Vidyaputra
(DHIRAJ S. DIVIEDI)

CAA/RADE Basics

STAY TUNED..STAY UPDATED AND SMART……

(Subscribe us for update and notifications by clicking on follow tab which is located at your right bottom corner of website)

Abstract

This teleconference will go through our thought process in deciding to use CAA/RADE and will discuss the basics of creating a CAA/RADE program. Programming with CAA/RADE is not easy, and can be very intimidating. This session will go through a methodology that can be used to break this programming problem into smaller parts, making it more feasible. Half of the conference will be spent laying down good programming principles, and the other half will be a hands-on experience creating your own workspace.

Overview

- Strengths and Weaknesses of Various Methods of Programming CATIA
- My Language Choices
- Advantages of Object Oriented Programming
- Different Object Oriented Options
  - VB 6.0
  - CAA/RADE
  - Combination
- Understanding CAA/RADE

CATIA Programming Options

- Visual Basic
  - VBScript and CATScript - the Quick and Dirty Method
  - VBA – A Script Created in a MSDev Environment
  - Visual Basic 6.0 - Simplest Object Oriented Approach
  - Visual Studio.Net - Up and Coming Approach
- CAA/RADE
  - C++ - the Primary Language
  - Java - Supported but not as Common

My Language Choices

- Methods I am Not Using
  - Script Language Has No Debugging
  - VBA Environment is Not Object Oriented
  - VB.Net Should Be There Soon, But Not Yet
  - Java is Not Well Supported and May Not Ever Be
- Languages I am Using
  - VB 6.0 is a Relatively Easy yet Powerful Tool
  - C++ is the Most Supported CAA/RADE Language

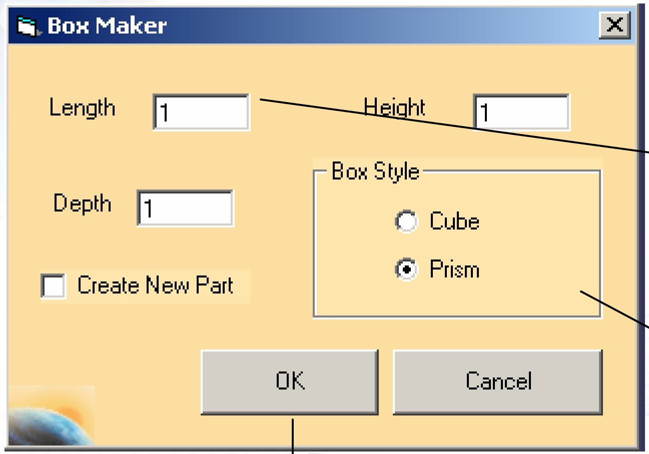
Object Oriented Power

- Ease of Programming; Re-use
  - Initial Investment of Programming Ability
  - Only Write Functions Once
  - Organize Functions into your Own Structure
  - Share Objects Throughout Company
- Facilitate Maintenance
  - Problems Only Need to be Fixed in One Place
  - Feasible to Document Functions Well

Object Oriented Options – Overview

	VB 6.0	CAA/RADE	Combination
Programming Difficulty	Easy	Hard	Medium
GUI Capability	High	Medium	High
API Exposure Quantity	Low	High	High
API Exposure Time	Late	Early	Early
License Costs	Low	High	Medium
Maintenance Costs	Low	High	Medium

Object Oriented Options – VB 6.0 GUI



```
Private Sub txtLength_Validate(Cancel As Boolean)

    If Not IsNumeric(txtLength) Then

        MsgBox "Number Required!"
        txtLength = 1

    End If

    If OptCube = True Then

        txtDepth = txtLength
        txtHeight = txtLength

    End If


End Sub
```

```
Private Sub OptCube_Click()

    txtDepth.Enabled = False
    txtHeight.Enabled = False
    txtDepth = txtLength
    txtHeight = txtLength

End Sub
```

```
Private Sub OptPrism_Click()

    txtDepth.Enabled = True
    txtHeight.Enabled = True

End Sub
```

See Next Slide

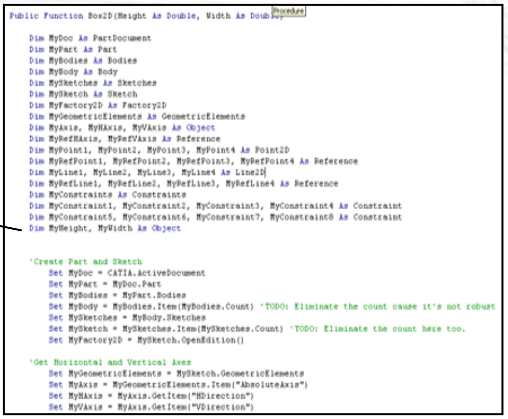
The VB GUI Code is Very Simple

VB Toolbars Can Have Unpredictable Behavior and are Not Directly Integrated

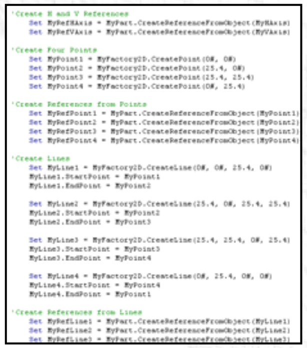
Object Oriented Options – VB 6.0 Structure

```
Private Sub cmdOK_Click()

    Dim MyCatia As Eric.CATIA
    Dim MySketchPlane As Object


    GetUserVals
    Set MyCatia = New Eric.CATIA

    'Start CATIA
    MyCatia.StartCATIA True

    If chkCreatePart.Value = 1 Then

        MyCatia.NewPartDoc

    End If

    'Retrieve XY Plane and create a sketch
    Set MySketchPlane = MyCatia.GetPlaneXY
    MyCatia.CreateSketch MySketchPlane

    'Draw a Box in the Sketch
    MyCatia.Box2D mDepth, mLength

    'Pad the Sketch
    MyCatia.CreatePad mHeight

    Unload Me

End Sub
```

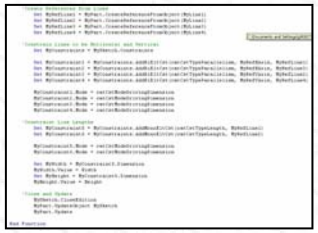
```
Public Function StartCatia(Optional visible As Boolean = True) As INFITF.Application

    On Error Resume Next

    Set StartCatia = GetObject(, "CATIA.Application")
    If Err.Number <> 0 Then
      Set StartCatia = CreateObject("CATIA.Application")
    End If
    StartCatia.visible = visible
    StartCatia.DisplayFileAlerts = visible
    On Error GoTo 0

End Function
```



```
Public Function Box2D(Height As Double, Width As Double)

    Dim MyDoc As PartDocument
    Dim MyPart As Part
    Dim MyBodies As Bodies
    Dim MyBody As Body
    Dim MySketches As Sketches
    Dim MySketch As Sketch
    Dim MyFactory2D As Factory2D
    Dim MyGeometricElements As GeometricElements
    Dim MyAxis, MyHAxis, MyVAxis As Object
    Dim MyRefHAxis, MyRefVAxis As Reference
    Dim MyPoint1, MyPoint2, MyPoint3, MyPoint4 As Point2D
    Dim MyRefPoint1, MyRefPoint2, MyRefPoint3, MyRefPoint4 As Reference
    Dim MyLine1, MyLine2, MyLine3, MyLine4 As Line2D
    Dim MyRefLine1, MyRefLine2, MyRefLine3, MyRefLine4 As Reference
    Dim MyConstraints As Constraints
    Dim MyConstraint1, MyConstraint2, MyConstraint3, MyConstraint4 As Constraint
    Dim MyConstraint5, MyConstraint6, MyConstraint7, MyConstraint8 As Constraint
    Dim MyHeight, MyWidth As Object


    'Create Part and Sketch
        Set MyDoc = CATIA.ActiveDocument
        Set MyPart = MyDoc.Part
        Set MyBodies = MyPart.Bodies
        Set MyBody = MyBodies.Item(MyBodies.Count) 'TODO: Eliminate the count cause it's not robust
        Set MySketches = MyBody.Sketches
        Set MySketch = MySketches.Item(MySketches.Count) 'TODO: Eliminate the count here too.
        Set MyFactory2D = MySketch.OpenEdition()

    'Get Horizontal and Vertical Axes
        Set MyGeometricElements = MySketch.GeometricElements
        Set MyAxis = MyGeometricElements.Item("AbsoluteAxis")
        Set MyHAxis = MyAxis.GetItem("HDirection")
        Set MyVAxis = MyAxis.GetItem("VDirection")
```



```
'Create H and V References
    Set MyRefHAxis = MyPart.CreateReferenceFromObject(MyHAxis)
    Set MyRefVAxis = MyPart.CreateReferenceFromObject(MyVAxis)

'Create Four Points
    Set MyPoint1 = MyFactory2D.CreatePoint(0#, 0#)
    Set MyPoint2 = MyFactory2D.CreatePoint(25.4, 0#)
    Set MyPoint3 = MyFactory2D.CreatePoint(25.4, 25.4)
    Set MyPoint4 = MyFactory2D.CreatePoint(0#, 25.4)

'Create References from Points
    Set MyRefPoint1 = MyPart.CreateReferenceFromObject(MyPoint1)
    Set MyRefPoint2 = MyPart.CreateReferenceFromObject(MyPoint2)
    Set MyRefPoint3 = MyPart.CreateReferenceFromObject(MyPoint3)
    Set MyRefPoint4 = MyPart.CreateReferenceFromObject(MyPoint4)

'Create Lines
    Set MyLine1 = MyFactory2D.CreateLine(0#, 0#, 25.4, 0#)
    MyLine1.StartPoint = MyPoint1
    MyLine1.EndPoint = MyPoint2

    Set MyLine2 = MyFactory2D.CreateLine(25.4, 0#, 25.4, 25.4)
    MyLine2.StartPoint = MyPoint2
    MyLine2.EndPoint = MyPoint3

    Set MyLine3 = MyFactory2D.CreateLine(25.4, 25.4, 0#, 25.4)
    MyLine3.StartPoint = MyPoint3
    MyLine3.EndPoint = MyPoint4

    Set MyLine4 = MyFactory2D.CreateLine(0#, 25.4, 0#, 0#)
    MyLine4.StartPoint = MyPoint4
    MyLine4.EndPoint = MyPoint1

'Create References from Lines
    Set MyRefLine1 = MyPart.CreateReferenceFromObject(MyLine1)
    Set MyRefLine2 = MyPart.CreateReferenceFromObject(MyLine2)
    Set MyRefLine3 = MyPart.CreateReferenceFromObject(MyLine3)
```



