Introduction

The AutoCAD Map 3D SDK includes ObjectARX, ObjectARX .NET, Feature Data Objects (FDO), and Geospatial Platform .NET.

ObjectARX is an unmanaged C++ API. ObjectARX .NET is the managed C++ equivalent.

FDO, like ObjectARX, comprises both a C++ API and a managed C++ equivalent.

The Geospatial Platform is exposed as .NET only.

Topics in This Section

AutoCAD Map 3D 2009 SDK

The AutoCAD Map 3D SDK includes ObjectARX, Geospatial Platform, and Feature Data Objects (FDO).

ObjectARX and FDO are exposed as C++ and also .NET. Geospatial Platform is exposed as .NET only.

ObjectARX, Geospatial Platform, and FDO are installed with the AutoCAD Map 3D SDK only. However the documentation for these APIs is included in both AutoCAD Map 3D and AutoCAD Map 3D SDK installations.

SDK Documentation

AutoCAD Map 3D SDK Help and the following component Help files are located in AutoCAD Map 3D\Help and also in AutoCAD Map 3D SDK\Docs. So long as these files reside in the same folder, all component Help content is available on the Contents tab of AutoCAD Map 3D SDK Help when you open it.

Title	File Name
AutoCAD Map 3D ObjectARX C++ Developer's Guide	sdk.arx.cpp.dev.chm
AutoCAD Map 3D ObjectARX C++ Reference	sdk.arx.cpp.ref.chm
AutoCAD Map 3D ObjectARX .NET Developer's	sdk.arx.net.dev.chm

Guide	
AutoCAD Map 3D ObjectARX .NET Reference	sdk.arx.net.ref.chm
AutoCAD Map 3D Geospatial Platform Developer's Guide	sdk.geospatial.platform.dev.chm
AutoCAD Map 3D Geospatial Platform Reference	sdk.geospatial.platform.ref.chm
AutoCAD Map 3D Geospatial Platform Supplement Reference	sdk.geospatial.platform.supplement.ref.chm
AutoCAD Map 3D SDK Readme	sdk._readme.chm

The three developer's guides are also available as PDF files, which are located in the same folder as their CHM equivalents.

In addition to the documentation components with ObjectARX .NET in their titles, note that the ObjectARX C++ Developer's Guide includes a .NET overview. To view this section, open AutoCAD Map 3D DK Help, and then click Developer's Guide > Managed Wrapper Classes.

Documentation for the FDO API, which is also a part of the SDK, is described in its own FDO section below.

SDK Scope

ObjectARX C++ and .NET correspond closely. Each covers most of AutoCAD Map 3D functionality, except what is covered exclusively by Geospatial Platform.

Geospatial Platform coverage includes

- Resource Service
- Feature Service

- Mapping Service

The scope of the FDO API, which is also a part of the SDK, is described in its own FDO section below.

SDK Samples and C++ Header Files

SDK sample projects and C++ header files are installed with the AutoCAD Map 3D SDK only. They are not included with AutoCAD Map 3D.

The sample projects and C++ header files for FDO, which is also a part of the SDK, are described in their own FDO section below.

FDO Documentation

exposed as C++ and also .NET.

Like ObjectARX, FDO is exposed as C++ and also .NET.

The FDO documentation includes the following components. These files are located in AutoCAD Map 3D\Help and also in AutoCAD Map 3D SDK\FDO\Docs.

Title	File Name
The Essential FDO	FET_TheEssentialFDO.pdf
FDO Developer's Guide	FDG_FDODevGuide.pdf
FDO API Reference	FDO_API.chm
FDO API Reference - Managed	FDO_API_managed.chm
FDO SDK Readme	FDO_SDK_ReadMe.chm
FDO Provider for ArcSDE	ArcSDE_Provider_API.chm
FDO Provider for MySQL	MySQL_Provider_API.chm
FDO Provider for ODBC	ODBC_Provider_API.chm

FDO Provider for Oracle	Oracle_Provider_API.chm
FDO Provider for Raster	Raster_Provider_API.chm
FDO Provider for SDF	SDF_Provider_API.chm
FDO Provider for SHP	SHP_Provider_API.chm
FDO Provider for SQL Server	SQLServer_Provider_API.chm
FDO Provider for WMS	WMS_Provider_API.chm

FDO Scope

FDO provides a generic interface to a number of back-end data source technologies for the storage and retrieval of GIS data. It also provides a model for extending this generic interface to additional data source technologies. The implementation of the interface for a particular technology is called an FDO provider.

