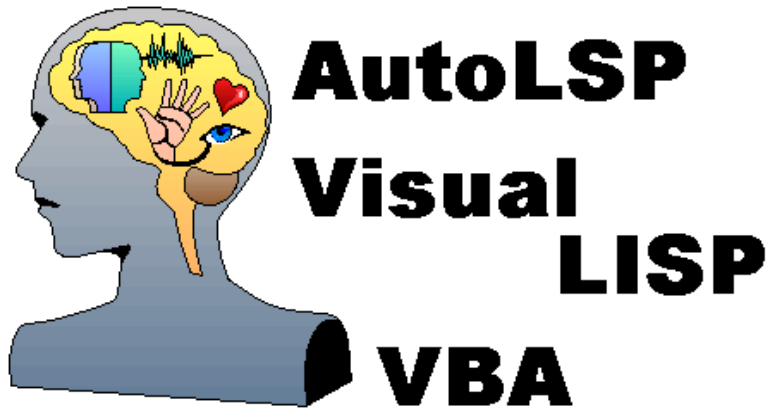


# 第一章 AutoCAD 程式設計概論

在本章中，我們將先為您介紹 AutoCAD 的程式設計概論，以讓您在學習它們以前，能有正確的觀念。



## 1-1 AutoLISP 概論

AutoLISP 語言是從 AutoCADv2.17 版開始略具雛型的。當時，人們對這種附於 CAD 軟體裡的高階語言雖陌生但卻充滿興趣。那個時候，LISP 的正統版本很多，如 Common LISP、Golden LISP、Turbo LISP ....等。而 AutoLISP 就是取正統的 LISP 語法，再加以 AutoCAD 本身的特有指令功能而成。怎麼說呢？因為一般正統的 LISP 與 AutoLISP 在輸入與計算方面的語法是一樣的，但是在輸出方面（即顯示成圖形），AutoLISP 因為有整個 AutoCAD 的畫圖功能來當後盾，所以輸出時，只要去呼叫 AutoCAD 的畫圖指令就行了；但是一般正統的 LISP 卻要自己去想辦法讓圖形顯示出來（換句話說，就是還要自己設計圖形副程式庫），對一般使用者來說，這將是一龐大的工程，也導致正統 LISP 的殒落。但是對 AutoLISP 來說，雖然 LISP 的語法艱難，但是有 AutoCAD 做後盾，隨著 AutoCAD 版本的更新與指令的成熟豐富，AutoLISP 一直盛極不衰。

到了 AutoCAD R12 版，因為在該版本裡大量出現的交談式視窗畫面是根據 DCL 語言來設計的，所以為了讓使用者也能設計交談式視窗，遂讓 AutoLISP 加入可控制 DCL 的語法。DCL 全名為 Dialogue Control Language，它與 AutoLISP 一樣，只要將一使用文字處理程式並依其語法撰寫而成的文字檔（其副檔名為 .DCL）放在 AutoLISP 可以呼叫到的地方，再配合螢幕功能表設計就可以執行它們了。這也是 AutoLISP 自歷次版本中改變最大的地方。AutoLISP 各版次新增的語法功能都是該版 AutoCAD 所新增的指令或功能。

## 1-2 AutoCAD 的程式語言演進

由於 AutoLISP 存在自身無法解決的問題，爲了適應 AutoCAD 環境下的較大型使用者化 CAD 應用系統的開發，Autodesk 公司就從 AutoCAD R11 版開始，推出採用了以 C 語言爲主的開發環境 ADS (Advance Development System)，藉以使用 C/C++ 語言的性能和資源，來完成 AutoLISP 語言難以做到的任務。

隨著 Windows 作業系統、物件導向程式設計等技術日益成熟與普及，AutoCAD 也和其他系統一樣迫切需要一種全新的面向物件編程的開發環境。於 AutoCAD R13 版以後，Autodesk 就又推出了新一代物件導向的二次開發工具，就是 ARX (AutoCAD Runtime Extension，即 C 語言) 與更新的 AutoCAD R14 Object ARX SDK 開發套件。爲開發高自動化程度、高整合性與高性能的使用者化 CAD 系統提供了一種極爲有效的工具。這也是 AutoCAD 目前顧客化的主流開發工具。

