

HFSS 3D Layout

二次開發概論



模擬遇到的挑戰

- 設計創新與多樣性

- 台灣的半導體及電子產業已居於世界領先地位。產業的多樣性對於EDA的需求也大不相同。再加上許多設計的創新與發想更迭速度極快，EDA工具更新速度已經無法滿足研發人員的要求。這需要研發人員基於現有的工具上做二次開發來彌補EDA工具的不足之處。

- 複雜的設定

- 電子系統的複雜度的增加。模擬工程師已經無法快速掌握系統全貌並完成模擬設定，必須有新的工法或流程來加快設定及模擬的速度。這也可以透過二次開發來解決。

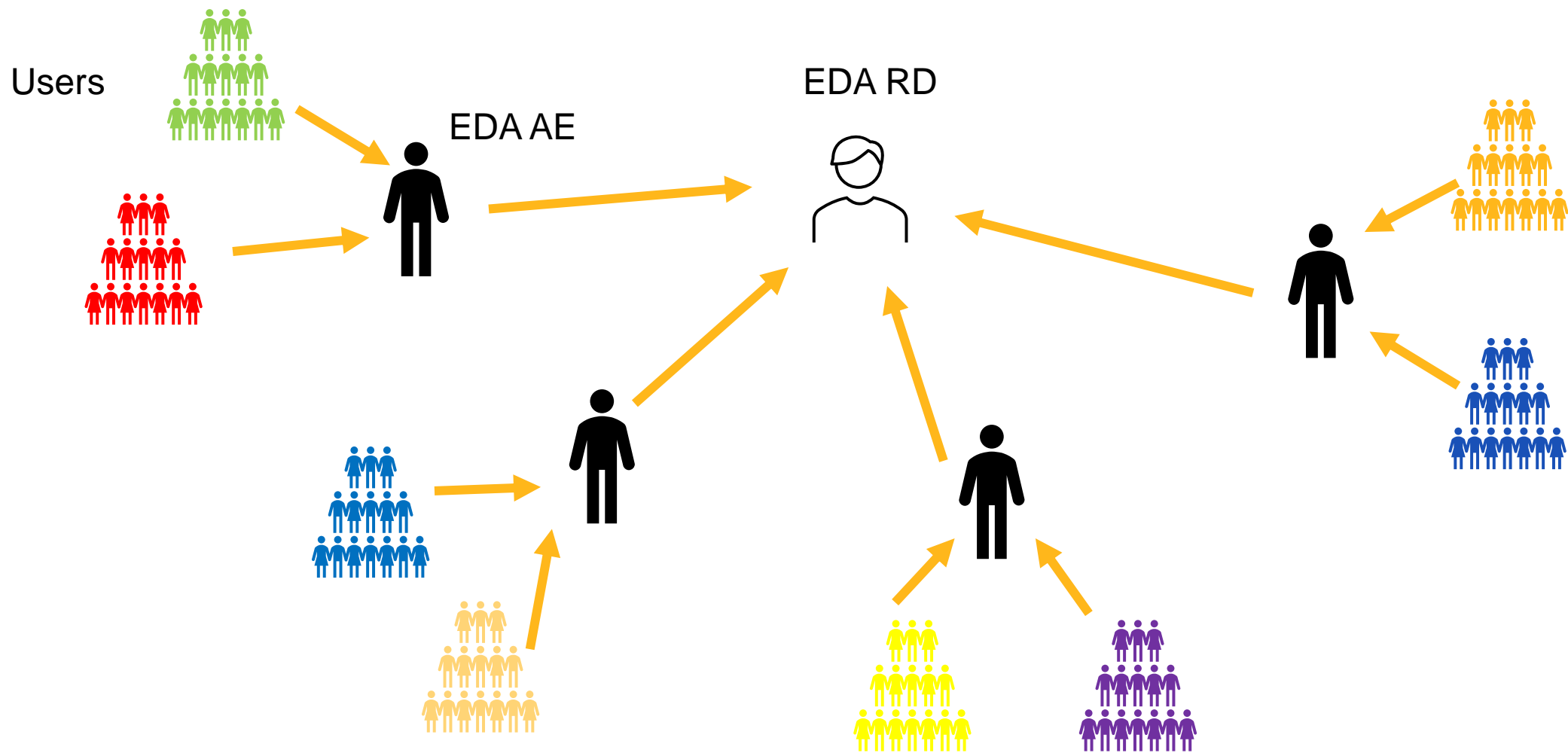