Besides using the FDO API to develop FDO providers to support additional data source technologies, you can also use it to build applications that connect to and configure FDO providers.

FDO Samples and C++ Header Files

FDO sample projects and C++ header files are installed with the AutoCAD Map 3D SDK only. They are not included with AutoCAD Map 3D.

Before You Begin

To develop applications using AutoCAD Map 3D ObjectARX, you should be familiar with AutoCAD ObjectARX and also the AutoCAD Map 3D and AutoCAD applications.

For information about...	Refer to...
AutoCAD ObjectARX	AutoCAD ObjectARX Documentation, arxdoc.chm, which is located in the docs folder of your ObjectARX installation.
AutoCAD Map 3D and AutoCAD	AutoCAD Map 3D Help, which is located in the Help folder of AutoCAD Map 3D installations.

Important Note AutoCAD Map 3D Help is especially useful for undestanding how AutoCAD Map 3D models its domain, for understanding drawing sets and queries, for example. Since detailed explanations of these paradigms are available in AutoCAD Map 3D Help, AutoCAD Map 3D ObjectARX Help explains them only briefly or not at all. Before you attempt to automate or extend an AutoCAD Map 3D feature, review the subject in AutoCAD Map 3D Help.

Compatability of ObjectARX and AutoCAD Map 3D

AutoCAD Map 3D ObjectARX must be installed in an existing AutoCAD ObjectARX installation, and the AutoCAD Map 3D and AutoCAD ObjectARX versions must be compatible with each other and with the version of AutoCAD Map 3D that you are extending. For example, AutoCAD Map 3D ObjectARX

2007 must be installed into an existing AutoCAD ObjectARX 2007 installation, and you need both APIs to build ObjectARX applications for AutoCAD Map 3D 2007.

Components Not Intended for Public Use

In many cases, but not all, where a component is documented but not intended for public use, this fact is noted in the component description. With the following components, however, this note may not be present. Avoid them nevertheless. They are intended for internal Autodesk use only.

Important Note Be aware of these components. Avoid using them in your applications.

Components to avoid:

AcMapMbTileNameGeneratorGrid Class
AcMapMbTileNameGeneratorData Class
AcMapMbTileNameGeneratorSequence Class
AcMapMbTileNameGenerator Class

Review What's New

Besides highlighting new functionality, the What's New topic also lists ObjectARX components that are not intended for public use, in addition to the components that are already listed in "Components Not Intended for Public Use" above. Even though they are present in public header files and may even be partially documented, they are intended for internal Autodesk use only.

Important Note Be aware of these components. Avoid using them in your applications.

Return Descriptions With Multiple Return Codes Cited

With return descriptions of functions where multiple return codes are cited, note that the list of return codes is not necessarily exhaustive. Undocumented return codes are possible.

For example, the return descriptions for ObjectARX functions FullEdge::GetHalfEdge() and AcDbBasicFilter::FilterObjects() should simply say, "Returns eOk or an error code." Their lists of return codes are lengthy enough that you might think they are exhaustive, but they are not.

Typographic Conventions

How It Looks	What It Is
anteatereater	Text you enter at the AutoCAD Map 3D command prompt.
dirname\filename.ext	Names of files and directories.
dir *.h /b/o:n > dir.txt	Code.

Note All file names and directory paths in AutoCAD Map 3D are case sensitive.

AutoCAD Map 3D 2009 SDK: What's New

What's new or changed since the previous release.