然而，不論是 ADS、ARX 或是 Object ARX SDK 環境，它們都是以 C/C++/VC 爲基礎的程式語言。對於全球超過 180 萬 AutoCAD 合法使用者中的多數使用者來說，要在短時期內學好 C/C++/VC 程式語言，並熟練用於 AutoCAD 系統和程式開發是一件相當苛求的事。而且從 1986 年引入 AutoLISP 以來，全球大量的電腦軟體開發商和使用者已經使用 AutoLISP 撰寫出數以萬計的實用套裝系統和應用程式。對 AutoCAD 來說，這無疑是一筆巨大的資源和財富，也是其賴以發展的基礎之一。若輕易放棄，將會是一種巨大的資源浪費，甚至有可能影響到 AutoCAD 今後的發展。因此，Autodesk 公司就於 1998 年 3 月底宣佈推出新一代的 LISP 程式語言 — Visual LISP 以及 VBA，並將之加在 AutoCAD 2000 版上發表。

## 1-3 AutoLISP 與 Visual LISP 的關係

那麼究竟 AutoLISP 與 Visual LISP 差別在哪裡呢？Visual LISP 將具有以下優點：

### 1. 與 AutoLISP 完全相容

爲了充分利用 AutoLISP 語言的優勢和資源，新設計的 Visual LISP 將採用與 AutoLISP 完全相容的模式。換句話說，您原有使用 AutoLISP 所撰寫的程式並不需任何的修改或稍稍加以修改，就可在 Visual LISP 環境中執行。此舉將支援和維護了您在 AutoCAD 套裝應用程式上的投資。

那爲什麼 Visual LISP 可以與 AutoLISP 相容呢？這是因爲 Visual LISP 將它的程式語法設計成與 AutoLISP 相同，同時還新增了許多函數和系統變

數，這使得 AutoCAD 的應用程式開發工作變得更加容易。

只是，與 AutoLISP 不同的地方是：Visual LISP 採用了向標準 LISP 語言看齊的標準，以讓程式設計師能更加相容由 LISP 語言開發出其他人工智慧系統，這也是程式設計語言走向標準化的重要步驟。因此，Visual LISP 有些地方會表現得與 AutoLISP 不太一樣。

## 2. 物件導向編程技術

Visual LISP 與 Microsoft ActiveX、Object ARX 以及 Microsoft Visual Basic 一樣都是物件導向程式設計語言。經由 Autodesk 公司開發的 Visual LISPActiveX 介面，AutoCAD 的物件模型在交叉應用整合方面將具有更好的適應性，這將使使用者所開發的套裝應用程式不僅相容於 AutoCAD 軟體，而且與其他 ActiveX-Compliant 應用程式一樣，透過聯合資料庫就可以方便的套用，從而解決了多年帶來的應用程式智慧化、整合性的問題。Visual LISP 也允許使用者使用現成的 AutoLISP 套裝應用程式來叫用或轉換成一個 ADS 或 ARX 模組，因此，使用者可以從外部載入和更新 AutoCAD 軟體的多種版本。這種 AutoCAD 軟體的結構化合成模式，能為使用者在無需更新整個 AutoCAD 系統的情況下做系統更新。

## 3. 功能強大的整合開發環境

Visual LISP 也將是一個功能強大的整合開發環境。它整合了 AutoLISP 程式開發期間所需的幾大主要工具和功能，包括：

- Visual LISP 採用「Compile-during-Load」（隨載即編譯）技術，來達到與 AutoLISP 完全相容的境界。
- Visual LISP 採用可支援 AutoLISP 與 DCL 色彩編碼以及其 AutoLISP 語法的全螢幕本文編輯器。如此，將可方便使用者輸入 AutoLISP 原始程式，並透過色彩編碼對原始程式的不同部分加以顏色區分，以改善 AutoLISP 原始程式的可讀性。
- Visual LISP 支援多種檢查器。其中，語法檢查器可以用來檢查 AutoLISP 程式結構錯誤和內部函數中的變數錯誤。綜合檢查器可以提供對資料結構中變數和運算式值的瀏覽和編輯功能。
- 將 Visual LISP 的動態調整功能用於專門調整 AutoLISP 原始程式上極具靈活性。它可以在一個視窗單一執行 AutoLISP 的原始碼，而在 AutoCAD 視窗中同時顯示該代程式碼所執行效果。
- Visual LISP 先進的原始程式編譯器可以將 AutoLISP 的原始程式編譯成二進位檔案。如此，將可大大的改善了程式的執行速度與其安全

性。Visual LISP 的原始程式檔案 (.LSP) 或已編譯檔案 (.FAS) 都可以利用系統提供的 Application Wizard 軟體，將之包裝成一個單一的 ADS 或 ARX 模組。

因此，經由上述我們可以看出：Visual LISP 是一種已將 AutoLISP 語言的優點完全保留，缺點完全克服，並與最新的程式設計技術相結合的全新的整合開發系統，它已成為 AutoCAD 的下一代語言標準。Visual LISP 也將充分地利用現有的 AutoLISP 資源、極力的保護使用者投資，所以它的推出已引起廣大的 AutoCAD 使用者及專業開發技術人員的強烈興趣。

