

See Class reference for more information.

6.180.3.2 GeometryManager^ X3g::Plugin::Plugin::Geometry [get]

Covers various export formats and can be used for custom geometry exports.

6.180.3.3 Layout:: LayoutManager^ X3g::Plugin::Plugin::Layouts [get]

Manage Layout.

See Class reference for more information.

6.180.3.4 Logger^ X3g::Plugin::Plugin::Log [get]

Logging functionality.

See Class reference for more information.

6.180.3.5 RendererManager^ X3g::Plugin::Plugin::Renderers [get]

Manage Renderer available in the pCon.planner.

See Class reference for more information.

6.180.3.6 Room:: RoomModule^ X3g::Plugin::Plugin::Rooms [get]

Manage Rooms.

See Class reference for more information.

6.180.3.7 ViewManager^ X3g::Plugin::Plugin::Viewer [get]

Manage the primary views and their arrangement.

See Class reference for more information.

6.180.3.8 WindowManager^ X3g::Plugin::Plugin::Windows [get]

Create and manage dockable windows.

See Class reference for more information.

6.181 X3g::Plugin::PropertiesChangedEventArgs Class Reference

Provides information about property changes.

Inherits EventArgs.

Properties

- bool ValuesOnly [get]

  True if only property values have changed.

6.181.1 Detailed Description

Provides information about property changes.

6.181.2 Property Documentation

6.181.2.1 bool X3g::Plugin::PropertiesChangedEventArgs::ValuesOnly [get]

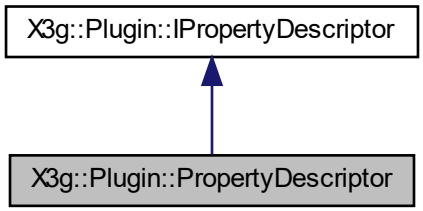
True if only property values have changed.

Otherwise the list of properties may have changed.

6.182 X3g::Plugin::PropertyDescriptor Class Reference

Default implementation of IPropertyDescriptor.

Inheritance diagram for X3g::Plugin::PropertyDescriptor:



Public Member Functions

- PropertyDescriptor (System::String^ pKey, PropertyType pType, System::String^ pName, System::String^ pDescription)

  Constructor.
- void AddSubProperty (PropertyDescriptor^ pSubProp)

  Adds a property to a group property descriptor.
- void AddChoiceValue (ChoicePropertyValue^ pValue)

  Adds a value to a choice property.

Properties

- virtual System::String^ Key [get]

  The key of the property.
- virtual PropertyType Type [get]

The type of the property.

- virtual System::String^ Name [get]

  The name of the property.

- virtual System::String^ Description [get]

  The description of the property.

- virtual ReadOnlyCollection < PropertyValue^ >^ ValueSet [get]

  Returns an optional list of valid values for this property.

- virtual ReadOnlyCollection < IPropertyDescriptor^ >^ SubProperties [get]

  Returns an optional list of grouped properties.

6.182.1 Detailed Description

Default implementation of IPropertyDescriptor.

6.182.2 Constructor & Destructor Documentation

6.182.2.1 X3g::Plugin::PropertyDescriptor::PropertyDescriptor (System::String^ *pKey,* PropertyType *pType,* System::String^ *pName,* System::String^ *pDescription*)

Constructor.

Parameters

pKey	The key of the property.
pType	The type of the property.
pName	The name of the property.
pDescription	The description of the property.

6.182.3 Member Function Documentation

6.182.3.1 void X3g::Plugin::PropertyDescriptor::AddChoiceValue (ChoicePropertyValue^ *pValue*)

Adds a value to a choice property.

6.182.3.2 void X3g::Plugin::PropertyDescriptor::AddSubProperty (PropertyDescriptor^ *pSubProp*)

Adds a property to a group property descriptor.

6.182.4 Property Documentation

6.182.4.1 virtual System:: String^ X3g::Plugin::PropertyDescriptor::Description [get]

The description of the property.

6.182.4.2 virtual System:: String^ X3g::Plugin::PropertyDescriptor::Key [get]

The key of the property.

6.182.4.3 virtual System:: String^ X3g::Plugin::PropertyDescriptor::Name [get]

The name of the property.

6.182.4.4 virtual ReadOnlyCollection< IPropertyDescriptor^ >^ X3g::Plugin::PropertyDescriptor::SubProperties [get]

Returns an optional list of grouped properties.

6.182.4.5 virtual PropertyType X3g::Plugin::PropertyDescriptor::Type [get]

The type of the property.

6.182.4.6 virtual ReadOnlyCollection< PropertyValue^ >^ X3g::Plugin::PropertyDescriptor::ValueSet [get]

Returns an optional list of valid values for this property.

6.183 X3g::Plugin::PropertyProvider Class Reference

Property Provider.

Inherited by X3g::Plugin::IPropertyProvider.

Public Member Functions

- ReadOnlyCollection< IProperty^ >^ GetProperties ()

  Returns a list of supported properties.
- IProperty^ GetProperty (System::String^ pKey)

  Returns a property by key.
- bool BeginChangeSection ()

  Notifies about a section where multiple properties may change.
- bool EndChangeSection ()

  Notifies about the end of a property change section.
- void Add (IPropertyCallbacks^ pCallbacks)

  Adds custom properties to a provider.
- void Remove (IPropertyCallbacks^ pCallbacks)

  Removes custom properties from a provider.
- void Update (bool pValuesOnly)

  Notifies observers that properties have changed.

Events

- PropertiesChangedEventHandler^ PropertiesChanged [add, remove, raise]

  Notifies about change of property list or property values.

6.183.1 Detailed Description

Property Provider.

6.183.2 Member Function Documentation

6.183.2.1 void X3g::Plugin::PropertyProvider::Add (IPropertyCallbacks^ *pCallbacks*)

Adds custom properties to a provider.

6.183.2.2 bool X3g::Plugin::PropertyProvider::BeginChangeSection ( )

Notifies about a section where multiple properties may change.

The provider may collect all changes made within this section, and may notify observers only once on finish. Subsections are also allowed, but notifications will be sent only when the outer section is closed.

6.183.2.3 bool X3g::Plugin::PropertyProvider::EndChangeSection ( )

Notifies about the end of a property change section.

6.183.2.4 ReadOnlyCollection< X3g::Plugin::IProperty^ > X3g::Plugin::PropertyProvider::GetProperties ( )

Returns a list of supported properties.

Includes custom properties.

6.183.2.5 IProperty X3g::Plugin::PropertyProvider::GetProperty (System::String^ *pKey*)

Returns a property by key.

Returns null if property doesn't exist.

6.183.2.6 void X3g::Plugin::PropertyProvider::Remove (IPropertyCallbacks^ *pCallbacks*)

Removes custom properties from a provider.

6.183.2.7 void X3g::Plugin::PropertyProvider::Update (bool *pValuesOnly*)

Notifies observers that properties have changed.

Parameter pValuesOnly should be true if list of properties is still the same and only values or states have changed.

If this method is called within a change section observers are notified when EndChangeSection is called.

