

計劃名稱: 99-100年度教育部資訊軟體人才培育推廣計畫

跨校資源中心: 雲端計算與服務 (國立中山大學)

課程名稱: 虛擬化技術

## Part3-課程教材

教材名稱: 伺服器效能評估和精算虛擬化  
之合併率及TCO/ROI

# 大綱

1. 虛擬化之前IT效能分析
2. **Windows**系統的效能評估
3. **Linux**系統的效能評估
4. 精算虛擬化之合併率、**TCO** 以及**ROI**的意義
5. 伺服器整合的合併率計算
6. 桌上虛擬化的合併率計算
7. 伺服器整合**TCO/ROI** 計算
8. **TCO/ROI** 計算器詳細說明
9. 桌上虛擬化**TCO/ROI** 計算
10. 參考文獻

# 1. 虛擬化之前IT效能分析

- 企業考慮虛擬化的轉換，最關切的問題是使用虛擬化環境之後，會不會影響效能上的問題。由於每一種IT應用的情況不同，因此無法保證使用虛擬化之後會不會發生資源短缺的問題。所以在企業進行IT環境做虛擬化轉換之前，確實去了解每一個伺服器上服務的效能瓶頸是最重要的。
- 在一般IT系統來說，CPU、Memory、Storage、Network及其它I/O性能是我們關心的重點。在進行虛擬化的轉換前，使用專業的工具來分析其效能是最正確的作法。

# 虛擬化之前IT效能分析 (續)

- 在尚未了解正常使用中的伺服器效能之前，就逕行實施虛擬化是相當危險的過程，有可能會造成不可挽回的損失。
- 事實上虛擬化的過程是將實體機轉換成虛擬機，因此實體機的硬體效能若不夠強大，或是虛擬機的服務佔用太多系統資源，都會影響IT環境中的其它實體機/虛擬機。

## 2. Windows系統的效能評估

- 此時會決定一台實體機能使用多少個虛擬機的因素很多，但在將儲存的負載移到SAN上面之後，最重要的就是CPU、記憶體和網路卡。
- 由於記憶體可以用直接對應方式來安裝，而網路卡則是以服務的種類而定，因此最重要的就是CPU了。

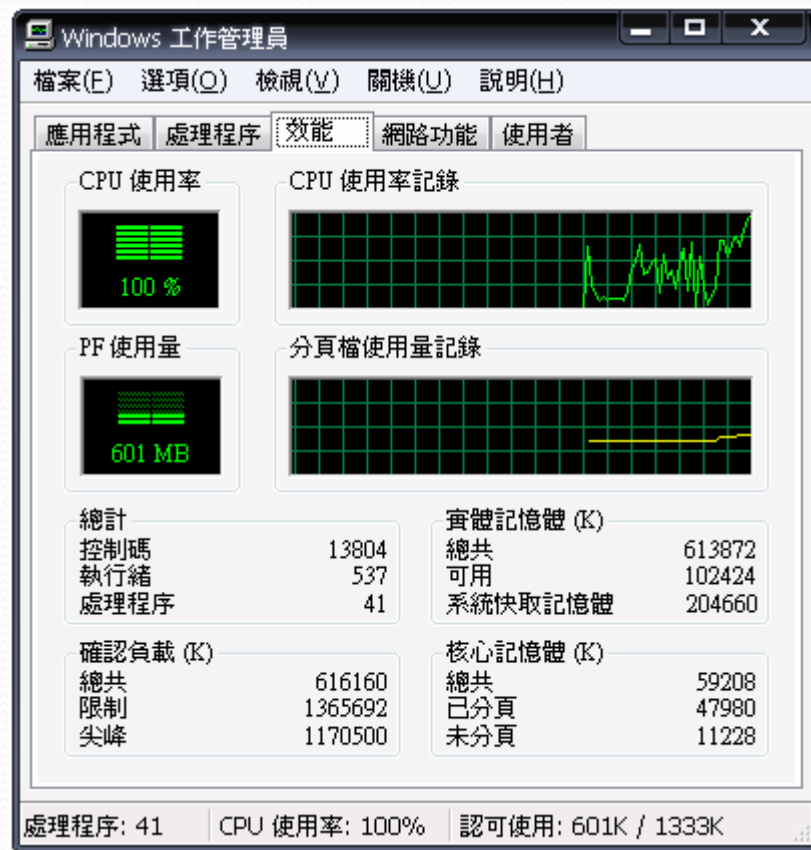


Fig. 2.1 一個滿載的伺服器是不適合移到虛擬機上的，尤其是CPU

# Windows系統的效能評估 (續)

- 在XP 和Windows Server 2003 中有一個「效能」的功能。這個功能可以將CPU 的使用率持續記錄下來，爲了了解整個伺服器的CPU 應用，我們可以針對某個伺服器（如網頁伺服器IIS）的CPU 使用率，做長時間的記錄。
- 在[開始]->[控制台]->[系統管理工具]->[效能]
- 我們可以自己增加一個CPU 的計數器，直接在空白處按下右鍵，並且選擇「New Log Settings..(新增記錄檔設定)」。

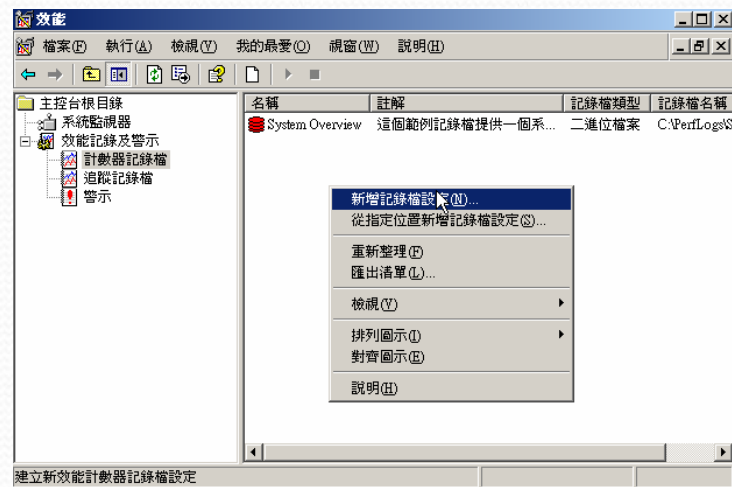


Fig. 2.2 選擇標準的系統部分

# Windows系統的效能評估 (續)

- 此時會出現要輸入name的視窗，輸入「CPU」，並且按下「OK(確定)」。



Fig. 2.3. 輸入「CPU」，這是以後計數器的名稱

- 出現新視窗，我們要先增加物件或計數器，我們以CPU為主，按下「Add Objects(新增物件)」。

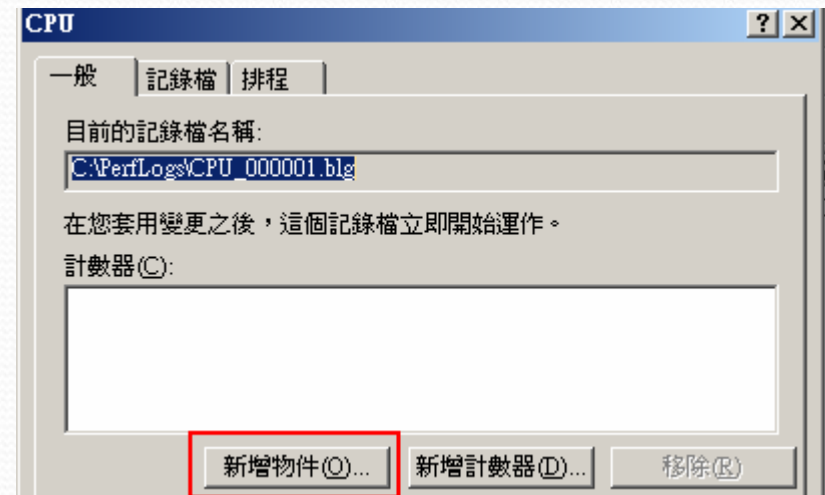


Fig. 2.4 需要增加物件，就是要系統監視的對象

# Windows系統的效能評估 (續)

- 從「Performance objects」中選擇「Process」，可以計算每一個CPU的負載，選擇完之後按下「Add」。
- 如果想看整機的CPU平均負載，也可以再選擇「Processor」選擇完之後按下「Add」。
- 在所有的項目選擇完之後按下「Close」。

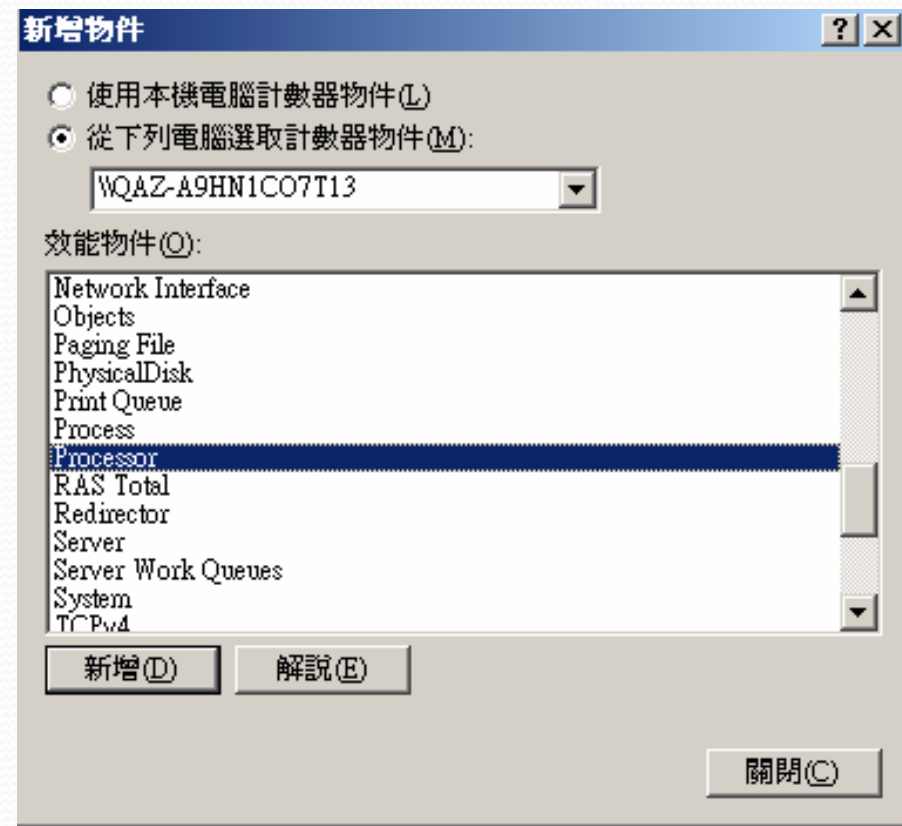


Fig. 2.5 CPU 物件



# Windows系統的效能評估 (續)

- 之後會回到「CPU」的視窗，選擇的取樣標的已在視窗中出現了。我們可以在這邊設定多久取樣一次，一般使用預設值的15秒即可。

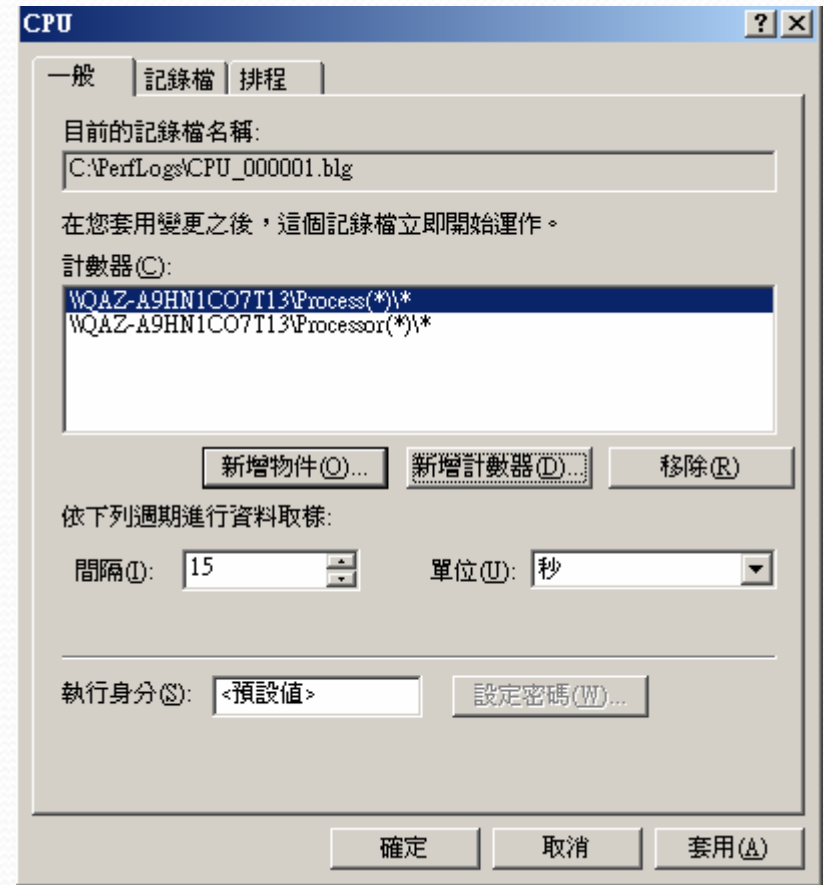


Fig. 2.6 必須設定取樣頻率，也就是時間的間隔

# Windows系統的效能評估 (續)

- 此時選擇「Log Files」的標籤，然後將「Log file type」選擇成「Text File(Comma delimited)」 「文字檔(以逗號分隔)」。
- 如果選擇一般的二進位檔案，只能用MMC 開啓，無法計算長時間的平均值。另外也要記住這個檔案的存放位置及檔名。之後按下「OK(確定)」。
- 如果出現找不到預設路徑，目前沒有這個目錄，請選擇「是」建立它。

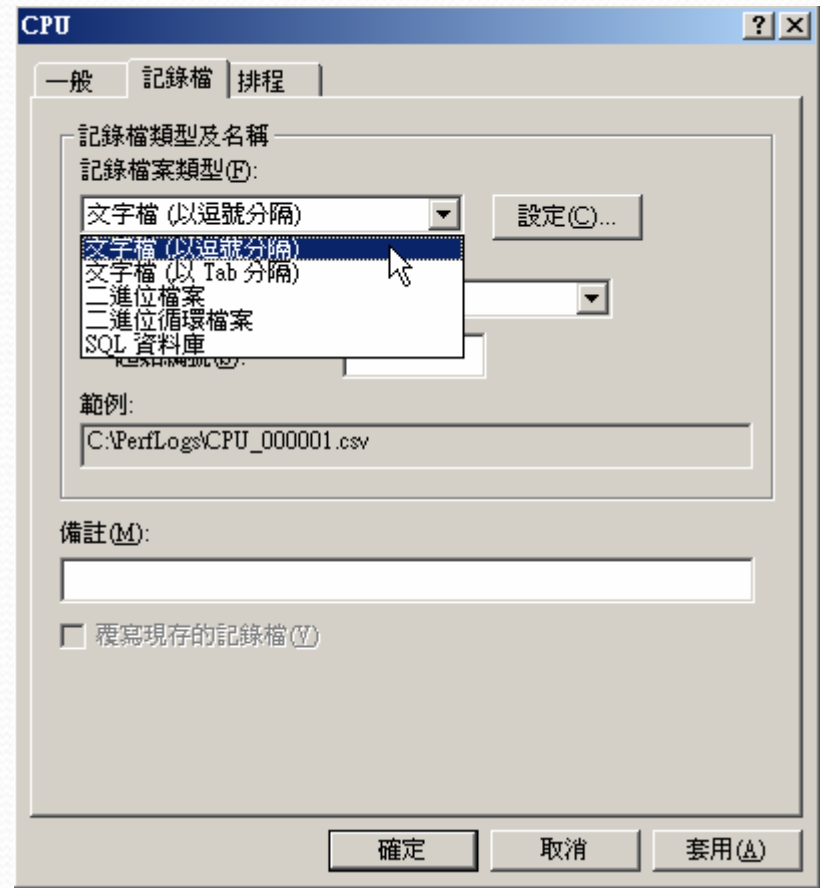


Fig. 2.7 在這個標籤的地方來選擇檔案格式

# Windows系統的效能評估 (續)

- 此時CPU的圖示由紅色變成綠色，表示記錄開始執行了。

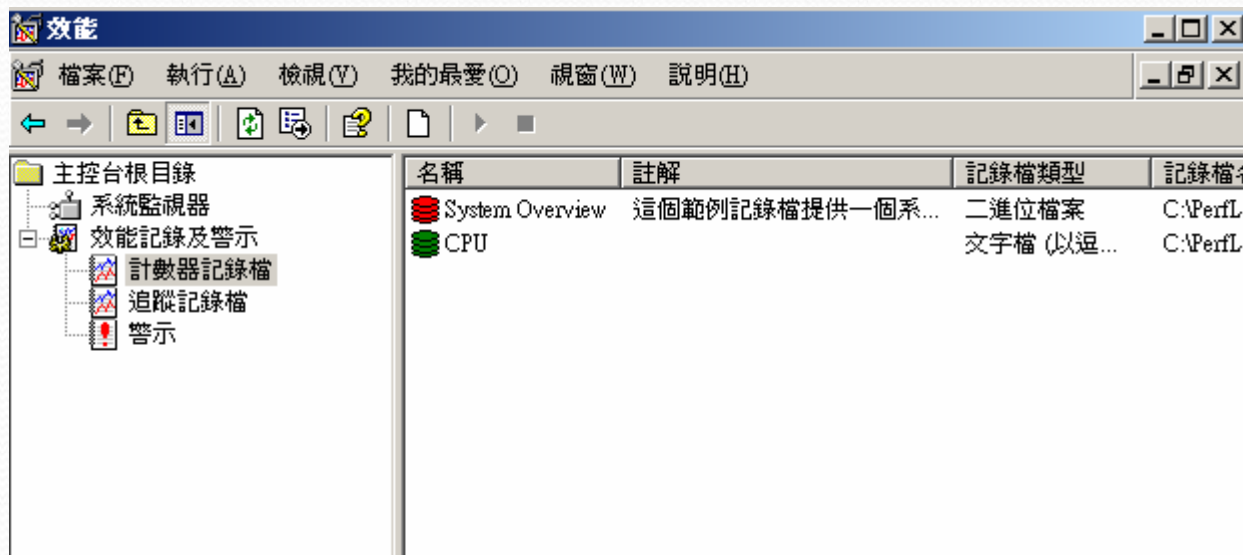


Fig. 2.8 開始記錄了

# Windows系統的效能評估 (續)

- 當取樣開始之後，系統即開始記錄，並且將資料寫入記錄檔案中。當記錄時間到達，我們使用Excel開啓這個記錄檔。

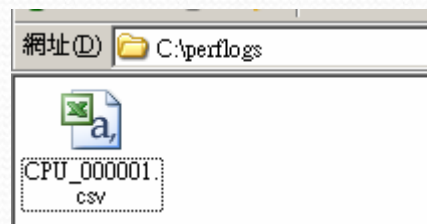
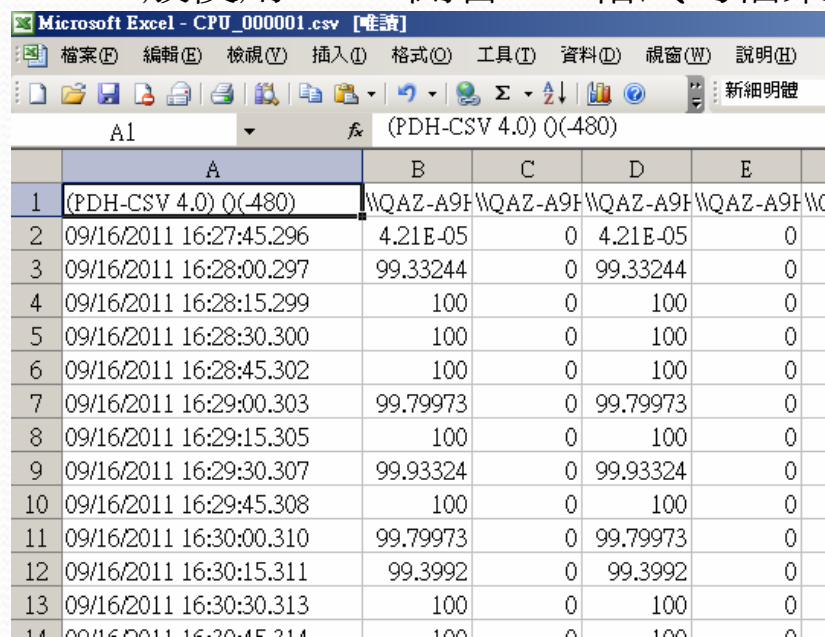


Fig. 2.9 一般使用Excel開啓Csv格式的檔案



	A	B	C	D	E
1	(PDH-CSV 4.0) 0(480)	\WQAZ-A9F\WQAZ-A9F\WQAZ-A9F\WQAZ-A9F\W			
2	09/16/2011 16:27:45.296	4.21E-05	0	4.21E-05	0
3	09/16/2011 16:28:00.297	99.33244	0	99.33244	0
4	09/16/2011 16:28:15.299	100	0	100	0
5	09/16/2011 16:28:30.300	100	0	100	0
6	09/16/2011 16:28:45.302	100	0	100	0
7	09/16/2011 16:29:00.303	99.79973	0	99.79973	0
8	09/16/2011 16:29:15.305	100	0	100	0
9	09/16/2011 16:29:30.307	99.93324	0	99.93324	0
10	09/16/2011 16:29:45.308	100	0	100	0
11	09/16/2011 16:30:00.310	99.79973	0	99.79973	0
12	09/16/2011 16:30:15.311	99.3992	0	99.3992	0
13	09/16/2011 16:30:30.313	100	0	100	0
14	09/16/2011 16:30:45.314	100	0	100	0

Fig. 2.10 記錄檔內容

- 此時會有很多筆記錄如圖所示，其中第一項就是CPU的負載值。

# Windows系統的效能評估 (續)

- 使用PAL Tool 來分析剛剛產生的Log 檔案。
- 你的視窗系統的記錄檔請複製到「C:\Program Files\PAL\PAL v2.0 Alpha 5」下，並在這個路徑下讀入。當然也可以設定你要分析的時間段。選擇完記錄檔之後就按「Next」。

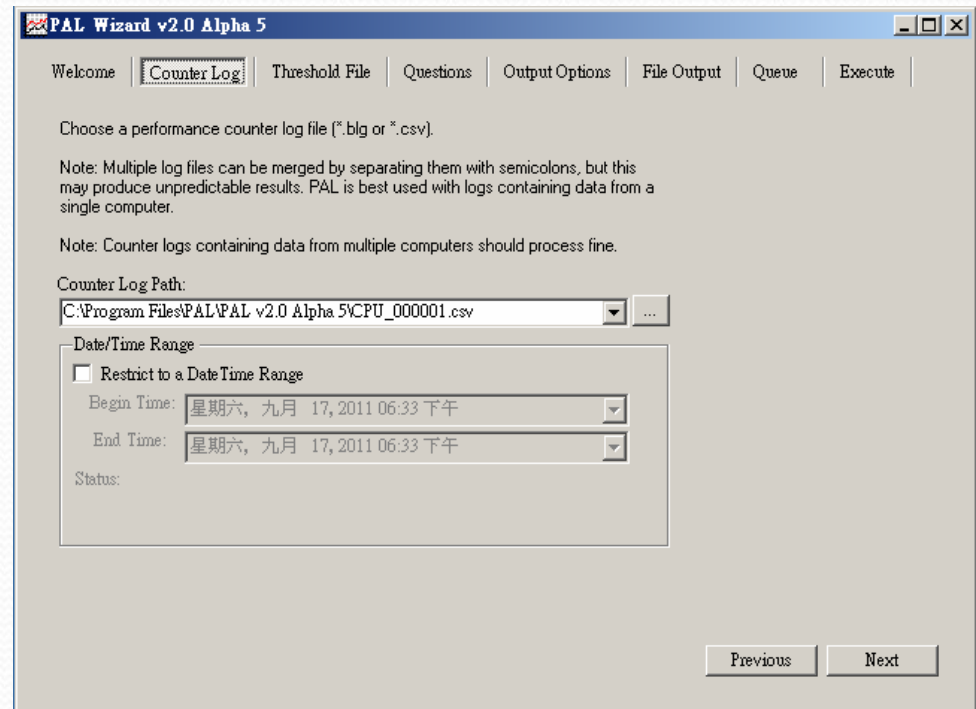


Fig. 2.11 選擇記錄檔位置

# Windows 系統的效能評估 (續)

- 我們可以從下拉選單中選擇你要評估的系統過濾項。由於PAL Tool 中已經有不同的設定檔，可以針對不同的Windows Server 2008 服務來設定分析的對象。
- 舉例來說，如果分析的機器是用來做為微軟的Hyper-V，就可以從這些預設的檔案中選擇。選擇完之後按下「Next」。