此外，我們也可以看出：為了使過去已開發的大量 AutoLISP 程式不至於毀棄而造成浪費，Autodesk 公司在 AutoLISP 的基礎上加入了 ActiveX 的支援，並加上了高效率的整合式開發環境，即 Visual LISP。Visual LISP 與 AutoLISP 相比，增加了許多如 vl-、vlx-、vla-、vlr- 等開頭，專門用來處理 ActiveX 物件的函數。ActiveX 的導入使呼叫 AutoCAD 圖素更簡易直接。但是在這世上，魚與熊掌總是難以兼得的，語法簡易就會導致程式碼變長，一些簡單的 LISP 函數就要經過幾行或十幾行或更多的程式來實現，這就增加了程式開發的時間。因此，靈活地運用 Visual LISP，就可以有效地結合 AutoLISP 與 ActiveX 物件程式來大量地縮短開發的時間。這也是我們使用 AutoLISP 的理由。

#### 1-4 VBA 概論

所謂 VBA 就是 Visual Basic for Application 的開頭字母大寫。由於它是一個物件導向程式設計的環境，提供我們類似 Visual Basic (VB) 語言所擁有的功能，所以後續潛力無窮。一般說來，VBA 和 VB 的主要差異類似 AutoLISP 與 LISP 間的差異，就是：VBA 可以執行並存取 AutoCAD 環境的所有資源。

此外，VBA 可以整合其它具有 VBA 能力的應用程式。這使得 AutoCAD 可以成為其它應用程式（如，Word 或 Excel）的 Automation 控制器。

從運作上來看，VBA 將經由 AutoCADActiveX Automation 介面將訊息傳送給 AutoCAD。而 AutoCADVBA 也允許 VBA 環境與 AutoCAD 同時執行並經由 ActiveXAutomation 介面提供 AutoCAD 的程式控制。如此，AutoCAD、ActiveX Automation 與 VBA 的結合方式，提供了功能極強大的介面，不僅可以操控 AutoCAD 的內部物件，更可以在其它的應用程式之間傳送或擷取資料。

AutoCADActiveX 提供了我們可以從 AutoCAD 內部或外部，以程式設計的方式來控管 AutoCAD 的機構。其方式是開放 AutoCAD 物件給「外面的世界」。並於開放這些物件後，就可以使用許多不同的程式語言（如，Word VBA、Excel VBA、Java..等）來存取這些物件。如圖 1-1 所示。



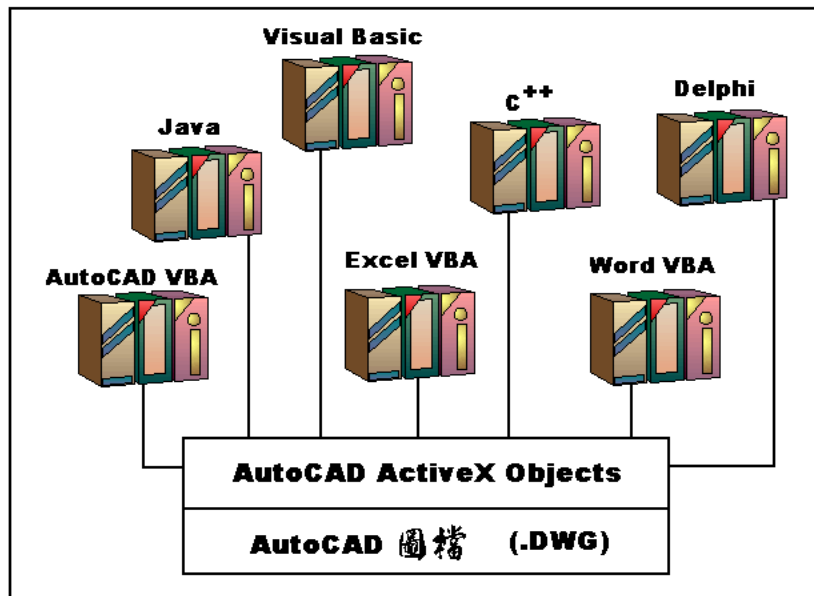


圖 1-1 AutoCAD 的語言結構與 ActiveX 的關係

特別說明：「ActiveX Objects」（ActiveX 物件模型）

物件(Objects)是 ActiveX 技術的基礎，每一個物件就代表 AutoCAD 的一個元素。AutoCAD 是以層結構來組織物件的，其頂層物件是 Application，下面是 Preferences 與 Document。然後，再透過對 Preferences 物件的操作，我們可以於執行 VBA 的「編輯視窗」功能後，在所出現的視窗中點取「工具」下拉式功能表下的「選項(O)」選項，來於其中設定所需的值。Document 物件將包含：ModelSpace、PaperSpace、Blocks、Layers、Plot 與 Utility 等物件。最底層的物件是點、線、圓、圖塊等 AutoCAD 能畫出的圖形。

