Capitolo 1: Il Nuovo Processo di Boot Chaining di Windows 7 e Windows Server 2008 R2

Introduzione

A partire da Windows Vista, Microsoft ha completamente rivoluzionato la gestione del processo di avvio (spesso indicato con il termine inglese *boot-chaining*) del sistema operativo. Questa modifica si è resa necessaria sia per motivi di sicurezza che per venire incontro alla complessità dell'hardware e firmware dei nuovi computer.

All'interno di questa nuova procedura di boot, viene utilizzato un nuovo *repository* o *data store* contenente le informazioni relative alle applicazioni che devono gestire il (re-)*boot* di un sistema operativo o l'avvio di eventuali applicazioni diagnostiche (e.g.: Windows Preinstallation Environment (WinPE)). Il nuovo processo di *boot-chaining* risulta completamente indipendente dal firmware utilizzato (*firmware-independent*) sul generico computer. La configurazione e le regole per la gestione del processo di boot sono salvate in un database di *boot* identificato come *Boot Configuration Data* (BDC), utilizzato dai nuovi sistemi operativi Microsoft a partire da Windows Vista.

Il Processo di Boot-Chaining: Generalità

L'accensione o riavvio di un computer provoca l'esecuzione in automatico di una serie di procedure che, nel caso di un computer standard PC/AT BIOS-based, sono integrate nel BIOS (*Basic I/O System*) contenuto a bordo della scheda madre del computer. Queste procedure sono globalmente indicate con l'acronimo POST (*Power On Self Test*). Oltre a verificare la presenza di alcuni componenti hardware necessari per l'avvio ed il corretto funzionamento di un computer (e.g.: RAM, tastiera, scheda grafica, ecc.), l'obiettivo finale del POST è di identificare sul disco di boot la struttura dati chiamata MBR (*Master Boot Record*) e trasferire il controllo al codice in essa contenuto.

In casi particolari è possibile predisporre un computer installando diversi sistemi operativi (Microsoft o non) selezionabili all'avvio. Configurazioni di questo tipo vengono spesso identificate come *dual-boot* o *multi-boot* a seconda del numero di sistemi operativi disponibili.

Alcuni esempi di applicazioni *boot loader* sono:

- Nel caso di sistemi operativi Microsoft:

- IO.SYS/MSDOS.SYS: per sistemi operativi Microsoft MS-DOS o Windows 3.x/9x.
- NTLDR (da notare che il file non possiede alcuna estensione): per sistemi operativi Windows NT/2000/XP/2003/2003-R2.
- BOOTMGR (da notare che il file non possiede alcuna estensione): per sistemi operativi Windows Vista, Windows 7, Windows Server 2008/2008-R2.

OS Bootstrap Loader e compatibilità con diversi sistemi operativi Microsoft

In caso di coabitazione di più sistemi operativi Microsoft sullo stesso computer (configurazione multi-boot) è bene prestare attenzione alla compatibilità tra i differenti programmi di boot loader. Infatti, i vecchi loader dei sistemi operativi Windows 9x non sono capaci nè di gestire l'avvio di un sistema operativo successivo, nè tanto meno di gestire una lista di sistemi operativi selezionabili all'avvio del computer.

Pertanto è bene attenersi alla sempre valida regola di installare i sistemi operativi procedendo dal più vecchio al più nuovo (e ciò anche a parità di file di boot (e.g.: Windows NT/2000/XP/2003/2003-R2*) in quanto i file loader più aggiornati potranno avere delle nuove funzionalità o semplicemente aver risolto problemi presenti nelle precedenti versioni, ed essere comunque in grado di avviare sistemi operativi precedenti).*

- Nel caso di sistemi operativi GNU Linux:
  - LILO.
  - GRUB.

In generale, come già detto precedentemente, qualunque sia il sistema operativo installato su un computer, il BIOS ha il compito di scatenare il processo di *bootstrap* del sistema operativo. La fase principale di questo processo consiste nell'identificare quella porzione di informazioni contenuta nel primo settore del disco di boot e chiamata *Master Boot Record* (MBR).

Il MBR contiene la tabella delle partizioni (*Partition Table*) ed il codice necessario per effettuare il boot del sistema operativo. L'obiettivo di questo codice è di rintracciare dalla tabella delle partizioni la partizione attiva e passare il controllo al *boot sector* contenuto nella partizione attiva (*Partition Boot Record* (PBR)).

Il PBR è responsabile del caricamento del sistema operativo; tale operazione necessita la ricerca ed il caricamento in memoria del relativo *OS Bootstrap Loader* o semplicemente *boot loader* (a volte indicato anche come IPL, *Initial o Interactive Program Loader*).

Il MBR possiede una dimensione standard di 512 bytes ed è così strutturato:

- 446 bytes: programma di *bootstrap* del SO (*OS Bootstrap Loader*)
- 64 bytes: *partition table* (max 4 entry)
- 2 bytes: *checksum* per la verifica di integrità del MBR

Una volta ottenuto il controllo del processo di boot dal MBR, il programma di *bootstrap* contenuto nel *boot sector* della partizione attiva – se previsto tra le proprie funzionalità – permette di scegliere quale sistema operativo avviare presentando a video una lista. Tutti i

programmi prima menzionati – tranne quelli per MS-DOS e Win9x – offrono la possibilità di gestire una lista di sistemi operativi avviabili.

Nei nuovi computer che utilizzano un firmware di tipo EFI (*Extensible Firmware Interface*), il processo di boot si svolge in modo completamente differente rispetto a vecchi computer PC/AT BIOS-based. In tal caso il firmware EFI contiene già un codice di *boot manager* il quale determina l'avvio del programma di boot loader del sistema operativo selezionato in base alla configurazione specificata in apposite variabili contenute in una porzione di RAM non-volatile (NVRAM). Nel caso dei sistemi operativi Windows Vista, Windows 7, Windows Server 2008/2008-R2 installati su computer con firmware EFI, lo scopo del database BCD (*Boot Configuration Data*) è di predisporre e gestire le variabili necessarie per la configurazione NVRAM.

Il Processo di Boot-Chaining di un sistema operativo Microsoft precedente a Windows Vista

Il programma che determina l'avvio di un sistema operativo Windows NT/2000/XP/2003/2003-R2 si chiama NTLDR (ovvero *NT Loader*) e si trova nella directory di root della partizione attiva di sistema, ovvero quella contenente i file di boot del sistema operativo; da notare che il file NTLDR non ha nessuna estensione.

Partizione di Sistema (Boot Partition) vs Partizione di Boot (System Partition)

Secondo terminologia Microsoft la differenza tra partizione di sistema e di boot è la seguente:

- *System Partition: è la partizione primaria attiva che contiene i file di boot; ovvero:*
  - *nel caso Windows NT/2000/XP/2003: ntldr, boot.ini e ntdetect.com (oltre eventualmente ai file bootsect.dos e ntdd.sys);*
  - *nel caso di Windows Vista/7, Windows Server 2008/2008 R2: bootmgr ed il database BCD contenente la configurazione di boot.*

- *Boot Partition: è la partizione primaria (o unità logica di una partizione estesa) che contiene i file di sistema, ovvero la directory di installazione identificata dalla variabile d'ambiente %SystemRoot%.*

Per identificare le suddette partizioni è necessario utilizzare la console grafica DiskMgmt.msc oppure la utility DiskPart.exe (con i seguenti comandi: list disk; select disk 0; detail disk) da una console a prompt di comando (CLI, Command Line Interface) con privilegi elevati.

L'obiettivo del programma NTLDR è di predisporre l'ambiente di esecuzione per poter caricare in memoria ed eseguire il Kernel del sistema operativo (che si trova di default in %SystemRoot%\System32*NTOSKRNL.EXE*). Successivamente il *kernel* predispone il caricamento dei driver per i dispositivi rilevati dalla utility NTDETECT.COM (*Hardware Recognizer*) e prepara le apposite chiavi dei registry (*System Registry Hives*) necessarie

successivamente per poter avviare i sotto-sistemi (di default: SMSS.EXE, Win32K.SYS, Posix, CSRSS.EXE) ed i servizi.

La ricerca del *Kernel* del sistema operativo da parte del loader NTLDR avviene attraverso le indicazioni contenute nel file BOOT.INI (cf. figura 1) che si trova nella directory root della partizione attiva di sistema. Il file BOOT.INI è di tipo "ASCII puro", con una struttura interna tipica di un file .INI, organizzato in due sezioni:

- Sezione [Boot Loader]: contiene le regole di boot ovvero il tempo di attesa (contenuto nella variabile *timeout*) in secondi dopo il quale avviare il sistema operativo definito come default (contenuto nella variabile *default*).
- Sezione [Operating Systems]: contiene la lista dei sistemi operativi installati e selezionabili al momento dell'avvio o riavvio del computer. Per ciascuno di essi deve esistere una riga con le opzioni necessarie per personalizzare il suo avvio ed il relativo comportamento. Da notare che la lista non viene visualizzata nel caso in cui sia installato un solo sistema operativo oppure se la variabile *timeout* è impostata a 0.

Per la identificazione del *kernel* del sistema operativo da avviare viene utilizzata all'interno del file BOOT.INI la convenzione ARC (*Advanced Risc Computing*) – originariamente definita da Dave Cutler, uno degli architetti di Windows NT – come di seguito indicato:

controller(n),disk(n),rdisk(n),partition(n)\cartella="Descrizione SO da Avviare" [opzioni]

- Esempio:

```
[boot loader]
timeout=15
default=multi(0)disk(0)rdisk(0)partition(2)\Win2K3
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\Windows="WinXP-Pro-SP3" /fastdetect /noexecute=optin
multi(0)disk(0)rdisk(0)partition(2)\Win2K3="WS03-EE (ie-mi-dc-01.isoleeolie.org)" /fastdetect
```

La configurazione contenuta nel file BOOT.INI sopra indicato, determina dopo 15 secondi di attesa, l'avviamento in automatico (default=15) del sistema operativo Windows Server 2003 dalla cartella \Win2K3 presente sulla seconda partizione (n=2), del primo disco (n=0), connesso al primo controller (n=0).

Come si può notare la personalizzazione del percorso di installazione del sistema operativo può arrivare fino a livello di cartella contenente i file di sistema (e.g.: C:\Win2K3). Una volta avviato il sistema operativo questo percorso è salvato nella variabile di ambiente %SystemRoot% (o %windir%) inizializzata allo startup del computer ed all'avvio del sistema operativo (ancora prima del logon di un utente).

Nel caso in cui nel file BOOT.INI sia presente un solo sistema operativo avviabile (oppure se il parametro *timeout* è impostato a 0 (zero)) non viene presentato nessun menu di scelta e viene avviato automaticamente l'unico sistema operativo presente (oppure quello impostato come default nel caso in cui *timeout*=0). Viceversa se il valore del parametro *timeout* è uguale a -1, il processo di boot si blocca sul menu di scelta, costringendo l'utente a selezionare comunque un sistema operativo da avviare.