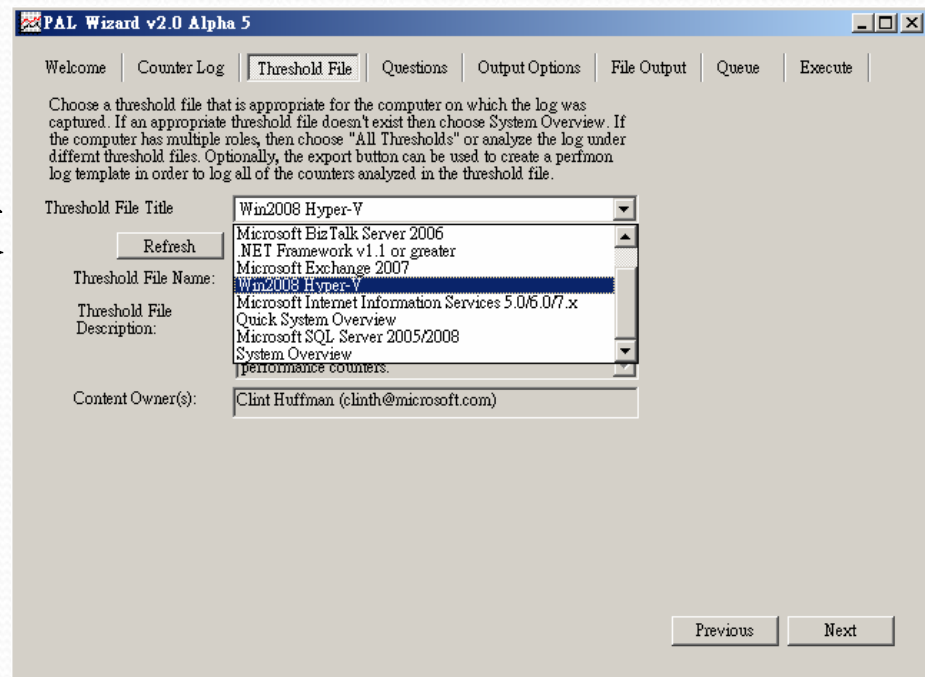


Fig. 2.12 選Hyper-V

# Windows 系統的效能評估 (續)

- 接下來是設定分析的選項等等，如使用CPU 的數量，總記憶體等等。之後按下「Next」。

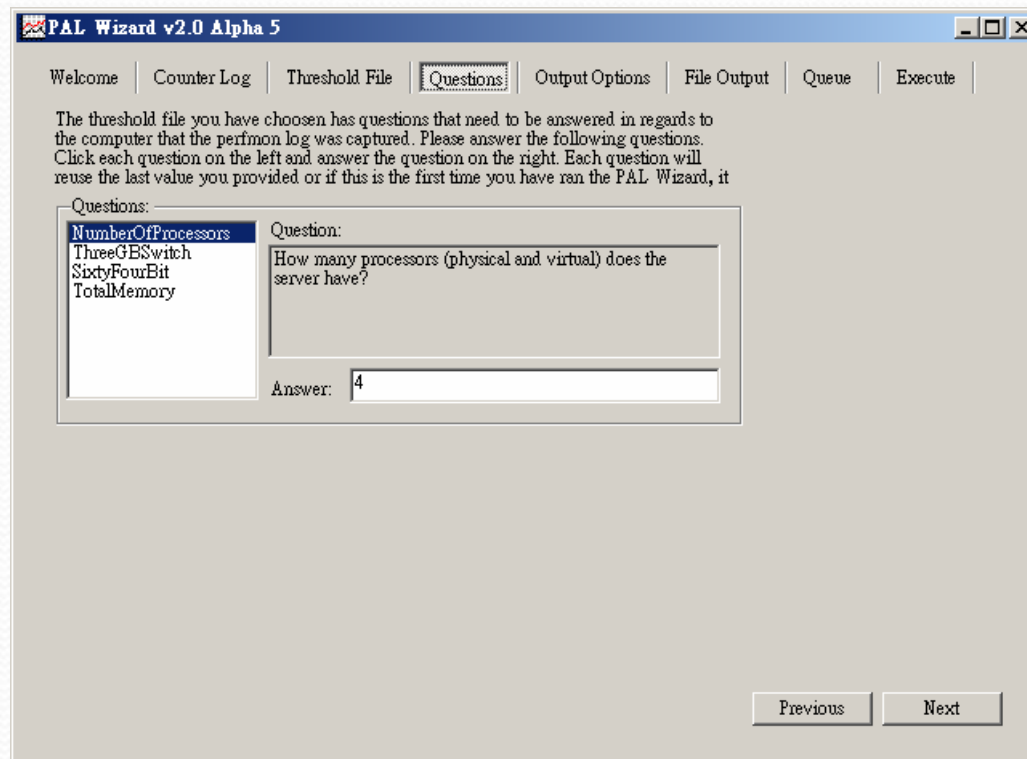


Fig. 2.13 選擇選項

# Windows 系統的效能評估 (續)

- 接下來是輸出的時間間隔，按下「Next」。

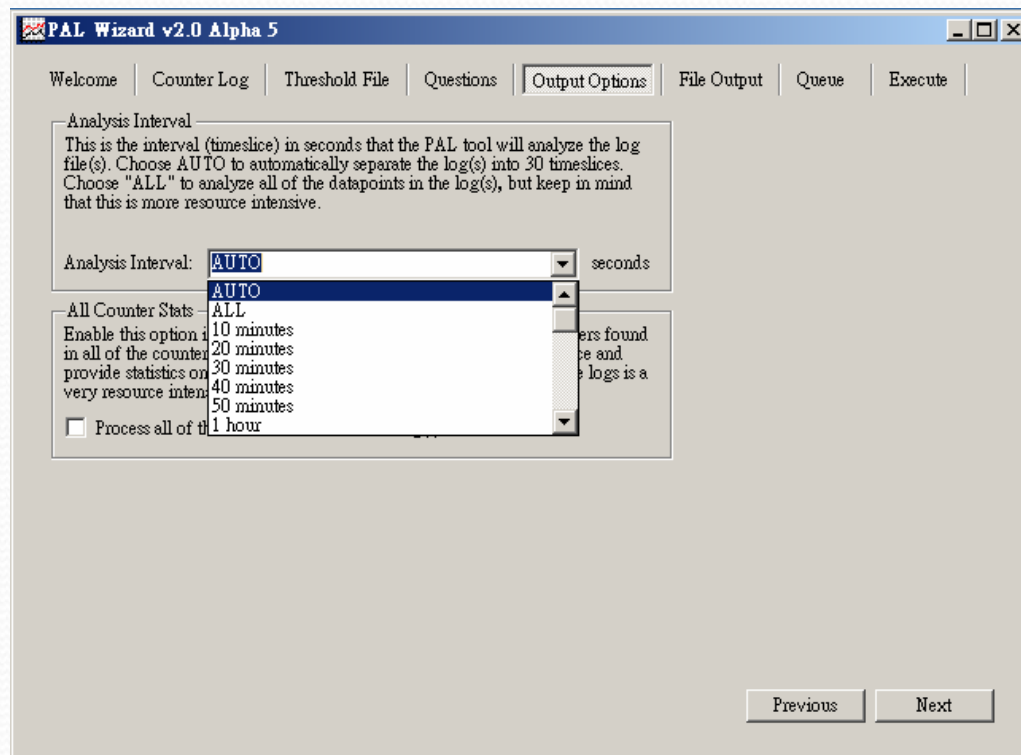


Fig. 2.14 選擇時間間隔



# Windows系統的效能評估 (續)

- 再來是輸出的檔案檔式，選HTML 較方便。

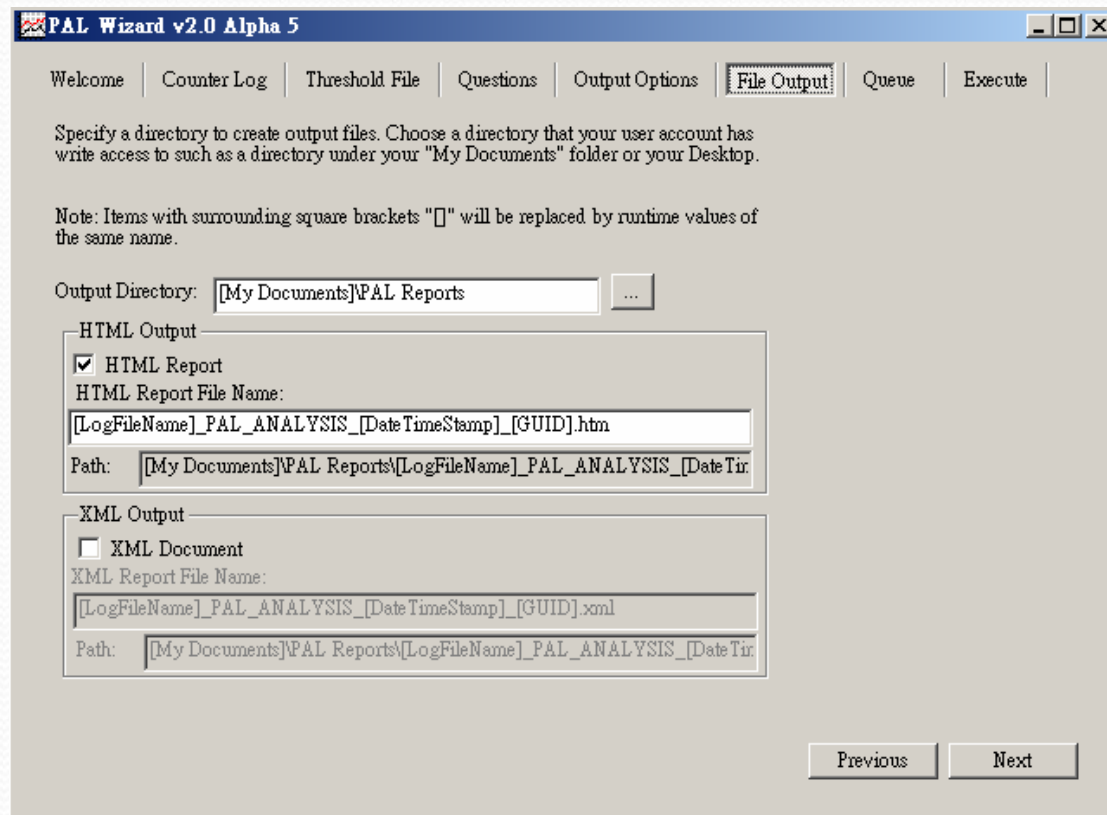


Fig. 2.15 輸入格式的選擇

# Windows 系統的效能評估 (續)

- 由於PAL Tool 就是一個vbscript，下一步是vbscript 的參數回顧部分。如果沒有要更改的地方，就直接按下「Next」。

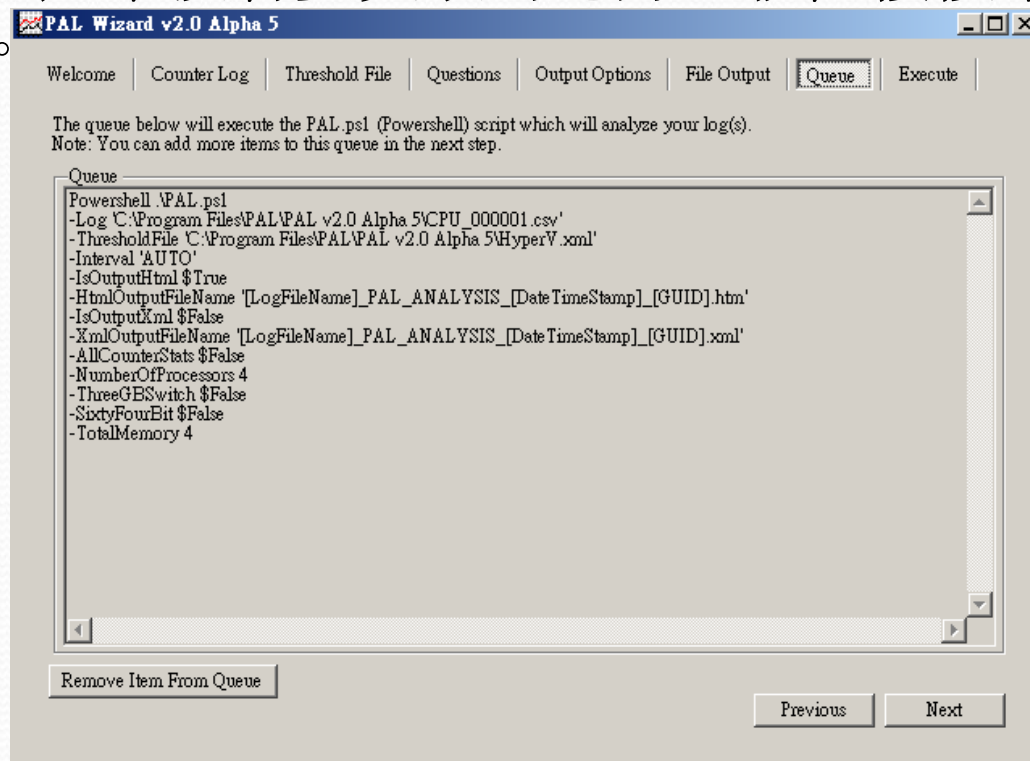


Fig. 2.16 這邊是輸出參數

# Windows系統的效能評估 (續)

- 最後一個畫面，就是直接執行這個vbscript。按下「Finish」就會直接開始執行分析並且輸出。

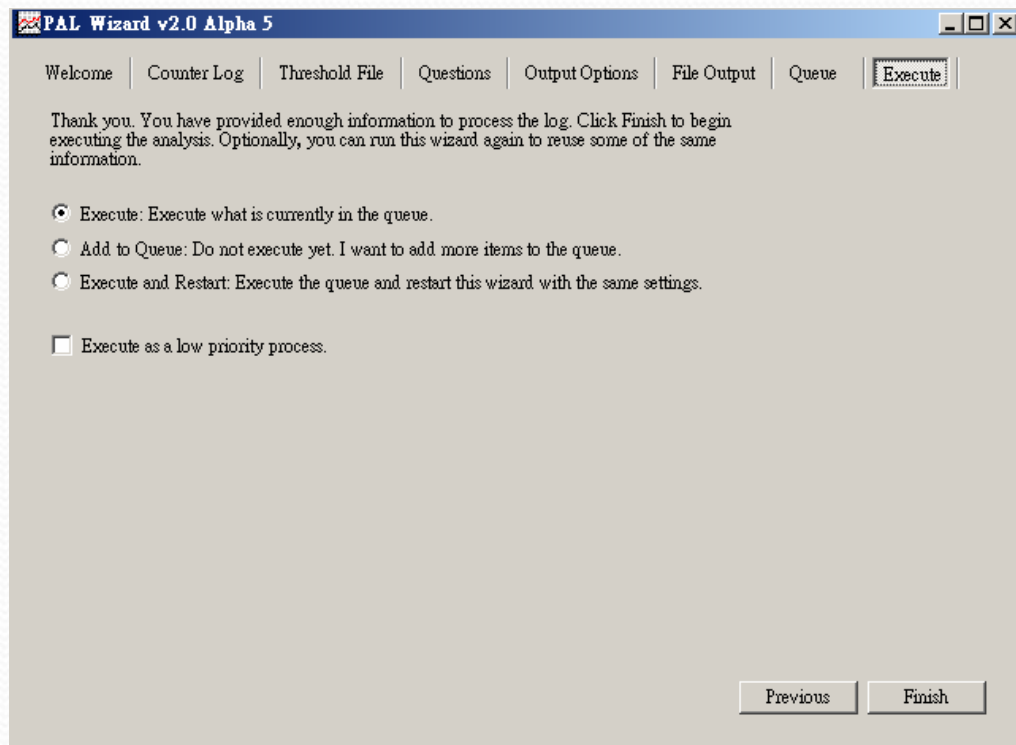
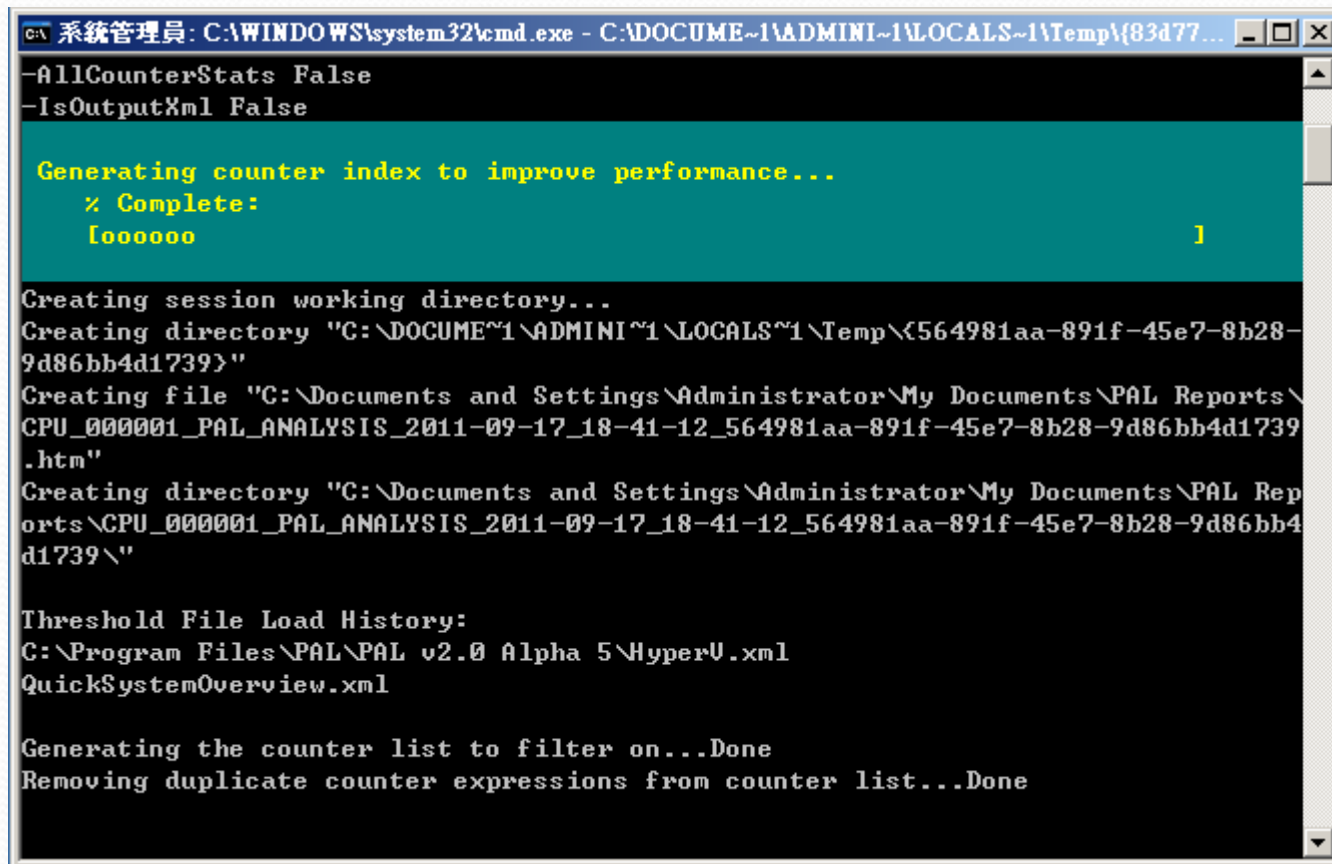


Fig. 2.17 按Finish就可以執行了

# Windows 系統的效能評估 (續)

- 開始進行分析，通常會花上數分鐘。



```
C:\WINDOWS\system32\cmd.exe - C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\{83d77...
-AllCounterStats False
-IsOutputXml False

Generating counter index to improve performance...
  % Complete:
  [ooooooooo]

Creating session working directory...
Creating directory "C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\{564981aa-891f-45e7-8b28-9d86bb4d1739}"
Creating file "C:\Documents and Settings\Administrator\My Documents\PAL Reports\CPU_000001_PAL_ANALYSIS_2011-09-17_18-41-12_564981aa-891f-45e7-8b28-9d86bb4d1739.htm"
Creating directory "C:\Documents and Settings\Administrator\My Documents\PAL Reports\CPU_000001_PAL_ANALYSIS_2011-09-17_18-41-12_564981aa-891f-45e7-8b28-9d86bb4d1739\"

Threshold File Load History:
C:\Program Files\PAL\PAL v2.0 Alpha 5\HyperU.xml
QuickSystemOverview.xml

Generating the counter list to filter on...Done
Removing duplicate counter expressions from counter list...Done
```