瞭解了 VBA 與使用 ActiveX 物件間的關係後，我們再來說明它們各自的優點為何。如表 1-1 所示。

表 1-1 使用 VBA 與使用 ActiveX 的優點

使用 ActiveX 物件的優點	VBA 的優點
ActiveX 採用的是物件導向技術，程式的重用性高。	具學習親和力的 Visual Basic 程式
執行速度比 AutoLISP 與 ADS 程式更快。	VBA 同時也提供與其他具有 VBA 語言的應用程式整合。這意味著 AutoCAD 可以利用此功能來控制其他應用程式（如：Word、Excel、Chart 或 Graph 等）。
不必掌握 Object ARX 複雜的開發技術，非專業的技术人員也可以輕易的	VBA 與 AutoCAD 執行於同一處理程序，所以執行速度快。

使用 ActiveX 技術，使程式開發的工作更簡易。	
除了 VBA 以外，其他如 Delphi、VB、Java、C++ 等語言都可以透過 ActiveX 來存取 AutoCAD，使得 AutoCAD 的開發工具更多。	在 VBA 中設計交談視窗設計很快速。
物件模型的架構使 AutoCAD 與其他 Windows 應用程式間的資源分享更容易，ActiveX 更提供了在 AutoCAD 程式內外對其進行操控的方法。	專案可以獨立或內嵌在圖面中。這種功能在提供開發者分送應用程式上有很大的用處與靈活性。
	內含於 AutoCAD 中，無須另外購買，節省了投資成本。

那麼為什麼要使用 VBA 呢？最簡單的原因就是：Basic 要比 LISP、C/C++ 等易學，同時再也沒有比什麼易上手又功能強大的開發工具更重要。VBA 的優勢在於：

- 與 LISP 的情況類似，AutoCAD 2000 所包含的 VBA 可以與 AutoCAD 一起共用位址空間。而傳統的 VB 與 ActiveXAutomation 的介面則需要透過 IPC（內部處理連結）來驅動 AutoCAD，所以使用 AutoCAD 裡的 VBA 當然要比一般的 VB 語言具有明顯的性能優勢。
- 無需另外購買。VBA 是 AutoCAD2000 版以後的標準內建語言。
- 與 AutoLISP/Visual LISP 相比，VBA 在製作交談式視窗方面有獨到的優勢，所以它是取代 DCL 很好的工具。

本電子書將透過一些實用的 Visual LISP 與 VBA 範例，由淺入深，循序漸進的論述這兩種 CAD 開發工具的使用，以讓您可以在最短的時間內掌握這兩種 AutoCAD 開發語言最常用的功能與一些 CAD 系統設計常識。當然，在您學習之前，我們仍要提醒您：最好先熟悉本系列書籍裡，平面與立體畫圖方面的操作，因為擁有良好且正確的 AutoCAD 操作經驗，再加上自己的創造力，將可寫出比本電子書範例更好的實用程式。所以，當您奮力依練習實作完本電子書後，您就是 AutoCAD 的「**程式設計專家**」了。所有 AutoCAD 的內心世界，您都將是知音，同時我們也希望您「青出於藍而勝於藍」。

## 啓發性習題

## 一.選擇題(單複選混合)

- 1.( )AutoLISP 是開始自 AutoCAD 的哪一個版本？
  - (a)AutoCAD v2.17
  - (b)AutoCAD v2.6
  - (c)AutoCAD R12
  - (d)AutoCAD R14
- 2.( ) 下述何者是 AutoLISP/Visual LISP/VBA 等程式語言的作用？
  - (a)開發適用自己的 CAD 功能
  - (b)專門設計交談式視窗
  - (c)自動畫出符合設計條件的圖
  - (d)以上皆非
- 3.( ) 下述有關 DCL 的敘述，何者正確？
  - (a)可用來開發適用自己的 CAD 功能
  - (b)可被 VBA 取代
  - (c)專門設計交談式視窗
  - (d)以上皆是
- 4.( ) 下述有關 ActiveX 的優點敘述，何者為非？
  - (a)執行速度要比 AutoLISP 與 ADS 快
  - (b)採用物件導向技術
  - (c)專案可以獨立或內嵌在圖面中，對提供開發者分送應用程式上有很大的用處與靈活性
  - (d)以上皆非

## 二.實作問答題

1. 請試比較 AutoLISP 與 Visual LISP 的異同？
2. 請試比較 VBA 與 ActiveX 物件的各自優點？