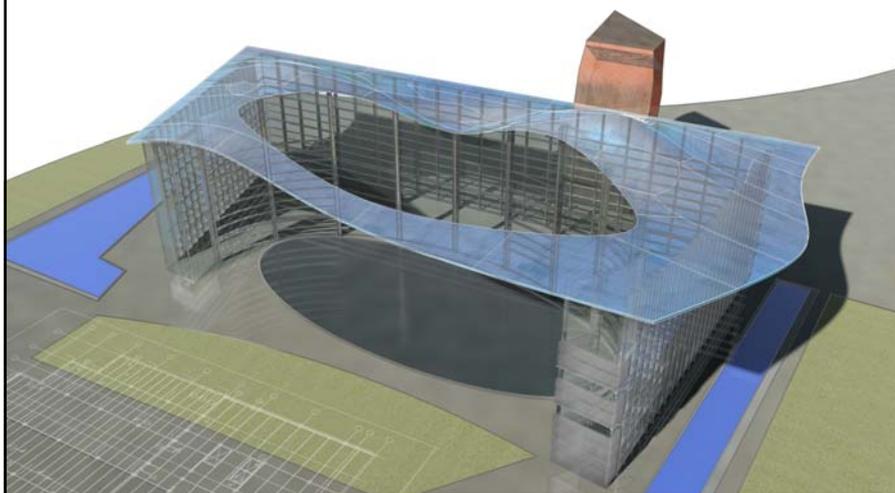


DevTV AutoCAD.NET プログラミングのはじめ



Autodesk

このDevTVの内容

- 簡単なアプリケーションの作成
- 図形を図面に追加
- 図形の編集
- アプリケーションのデータの保存
- イベントの監視

© 2006 Autodesk

2

Autodesk

簡単なアプリケーションの作成

© 2006 Autodesk

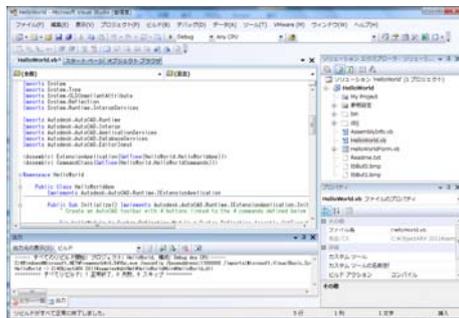
3

Autodesk

簡単なアプリケーションの作成

AutoCAD 2011 の .NET アプリケーションの開発ツール

- Visual Studio 2008 SP 1



- Visual Studio Express Edition も使用可

© 2006 Autodesk

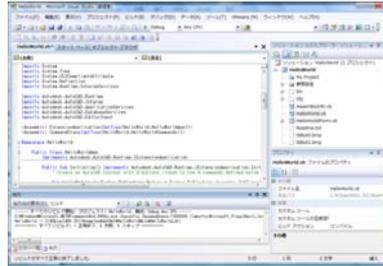
4

Autodesk

簡単なアプリケーションの作成

プログラミング言語

- Visual Basic
- C#
- C++ /CLI



今回はVisual Basic を使用します...

© 2006 Autodesk

5

Autodesk

簡単なアプリケーションの作成

AutoCAD .NET APIを使用するためアセンブリを参照する

AcMgd.dll

AutoCAD の機能を提供

AcDbMgd.dll

図面データベースの編集機能を提供

© 2006 Autodesk

6

Autodesk

簡単なアプリケーションの作成

AutoCAD .NET アセンブリの場所

- ObjectARX SDK
 - C:\ObjectARX 2011 SDK\inc **推奨**
 - <http://www.objectarx.com> から無償ダウンロード
- AutoCAD 製品のインストールフォルダ
 - C:\Program Files\Autodesk\AutoCAD 2011

© 2006 Autodesk

7

Autodesk

簡単なアプリケーションの作成

Importsステートメントで名前空間を参照する

```
Imports Autodesk.AutoCAD.ApplicationServices  
Imports Autodesk.AutoCAD.EditorInput  
Imports Autodesk.AutoCAD.Runtime
```