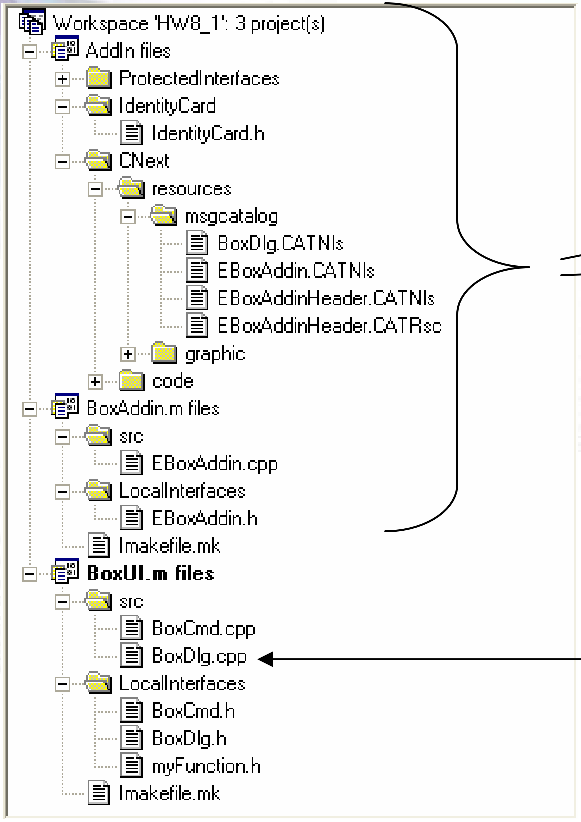
```
'Create References from Lines
    Set MyRefLine1 = MyPart.CreateReferenceFromObject(MyLine1)
    Set MyRefLine2 = MyPart.CreateReferenceFromObject(MyLine2)
    Set MyRefLine3 = MyPart.CreateReferenceFromObject(MyLine3)
    Set MyRefLine4 = MyPart.CreateReferenceFromObject(MyLine4)

'Constrain Lines to be Horizontal and Vertical
    Set MyConstraints = MySketch.Constraints

    Set MyConstraint1 = MyConstraints.AddBiEltCst(catCstTypeParallelism, MyRefHAxis, [illegible]
    Set MyConstraint2 = MyConstraints.AddBiEltCst(catCstTypeParallelism, MyRefHAxis, [illegible]
    Set MyConstraint3 = MyConstraints.AddBiEltCst(catCstTypeParallelism, MyRefVAxis, [illegible]
    Set MyConstraint4 = MyConstraints.AddBiEltCst(catCstTypeParallelism, MyRefVAxis, [illegible]

    MyConstraint1.Mode = [illegible]
    MyConstraint2.Mode = [illegible]
    MyConstraint3.Mode = [illegible]
    MyConstraint4.Mode = [illegible]

'Constrain Line Lengths
    Set MyConstraint5 = MyConstraints.AddMonoEltCst(catCstTypeLength, [illegible]
    Set MyConstraint6 = MyConstraints.AddMonoEltCst(catCstTypeLength, [illegible]

    MyConstraint5.Mode = [illegible]
    MyConstraint6.Mode = [illegible]

    Set MyWidth = MyConstraint5.Dimension
    MyWidth.Value = Width
    Set MyHeight = MyConstraint6.Dimension
    MyHeight.Value = Height

'Close and Update
    MySketch.CloseEdition
    MyPart.UpdateObject MySketch
    MyPart.Update

End Function
```

Three Pages of Code
implemented in One Line

GOODRICH

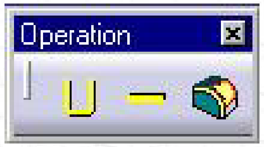
Founding Partners
IBM | DASSAULT SYSTEMES

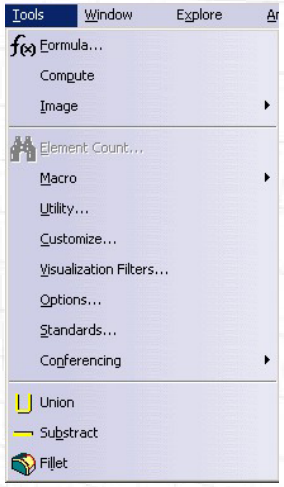
Object Oriented Options CAA/RADE GUI



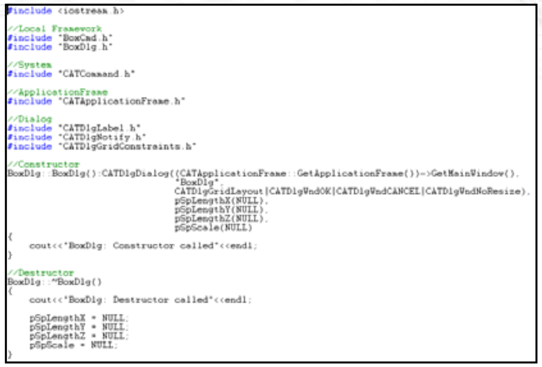
```
Workspace 'HW8_1': 3 project(s)
  AddIn files
    ProtectedInterfaces
    IdentityCard
      IdentityCard.h
    CNext
      resources
        msgcatalog
          BoxDlg.CATNls
          EBoxAddin.CATNls
          EBoxAddinHeader.CATNls
          EBoxAddinHeader.CATRsc
        graphic
      code
  BoxAddin.m files
    src
      EBoxAddin.cpp
    LocalInterfaces
      EBoxAddin.h
    Imakefile.mk
  BoxUI.m files
    src
      BoxCmd.cpp
      BoxDlg.cpp
    LocalInterfaces
      BoxCmd.h
      BoxDlg.h
      myFunction.h
    Imakefile.mk
```

