

# RootPro CAD

## Enterprise

～ アドイン作成 チュートリアル～

Visual Studio Tools for Applications を利用したアドインの作成方法

2008.04.01 Ver 1.00

### アドイン概要

「円の作図」

中心座標を (100, 100)、半径 100 の円を作図する

### アドイン作成環境

バージョン: RootPro CAD Enterprise Version 3.40

使用言語: Visual Basic



株式会社 ルートプロ

<http://www.rootprocad.com/>

Copyright © 2008 RootPro Co.,Ltd. All Rights Reserved

## アドイン作成手順

### 1. Visual Studio 2005 Tools for Applications ( VSTA ) の起動

スタートメニューから、RootPro CAD 3 Enterprise を選択して、その中にある、Visual Studio 2005 Tools for Applications を選択して起動します。



以降「Visual Studio 2005 Tools for Applications」は「VSTA」と略します。

### 2. プロジェクトの作成

VSTA が起動したら、メニューから [ファイル] - [新しいプロジェクト] を選択します。

新しいプロジェクトのダイアログボックスが開きますので、プロジェクトの種類から、[Visual Basic] を選択して、プロジェクト名を、今回は仮に [RootProCADAddInCircle] とします。

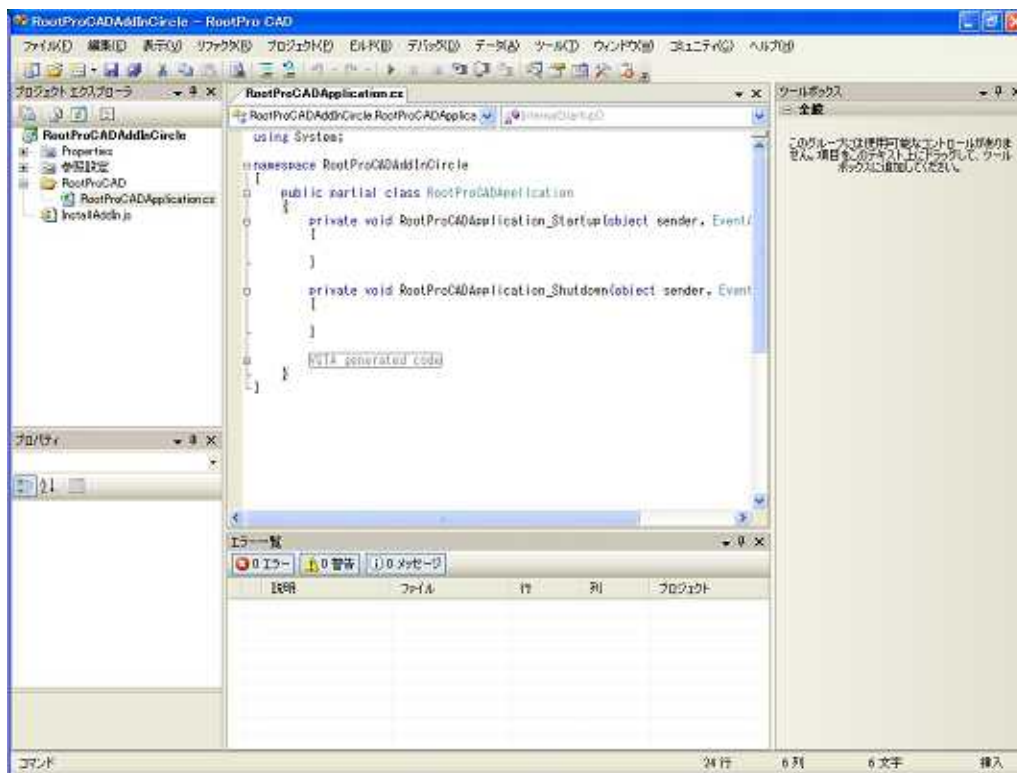
OK ボタンを押すとプロジェクトが作成されます。



### 3. コードの記述方法

プロジェクトが作成されると、いくつかのファイルが自動的に追加されます。

[プロジェクト エクスプローラ] から [RootProCADApplications.vb] をダブルクリックして開きます。  
上記のファイルにコードを追加していくこととなります。