Fig. 2.18 分析過程

# Windows系統的效能評估 (續)

- 當出現如下圖的畫面時，表示分析完畢，並且輸出到指定的格式檔案了。

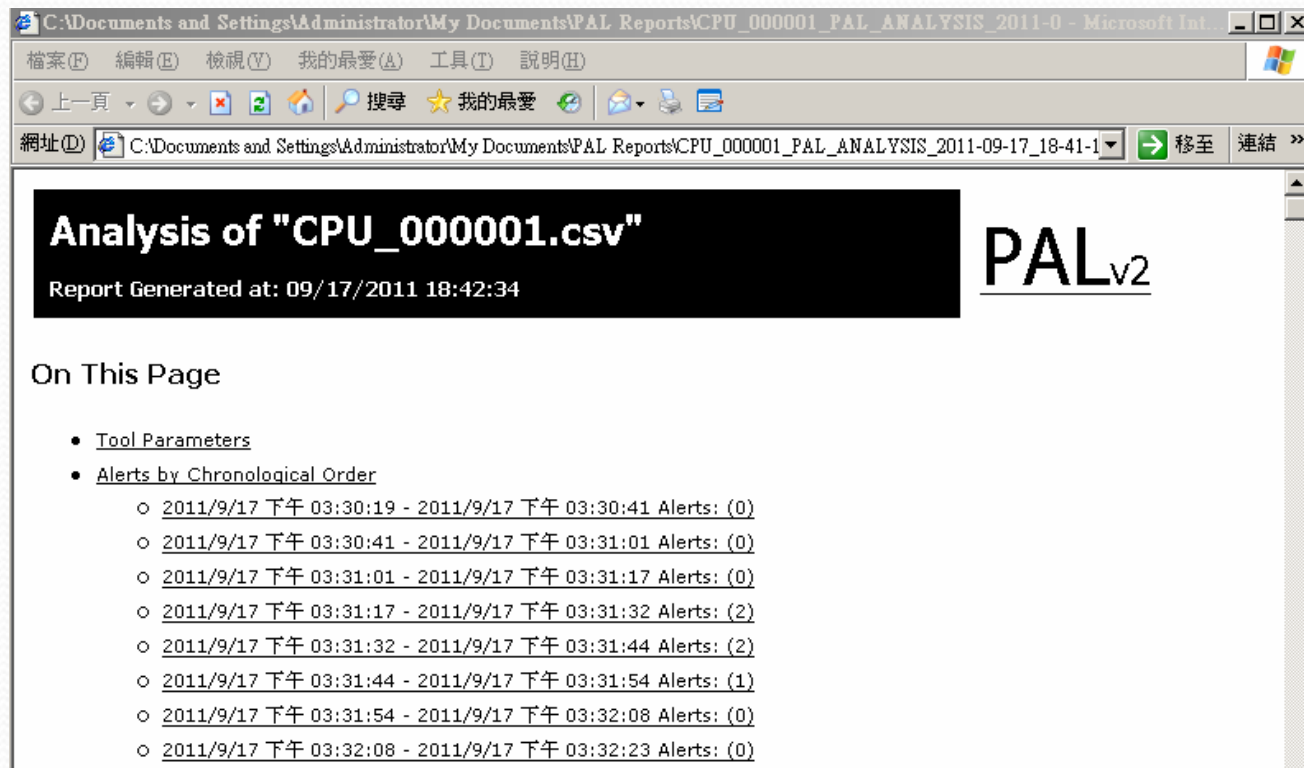
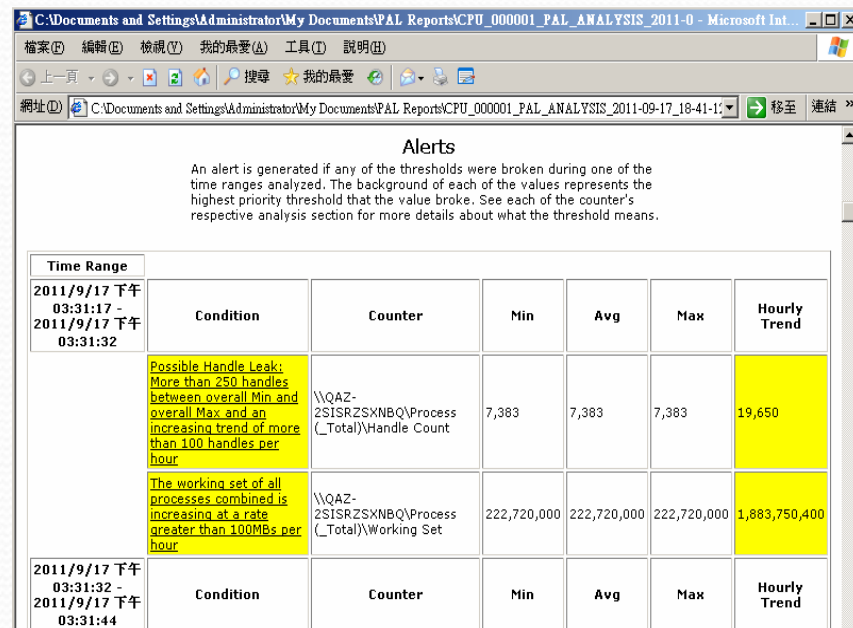


Fig. 2.19 html檔案完成了

# Windows系統的效能評估 (續)

- 輸出系統效能到了HTML 之後，就可以從表單或顏色來看出整個系統的瓶頸在哪邊了。
- 效能警示：在整張圖表中(右圖)，任何紅色的部分都為警示 (Alerts)，出現警示時表示該部分的效能出現85% 以上的佔有率或是出現錯誤。



The screenshot shows a web browser window displaying a performance report. The main heading is "Alerts". Below the heading is a paragraph explaining that an alert is generated if any thresholds were broken during the analyzed time ranges. The report contains two tables of alert data.

Time Range	Condition	Counter	Min	Avg	Max	Hourly Trend
2011/9/17 下午 03:31:17 - 2011/9/17 下午 03:31:32	Possible Handle Leak: More than 250 handles between overall Min and overall Max, and an increasing trend of more than 100 handles per hour	\\QA-ZSISRZSXNBQ\Process (_Total)\Handle Count	7,383	7,383	7,383	19,650
	The working set of all processes combined is increasing at a rate greater than 100MBs per hour	\\QA-ZSISRZSXNBQ\Process (_Total)\Working Set	222,720,000	222,720,000	222,720,000	1,883,750,400
2011/9/17 下午 03:31:32 - 2011/9/17 下午 03:31:44						

Fig. 2.20 效能警示

# Windows系統的效能評估 (續)

- 處理器部分 (Processor)：處理器最重要就是佔用率。我們可以先看到一個表格(下表)，顯示在這段時間內的CPU使用率，以虛擬機來說，長時間的平均值是最重要的，在表格之下也有文字及數字來顯示CPU的使用率。

## Overall Counter Instance Statistics

Overall statistics of each of the counter instances. Min, Avg, and Max are the minimum, average, and Maximum values in the entire log. Hourly Trend is the calculated hourly slope of the entire log. 10%, 20%, and 30% of Outliers Removed is the average of the values after the percentage of outliers furthest away from the average have been removed. This is to help determine if a small percentage of the values are extreme which can skew the average.

Condition	\Processor(*)\% Interrupt Time	Min	Avg	Max	Hourly Trend	Std Deviation	10% of Outliers Removed	20% of Outliers Removed	30% of Outliers Removed
OK	QAZ-2SISRZSXNBQ/0	0	2	5	1,080	2	1	1	1
OK	QAZ-2SISRZSXNBQ/1	0	0	1	0	0	0	0	0
OK	QAZ-2SISRZSXNBQ/2	0	0	1	360	0	0	0	0
OK	QAZ-2SISRZSXNBQ/3	0	0	1	360	1	0	0	0
OK	QAZ-2SISRZSXNBQ/_Total	0	1	2	360	1	0	0	0

Fig. 2.21 處理器效能

# 3. Linux系統的效能評估

- 在Linux 下要監測伺服器各項性能指標有太多選擇了，最常聽到就是**MRTG**。
- MRTG 全稱為多重路由流量表（Multi Router Traffic Graphic），利用圖表的方式顯示主機上一些指標的效能，可以透過Apache 或Tomcat，用網頁的圖表方式來查看。



# Linux系統的效能評估 (續)

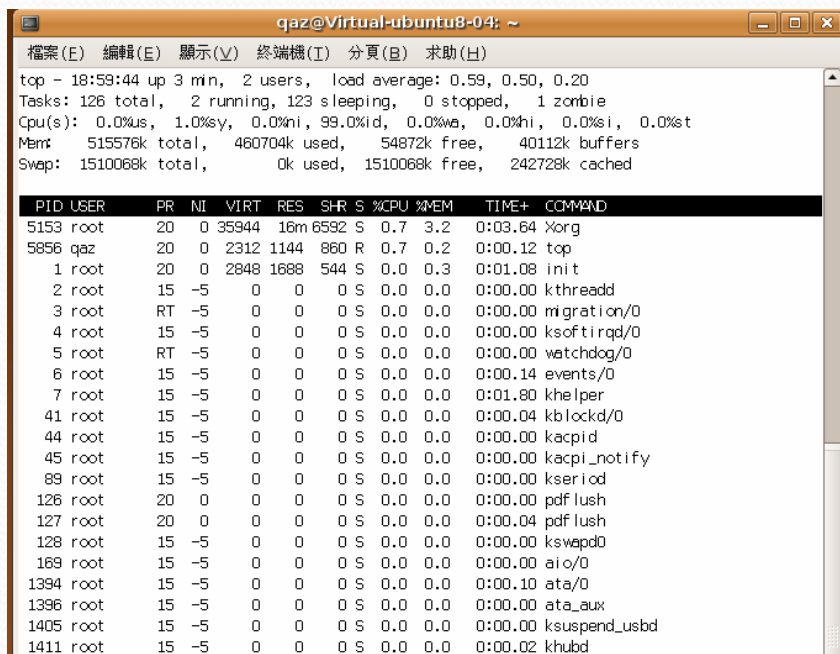
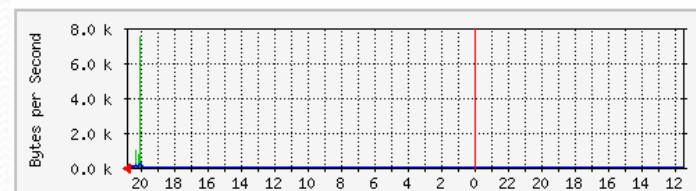


Fig. 3.1 一般在Linux下是使用top來查看性能，但無法進行長時間記錄

## `Daily' Graph (5 Minute Average)

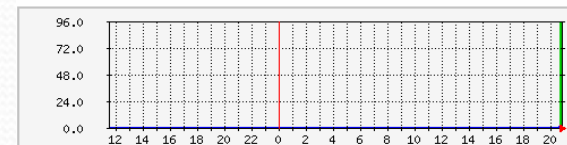


	Max	Average	Current
In	7499.0 B/s (0.6%)	880.0 B/s (0.1%)	6.0 B/s (0.0%)
Out	197.0 B/s (0.0%)	46.0 B/s (0.0%)	1.0 B/s (0.0%)

## CPU Loading

The statistics were last updated Saturday, 17 September 2011 at 20:50

## `Daily' Graph (5 Minute Average)



	Max	Average	Current
CPU sy;	96.0 %	25.0 %	3.0 %
CPU us;	1.0 %	1.0 %	1.0 %

Fig. 3.2 MRTG能做的事情非常多，如網路流量及CPU等

# Linux系統的效能評估 (續)

- 雖然MRTG 在一開始是用來查看網路流量的，但在使用不同的定義檔來產生數據之後，也可以顯示其它數據在MRTG 的圖表上，如CPU、RAM、DISK等方面。
- MRTG 原來的用處就是看網路的流量，因此在Linux 下，會以預設的網路卡（eth0）來作為顯示流量，也就是說，本小節所有的步驟都完成後，MRTG 就直接可以查看Linux 下第一片網路卡eth0 的流量了。

# Linux系統的效能評估 (續)

- 對在安裝MRTG 之前，你必須作好下面的準備。
  - 對Linux 有基本的認識，如安裝binary 套件、權限、檔案等觀念會使用vi 編輯器
  - Linux 已安裝LAMP 環境
  - Linux 的網路卡可正常使用，取得IP 可和外網連接
  - 能從網上或光碟機更新套件


# Linux系統的效能評估 (續)

- 當你準備好使用Linux 之後，就可以安裝MRTG 了，MRTG 是由下列的幾個套件組成的，必須設定下列的套件。
  - 安裝snmpd 服務以及snmp 工具列程式。
  - 設定snmpd 參數。
  - 測試snmpd 參數。
  - 安裝mrtg 套件。
  - 設定mrtg 的執行環境。
  - 建立mrtg 的設定檔。
  - 以設定檔中的參數來執行mrtg。
  - 建立Apache 下的mrtg 目錄。
  - 建立Apache 下的index.html 檔案
  - 從網頁查看mrtg 的圖表。

# Linux系統的效能評估 (續)


- 安裝snmpd 服務以及snmp 工具列程式:

1. 進入Linux 的命令提示列下，並且輸入「sudo apt-get install snmpd snmp」。
2. 安裝時系統要求輸入「y」，表示同意安裝，之後就會進行說明畫面，安裝完畢時回到命令列提示下。
3. 當安裝完畢時，snmpd 就會啓動。我們可以用「ps aux | grep snmpd」指令查看是否有啓動。



```
qaz@Virtual-ubuntu8-04: ~  
檔案(E) 編輯(E) 顯示(V) 終端機(I) 分頁(B) 求助(H)  
qaz@virtual-ubuntu8-04:~$ sudo apt-get install snmpd snmp  
sudo: unable to resolve host Virtual-ubuntu8-04  
[sudo] password for qaz:  
讀取套件清單中... 完成  
了解套件依存關係中  
Reading state information... 完成  
下列的套件曾因其他套件而被自動安裝，現在已不需要：  
libopenjdk10-dev libgnutlsxx13 libtasn1-3-dev libaudio-dev  
comerr-dev libgcrypt11-dev mesa-common-dev libkrb5-dev lib  
libgnutls-dev libqt3-headers libgl1-mesa-dev liblcm1-dev  
linux-headers-2.6.24-26-generic liblzo2-dev libkadm5 libm
```

Fig. 3.3. 在Linux 下安裝snmpd 及snmp 的套件



```
qaz@Virtual-ubuntu8-04: ~  
檔案(E) 編輯(E) 顯示(V) 終端機(I) 分頁(B) 求助(H)  
qaz@virtual-ubuntu8-04:~$ ps aux | grep snmpd  
snmp      6038  0.0  0.7  8768  3840 ?        S    19:02  
0:00 /usr/sbin/snmpd -Lsd -Lf /dev/null -u snmp -I -smux -p  
/var/run/snmpd.pid 127.0.0.1  
qaz      6101  0.0  0.1   3012   764 pts/0    R+   19:07  
0:00 grep snmpd  
qaz@virtual-ubuntu8-04:~$
```

Fig. 3.4 查看snmpd 是否啓動

# Linux系統的效能評估 (續)

- 設定snmpd 參數:

1. 例:在Ubuntu下，snmpd 的參設定檔放在 /etc/snmp/snmpd.conf 下，我們就直接編輯這個檔案，輸入「`sudo gedit /etc/snmp/snmpd.conf`」或是「`sudo vim /etc/snmp/snmpd.conf`」
2. 進入之後，找到如下頁圖中的地方，改成我們要的樣子。
3. 更改完之後，存檔離開，輸入「`sudo /etc/init.d/snmpd restart`」重新啓動snmpd 服務。

# Linux系統的效能評估 (續)

```
syslocation Unknown (configure /etc/snmp/snmpd.local.conf)
syscontact Root <root@localhost> (configure /etc/snmp/snmpd.local.conf)
```

Fig. 3.5 這是辯視資料，在MRTG的網頁上會顯示

```
syslocation Vaemon.com
syscontact vaemon
```

Fig. 3.6 改成你要的樣版

# Linux系統的效能評估 (續)

```
#      sec.name  source      community
com2sec paranoid default      public
#com2sec readonly default      public
#com2sec readwrite default      private
```

Fig. 3.7 找到這個地方

```
#      sec.name  source      community
#com2sec paranoid default      public
com2sec readonly default      public
#com2sec readwrite default      private
```

Fig. 3.8 將# 號拿掉

```
qaz@virtual-ubuntu8-04:~$ sudo /etc/init.d/snmpd restart
sudo: unable to resolve host Virtual-ubuntu8-04
[sudo] password for qaz:
Restarting network management services: snmpd.
```

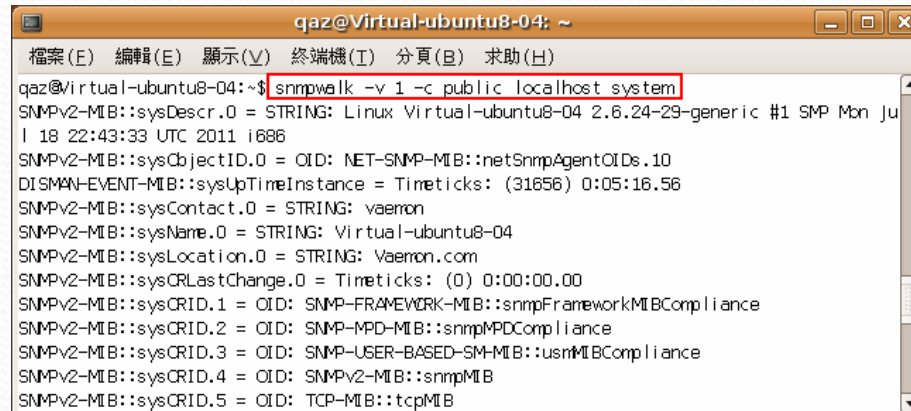
Fig. 3.9 重新啓動snmp



# Linux系統的效能評估 (續)

- 測試snmp 參數:

1. 輸入「snmpwalk -v 1 -c public localhost system」如果出現了一大堆參數，表示snmpd 設定已經成功了。
2. 如果出現timeout 或是no response 之類的訊息，表示snmpd 沒有安裝好。這時要檢查網路卡是否有啟動，檢查snmpd 是否在服務中啟動等等。一般來說，snmpd 不太會出問題，所以如果這個參數有問題，要多看看其它網路、IP、服務等問題。

A terminal window titled 'qaz@Virtual-ubuntu8-04: ~' showing the execution of the command 'snmpwalk -v 1 -c public localhost system'. The command is highlighted with a red box. The output displays various SNMPv2-MIB system information, including system description, contact, name, location, and various CRIDs.

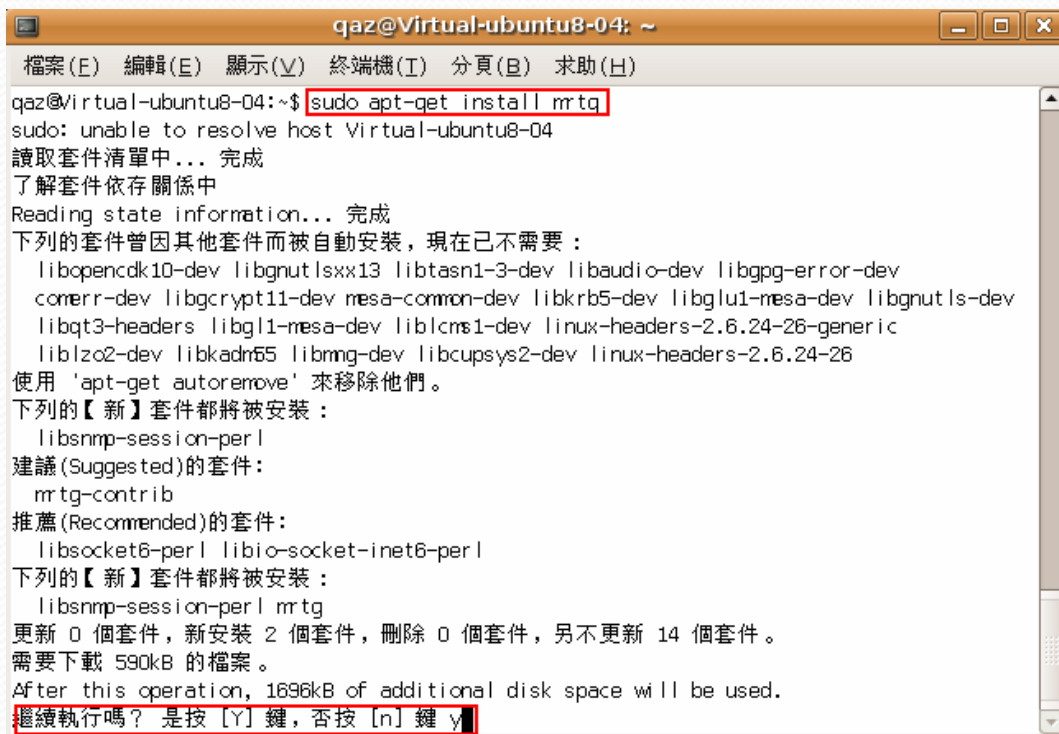
```
qaz@Virtual-ubuntu8-04: ~  
檔案(E) 編輯(E) 顯示(V) 終端機(T) 分頁(B) 求助(H)  
qaz@virtual-ubuntu8-04:~$ snmpwalk -v 1 -c public localhost system  
SNMPv2-MIB::sysDescr.0 = STRING: Linux Virtual-ubuntu8-04 2.6.24-29-generic #1 SMP Mon Ju  
l 18 22:43:33 UTC 2011 i686  
SNMPv2-MIB::sysObjectID.0 = OID: NET-SNMP-MIB::netSnmpAgentOIDs.10  
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (31656) 0:05:16.56  
SNMPv2-MIB::sysContact.0 = STRING: vaeron  
SNMPv2-MIB::sysName.0 = STRING: Virtual-ubuntu8-04  
SNMPv2-MIB::sysLocation.0 = STRING: Vaeron.com  
SNMPv2-MIB::sysCRLastChange.0 = Timeticks: (0) 0:00:00.00  
SNMPv2-MIB::sysCRID.1 = OID: SNMP-FRAMEWORK-MIB::snmpFrameworkMIBCompliance  
SNMPv2-MIB::sysCRID.2 = OID: SNMP-MPD-MIB::snmpMPDCompliance  
SNMPv2-MIB::sysCRID.3 = OID: SNMP-USER-BASED-SM-MIB::usnMIBCompliance  
SNMPv2-MIB::sysCRID.4 = OID: SNMPv2-MIB::snmpMIB  
SNMPv2-MIB::sysCRID.5 = OID: TCP-MIB::tcpMIB
```

Fig. 3.10 出現圖中的訊息即可

# Linux系統的效能評估 (續)

- 安裝mrtg 套件:

1. 輸入「sudo apt-get install mrtg」



```
qaz@Virtual-ubuntu8-04: ~  
檔案(E) 編輯(E) 顯示(V) 終端機(T) 分頁(B) 求助(H)  
qaz@Virtual-ubuntu8-04:~$ sudo apt-get install mrtg  
sudo: unable to resolve host Virtual-ubuntu8-04  
讀取套件清單中... 完成  
了解套件依存關係中  
Reading state information... 完成  
下列的套件曾因其他套件而被自動安裝，現在已不需要：  
  libpencdk10-dev libgnutlsxx13 libtasn1-3-dev libaudio-dev libgpg-error-dev  
  comerr-dev libcrypt11-dev mesa-common-dev libkrb5-dev libglu1-mesa-dev libgnutls-dev  
  libqt3-headers libgl1-mesa-dev liblcms1-dev linux-headers-2.6.24-26-generic  
  liblzo2-dev libkadm5 librng-dev libcupsys2-dev linux-headers-2.6.24-26  
使用 'apt-get autoremove' 來移除他們。  
下列的【新】套件都將被安裝：  
  libsnmp-session-perl  
建議 (Suggested) 的套件：  
  mrtg-contrib  
推薦 (Recommended) 的套件：  
  libsocket6-perl libio-socket-inet6-perl  
下列的【新】套件都將被安裝：  
  libsnmp-session-perl mrtg  
更新 0 個套件，新安裝 2 個套件，刪除 0 個套件，另不更新 14 個套件。  
需要下載 590kB 的檔案。  
After this operation, 1696kB of additional disk space will be used.  
繼續執行嗎？ 是按 [Y] 鍵，否按 [n] 鍵 y
```

Fig. 3.11 輸入安裝，會要求安裝套件，輸入「y」即可

# Linux系統的效能評估 (續)

- 安裝mrtg 套件:
  2. 在安裝完畢之後，系統會在Linux 下建立一個mrtg 的cronjob，每五分鐘會執行一次mrtg 工作來更新監視的項目，預設值就是eth0 網路卡的流量。這個步驟完成後，表示mrtg 監視安裝完畢了，但mrtg的監視結果還不知道，我們必須建立能從Apache 上觀察的mrtg 網頁和設定檔才行。

```
gaz@virtual-ubuntu8-04:~$ cat /etc/cron.d/mrtg
*/5 * * * * root    if [ -d /var/lock/mrtg ]; then if [ -x /u
sr/bin/mrtg ] && [ -r /etc/mrtg.cfg ]; then env LANG=C /usr/bin/m
rtg /etc/mrtg.cfg >> /var/log/mrtg/mrtg.log 2>&1; fi else mkdir /
var/lock/mrtg; fi
```

Fig. 3.12 系統會建立一個Linux 下的Cron Job 來五分鐘執行一次



Fig. 3.13 照著其步驟進行即可

# Linux系統的效能評估 (續)

- 設定mrtg的執行環境:
  1. 輸入「sudo su」，自此以後都以root的身份來執行指令
  2. 在Ubuntu下，打上「apt-get install apache2」先安裝Apache2，然後Apache2的文件目錄是安裝在/var/www中，因此我們要建立一個/var/www/mrtg。輸入「mkdir /var/www/mrtg」。

```
qaz@virtual-ubuntu8-04:~$ sudo su  
sudo: unable to resolve host Virtual-ubuntu8-04  
root@virtual-ubuntu8-04:/home/qaz#
```