Complex File Structure

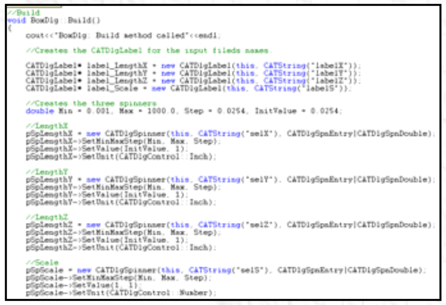




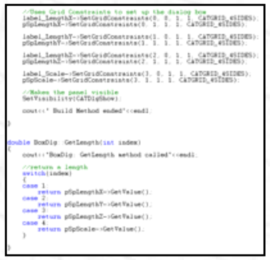
The Commands Are Directly Integrated Into CATIA



```
#include <iostream.h>

//Local Framework
#include "BoxCmd.h"
#include "BoxDlg.h"

//System
#include "CATCommand.h"

//ApplicationFrame
#include "CATApplicationFrame.h"

//Dialog
#include "CATDlgLabel.h"
#include "CATDlgNotify.h"
#include "CATDlgGridConstraints.h"

//Constructor
BoxDlg::BoxDlg():CATDlgDialog((CATApplicationFrame::GetApplicationFrame())->GetMainWindow(),
                              "BoxDlg",
                              CATDlgGridLayout|CATDlgWndOK|CATDlgWndCANCEL|CATDlgWndNoResize),
                              pSpLengthX(NULL),
                              pSpLengthY(NULL),
                              pSpLengthZ(NULL),
                              pSpScale(NULL)
{
    cout<<"BoxDlg: Constructor called"<<endl;
}

//Destructor
BoxDlg::~BoxDlg()
{
    cout<<"BoxDlg: Destructor called"<<endl;

    pSpLengthX = NULL;
    pSpLengthY = NULL;
    pSpLengthZ = NULL;
    pSpScale = NULL;
}
```



```
//Build
void BoxDlg::Build()
{
    cout<<"BoxDlg: Build method called"<<endl;

    //Creates the CATDlgLabel for the input fileds names.

    CATDlgLabel* label_LengthX = new CATDlgLabel(this, CATString("labelX"));
    CATDlgLabel* label_LengthY = new CATDlgLabel(this, CATString("labelY"));
    CATDlgLabel* label_LengthZ = new CATDlgLabel(this, CATString("labelZ"));
    CATDlgLabel* label_Scale = new CATDlgLabel(this, CATString("labelS"));

    //Creates the three spinners
    double Min = 0.001, Max = 1000.0, Step = 0.0254, InitValue = 0.0254;

    //LengthX
    pSpLengthX = new CATDlgSpinner(this, CATString("selX"), CATDlgSpnEntry|CATDlgSpnDouble);
    pSpLengthX->SetMinMaxStep(Min, Max, Step);
    pSpLengthX->SetValue(InitValue, 1);
    pSpLengthX->SetUnit(CATDlgControl::Inch);

    //LengthY
    pSpLengthY = new CATDlgSpinner(this, CATString("selY"), CATDlgSpnEntry|CATDlgSpnDouble);
    pSpLengthY->SetMinMaxStep(Min, Max, Step);
    pSpLengthY->SetValue(InitValue, 1);
    pSpLengthY->SetUnit(CATDlgControl::Inch);

    //LengthZ
    pSpLengthZ = new CATDlgSpinner(this, CATString("selZ"), CATDlgSpnEntry|CATDlgSpnDouble);
    pSpLengthZ->SetMinMaxStep(Min, Max, Step);
    pSpLengthZ->SetValue(InitValue, 1);
    pSpLengthZ->SetUnit(CATDlgControl::Inch);

    //Scale
    pSpScale = new CATDlgSpinner(this, CATString("selS"), CATDlgSpnEntry|CATDlgSpnDouble);
    pSpScale->SetMinMaxStep(Min, Max, Step);
    pSpScale->SetValue(1, 1);
    pSpScale->SetUnit(CATDlgControl::Number);
```



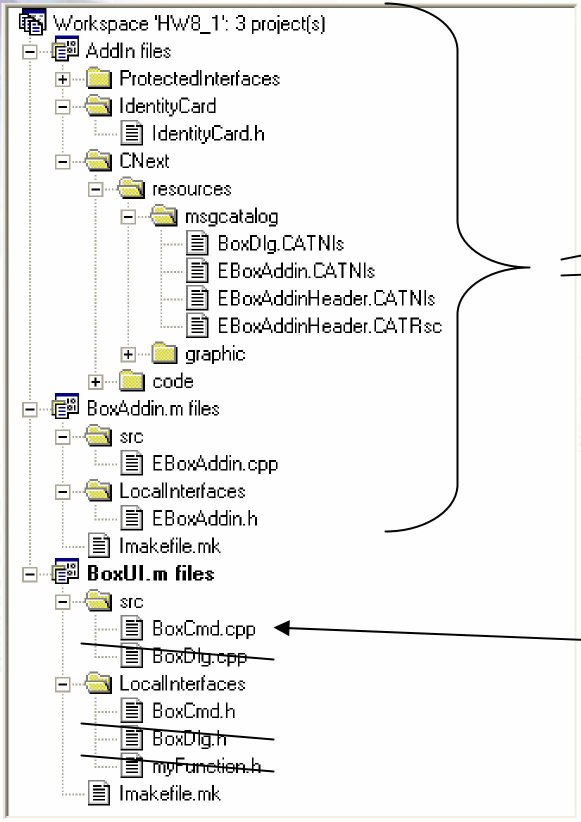
```
    //Uses Grid Constraints to set up the dialog box
    label_LengthX->SetGridConstraints(0, 0, 1, 1, CATGRID_4SIDES);
    pSpLengthX->SetGridConstraints(0, 1, 1, 1, CATGRID_4SIDES);

    label_LengthY->SetGridConstraints(1, 0, 1, 1, CATGRID_4SIDES);
    pSpLengthY->SetGridConstraints(1, 1, 1, 1, CATGRID_4SIDES);

    label_LengthZ->SetGridConstraints(2, 0, 1, 1, CATGRID_4SIDES);
    pSpLengthZ->SetGridConstraints(2, 1, 1, 1, CATGRID_4SIDES);

    label_Scale->SetGridConstraints(3, 0, 1, 1, CATGRID_4SIDES);
    pSpScale->SetGridConstraints(3, 1, 1, 1, CATGRID_4SIDES);

    //Makes the panel visible
    SetVisibility(CATDlgShow);

    cout<<" Build Method ended"<<endl;

}


double BoxDlg::GetLength(int index)
{
    cout<<"BoxDlg: GetLength method called"<<endl;

    //return a length
    switch(index)
    {
    case 1:
        return pSpLengthX->GetValue();
    case 2:
        return pSpLengthY->GetValue();
    case 3:
        return pSpLengthZ->GetValue();
    case 4:
        return pSpScale->GetValue();
    }

}
```

Creating A Form is No Picnic

Object Oriented Options CAA/RADE

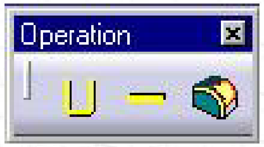
- Added Power
  - Access to More Functions in Most Areas of CATIA
  - Access to Entire Areas That are Not Available in VB
    - Delmia
    - Enovia
    - CATIA Geometric Modeler
    - Machining
    - Etc.
- Added Cost and Complexity

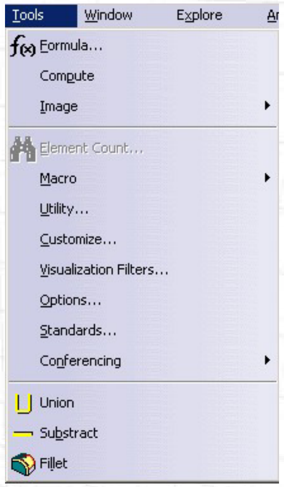
Object Oriented Options – Combined



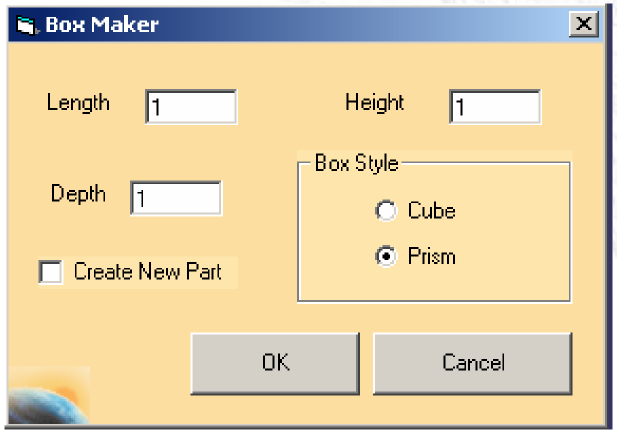
```
Workspace 'HW8_1': 3 project(s)
  AddIn files
    ProtectedInterfaces
    IdentityCard
      IdentityCard.h
    CNext
      resources
        msgcatalog
          BoxDlg.CATNls
          EBoxAddin.CATNls
          EBoxAddinHeader.CATNls
          EBoxAddinHeader.CATRsc
        graphic
      code
  BoxAddin.m files
    src
      EBoxAddin.cpp
    LocalInterfaces
      EBoxAddin.h
    Imakefile.mk
  BoxUI.m files
    src
      BoxCmd.cpp
      ~~BoxDlg.cpp~~
    LocalInterfaces
      BoxCmd.h
      ~~BoxDlg.h~~
      ~~myFunction.h~~
    Imakefile.mk
```