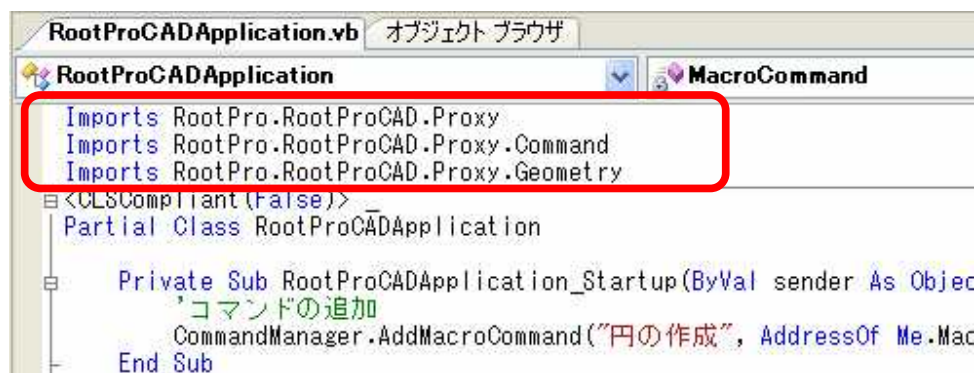
## 4. 名前空間の定義

まず、RootPro CAD API を使用するための名前空間を定義します。名前空間は定義しなくても、コードを書くことは可能ですが、名前空間を定義することによりオブジェクト名を指定する必要がなくなります。

ソースコード上部に

```
Imports RootPro.RootProCAD.Proxy
Imports RootPro.RootProCAD.Proxy.Command
Imports RootPro.RootProCAD.Proxy.Geometry
```

と記述します。



## 5. アドインの表示名と、関数名の定義

次に、アドインメニューに表示されるメニュー名と、実行時に呼ばれる関数名を指定します。

起動時にコマンドを追加する記述を

`Private Sub` RootProCADApplication\_Startup 関数の中に以下の行を追加してください。

```
CommandManager.AddMacroCommand("円の作成", AddressOf Me.MacroCommand)
```

また終了時にコマンドを削除する記述を、

`Private Sub` RootProCADApplication\_Shutdown 関数の中に以下の行を追加してください。

```
CommandManager.RemoveMacroCommand(AddressOf Me.MacroCommand)
```

MacroCommand が、アドインを実行する時の関数名

"円の作成" がアドインメニューに表示されるメニュー名です。

```
Imports Microsoft.Office.Interop.AutoCAD
<CLSCompliant(False)>
Partial Class RootProCADApplication
    Private Sub RootProCADApplication_Startup(ByVal sender As Object, ByVal e As System.EventArgs) |
        コマンドの追加
        CommandManager.AddMacroCommand("円の作成", AddressOf Me.MacroCommand)
    End Sub

    Private Sub RootProCADApplication_Shutdown(ByVal sender As Object, ByVal e As System.EventArgs) |
        コマンドの削除
        CommandManager.RemoveMacroCommand(AddressOf Me.MacroCommand)
    End Sub
End Class
```