- 有限的資源

- EDA演算法的進步速度無法跟上系統複雜度增加的速度，模擬的計量單位已從小時到天甚至到以周為單位。記憶體也從數十GB到數TB。結構的簡化勢在必行。二次開發對於簡化工作帶來極大的效益。

- 龐大的資料

- EDA工具の後處理能力對應付龐大的模擬資料已顯得力不從心，必須借助外部工具的協助。統計及資料處理是二次開發的關鍵能力。

客戶需求反饋流程

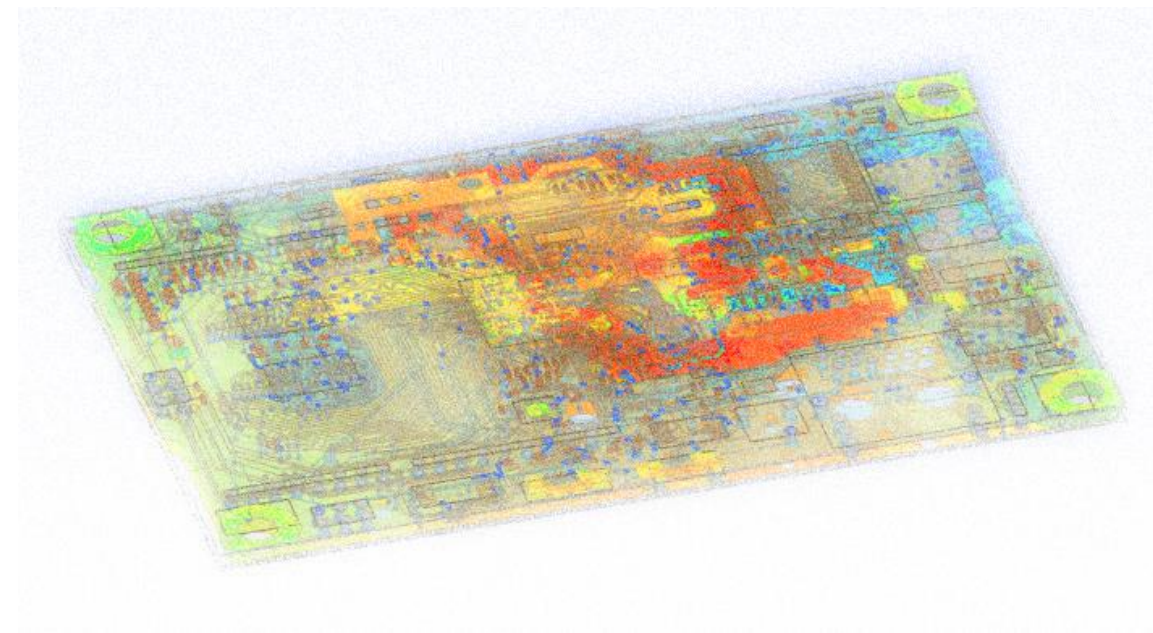
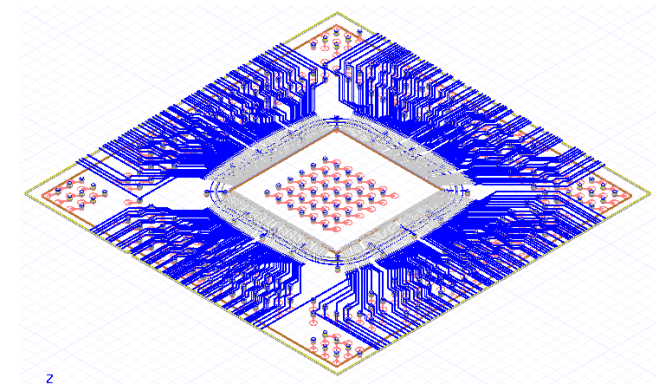