Fig. 3.14 先以root的身份執行較方便

```
root@virtual-ubuntu8-04:/home/qaz# mkdir /var/www/mrtg  
root@virtual-ubuntu8-04:/home/qaz#
```

Fig. 3.15 建立一個目錄來存放mrtg的設定檔

# Linux系統的效能評估 (續)

- 設定mrtg 的執行環境:
  3. 建立一個mrtg.cfg 的設定檔。在Linux 中，mrtg 的執行都是讀取這個設定檔的內容，因此以後不管是CPU、RAM 或是其它的監視項目，都是以這個檔案為主。因此我們先建立，輸入「`cfgmaker public@localhost > /etc/mrtg.cfg`」

```
root@virtual-ubuntu8-04:/home/qaz# cfgmaker public@localhost > /etc/mrtg.cfg
root@virtual-ubuntu8-04:/home/qaz#
```

Fig. 3.16 建立設定檔

# Linux系統的效能評估 (續)

- 設定mrtg的執行環境:
  4. 執行mrtg這個程式，輸入「env LANG=C mrtg /etc/mrtg.cfg」，這個指令的目的是讓mrtg讀取/etc/mrtg.cfg這個檔案。

```
root@virtual-ubuntu8-04:/home/qaz# env LANG=C mrtg /etc/mrtg.cfg
2011-09-17 19:50:19, Rateup WARNING: /usr/bin/rateup Can't remove localhost_eth3.ol
d updating log file
root@virtual-ubuntu8-04:/home/qaz#
```

Fig. 3.17 執行mrtg程式

# Linux系統的效能評估 (續)

- 設定mrtg的執行環境:
  5. 執行之後會有很多警告訊息，多執行三、四次直到所有的警告訊息都沒有即可。

```
root@virtual-ubuntu8-04:/home/qaz# env LANG=C mrtg /etc/mrtg.cfg
2011-09-17 19:50:19, Rateup WARNING: /usr/bin/rateup Can't remove localhost_eth3.o
d updating log file
root@virtual-ubuntu8-04:/home/qaz# env LANG=C mrtg /etc/mrtg.cfg
root@virtual-ubuntu8-04:/home/qaz#
```

Fig.3.18 多執行幾次警告訊息會消失

# Linux系統的效能評估 (續)

- 設定mrtg的執行環境:
  6. 建立index.html檔案。也是輸入「indexmaker /etc/mrtg.cfg > /var/www/mrtg/index.html」

```
root@virtual-ubuntu8-04:/home/qaz# indexmaker /etc/mrtg.cfg > /var/www/mrtg/index.html
root@virtual-ubuntu8-04:/home/qaz#
```

Fig. 3.19 建立網頁的index.html 檔案



# Linux系統的效能評估 (續)

- 設定mrtg的執行環境:
  7. 到mrtg的目錄下看到已經建立的圖型及HTML檔案。

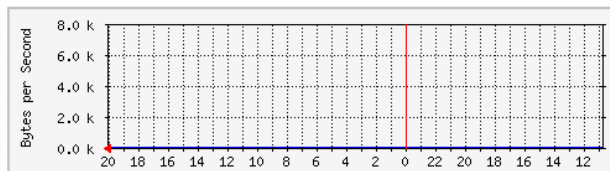
```
root@virtual-ubuntu8-04:/home/qaz# ls -l /var/www/mrtg/
總計 136
-rw-r--r-- 1 root root 2356 2011-09-17 19:54 index.html
-rw-r--r-- 1 root root 1511 2011-09-17 19:55 localhost_eth3-day.png
-rw-r--r-- 1 root root 6437 2011-09-17 19:55 localhost_eth3.html
-rw-r--r-- 1 root root 48209 2011-09-17 19:55 localhost_eth3.log
-rw-r--r-- 1 root root 1396 2011-09-17 19:50 localhost_eth3-month.png
-rw-r--r-- 1 root root 48209 2011-09-17 19:52 localhost_eth3.old
-rw-r--r-- 1 root root 1469 2011-09-17 19:50 localhost_eth3-week.png
-rw-r--r-- 1 root root 1815 2011-09-17 19:50 localhost_eth3-year.png
-rw-r--r-- 1 root root 538 2011-09-17 19:50 mrtg-l.png
-rw-r--r-- 1 root root 414 2011-09-17 19:50 mrtg-m.png
-rw-r--r-- 1 root root 1759 2011-09-17 19:50 mrtg-r.png
root@virtual-ubuntu8-04:/home/qaz#
```

Fig. 3.20 可以看到文字和圖檔都已經建立了

# Linux系統的效能評估 (續)

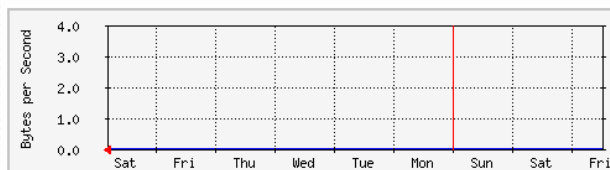
- 設定mrtg的執行環境:
  8. 我們能從電腦的瀏覽器進入，本例從「<http://192.168.0.103/mrtg>」進入。進入後只有一個圖表，我們可以按下去，可以看到更詳細的圖表。

~Daily' Graph (5 Minute Average)



	Max	Average	Current
In	7499.0 B/s (0.6%)	3896.0 B/s (0.3%)	7500.0 B/s (0.6%)
Out	197.0 B/s (0.0%)	159.0 B/s (0.0%)	198.0 B/s (0.0%)

~Weekly' Graph (30 Minute Average)



	Max	Average	Current
In	0.0 B/s (0.0%)	0.0 B/s (0.0%)	0.0 B/s (0.0%)
Out	0.0 B/s (0.0%)	0.0 B/s (0.0%)	0.0 B/s (0.0%)

~Monthly' Graph (2 Hour Average)

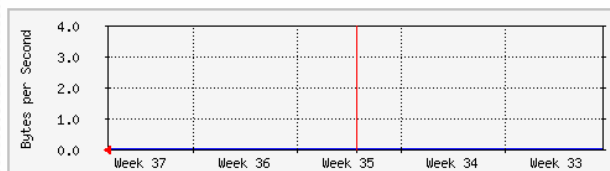


Fig. 3.21 會有不同時段的圖檔，對於長時間的資源使用率掌握很有幫助

# Linux系統的效能評估 (續)

- MRTG 安裝起來之後，Linux 已經具有監視eth0 的功能了。當然要將Linux主機轉換到虛擬機之前，你還得了解最重要的CPU 等效率，在P2V之後才知道要如何增強這些效能。
- 要將MRTG 拿來評估CPU 效能，下面是必須執行的步驟：
  - 安裝sysstat CPU 監測程式。
  - 建立CPU 監測執行指令檔。
  - 更改指令檔為可執行權限。
  - 更改mrtg.cfg 執行環境參數檔。
  - 以設定檔中的參數來執行mrtg。
  - 建立網頁index.html。
  - 如果需要，可以安裝燒機程式來燒CPU。
  - 從網頁查看mrtg 的圖表。

# Linux系統的效能評估 (續)

- 安裝sysstat CPU 監測程式:
  1. 輸入「`sudo apt-get install sysstat`」
  2. 當安裝完畢，我們可以用下面的指令查看是否有安裝成功「`sar -u 2 5`」，表示每2 秒監視一次CPU 的使用率，監測5 次。

```
root@virtual-ubuntu8-04:/home/qaz# sar -u 2 5
Linux 2.6.24-29-generic (Virtual-ubuntu8-04) 2011年09月17日

20時09分10秒    CPU      %user   %nice   %system %iwait   %steal   %idle
20時09分12秒    all      0.50    0.00    2.00    0.00    0.00    97.50
20時09分14秒    all      0.00    0.00    0.99    0.00    0.00    99.01
20時09分16秒    all      0.00    0.00    0.50    0.00    0.00    99.50
20時09分18秒    all      0.00    0.00    0.50    0.00    0.00    99.50
20時09分20秒    all      0.00    0.00    1.00    0.00    0.00    99.00
Average:        all      0.10    0.00    1.00    0.00    0.00    98.90
root@virtual-ubuntu8-04:/home/qaz#
root@virtual-ubuntu8-04:/home/qaz#
```

Fig. 3.22 可以用這個程式來監視CPU 的使用率

# Linux系統的效能評估 (續)

- 建立CPU 監測指令檔

1. 輸入「`sudo mkdir /opt/mrtg`」，建立目錄。
2. 輸入「`vim /opt/mrtg/mrtg.cpu`」，編輯CPU 的指令檔。
3. 將下列的程式碼輸入到這個mrtg.cpu 檔案中，存檔離開Vim。

```
#!/bin/bash
cpuusr=`/usr/bin/sar -u 1 3 | grep Average | awk '{print $3}'`
cpusys=`/usr/bin/sar -u 1 3 | grep Average | awk '{print $5}'`
Uptime=`/usr/bin/uptime | awk '{print $3 " " $4 " " $5}'`
echo $cpuusr
echo $cpusys
echo $Uptime
hostname
```

# Linux系統的效能評估 (續)

- 將mrtg.cpu 設定為可執行檔
  1. 在建立完mrtg.cpu 之後，要將這個檔案設定成可執行。我們輸入「`chmod 755 /opt/mrtg/mrtg.cpu`」這個檔案變成綠色之後，表示可以執行了，或是可以查看其執行的權限是否有打開。

```
root@virtual-ubuntu8-04:/home/qaz# chmod 755 /opt/mrtg/mrtg.cpu
root@virtual-ubuntu8-04:/home/qaz# ls -l /opt/mrtg/
總計 8
-rwxr-xr-x 1 root root 242 2011-09-17 20:11 mrtg.cpu
-rw-r--r-- 1 root root 243 2011-09-17 20:11 mrtg.cpu~
root@virtual-ubuntu8-04:/home/qaz#
```

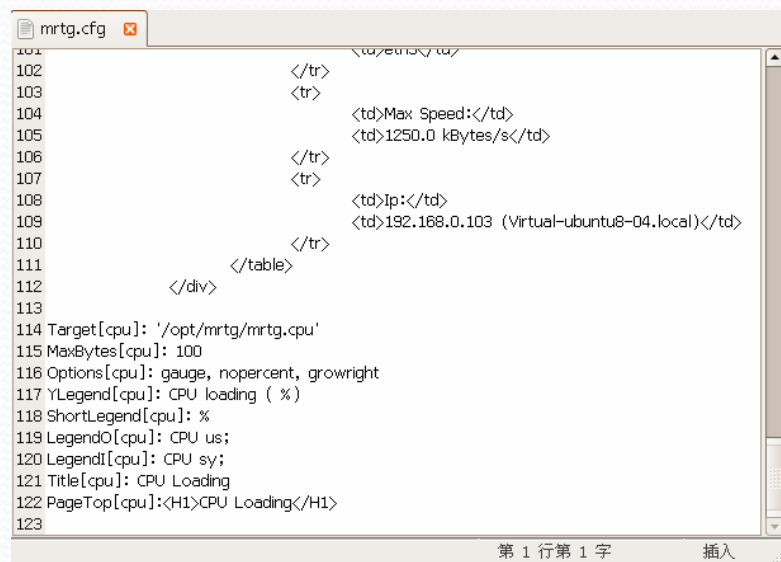
Fig. 3.23 將mrtg.cpu 變成可執行檔

# Linux系統的效能評估 (續)

- 修改mrtg的執行參數檔mrtg.cfg

1. 接下來我們要編修mrtg.cfg這個參數檔，只要輸入「`sudo gedit /etc/mrtg.cfg`」即可。將下面的程式碼貼到mrtg.cfg的最後面，注意千萬不要動到mrtg.cfg的其它地方！

```
Target[cpu]: `/opt/mrtg/mrtg.cpu`  
MaxBytes[cpu]: 100  
Options[cpu]: gauge, nopercent, growright  
YLegend[cpu]: CPU loading ( % )  
ShortLegend[cpu]: %  
LegendO[cpu]: CPU us;  
LegendI[cpu]: CPU sy;  
Title[cpu]: CPU Loading  
PageTop[cpu]:<H1>CPU Loading</H1>
```



```
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114 Target[cpu]: `/opt/mrtg/mrtg.cpu`  
115 MaxBytes[cpu]: 100  
116 Options[cpu]: gauge, nopercent, growright  
117 YLegend[cpu]: CPU loading ( % )  
118 ShortLegend[cpu]: %  
119 LegendO[cpu]: CPU us;  
120 LegendI[cpu]: CPU sy;  
121 Title[cpu]: CPU Loading  
122 PageTop[cpu]:<H1>CPU Loading</H1>  
123
```

Fig. 3.24 修改mrtg.cfg

# Linux系統的效能評估 (續)

- 設定及建立mrtg的執行環境
  1. 接下來就要執行mrtg這個程式了。輸入「env LANG=C mrtg /etc/mrtg.cfg」，執行之後會有很多警告訊息，多執行三、四次直到所有的警告訊息都沒有即可。

```
root@virtual-ubuntu8-04:/home/qaz# env LANG=C mrtg /etc/mrtg.cfg
awk: {print $3 * $4 * $5}
awk:      ^ invalid char '0' in expression
```

Fig. 3.25 重新設定mrtg的設定檔



# Linux系統的效能評估 (續)

- 設定及建立mrtg的執行環境
  2. 輸入「indexmaker /etc/mrtg.cfg > /var/www/mrtg/index.html」。我們可以到mrtg的目錄下看到已經建立的圖型及html檔案。

```
root@virtual-ubuntu8-04:/home/qaz# indexmaker /etc/mrtg.cfg > /var/www/mrtg/index.html
root@virtual-ubuntu8-04:/home/qaz#
```

```
root@virtual-ubuntu8-04:/home/qaz# ls -l /var/www/mrtg/
總計 256
-rw-r--r-- 1 root root 1292 2011-09-17 20:35 cpu-day.png
-rw-r--r-- 1 root root 5581 2011-09-17 20:35 cpu.html
-rw-r--r-- 1 root root 48199 2011-09-17 20:35 cpu.log
-rw-r--r-- 1 root root 1230 2011-09-17 20:14 cpu-month.png
-rw-r--r-- 1 root root 48199 2011-09-17 20:34 cpu.old
-rw-r--r-- 1 root root 1240 2011-09-17 20:14 cpu-week.png
-rw-r--r-- 1 root root 1580 2011-09-17 20:14 cpu-year.png
-rw-r--r-- 1 root root 2550 2011-09-17 20:34 index.html
-rw-r--r-- 1 root root 1598 2011-09-17 20:35 localhost_eth3-day.png
-rw-r--r-- 1 root root 6558 2011-09-17 20:35 localhost_eth3.html
-rw-r--r-- 1 root root 48258 2011-09-17 20:35 localhost_eth3.log
-rw-r--r-- 1 root root 1396 2011-09-17 19:50 localhost_eth3-month.png
-rw-r--r-- 1 root root 48256 2011-09-17 20:34 localhost_eth3.old
-rw-r--r-- 1 root root 1549 2011-09-17 20:20 localhost_eth3-week.png
-rw-r--r-- 1 root root 1815 2011-09-17 19:50 localhost_eth3-year.png
-rw-r--r-- 1 root root 538 2011-09-17 19:50 mrtg-l.png
-rw-r--r-- 1 root root 414 2011-09-17 19:50 mrtg-m.png
-rw-r--r-- 1 root root 1759 2011-09-17 19:50 mrtg-r.png
root@virtual-ubuntu8-04:/home/qaz#
```

Fig. 3.26 重新產生網頁檔案

Fig. 3.27 檢查有否新建立網頁檔

# Linux系統的效能評估 (續)

- 從瀏覽器中觀察mrtg的圖表:

1. 我們能從該電腦的瀏覽器進入，本例為「<http://192.168.0.1033/mrtg>」。就可以進入。進入後變成兩個圖表了，左邊的是原來的eth0，右邊的就是新的CPU。我們可以按下去，可以看到更詳細的圖表。

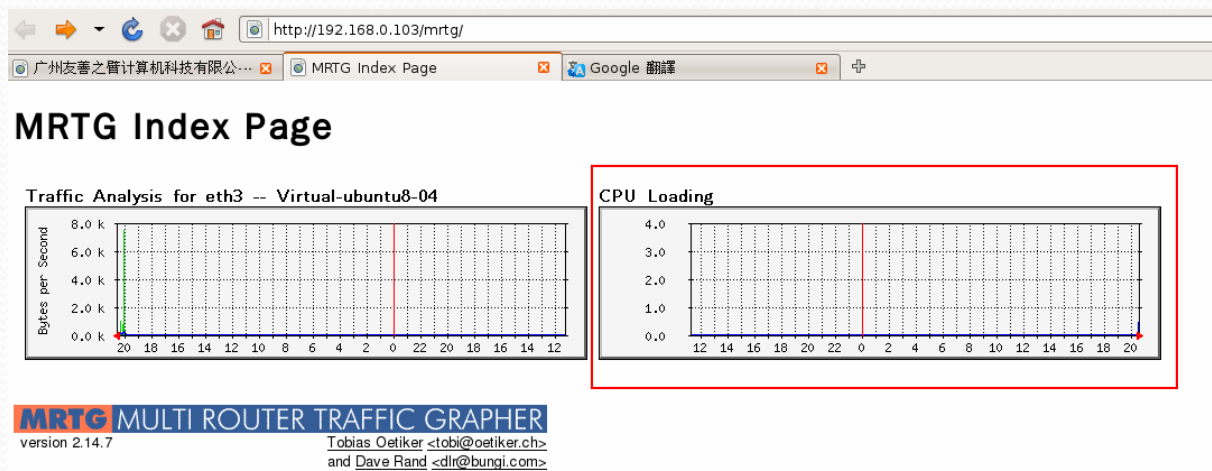


Fig. 3.28 CPU圖表 出現了

# Linux系統的效能評估 (續)

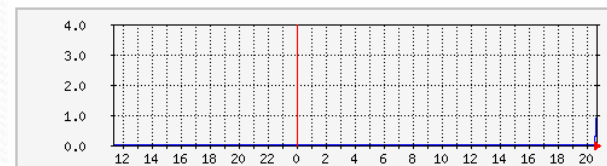
- 從瀏覽器中觀察mrtg的圖表:

2. 至此，我們的mrtg已經安裝好eth0和CPU了。如果我們想要監視其它的項目，只要更改「/etc/mrtg.cfg」，並且重新執行mrtg，並且重新產生index.html，就可以加入新的mrtg監視項目，只要將這邊mrtg的功能安裝好之後，以後就方便了。

## CPU Loading

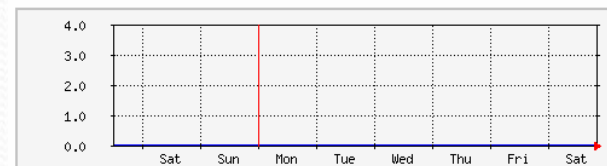
The statistics were last updated Saturday, 17 September 2011 at 20:40

### `Daily' Graph (5 Minute Average)



	Max	Average	Current
CPU sy;	0.0 %	0.0 %	0.0 %
CPU us;	1.0 %	1.0 %	1.0 %

### `Weekly' Graph (30 Minute Average)

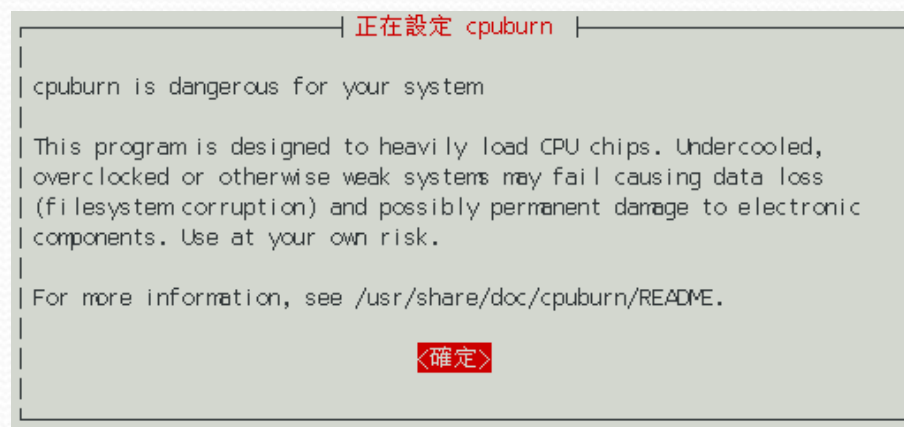


	Max	Average	Current
CPU sy;	0.0 %	0.0 %	0.0 %
CPU us;	0.0 %	0.0 %	0.0 %

Fig. 3.29 CPU的效能監視

# Linux系統的效能評估 (續)

- (額外)安裝CPU 燒機程式:
  1. 如果你想要安裝個CPU 燒機程式，可輸入「`sudo apt-get install cpuburn`」，就可以進行CPU 的燒機，只要安裝完之後輸入：`burnP5`，就可以讓你的CPU的loading 衝到頂了。



```
| 正在設定 cpuburn |  
| cpuburn is dangerous for your system  
|  
| This program is designed to heavily load CPU chips. Undercooled,  
| overclocked or otherwise weak systems may fail causing data loss  
| (filesystem corruption) and possibly permanent damage to electronic  
| components. Use at your own risk.  
|  
| For more information, see /usr/share/doc/cpuburn/README.  
|  
| <確定> |
```

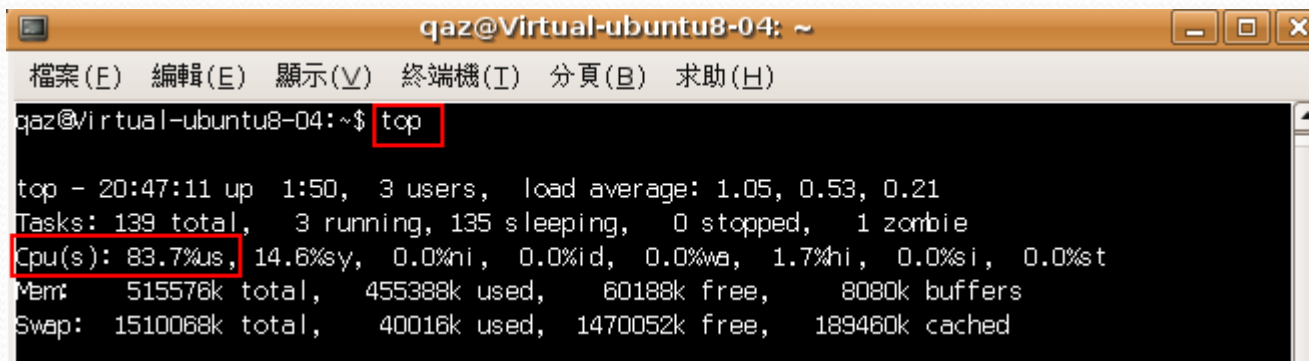
Fig. 3.30 會有警告畫面，因為有可能把系統弄當機

# Linux系統的效能評估 (續)

- (額外)安裝CPU 燒機程式:

```
root@virtual-ubuntu8-04:/home/qaz# burn  
burnBX  burnK6  burnK7  burnMMX  burnP5  burnP6  
root@virtual-ubuntu8-04:/home/qaz# burnP5
```