6.183.3 Event Documentation

6.183.3.1 PropertiesChangedEventHandler^ X3g::Plugin::PropertyProvider::PropertiesChanged [add], [remove],[raise]

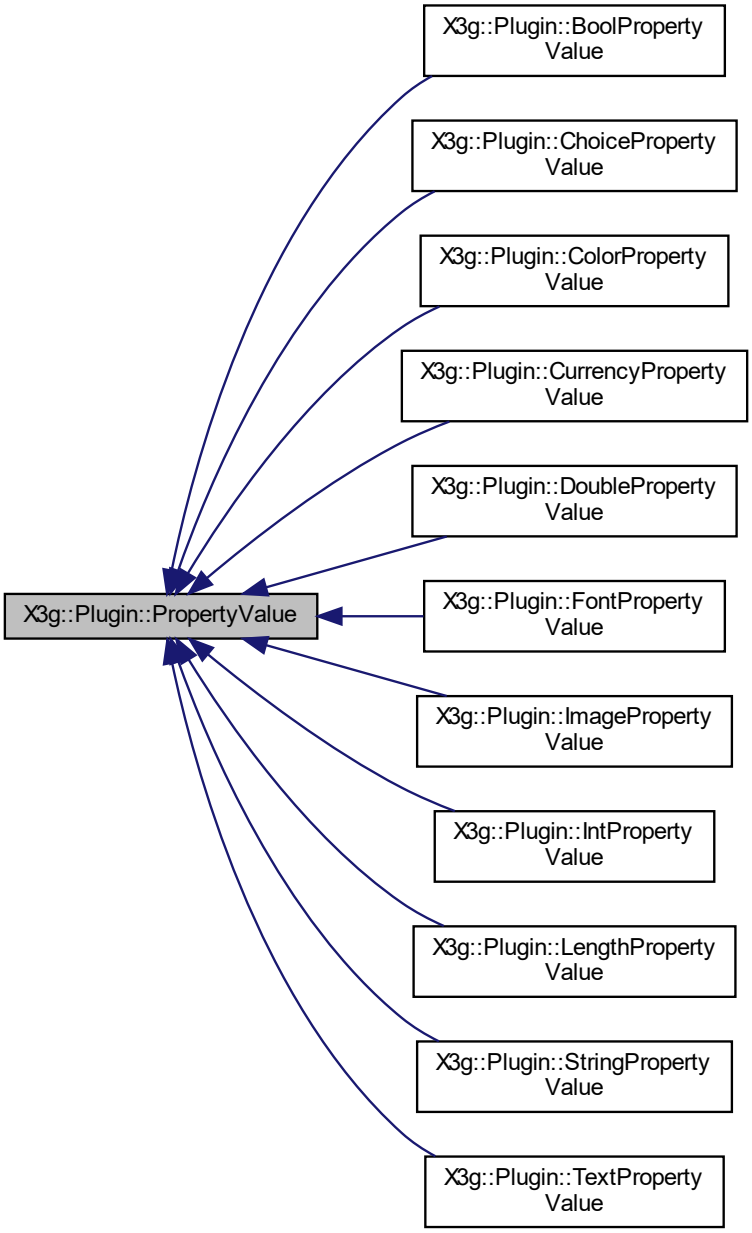
Notifies about change of property list or property values.

Supported for top-level entities only.

6.184 X3g::Plugin::PropertyValue Class Reference

Base class for various value properties.

Inheritance diagram for X3g::Plugin::PropertyValue:



Properties

- PropertyType Type `[get]`

  The type of the value.

- String^ ValueAsString `[get, set]`

  The string representation of the value.

6.184.1 Detailed Description

Base class for various value properties.

6.184.2 Property Documentation

6.184.2.1 PropertyType X3g::Plugin::PropertyValue::Type [get]

The type of the value.

6.184.2.2 String^ X3g::Plugin::PropertyValue::ValueAsString [get], [set]

The string representation of the value.

Exceptions

System::Exception	The setter throws an exception if the given string does not represent a valid value.

6.185 X3g::Plugin::Renderer Class Reference

The base class for all renderer.

Public Member Functions

- virtual void StartRendering (int pWidth, int pHeight)

  Starts rendering of frame.
- virtual bool StartRendering (int pWidth, int pHeight, bool pSequenced)=0

  Starts rendering.
- virtual void PauseRendering ()=0

  Pause rendering.
- virtual void ContinueRendering ()=0

  Resumes rendering.
- virtual void FinishRendering ()=0

  End all rendering or exporting.
- virtual void ReadSettings (System::String^ pJson)=0

  Set the current settings.
- virtual System::String^ WriteSettings ()=0

  Get the current settings.
- virtual void ResetSettings ()=0

  This loads the default settings.
- virtual void ExportRenderJob (int pWidth, int pHeight, ReadOnlyCollection< ICamera^ >^pCameras, System::String^ pFileName, System::String^ pEmail)=0

  This will export the current planning.

Properties

- System::String^ Name [get]

  Unique name of the renderer.
- System::String^ Description [get]

  Description of the renderer.

- System::String^ ExportFormatExtension [get]

  Default file extension of export format.
- ICamera^ Camera [get]

  Camera parameters.
- IRenderCallback^ Callback

  Register a callback.

Events

- RenderProgressChangedEventHandler^ PassProgressChanged

  Register an event to progress change.

6.185.1 Detailed Description

The base class for all renderer.

6.185.2 Member Function Documentation

6.185.2.1 virtual void X3g::Plugin::Renderer::ContinueRendering ( ) [pure virtual]

Resumes rendering.

Depending on the implementation it will just restart the rendering.

6.185.2.2 virtual void X3g::Plugin::Renderer::ExportRenderJob (int *pWidth*, int *pHeight*, ReadOnlyCollection< ICamera^ >^ *pCameras*, System::String^ *pFileName*, System::String^ *pEmail*) [pure virtual]

This will export the current planning.

Parameters

pWidth	Frame width.
pHeight	Frame height.
pCameras	A list of cameras to be rendered.
pFileName	Filename to write. This name should contain the render specific extension. e.g.: An OSPRay export filename should end with `.rjob`
pEmail	The email of the User who exports the job.

6.185.2.3 virtual void X3g::Plugin::Renderer::FinishRendering ( ) [pure virtual]

End all rendering or exporting.

6.185.2.4 virtual void X3g::Plugin::Renderer::PauseRendering ( ) [pure virtual]

Pause rendering.

Depending on the implementation it will just stop the rendering.

6.185.2.5 virtual void X3g::Plugin::Renderer::ReadSettings (System::String^ *pJson*) [pure virtual]

Set the current settings.

6.185.2.6 virtual void X3g::Plugin::Renderer::ResetSettings ( ) [pure virtual]

This loads the default settings.

6.185.2.7 virtual void X3g::Plugin::Renderer::StartRendering (int *pWidth*, int *pHeight*) [virtual]

Starts rendering of frame.

Parameters

pWidth	Frame width.
pHeight	Frame height.

Returns

Returns false if rendering has been aborted.

6.185.2.8 virtual bool X3g::Plugin::Renderer::StartRendering (int *pWidth*, int *pHeight*, bool *pSequenced*) [pure virtual]

Starts rendering.

Parameters

pWidth	Frame width.
pHeight	Frame height.
pSequenced	True if additional frames of same scene will follow.

Returns

Returns false if rendering has been aborted.

6.185.2.9 virtual System::String ^ X3g::Plugin::Renderer::WriteSettings ( ) [pure virtual]

Get the current settings.

6.185.3 Property Documentation

6.185.3.1 IRenderCallback^ X3g::Plugin::Renderer::Callback

Register a callback.

Remove callback by setting this to null.

6.185.3.2 ICamera^ X3g::Plugin::Renderer::Camera [get]

Camera parameters.

It's not supported to assign another camera reference. Use ICamera::CopyFrom() instead.

6.185.3.3 System:: String^ X3g::Plugin::Renderer::Description [get]

Description of the renderer.

6.185.3.4 System:: String^ X3g::Plugin::Renderer::ExportFormatExtension [get]

Default file extension of export format.

6.185.3.5 System:: String^ X3g::Plugin::Renderer::Name [get]

Unique name of the renderer.

6.185.4 Event Documentation

6.185.4.1 RenderProgressChangedEventHandler^ X3g::Plugin::Renderer::PassProgressChanged

Register an event to progress change.

6.186 X3g::Plugin::RendererManager Class Reference

Manages the Renderer available in the pCon.planner.

Public Member Functions

- void RegisterRenderer (Renderer^ pRenderer)

  Register a Renderer.
- void UnregisterRenderer (Renderer^ pRenderer)

  Unregister a Renderer.
- Renderer^ GetRenderer (System::String^ pName)

  Get registered Renderer by name.
- ReadOnlyCollection< Renderer^ >^ GetAllRenderers ()

  Get all registered Renderer.

Properties

- IOpenGLRenderer^ OpenGL [get]

  Built-in OpenGL renderer.
- IVectorRenderer^ Vector [get]

  Built-in vector image renderer.
- RenderStyleManager^ RenderStyles [get]

  OpenGL image rendering styles.

6.186.1 Detailed Description

Manages the Renderer available in the pCon.planner.

6.186.2 Member Function Documentation

6.186.2.1 ReadOnlyCollection< Renderer^ > X3g::Plugin::RendererManager::GetAllRenderers ( )

Get all registered Renderer.

6.186.2.2 Renderer X3g::Plugin::RendererManager::GetRenderer (System::String^ *pName*)

Get registered Renderer by name.

6.186.2.3 void X3g::Plugin::RendererManager::RegisterRenderer (Renderer^ *pRenderer*)

Register a Renderer.

6.186.2.4 void X3g::Plugin::RendererManager::UnregisterRenderer (Renderer^ *pRenderer*)

Unregister a Renderer.

6.186.3 Property Documentation

6.186.3.1 IOpenGLRenderer^ X3g::Plugin::RendererManager::OpenGL [get]

Built-in OpenGL renderer.