二次開發

- 二次開發簡單的說就是在現有的軟體上進行定製修改，功能的擴展，然後達到自己想要的功能，不會改變原有系統的核心。舉例來說，一些大公司開發了一個大型的軟體系統平台，使用者根據需求在該平台上進行第二次有針對性的開發。
- 能力要求：
 - 要有這個開源產品的所用語言的語言基礎。
 - 要對這個開源產品的功能和使用要有比較熟悉，因為你熟悉了，你才知道一個需求下來，你要改什麼，什麼是系統自帶的，大概要怎麼改。
 - 要熟悉這個開源產品的資料結構，代碼結構，系統的架構結構，核心是哪裏，附屬功能是在哪裏。簡單點說，就是資料庫，代碼邏輯，檔案目錄的熟悉。如果是用接口式的二次開發，則需要你對這個接口比較熟悉，一般來說會有相應的文檔。
 - 根據你的需求，然後利用開源產品的核心，進行系統的擴展和修改，以達到你的需求。
 - 對其提供的SDK中的API函式有一定了解，以利于你對SDK中個函式的使用更加靈活方便

3D Layout二次開發

- 3D Layout適合PCB或封裝等多層平面結構處理
- 適當的物件分類(net, layer, component, polygon)
- 可錄製Python腳本
- 完整的查詢函式
- 可匯出HFSS/Q3D/Slwave/Icepak專案



Semi-Automation (半自動化)

- 結合人工判斷與自動化操作的模擬自動化流程
- 對於某些高複雜度的Layout設計，全自動化難度較高，可以採用半自動化的流程來加速設定流程
- 例子：
 - Stackup的設置
 - Bondwire端點位置的設定
 - 元件型別的切換
 - 設計的修改