Complex File Structure





The Commands Are Directly Integrated Into CATIA



Box Maker

Length	1	Height	1
Depth	1		

Box Style: ○ Cube [✓ Prism]

☐ Create New Part

OK | Cancel

Call Your VB Form

Founding Partners IBM | DASSAULT SYSTEMES

GOODRICH

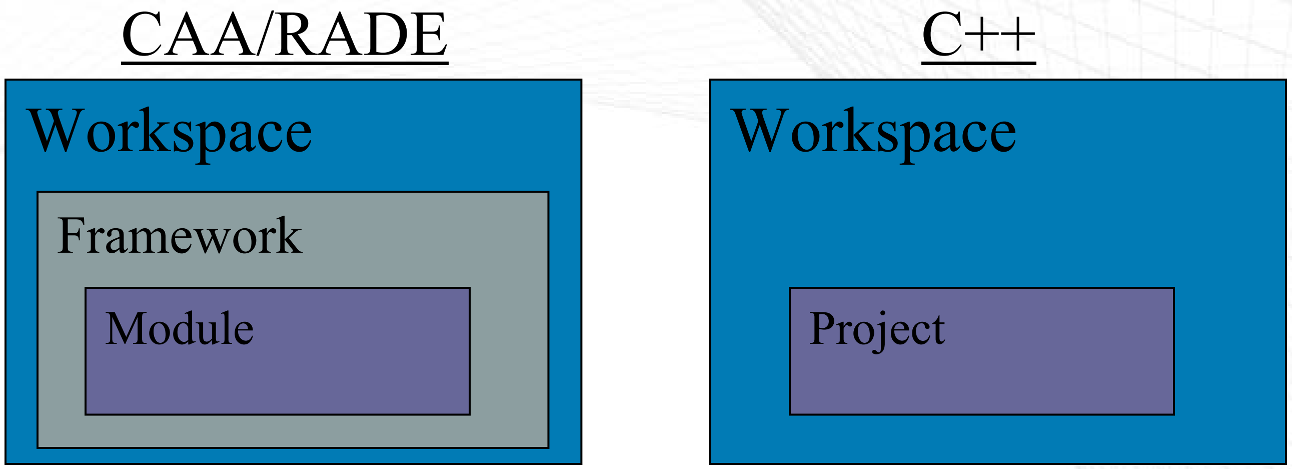
Understanding CAA/RADE – Data Structure (1/3)

- CAA/RADE Has a Different Data Structure than Visual C++
- CAA/RADE Has Three Elements
  - Workspace
  - Framework
  - Module

CAA/RADE	C++
Workspace	Workspace
Framework	
Module	Project

Understanding CAA/RADE – Data Structure (2/3)

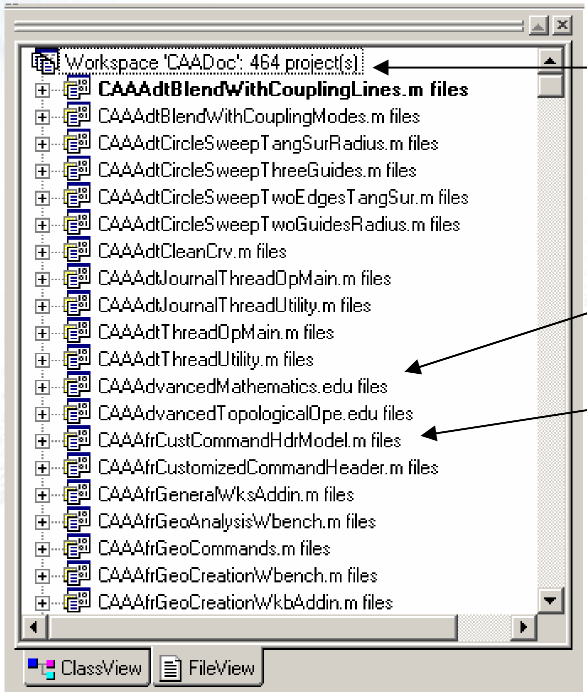
- A Framework Has No C++ Equivalent, but is Similar to a Project
- It is Basically an Added Tier of Organization



Understanding CAA/RADE – Data Structure (3/3)

Hard to Understand This

Easier to Understand This

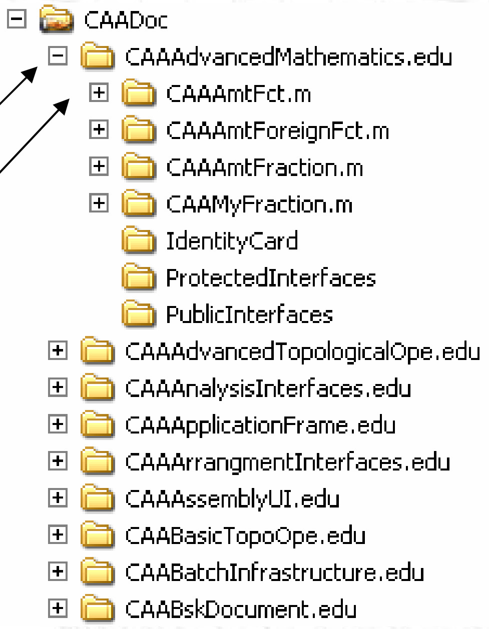


Workspace 'CAADoc': 464 project(s)
- CAAAdtBlendWithCouplingLines.m files
- CAAAdtBlendWithCouplingModes.m files
- CAAAdtCircleSweepTangSurRadius.m files
- CAAAdtCircleSweepThreeGuides.m files
- CAAAdtCircleSweepTwoEdgesTangSur.m files
- CAAAdtCircleSweepTwoGuidesRadius.m files
- CAAAdtCleanCrv.m files
- CAAAdtJournalThreadOpMain.m files
- CAAAdtJournalThreadUtility.m files
- CAAAdtThreadOpMain.m files
- CAAAdtThreadUtility.m files
- CAAAdvancedMathematics.edu files
- CAAAdvancedTopologicalOpe.edu files
- CAAAfrCustCommandHdrModel.m files
- CAAAfrCustomizedCommandHeader.m files
- CAAAfrGeneralWksAddin.m files
- CAAAfrGeoAnalysisWbench.m files
- CAAAfrGeoCommands.m files
- CAAAfrGeoCreationWbench.m files
- CAAAfrGeoCreationWkbAddin.m files

Workspace

Framework

Module

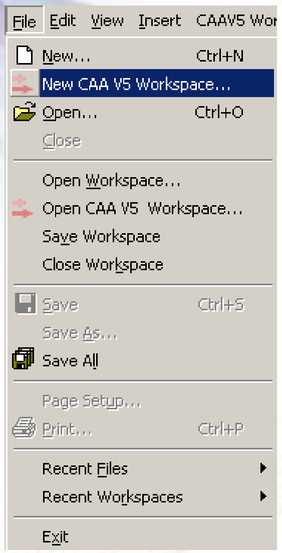


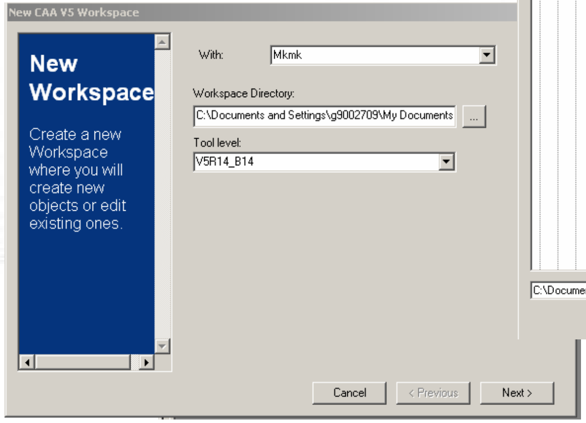
- CAADoc
  - CAAAdvancedMathematics.edu
    - CAAAmtFct.m
    - CAAAmtForeignFct.m
    - CAAAmtFraction.m
    - CAAMyFraction.m
    - IdentityCard
    - ProtectedInterfaces
    - PublicInterfaces
  - CAAAdvancedTopologicalOpe.edu
  - CAAAnalysisInterfaces.edu
  - CAAApplicationFrame.edu
  - CAAArrangmentInterfaces.edu
  - CAAAssemblyUI.edu
  - CAABasicTopoOpe.edu
  - CAABatchInfrastructure.edu
  - CAABskDocument.edu

Careful Naming Conventions and File Organization are Vital!