6.186.3.2 RenderStyleManager^ X3g::Plugin::RendererManager::RenderStyles [get]

OpenGL image rendering styles.

6.186.3.3 IVectorRenderer^ X3g::Plugin::RendererManager::Vector [get]

Built-in vector image renderer.

6.187 X3g::Plugin::RenderLineStyle Class Reference

Describes a line style of a render style.

Properties

- bool Enabled [get, set]

  Allows to enable rendering of lines.

6.187.1 Detailed Description

Describes a line style of a render style.

6.187.2 Property Documentation

6.187.2.1 bool X3g::Plugin::RenderLineStyle::Enabled [get],[set]

Allows to enable rendering of lines.

6.188 X3g::Plugin::RenderProgressChangedEventArgs Class Reference

Event arguments for render progress changes.

Inherits ProgressChangedEventArgs.

Public Member Functions

- RenderProgressChangedEventArgs (int pProgressPercentage)

  Constructor.

Properties

- bool ProgressStarted `[get]`

  True if this is the first progress change for current task.
- bool ProgressFinished `[get]`

  True if this is the last progress change for current task.

6.188.1 Detailed Description

Event arguments for render progress changes.

6.188.2 Constructor & Destructor Documentation

6.188.2.1 X3g::Plugin::RenderProgressChangedEventArgs::RenderProgressChangedEventArgs (int *pProgressPercentage*)

Constructor.

Parameters

pProgressPercentage	Task progress percentage.

6.188.3 Property Documentation

6.188.3.1 bool X3g::Plugin::RenderProgressChangedEventArgs::ProgressFinished `[get]`

True if this is the last progress change for current task.

6.188.3.2 bool X3g::Plugin::RenderProgressChangedEventArgs::ProgressStarted `[get]`

True if this is the first progress change for current task.

6.189 X3g::Plugin::RenderStyleManager Class Reference

Manager for render styles.

Public Member Functions

- IList< IRenderStyle^ >^ GetAllRenderStyles ()

  Returns all available styles.

- IRenderStyle^ GetRenderStyle (System::String^ pName)

  Returns render style by name.

- IRenderStyle^ CreateRenderStyle (System::String^ pName)

  Creates a new render style.

- void DeleteRenderStyle (IRenderStyle^ pStyle)

  Deletes a render style.

6.189.1 Detailed Description

Manager for render styles.

6.189.2 Member Function Documentation

6.189.2.1 IRenderStyle X3g::Plugin::RenderStyleManager::CreateRenderStyle (System::String^ *pName*)

Creates a new render style.

6.189.2.2 void X3g::Plugin::RenderStyleManager::DeleteRenderStyle (IRenderStyle^ *pStyle*)

Deletes a render style.

6.189.2.3 IList< IRenderStyle^ > X3g::Plugin::RenderStyleManager::GetAllRenderStyles ( )

Returns all available styles.

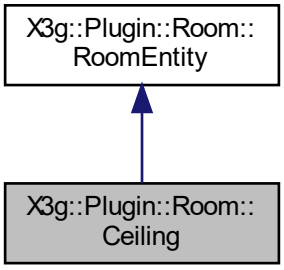
6.189.2.4 IRenderStyle X3g::Plugin::RenderStyleManager::GetRenderStyle (System::String^ *pName*)

Returns render style by name.

6.190 X3g::Plugin::Room::Ceiling Class Reference

A ceiling.

Inheritance diagram for X3g::Plugin::Room::Ceiling:



Public Member Functions

- Ceiling (array< GeVec2d >^pVertices, double pElevation)

  Creates a Ceiling at specified elevation with the specified vertices.
- array< GeVec2d >^ GetVertices ()

  Returns the vertices of this ceiling.

Properties

- String^ Id `[get]`

  Unqiue id of this ceiling within the current document.
- double Thickness `[get, set]`

  Returns the thickness of this ceiling.
- double Elevation `[get]`

  Returns the elevation of this ceiling.

6.190.1 Detailed Description

A ceiling.

6.190.2 Constructor & Destructor Documentation

6.190.2.1 X3g::Plugin::Room::Ceiling::Ceiling (array< GeVec2d >^ *pVertices*, double *pElevation*)

Creates a Ceiling at specified elevation with the specified vertices.

6.190.3 Member Function Documentation

6.190.3.1 array< GeVec2d > X3g::Plugin::Room::Ceiling::GetVertices ( )

Returns the vertices of this ceiling.

6.190.4 Property Documentation

6.190.4.1 double X3g::Plugin::Room::Ceiling::Elevation [get]

Returns the elevation of this ceiling.

6.190.4.2 String^ X3g::Plugin::Room::Ceiling::Id [get]

Unqiue id of this ceiling within the current document.

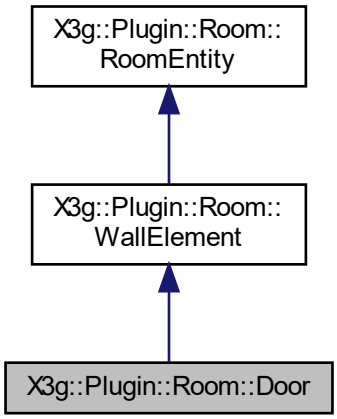
6.190.4.3 double X3g::Plugin::Room::Ceiling::Thickness [get], [set]

Returns the thickness of this ceiling.

6.191 X3g::Plugin::Room::Door Class Reference

A door.

Inheritance diagram for X3g::Plugin::Room::Door:



Public Member Functions

- Door (X3g::Plugin::Room::Wall^ pWall, GeVec3d pPos)

  Creates a door at specified Wall and specified position.
- Door (X3g::Plugin::Room::PolyWall^ pWall, GeVec3d pPos)

  Creates a door at specified Wall and specified position.

Properties

- double FrameDepth [get]

  Returns the frame depth of this door.

- double FrameThickness [get]

  Returns the frame thickness of this door.

6.191.1 Detailed Description

A door.

6.191.2 Constructor & Destructor Documentation

6.191.2.1 X3g::Plugin::Room::Door::Door (X3g::Plugin::Room::Wall^ *pWall,* GeVec3d *pPos*)

Creates a door at specified Wall and specified position.

6.191.2.2 X3g::Plugin::Room::Door::Door (X3g::Plugin::Room::PolyWall^ *pWall,* GeVec3d *pPos*)

Creates a door at specified Wall and specified position.

6.191.3 Property Documentation

6.191.3.1 double X3g::Plugin::Room::Door::FrameDepth [get]

Returns the frame depth of this door.

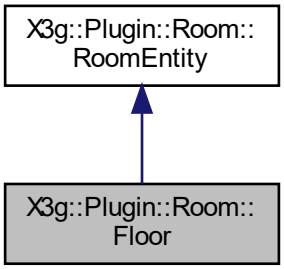
6.191.3.2 double X3g::Plugin::Room::Door::FrameThickness [get]

Returns the frame thickness of this door.

6.192 X3g::Plugin::Room::Floor Class Reference

A Floor.

Inheritance diagram for X3g::Plugin::Room::Floor:



Public Member Functions

- Floor (array< GeVec2d >^pVertices)

  Creates a floor within the current document.
- array< GeVec2d >^ GetVertices ()

  Returns the vertices of this floor.

Properties

- String^ Id [get]

  Unqiue id of this floor within the current document.
- double Elevation [get]

  Returns the elevation of this floor.

6.192.1 Detailed Description

A Floor.

6.192.2 Constructor & Destructor Documentation

6.192.2.1 X3g::Plugin::Room::Floor::Floor (array< GeVec2d >^ *pVertices*)

Creates a floor within the current document.

6.192.3 Member Function Documentation

6.192.3.1 array< GeVec2d > X3g::Plugin::Room::Floor::GetVertices ( )

Returns the vertices of this floor.

6.192.4 Property Documentation

6.192.4.1 double X3g::Plugin::Room::Floor::Elevation [get]

Returns the elevation of this floor.

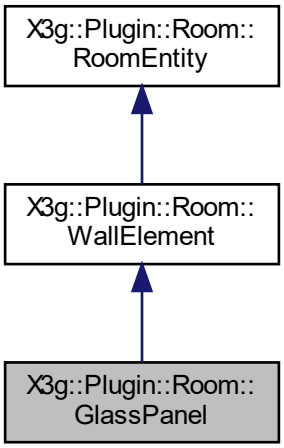
6.192.4.2 String^ X3g::Plugin::Room::Floor::Id [get]

Unqiue id of this floor within the current document.