/ 範例演示

新型的結構

- TSV
- Multi Die
- G Shape Via

複雜度

- >100 layers
- Million vias (interposer)
- PDN

結構的簡化

- Cutout
- Adjustment
- Remove

大量資料處理

- 統計分析
- 報表產生

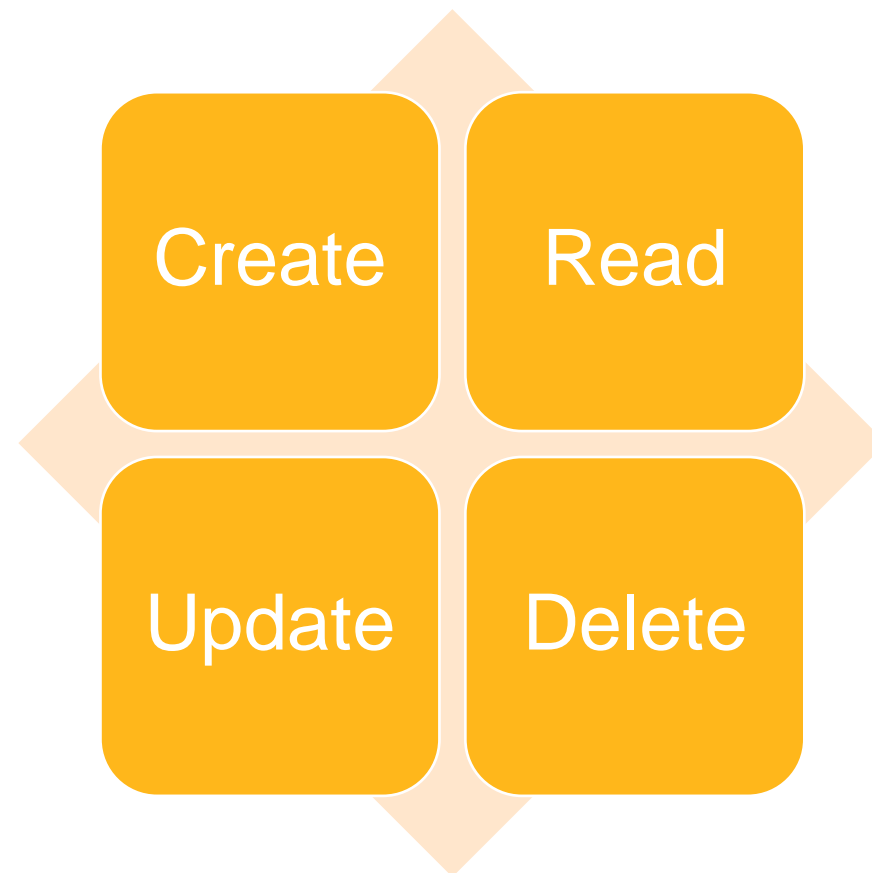
3D Layout Toolkits Demo

- How to Fine Tune Location of Bondwires in 3D Layot
 - <https://www.youtube.com/watch?v=0MIHqLXk224>
- How to Setup PCB Stackup in CSV Format
 - <https://www.youtube.com/watch?v=DvjDggmajGg>
- How to Display Insertion Loss of S Parameter Quickly
 - <https://www.youtube.com/watch?v=3YbZ3wwaFrM>
- How to Setup Hotkey for Script
 - <https://www.youtube.com/watch?v=zK4O5TVZPeM>

3D Layout資訊查詢

/ CRUD

- CRUD全稱
 - 增加 (Create)
 - 查詢 (Read)
 - 更新 (Update)
 - 刪除 (Delete)
- 在二次開發當中，C、U、D都是可以透過錄製快速取得程式碼範本，唯獨R不行，因此接下來介紹Read的相關函數。



3D Layout自動化範例

- 找出(R)所有VDD/VSS Net的bond-wires並將其寬度改成30um(U)
- 找出(R)所有方形poly void並將其刪除(D)
- 找出(R)所有在某polygon上的poly void並將其刪除(D)
- 在元件所有的signal pads上面(R)產生ports(C)

3D Layout函式庫列表

20R2_3D Layout

oDesktop(69)