Understanding CAA/RADE – Example (1/7)

Choose Your folder Carefully Because Many Folders and Files Will be Automatically Generated




New CAA V5 Workspace

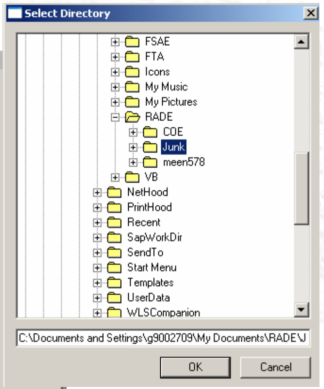
New Workspace

Create a new Workspace where you will create new objects or edit existing ones.

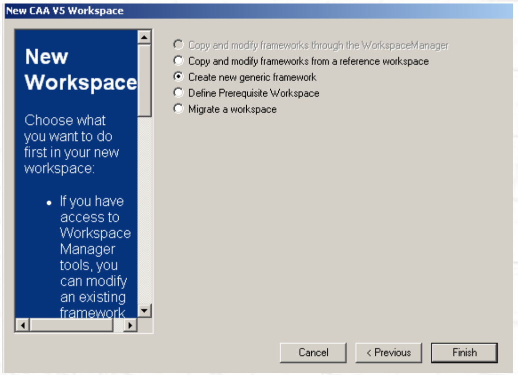
With: Mkmk

Workspace Directory: C:\Documents and Settings\g9002709\My Documents

Tool level: V5R14_B14


Select Directory

C:\Documents and Settings\g9002709\My Documents\RADE\J

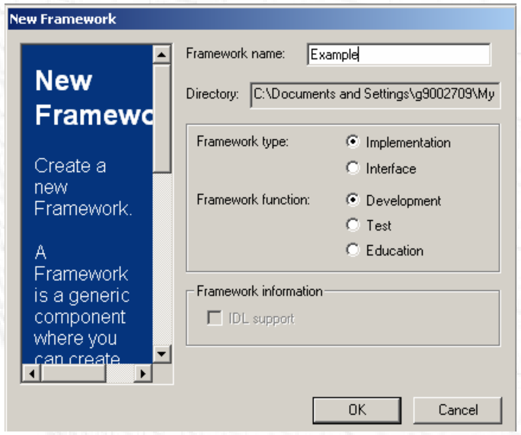

New CAA V5 Workspace

New Workspace

Choose what you want to do first in your new workspace:

- If you have access to Workspace Manager tools, you can modify an existing framework

Copy and modify frameworks through the WorkspaceManager
Copy and modify frameworks from a reference workspace
[✓ Create new generic framework]
Define Prerequisite Workspace
Migrate a workspace


New Framework

New Framework

Create a new Framework.

A Framework is a generic component where you can create

Framework name: Example

Directory: C:\Documents and Settings\g9002709\My

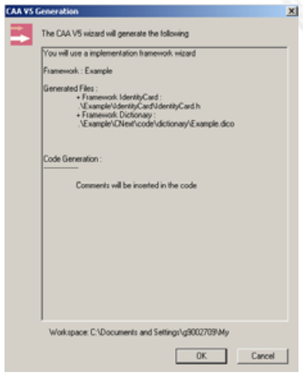
Framework type: [✓ Implementation] Interface

Framework function: [✓ Development] Test Education

Framework information

☐ IDL support

A Workspace is Just a Container, A Framework is a Container With Some Intelligence

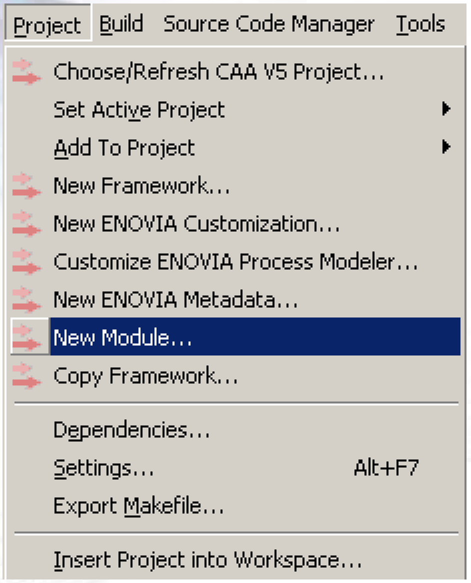

CAA V5 Generation

The CAA V5 wizard will generate the following

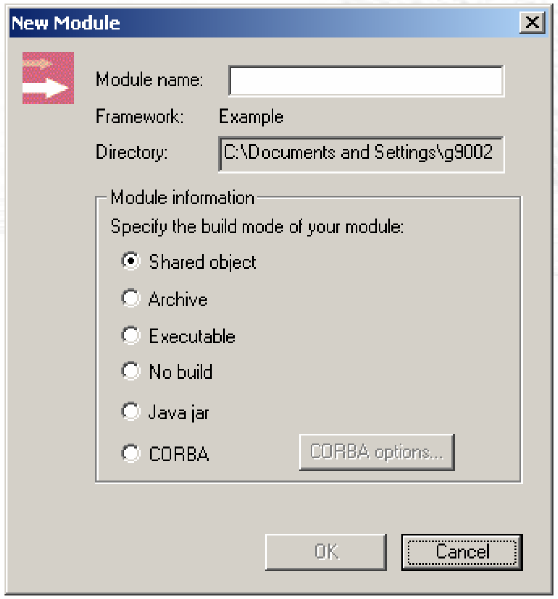
```
You will use a implementation framework wizard

Framework : Example

Generated Files :
        + Framework IdentityCard :
         .\Example\IdentityCard\IdentityCard.h
        + Framework Dictionary :
         .\Example\CNext\code\dictionary\Example.dico



Code Generation :
-----------

        Comments will be inserted in the code
```

Workspace: C:\Documents and Settings\g9002709\My

Understanding CAA/RADE – Example (2/7)

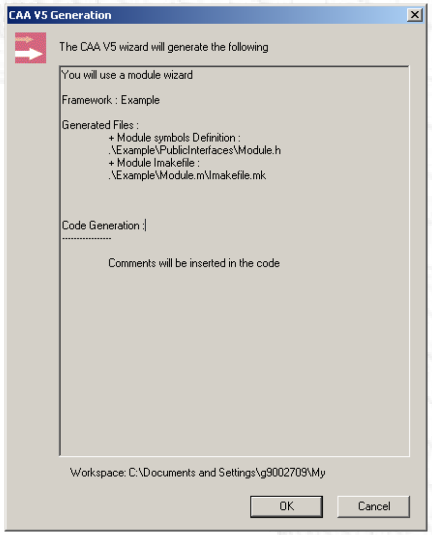

Project Build Source Code Manager Tools
Choose/Refresh CAA V5 Project...
Set Active Project
Add To Project
New Framework...
New ENOVIA Customization...
Customize ENOVIA Process Modeler...
New ENOVIA Metadata...
New Module...
Copy Framework...
Dependencies...
Settings... Alt+F7
Export Makefile...
Insert Project into Workspace...

Visual Studio Organizes both Frameworks and Modules Alphabetically as if They are Both Projects. Good Naming Conventions Can Help Make Framework and Module Navigation Much Less Confusing.