6.193 X3g::Plugin::Room::GlassPanel Class Reference

A glass panel

Inheritance diagram for X3g::Plugin::Room::GlassPanel:



Public Member Functions

- GlassPanel (X3g::Plugin::Room::Wall^ pWall, GeVec3d pPos)

  Creates a glass panel at given wall and position
- GlassPanel (X3g::Plugin::Room::PolyWall^ pWall, GeVec3d pPos)

  Creates a glass panel at given wall and position

Properties

- double FrameDepth `[get]`

  Returns the frame depth of this glass panel.
- double FrameThickness `[get]`

  Returns the frame thickness of this glass panel.

6.193.1 Detailed Description

A glass panel

6.193.2 Constructor & Destructor Documentation

6.193.2.1 X3g::Plugin::Room::GlassPanel::GlassPanel (X3g::Plugin::Room::Wall^ *pWall,* GeVec3d *pPos*)

Creates a glass panel at given wall and position

6.193.2.2 X3g::Plugin::Room::GlassPanel::GlassPanel (X3g::Plugin::Room::PolyWall^ *pWall,* GeVec3d *pPos*)

Creates a glass panel at given wall and position

6.193.3 Property Documentation

6.193.3.1 double X3g::Plugin::Room::GlassPanel::FrameDepth [get]

Returns the frame depth of this glass panel.

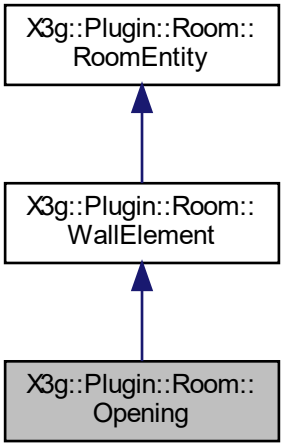
6.193.3.2 double X3g::Plugin::Room::GlassPanel::FrameThickness [get]

Returns the frame thickness of this glass panel.

6.194 X3g::Plugin::Room::Opening Class Reference

An opening.

Inheritance diagram for X3g::Plugin::Room::Opening:



Public Member Functions

- Opening (X3g::Plugin::Room::Wall^ pWall, GeVec3d pPos)

  Creates an opening at specified wall and specified position
- Opening (X3g::Plugin::Room::PolyWall^ pWall, GeVec3d pPos)

  Creates an opening at specified wall and specified position

Additional Inherited Members

6.194.1 Detailed Description

An opening.

6.194.2 Constructor & Destructor Documentation

6.194.2.1 X3g::Plugin::Room::Opening::Opening (X3g::Plugin::Room::Wall^ *pWall,* GeVec3d *pPos*)

Creates an opening at specified wall and specified position

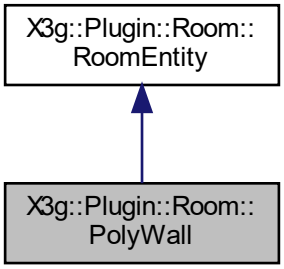
6.194.2.2 X3g::Plugin::Room::Opening::Opening (X3g::Plugin::Room::PolyWall^ *pWall,* GeVec3d *pPos*)

Creates an opening at specified wall and specified position

6.195 X3g::Plugin::Room::PolyWall Class Reference

A PolyWall.

Inheritance diagram for X3g::Plugin::Room::PolyWall:



Public Member Functions

- PolyWall (array< GeVec2d >^pVertices)

  Creates a polywall with the given edge points
- PolyWall (array< GeVec2d >^pVertices, double pHeight)

  Creates a polywall with the given edge points and wall height
- PolyWall (array< GeVec2d >^pVertices, double pHeight, double pLevelHeight)

  Creates a polywall with the given edge points, wall height and floor level height
- array< GeVec2d >^ GetEdges ()

  Returns the edges of this polywall.
- array< WallElement^ >^ GetWallElements ()

  Returns all wall elements of this wall.

Properties

- String^ Id `[get]`

  Unqiue id of this wall within the current document.
- int Level `[get]`

  Returns the level of this wall.
- double Elevation `[get]`

  Returns the elevation of this wall.

6.195.1 Detailed Description

A PolyWall.

6.195.2 Constructor & Destructor Documentation

6.195.2.1 X3g::Plugin::Room::PolyWall::PolyWall (array< GeVec2d >^ *pVertices*)

Creates a polywall with the given edge points

6.195.2.2 X3g::Plugin::Room::PolyWall::PolyWall (array< GeVec2d >^ *pVertices,* double *pHeight*)

Creates a polywall with the given edge points and wall height

6.195.2.3 X3g::Plugin::Room::PolyWall::PolyWall (array< GeVec2d >^ *pVertices,* double *pHeight,* double *pLevelHeight*)

Creates a polywall with the given edge points, wall height and floor level height

6.195.3 Member Function Documentation

6.195.3.1 array< GeVec2d > X3g::Plugin::Room::PolyWall::GetEdges ( )

Returns the edges of this polywall.

6.195.3.2 array< WallElement^ > X3g::Plugin::Room::PolyWall::GetWallElements ( )

Returns all wall elements of this wall.

6.195.4 Property Documentation

6.195.4.1 double X3g::Plugin::Room::PolyWall::Elevation [get]

Returns the elevation of this wall.

6.195.4.2 String^ X3g::Plugin::Room::PolyWall::Id [get]

Unqiue id of this wall within the current document.

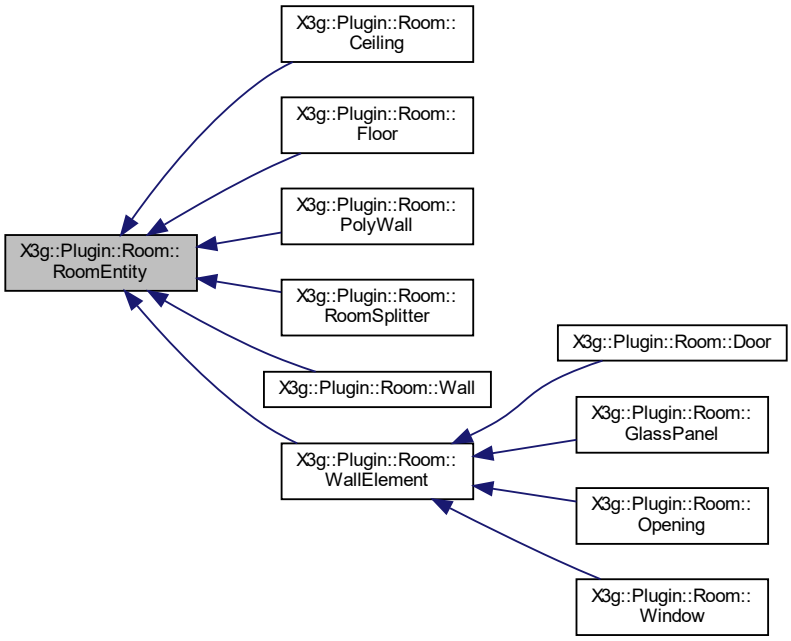
6.195.4.3 int X3g::Plugin::Room::PolyWall::Level [get]

Returns the level of this wall.

6.196 X3g::Plugin::Room::RoomEntity Class Reference

Common base class for room entities.

Inheritance diagram for X3g::Plugin::Room::RoomEntity:



Properties

- String^ Id [get]

  Unqiue id of the entity within the current document.

6.196.1 Detailed Description

Common base class for room entities.

6.196.2 Property Documentation

6.196.2.1 String^ X3g::Plugin::Room::RoomEntity::Id [get]

Unqiue id of the entity within the current document.

6.197 X3g::Plugin::Room::RoomModule Class Reference

Room Manager.

Public Member Functions

- array< Ceiling^ >^ GetCeilings ()

  Returns all ceilings.
- array< Floor^ >^ GetFloors ()

Returns all floors.

- array< RoomSplitter^ >^ GetRoomSplitter ()

  Returns all room splitter.

Properties

- RoomSettings^ Settings [get]

  Room default settings.
- Room::WallManager^ WallManager [get]

  The wall manager contains all walls and wall elements.

6.197.1 Detailed Description

Room Manager.

6.197.2 Member Function Documentation

6.197.2.1 array< Ceiling^ > X3g::Plugin::Room::RoomModule::GetCeilings ( )

Returns all ceilings.

6.197.2.2 array< Floor^ > X3g::Plugin::Room::RoomModule::GetFloors ( )

Returns all floors.

6.197.2.3 array< RoomSplitter^ > X3g::Plugin::Room::RoomModule::GetRoomSplitter ( )

Returns all room splitter.

6.197.3 Property Documentation

6.197.3.1 RoomSettings^ X3g::Plugin::Room::RoomModule::Settings [get]

Room default settings.