Geospatial Platform API Promoted from Preview to Production

Likewise its documentation:

Title	File Name
AutoCAD Map 3D Geospatial Platform Developer's Guide	sdk.gis.platform.dev.chm sdk.gis.platform.dev.pdf
AutoCAD Map 3D Geospatial Platform Reference	sdk.gis.platform.ref.chm
AutoCAD Map 3D Geospatial Platform Supplement Reference	sdk.gis.platform.supplement.ref.chm

The Geospatial Platform API is exposed as .NET only. It covers the following areas of functionality:

- Resource Service
- Feature Service
- Mapping Service

New Documentation

AutoCAD Map 3D SDK Help, sdk.doc.main.chm, includes two new documentation components.

Title	File Name
AutoCAD Map 3D Developer Samples	sdk.samples.chm sdk.samples.pdf

AutoCAD Map 3D SDK Help and its component Help files are installed in the Docs folder of SDK installations. So long as these files reside in the same folder, all component Help content is available on the Contents tab of AutoCAD Map 3D SDK Help when you open it.

New Sample Projects

Classify
FeatureExplorer
FeatureInspector
QueryAndLocate

Other Changes

The Geospatial Platform API is now more consistent and more efficient. Changes include

- Bridging gaps in the area of interoperating between CAD entities and FDO features. Specifically, we have added methods to AcMapFeatureEntityService to improve support for applications working with the AutoCAD and Geospatial Platform APIs to manage AutoCAD selection sets, checked entities, and FDO features.
- Adding AcMapMap.SetViewScale, which zooms the map to a given scale, to complement the existing AcMapMap.GetViewScale, which merely reports the scale.
- Renaming AcMapApiMgd.dll to Autodesk.Map.Platform.dll to comply with .NET naming conventions.

- Enhancing events to include <ResourceName><OperationName>Canceled events corresponding to <ResourceName>ToBe<OperationName> events. (By <OperationName> we mean operations like Added, Modified, Updated, and so on.) With these events added, applications that do bookkeeping while an operation proceeds can roll back their bookkeeping if the operation fails.
- Adding an MgFeatureQueryOptions parameter to AcMapLayer.SaveFeatureChanges so an application can commit a subset of features that are modified instead of all.
- Adding AcMapLayer.GetLockedFeatures to be consistent with what is in FeatureService and with the ability of applications to use lock and unlock commands with the UpdateFeatures method.
- Removing direct instantiation of AcMapLayer and introducing a static AcMapLayer.Create method, which lets applications return the correct type of layer (vector or grid) depending on the type of layer definition the new layer is based on. And besides, with AcMapLayer.Create, you can write cleaner, more easily maintained code.

What's New in Previous Releases

Features that were new or changed in previous releases.

2008 2007
2006
2005
2004

AutoCAD Map 3D SDK 2008: What's New

The following features are new or changed.

New Geospatial Platform API Preview

This is a preview release, which means that the Geospatial Platform API is by no means set in stone. We invite your comments and suggestions, especially where you can identify missing pieces.

Please let us know what else you need!

The Geospatial Platform is exposed as .NET only. It covers the following areas of functionality:

- Resource Service
- Feature Service
- Mapping Service

New Documentation

AutoCAD Map 3D SDK Help, sdk.doc.main.chm, includes four new documentation components.

All of the new documentation components are available on the Contents tab of AutoCAD Map 3D SDK Help when you open it, so long as they all reside in the same folder. And all of them are included in the Beta release except as noted. Developer's guide components are duplicated in PDF format.

Title	File Name
AutoCAD Map 3D Geospatial Platform .NET Developer's Guide (Preview) **Note** Included in the Beta 2 release, but it is incomplete.	sdk.dev.net.gis.chm
AutoCAD Map 3D Geospatial Platform .NET Reference (Preview)	sdk.ref.gis.platform.chm
AutoCAD Map 3D .NET Reference Supplement (Preview)	sdk.ref.net.supp.chm
AutoCAD Map 3D .NET Developer's Guide **Note** Included in the Beta 2 release, but it is incomplete.	sdk.dev.net.chm

AutoCAD Map 3D SDK Help and its component Help files are located in the Docs folder of the SDK installation.