## 6. 実行処理の記述

実際にアドインの実行処理を MacroCommand に記述します。

MacroCommand 関数を、ソースコードに記述します。

```
'円作成のマクロコマンド
Private Sub MacroCommand()
    'アクティブなファイルを取得
    Dim doc As Document = ActiveDocument

    'カレントの部分図を取得
    Dim drawing As Drawing = doc.CurrentDrawing

    ' UNDO 可能な操作を開始
    doc.UndoManager.BeginUndoUnit()

    '中心点
    Dim pointCenter As Point2d = Geometry.CreatePoint(100, 100)

    '半径
    Dim radius As Double = 100.0

    '円を部分図に追加
    drawing.Shapes.AddCircle(pointCenter, radius)

    ' UNDO 可能な操作を終了
    doc.UndoManager.EndUndoUnit()

End Sub
```

```
RootProCADApplication.vb オブジェクト ブラウザ
(全般) (宣言)
Imports RootPro.RootProCAD.Proxy
Imports RootPro.RootProCAD.Proxy.Command
Imports RootPro.RootProCAD.Proxy.Geometry
<CLSCompliant(False)>
Partial Class RootProCADApplication
    Private Sub RootProCADApplication_Startup(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Startup
        'コマンドの追加
        CommandManager.AddMacroCommand("円の作成", AddressOf Me.MacroCommand)
    End Sub

    Private Sub RootProCADApplication_Shutdown(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Shutdown
        'コマンドの削除
        CommandManager.RemoveMacroCommand(AddressOf Me.MacroCommand)
    End Sub

    '円作成のマクロコマンド
    Private Sub MacroCommand()
        'アクティブなファイルを取得
        Dim doc As Document = ActiveDocument

        'カレントの部分図を取得
        Dim drawing As Drawing = doc.CurrentDrawing

        'UNDO 可能な操作を開始
        doc.UndoManager.BeginUndoUnit()

        '中心点
        Dim pointCenter As Point2d = Geometry.CreatePoint(100, 100)

        '半径
        Dim radius As Double = 100.0

        '円を部分図に追加
        drawing.Shapes.AddCircle(pointCenter, radius)

        'UNDO 可能な操作を終了
        doc.UndoManager.EndUndoUnit()
    End Sub
End Class
```

まず、アクティブなファイルのカレント部分図を取得します。

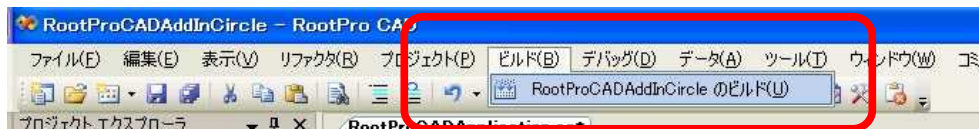
円の中心点と、半径を定義して、その後に、AddCircle メソッドで部分図に円を追加します。