© 2006 Autodesk

8

Autodesk

簡単なアプリケーションの作成

AutoCAD カスタムコマンドの定義

- 関数に属性を付けてコマンドを定義

```
Public Class Class1
  <CommandMethod("HelloWorld")> _
  Public Sub HelloWorld()
    ' コマンドの振る舞いを実装する
  End Sub
End Class
```

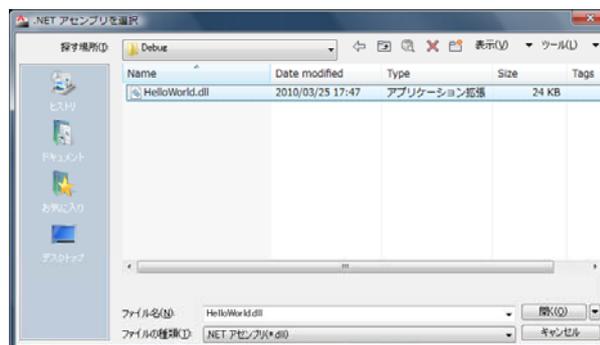
© 2006 Autodesk

9

Autodesk

簡単なアプリケーションの作成

NETLOAD コマンドでアプリケーションをロード



© 2006 Autodesk

10

Autodesk

Hello World



© 2006 Autodesk

11

Autodesk

図形を図面に追加

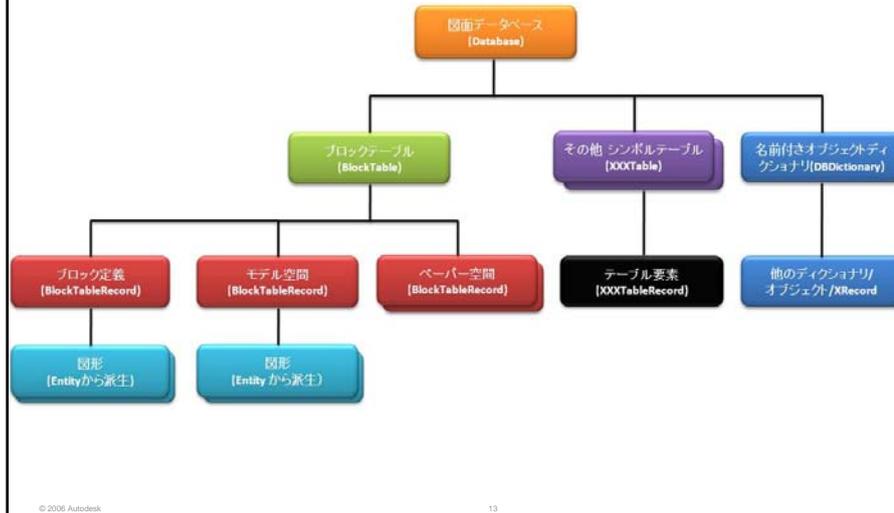
© 2006 Autodesk

12

Autodesk

図形を図面に追加

図面データベースの構造



図形を図面に追加

AutoCAD .NET オブジェクトモデルを理解する

- AutoCAD 2010 .NET デベロッパガイド (日本語版)

- www.autodesk.co.jp/developautocad

- AutoCAD .NET Developer's Guide (英語版)

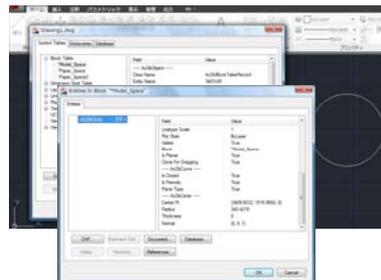
- AutoCAD 2011 ヘルプ > 開発者ドキュメント



図形を図面に追加

AutoCAD .NET オブジェクトモデルを理解する

- **ArxDbg (C++)** ObjectARX SDK サンプル
 <ObjectARX 2011 SDK>\samples\database\ARXDBG