What's New in 2007

Note The product name for SDK releases 2007 and earlier was Autodesk Map 3D ObjectARX.

The following features are new or changed in the 2007 release.

New Areas of Functionality

There are no new areas of functionality in either ObjectARX or Managed ObjectARX.

Deleted Areas of Functionality

FDO Enabler API has been removed from both ObjectARX and Managed ObjectARX.

Unicode Support

Unicode characters are now acceptable in all string arguments. See the AutoCAD documentation for more information.

Microsoft Visual C++ Version 8.0 Required

Autodesk Map 3D ObjectARX requires Microsoft Visual C++ version 8.0.

Object Class and *Feature Class*

Where the ObjectARX documentation uses the terms *feature class*, keep in mind that the UI documentation has replaced this term with *object class*. The reason for the change is to restrict *feature* terminology to the FDO context (which incidentally is not addressed by the ObjectARX API), so that we can make a clear distinction between the following two kinds of map components:

- Geospatial components, which is stored in FDO feature sources, and are referred to as *features*.
- Drawing components (DWG-based data), which is stored in Autodesk Map drawings, and are referred to as *objects*.

We are not changing the terminology in the API because some API components have *feature* in their names, and changing names of API components is too disruptive.

New Components Not Intended for Public Use

Some components in the following section, "Changes to Existing Functionality," are not intended for public use.

Important Note Be aware of these components. Avoid using them in your applications.

Avoid components that are altogether new. But it is safe to use components where all that has changed is their name (or the name of a parameter or an enumerator element), and their functionality is the same as before. With enumerations, it is also safe to use any enumerator elements that have been added to them.

Changes to Existing Functionality

Changes are summarized in the following table.

Header Files and Classes	Changes
acmapmbmapbook.h	Public bitmask enum AcMapMbMapBook::EModificationType

	enumerators:
	kMapBookTileNameChanged = 8
	kMapBookTileAdjacencyModified = 16

acmapmbmapsheetlayoutsettings.h AcMapMbMapSheetLayoutSettings acmapmbtilegeneratorsettings.h AcMapMbTileGeneratorSettings acmapmbtilegeneratorsettingsarea.h AcMapMbTileGeneratorSettingsArea acmapmbtilegeneratorsettingsgrid.h AcMapMbTileGeneratorSettingsGrid acmapmbtilegeneratorsettingsmanual.h AcMapMbTileGeneratorSettingsManual acmapmbtilenamegeneratorsettings.h AcMapMbTileNameGeneratorSettings acmapmbtilenamegeneratorsettingsdata.h AcMapMbTileName-	Each of these AcMapMbXX classes has two n virtual functions: virtual Acad::ErrorStatus xmlInFields(AcMapMbXmlFiler* pFiler); virtual Acad::ErrorStatus xmlOutFields(AcMapMbXmlFiler* pFiler) const;

GeneratorSettingsData acmapmbtilename-generatorsettingsgrid.h AcMapMbTileName-GeneratorSettingsGrid acmapmbtilename-generatorsettingssequence.h AcMapMbTileName-GeneratorSettingsSequence	
acmapmbtilename-generatorsettings.h AcMapMbTileName-GeneratorIndexer	Class AcMapMbTileNameGeneratorIndexe two new virtual functions: virtual Acad::ErrorStatus StreamIn(AcMapMbXmlFiler* pFiler); virtual Acad::ErrorStatus StreamOut(AcMapMbXmlFiler* pFiler) const;
adeads.h Global data-extension (ADE) functions mapads.h Global mapping functions topoads.h Global topology functions	All functions defined in the xxads.h header files declared as extern "C" for C++ compilation. No these functions are deprecated in favor of newer based APIs.
dmdefaultelement.h	New file.