6.197.3.2 Room:: WallManager^ X3g::Plugin::Room::RoomModule::WallManager [get]

The wall manager contains all walls and wall elements.

6.198 X3g::Plugin::Room::RoomSettings Class Reference

Room default settings.

Properties

- double WallHeight [get]

  Default wall height.

6.198.1 Detailed Description

Room default settings.

6.198.2 Property Documentation

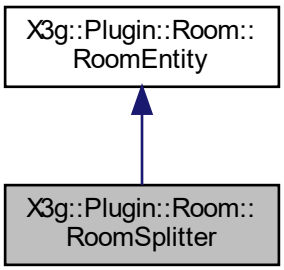
6.198.2.1 double X3g::Plugin::Room::RoomSettings::WallHeight [get]

Default wall height.

6.199 X3g::Plugin::Room::RoomSplitter Class Reference

Divide rooms conceptually not physically.

Inheritance diagram for X3g::Plugin::Room::RoomSplitter:



Public Member Functions

- GeVec2d GetPointOnArc (double progress)

  Get the geometry of an arc, one point at a time.

Properties

- String^ Id [get]

  Unqiue id of this floor within the current document.
- int Level [get]

  Returns the level of this room splitter.
- GeVec2d Start [get, set]

  Start position of the given room splitter
- GeVec2d End [get, set]

  End position of the given room splitter
- GeVec2d MidPoint [get, set]

  The mid point.
- bool IsArc [get]

  Get the information if this room splitter forms a straight line or an arc

6.199.1 Detailed Description

Divide rooms conceptually not physically.

6.199.2 Member Function Documentation

6.199.2.1 GeVec2d X3g::Plugin::Room::RoomSplitter::GetPointOnArc (double *progress*)

Get the geometry of an arc, one point at a time.

Parameters

progress	Interpolate between 0 (start) and 1 (end) and retrieve the position.

Returns

The requested point in the interpolation.

6.199.3 Property Documentation

6.199.3.1 GeVec2d X3g::Plugin::Room::RoomSplitter::End [get],[set]

End position of the given room splitter

6.199.3.2 String^ X3g::Plugin::Room::RoomSplitter::Id [get]

Unqiue id of this floor within the current document.

6.199.3.3 bool X3g::Plugin::Room::RoomSplitter::IsArc [get]

Get the information if this room splitter forms a straight line or an arc

6.199.3.4 int X3g::Plugin::Room::RoomSplitter::Level [get]

Returns the level of this room splitter.

6.199.3.5 GeVec2d X3g::Plugin::Room::RoomSplitter::MidPoint [get],[set]

The mid point.

Relevant to make or identify a room splitter with the shape of an arc.

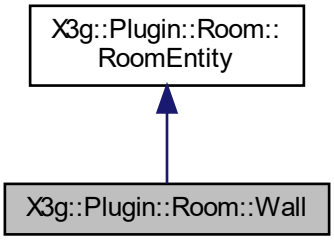
6.199.3.6 GeVec2d X3g::Plugin::Room::RoomSplitter::Start [get],[set]

Start position of the given room splitter

6.200 X3g::Plugin::Room::Wall Class Reference

A Wall.

Inheritance diagram for X3g::Plugin::Room::Wall:



Public Member Functions

- Wall (GeVec2d pStart, GeVec2d pEnd)

  Creates a wall within the current document.
- Wall (GeVec2d pStart, GeVec2d pEnd, double pLevelHeight)

  Creates a wall within the current document at given Level.
- void GetEdges ([Out] GeVec2d% pRightStart,[Out] GeVec2d% pRightEnd,[Out] GeVec2d% pLeftStart,[Out] GeVec2d% pLeftEnd)

  Returns the edges of this wall.
- array< WallElement^ >^ GetWallElements ()

  Returns all wall elements of this wall.

Properties

- String^ Id `[get]`

  Unqiue id of this wall within the current document.
- GeVec2d Start `[get, set]`

  Returns the start point of this wall.
- GeVec2d End `[get, set]`

  Returns the end point of this wall.
- int Level `[get]`

  Returns the level of this wall.
- double Elevation `[get]`

  Returns the elevation of this wall.
- double HeightStart `[get, set]`

  Returns the height at start point.
- double HeightEnd `[get, set]`

  Returns the height at end point.
- double Thickness `[get, set]`

  Returns the thickness of this wall.
- WallSide InnerSide `[get]`

  Returns the inner side of this wall.
- Wall^ WallAtStart `[get, set]`

  Returns the wall connected at start point.
- Wall^ WallAtEnd `[get, set]`

  Returns the wall connected at end point.

6.200.1 Detailed Description

A Wall.

6.200.2 Constructor & Destructor Documentation

6.200.2.1 X3g::Plugin::Room::Wall::Wall (GeVec2d *pStart,* GeVec2d *pEnd*)

Creates a wall within the current document.

6.200.2.2 X3g::Plugin::Room::Wall::Wall (GeVec2d *pStart,* GeVec2d *pEnd,* double *pLevelHeight*)

Creates a wall within the current document at given Level.

6.200.3 Member Function Documentation

6.200.3.1 void X3g::Plugin::Room::Wall::GetEdges ([Out] GeVec2d% *pRightStart,* [Out] GeVec2d% *pRightEnd,* [Out] GeVec2d% *pLeftStart,* [Out] GeVec2d% *pLeftEnd*)

Returns the edges of this wall.

6.200.3.2 array< WallElement^ > X3g::Plugin::Room::Wall::GetWallElements ( )

Returns all wall elements of this wall.

6.200.4 Property Documentation

6.200.4.1 double X3g::Plugin::Room::Wall::Elevation [get]

Returns the elevation of this wall.

6.200.4.2 GeVec2d X3g::Plugin::Room::Wall::End [get],[set]

Returns the end point of this wall.

6.200.4.3 double X3g::Plugin::Room::Wall::HeightEnd [get],[set]

Returns the height at end point.

6.200.4.4 double X3g::Plugin::Room::Wall::HeightStart [get],[set]

Returns the height at start point.

6.200.4.5 String^ X3g::Plugin::Room::Wall::Id [get]

Unqiue id of this wall within the current document.

6.200.4.6 WallSide X3g::Plugin::Room::Wall::InnerSide [get]

Returns the inner side of this wall.

Considered from start to end point.

6.200.4.7 int X3g::Plugin::Room::Wall::Level [get]

Returns the level of this wall.

6.200.4.8 GeVec2d X3g::Plugin::Room::Wall::Start [get], [set]

Returns the start point of this wall.

6.200.4.9 double X3g::Plugin::Room::Wall::Thickness [get], [set]

Returns the thickness of this wall.

6.200.4.10 Wall^ X3g::Plugin::Room::Wall::WallAtEnd [get], [set]

Returns the wall connected at end point.

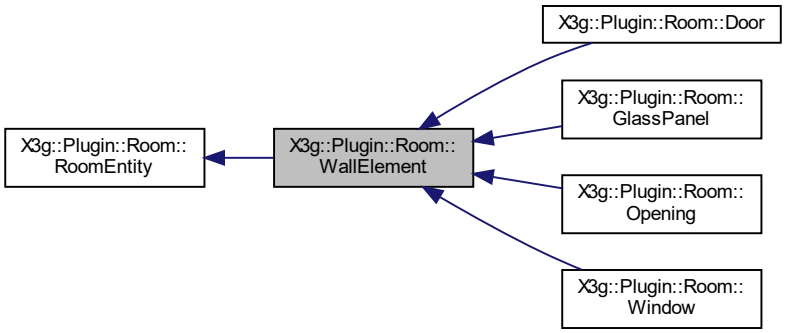
6.200.4.11 Wall^ X3g::Plugin::Room::Wall::WallAtStart [get], [set]

Returns the wall connected at start point.

6.201 X3g::Plugin::Room::WallElement Class Reference

A wall element.

Inheritance diagram for X3g::Plugin::Room::WallElement:



Public Member Functions

- void GetEdges ([Out] GeVec2d% pRightStart,[Out] GeVec2d% pRightEnd,[Out] GeVec2d% pLeftStart,[Out] GeVec2d% pLeftEnd)

Returns the edges of this wall element.

Properties

- String^ Id [get]

  Unqiue id of this wall element within the current document.
- Wall^ Wall [get]

  Returns the connected wall.
- GeVec2d Position [get, set]

  Returns the position of this wall element.
- double Width [get, set]

  Returns ands sets the width of this wall element.
- double Height [get, set]

  Returns and sets the height of this wall element.
- double DistanceToFloor [get, set]

  Returns the distance to the floor of this wall element.
- bool Flipped [get, set]

  Returns true if wall element is flipped in relation to the wall side.
- bool Mirrored [get, set]

  Returns true if the wall element is mirrored along the wall direction.