Fig. 3.31 執行burnP5



```
qaz@Virtual-ubuntu8-04: ~  
檔案(E) 編輯(E) 顯示(V) 終端機(T) 分頁(B) 求助(H)  
qaz@virtual-ubuntu8-04:~$ top  
top - 20:47:11 up 1:50, 3 users, load average: 1.05, 0.53, 0.21  
Tasks: 139 total, 3 running, 135 sleeping, 0 stopped, 1 zombie  
Cpu(s): 83.7%us, 14.6%sy, 0.0%ni, 0.0%id, 0.0%wa, 1.7%hi, 0.0%si, 0.0%st  
Mem: 515576k total, 455388k used, 60188k free, 8080k buffers  
Swap: 1510068k total, 40016k used, 1470052k free, 189460k cached
```

Fig. 3.32 CPU 飆高到80~99%

# 4. 精算虛擬化之合併率、TCO 以及ROI的意義

- 每一個企業在思考虛擬化時都會產生幾個問題，
  - 第一就是「一台實體機能裝幾台虛擬機？」
  - 第二就是「虛擬化能幫我省多少錢？」
  - 第三就是「需要再花什麼錢？」
  - 第四就是「還需要什麼其它的設備？」
- 這些問題的答案會因不同條件而異：
  - 不同的需求
  - 不同行業別
  - 不同企業規模
  - 不同虛擬機的合併率
- 實際上虛擬化已經進行許多年，業界也有了一致的標準，利用這些標準做爲例子，可以對企業進行虛擬化的評估。

# 4. 精算虛擬化之合併率、TCO 以及ROI的意義 (續)

- 在虛擬化的過程中，「一台實體機上能擺進幾個虛擬機」是必須經過規劃的。這個規劃的過程稱之為**隨需配置 (Provisioning)**。隨需配置的過程繁雜，通常會花上很長時間來做記錄並加以分析，以便計算出所謂的
  - 合併率 Consolidation Rate
  - 整體擁有成本 (Total Cost of Ownership, TCO)
  - 投資報酬率 (Return of Investment, ROI)
- 我們將逐一詳細介紹如何精算虛擬機的合併率、TCO 以及ROI，如此才能協助企業解決上述的問題。

# 5. 伺服器伺服器整合的合併率計算

- 一般而言，伺服器CPU執行服務時，使用率的目標不要超過90%。
- 執行服務時必須注意高峰值的發生，舉例來說，如果一台內部的目錄伺服器平均的使用率在5%，但是在上班期間大量使用者登錄後，CPU的使用率會衝到50%以上，並且會持續一段時間，那麼在做Provisioning時，就得為這台虛擬機預留一些較多的CPU資源，因此這台虛擬機所在的實體機就不要承載太多虛擬機，或是將這台虛擬機上的CPU資源多分配一些。



# 伺服器整合的合併率計算(續)

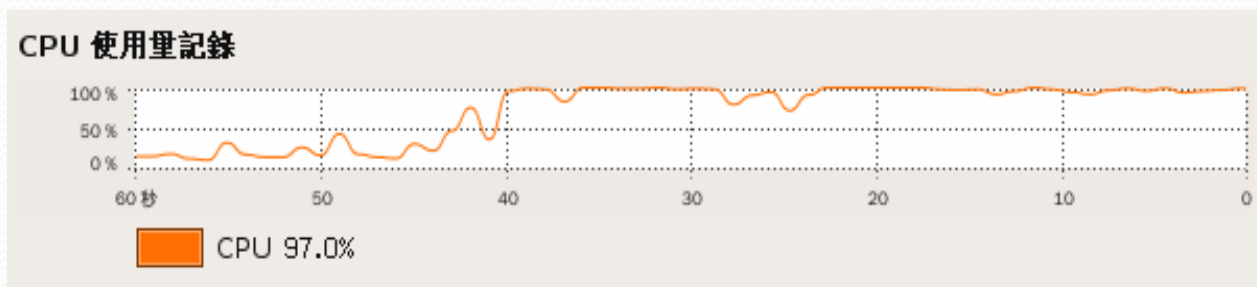


Fig. 5.1 高達90% 的CPU 使用率的服務會讓服務慢下來，因此不可超過

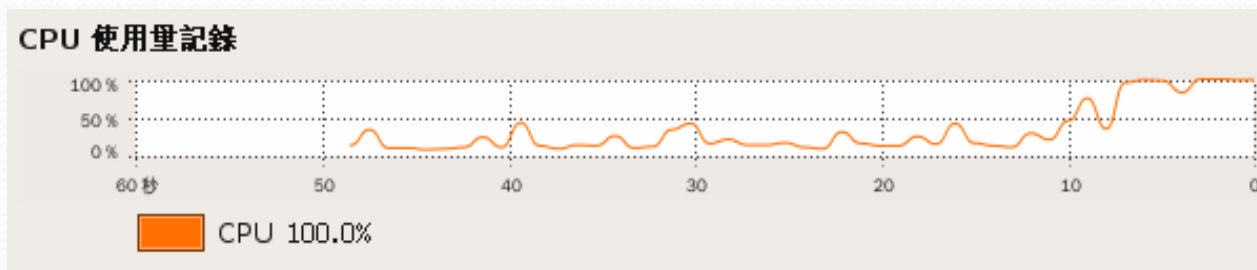


Fig. 5.2 CPU 使用率並不平均的服務就必須多注意

# 伺服器整合的合併率計算(續)

- SMP 及多核心的計算：每實體核心相當**3-5** 個vCPU
- 記憶體數量：**2GB** 到**4GB**/ 每實體CPU 核心
- 網路卡的分配原則：**5-7** 個虛擬機/ 每網路介面
- 實體伺服器設定：**多台實體機**或多片刀鋒伺服器
- 依上述原則去精算一個例子：
  - 企業中有**70** 台實體機伺服器
  - 其中有**5** 台是高負載的伺服器，如SQL 或是運算用
  - **65** 台是可以虛擬化的伺服器，如網頁或是其它服務
  - 有**45** 台虛擬機的負載較小，每台使用**1** 個vCPU
  - 有**20** 台虛擬機的負載較大，每台使用**2** 個vCPU

# 伺服器整合的合併率計算(續)

70台主機



Fig. 5.3 虛擬化之前的設定

5台實體伺服器

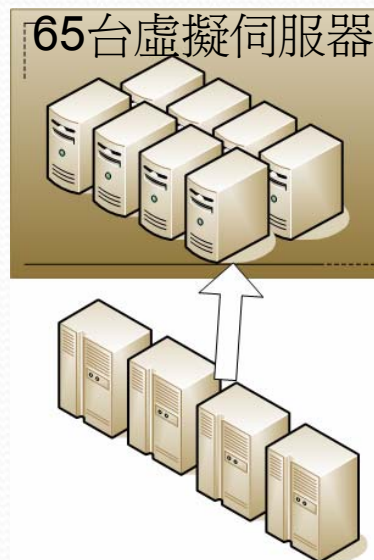


Fig. 5.4 預計虛擬化後的設定

# 伺服器整合的合併率計算(續)

- SMP 及多核心的計算：每實體核心相當**3-5** 個vCPU。從上述的CPU需求得知，我們需要的實體CPU 核心（pCPU）數量如下列計算公所示。
- 需求虛擬CPU數量為：  
 $(45 \text{ 台} \times 1 \text{ vCPU}) + (20 \text{ 台} \times 2 \text{ vCPU}) = 45 \text{ vCPU} + 40 \text{ vCPU} = \mathbf{85 \text{ vCPU}}$ 。
- 前面說過一個實體核心就算一個pCPU。如果不想實體機的負載太高（超過90%），就讓一個pCPU只需承載3個vCPU，我們所需要的實體CPU 核心數量就是  $85 \text{ vCPU} / 3 = \mathbf{29 \text{ 實體CPU 核心}}$ 。

# 伺服器整合的合併率計算(續)

- 如果我們希望些實體機的合併率再高一點，那麼就讓一個pCPU承載5個vCPU，我們所需要的實體CPU核心數量就是 $85 \text{ vCPU} / 5 = 17$  實體CPU 核心。
- 目前的伺服器市場使用的產品大多是4核心的CPU，因此在上面的例子中，需要的實體4核心CPU 數量為： $29 \text{ Core} / 4 = 8 \text{ CPU Packages}$  或是 $17 \text{ Core} / 4 = 5 \text{ CPU Packages}$ 。

# 伺服器整合的合併率計算(續)

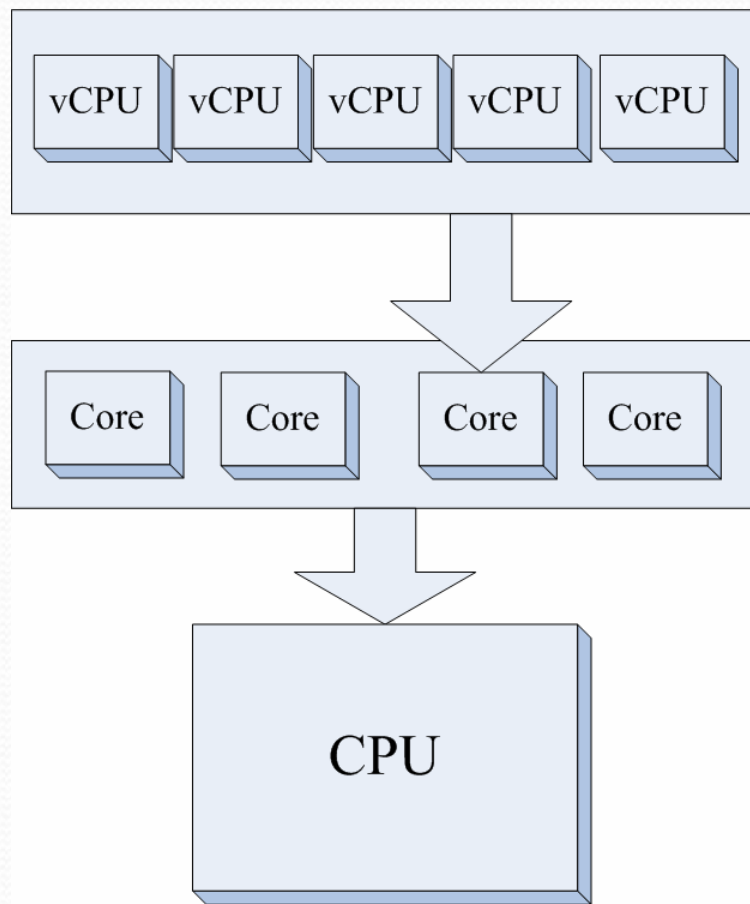


Fig. 5.5 需要的虛擬CPU、實體CPU 核心數和真正CPU 個數，以4 核心為例

# 伺服器整合的合併率計算(續)

- 記憶體數量：2GB 到4GB/ 每實體CPU 核心：
  - 伺服器的主機板通常能插上32GB或以上的記憶體，不過必須滿足我們的基本要求，就是每一實體CPU核心對應到2GB 到4GB 的記憶體。如此一來，如果有一台4 SMP 的主機板，上面最大的實體CPU核心數量可能會到 $4 \times 4 = 16$ 個Core，那麼所需的記憶體數量將會有32GB到64GB，因此一台主機可能就不夠，而必須有兩台主機來承載這29個Core衍生出來虛擬機了。
  - $(2\text{GB 或 } 4\text{GB}) \times (17\text{核心 或 } 29\text{核心}) = 34\text{GB 至 } 116\text{GB}$
  - 因此伺服器整合的記憶體數量是由**實體CPU核心**決定。

# 伺服器整合的合併率計算(續)

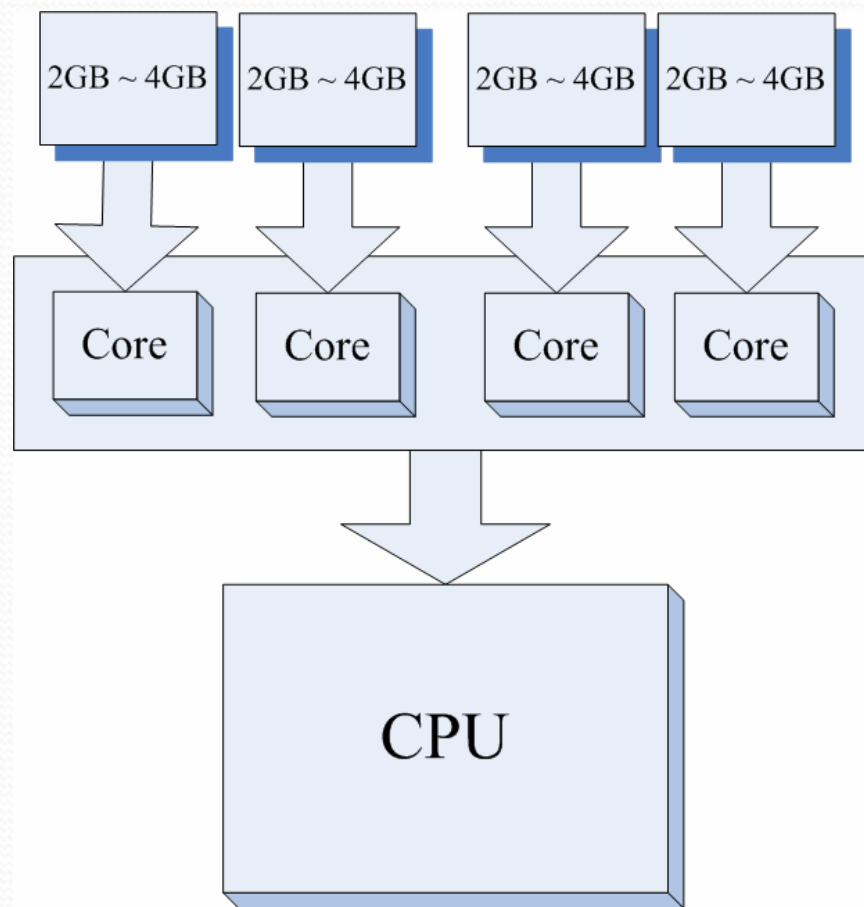


Fig. 5.6 每一實體核心對應2GB 到4GB 的記憶體



# 伺服器整合的合併率計算(續)

- 網路卡的分配原則：**5~7** 個虛擬機/ 每網路介面。
- 65 個虛擬機 / ( 一個網路介面為5~7 個虛擬機 ) = 10~13 個網路介面

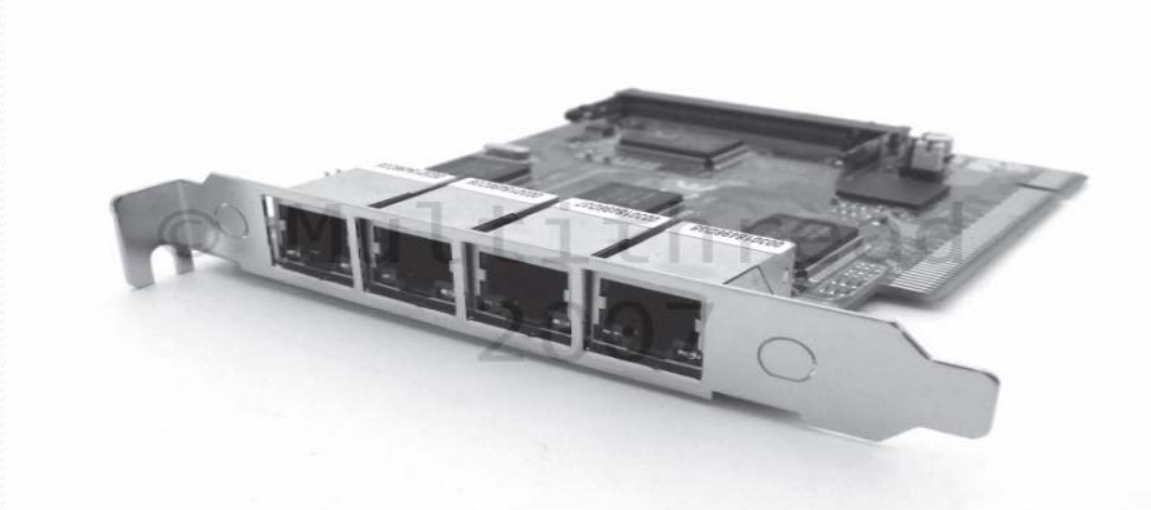


Fig. 5.7 需要時可以使用這種一卡四介面的產品

Table 5.1 伺服器整合的合併率計算

項目	實體伺服器設定	項目	最小實體伺服器設定 (高合併率計算5v/1p)	較大負擔的實體伺服器設定 (低合併率計算3v/1p)
需求vCPU 數目	85 個	CPU 設定	2 個4 核CPU ( 共8 核心)	刀鋒伺服器: 2 個4 核心或6核 心 ( 共8 核心或12核心)
需求pCPU 核心數	17或29 核心	每台虛擬機 量	22 個 ( 15 個1vCPU 及7個2 vCPU)	17 個 ( 12個1vCPU 及5 個2 vCPU)
實體CPU 數量	5 或8 個4 核心 CPU ( 使用 Xeon53XX 或 74XX 家族)	網路卡	內建2 個網路介面+1 張2 Port 的網路卡	使用刀鋒的Network Bus 以及 Module 擴充 (N個port * 5 只 要大於下方所寫的68個虛擬 機數目即可)
實際記憶體 數量	34GB ~116GB	記憶體	64GB ( 每pCPU 分配4GB, 剩 下的64-8*4給自己用)	64GB ( 每pCPU 分配4GB, 剩下的64-8*4給自己用)
實體網路介 面	10~13 個網路介 面	機器數量	3 台伺服器共66個虛擬機 ( 3*8 核=24核, 大於最低要求17核 即可, 最多可有24*5個虛擬 機, 大於85vCPU即可)	4片刀鋒共68個虛擬機 ( 4*8 核=32核, 大於最低要求29核 即可, 最多可有32*3個虛擬 機, 大於85vCPU即可)

# 6. 桌上虛擬化的合併率計算

- 桌上虛擬化的合併率和伺服器大同小異，計算方式也較單純，在應用上較著重於記憶體的直接對應關係。
  - CPU 計算：每實體核心相當**5~8** 個vCPU
  - 記憶體數量：**1GB/XP**，**2GB/Vista (Window 7)**
  - 網路卡的分配原則：**5~8** 個虛擬機/ 每網路介面
  - 實體伺服器設定：多台伺服器或多片刀鋒伺服器

# 桌上虛擬化的合併率計算(續)

- 依上述原則去精算以下範例:
  - 企業中有**120** 台實體機桌上機器
  - 其中有**20** 台是必須使用獨立的顯示卡，不適合虛擬化
  - **100** 台是可以虛擬化的日常工作個人電腦
  - **60** 台使用**1GB** 的**XP** 作業系統
  - **40** 台使用**2GB** 的**Windows 7** 作業系統

# 桌上虛擬化的合併率計算(續)

120 台實體桌上機器

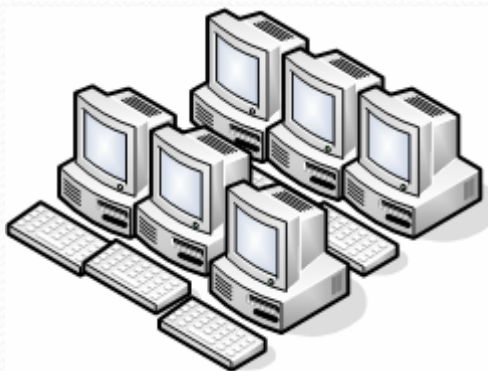
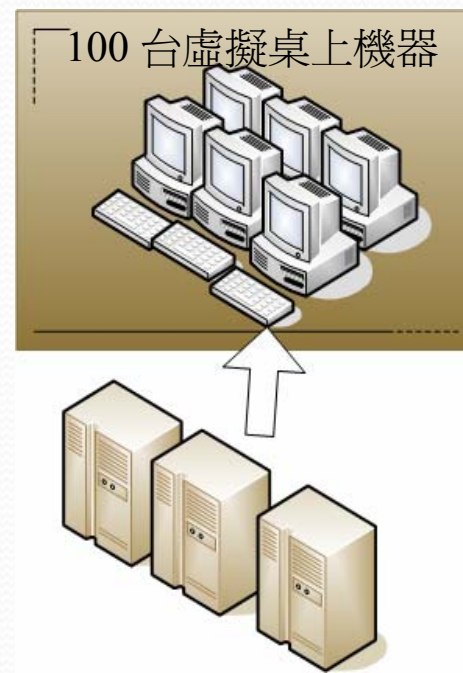


Fig. 6.1 桌上虛擬化之前

20 台實體桌上機器



Fig. 6.2 預計的桌上虛擬化之後



# 桌上虛擬化的合併率計算(續)

- CPU 計算：每實體核心相當5-8 個vCPU

桌上系統使用的工作較為單純，主要為辦公室應用以及網頁瀏覽，因此我們可以讓實體CPU 多負擔一些。從上述的CPU需求得知，我們需要的實體CPU 核心（pCPU）數量如下列計算公所示。

- 需求虛擬CPU數量為:

$$(100\text{台} \times 1\text{ vCPU}) = 100\text{ vCPU}$$

- 一個實體核心承載5個vCPU，我們所需要的實體CPU 核心數量就是  $100\text{ vCPU} / 5 = 20\text{ 實體CPU 核心}$

# 桌上虛擬化的合併率計算(續)

- CPU 計算：每實體核心5~8 個vCPU
  - 如果我們希望些實體機的合併率再高一點，那麼就讓一個pCPU 承載8個vCPU，我們所需要的實體CPU 核心數量就是 $100 \text{ vCPU} / 8 = 13$  實體CPU 核心
  - 目前的伺服器市場使用的產品大多是4核心的CPU，因此在上面的例子中，需要的實體4核心CPU 數量為：  
 $20 \text{ Core} / 4 = 5 \text{ CPU Packages}$  或是  $13 \text{ Core} / 4 = 4 \text{ CPU Packages}$

# 桌上虛擬化的合併率計算(續)

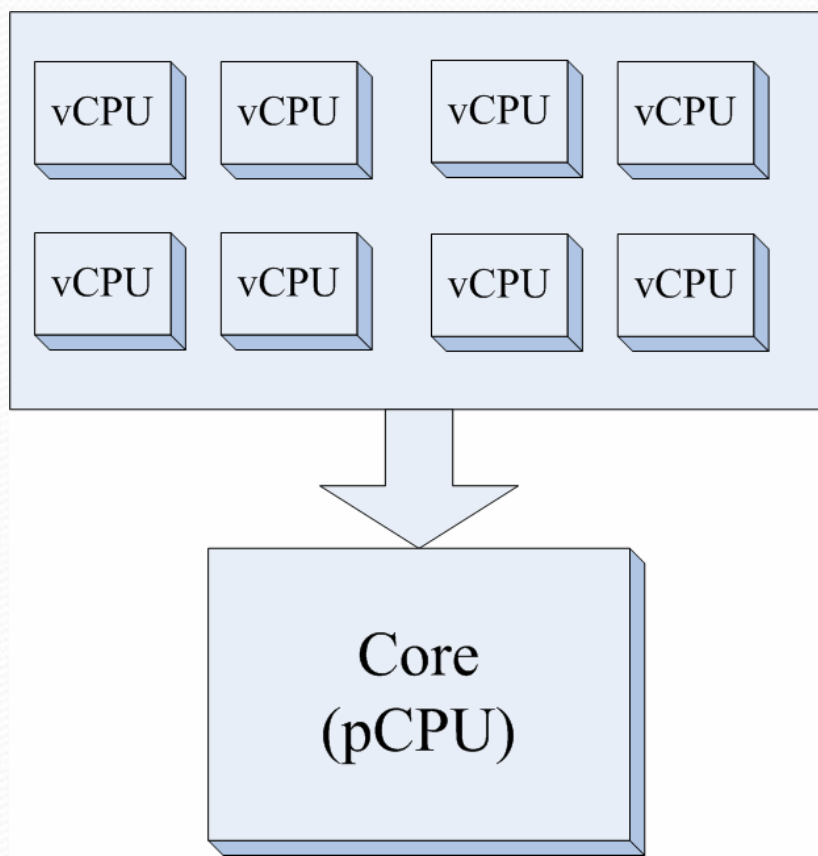


Fig. 6.3 桌上虛擬化的CPU 數量指示圖



# 桌上虛擬化的合併率計算(續)

- 記憶體數量：**1GB/XP**，**2GB/Vista (Windows 7)**:
  - 桌上系統使用1GB 的記憶體已足夠，就算是Vista Business 也不會造成瓶頸，但分配2GB 給使用Vista (Windows 7)的桌上系統，因此實際需要的記憶體數目為：  
 $1\text{GB} \times 60\text{台} + 2\text{GB} \times 40\text{台} = 60\text{GB} + 80\text{GB} = 140\text{GB}$ 的記憶體。
  - 因此桌上虛擬化的記憶體數量是由**虛擬機作業系統**決定。