dmdisplayelement.h	Class AcMapDMStyleReferenceIterator has renamed AcMapDMAllStyleReferencesIter Class AcMapDMElement has two new functio DisplayName and RemoveEntityFromAcqu Function GetStyleReferenceIterator has been by GetAllStyleReferencesIterator. Functions AddStyle, RemoveStyle, and Mov which used the old iterator name, have been chan accordingly. The following functions have been added: virtual Acad::ErrorStatus Render(Renderer &renderer;); virtual void OnMapCSChanged(const ACHAR *oldCS, const ACHAR *newCS); Acad::ErrorStatus ExcludeEntitiesFromStylization(const AcDbObjectIdArray& ids); Acad::ErrorStatus RemoveEntitiesFromStylizationExclusion(const AcDbObjectIdArray& ids);

dmdisplayitem.h	Class AcMapDMItemIterator has been renam AcMapDMAllItemsIterator. Class AcMapDMItemIterator was subclassed from AcMapDMAllDrawOrderItemsIterator, but replacement, AcMapDMAllItemsIterator, is r Member function Rewind has been added. Member function GetObject has been removed. Enumeration AcMapDMItem::ELegendDeta no longer includes the enumerator kLegendCol
dmdisplaymanagement.h	Class AcMapDMDisplayManagement has th functions: Acad::ErrorStatus GetGWS(IGWS** pGWS, AcDbDatabase* pDb); Acad::ErrorStatus GetVectorElementIdForFeatureLayer(AcDbObjectId& elementId, AcDbDatabase* pDb, const TCHAR* layerSourceName); Acad::ErrorStatus GetEntityIdForFeatureOnLayer(AcDbObjectId& entId, AcDbDatabase* pDb, const TCHAR* layerSourceName, int featureId);

dmdisplaystyle.h

Class AcMapDMStyle has a new enumeration

```
enum CoExistenceFlags
{
    kNotApplicable = 0,
    kAllStyles,
    kSingleton,
    kLikeKind
};
```

And four new functions:

```
virtual const ACHAR* DisplayName() const;


virtual Acad::ErrorStatus Refresh(
    void*& pCookie,
    AcDbObjectId entityId,
    Adesk::UInt32 flag = 0);


virtual Acad::ErrorStatus Refresh(
    void*& pCookie,
    const ACHAR* pTopoName,
    long lTopoElemId,
    Adesk::UInt32 flag = 0);


virtual Acad::ErrorStatus deepClone(
    AcDbObject* pOwnerObject,
    AcDbObject*& pClonedObject,
    AcDbIdMapping& idMap,
```

<table>
<tr><td></td><td>Adesk::Boolean isPrimary = true) const;</td></tr>
<tr><td>dmgroup.h</td><td>To accommodate changing the name of AcMapDMItemIterator to AcMapDMAllItemsIterator, member functio class AcMapDMGroup have been changed accordingly.</td></tr>
<tr><td>dmmap.h</td><td>Class AcMapDMMap has changed:
Function GetDrawOrderIterator has been rem
AcMapDMDrawOrderIterator has been replaced AcMapDMAllDrawOrderItemsIterator.
The following functions have been renamed:
<table>
<tr><th>Old Name</th><th>New Name</th></tr>
<tr><td>NumScales</td><td>NumScaleThresholds</td></tr>
<tr><td>GetScale</td><td>GetScaleThreshold</td></tr>
<tr><td>SetCurrentScale</td><td>SetCurrentScaleThre</td></tr>
<tr><td>GetCurrentScale</td><td>GetCurrentScaleThre</td></tr>
<tr><td>AddScale</td><td>AddScaleThreshold</td></tr>
<tr><td>CopyScale</td><td>CopyScaleThreshold</td></tr>
<tr><td>RemoveScale</td><td>RemoveScaleThresho</td></tr>
<tr><td>GetMapScaleFor</td><td>GetThresholdFor</td></tr>
<tr><td>ModifyScale</td><td>ModifyScaleThresho</td></tr>
</table></td></tr>
</table>

Function SetCurrentScale has an additional pa
regen.