New Module

Module name:
Framework: Example
Directory: C:\Documents and Settings\g9002

Module information
Specify the build mode of your module:
[✓ Shared object]
Archive
Executable
No build
Java jar
CORBA


CAA V5 Generation

The CAA V5 wizard will generate the following

```
You will use a module wizard

Framework : Example

Generated Files :
            + Module symbols Definition :
            .\Example\PublicInterfaces\Module.h
            + Module Imakefile :
            .\Example\Module.m\Imakefile.mk



Code Generation :
-----------------

            Comments will be inserted in the code
```

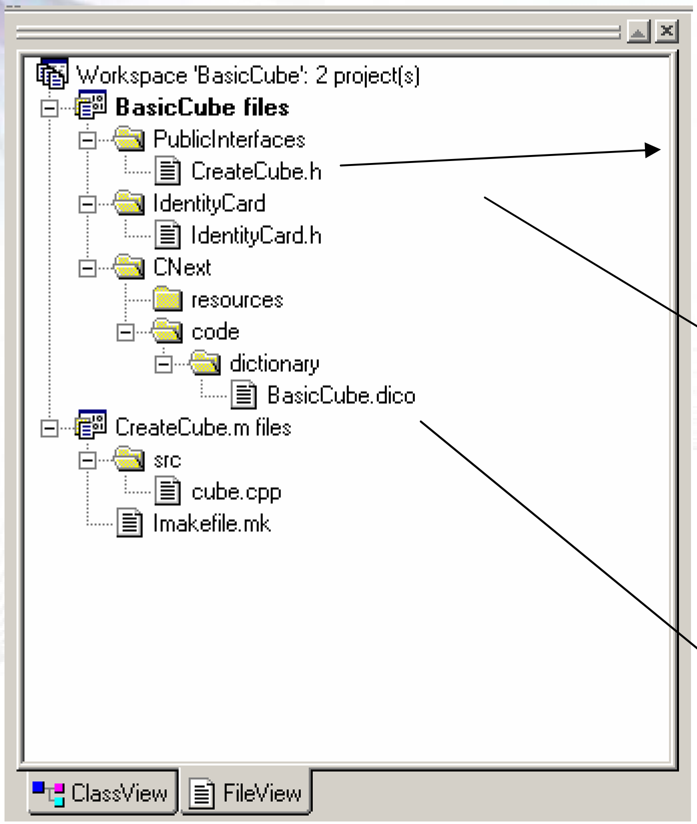
Workspace: C:\Documents and Settings\g9002709\My

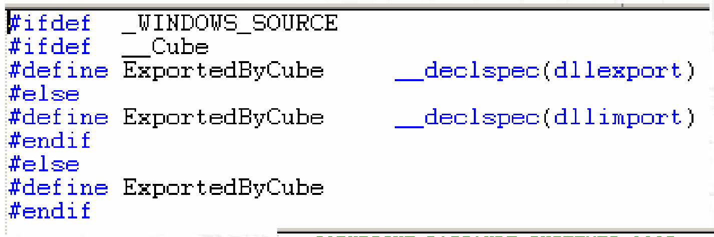
A Module is Where Everything Actually Happens

GOODRICH

Founding Partners IBM DASSAULT SYSTEMES

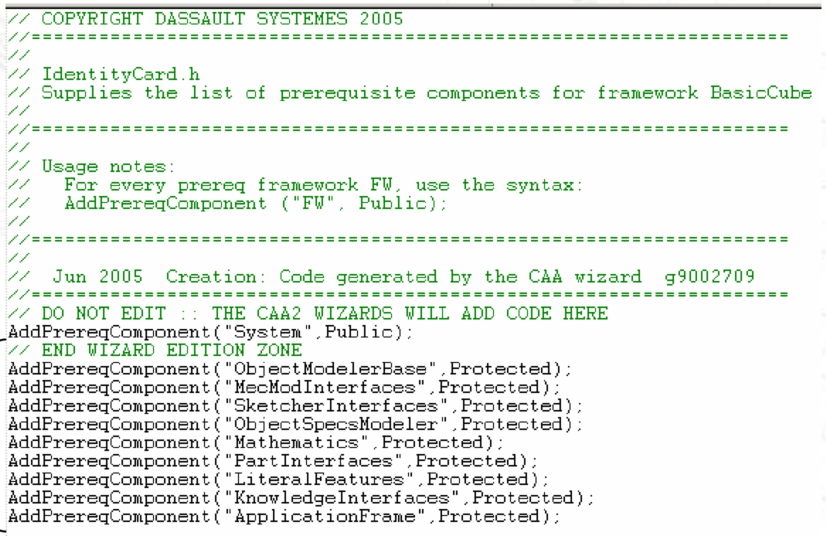
Understanding CAA/RADE – Example (3/7)





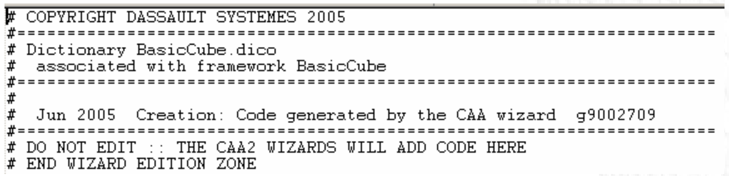
```
#ifdef   _WINDOWS_SOURCE
#ifdef   __Cube
#define ExportedByCube     __declspec(dllexport)
#else
#define ExportedByCube     __declspec(dllimport)
#endif
#else
#define ExportedByCube
#endif
```

These Files Are Generated Automatically



```
// COPYRIGHT DASSAULT SYSTEMES 2005
//======================================================================
//
// IdentityCard.h
// Supplies the list of prerequisite components for framework BasicCube
//
//======================================================================
//
// Usage notes:
//   For every prereq framework FW, use the syntax:
//   AddPrereqComponent ("FW", Public);
//
//======================================================================
//
//  Jun 2005  Creation: Code generated by the CAA wizard  g9002709
//======================================================================
// DO NOT EDIT :: THE CAA2 WIZARDS WILL ADD CODE HERE
AddPrereqComponent("System",Public);
// END WIZARD EDITION ZONE
AddPrereqComponent("ObjectModelerBase",Protected);
AddPrereqComponent("MecModInterfaces",Protected);
AddPrereqComponent("SketcherInterfaces",Protected);
AddPrereqComponent("ObjectSpecsModeler",Protected);
AddPrereqComponent("Mathematics",Protected);
AddPrereqComponent("PartInterfaces",Protected);
AddPrereqComponent("LiteralFeatures",Protected);
AddPrereqComponent("KnowledgeInterfaces",Protected);
AddPrereqComponent("ApplicationFrame",Protected);
```

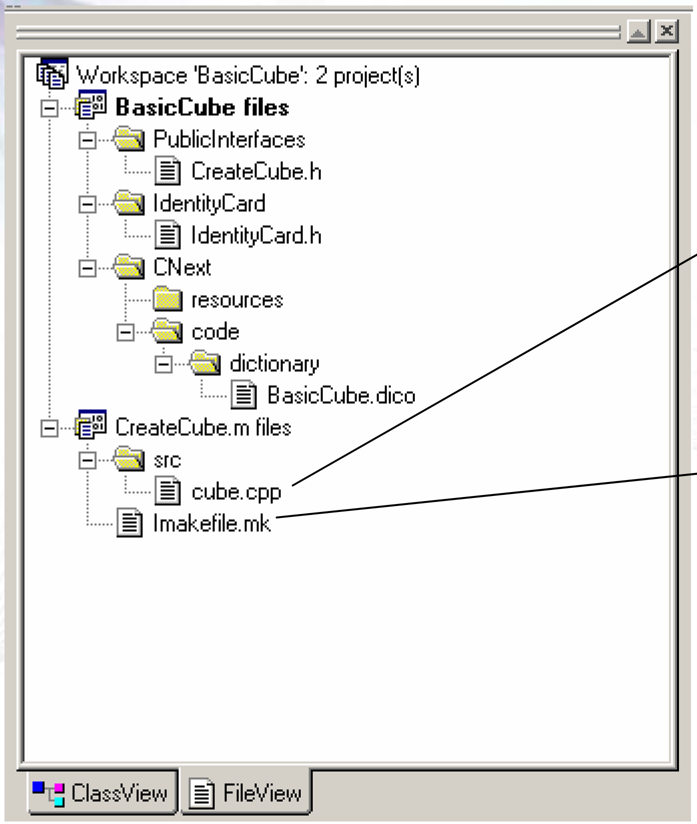
Notice The Prerequisites



```
# COPYRIGHT DASSAULT SYSTEMES 2005
#======================================================================
# Dictionary BasicCube.dico
#  associated with framework BasicCube
#======================================================================
#
#  Jun 2005  Creation: Code generated by the CAA wizard  g9002709
#======================================================================
# DO NOT EDIT :: THE CAA2 WIZARDS WILL ADD CODE HERE
# END WIZARD EDITION ZONE
```