6.201.1 Detailed Description

A wall element.

6.201.2 Member Function Documentation

6.201.2.1 void X3g::Plugin::Room::WallElement::GetEdges ([Out] GeVec2d% *pRightStart*, [Out] GeVec2d% *pRightEnd*, [Out] GeVec2d% *pLeftStart*, [Out] GeVec2d% *pLeftEnd*)

Returns the edges of this wall element.

6.201.3 Property Documentation

6.201.3.1 double X3g::Plugin::Room::WallElement::DistanceToFloor [get],[set]

Returns the distance to the floor of this wall element.

6.201.3.2 bool X3g::Plugin::Room::WallElement::Flipped [get],[set]

Returns true if wall element is flipped in relation to the wall side.

6.201.3.3 double X3g::Plugin::Room::WallElement::Height [get],[set]

Returns and sets the height of this wall element.

6.201.3.4 String^ X3g::Plugin::Room::WallElement::Id [get]

Unqiue id of this wall element within the current document.

6.201.3.5 bool X3g::Plugin::Room::WallElement::Mirrored [get], [set]

Returns true if the wall element is mirrored along the wall direction.

Mirroring doesn't change orientation in relation to the wall side.

6.201.3.6 GeVec2d X3g::Plugin::Room::WallElement::Position [get], [set]

Returns the position of this wall element.

6.201.3.7 Wall^ X3g::Plugin::Room::WallElement::Wall [get]

Returns the connected wall.

6.201.3.8 double X3g::Plugin::Room::WallElement::Width [get], [set]

Returns ands sets the width of this wall element.

6.202 X3g::Plugin::Room::WallManager Class Reference

Base class for all walls and wall elements.

Public Member Functions

- double GetElevation (int pLevel)

  Returns the elevation to a given level.
- array< Wall^ >^ GetWalls (int pLevel)

  Returns all walls of a level.
- array< PolyWall^ >^ GetPolyWalls (int pLevel)

  Returns all polywalls of a level.

Properties

- int LevelCount [get]

  The number of levels.

6.202.1 Detailed Description

Base class for all walls and wall elements.

6.202.2 Member Function Documentation

6.202.2.1 double X3g::Plugin::Room::WallManager::GetElevation (int *pLevel*)

Returns the elevation to a given level.

6.202.2.2 array< PolyWall^ > X3g::Plugin::Room::WallManager::GetPolyWalls (int *pLevel*)

Returns all polywalls of a level.

6.202.2.3 array< Wall^ > X3g::Plugin::Room::WallManager::GetWalls (int *pLevel*)

Returns all walls of a level.

6.202.3 Property Documentation

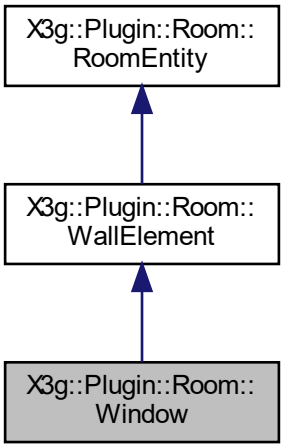
6.202.3.1 int X3g::Plugin::Room::WallManager::LevelCount [get]

The number of levels.

6.203 X3g::Plugin::Room::Window Class Reference

A window.

Inheritance diagram for X3g::Plugin::Room::Window:



Public Member Functions

- Window (X3g::Plugin::Room::Wall^ pWall, GeVec3d pPos)

  Creates a window at given wall and position.
- Window (X3g::Plugin::Room::PolyWall^ pWall, GeVec3d pPos)

  Creates a window at given wall and position.

Properties

- double FrameDepth [get]

  Returns the frame depth of this window.
- double FrameThickness [get]

  Returns the frame thickness of this window.
- bool SillEnabled [get]

  Returns true if a sill is presented.

6.203.1 Detailed Description

A window.

6.203.2 Constructor & Destructor Documentation

6.203.2.1 X3g::Plugin::Room::Window::Window (X3g::Plugin::Room::Wall^ *pWall,* GeVec3d *pPos*)

Creates a window at given wall and position.

6.203.2.2 X3g::Plugin::Room::Window::Window (X3g::Plugin::Room::PolyWall^ *pWall,* GeVec3d *pPos*)

Creates a window at given wall and position.

6.203.3 Property Documentation

6.203.3.1 double X3g::Plugin::Room::Window::FrameDepth [get]

Returns the frame depth of this window.

6.203.3.2 double X3g::Plugin::Room::Window::FrameThickness [get]

Returns the frame thickness of this window.

6.203.3.3 bool X3g::Plugin::Room::Window::SillEnabled [get]

Returns true if a sill is presented.

6.204 X3g::Plugin::SaveBlockParams Struct Reference

Block export parameters.

Public Attributes

- DocumentFileFormat Format

  DWG format version.
- bool ClearMaterials

  Save block without materials.
- bool ClearLayers

  Set all entities to layer zero.
- bool ResolveBlockRefs

  Resolve block references.
- Dictionary< String^ , String^ >^ LayerMapping

  Rename layers in saved block.

6.204.1 Detailed Description

Block export parameters.

6.204.2 Member Data Documentation

6.204.2.1 bool X3g::Plugin::SaveBlockParams::ClearLayers

Set all entities to layer zero.

6.204.2.2 bool X3g::Plugin::SaveBlockParams::ClearMaterials

Save block without materials.

6.204.2.3 DocumentFileFormat X3g::Plugin::SaveBlockParams::Format

DWG format version.

6.204.2.4 Dictionary<String^, String^> ^ X3g::Plugin::SaveBlockParams::LayerMapping

Rename layers in saved block.

Keys are source names, values are target names.

6.204.2.5 bool X3g::Plugin::SaveBlockParams::ResolveBlockRefs

Resolve block references.

6.205 X3g::Plugin::SaveEventInfo Class Reference

Parameter for the save events.

Public Member Functions

- SaveEventInfo (System::String^ pFileName, SaveMode pMode)

  Constructor to initialize the FileName and the Mode.

Public Attributes

- System::String^ FileName

  Target file path.
- SaveMode Mode

  The save mode.

6.205.1 Detailed Description

Parameter for the save events.

6.205.2 Constructor & Destructor Documentation

6.205.2.1 X3g::Plugin::SaveEventInfo::SaveEventInfo (System::String^ *pFileName,* SaveMode *pMode*)

Constructor to initialize the FileName and the Mode.

6.205.3 Member Data Documentation

6.205.3.1 System::String ^ X3g::Plugin::SaveEventInfo::FileName

Target file path.

6.205.3.2 SaveMode X3g::Plugin::SaveEventInfo::Mode

The save mode.

6.206 X3g::Plugin::SoundAbsorption Struct Reference

Sound absorption values.

Public Attributes

- float Frequency

  Frequency.

- float Value

  Value.

6.206.1 Detailed Description

Sound absorption values.

6.206.2 Member Data Documentation

6.206.2.1 float X3g::Plugin::SoundAbsorption::Frequency

Frequency.

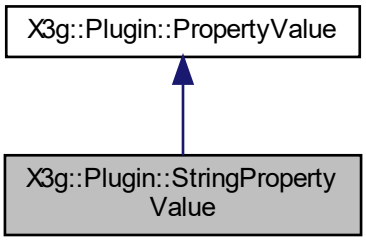
6.206.2.2 float X3g::Plugin::SoundAbsorption::Value

Value.

6.207 X3g::Plugin::StringPropertyValue Class Reference

String Property Value.

Inheritance diagram for X3g::Plugin::StringPropertyValue:



Public Member Functions

- StringPropertyValue (System::String^ pValue)

  Constructor.

Properties

- System::String^ Value `[get, set]`

  The string value.

6.207.1 Detailed Description

String Property Value.

6.207.2 Constructor & Destructor Documentation

6.207.2.1 X3g::Plugin::StringPropertyValue::StringPropertyValue (System::String^ *pValue*)

Constructor.

Parameters

pValue	Initial string value.

6.207.3 Property Documentation

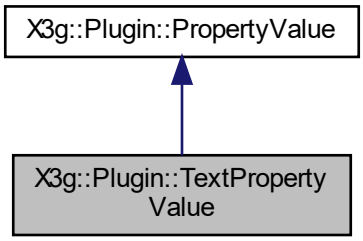
6.207.3.1 System:: String^ X3g::Plugin::StringPropertyValue::Value `[get],[set]`

The string value.