# 桌上虛擬化的合併率計算(續)

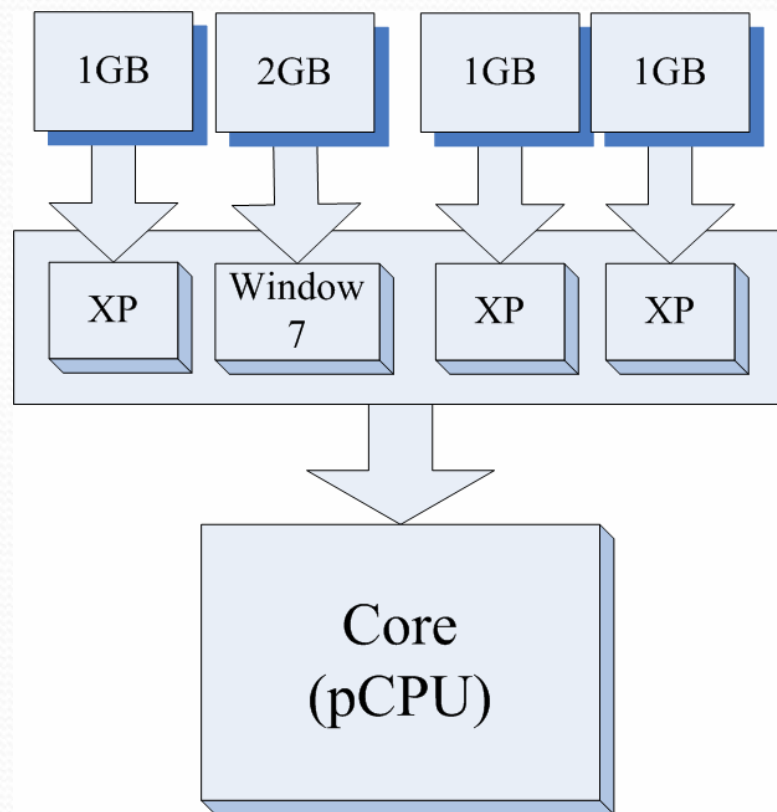


Fig. 6.4 使用Vista 或 Window 7 的系統要多一點記憶體

# 桌上虛擬化的合併率計算(續)

- 網路卡的分配原則：**5~8 個虛擬機/ 每網路介面**：
  - 大部分的桌上使用者主要是使用RDP 或是ICA 的遠端桌面來傳送畫面，在LAN 內的速度100MB 的速度已足夠，因此我們可以讓一個網路介面上傳送5~8個虛擬機，因此需要的網路介面為：  
100 個虛擬機/（一個網路介面為5~8 個虛擬機）  
=13~20 個網路介面

# 桌上虛擬化的合併率計算(續)

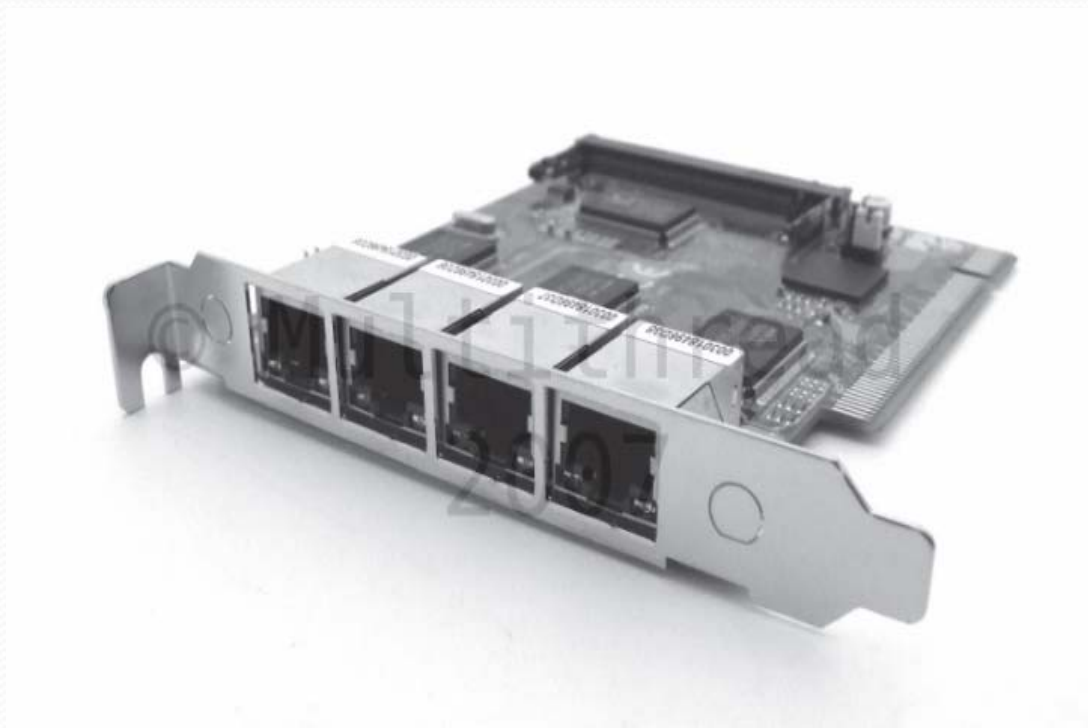


Fig. 6.5 需要時可以使用這種一卡四介面的產品

Table 6.1 桌上虛擬化的合併率計算

項目	實體伺服器設定	項目	最小實體伺服器設定 (高合併率計算8v/1p)	較大負擔的實體伺服器設定 (低合併率計算5v/1p)
需求vCPU數目	100 個	CPU 設定	1 個4 核心或2 個雙核心 CPU (共4 核心)	刀鋒伺服器: 2 個4 核心SMP (共8 核心)
需求pCPU 核心數	13 或20 核心	每台虛擬機量	25個虛擬機 (15 台XP 及10 台W7)	34個虛擬機(20 台XP 及14台 W7)
實體CPU 數量	4 或5 個 4核心 CPU (使用 Xeon 53XX 或 74XX 家族)	網路卡	內建2 個網路介面+1 張 2port的網路卡 ( 共4 個網路介面, 4*8=32大於上面 虛擬機台數25台即可)	使用刀鋒的Network Bus 以及 Module 擴充 (N個port * 5 只要大於底下的102虛擬機數量即可)
實際記憶體數量	140GB	記憶體	15*1+10*2+13=48GB (35GB 給虛擬機, 剩下 13G給自己使用)	20*1+14*2+16=64GB (48GB 給虛擬機, 剩下16G自己用)
實體網路介面	13~20 個網路介面	機器數量	4 台伺服器共100個虛擬機 (4*4=16核心, 大於最低 13核心即可, 16*8則最大 可承載128個虛擬機)	3 片刀鋒共102個虛擬機 (3*8 核=24核大於最低20核即可, 24*5則最大可承載120個虛擬 機)

# 7. 伺服器整合TCO/ROI 計算

- VMware公司是虛擬機產品大廠之一，這幾年不斷發展企業IT整體結構的虛擬化，目標是要將虛擬化導入企業的基礎建設中，而非只是一種單一產品或解決方案。
- 因此VMware 在推出自己的虛擬化產品之前，也委託了協力廠商的調查機構推出了成本估算工具，讓企業在進入虛擬化之前，可以先瞭解自己是否需要虛擬化。而藉著這個工具，可以推算出在虛擬化之後的成本TCO節省以及整體回報ROI。

# 伺服器整合TCO/ROI 計算(續)

- <http://roitco.vmware.com/vmw/>
- 這個網頁，顯示的就是標準VMware TCO/ROI 計算器，首先我們選擇地區和地點...等，選擇完地點之後，幣值選擇美金USD，專案名稱則隨意打。

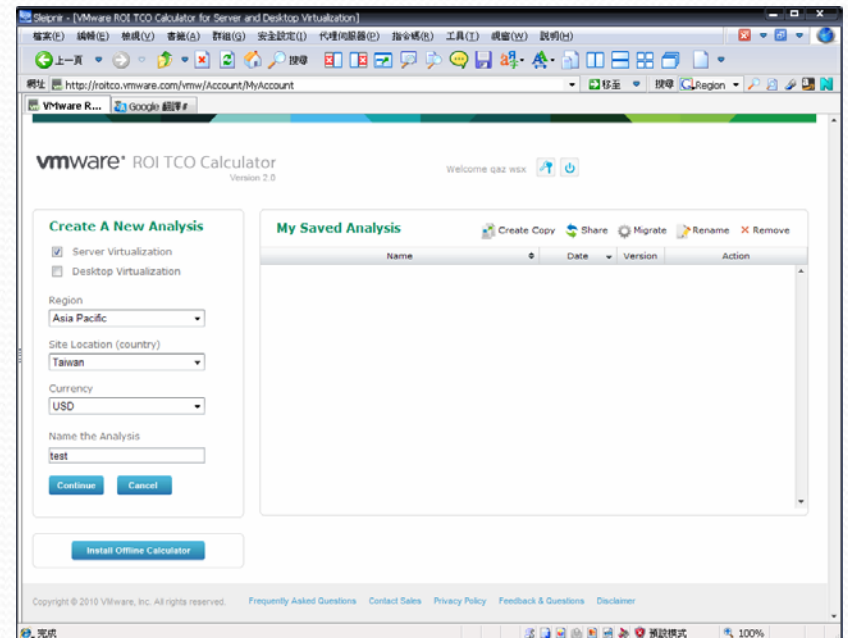


Fig. 7.1 登錄基本資料

# 伺服器整合TCO/ROI 計算(續)

- 選擇想要虛擬化的領域，一共有兩個選項。如果你的企業環境有很多伺服器想虛擬化，那麼就選擇第一項**Server Virtualization**。
- 如果你的企業伺服器不多，但每一個人都有一個桌上電腦或筆記電腦，那麼你需要的就是桌上整合，選擇**Desktop Virtualization**，當然以上的選擇可以複選。



# 伺服器整合TCO/ROI 計算(續)

## Create A New Analysis

- Server Virtualization
- Desktop Virtualization

Fig. 7.2. 選擇想要虛擬化的領域

# 伺服器整合TCO/ROI 計算(續)

- 在瞭解了基本資料之後，我們就可以根據需求來進行ROI的計算。在伺服器整合部分，主要就是將多台實體伺服器上的多個服務轉移到虛擬機器上。
- 以現代的硬體實力來說，平均的合併率可以高達1:20，換句話說，可以在一台功能較強的伺服器上執行20個虛擬機器，當然省下的錢不一定能到1:20，但整個成本也有明顯的下降，我們就來看看有關伺服器整合的TCO/ROI運算。

# 伺服器整合TCO/ROI 計算(續)

- 舉例來說，某公司機房伺服器目前的硬體設定如下
  - 未虛擬化之前的硬體設定：
    - 單CPU雙核心的伺服器：50 台
    - 雙CPU雙核心的伺服器：100 台
    - 四CPU雙核心的伺服器：50 台
    - 每台伺服器都有150GB的硬碟空間
  - 想要有200台虛擬化主機，那麼虛擬化五年後，能省下多少錢呢？
    - CPU 計算：每實體核心3個vCPU(虛擬機)
    - 假設公司工作量年增長率:3%。
    - 新主機服務器使用年限：3年
- 接下來就用VMware 提供的TCO/ROI工具來精算。

# 伺服器整合TCO/ROI 計算(續)

- 首先我們進入該網頁，並且填寫好地點、貨幣。選擇「continue」這一項。

vmware<sup>®</sup> ROI TCO Calculator  
Version 2.0

Welcome qaz wsx

Server Virtualization  
 Desktop Virtualization

Region  
Select One

Site Location (country)  
Select One

Currency  
USD

Name the Analysis

Continue Cancel

Install Offline Calculator

**My Saved Analysis**

Create Copy Share Migrate Rename Remove

Name	Date	Version	Action
test	23 Aug 2011	2.0	
twd	22 Aug 2011	2.0	
t2	19 Aug 2011	2.0	

Fig. 7.3 在進行之前，記得更改幣值、地區

# Room-in

vmware® ROI TCO Calculator  
Version 2.0

Welcome qaz wsx



## Create A New Analysis

- Server Virtualization
- Desktop Virtualization

Region

Select One

Site Location (country)

Select One

Currency

USD

Name the Analysis

Continue

Cancel

Install Offline Calculator

## My Saved Analysis

Create Copy Share Migrate Rename Remove

Name	Date	Version	Action
test	23 Aug 2011	2.0	
twd	22 Aug 2011	2.0	
t2	19 Aug 2011	2.0	

# 伺服器整合TCO/ROI 計算(續)

- 首先我們看到第一個畫面。上面選擇目前沒虛擬化之前的伺服器數量，這邊是以不同CPU 為主要區分的主機。
- 下面是已經虛擬化後的伺服器數量，還有目前這些虛擬化伺服器所執行的虛擬機數量，最後是虛擬化後可以使用最大虛擬機數量。

vSphere & vCenter Server [ vSphere & vCenter Server   < > >> >>> Finish ]					
1 Server Configuration 2 Server Consolidation 3 Product Selection 4 Beyond Server Consolidation 5 ROI					
<b>Year 0: Physical Servers (before virtualization)</b>					
Define Configuration of Physical Servers	1	2	50	50	\$ 1,751
(server costs excluding HBA's and TOE cards)	2	2	100	100	\$ 4,114
	4	2	50	50	\$ 12,657
					\$
					\$
Number of Unvirtualized Workloads & Servers			200	200	
<b>Year 0: New Host Servers (after virtualization)</b>					
Define Configuration of Host Servers	2	4	200	9	\$ 5,915
(server costs excluding HBA's and TOE cards)					\$
					\$
					\$
Number of Virtual Machines & Host Servers			200	9	
Current Consolidation Ratio (Workloads / Server)					22:1
Max Consolidation Ratio (Workloads / Server)					24:1

Fig. 7.4 將所有資料填入



# 伺服器整合TCO/ROI 計算(續)

- 接下來是CPU 資源的計算。  
目前的Data Center 如果有50 台1 個CPU 的主機，100 台2 個CPU 的主機，50 台4 CPU 的主機，就照著填入。
- 注意這些都是以雙核心的CPU 為預設值。

CPU's / Server	Cores / CPU	Number of Workloads	Number of Servers
1	2	50	50
2	2	100	100
4	2	50	50
		200	200

Fig. 7.5 將所有資料填入











# 伺服器整合TCO/ROI 計算(續)

- 在過程中填入企業環境的實體主機相關設定以及預計模擬幾年後...等選項，填完之後一直按下「Next」進入下一頁。
- 最後系統就會列出TCO/ROI 試算表，並且會將沒有虛擬化和虛擬化之後的每一個成本細項作比較。

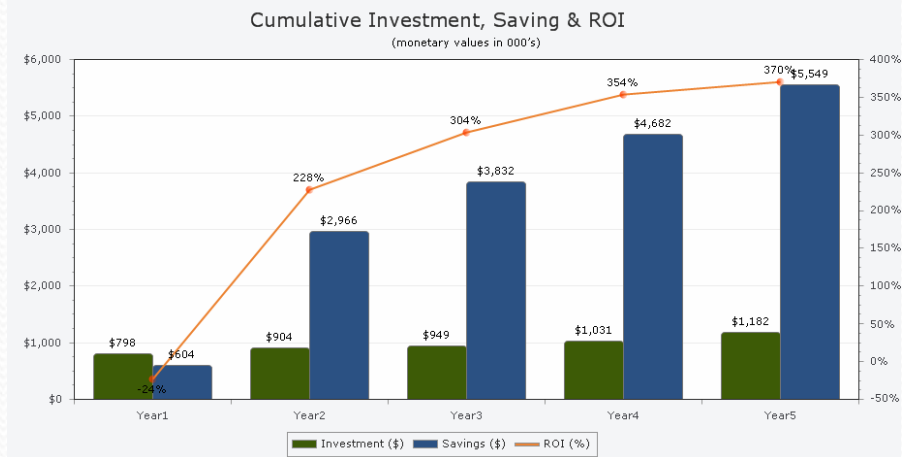
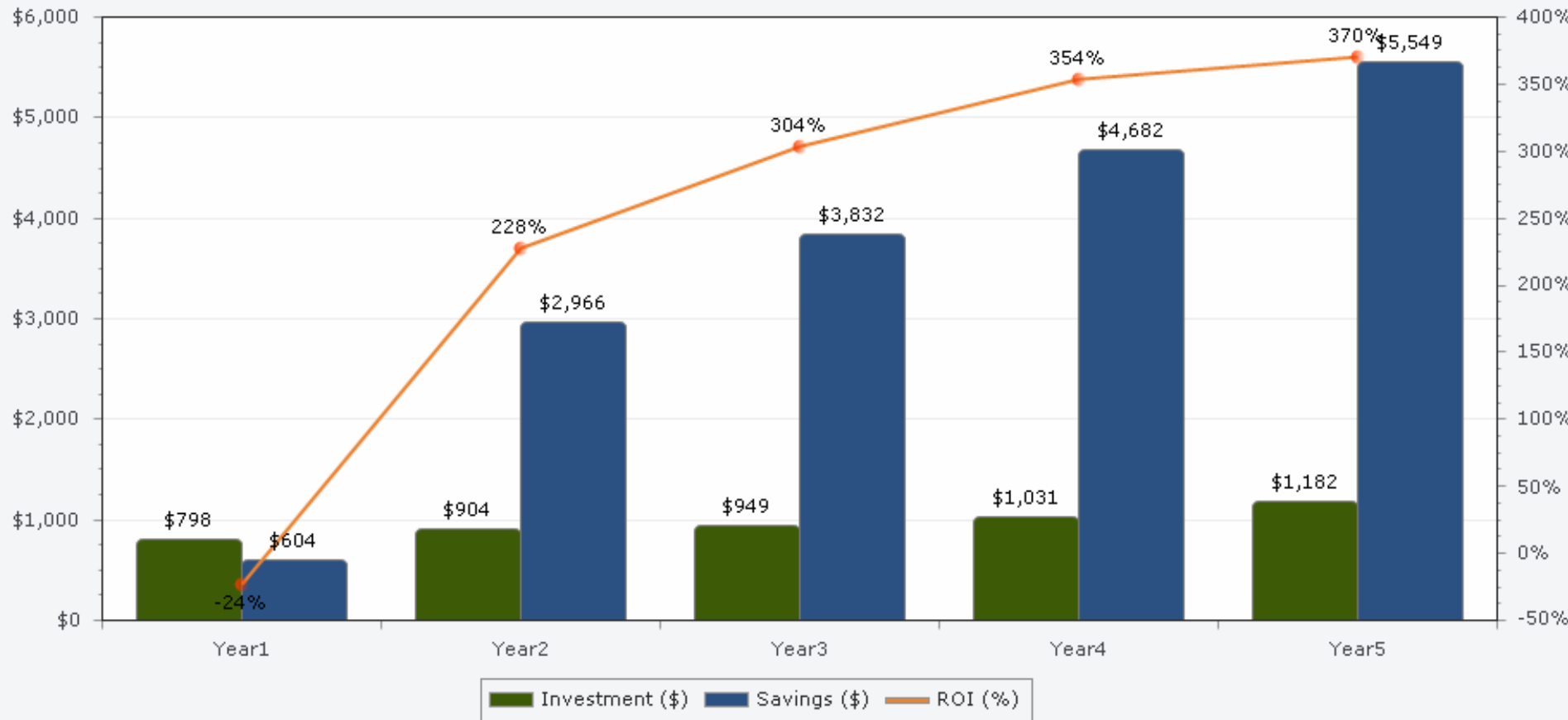


Fig. 7.6 圖表顯示出來其金額差距之大

# Room-in

## Cumulative Investment, Saving & ROI

(monetary values in 000's)

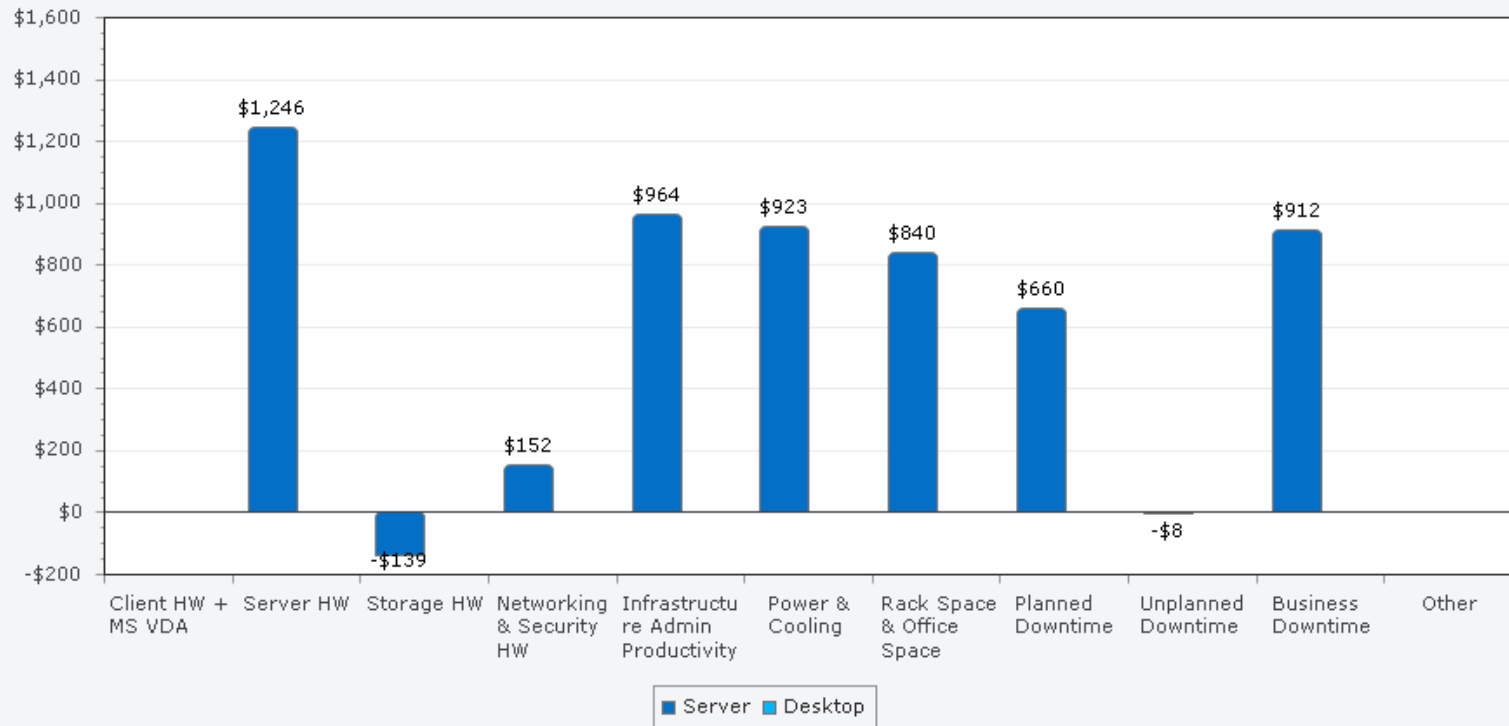


# Room-in

Total Savings: \$ 5,549,353

## Distribution of Savings: 5 Year Total

(monetary values in 000's)