上記コードでは BeginUndoUnit、EndUndoUnit で Undo が使用できるように設定しています。

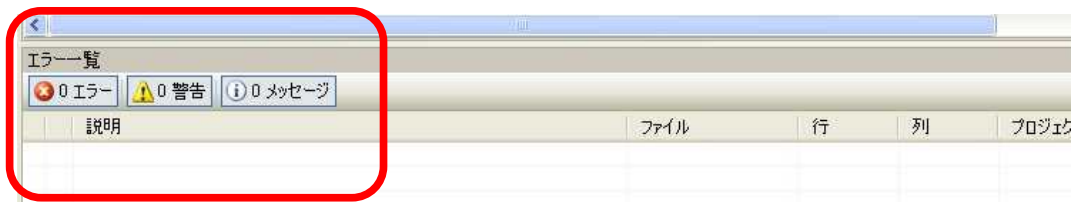
上記でソースコードの記述は終わりです。

## 7. プロジェクトのビルド

プロジェクトのビルド（コンパイル）を行います。

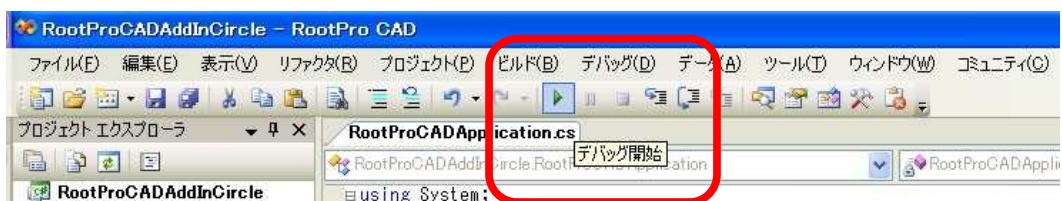


ここで、エラーが表示される場合は、コードの記述に誤りがあります。  
もう一度コードを見直して、ビルドを実行してください。



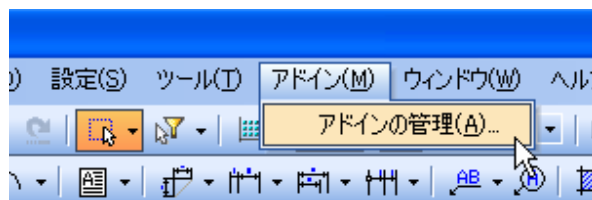
## 8. 実行（デバッグ）の開始

ビルドが成功したら、デバッグ開始を選択してプログラムが実行できるか検査します。

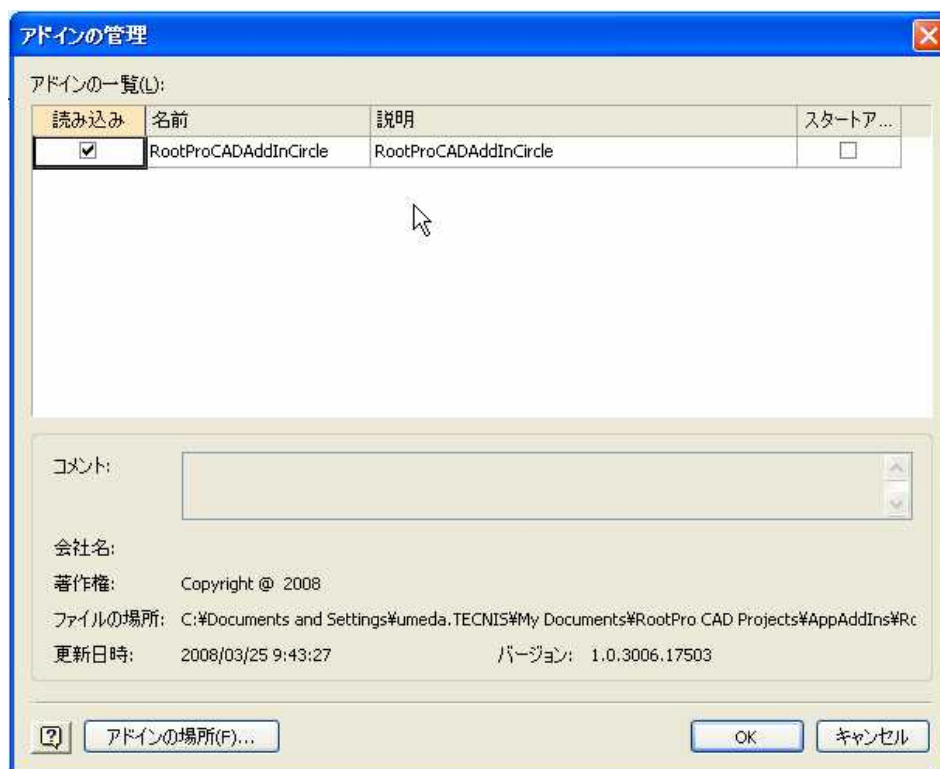


[デバッグ開始] を選択すると、自動的に「RootPro CAD」が起動します。

[RootPro CAD] のメニューから、[アドインの管理] を選択します。

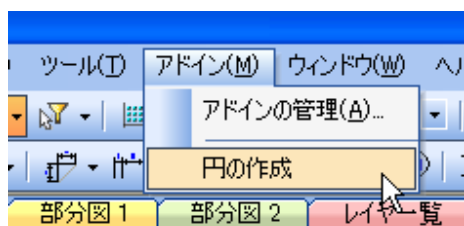