Se il file BOOT.INI non esiste, NTLDR tenta di rintracciare il *kernel* del sistema operativo tenendo conto del contesto dal quale esso trae origini (ovvero a partire da quale sistema operativo è stato installato o copiato) e dei parametri di installazione di default previsti in tale contesto:

- Nel caso di NTLDR di WinNT/2000: viene ricercato il kernel nella directory che di default è previsto si chiami \Winnt.
- Nel caso di NTLDR di WinXP/2003/2003-R2: viene ricercato il kernel nella directory che di default è previsto si chiami \Windows.

Pertanto, qualora in fase di installazione del sistema operativo non sia stato modificato il nome della cartella di installazione di default (ad esempio nel caso di Windows XP si è confermato il nome standard C:\WINDOWS), pur essendo visualizzato un messaggio di errore del tipo "*Invalid BOOT.INI File. Booting From C:\WINDOWS*", il sistema operativo viene avviato correttamente; viceversa, in caso di modifica del nome di default della cartella di sistema, il programma di NTLDR visualizza l'errore, bloccando il processo di boot.

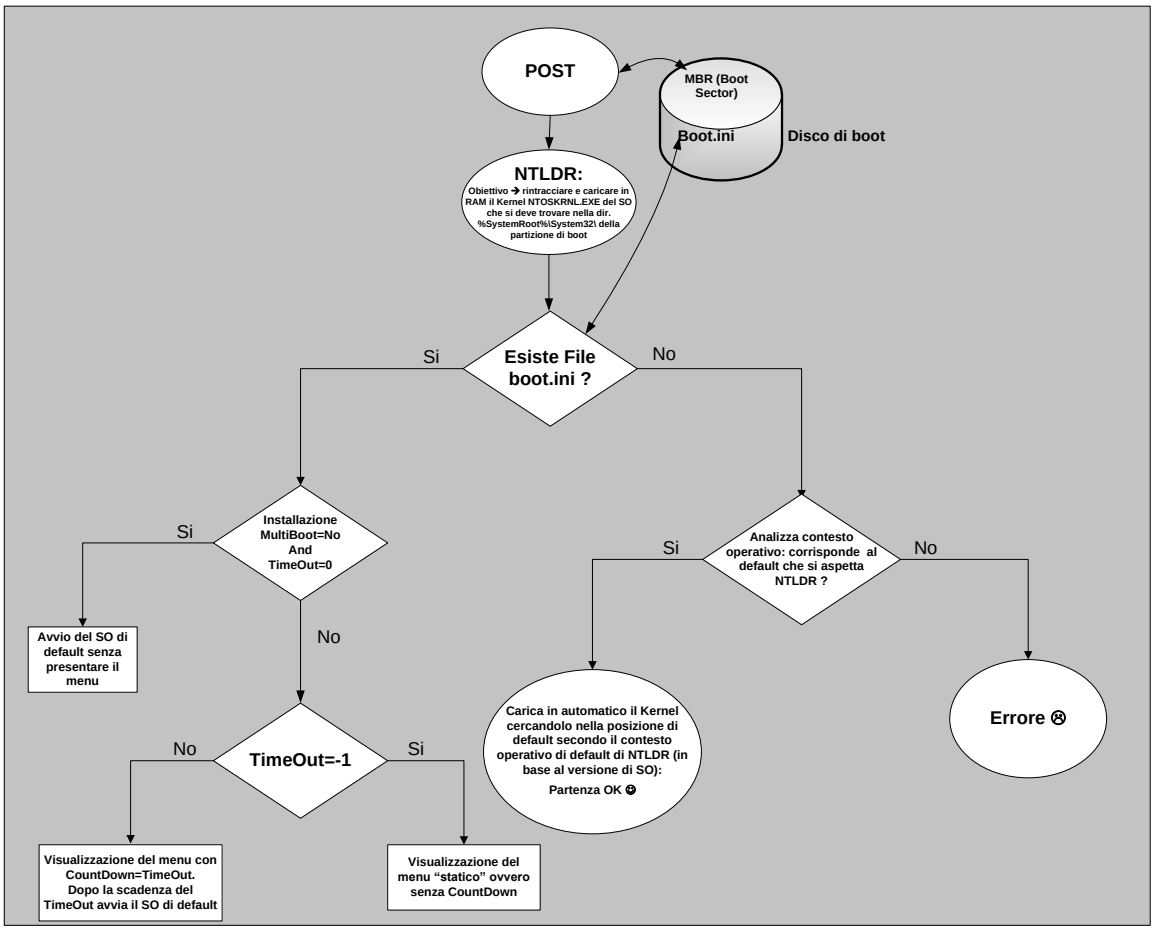

Figura 1: Diagramma di flusso dell'avvio di un sistema operativo precedente a Windows Vista

La gestione del file *boot.ini* può essere effettuata o tramite un qualsiasi editor ASCII (e.g.: Notepad, Ultra Edit, VIM, ecc.) oppure tramite la utility da una CLI *bootcfg.exe*, oppure indirettamente dalle proprietà del *Computer* (*Advanced, Startup & Recovery, Edit*), tramite l'*applet* del Control Panel *sysdm.cpl*.

Creazione di un floppy disk di startup per un computer con sistema operativo Windows NT/2000/XP/2003/2003-R2

Una buona abitudine è quella di "corredare" ogni computer Windows NT/2000/XP/2003/2003-R2 di un floppy disk di startup (FDS) o quanto meno crearne uno e poi, tenendo conto della diversità di configurazione, adattarlo alle varie situazioni.

Un FDS deve essere creato nel modo seguente:

- *Formattare un floppy disk su un computer dotato di sistema operativo Windows NT/2000/XP/2003/2003-R2.*
- *Copiare i file di boot necessari per il sistema operativo:*
  - *Ntldr*
  - *Boot.ini*
  - *Ntdetect.com*
  - *Ed eventualmente il file Ntbootdd.sys nel caso si disponga di controller SCSI.*

Il FDS può essere utile per risolvere situazione del genere:

- *Danneggiamenti al boot sector e/o MBR del disco che ne compromettono l'avvio.*
- *Infezioni da virus.*
- *Mancanza o danneggiamento di uno dei file di boot del sistema operativo (i.e.: ntldr, boot.ini o ntdetect.com).*
- *Effettuare il boot da un membro di un set di dischi in mirroring software in seguito al danneggiamento del disco master.*

E' bene prestare attenzione al fatto che, eseguendo la formattazione del floppy disk su un computer con sistema operativo precedente a WinNT (e.g.: Win98) il comando format di questo sistema operativo scriverà sul boot sector del floppy disk che il boot loader da invocare è MSDOS.SYS (e *IO.SYS*)*. Non essendo questi tra i file copiati sul FDS, al riavvio del computer verrà visualizzato il classico errore "Disco non di sistema" oppure in inglese "Non System Disk".*

Il nuovo Processo di Boot-Chaining di un sistema operativo Microsoft a partire da Windows Vista

A partire da Windows Vista il processo di boot è stato modificato rispetto ai sistemi operativi precedenti. Infatti, il *bootstrap* del SO (*OS Bootstrap Loader*) contenuto nel MBR non avvia più direttamente il loader del sistema operativo (e.g.: NTLDR nel caso di WinNT/2000/XP/2003) ma carica un nuovo esemplare di *Boot Manager* costitutito dal programma *bootmgr* contenuto nella cartella di root della partizione attiva di sistema; da notare che il file *bootmgr* non ha nessuna estensione.

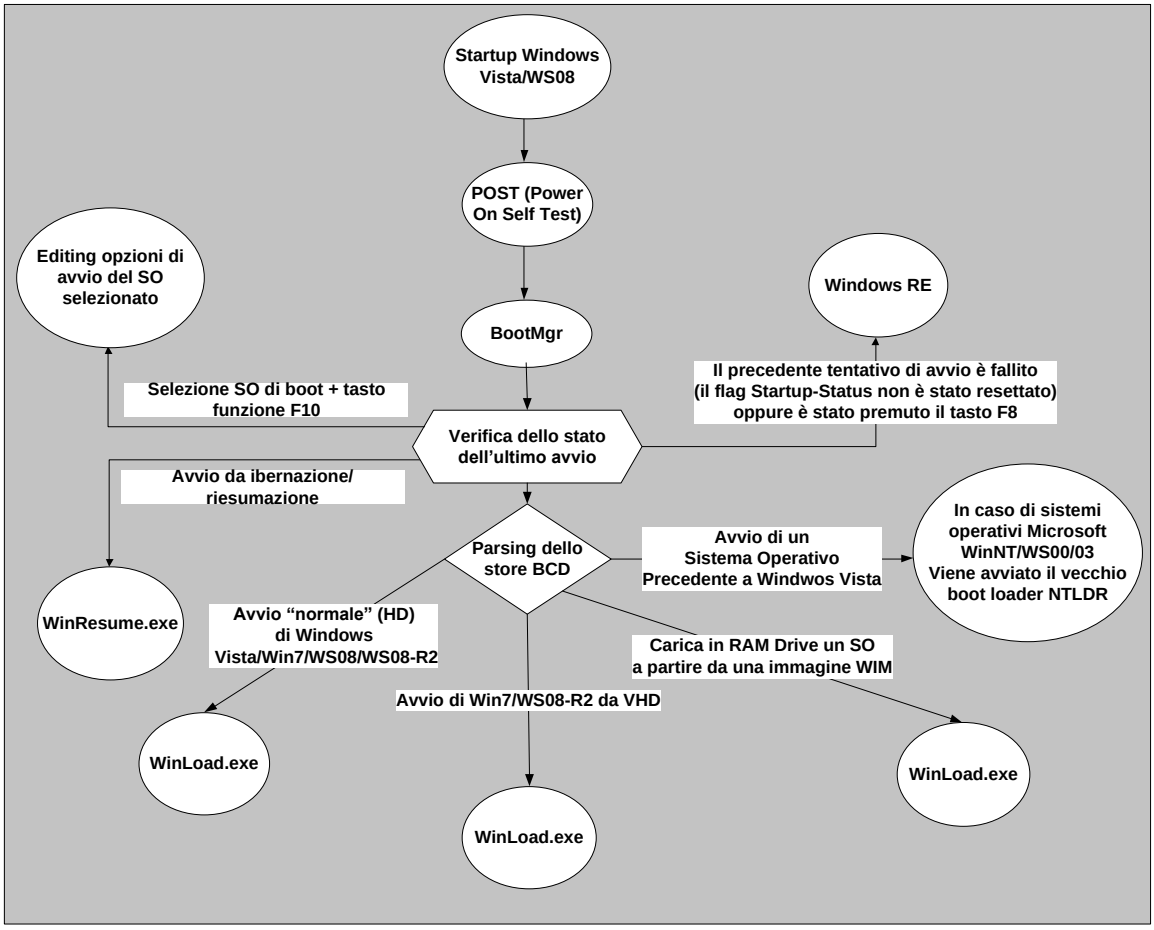

Figura 2: Diagramma di flusso dell'avvio di un sistema operativo a partire da Windows Vista

Come di può dedurre dalla figura 2, una volta ottenuto il controllo del processo di boot, il nuovo *bootmgr* effettua alcuni controlli per identificare in che modo avviare il sistema operativo:

- Innanzitutto verifica lo stato di chiusura della precedente sessione (startup precedente), per accertarsi che esso sia andato a buon fine. A tale scopo viene analizzato il contenuto del bit/flag *Startup-Status*. Nel caso in cui il tentativo di avvio precedente non sia andato a buon fine, viene avviato in automatico l'ambiente di ripristino *Windows Repair* (WinRE).
- Successivamente verifica se l'ultimo spegnimento è stato fatto attraverso una operazione di ibernazione o *stand-by*: in tal caso procede con l'avvio di un particolare loader identificato dall'applicazione %SystemRoot\System32\WinResume.exe.
- Viceversa nel caso di avvio/riavvio "normale" di Windows Vista, Windows 7, Windows Server 2008/2008-R2, viene caricato il nuovo loader identificato dall'applicazione %SystemRoot\System32\WinLoad.exe. Da notare che a partire da Windows 7 (solo per le versioni Ultimate e Enterprise) e Windows Server 2008 R2 (tutte le versioni), il nuovo bootmgr è capace di avviare un sistema operativo anche a partire da un disco virtuale VHD.
- In caso di configurazioni di tipo multi-boot ed in seguito alla scelta di avviare un sistema operativo Microsoft precedente a Windows Vista (e.g.: WinNT/2000/XP/2003/2003-R2), il

programma *bootmgr* trasferisce il controllo al vecchio loader NTLDR. A questo punto possono essere gestiti anche situazioni di *dual*/*multi-boot* tra vecchi sistemi operativi, sempre sotto il controllo del loader NTLDR utilizzando il vecchio file BOOT.INI. In tal caso si viene a determinare una sorta di catena di boot loader con differenti menu di boot: un primo menu gestito da *bootmgr* ed un successivo menu gestito da *ntldr*.