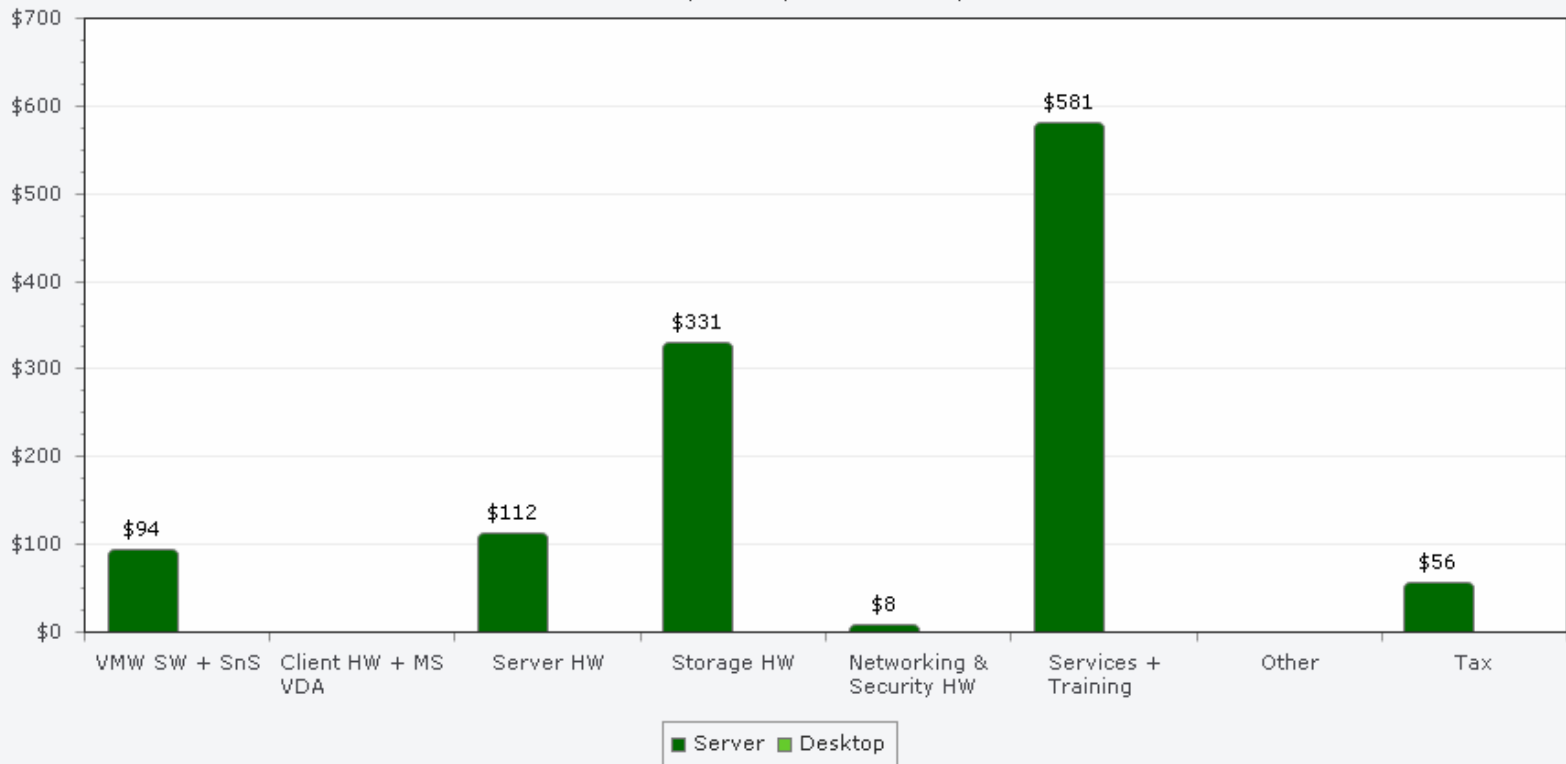


# Room-in

Total Investment: \$ 1,181,876

## Distribution of Investment: 5 Year Total

(monetary values in 000's)





# Room-in

Physical Total:

\$ 8,461,270

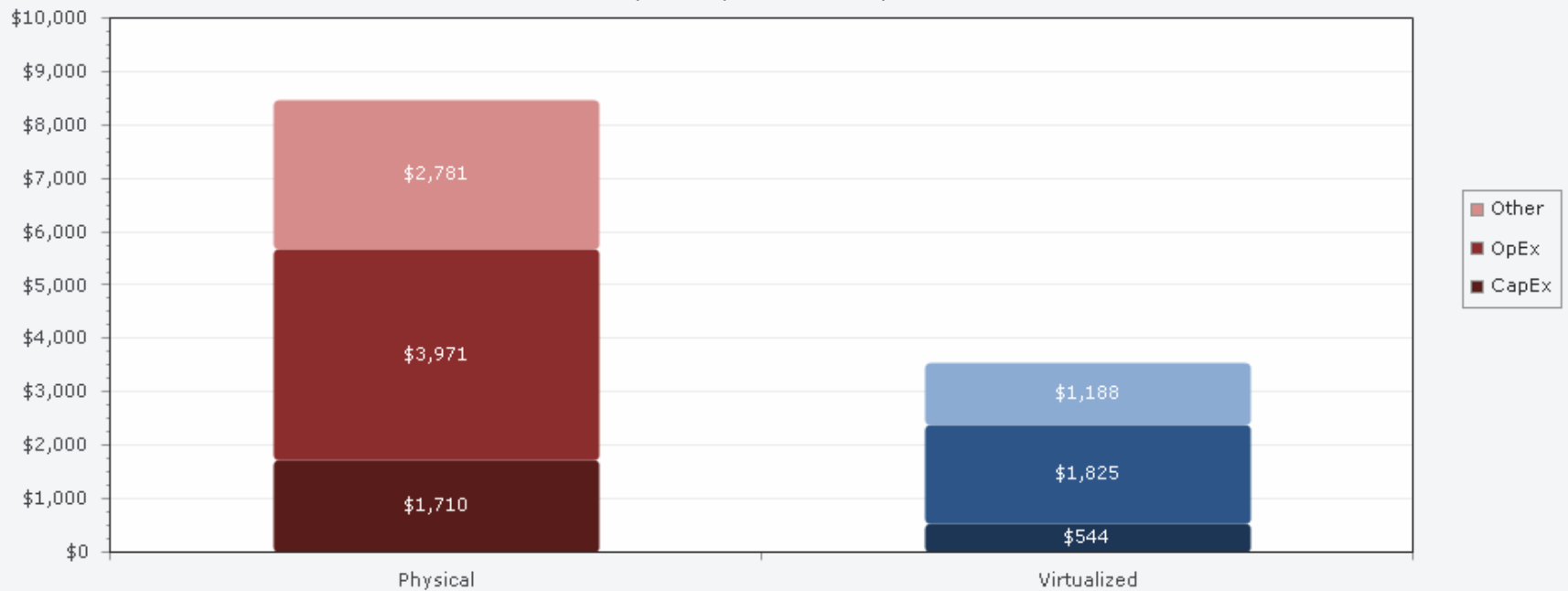
Virtualized Total:

\$ 3,557,421

58% Reduction

## Total Cost of Ownership: 5 Year Total

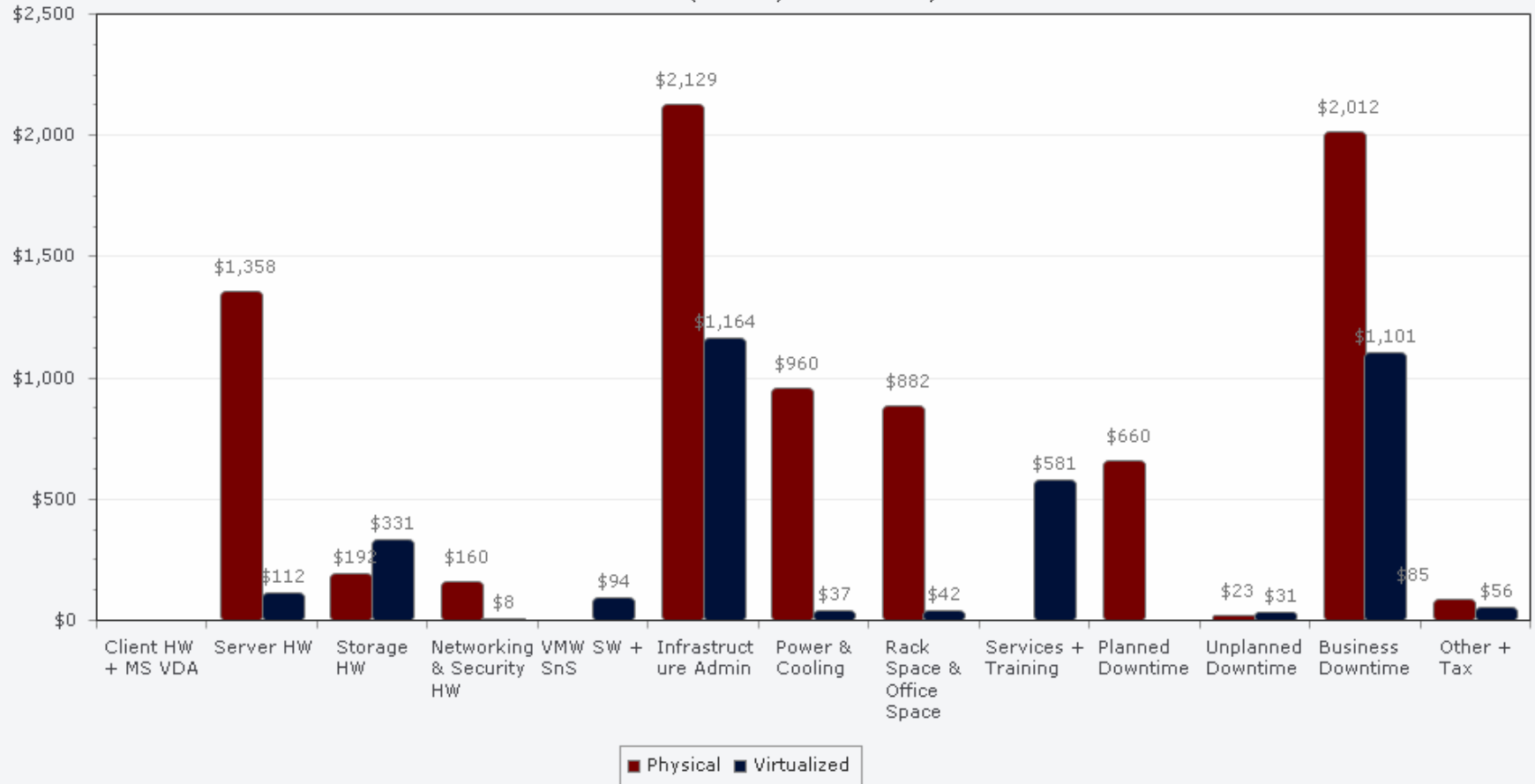
(monetary values in 000's)



# Room-in

## Total Cost of Ownership: 5 Year Total

(monetary values in 000's)



# 伺服器整合TCO/ROI 計算(續)

- 你可以直接按下圖表，顯示詳細的比較圖，或是按上面按鈕選擇輸出至PPT檔案或是電子郵件等功能。
- 在本圖表中第三列的百分比，顯示的是ROI，圖中是**370%**，相當高的數字！Total Savings 為5年省下的花費淨值，第四列則是回本的時間，本例中是大約1.2年。

Export Analysis to PowerPoint

Fig. 7.7. 將分析的報表作成投影片下載

Cost Benefit Analysis		[Expand]
	Over 3 yrs.	Over 5 yrs.
Investment	\$ 948,635	\$ 1,181,876
Total Savings	\$ 3,831,681	\$ 5,549,353
ROI	304 %	370 %
Payback	1.1 years	1.2 years

Fig. 7.8 本例中的ROI 相當高，有370%

# Cumulative Investment, Savings & ROI

	Year 1	Year 2	Year 3	Year 4	Year 5
<b><u>Investment vs. Savings vs. ROI (cumulative)</u></b>					
Investment (\$)	\$ 797,613	\$ 904,269	\$ 948,635	\$ 1,030,989	\$ 1,181,876
Savings (\$)	\$ 604,068	\$ 2,965,637	\$ 3,831,681	\$ 4,682,420	\$ 5,549,353
ROI (%)	(24)%	228%	304%	354%	370%
Annual Savings	\$ 604,068	\$ 2,361,569	\$ 866,044	\$ 850,739	\$ 866,933
3 Year Payback	NA	1.1	NA	NA	NA
5 Year Payback	NA	1.2	NA	NA	NA

# Distribution of Savings

		Total 5 Yrs		
		Server	Desktop	Total
CapEx	Client HW + MS VECD	\$ 0	\$ 0	\$ 0
	Server HW	\$ 1,245,835	\$ 0	\$ 1,245,835
	Storage HW	\$ -138,800	\$ 0	\$ -138,800
	Networking & Security HW	\$ 152,000	\$ 0	\$ 152,000
OpEx	Infrastructure Admin Productivity	\$ 964,408	\$ 0	\$ 964,408
	Power & Cooling	\$ 922,556	\$ 0	\$ 922,556
	Rack Space & Office Space	\$ 840,000	\$ 0	\$ 840,000
Other	Planned Downtime	\$ 660,000	\$ 0	\$ 660,000
	Unplanned Downtime	\$ -8,400	\$ 0	\$ -8,400
	Business Downtime	\$ 911,754	\$ 0	\$ 911,754
	Other	\$ 0	\$ 0	\$ 0
	<b>Total</b>	<b>\$ 5,549,353</b>	<b>\$ 0</b>	<b>\$ 5,549,353</b>

# Distribution of Investment

		Total 5 Yrs		
		Server	Desktop	Total
Investment	VMW SW + SnS	\$ 93,619	\$ 0	\$ 93,619
	Client HW + MS VECD	\$ 0	\$ 0	\$ 0
	Server HW	\$ 112,376	\$ 0	\$ 112,376
	Storage HW	\$ 330,500	\$ 0	\$ 330,500
	Networking & Security HW	\$ 8,000	\$ 0	\$ 8,000
	Services + Training	\$ 581,101	\$ 0	\$ 581,101
	Other	\$ 0	\$ 0	\$ 0
	Tax	\$ 56,280	\$ 0	\$ 56,280
	<b>Total</b>	<b>\$ 1,181,876</b>	<b>\$ 0</b>	<b>\$ 1,181,876</b>

# Total Cost of Ownership

		Total 5 Yrs		
		Physical	Virtualized	% Reduction
CapEx		\$ 1,709,911	\$ 544,495	68%
OpEx		\$ 3,970,619	\$ 1,824,756	54%
Other		\$ 2,780,740	\$ 1,188,170	57%
	<b>Total</b>	<b>\$ 8,461,270</b>	<b>\$ 3,557,421</b>	<b>58%</b>

# Total Cost of Ownership - 5 Year Total

		Physical			Virtualized			% Reduction		
		Server	Desktop	Total	Server	Desktop	Total	Server	Desktop	Total
CapEx	Client HW + MS VECD/VDA	\$ 0	\$ 0	<b>\$ 0</b>	\$ 0	\$ 0	<b>\$ 0</b>	0%	0%	<b>0%</b>
	Server HW	\$ 1,358,211	\$ 0	<b>\$ 1,358,211</b>	\$ 112,376	\$ 0	<b>\$ 112,376</b>	92%	0%	<b>92%</b>
	Storage HW	\$ 191,700	\$ 0	<b>\$ 191,700</b>	\$ 330,500	\$ 0	<b>\$ 330,500</b>	(72)%	0%	<b>(72)%</b>
	Networking & Security HW	\$ 160,000	\$ 0	<b>\$ 160,000</b>	\$ 8,000	\$ 0	<b>\$ 8,000</b>	95%	0%	<b>95%</b>
	VMW SW + SnS	\$ 0	\$ 0	<b>\$ 0</b>	\$ 93,619	\$ 0	<b>\$ 93,619</b>	0%	0%	<b>0%</b>
OpEx	Infrastructure Admin	\$ 2,128,663	\$ 0	<b>\$ 2,128,663</b>	\$ 1,164,255	\$ 0	<b>\$ 1,164,255</b>	45%	0%	<b>45%</b>
	Power & Cooling	\$ 959,956	\$ 0	<b>\$ 959,956</b>	\$ 37,400	\$ 0	<b>\$ 37,400</b>	96%	0%	<b>96%</b>
	Rack Space & Office Space	\$ 882,000	\$ 0	<b>\$ 882,000</b>	\$ 42,000	\$ 0	<b>\$ 42,000</b>	95%	0%	<b>95%</b>
	Services + Training	\$ 0	\$ 0	<b>\$ 0</b>	\$ 581,101	\$ 0	<b>\$ 581,101</b>	0%	0%	<b>0%</b>
Other	Planned Downtime	\$ 660,000	\$ 0	<b>\$ 660,000</b>	\$ 0	\$ 0	<b>\$ 0</b>	100%	0%	<b>100%</b>
	Unplanned Downtime	\$ 22,800	\$ 0	<b>\$ 22,800</b>	\$ 31,200	\$ 0	<b>\$ 31,200</b>	(37)%	0%	<b>(37)%</b>
	Business Downtime	\$ 2,012,444	\$ 0	<b>\$ 2,012,444</b>	\$ 1,100,690	\$ 0	<b>\$ 1,100,690</b>	45%	0%	<b>45%</b>
	Other + Tax	\$ 85,496	\$ 0	<b>\$ 85,496</b>	\$ 56,280	\$ 0	<b>\$ 56,280</b>	34%	0%	<b>34%</b>
	<b>Total</b>	<b>\$ 8,461,270</b>	<b>\$ 0</b>	<b>\$ 8,461,270</b>	<b>\$ 3,557,421</b>	<b>\$ 0</b>	<b>\$ 3,557,421</b>	<b>58%</b>	<b>0%</b>	<b>58%</b>



# 8. TCO/ROI 計算器詳細說明

- 這個TCO/ROI 計算器的表格中的專案還是很詳細的，在 vSphere 的TCO/ROI 表格中，共有下面幾個大類：
- 伺服器硬體（Datacenter Server Hardware）：虛擬化前後的三年硬體成本比較，是最大省錢的部分。
- 省下軟體授權費用（Software License Cost Avoidance）：這邊指的是原來在實體機上的作業系統費用。
- 存放裝置（Datacenter Server Storage）：要注意的是，此項為虛擬化時必須增加的設備，因此會花錢，投資回報率為負值。

# TCO/ROI 計算器詳細說明 (續)

- 網路設備 (Datacenter Server Networking) : 主要是網路卡和交換機、路由器的設備，虛擬化後多為正值。
- 冷卻及電力 (Datacenter Server Power and Cooling Consumption) : 電力和冷卻伺服器的花費，虛擬化後多為正值。
- 場地租金 (Datacenter Server Space) : 放伺服器和設備的場地租金，虛擬化後多為正值。

# TCO/ROI 計算器詳細說明 (續)

- 伺服器規劃 (Datacenter Server Provisioning)：這是規劃伺服器的人工費用，虛擬化後多為正值，因為虛擬化時，我們可以用模版來直接套用而省去安裝伺服器的麻煩。
- 伺服器性能管理費用 (Datacenter Server Performance Management)：電腦少了，再加上一些防災的功能 (VMotion, DRS, HA)，這個費用也會少，虛擬化後多為正值。
- 網路改變及設定管理 (Network Change and Configuration Management)：當網路出現變化時的成本。

# TCO/ROI 計算器詳細說明 (續)

- 伺服器管理 (Datacenter Server Administration) : 管理伺服器的成本。
- 因災難而損失的費用 : vSphere 在這邊認定為正值，即災難發生的損失較少。
- 伺服器停機時間損失 (Datacenter Server Unplanned Downtime) : 由於虛擬化後，停機時間較少，虛擬化後多為正值。

# TCO/ROI 計算器詳細說明 (續)

- vSphere 軟體授權費用：這是虛擬化中最花錢的部分，此例中，一台伺服器安裝vSphere 要4 萬，這比一台4CPU 的伺服器還要貴了，當然是負值。
- vSphere 設計及部署：vSphere 的設計、規劃及部署也是收費的，因此為負值。
- vSphere 教育訓練：使用vSphere 必須上課。

# 9. 桌上虛擬化TCO/ROI 計算

- 在瞭解了伺服器合併的成本計算後，桌上虛擬化的TCO/ROI 就相對簡單多了。在桌上虛擬化的過程中，將桌上系統合併至伺服器上，必須購買(a)新的伺服器、(b)用戶端設備及(c)存放裝置，再加上(d)虛擬化的軟體，因此回收率ROI會比伺服器合併較低。
- 桌上虛擬化的合併率可比伺服器高一些，下面仍然是以VMware 的TCO/ROI工具來計算。

# 桌上虛擬化TCO/ROI 計算 (續)

- 首先仍然進入VMware 的 TCO/ROI 計算頁，此時選「Desktop」部分。



**Create A New Analysis**

Server Virtualization  
 Desktop Virtualization

Region  
Asia Pacific

Site Location (country)  
Taiwan

Currency  
USD

Name the Analysis  
t2

Continue Cancel

Fig. 9.1 選擇桌上虛擬化的部分

# 桌上虛擬化TCO/ROI 計算 (續)

- 之後會出現Fig. 9.2 的畫面，分別為需要虛擬化的桌上系統數量，再來是每一台電腦的成本，包括了硬體成本和作業系統的成本，調整一個核心要執行幾個虛擬機器、虛擬機要多少記憶體等等選項，我們就依次輸入。

Personal Computers			
Number of PC's to virtualize	200		
Average cost per PC	\$ 1,500		
Servers to Host PC's			
	CPU's / Server	Cores / CPU	Avg Price / Server
Define Configuration of Host Servers	4	4	\$ 14,555
Additional RAM + Other			\$ 6,280
Total Cost per New Server			\$ 20,835