アドインの管理画面に、作成した RootProCADAddInCircle が表示されるので読み込みの欄にチェックを入れて OK を押します。

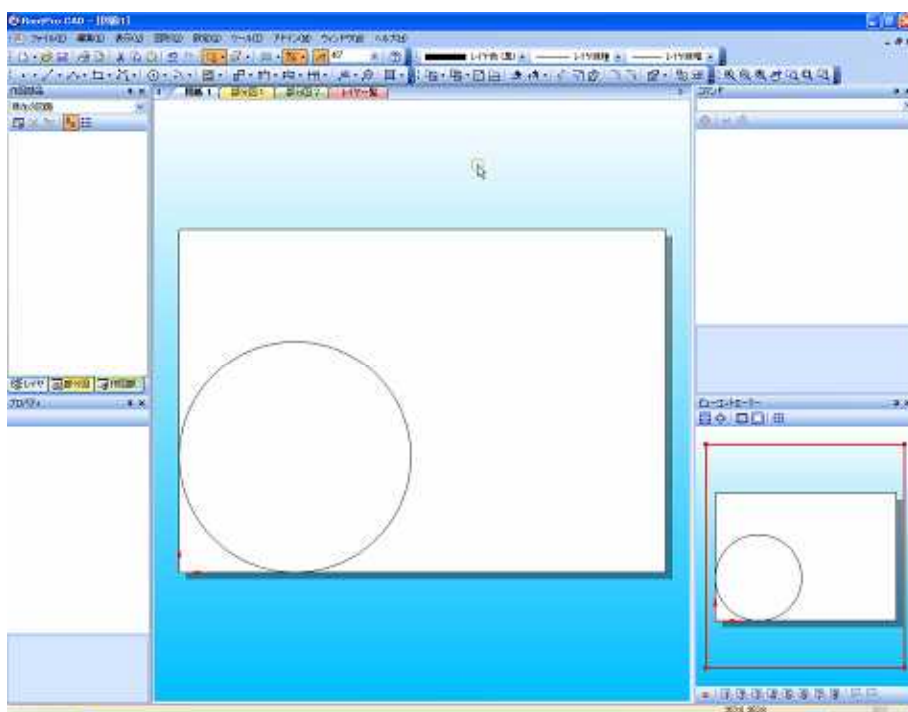


これで、アドインのメニューに「円の作成」が追加されますので、選択します。選択すると、コ

ードの記述内容がすぐに実行されます。



図のように、円が作成できれば成功です。

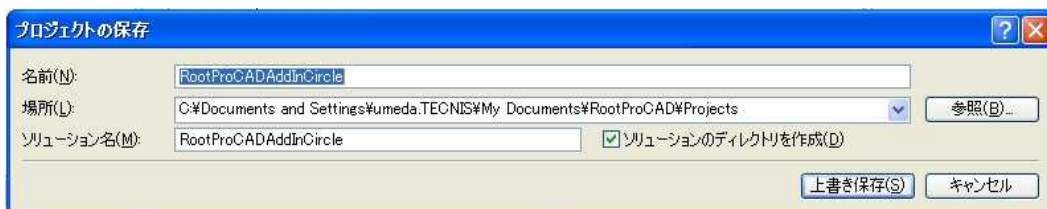
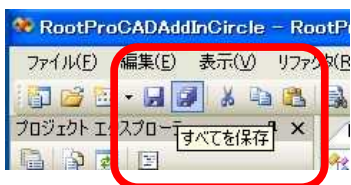


作成したアドインを次回から利用するためには、RootPro CAD Enterprise を起動して、同様にアドイン管理メニューからの読み込みにチェックを入れてください。スタートアップにチェックを入れておくと、RootPro CAD 起動時に自動的に読み込まれます。

## 9. プロジェクトの保存

最後に VSTA で作成したプロジェクトを保存します。

すべて保存を選択すると、プロジェクトの保存ダイアログボックスが出てきますので場所を指定して [上書き保存] を実行します。



\*\* 以上でチュートリアルは終了です \*\*