Osservando ancora la figura 2, si può notare come le informazioni relative alla modalità di boot (e.g.: *timeout* di attesa e/o sistema operativo da caricare di default) e la lista dei sistemi operativi disponibili sono contenuti in un nuovo database o *data store* identificato come *Boot Configuration Data* (BCD).

Il database Boot Configuration Data (BCD)

In generale il BCD identifica un database avente una struttura logica architettura (*namespace*), pensata per ospitare un insieme di oggetti all'interno di speciali contenitori chiamati *Boot Configuration Data Store*.

I database BCD possono essere di due tipi:

- **System Store**: per ogni computer equipaggiato con sistema operativo Windows Vista o successivo, può esistere uno ed un solo *System Store*. Esso viene creato di default all'atto dell'installazione del sistema operativo e serve per ospitare gli oggetti necessari per gestire il processo di boot (tempistiche, sistema operativo da avviare di default in caso di multi-boot) o l'avvio di altre applicazioni da eseguire allo startup (e.g.: *memory test*, immagine WIM di avvio caricata in RAM, ecc.).

  Lo store di sistema del BCD prende il posto del vecchio file di testo BOOT.INI. Contrariamente a quest'ultimo esso non è in disponibile come file in formato ASCII direttamente editabile ma è disponibile solamente in formato binario. Tale file ha nome BCD (senza nessuna estensione) ed è contenuto nella cartella nascosta \boot contenuta nella partizione attiva di sistema (cf. figura 3).

Come identificare la partizione contenente i nuovi file di boot, bootmgr e BCD ?

Per rilevare la partizione primaria attiva contenente i file di boot (bootmgr e BCD) è possibile utilizzare la console DiskMgmt.msc oppure la utility DiskPart.exe (ed i comandi seguenti: list disk; select disk 0; detail disk); in entrambi i casi sono richiesti privilegi amministrativi.

Da notare che a partire da Windows 7 e Windows Server 2008 R2, di default viene creata una partizione nascosta di sistema e attiva contenente i file di boot; essa viene identificata di solito come "System Reserved".

Utilizzando il comando bcdedit.exe, il volume contenente i file di boot è indicato come \Device\HarddiskVolume1 attraverso la proprietà "device" del record "Windows Boot

Manager" del database BCD:

device partition=\Device\HarddiskVolume2

Le suddette informazioni relative al volume contenente i file di boot sono anche salvate nella variabile "\Registry\Machine\BCD00000000" nel seguente hive dei Registry:

HKLM\System\CurrentControlSet\Control\Hivelist.

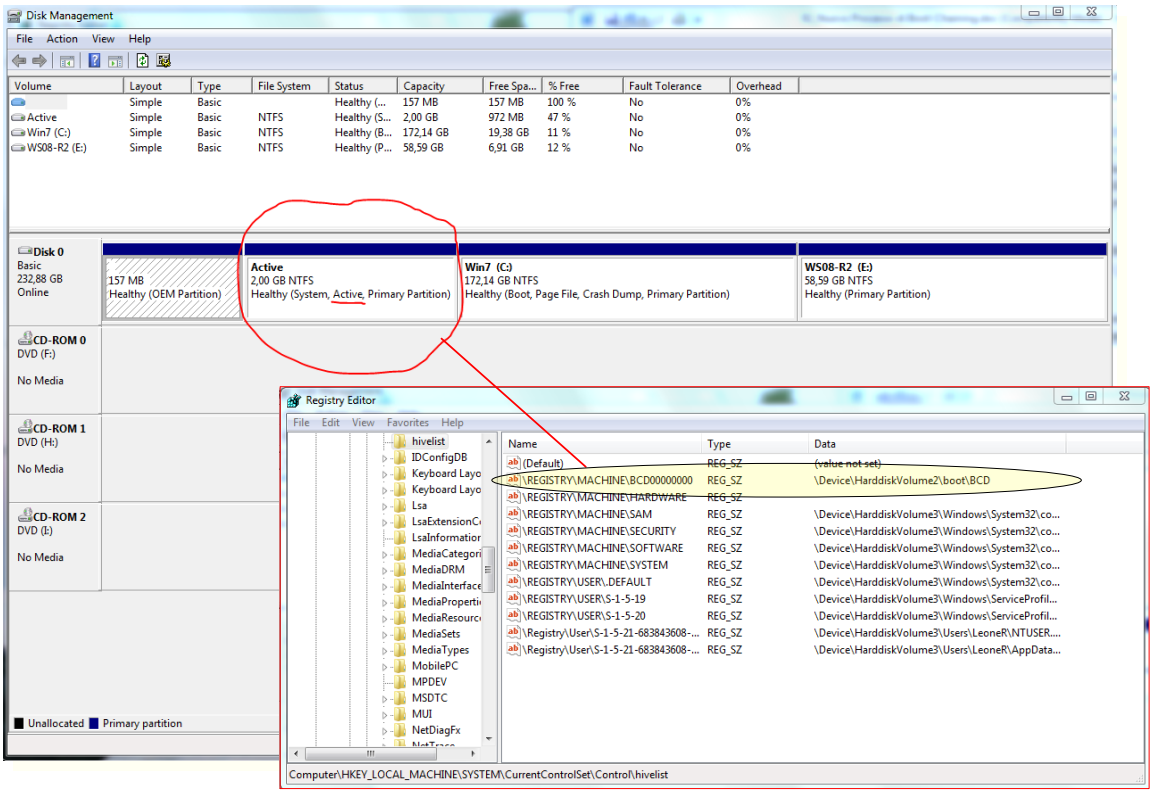

Figura 3: Identificare la partizione attiva tramite DiskMgmt.msc e/o Registry

All'avvio del sistema operativo, il contenuto del file BCD viene caricato in un *hive* dei registry del sistema operativo in *HKLM\BCD00000000*. E' bene osservare che non è possibile editare gli oggetti del BCD direttamente dai registry.

In caso di computer basati su hardware EFI (*Extensible Firmware Interface*) il *system store* BCD è contenuto nella cartella \EFI\Microsoft\Boot della partizione di sistema EFI identificata come ESP (*Efi System Partition*).

- **Non-System Store**: all'occorrenza possono essere creati da un amministratore altri store BCD non di sistema, per scopi di recovery, repair e imaging da eseguire in RAM. Per creare

uno store BCD non di sistema è possibile utilizzare il comando BcdEdit.exe con l'opzione /CreateStore; ad esempio: bcdedit /CreateStore c:\Data-Store\Bcd.

Per la gestione del database BCD è disponibile nativamente l'utility di sistema *bcdedit.exe*, la quale richiede privilegi amministrativi, anche solo per la semplice lettura.

L'architettura del database BCD di sistema

Dal punto di vista logico, uno store di sistema BCD corrisponde ad un *namespace* all'interno della quale sono organizzati in modo gerarchico gli oggetti o *entry* BCD; alcuni esempi di strutture gerarchiche BCD sono indicate nelle figure 2 e 3.

Gli oggetti contenuti nello store di sistema BCD sono utilizzati per scopi di boot di un sistema operativo o per avviare applicazioni diagnostiche (e.g.: memory test o immagini di boot (e.g.: WinPE) da eseguire come RAM-Disk e selezionabili all'avvio di un computer).

Ogni oggetto di uno store BCD è identificato da un insieme di attributi o elementi. Uno di questi attributi, chiamato *identifier* o semplicemente *id*, viene utilizzato come "chiave primaria" per identificare univocamente ciascuno oggetto all'interno della gerarchia o *namespace* BCD. Questa chiave può essere espressa sia in forma di GUID (*Globally Unique Identifier*). Ogni GUID è lungo 128 bit ed assume la forma seguente (da notare che i caratteri '-', '{' e '}' sono obbligatori):

{xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx}

Per semplicità è possibile utilizzare al posto dei GUID delle parole chiave predefinite o *well known name*.

In entrambi i casi il GUID o la parola chiave, devono essere incluse all'interno di parentesi graffe come indicato negli esempi seguenti:

- Esempio di *identifier* espresso tramite GUID: {51188dde-7ddf-11db-921e-005056c00008}.
- Esempio di *identifier* espresso tramite parola chiave: {bootmgr}, {ntldr}, {current}, {default}, {ramdiskoptions}.

Per identificare tutte le parole chiave predefinite è possibile utilizzare il comando seguente: "BcdEdit -id -?".

Ad ogni oggetto di un database BCD è possibile associare anche una breve descrizione (32 bit) tramite l'attributo *description*; a tal proposito è possibile utilizzare l'opzione /d in fase di creazione dell'oggetto oppure successivamente utilizzando l'opzione /set del comando BcdEdit indicando l'identificatore GUID dell'oggetto di cui si vuole modificare la descrizione).

Per visualizzare il contenuto del BCD è possibile inserire il comando BcdEdit.exe (o BcdEdit /v, per visualizzare il GUID anziché le chiavi well-known) da una shell a linea di comando con privilegi elevati.

Privilegi necessari per l'accesso al BCD

Da notare che operando in un contesto con il sistema UAC (User Account Control) abilitato (default a partire da Windows Vista) ed utilizzando un account sottoposto al suo controllo (di default, qualsiasi utente membro del gruppo locale Administrators di un computer stand-alone/workgroup o membro di un dominio oppure membro del gruppo globale Domain Admins che sia diverso dall'account nativo/built-in Administrator), per poter editare o anche semplicemente consultare il contenuto dello store BCD, è necessario aprire una CLI richiedendo esplicititamente la elevazione dei privilegi, cliccando con il tasto destro del mouse sulla icona CMD (Command Prompt) e selezionando "Run as Administrator".