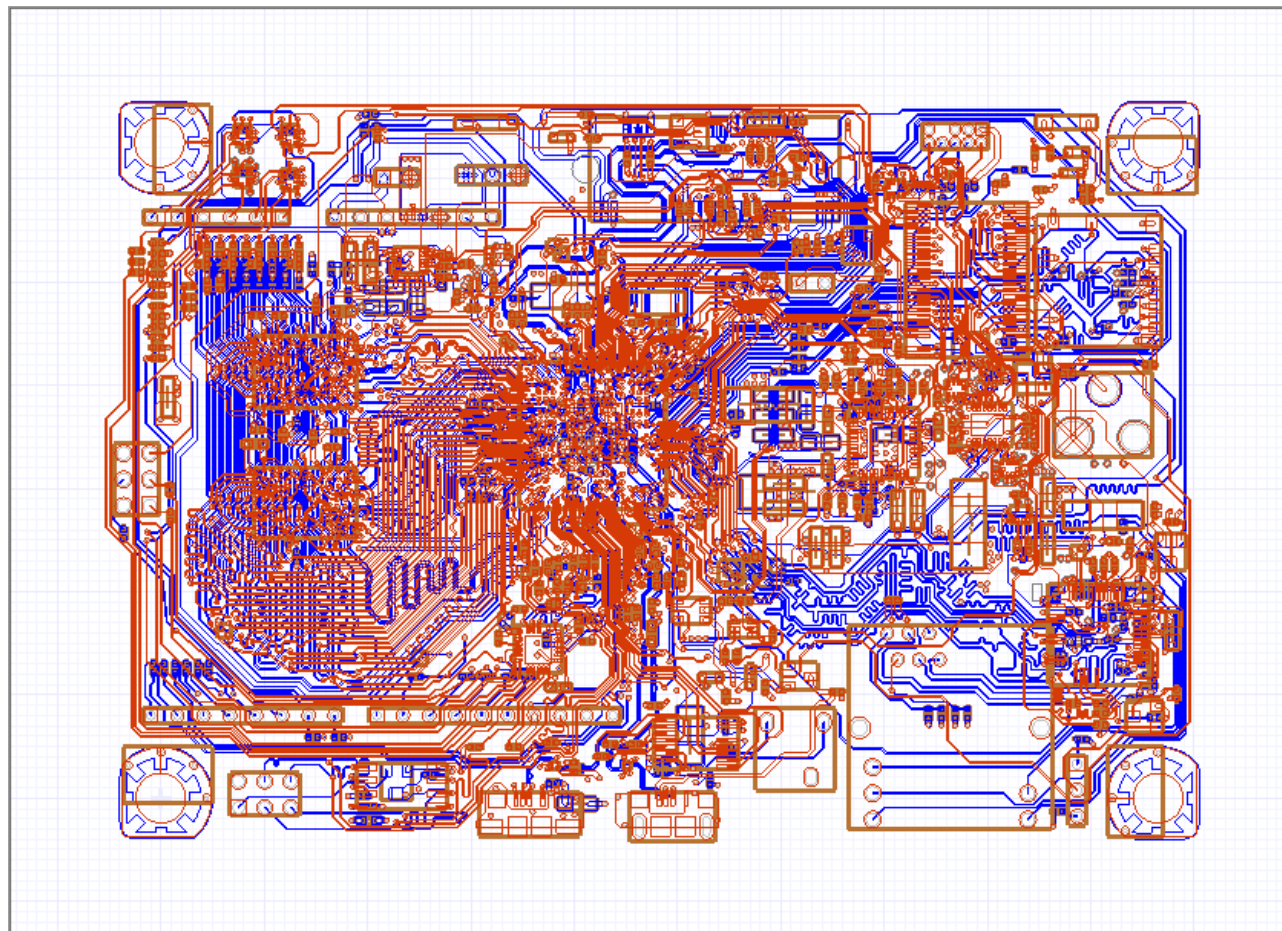
AddMessage	ClearMessages	CloseAllWindows
CloseProject	CloseProjectNoForce	DeleteProject
DeleteRegistryEntry	DoesRegistryValueExist	DownloadJobResults
EnableAutoSave	ExportOptionsFiles	GetActiveProject
GetAutoSaveEnabled	GetBuildDateTimeString	GetDefaultUnit
GetDistributedAnalysisMachines	GetDistributedAnalysisMachinesForDesignType	GetExeDir
GetGDIObjectCount	GetLibraryDirectory	GetLocalizationHelper
GetMessages	GetPersonalLibDirectory	GetProcessID
GetProjectDirectory	GetProjectList	GetProjects
GetRegistryInt	GetRegistryString	GetRunningInstancesMgr
GetScriptingToolsHelper	GetSysLibDirectory	GetTempDirectory
GetTool	GetUserLibDirectory	GetVersion
LaunchJobMonitor	NewProject	OpenAndConvertProject
OpenMultipleProjects	OpenProject	OpenProjectWithConversion
PageSetup	PauseRecording	PauseScript
Print	QuitApplication	RefreshJobMonitor
ResetLogging	RestoreProjectArchive	RestoreWindow
ResumeRecording	RunACTWizardScript	RunProgram
RunScript	RunScriptWithArguments	SelectScheduler
SetActiveProject	SetActiveProjectByPath	SetLibraryDirectory
SetProjectDirectory	SetRegistryFromFile	SetRegistryInt
SetRegistryString	SetTempDirectory	ShowDockingWindow
Sleep	SubmitJob	TileWindows

oProject(61)

AddDataset	AddMaterial	AnalyzeAll
ChangeProperty	ClearMessages	CloneMaterial
Close	CopyDesign	CutDesign

3D Layout的資訊

- Stackup
 - Layers
 - Materials
- Nets
 - Vias
 - Length
- Components
 - Types and Models
 - Pads
- Ports
- Primitives
 - Polygon, Square, Circle, Line, Point,...