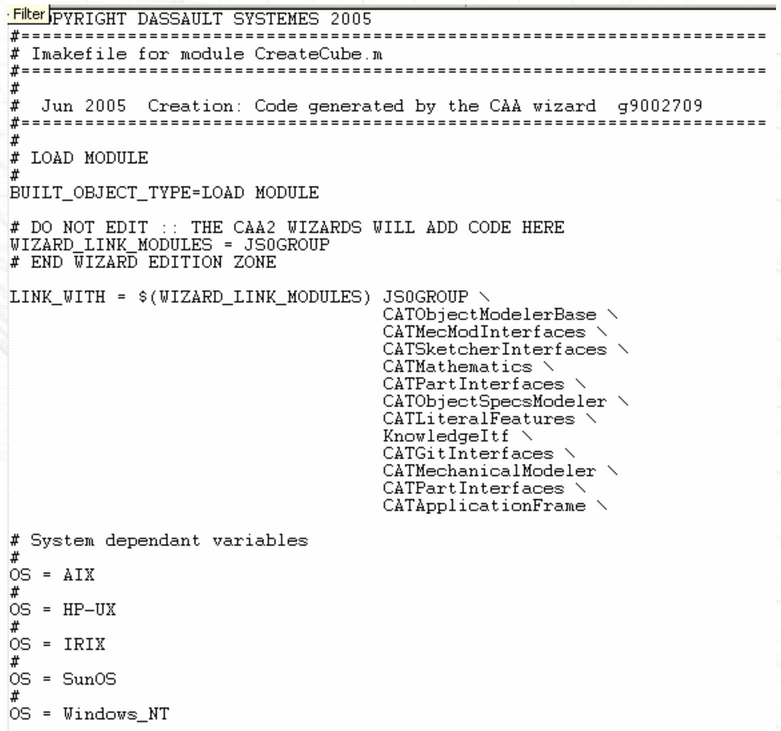
Sometimes You Have to Add them to the Folders Even Though They Already Exist

Understanding CAA/RADE – Example (4/7)



```
Workspace 'BasicCube': 2 project(s)
 BasicCube files
   PublicInterfaces
     CreateCube.h
   IdentityCard
     IdentityCard.h
   CNext
     resources
     code
       dictionary
         BasicCube.dico
 CreateCube.m files
   src
     cube.cpp
   Imakefile.mk
```

We Will Create This .cpp File in the Following Slides

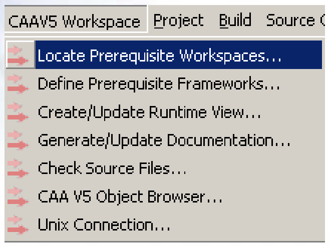


```
PYRIGHT DASSAULT SYSTEMES 2005
#======================================================================
# Imakefile for module CreateCube.m
#======================================================================
#
#  Jun 2005  Creation: Code generated by the CAA wizard  g9002709
#======================================================================
#
# LOAD MODULE
#
BUILT_OBJECT_TYPE=LOAD MODULE

# DO NOT EDIT :: THE CAA2 WIZARDS WILL ADD CODE HERE
WIZARD_LINK_MODULES = JS0GROUP
# END WIZARD EDITION ZONE

LINK_WITH = $(WIZARD_LINK_MODULES) JS0GROUP \
                                   CATObjectModelerBase \
                                   CATMecModInterfaces \
                                   CATSketcherInterfaces \
                                   CATMathematics \
                                   CATPartInterfaces \
                                   CATObjectSpecsModeler \
                                   CATLiteralFeatures \
                                   KnowledgeItf \
                                   CATGitInterfaces \
                                   CATMechanicalModeler \
                                   CATPartInterfaces \
                                   CATApplicationFrame \


# System dependant variables
#
OS = AIX
#
OS = HP-UX
#
OS = IRIX
#
OS = SunOS
#
OS = Windows_NT
```

In This Example, Only cube.cpp is not automatically generated

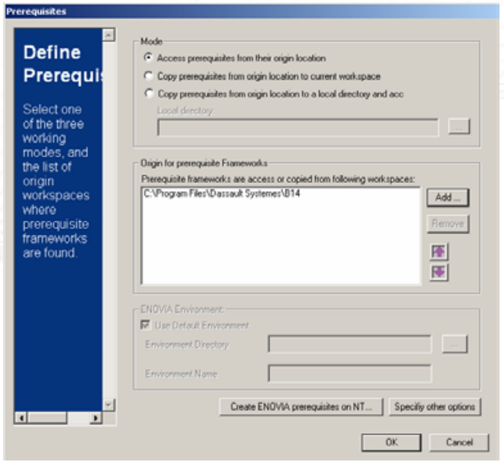
Understanding CAA/RADE – Example (5/7)

Workspaces



CAAV5 Workspace | Project | Build | Source

- Locate Prerequisite Workspaces...
- Define Prerequisite Frameworks...
- Create/Update Runtime View...
- Generate/Update Documentation...
- Check Source Files...
- CAA V5 Object Browser...
- Unix Connection...



Prerequisites

Define Prerequi

Select one of the three working modes, and the list of origin workspaces where prerequisite frameworks are found.

Mode
- [✓ Access prerequisites from their origin location]
- Copy prerequisites from origin location to current workspace
- Copy prerequisites from origin location to a local directory and acc

Local directory:

Origin for prerequisite Frameworks

Prerequisite frameworks are access or copied from following workspaces:

C:\Program Files\Dassault Systemes\B14

Add ... | Remove

ENOVIA Environment:
- [✓ Use Default Environment]

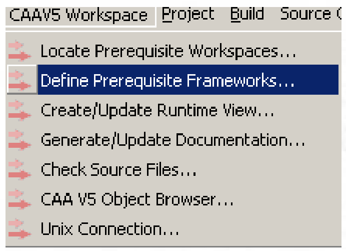
Environment Directory

Environment Name

Create ENOVIA prerequisites on NT... | Specify other options

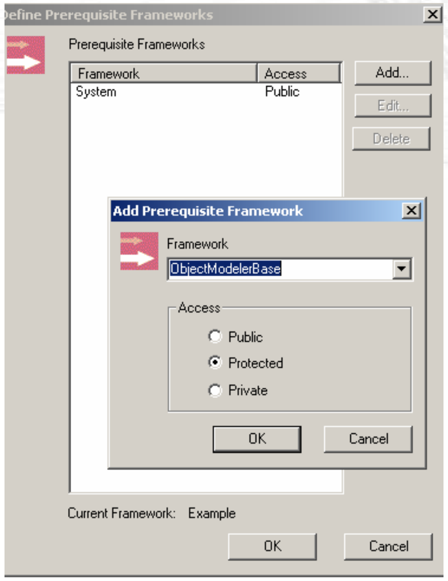
OK | Cancel

Frameworks



CAAV5 Workspace | Project | Build | Source

- Locate Prerequisite Workspaces...
- Define Prerequisite Frameworks...
- Create/Update Runtime View...
- Generate/Update Documentation...
- Check Source Files...
- CAA V5 Object Browser...
- Unix Connection...