Di seguito vengono riportate alcune entry visualizzate dal comando bcdedit:

- Oggetto Windows Boot Manager:

Identifier	**{bootmgr}**
device	partition=C:
description	Windows Boot Manager
locale	en-US
inherit	{globalsettings}
default	{default}
displayorder	{current}
	{default}
	{51188dde-7ddf-11db-921e-005056c00008}
Toolsdisplayorder	{memdiag}
Timeout	15

- Oggetto Windows Boot Loader:

Identifier	**{current}**
Device	partition=C:
Path	\Windows\system32\winload.exe
Description	Microsoft Windows Vista
Locale	en-US
Inherit	{bootloadersettings}
Osdevice	partition=C:
Systemroot	\Windows
Resumeobject	{3e099733-7cae-11db-82cb-a75cd2b9002e}
Nx	OptIn

- Oggetto Windows Legacy OS Loader:

Identifier	**{default}**
Device	partition=C:
Path	\ntldr
Description	Windows Server 2003, Enterprise

- Oggetto Windows Boot Loader:

Identifier	**{51188dde-7ddf-11db-921e-005056c00008}**
Device	partition=C:
Path	\Windows\system32\winload.exe
Description	Windows Vista No-DEP
Locale	en-US
Inherit	{bootloadersettings}
Osdevice	partition=C:
Systemroot	\Windows

Resumeobject	{3e099733-7cae-11db-82cb-a75cd2b9002e}
Nx	AlwaysOff
Vga	No
Sos	Yes

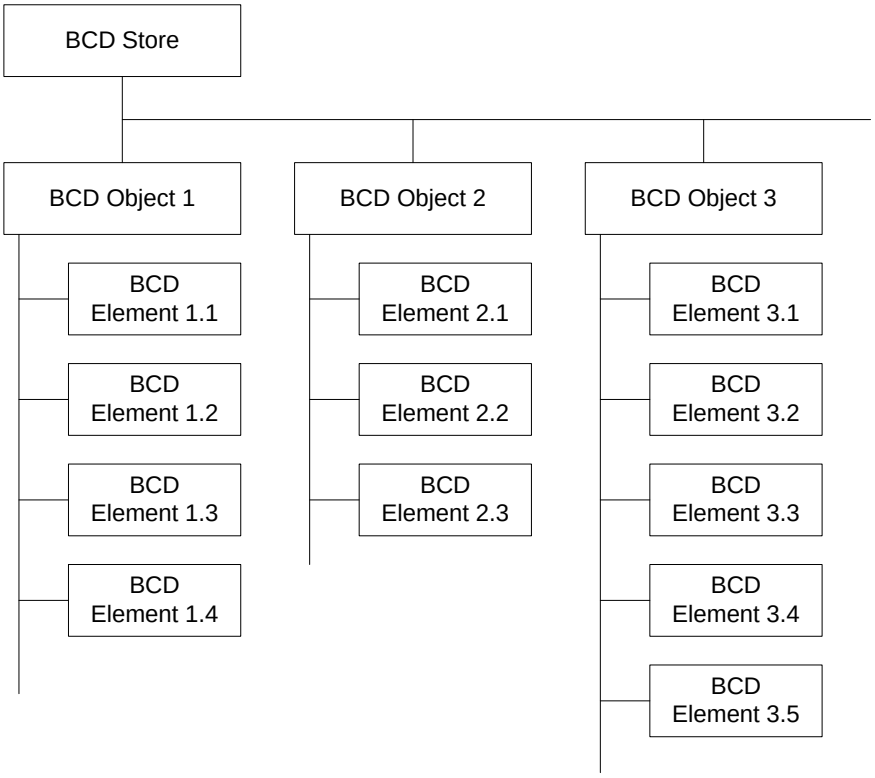

Figura 4: Struttura gerarchica di uno store BCD

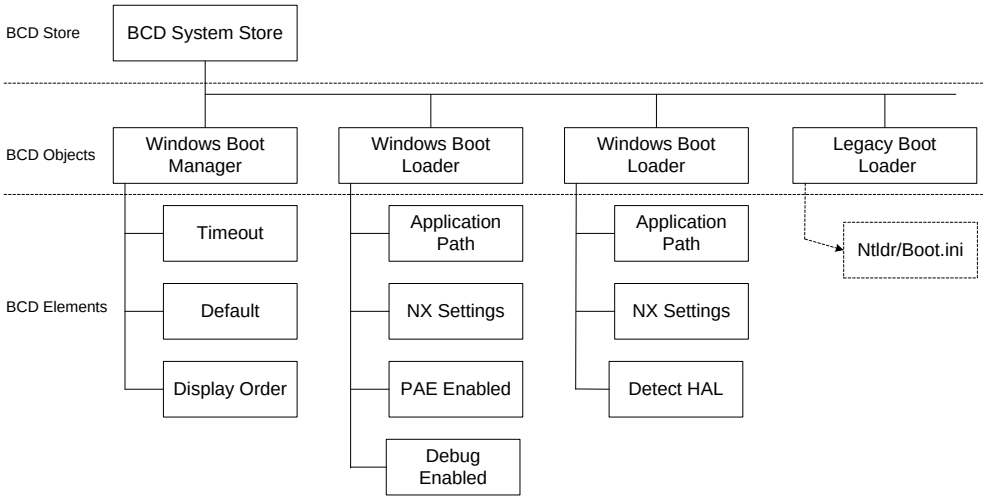

Figura 5: Store BCD di sistema

Nella figura 4 è indicata la relazione tra gli elementi di un file Boot.ini e gli oggetti ed elementi che caratterizzano uno store di sistema BCD.

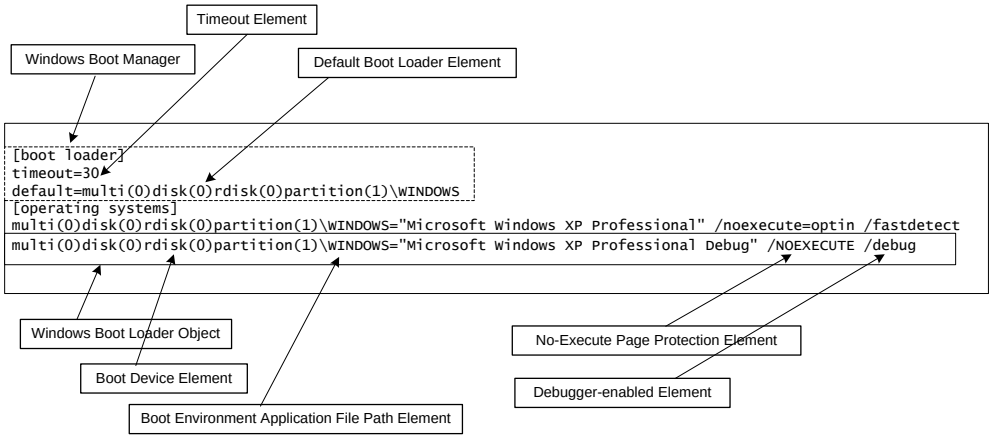

Figura 6: Relazione tra Boot.ini e BCD

Gestione del datastore Boot Configuration Data (BCD) attraverso strumenti nativi Microsoft o prodotti da terze parti

Per la manutenzione del BCD sono disponibili diversi strumenti nativi Microsoft, ciascuno dei quali consente di gestire dettagli più o meno avanzati.

- Editor da prompt di comando: %SystemRoot%\System32\BcdEdit.exe.
- Interfaccia grafica MSCONFIG.EXE (corrispondente alla utility *System Configuration* appartenente al gruppo di applicazioni *Administrative Tools*), come indicato in figura 6.

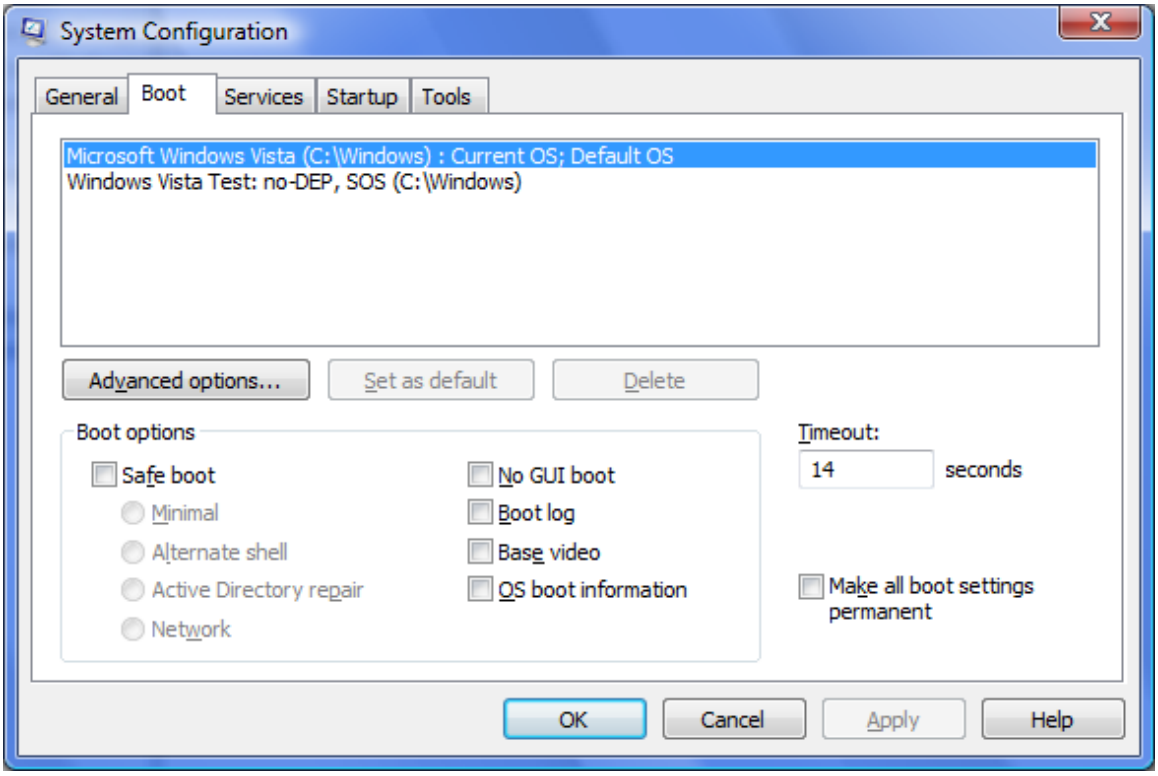

Figura 7: Utility System Configuration (msconfig.exe)

- Libreria di sviluppo BCD WMI API per sviluppare applicazioni specifiche o gestire il BCD anche via VBScript.

- Mediante il pannello di controllo agendo sulle proprietà del computer oppure eseguendo direttamente il corrispondente *applet* tramite il comando *sysdm.cpl*. In questo modo è possibile gestire solamente le impostazioni a livello globale del BCD, ovvero il timeout di attesa ed il sistema operativo (con relative opzioni di configurazione) da caricare come default, come indicato in figura 7.