6.208 X3g::Plugin::TextPropertyValue Class Reference

Text (multi-line string) Property Value.

Inheritance diagram for X3g::Plugin::TextPropertyValue:



Public Member Functions

- TextPropertyValue (System::String^ pValue)

  Constructor.

Properties

- System::String^ Value `[get, set]`

  The text value.

6.208.1 Detailed Description

Text (multi-line string) Property Value.

6.208.2 Constructor & Destructor Documentation

6.208.2.1 X3g::Plugin::TextPropertyValue::TextPropertyValue (System::String^ *pValue*)

Constructor.

Parameters

pValue	Initial text value.

6.208.3 Property Documentation

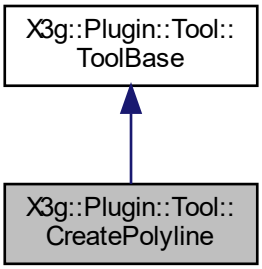
6.208.3.1 System:: String^ X3g::Plugin::TextPropertyValue::Value `[get],[set]`

The text value.

6.209 X3g::Plugin::Tool::CreatePolyline Class Reference

This tool let the user create a new polyline.

Inheritance diagram for X3g::Plugin::Tool::CreatePolyline:



Public Member Functions

- CreatePolyline ()

  The user will create a non filled polyline without a point limit.
- CreatePolyline (bool pCreateFilled, int pMaxPoints)

  The user will create a new polyline.

Additional Inherited Members

6.209.1 Detailed Description

This tool let the user create a new polyline.

6.209.2 Constructor & Destructor Documentation

6.209.2.1 X3g::Plugin::Tool::CreatePolyline::CreatePolyline ( )

The user will create a non filled polyline without a point limit.

6.209.2.2 X3g::Plugin::Tool::CreatePolyline::CreatePolyline (bool *pCreateFilled*, int *pMaxPoints*)

The user will create a new polyline.

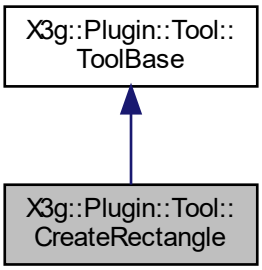
Parameters

pCreateFilled	If true the polyline will be created as a region but only if the created polyline is closed.
pMaxPoints	Defines the max number of vertices of the polyline. If the parameter is less than 2 it will be ignored.

6.210 X3g::Plugin::Tool::CreateRectangle Class Reference

This tool let the user create a new rectangle.

Inheritance diagram for X3g::Plugin::Tool::CreateRectangle:



Public Member Functions

- CreateRectangle ()

  The user will create a non filled rectangle out of two points.
- CreateRectangle (bool pCreateFilled, bool pPreferRotatedCreation)

  The user will create a new rectangle.

Additional Inherited Members

6.210.1 Detailed Description

This tool let the user create a new rectangle.

6.210.2 Constructor & Destructor Documentation

6.210.2.1 X3g::Plugin::Tool::CreateRectangle::CreateRectangle ( )

The user will create a non filled rectangle out of two points.

This is the same as RectangleTool(false, false).

6.210.2.2 X3g::Plugin::Tool::CreateRectangle::CreateRectangle (bool *pCreateFilled*, bool *pPreferRotatedCreation*)

The user will create a new rectangle.

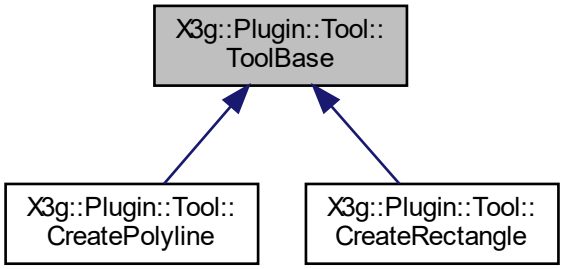
Parameters

pCreateFilled	If true the rectangle will be created as a region.
pPreferRotated-Creation	If true the user define 3 points to create the rectangle.

6.211 X3g::Plugin::Tool::ToolBase Class Reference

Base class of internal tools.

Inheritance diagram for X3g::Plugin::Tool::ToolBase:



Public Member Functions

- bool Start ()

  Try to start the tool.

Properties

- DbEntity^ CreatedEntity `[get]`

  Returns the created entity.
- bool IsRunning `[get]`

  True if the tool is running.

Events

- ToolCallbackEventHandler^ ToolUpdated

  Occurs when tool finished work.

6.211.1 Detailed Description

Base class of internal tools.

6.211.2 Member Function Documentation

6.211.2.1 bool X3g::Plugin::Tool::ToolBase::Start ( )

Try to start the tool.

On success true is returned.

6.211.3 Property Documentation

6.211.3.1 DbEntity^ X3g::Plugin::Tool::ToolBase::CreatedEntity `[get]`

Returns the created entity.

6.211.3.2 bool X3g::Plugin::Tool::ToolBase::IsRunning [get]

True if the tool is running.

6.211.4 Event Documentation

6.211.4.1 ToolCallbackEventHandler^ X3g::Plugin::Tool::ToolBase::ToolUpdated

Occurs when tool finished work.

6.212 X3g::Plugin::Tool::ToolCallbackEventArgs Class Reference

Event args of ToolBase callback.

Inherits EventArgs.

Properties

- DbEntity^ CreatedEntity [get]

  Returns the created entity.
- bool Aborted [get]

  True if the tool was aborted.

6.212.1 Detailed Description

Event args of ToolBase callback.

6.212.2 Property Documentation

6.212.2.1 bool X3g::Plugin::Tool::ToolCallbackEventArgs::Aborted [get]

True if the tool was aborted.

6.212.2.2 DbEntity^ X3g::Plugin::Tool::ToolCallbackEventArgs::CreatedEntity [get]

Returns the created entity.

6.213 X3g::Plugin::View Class Reference

A View.

Public Member Functions

- ViewSpecificLayerVisibility GetLayerVisibility (Layer^ pLayer)

  Returns weather Layervisibility is enabled/disabled ort global for this view.
- void SetLayerVisibility (Layer^ pLayer, ViewSpecificLayerVisibility pVisible)

  If isVisibilityPerViewEnabled is enabled, it sets layer visible/invisible/global for this view.

Properties

- int Index [get]

  The index of this view.
- int Width [get]

  The width this view.
- int Height [get]

  The height this view.
- ViewCameraMode CameraMode [get, set]

  The camera mode of this view.
- ICamera^ Camera [get]

  Interface to camera parameters of the view.
- CameraEntity^ AssignedCameraEntity [get, set]

  Interface to camera entity to which the view is currently set.
- IRenderStyle^ RenderStyle [get, set]

  Active render style of view.

Events

- System::EventHandler^ CameraChanged [add, remove, raise]

  Occurs when camera parameters have changed.
- System::EventHandler^ RenderStyleChanged [add, remove, raise]

  Occurs when active render style has changed.
- System::EventHandler^ Resize [add, remove, raise]

  Occurs when view size changed.

6.213.1 Detailed Description

A View.

6.213.2 Member Function Documentation

6.213.2.1 ViewSpecificLayerVisibility X3g::Plugin::View::GetLayerVisibility (Layer^ *pLayer*)

Returns weather Layervisibility is enabled/disabled ort global for this view.

6.213.2.2 void X3g::Plugin::View::SetLayerVisibility (Layer^ *pLayer*, ViewSpecificLayerVisibility *pVisible*)

If isVisibilityPerViewEnabled is enabled, it sets layer visible/invisible/global for this view.

6.213.3 Property Documentation

6.213.3.1 CameraEntity^ X3g::Plugin::View::AssignedCameraEntity [get],[set]

Interface to camera entity to which the view is currently set.

May be null.

6.213.3.2 ICamera^ X3g::Plugin::View::Camera [get]

Interface to camera parameters of the view.

It's not allowed to assign another camera reference. Use ICamera::CopyFrom() instead.

6.213.3.3 ViewCameraMode X3g::Plugin::View::CameraMode [get],[set]

The camera mode of this view.

6.213.3.4 int X3g::Plugin::View::Height [get]

The height this view.

(pixel)

6.213.3.5 int X3g::Plugin::View::Index [get]

The index of this view.

6.213.3.6 IRenderStyle^ X3g::Plugin::View::RenderStyle [get],[set]

Active render style of view.

6.213.3.7 int X3g::Plugin::View::Width [get]

The width this view.