3D Layout Editor的查詢函數

- 在AEDT的函式庫當中，C、U、D都是可以錄製的，唯獨R必須透過自行編寫來完成。多數以Get開頭，比方說GetAllLayerNames()，GetPropertyValue()等等
- 以下這些函數可以執行多數Layout資訊的查詢。

ExportToHFSSUDP

FindObjects

FlipHorizontal

GeometryCheckAndAutofix

GetBBox

GetComponentInfo

GetEditorName

GetNetClassNets

GetNets

GetPolygonInfo

GetPortInstances

GetSelections

ExportToQ3D

FindObjectsByPoint

FlipVertical

GetActiveUnits

GetCSObjects

GetComponentPinInfo

GetLayerInfo

GetNetClasses

GetPolygon

GetPolygonVoids

GetProperties

GetStackupLayerNames

FilterObjectList

FindObjectsByPolygon

GenerateFacetDataForModel

GetAllLayerNames

GetCompInstanceFromRefDes

GetComponentPins

GetMaterialList

GetNetConnections

GetPolygonDef

GetPortInfo

GetPropertyValue

GetUserGeometryList

FindObjects() 函數

```
oProject = oDesktop.GetActiveProject()
oDesign = oProject.GetActiveDesign()
oEditor = oDesign.SetActiveEditor("Layout")
```

```
stackup=[]
objs=[]
stackup.append(('layers',len(oEditor.GetStackupLayerNames()))))
stackup.append(('nets',len(oEditor.GetNets()))))
```

```
types=['bondwire','pin','via','rect','arc','line','poly','plg','circle void','line void','rect void','poly void','plg void','text','cell','Measurement',
'Port','Port Instance','Port','Instance Port','Edge Port','component','CS','S3D','Via Group','PinGroup']
```

```
x = {}
for t in types:
    x[t] = oEditor.FindObjects('Type', t)
AddWarningMessage(str(x))
```

```
x = oEditor.FindObjects('Layer', 'Top')
AddWarningMessage(str(x))
```

```
x= oEditor.FindObjects('Net', 'M_DQ<0>')
AddWarningMessage(str(x))
```

/ Data Query

- Stackup
 - GetAllLayerNames()
 - GetLayerInfo()
 - GetMaterialList()
 - GetStackupLayerNames()
- Net
 - GetNetClasses()
 - GetNetClassNets()
 - GetNetConnections()
- Component
 - GetCompInstanceFromRefDes()
 - GetComponentInfo()
 - **GetComponentPinInfo()**
 - GetComponentPins()
- 一般屬性查詢
 - GetProperties()
 - GetPropertyValue()

幾何物件查詢與操作

- Polygon

- GetPolygonDef()
- GetPolygonInfo()
- GetPolygonVoids()

- Polygon Methods

- AddPoint()
- SetClosed()
- IsClosed()
- Move()
- Rotate()
- Scale()
- MirrorX()

- GetPoints()
- AddHole()
- HasHoles()
- GetHoles()
- HasArcs()
- HasSelfIntersections()
- BBoxLL()
- BBoxUR()
- IsParametric()
- IsConvex()
- IsPoint()
- IsSegment()
- IsArc()
- IsBox()

- IsCircle()
- GetBounding()
- CircleCenter()
- GetBoundingCircleRadius()
- Area()
- PointInPolygon()
- CircleIntersectsPolygon()
- GetIntersectionType()
- GetClosestPoint()
- Unite()
- Intersect()
- Subtract()
- Xor()