- **MgdDbg(C#)** ADNメンバに提供

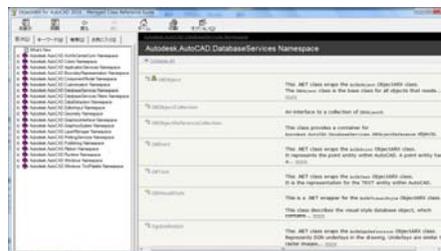
© 2006 Autodesk

15

Autodesk

図形を図面に追加

- AutoCAD .NET クラスリファレンス (英語版)
 - <ObjectARX 2011 SDK>\docs\larxmgd.chm



- AutoCAD .NET サンプル
 - <ObjectARX 2011 SDK>\samples\dotNet

© 2006 Autodesk

16

Autodesk

図形を図面に追加

オブジェクトID

- 図面データベース内の全てのオブジェクトはオブジェクトIDという識別子を持つ
- オブジェクトIDで図面データベース内のオブジェクトを特定

© 2006 Autodesk

17

Autodesk

図形を図面に追加

図面データベースアクセスにトランザクション処理を使用

- 図面データベース編集の単位
 - コミット – 編集内容を図面データベースに反映
 - 破棄 – 編集内容を無効化

© 2006 Autodesk

18

Autodesk

図形を図面に追加

トランザクション内でオブジェクトをオープン

- Autodesk.AutoCAD.DatabaseServices.Transaction.GetObject 関数
- 第一引数はオブジェクトの識別子
- 第二引数はオープンモード
 - ForRead – 読み込み
 - ForWrite – 書き込み

© 2006 Autodesk

19

Autodesk

図形を図面に追加

基本的な図形追加の手順

1. 図形のインスタンスを作成
2. トランザクションを開始
3. 図形の格納先をトランザクションでオープン
4. 図形を格納先に追加
5. トランザクションにオブジェクトの新規追加を通知
6. トランザクションをコミット

© 2006 Autodesk

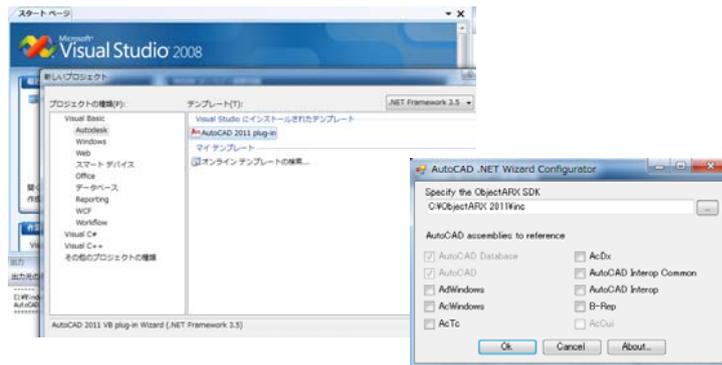
20

Autodesk

AutoCAD .NET Wizard

AutoCAD 2011 .NET Wizard を使用してプロジェクトを作成

- AutoCAD 2011 .NET Wizard
 - www.autodesk.co.jp/developautocad



© 2006 Autodesk

21

Autodesk

図面に円を追加



© 2006 Autodesk

22

Autodesk

図形の編集

© 2006 Autodesk

23

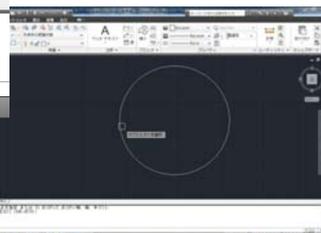
Autodesk

図形の編集

ユーザー入力の取得

- プロンプトは PromptXXXOptions クラスで作成
 - 例： PromptEntityOptions PromptStringOptions
- プロンプト出力は Editor オブジェクトの GetXXX メンバ関数を使用
 - 例： GetEntity GetString GetAngle
- 取得結果は PromptResult または派生クラスに格納
 - 例： PromptEntityResult PromptDoubleResult

```
モデル レイアウト1 / レイアウト2  
次の点を指定 または [元に戻す (U)]:  
次の点を指定 または [元に戻す (U)]:  
コマンド:  
コマンド:  
コマンド: _MOVE  
オブジェクトを選択:  
983.8954, 1302.8156, 0.0000
```



© 2006 Autodesk

24

Autodesk

図形の編集

基本的な手順

1. 編集する図形を特定（例: 選択処理）
2. トランザクションを開始
3. 特定した図形をトランザクションでオープン
4. 図形を編集
5. トランザクションをコミット