(pixel)

6.213.4 Event Documentation

6.213.4.1 System:: EventHandler^ X3g::Plugin::View::CameraChanged [add],[remove],[raise]

Occurs when camera parameters have changed.

6.213.4.2 System:: EventHandler^ X3g::Plugin::View::RenderStyleChanged [add],[remove],[raise]

Occurs when active render style has changed.

6.213.4.3 System:: EventHandler^ X3g::Plugin::View::Resize [add],[remove],[raise]

Occurs when view size changed.

6.214 X3g::Plugin::ViewManager Class Reference

View manager.

Public Member Functions

- View^ GetView (int pIndex)

  Returns the view with the given index.
- void RenderFrame ()

  Manually triggers a rendering of a frame.

Properties

- BackgroundParams^ Background `[get]`

  Background settings used for all views.
- ViewLayout Layout `[get, set]`

  The layout of the views.
- int ViewCount `[get]`

  The number of the currently active/visible views (this depends on the layout).
- View^ SelectedView `[get, set]`

  The currently selected view.

Events

- DragDropEventHandler^ DragOver `[add, remove, raise]`

  Occurs repeatedly while an object is dragged over any view.
- DragDropEventHandler^ Drop `[add, remove, raise]`

  Occurs once when an object is dropped on any view.
- System::EventHandler^ LayoutChanged `[add, remove, raise]`

  Occurs when layout of the views has changed.
- System::EventHandler^ BackgroundChanged `[add, remove, raise]`

  Occurs when a background parameter has changed.
- System::EventHandler^ Rendering `[add, remove, raise]`

  Occurs when viewer is starting to render a frame.

6.214.1 Detailed Description

View manager.

6.214.2 Member Function Documentation

6.214.2.1 View X3g::Plugin::ViewManager::GetView (int *pIndex*)

Returns the view with the given index.

6.214.2.2 void X3g::Plugin::ViewManager::RenderFrame ( )

Manually triggers a rendering of a frame.

6.214.3 Property Documentation

6.214.3.1 BackgroundParams^ X3g::Plugin::ViewManager::Background `[get]`

Background settings used for all views.

6.214.3.2 ViewLayout X3g::Plugin::ViewManager::Layout [get],[set]

The layout of the views.

6.214.3.3 View^ X3g::Plugin::ViewManager::SelectedView [get],[set]

The currently selected view.

6.214.3.4 int X3g::Plugin::ViewManager::ViewCount [get]

The number of the currently active/visible views (this depends on the layout).

6.214.4 Event Documentation

6.214.4.1 System:: EventHandler^ X3g::Plugin::ViewManager::BackgroundChanged [add],[remove],[raise]

Occurs when a background parameter has changed.

6.214.4.2 DragDropEventHandler^ X3g::Plugin::ViewManager::DragOver [add],[remove],[raise]

Occurs repeatedly while an object is dragged over any view.

Listeners call DragDropEventArgs::AcceptDrop if they want to handle drop of data.

6.214.4.3 DragDropEventHandler^ X3g::Plugin::ViewManager::Drop [add],[remove],[raise]

Occurs once when an object is dropped on any view.

Listeners call DragDropEventArgs::AcceptDrop if drop was successfully handled.

6.214.4.4 System:: EventHandler^ X3g::Plugin::ViewManager::LayoutChanged [add],[remove],[raise]

Occurs when layout of the views has changed.

6.214.4.5 System:: EventHandler^ X3g::Plugin::ViewManager::Rendering [add],[remove],[raise]

Occurs when viewer is starting to render a frame.

6.215 X3g::Plugin::Widget Class Reference

Allows to manage widgets of pCon.planner.

Properties

- System::String^ Key [get]

  Returns key which identifies the widget.
- System::String^ Value [get, set]

  Gets or sets the widget value.
- System::String^ Caption [get, set]

  Gets or sets the widget caption.

- System::String^ Hint [get, set]

  Gets or sets the widget hint.
- bool Enabled [get, set]

  Gets or sets if the widget is enabled.
- bool Visible [get, set]

  Gets or sets the widgets visibility.

Events

- ValueEvent^ ValueChanged

  This event fires if the changed value is applied by the user.
- ValueEvent^ ValueChanging

  This event fires if the value is changing.

6.215.1 Detailed Description

Allows to manage widgets of pCon.planner.

6.215.2 Property Documentation

6.215.2.1 System:: String^ X3g::Plugin::Widget::Caption [get],[set]

Gets or sets the widget caption.

6.215.2.2 bool X3g::Plugin::Widget::Enabled [get],[set]

Gets or sets if the widget is enabled.

6.215.2.3 System:: String^ X3g::Plugin::Widget::Hint [get],[set]

Gets or sets the widget hint.

6.215.2.4 System:: String^ X3g::Plugin::Widget::Key [get]

Returns key which identifies the widget.

6.215.2.5 System:: String^ X3g::Plugin::Widget::Value [get],[set]

Gets or sets the widget value.

6.215.2.6 bool X3g::Plugin::Widget::Visible [get],[set]

Gets or sets the widgets visibility.

Changing visibility is high cost operation on GUI side. Better avoid it - changing the UI layout at runtime is bad style anyway.

6.215.3 Event Documentation

6.215.3.1 ValueEvent^ X3g::Plugin::Widget::ValueChanged

This event fires if the changed value is applied by the user.

6.215.3.2 ValueEvent^ X3g::Plugin::Widget::ValueChanging

This event fires if the value is changing.

Attention: Removing the focus from the widget in this event will cancel the current intput and reset the value.

6.216 X3g::Plugin::Window Class Reference

Plugin host window.

Public Member Functions

- ~Window ()

  Call Dispose() if the window shall be destroyed.
- void SetInitialClientArea (double pWidth, double pHeight)

  Sets initial size of window client area in device-independent units.

Properties

- Object^ Content `[get, set]`

  Allows to access the window content.
- System::String^ Name `[get]`

  Returns name which identifies the window.
- System::String^ Title `[get, set]`

  Gets or sets the window caption.
- bool Visible `[get, set]`

  Gets or sets the window visibility.

6.216.1 Detailed Description

Plugin host window.

6.216.2 Constructor & Destructor Documentation

6.216.2.1 X3g::Plugin::Window::~Window ( )

Call Dispose() if the window shall be destroyed.

6.216.3 Member Function Documentation

6.216.3.1 void X3g::Plugin::Window::SetInitialClientArea (double *pWidth*, double *pHeight*)

Sets initial size of window client area in device-independent units.

The initial size of the client area determines the size of a window which becomes visible for the first time. The method doesn't have any effect when called later. Also it has no effect on windows which were restored on application startup. The method should be called within Plugin::X3gSetWindowContent. If initial size of client area is not set explicitely it will be derived from content size. If content has no valid size the window will be created with a default size.

6.216.4 Property Documentation

6.216.4.1 Object^ X3g::Plugin::Window::Content [get],[set]

Allows to access the window content.

Usually set within X3gSetWindowContent

6.216.4.2 System:: String^ X3g::Plugin::Window::Name [get]

Returns name which identifies the window.

Window name is unique within a plugin.

6.216.4.3 System:: String^ X3g::Plugin::Window::Title [get],[set]

Gets or sets the window caption.

6.216.4.4 bool X3g::Plugin::Window::Visible [get],[set]

Gets or sets the window visibility.

6.217 X3g::Plugin::WindowManager Class Reference

Allows plugins to create and manage dockable windows.

Public Member Functions

- Window^ CreateDockableWindow (System::String^ pName)

  Creates a dockable host window and requests plugin to set the content.
- Window^ GetWindow (System::String^ pName)

  Returns existing window or null.
- Widget^ GetWidget (System::String^ pKey)

  Returns existing widget or null.

6.217.1 Detailed Description

Allows plugins to create and manage dockable windows.

6.217.2 Member Function Documentation

6.217.2.1 Window X3g::Plugin::WindowManager::CreateDockableWindow (System::String^ *pName*)

Creates a dockable host window and requests plugin to set the content.

Exceptions

System::ArgumentNullException	Window name is null or empty.
System::ArgumentException	Window with this name already exists.

Name must be unique within the plugin. Created window is invisible by default. Returns null if window is not supported by Plugin::X3gSetWindowContent.

See Also

Plugin::X3gSetWindowContent

6.217.2.2 Widget X3g::Plugin::WindowManager::GetWidget (System::String^ *pKey*)

Returns existing widget or null.

6.217.2.3 Window X3g::Plugin::WindowManager::GetWindow (System::String^ *pName*)

Returns existing window or null.

Index