/ 找出所有在某polygon上的poly void並將其刪除

```
import ScriptEnv

ScriptEnv.Initialize("Ansoft.ElectronicsDesktop")
oDesktop.RestoreWindow()
oProject = oDesktop.GetActiveProject()
oDesign = oProject.GetActiveDesign()
oEditor = oDesign.GetActiveEditor()

x = oEditor.FindObjects('Type', 'poly void')

y = [i for i in x if 'Pt4' not in oEditor.GetProperties('BaseElementTab', i)]

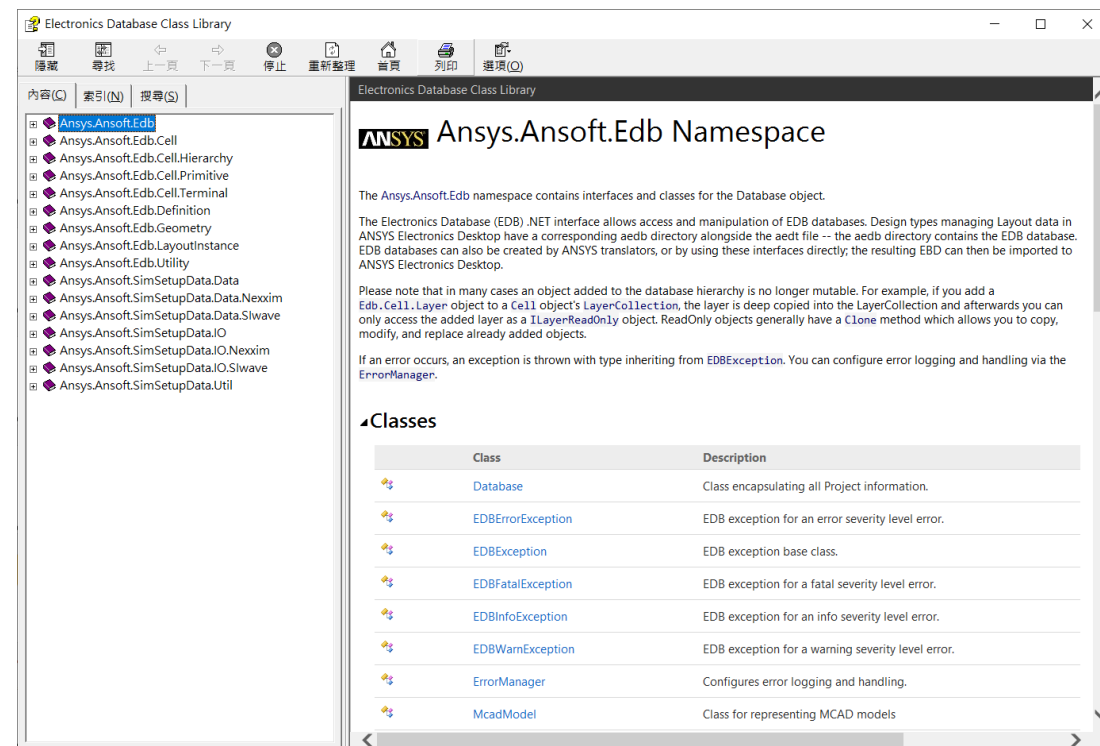
land = oEditor.GetPolygonDef('poly_743')
todelete = []
for i in y:
    poly_i = oEditor.GetPolygonDef(i)
    if land.GetIntersectionType(poly_i) != 0:
        todelete.append(i)

AddWarningMessage(str(todelete))
oEditor.Delete(todelete)
```

EDB框架簡介

EDB 函式庫

- 函式庫文檔
 - C:\Program Files\AnsysEM\AnsysEM20.2\Win64\Help\Edb\Documentation.chm
- 少數3D Layout函式庫無法達成的功能我們會借助EDB庫的函式來達成
- 抓取stackup設定：
 - <https://aedt-sim-automation.blogspot.com/2020/11/edb.html>
- 缺點：
 - 以C#框架來組織函式庫，
 - 專有名詞不易理解
 - 缺乏範例代碼



 **Ansys**