Fig. 9.2 依次輸入資料













# 桌上虛擬化TCO/ROI 計算 (續)

- 輸入完畢之後按下「Next」，就會出現報表。可以看到ROI 是314% 左右，比伺服器低了不少，5 年的淨值約一百四十萬美元，回收時長約為 8 個月。

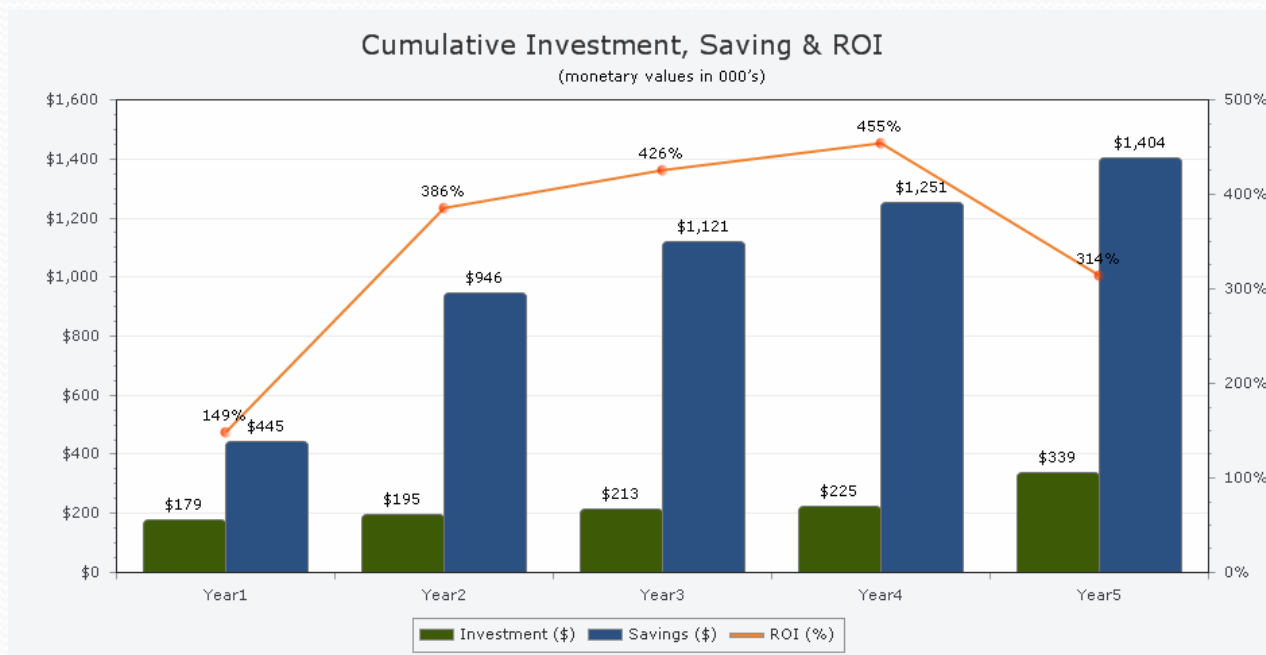
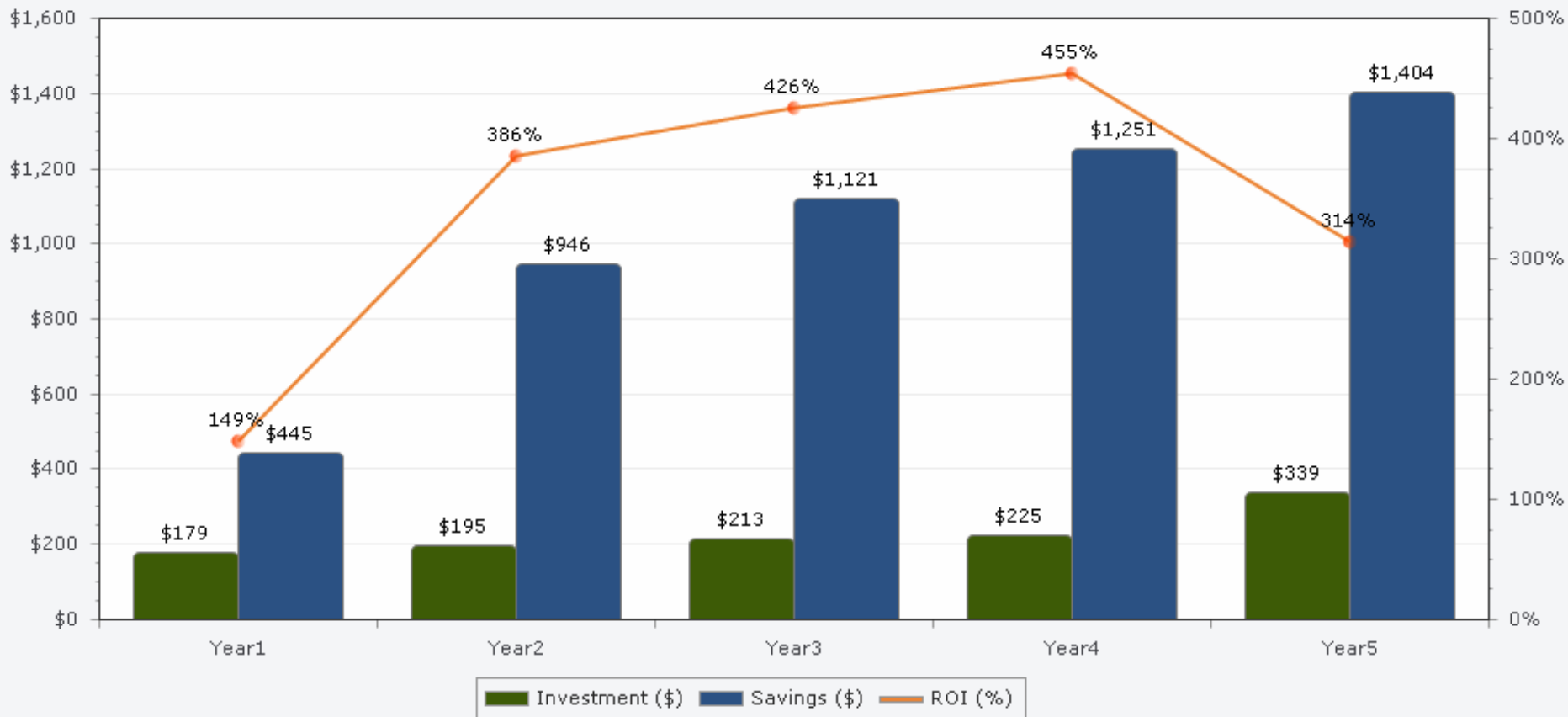


Fig. 9.3 桌上虛擬化的ROI 較低，但和實體比起來也有300% 以上

# Room-in

## Cumulative Investment, Saving & ROI

(monetary values in 000's)

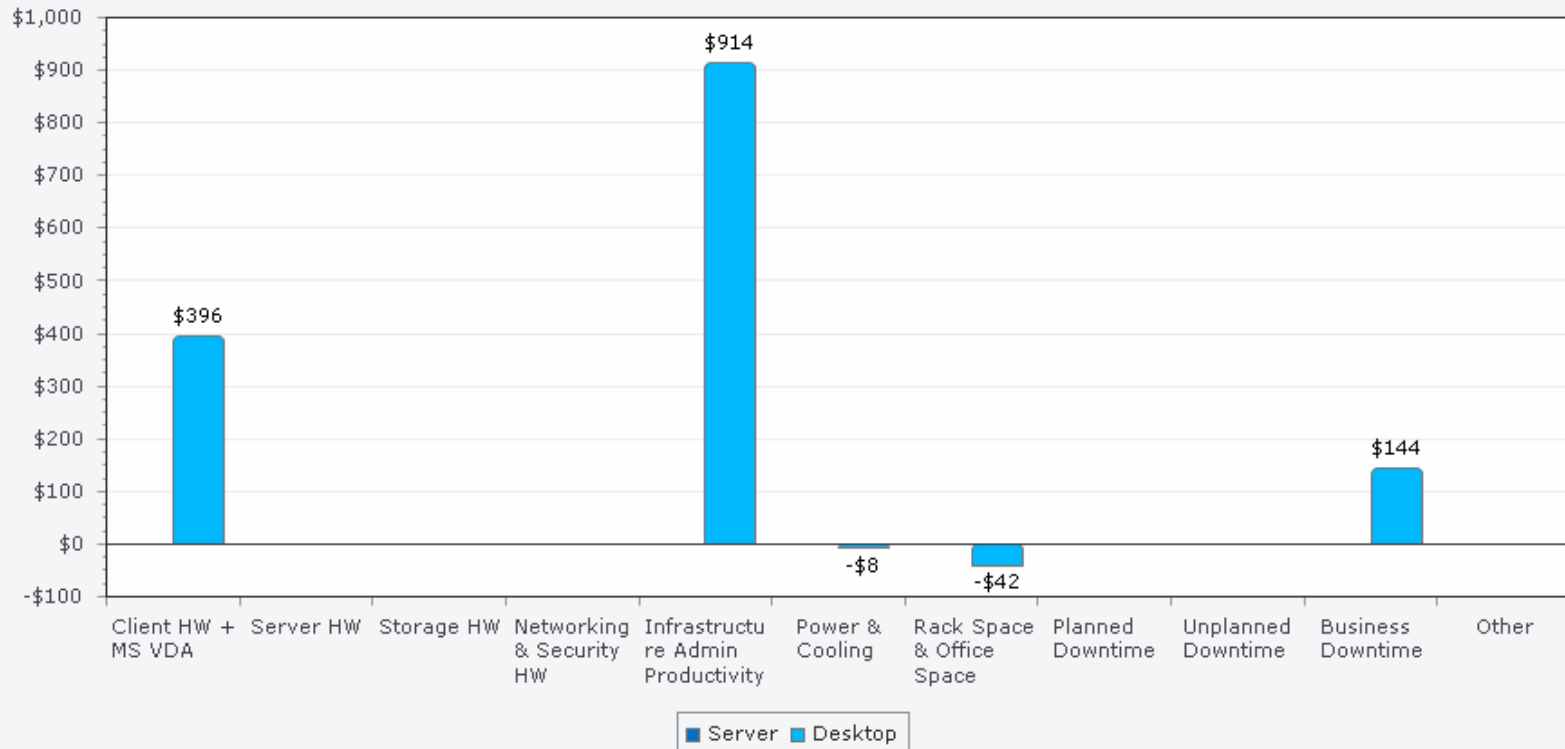


# Room-in

**Total Savings:**      **\$ 1,404,372**

## Distribution of Savings: 5 Year Total

(monetary values in 000's)





# Room-in

**Total Investment:** \$ 339,256

## Distribution of Investment: 5 Year Total

(monetary values in 000's)

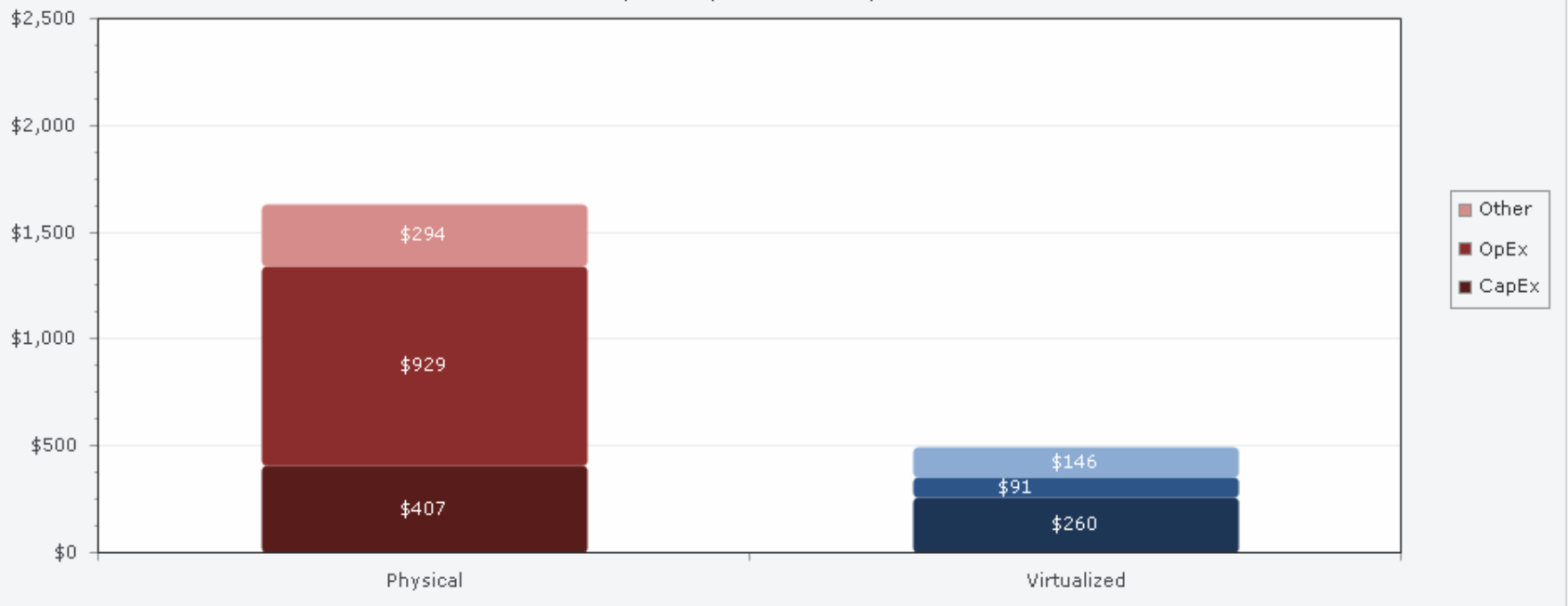


# Room-in

**Physical Total:** \$ 1,630,219      **Virtualized Total:** \$ 496,741      **70% Reduction**

## Total Cost of Ownership: 5 Year Total

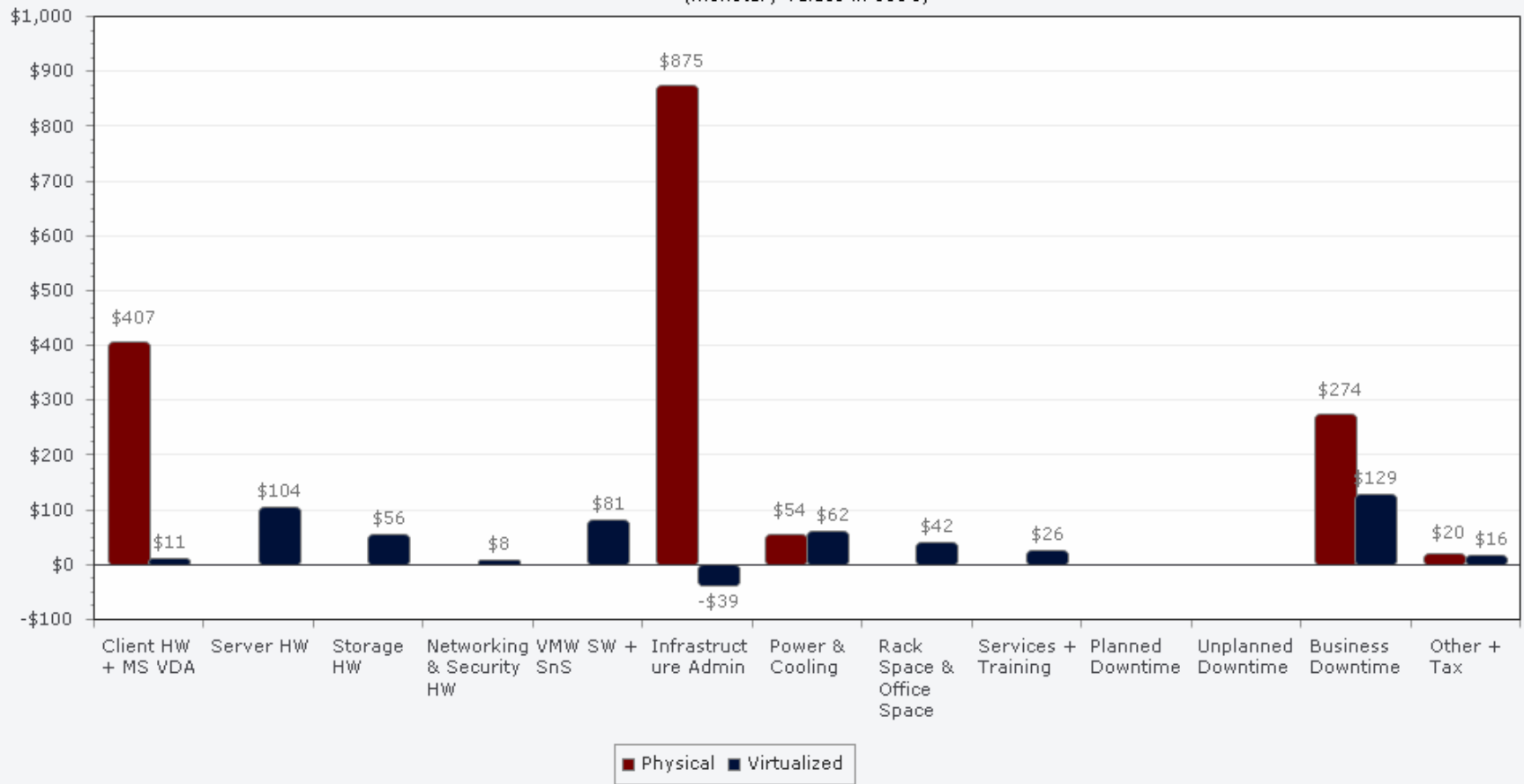
(monetary values in 000's)



# Room-in

## Total Cost of Ownership: 5 Year Total

(monetary values in 000's)



# 桌上虛擬化TCO/ROI 計算 (續)

- 在下面，公司5年後總獲益為一百四十萬美元，總支出為三十三萬美元(5年後的ROI比3年後的ROI還低的原因，是因為電腦等硬體設備的汰換)。

Cost Benefit Analysis		[Expand]
	Over 3 yrs.	Over 5 yrs.
Investment	\$ 213,085	\$ 339,256
Total Savings	\$ 1,121,370	\$ 1,404,372
ROI	426 %	314 %
Payback	0.5 years	0.8 years

Fig. 9.4 桌上虛擬化的回收也很高

# 桌上虛擬化TCO/ROI 計算 (續)

- 桌上虛擬化的花費：
  - 首先是VMware View Servers，每台需要將近37 萬元，這邊可以想見使用的伺服器都是4CPU 的。
  - VMware Virtual Center Server 是一台執行 Windows Server 2003 的資料庫，也需要購買。
  - FC SAN 交換機：這是在桌上虛擬化時需要的交換機。
  - FC SAN：在桌上虛擬化之後所需要的存放裝置。
  - Connection Broker：這也是一台安裝Connection Broker 的電腦硬體。

# Cumulative Investment, Savings & ROI

	Year 1	Year 2	Year 3	Year 4	Year 5
<b><u>Investment vs. Savings vs. ROI (cumulative)</u></b>					
Investment (\$)	\$ 178,570	\$ 194,883	\$ 213,085	\$ 225,492	\$ 339,256
Savings (\$)	\$ 445,239	\$ 946,358	\$ 1,121,370	\$ 1,251,299	\$ 1,404,372
ROI (%)	149%	386%	426%	455%	314%
Annual Savings	\$ 445,239	\$ 501,119	\$ 175,012	\$ 129,929	\$ 153,073
3 Year Payback	0.5	NA	NA	NA	NA
5 Year Payback	0.8	NA	NA	NA	NA

# Distribution of Savings

		Total 5 Yrs		
		Server	Desktop	Total
CapEx	Client HW + MS VECD	\$ 0	\$ 396,250	\$ 396,250
	Server HW	\$ 0	\$ 0	\$ 0
	Storage HW	\$ 0	\$ 0	\$ 0
	Networking & Security HW	\$ 0	\$ 0	\$ 0
OpEx	Infrastructure Admin Productivity	\$ 0	\$ 913,725	\$ 913,725
	Power & Cooling	\$ 0	\$ -7,746	\$ -7,746
	Rack Space & Office Space	\$ 0	\$ -42,000	\$ -42,000
Other	Planned Downtime	\$ 0	\$ 0	\$ 0
	Unplanned Downtime	\$ 0	\$ 0	\$ 0
	Business Downtime	\$ 0	\$ 144,143	\$ 144,143
	Other	\$ 0	\$ 0	\$ 0
	<b>Total</b>	<b>\$ 0</b>	<b>\$ 1,404,372</b>	<b>\$ 1,404,372</b>

# Distribution of Investment

		Total 5 Yrs		
		Server	Desktop	Total
Investment	VMW SW + SnS	\$ 0	\$ 80,625	\$ 80,625
	Client HW + MS VECD	\$ 0	\$ 48,000	<b>\$ 48,000</b>
	Server HW	\$ 0	\$ 104,177	<b>\$ 104,177</b>
	Storage HW	\$ 0	\$ 56,112	<b>\$ 56,112</b>
	Networking & Security HW	\$ 0	\$ 8,000	<b>\$ 8,000</b>
	Services + Training	\$ 0	\$ 26,187	<b>\$ 26,187</b>
	Other	\$ 0	\$ 0	<b>\$ 0</b>
	Tax	\$ 0	\$ 16,155	<b>\$ 16,155</b>
	<b>Total</b>	<b>\$ 0</b>	<b>\$ 339,256</b>	<b>\$ 339,256</b>



# Total Cost of Ownership

		Total 5 Yrs		
		Physical	Virtualized	% Reduction
CapEx		\$ 407,250	\$ 259,914	36%
OpEx		\$ 929,030	\$ 91,237	90%
Other		\$ 293,940	\$ 145,590	50%
	<b>Total</b>	<b>\$ 1,630,219</b>	<b>\$ 496,741</b>	<b>70%</b>

# Total Cost of Ownership - 5 Year Total

		Physical			Virtualized			% Reduction		
		Server	Desktop	Total	Server	Desktop	Total	Server	Desktop	Total
CapEx	Client HW + MS VECD/VDA	\$ 0	\$ 407,250	<b>\$ 407,250</b>	\$ 0	\$ 11,000	<b>\$ 11,000</b>	0%	97%	<b>97%</b>
	Server HW	\$ 0	\$ 0	<b>\$ 0</b>	\$ 0	\$ 104,177	<b>\$ 104,177</b>	0%	0%	<b>0%</b>
	Storage HW	\$ 0	\$ 0	<b>\$ 0</b>	\$ 0	\$ 56,112	<b>\$ 56,112</b>	0%	0%	<b>0%</b>
	Networking & Security HW	\$ 0	\$ 0	<b>\$ 0</b>	\$ 0	\$ 8,000	<b>\$ 8,000</b>	0%	0%	<b>0%</b>
	VMW SW + SnS	\$ 0	\$ 0	<b>\$ 0</b>	\$ 0	\$ 80,625	<b>\$ 80,625</b>	0%	0%	<b>0%</b>
OpEx	Infrastructure Admin	\$ 0	\$ 874,666	<b>\$ 874,666</b>	\$ 0	\$ -39,059	<b>\$ -39,059</b>	0%	104%	<b>104%</b>
	Power & Cooling	\$ 0	\$ 54,363	<b>\$ 54,363</b>	\$ 0	\$ 62,109	<b>\$ 62,109</b>	0%	(14)%	<b>(14)%</b>
	Rack Space & Office Space	\$ 0	\$ 0	<b>\$ 0</b>	\$ 0	\$ 42,000	<b>\$ 42,000</b>	0%	0%	<b>0%</b>
	Services + Training	\$ 0	\$ 0	<b>\$ 0</b>	\$ 0	\$ 26,187	<b>\$ 26,187</b>	0%	0%	<b>0%</b>
Other	Planned Downtime	\$ 0	\$ 0	<b>\$ 0</b>	\$ 0	\$ 0	<b>\$ 0</b>	0%	0%	<b>0%</b>
	Unplanned Downtime	\$ 0	\$ 0	<b>\$ 0</b>	\$ 0	\$ 0	<b>\$ 0</b>	0%	0%	<b>0%</b>
	Business Downtime	\$ 0	\$ 273,577	<b>\$ 273,577</b>	\$ 0	\$ 129,435	<b>\$ 129,435</b>	0%	53%	<b>53%</b>
	Other + Tax	\$ 0	\$ 20,363	<b>\$ 20,363</b>	\$ 0	\$ 16,155	<b>\$ 16,155</b>	0%	21%	<b>21%</b>
	<b>Total</b>	<b>\$ 0</b>	<b>\$ 1,630,219</b>	<b>\$ 1,630,219</b>	<b>\$ 0</b>	<b>\$ 496,741</b>	<b>\$ 496,741</b>	<b>0%</b>	<b>70%</b>	<b>70%</b>

Fig. 9.5 桌上虛擬化在伺服器的成本上相當高

# 10. 參考文獻

1. 全球最強VMware vSphere 4企業環境建構, 胡嘉璽, 上奇科技, 2011.  
[http://www.grandtech.com.tw/About/books\\_det.aspx?id=925](http://www.grandtech.com.tw/About/books_det.aspx?id=925)
2. VMware Virtualization, 2011.  
<http://www.vmware.com/virtualization>
3. VMware ROI TCO Calculator Version 2.0, VMware ROI TCO Calculator for Server and Desktop Virtualization, 2011.  
<http://roitco.vmware.com/vmw/>
4. Citrix System, XenServer, 2011.  
<http://www.citrix.com/English/ps2/products/product.asp?contentID=683148>

簡報完畢，謝謝！