© 2006 Autodesk

25

Autodesk

図形の編集



© 2006 Autodesk

26

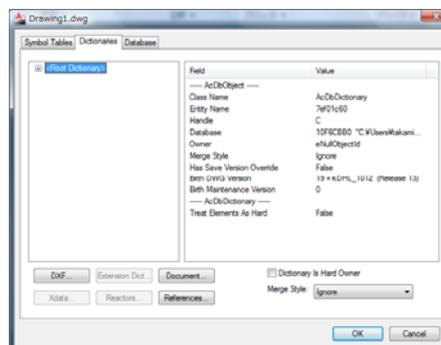
Autodesk

アプリケーションのデータの保存

アプリケーションのデータの保存

データの格納場所 1

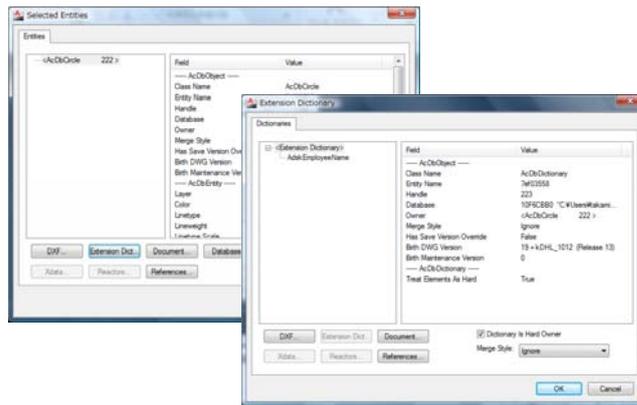
- 名前付オブジェクトディクショナリ (Named Object Dictionary)
 - 図面データベースに必ず一つ存在するルートディクショナリ



アプリケーションのデータの保存

データの格納場所 2

- 拡張ディクショナリ (Extension Dictionary)
 - 使用時にオブジェクトに作成

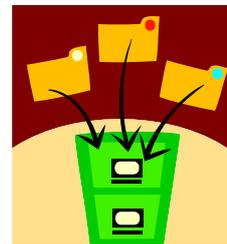


アプリケーションのデータの保存

XRecord オブジェクトをディクショナリに追加

- XRecordはデータを保存するためのオブジェクト
- データの型と値のペアのリンクリストを保管

アプリケーションのデータの保存



© 2006 Autodesk

31

Autodesk

イベントの監視

© 2006 Autodesk

32

Autodesk

イベントの監視

- イベント
 - 何か起きた事を伝えるためにオブジェクトから送られるメッセージ
 - デリゲートを用いて送信元と受信元をつないでイベントを監視
- デリゲート
 - クラスに類似
 - 引数の型が一致する関数の参照を保持

© 2006 Autodesk

33

Autodesk

イベントの監視

- 受信元の関数(イベントハンドラ)を定義

```
Sub objAppended(ByVal o As Object, ByVal e As ObjectEventArgs)
    MessageBox.Show("ObjectAppended!")
    '目的の処理を行う
End Sub
```

- イベントに受信元の関数を登録

```
Dim db As Database
db = HostApplicationServices.WorkingDatabase()
AddHandler db.ObjectAppended, New ObjectEventHandler(AddressOf objAppended)
```

- 受信元の関数の登録解除

```
RemoveHandler db.ObjectAppended, AddressOf objAppended
```

© 2006 Autodesk

34

Autodesk

イベントの監視



© 2006 Autodesk

35

Autodesk

補足

- Through the Interface (AutoCAD API ブログ)
 - <http://through-the-interface.typepad.com>
- AutoCAD デベロッパセンター
 - www.autodesk.co.jp/developautocad
- AutoCAD ディスカッショングループ
 - <http://discussion.autodesk.com/forums/category.jspa?categoryID=8>
- オートデスク デベロッパ ネットワーク
 - <http://www.autodesk.co.jp> (「パートナー」→「ADN」)

© 2006 Autodesk

36

Autodesk

ありがとうございました

Autodesk

© 2006 Autodesk

37