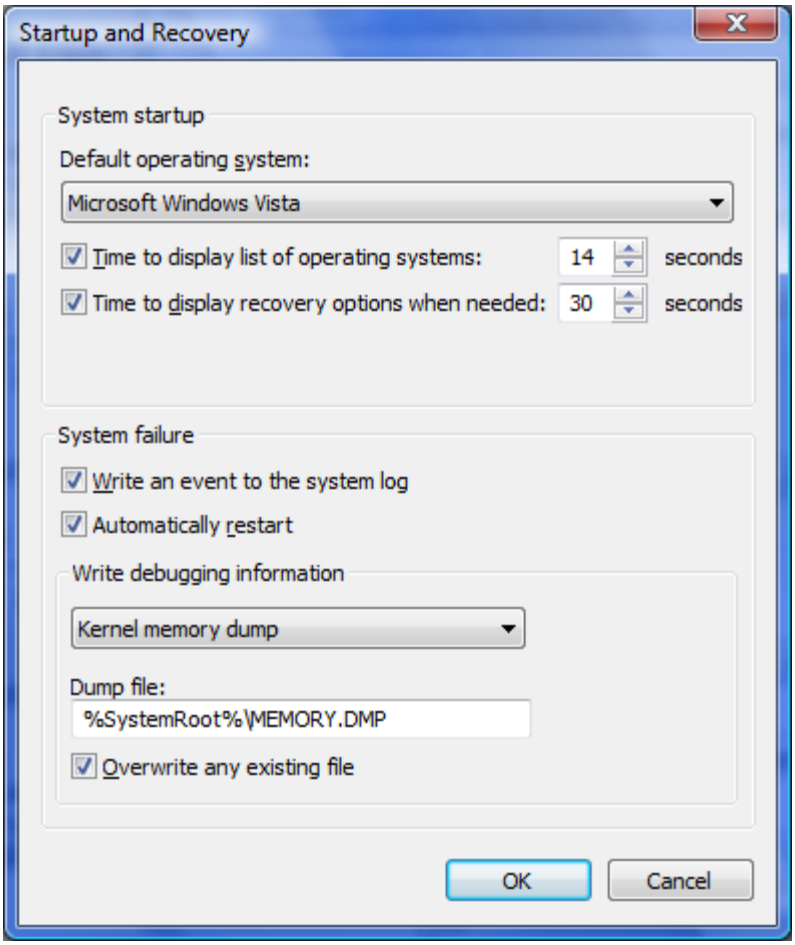

Figura 8: Configurare le opzioni di Startup e Recovery tramite pannello di controllo (sysdm.cpl)

Di seguito sono mostrati alcuni esempi sull'utilizzo del comando *BcdEdit* da eseguire da una CLI con privilegi elevati:

1. Consultare l'help del comando BcdEdit:
   - BcdEdit -?

2. Per avere una descrizione del concetto di "Identifier" ed avere la lista di tutti gli identificatori di tipo "Well-known":
   - BcdEdit -id -?

3. Consultare l'help per una generica opzione del comando BcdEdit:
   - BcdEdit -? <opzione>
   - Oppure BcdEdit <opzione> -?

4. Visualizzare il contenuto dello store BCD:
   - BcdEdit -Enum oppure semplicemente BcdEdit: visualizza le *entry* contenute nello store BCD utilizzando i termini *well-known* (e.g.: {bootmgr}, {current}) al posto dei *GUID*.

- BcdEdit -v: per ciascuna *entry* visualizza il *GUID* anziché i termini *well-known*.

```
Windows Boot Manager
--------------------
identifier              {bootmgr}
device                  partition=C:
description             Windows Boot Manager
locale                  en-US
inherit                 {globalsettings}
default                 {default}
displayorder            {current}
                        {default}
                        {51188dde-7ddf-11db-921e-005056c00008}
toolsdisplayorder       {memdiag}
timeout                 15

Windows Boot Loader
-------------------
identifier              {current}
device                  partition=C:
path                    \Windows\system32\winload.exe
description             Microsoft Windows Vista
locale                  en-US
inherit                 {bootloadersettings}
osdevice                partition=C:
systemroot              \Windows
resumeobject            {3e099733-7cae-11db-82cb-a75cd2b9002e}
nx                      OptIn

Windows Legacy OS Loader
------------------------
identifier              {default}
device                  partition=C:
path                    \ntldr
description             Windows Server 2003, Enterprise

Windows Boot Loader
-------------------
identifier              {51188dde-7ddf-11db-921e-005056c00008}
device                  partition=C:
path                    \Windows\system32\winload.exe
description             Windows Vista No-DEP
locale                  en-US
inherit                 {bootloadersettings}
osdevice                partition=C:
systemroot              \Windows
resumeobject            {3e099733-7cae-11db-82cb-a75cd2b9002e}
nx                      AlwaysOff
vga                     No
sos                     Yes
```

5. Visualizzare tutti gli identificativi *well-known* disponibili:
   - BcdEdit -id -?

6. Duplicare la entry di default (i.e.: quella corrispondente al sistema operativo utilizzato nella sessione di boot corrente): da notare che la entry BCD corrispondente al sistema operativo correntemente in uso è identificata dalla parola chiave {default}; questo comando produce una nuova entry BCD caratterizzata da un nuovo GUID che viene generato automaticamente:
    - BcdEdit -copy {default} -d "Windows Vista No-DEP"
    - Esempio:

    C:\>bcdedit -copy {default} -d "Windows Vista No-DEP"
    The entry was successfully copied to {51188dde-7ddf-11db-921e-005056c00008}.

7. Creazione di una entry per avviare un vecchio sistema operativo WinXP/WS03 il cui file loader ntldr risiede sulla prima partizione (\Device\HarddiskVolume1) primaria attiva nascosta del disco:
    - Bcdedit –create {ntldr} –d "WinXP-Pro SP3"
    - Bcdedit –set {ntldr} device partition=\Device\HarddiskVolume1
    - Bcdedit –set {ntldr} path \ntldr
    - Bcdedit –displayorder {ntldr} –addlast

8. Modificare il SO avviato di default:
    - BcdEdit -Default {GUID-Nuova-Entry-Default}

9. Modificare il timeout di avvio del SO di default:
    - BcdEdit -Timeout 14

10. Cancellare una entry dallo store BCD:
    - BcdEdit -Delete {GUID- Entry-Da-Cancellare} -f

11. Modificare una opzione di una entry dallo store BCD di sistema (default):
    - BcdEdit -Set {GUID- Entry-Da-Modificare} <opzione> <valore>
    - Da notare che l'opzione deve essere specificata così come visualizzata dal commando bcdedit e senza il segno "-" oppure "/" davanti !
    - Alcuni esempi:
        i. BcdEdit -Set {51188dde-7ddf-11db-921e-005056c00008} description "Nuova Descrizione da visualizzare nel menu di boot"
        ii. BcdEdit -Set {51188dde-7ddf-11db-921e-005056c00008} vga yes
        iii. BcdEdit -Set {51188dde-7ddf-11db-921e-005056c00008} nx AlwaysOff
        iv. BcdEdit -Set {51188dde-7ddf-11db-921e-005056c00008} sos on

12. Abilitare il servizio EMS (Emergency Management Service) sul sistema operativo Windows Server 2008 correntemente in uso:
    - Bcdedit -ems {current} on
    - Bcdedit -emssettings EMSPORT:1 EMSBAUDRATE:115200

13. Backup/Export del datastore BCD:

- BcdEdit -Export f:\backup\bcd\bcd-backup
- Per salvare il contenuto del proprio BCD potrebbe essere sufficiente eseguire anche la copia della cartella boot\ contenente i file del datastore. Naturalmente essendo i file utilizzati dal sistema operativo, per poter effettuare la copia si richiede di riavviare il computer con un kit di boot WinPE oppure riavviare con il DVD di installazione di Windows 7, ed aprire una CLI digitando la sequenza Shift + 10 dalla schermata di installazione (cf. figura 15).

14. Import del datastore BCD precedentemente salvato:
    - BcdEdit -Import f:\backup\bcd\bcd-backup

15. Creazione di uno store BCD non di sistema:
    - BcdEdit -createstore c:\Data-Store\Bcd

16. Modificare l'ordine di visualizzazione delle entry contenute nel BCD:
    - BcdEdit -DisplayOrder …
    - Esempio: bcdedit -displayorder {current} {51188dde-7ddf-11db-921e-005056c00008} {51188ddd-7ddf-11db-921e-005056c00008} {51188ddf-7ddf-11db-921e-005056c00008}

Oltre ai comandi nativi Microsoft esistono anche delle applicazioni sviluppate da terze parti che permmettono di gestire lo store BCD anche tramite interfaccia grafica.

Alcune di queste sono:

- EasyBCD

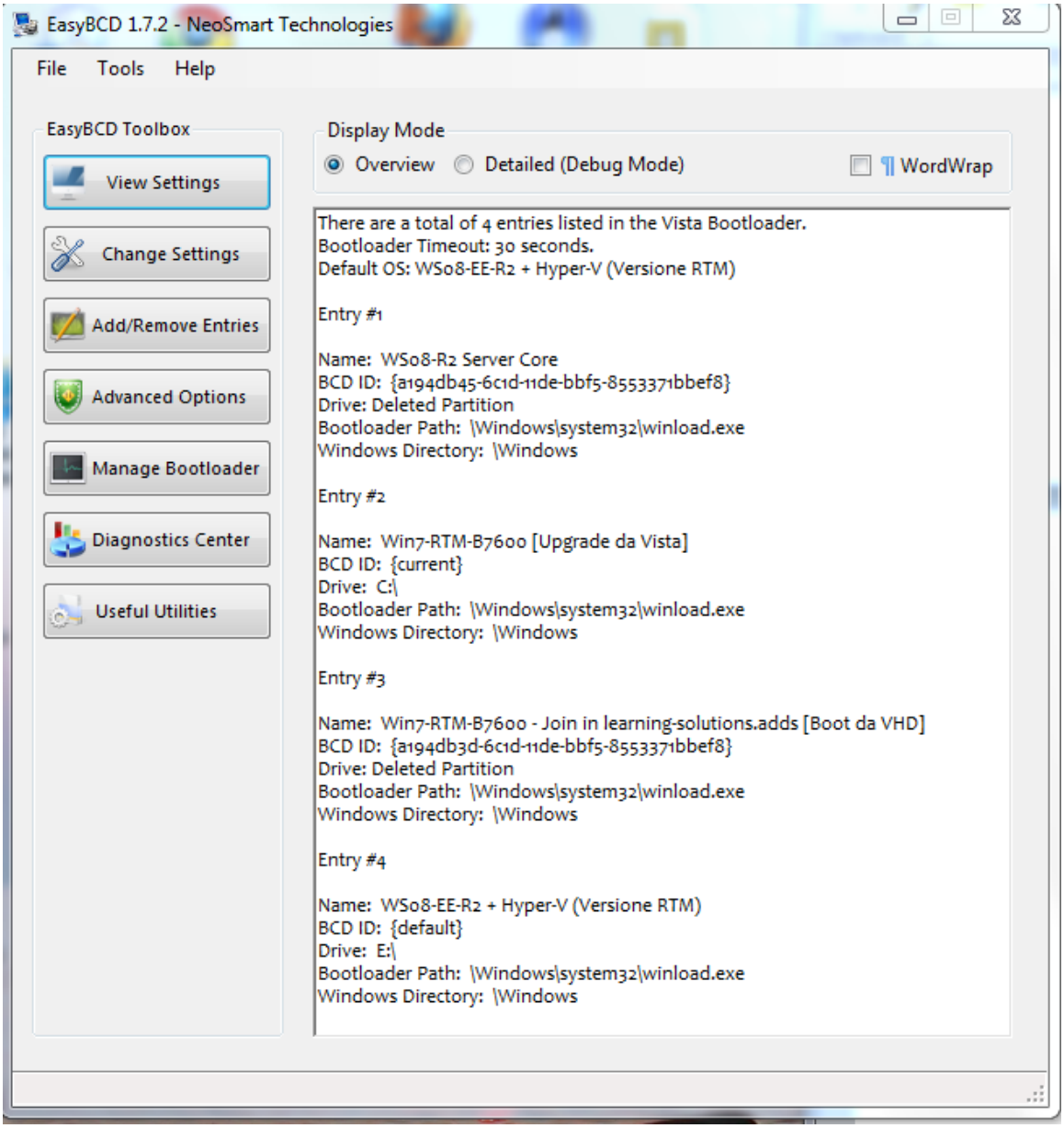

Figura 9: EasyBCD

- VistaBootPro

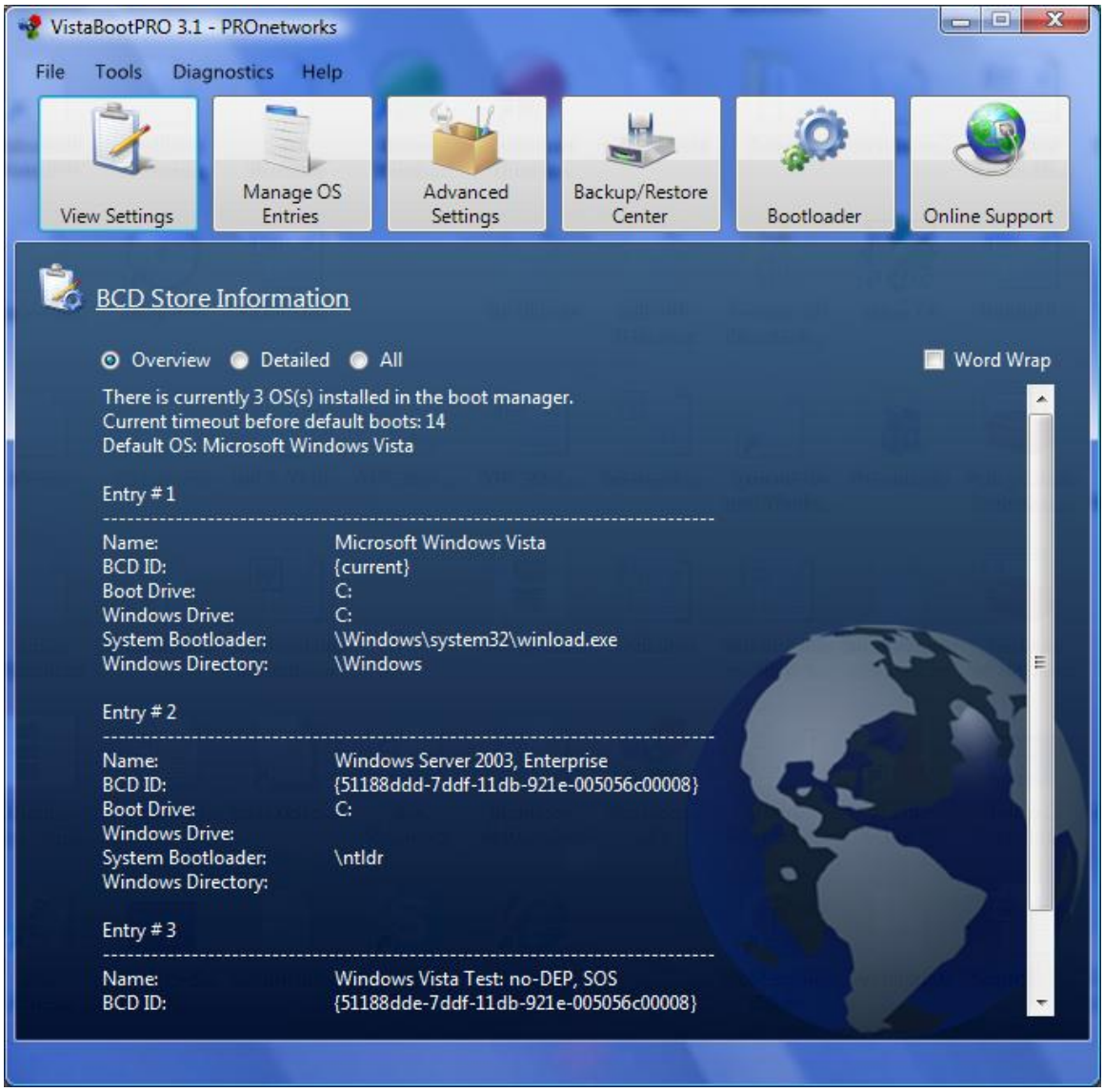

Figura 10: Vista Boot Pro

Operazioni Avanzate sul datastore BCD di sistema

Come esempio di operazione avanzata su uno store di sistema BCD, prendiamo in considerazione la creazione di una entry BCD per eseguire il boot di un computer tramite una immagine WIM di WinPE presente localmente su hard disk. Tale immagine può essere o quella presente di default nel DVD di installazione di Windows Vista (\sources\boot.wim) oppure una appositamente creata e personalizzata tramite il kit WAIK (*Windows Automated Installation Kit*) scaricabile gratuitamente dal sito Microsoft.

A tal proposito si assume che:

- L'unità o drive di boot sia c:\.

- L'immagine WIM sia contenuta in c:\sources\boot.wim.
- Il loader del sistema operativo Vista (o Windows 7) sia c:\windows\system32\winload.exe.
- Si copi il file "boot.sdi" direttamente dal DVD di installazione di Windows Vista o Windows 7 (presente in: \boot\boot.sdi) oppure da un computer sul quale si è installato il kit WAIK (*Windows Automated Installation Kit*) dalla cartella %ProgramFiles%\Windows AIK\Tools\PETools\x86\boot) nella cartella c:\boot (fare attenzione, poichè di default è nascosta).

  Per creare la nuova entry nel BCD di sistema procedere come indicato di seguito:

  - Creare un oggetto BCD di tipo "ramdiskoptions" e definire i suoi attributi; da notare che {ramdiskoptions} è una parola chiave o "well-known":
    i. bcdedit -create {ramdiskoptions} -d "Ramdisk options"
    ii. bcdedit -set {ramdiskoptions} ramdisksdidevice partition=c:
    iii. bcdedit -set {ramdiskoptions} ramdisksdipath \boot\boot.sdi
  - Creare una nuova entry BCD da utilizzare per effettuare il boot del sistema tramite WinPE:
    i. bcdedit -create -d "Windows PE boot" -application OSLOADER.
  - Il commando precedente ritorna un valore di GUID associato alla nuova entry di boot, come ad esempio:

    "The entry {51188ddf-7ddf-11db-921e-005056c00008} was successfully created."

    Nei comandi che seguono, sostituire al posto della stringa NewGUID indicata all'interno delle parentesi graffe il vero GUID ritornato dal comando precedente (mantenendo le parentesi graffe):

    ```
    bcdedit -set {NewGUID} device ramdisk=[c:]\sources\boot.wim,{ramdiskoptions}
    bcdedit -set {NewGUID} path \windows\system32\boot\winload.exe
    bcdedit -set {NewGUID} osdevice ramdisk=[c:]\sources\boot.wim,{ramdiskoptions}
    bcdedit -set {NewGUID} systemroot \windows
    bcdedit -set {NewGUID} winpe yes
    bcdedit -set {NewGUID} detecthal yes
    bcdedit -displayorder {NewGUID} -addlast
    ```

A questo punto, al prossimo reboot del sistema, sarà presente una nuova opzione di boot identificata come "Windows PE boot" attraverso la quale viene effettuato il boot dalla immagine c:\sources\boot.wim

Modificare le opzioni di boot di un sistema operativo in fase di avvio

A partire dal menu visualizzato dal nuovo boot manager *bootmgr* (cf. figura 11) è possibile modificare o aggiungere una delle opzioni di boot di un sistema operativo disponibile all'avvio, nel modo seguente:

1. Selezionare il sistema operativo del quale si desidera modificare o aggiungere una opzione (cf. figura 11).
2. Digitare il tasto funzione F10 per visualizzare il menu di modifica delle opzioni (cf. figura 12).
3. Inserire le modifiche desiderate utilizzando caratteri in maiuscolo (e.g.: nella figura 12 è stata inserita l'opzione /SOS per abilitare la visualizzazione dei componenti (kernel, HAL, sottosistemi, ecc.) e digitare INVIO.