The following functions are new:

```
static bool
IsInfintyThreshold(
    double dScale);


static double
InfinityThreshold();


Acad::ErrorStatus
SetExaggeration(
    double dValue);


double
GetExaggeration() const;


Acad::ErrorStatus
Set3dGridPercent(
    int iValue);


int
Get3dGridPercent() const;


Acad::ErrorStatus
SetExtent(
    double dMinX,
```

	double dMinY, double dMaxX, double dMaxY); Acad::ErrorStatus RemoveExtent(); Acad::ErrorStatus GetExtent(double& dMinX, double& dMinY, double& dMaxX, double& dMaxY) const; Acad::ErrorStatus GetSun(double& dAzimuth, double& dAltitude) const; Acad::ErrorStatus SetSun(double dAzimuth, double dAltitude);
dmseannotationstyle.h	Class AcMapDMSEAnnotationStyle has a n function: virtual AcMapDMStyle::CoExistenceFlags CoExistenceType() const;

dmthematicstyle.h	Class AcMapDMThematicBuildRangesErr has a new function: virtual AcMapDMStyle::CoExistenceFlags CoExistenceType() const;
dmthematictable.h	TBD.
dmtopoqueryelement.h	Class AcMapDMTopoQueryElement has tw functions: virtual Acad::ErrorStatus AddStyle(AcDbObjectId& styleRefId, AcMapDMStyle* pStyle, AcMapDMAllStyleReferencesIterator& Pos virtual Acad::ErrorStatus AddStyle(AcDbObjectId& styleRefId, const AcDbObjectId& styleId, AcMapDMAllStyleReferencesIterator& Pos
mapreactors.h	Class AcMapProjectReactor has three new fu virtual bool CoordinateSystemIsToBeChanged(AcMapProject *pproject, const ACHAR *oldCS,

```
    const ACHAR *newCS);


virtual void
CoordinateSystemChanged(
    AcMapProject *pproject,
    const ACHAR *oldCS,
    const ACHAR *newCS);


virtual void
CoordinateSystemChangeVetoed(
    AcMapProject *pproject,
    const ACHAR *oldCS,
    const ACHAR *newCS);
```

What's New 2006

Note The product name for SDK releases 2007 and earlier was Autodesk Map 3D ObjectARX.

The following features were new or changed in the 2006 release.

New Areas of Functionality

Although Feature Data Objects (FDO Client) and Map Plotting are not new areas of functionality, their class-based implementations in this release are entirely new.

Note The previous implementation for Feature Data Objects is no longer available. Applications based on it are no longer valid and will not compile. The previous implementation for Map Plotting, although deprecated, remains in place.

Feature	Header and Library Files
FDO Enabler	AcMapFdo*.h Acmapfdoenabler.lib
Map Book	AcMapMB*.h AcMapMapBook.lib
Custom Object Protocol Extensions	AcMapQueryPE.h AcMapGeometryPE.h maparx.lib

Changed Areas of Functionality

Feature	Header and Library Files
Display Manager	AcMapDM*.h Dm*.h AcMapDisplayManagement.lib

Managed Wrapper Classes

The Managed Wrapper Classes present all of ObjectARX as Managed C++, which lets you access ObjectARX functionality using VB.NET, C#, or any other .NET language.

What's New in 2005

Note The product name for SDK releases 2007 and earlier was Autodesk Map 3D ObjectARX.

The following features were new or changed in the 2005 release.

New ObjectARX APIs

The following new areas of functionality have more than doubled the size and scope of ObjectARX for Autodesk Map 3D by adding more than 100 new classes and almost 1700 new functions.

Annotation
Classification
Coordinate Systems
Create Centroids
Data Sources
Display Management
FDO Client
Import-Export
Object Filter
Topology
User Management

In addition, Autodesk Map 3D documentation now includes a developer's guide. Typically, each overview in the developer's guide is a single topic, however long, which makes it easy to print in a single step.

With this release of Autodesk Map 3D, ObjectARX API coverage and Autodesk Map 3D UI coverage are a close match.

Additions to Data Extension Classes

One new class, AcMapHostApplicationServices. Eight existing classes have new functions added. One new enumeration.

FDO API

You need the FDO API if you are developing custom FDO providers or FDO clients like the Oracle Spatial provider and the FDO client that are included with Autodesk Map 3D. You can find out more about the FDO API at the Autodesk Developer Center, where FDO documentation is available for download. Note that the FDO header and .lib files are included with Autodesk Map 3D ObjectARX.