Define Prerequisite Frameworks

Prerequisite Frameworks

Framework	Access
System	Public

Add... | Edit... | Delete

Add Prerequisite Framework

Framework: ObjectModelerBase

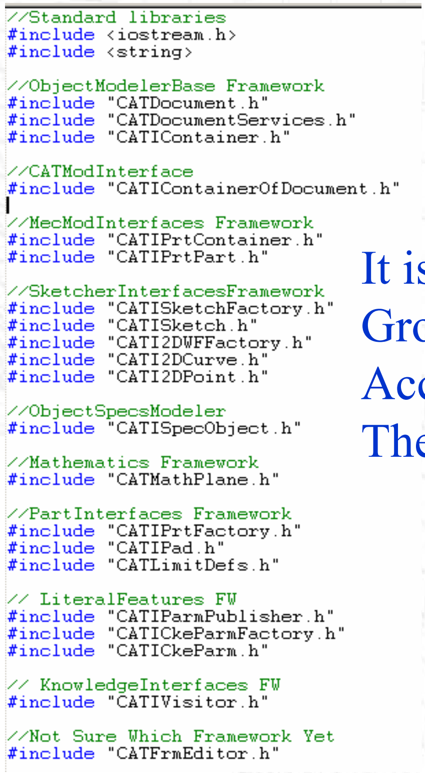
Access
- Public
- [✓ Protected]
- Private

OK | Cancel

Current Framework: Example

OK | Cancel

Modules



```
//Standard libraries
#include <iostream.h>
#include <string>

//ObjectModelerBase Framework
#include "CATDocument.h"
#include "CATDocumentServices.h"
#include "CATIContainer.h"

//CATModInterface
#include "CATIContainerOfDocument.h"

//MecModInterfaces Framework
#include "CATIPrtContainer.h"
#include "CATIPrtPart.h"

//SketcherInterfacesFramework
#include "CATISketchFactory.h"
#include "CATISketch.h"
#include "CATI2DWFFactory.h"
#include "CATI2DCurve.h"
#include "CATI2DPoint.h"

//ObjectSpecsModeler
#include "CATISpecObject.h"

//Mathematics Framework
#include "CATMathPlane.h"

//PartInterfaces Framework
#include "CATIPrtFactory.h"
#include "CATIPad.h"
#include "CATLimitDefs.h"

// LiteralFeatures FW
#include "CATIParmPublisher.h"
#include "CATICkeParmFactory.h"
#include "CATICkeParm.h"

// KnowledgeInterfaces FW
#include "CATIVisitor.h"

//Not Sure Which Framework Yet
#include "CATFrmEditor.h"
```

It is Helpful To Group Modules According to Their Frameworks

GOODRICH

Founding Partners

IBM | DASSAULT SYSTEMES

Understanding CAA/RADE – Example (6/7)

```
void main()
{
// Variables
    double dLenX, dLenY, dLenZ;
    char cName[256];
    HRESULT hr;

// Gets the user name
    cout << "Name the block: ";
    cin >> cName;

// Defines cube size:
    dLenX = 25.4;
    dLenY = 25.4;
    dLenZ = 25.4;

// Opens a CATPart document
    CATDocument* pDoc = GetDocument();

        if(pDoc == NULL)
    {
        cout<<"Error: getDocument"<<endl;

    }
    else
    {
        cout<<"OK: getDocument"<<endl;
    }

// Initializes the data
    CATIContainerOfDocument_var spDoc = (CATIContainerOfDocument_var)pDoc;
    CATIContainer *piContainer = NULL;
     if ( FAILED(spDoc->GetSpecContainer( piContainer)) )
     {
         cout<<"Error: GetSpecContainer"<<endl;

     }
     else
     {
         cout<<"OK: GetSpecContainer"<<endl;
     }
```

Some Things are Just Like C++

Most Things Are a Little Different

```
// Retrieves the root container
    CATIPrtContainer *piPrtCont = NULL;
    hr = piContainer->QueryInterface( IID_CATIPrtContainer, (void **)&piPrtCont );

   if(FAILED(hr))
   {
       cout << "ERROR: piPrtCont" << endl;

   }
   else
   {
       cout << "OK: piPrtCont" << endl;
   }
// Retrieves the sketch factory
   CATISketchFactory_var spSketchFactOnPrtCont(piPrtCont);

// Creates the sketch plane and retrieves the part reference planes
   CATIPrtPart_var spPart = piPrtCont->GetPart();
   CATListValCATISpecObject_var spListRefPlanes = spPart -> GetReferencePlanes();

// Defines the xy plane as the first sketch plane
   CATISpecObject_var spSketchPlane = spListRefPlanes[1];

// Instantiate the sketch
   CATISketch_var spSketch = spSketchFactOnPrtCont->CreateSketch(spSketchPlane);

// Retrieves the 2D Factory
   CATI2DWFFactory_var spWF2DFactOnSketch(spSketch);

// Creates the points
   double pt1[2] = {dLenX, dLenY},
          pt2[2] = {0., dLenY},
          pt3[2] = {0., 0.},
          pt4[2] = {dLenX, 0.};

// Opens the sketch and draws lines
   spSketch->OpenEdition();

   CATISpecObject_var spLine1 = spWF2DFactOnSketch->CreateLine(pt1,pt2);
   CATISpecObject_var spLine2 = spWF2DFactOnSketch->CreateLine(pt2,pt3);
   CATISpecObject_var spLine3 = spWF2DFactOnSketch->CreateLine(pt3,pt4);
   CATISpecObject_var spLine4 = spWF2DFactOnSketch->CreateLine(pt4,pt1);
```

Understanding CAA/RADE – Example (7/7)

```
// Opens the sketch and draws lines
    spSketch->OpenEdition();

    CATISpecObject_var spLine1 = spWF2DFactOnSketch->CreateLine(pt1,pt2);
    CATISpecObject_var spLine2 = spWF2DFactOnSketch->CreateLine(pt2,pt3);
    CATISpecObject_var spLine3 = spWF2DFactOnSketch->CreateLine(pt3,pt4);
    CATISpecObject_var spLine4 = spWF2DFactOnSketch->CreateLine(pt4,pt1);

//Defines end points
    CATI2DCurve_var spCurveOnLine1(spLine1);
    spCurveOnLine1->GetStartPoint();
    spCurveOnLine1->GetEndPoint();
    CATI2DCurve_var spCurveOnLine2(spLine2);
    spCurveOnLine2->GetStartPoint();
    spCurveOnLine2->GetEndPoint();
    CATI2DCurve_var spCurveOnLine3(spLine3);
    spCurveOnLine3->GetStartPoint();
    spCurveOnLine3->GetEndPoint();
    CATI2DCurve_var spCurveOnLine4(spLine4);
    spCurveOnLine4->GetStartPoint();
    spCurveOnLine4->GetEndPoint();

//Closes the sketch session
    spSketch->CloseEdition();

//Defines the direction of the pad
    CATMathDirection dirZ(0., 0., 1.);

//Retrieves the Mechanical Design Factory to create pad
    CATIPrtFactory_var spPrtFactOnPrtCont(piPrtCont);

//Creates pad
    CATISpecObject_var spSpecObj = spPrtFactOnPrtCont->CreatePad(spSketch);

    CATIPad_var spPadOnSpecObj(spSpecObj);
    spPadOnSpecObj->ModifyDirection(dirZ);
    spPadOnSpecObj->ModifyEndType(catOffsetLimit);
    spPadOnSpecObj->ModifyEndOffset(dLenZ);
    spPadOnSpecObj->ModifyStartType(catOffsetLimit);
    spPadOnSpecObj->ModifyStartOffset(0.);
```

The Flow and Format Parallels VB

```
//Builds pad "Update"
    spSpecObj->Update();

    CATIParmPublisher_var spPublisher = spPart;

    CATICkeParmFactory_var spParamFact = piPrtCont;
    piPrtCont->Release();
    piPrtCont = NULL;

    CATLISTV(CATBaseUnknown_var) ParamList;
    CATIVisitor_var spStandardVisitor =
        spParamFact->CreateStandardVisitor(IID_CATICkeParm, &ParamList);

    spPublisher->VisitChildren(spStandardVisitor, 1);

    CATICkeParm_var tempParam = NULL_var;
    CATUnicodeString tempUnicode = "CATPart1\\Definition";

    for (int j=1; j<=ParamList.Size(); j++)
    {
        tempParam = ParamList[j];
        if (tempUnicode == tempParam->Name())
        {
            tempParam->Valuate(cName);
        }
    }

    spSpecObj->Update();

//Saves the pad
    char tempName[256];
    sprintf(tempName,"C:\\%s.CATPart",cName);
    CATDocumentServices::SaveAs(*pDoc, tempName);

//Closes the section

    cout << "Program ended successfully" << endl;
    cout << "Part was saved at " << tempName << endl;
}
```

Summary

- Which Programming Methods You Use is an Important Decision.
- We Have Chosen to Use a Combination of VB and CAA/RADE Because That Best Fits Our Organization’s Needs.
- Taking The Time to Understand the Structure of CAA/RADE can be Difficult, but is Very Important.

Questions?

All The Best

-From-

Vidyaputra

(DHIRAJ S. DVIVEDI)

STAY TUNED..STAY UPDATED AND SMART……

(Subscribe us for update and notifications by clicking on follow tab which is located at your right bottom corner of website)