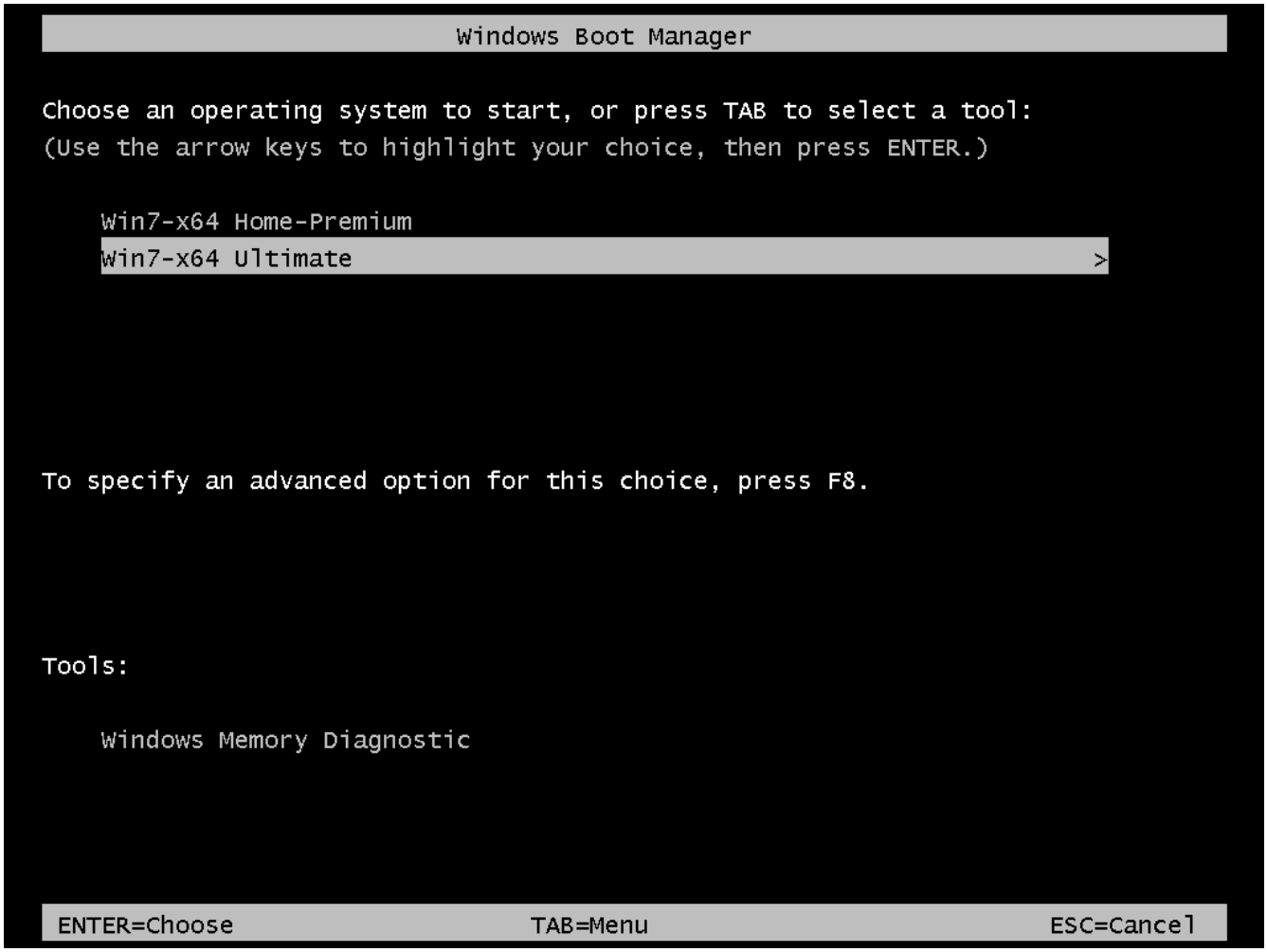

Figura 11: Menu di boot

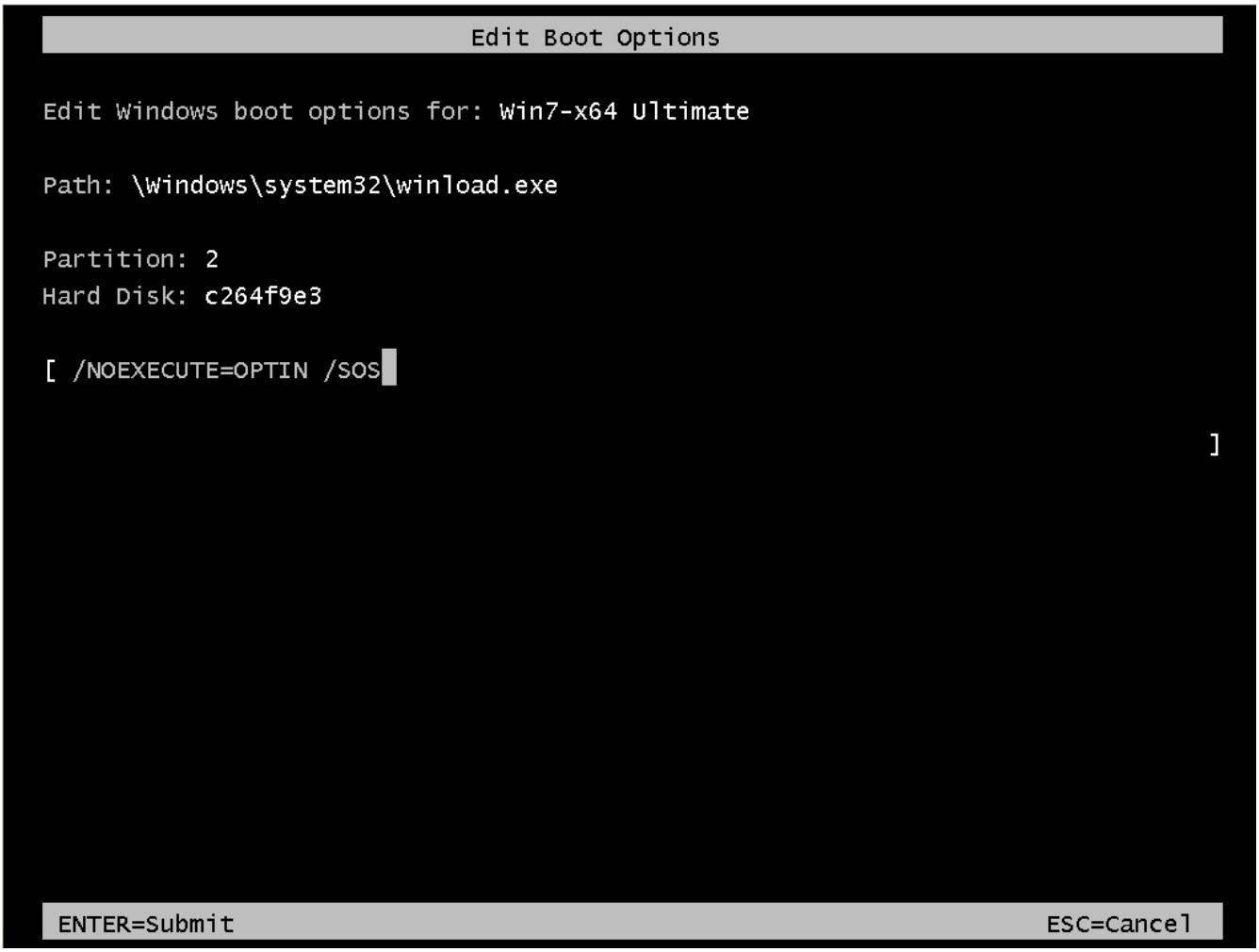

Figura 12: Modifica opzioni di boot (*Edit Boot Options*)

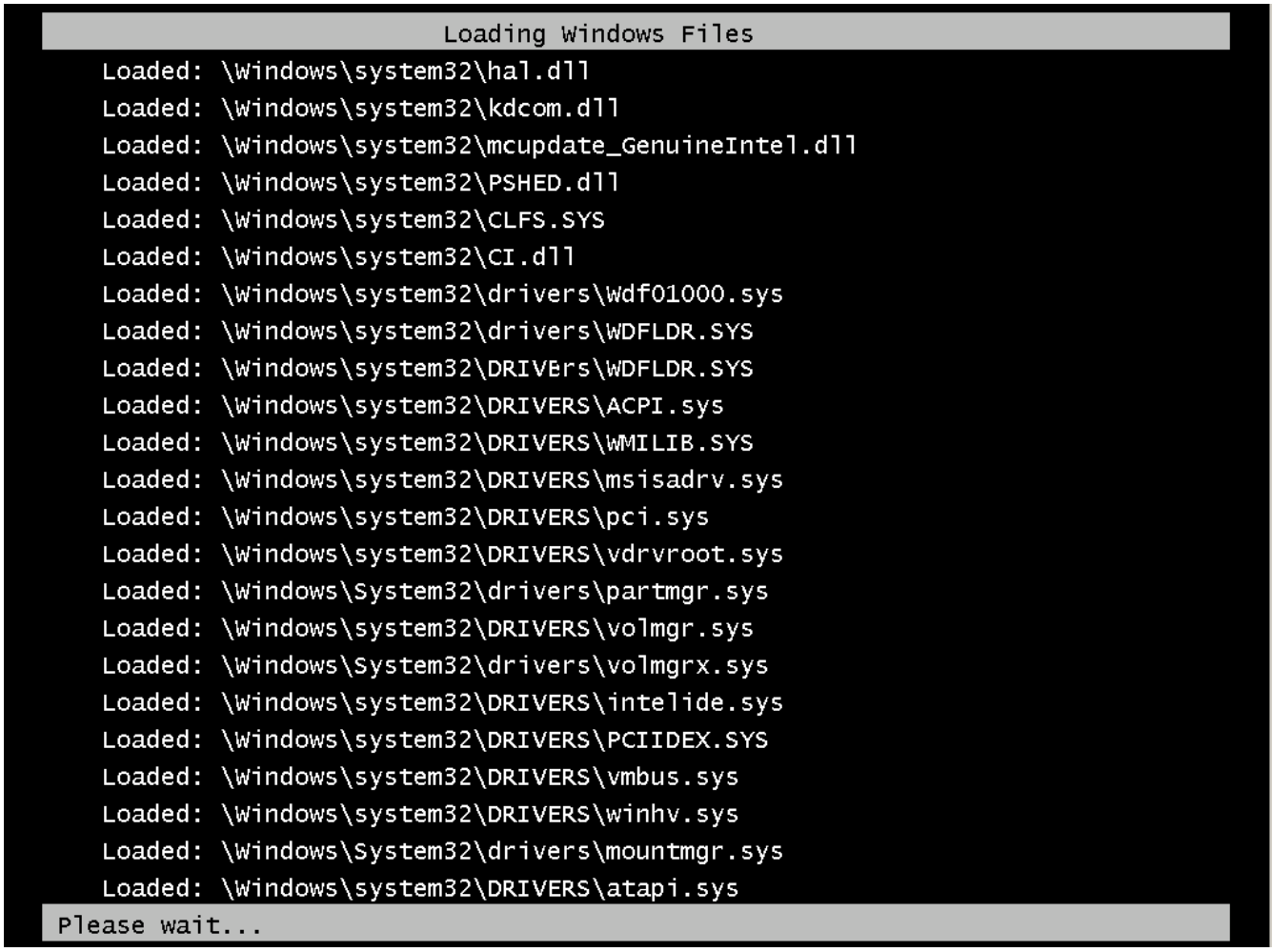

Figura 13: Visualizzazione della lista dei componenti caricati al momento del boot del sistema operativo (/SOS)

Novità riguardanti il nuovo boot manager di Windows 7 e Windows Server 2008 R2

A partire da Windows 7 e Windows Server 2008 R2, il nuovo boot manager (bootmgr) è capace di effettuare il boot di un sistema operativo installato all'interno di un disco virtuale in formato VHD: ovvero un file con estensione .VHD residente su una unità di storage locale (e.g.: [E:]\VHDs\Win7.vhd).

Per quali versioni di sistema operativo è possibile effettuare l'installazione su VHD ?

Il supporto nativo per il boot da VHD è disponibile solo per i seguenti sistemi operativi:

- *Windows Server 2008 R2: tutte le versioni.*

- *Windows 7 Enterprise e Ultimate.*

Per effettuare l'installazione del nuovo sistema operativo su un disco VHD è possibile seguire la seguente procedura:

1. Riavviare il proprio computer dal DVD di Windows 7.
2. Selezionare le opzioni per la localizzazione del sistema operativo (lingua e formato tastiera) come indicato in figura 13 e cliccare sul bottone Next.

Figura 14: Selezionare le opzioni di localizzazione del sistema operativo (Regional Options)

Figura 15: Pagina iniziale per avviare la procedura di installazione

3. Dalla schermata iniziale di avvio della procedura di installazione (cf. figura 14), digitare la sequenza "Shift+F10" per avviare una CLI. In effetti viene avviato il sistema operativo WinPE attraverso il quale viene gestita la prima di fase relativa alla installazione del sistema operativo.