Points and Surfaces API

Points and Surfaces is a new ActiveX API. ActiveX APIs are included with Autodesk Map 3D, but not with ObjectARX. To view Points and Surfaces documentation, do the following. In the Help folder of an Autodesk Map 3D installation, open acmapatm.chm.

Global Function Sets

The "ADS" function sets, which formerly were documented in Autodesk Map 3D ADS/AutoLISP Help, are now documented in the Developer's Guide and Reference for ObjectARX. All of these functions are global ObjectARX functions.

Data Extension Functions
Map Plotting Functions
Topology Functions

Autodesk Map 3D ADS/AutoLISP Help, acmapads.chm, is no longer included in ObjectARX documentation. The AutoLISP part of ADS/AutoLISP Help is now documented in a new Help file, the AutoLISP API Reference, acmaplisp.chm, which is available in the Help folder of Autodesk Map 3D installations.

New Drawing Cleanup Variables

New cleanup variables for

- Adding feature classes to the Include set.
- Excluding entities from the Anchor set.
- Marking apparent intersection errors.
- Deleting duplicates and snapping clustered nodes.
- Weeding polylines.

New Clean Group Type Constants

512	Apparent intersections
1024	Weed polylines

These constants are used with the following functions:

tpm_cleanactionlistgetat
tpm_cleanactionlistins
tpm_cleangrouptype

The complete set of constants is listed with tpm_cleangrouptype.

New Drawing Cleanup Functions

tpm_cleancreatedss	Gets created entities following a cleanup process.
tpm_cleanmodifiedss	Gets changed entities following a cleanup process.
tpm_cleanunchangedss	Gets unchanged entities following a cleanup process.

What's New in 2004

Note The product name for SDK releases 2007 and earlier was Autodesk Map 3D ObjectARX.

The following features were new or changed in the 2004 release.

AcDbMPolygon Moved

The AcDbMPolygon class has been moved from the Autodesk Map 3D ObjectARX API to the Autodesk ObjectARX API and is now documented in the Autodesk ObjectARX Reference.

True Color

Most Autodesk Map 3D color situations, including querying and property alteration, now support true-color formatted strings in addition to color indexes.

For example, you can use true colors in situations associated with the following API components.

- In the AcMap::EAlterationType enumeration, which identifies property alteration types, the kAlterationColor enumerator.
- In the AcMap::EPropertyType enumeration, which identifies the property type on which a property condition is based, the AcMap::kColor enumerator.
- In the AcMap::EErrCode enumeration, which specifies a set of return codes, the AcMap::kErrInvalidColor (1806) enumerator and the AcMap::kErrTopInvalidColor (2019) enumerator.

- The AcMapQueryCondition subclasses, AcMapDataCondition, AcMapLocationCondition, AcMapPropertyCondition, and AcMapSQLCondition, which represent the four types of query condition, with their condition-defining functions for conditions that can involve color.
- The AcMapPropertyAlteration class, which represents simple property alterations such as color changes. Text and hatch alterations, which add text labels and apply hatch patterns to queried objects, are represented by two AcMapPropertyAlteration subclasses: AcMapTextAlteration and AcMapHatchAlteration. A property alteration definition can contain any of the three.
- The Get and SetOptionValue functions of the AcMapProject class, with their pcName arguments, ColorForAdd and ColorForRemove, which are listed under Query Options in the Project Options topic.

Property and Data Queries Involving Feature Classes

To let you specify, when querying feature class objects, whether to include subclasses of the target feature class, an optional Flag parameter has been added to the AcMapDataCondition and AcMapPropertyCondition constructors. To let you view or change the Flag setting of a data or property condition, two functions, Flag() and SetFlag(), have been added to the AcMapDataCondition and AcMapPropertyCondition classes.

Microsoft Visual C++ Version 7.0 Required

Autodesk Map 3D ObjectARX requires Microsoft Visual C++ version 7.0. Microsoft Visual C++ .NET 2003, a component of Visual Studio .NET 2003, code-named Everett, is not supported.