CLI: CMD.EXE vs WinPE

A seconda del sistema operativo che si sta installando la CLI che viene aperta può essere:

- *CMD.EXE: in caso di WinXP o WS03*
- *WinPE: da Windows Vista in poi.*

4. Avviare **DiskPart** (figura 15).

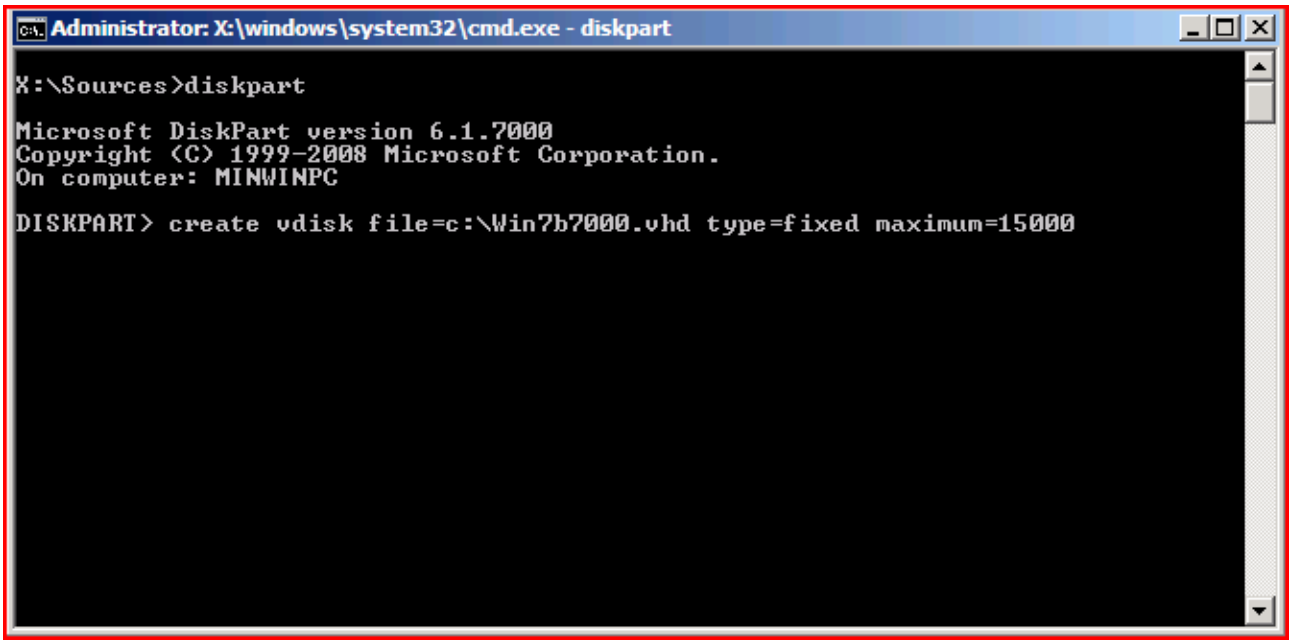

Figura 16: Shell del comando DiskPart

5. Eseguire i comandi seguenti per: a) creare un disco virtuale VHD con dimensioni fisse di almeno 15 GB; b) selezionare il disco VHD appena creato; c) montarlo (eseguire la procedura di attach):
    a. **Create vdisk file=E:\VHDs\Win7.vhd type=fixed maximum=15000**
    b. **select vdisk file=E:\VHDs\Win7.vhd**
    c. **attach vdisk**

Attenzione: creare il disco VHD con dimensione fissa e non dinamica per evitare problemi in fase di setup (e.g.: generazione della schermata "Blue Scren Of Dead" (BSOD)). Da notare che in caso di utilizzo di disco VHD dinamico, ad ogni riavvio esso viene espanso alla massima dimensione.

Le nuove funzionalità di creazione (Create VHD) e di mounting (Attach VHD) dei dischi VHD sono disponibili anche tramite la console grafica Disk Management (diskmgmt.msc) di Windows 7 e Windows Server 2008 R2, come mostrato nella figura 16. Naturalmente, questa console non è disponibile in fase di setup, e richiede l'utilizzo di un sistema operativo Windows 7 e Windows Server 2008 R2 "Full" (i.e.: non in modalità Server Core).

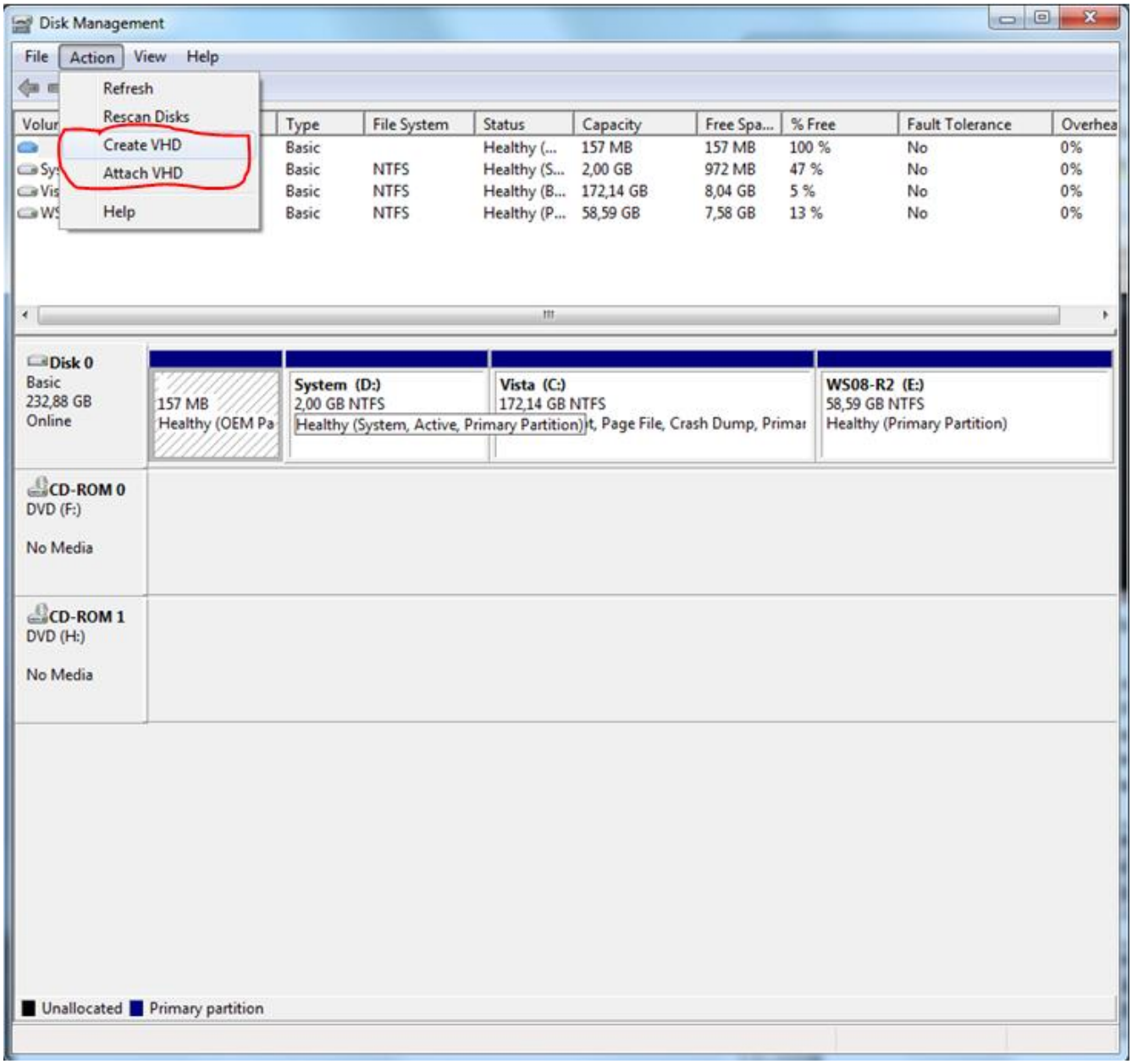

Figura 17: Creazione e Mount di un disco tramite la console Disk Management

6. Uscire dall'ambiente DiskPart inserendo il comando Exit.
7. Chiudere la console CLI di WinPE tramite il comando EXIT e ritornare sulla schermata di installazione di Windows 7 (cf. figura 14).
8. Iniziare l'installazione cliccando sul bottone "*Install now*".
9. Selezionare come destinazione il disco VHD precedentemente creato.

10. Alla fine della installazione del nuovo sistema operativo (e.g.: Windows 7), qualora sul computer utilizzato fosse presente un altro sistema operativo (e.g.: Windows XP Professional), questa modalità di installazione provvede ad apportare una modifica alla configurazione del database BCD in modo tale da predisporre un ambiente multi-boot, dando la possibilità di scegliere al momento dell'accensione del computer, quale sistema operativo avviare.

Creare una nuova entry nel database BCD per il boot di un sistema operativo Windows 7 o Windows Server 2008 R2 da VHD

Supponendo di avere a disposizione un file VHD (e.g.: vhd=[E:]\VHDs\Win7.vhd) all'interno del quale è stato installato (come spiegato nella precedente sezione) un sistema operativo Windows 7 e/o Windows Server 2008 R2, è possibile creare una nuova riga all'interno del menu di boot (alla quale corrisponde un nuovo record nel database BCD) nel modo seguente:

1. Duplicazione della entry BCD corrispondente al sistema operativo correntemente in uso:

   bcdedit -copy {current} -d "Windows 7 [Boot da VHD]"

2. Identificare il GUID assegnato alla nuova entry; esempio: {a194db36-6c1d-11de-bbf5-8553371bbef8}.

3. Configurazione degli attributi necessari per il boot:

```
bcdedit -set {a194db36-6c1d-11de-bbf5-8553371bbef8} device vhd=[E:]\VHDs\Win7.vhd
bcdedit -set {a194db36-6c1d-11de-bbf5-8553371bbef8} osdevice vhd=[E:]\VHDs\Win7.vhd
bcdedit -set {a194db36-6c1d-11de-bbf5-8553371bbef8} detecthal on
```

4. Configurare la nuova entry BCD in modo che venga visualizzata all'inizio della lista:

```
bcdedit -toolsdisplayorder {a194db36-6c1d-11de-bbf5-8553371bbef8} -addfirst
```

5. Configurare la nuova entry BCD in modo da avviarsi come sistema operativo di default:

```
bcdedit -default {a194db36-6c1d-11de-bbf5-8553371bbef8}
```

Ripristino della struttura di Boot Sector

Tramite l'utility BootSect.exe è possibile creare o ricreare la struttura del Boot Sector di un disco nel caso in cui questa sia stata danneggiata, ad esempio a causa della installazione di un SO Windows XP su un computer dove precedentemente era installato Windows Vista/Windows 7 oppure Windows Server 2008 R2 (come indicato nell'articolo KB 919529: http://support.microsoft.com/kb/919529/en-us) o viceversa.

L'utility BootSect.exe non è nativa dei sistemi operativi Windows Vista e Windows 7. Essa è disponibile all'interno del package WAIK oppure nella cartella \boot del DVD di installazione del SO Windows 7.

- Alcuni esempi:

a. Imprimere nel MBR il boot sector utilizzato fino a WinXP/WS03 (versione: NT52) sulla partizione attiva identificata da "C:":
    - bootsect /nt52 c:
b. Imprimere nel MBR il boot sector utilizzato fino a WinXP/WS03 (versione: NT52) sulla partizione attiva nascosta (solitamente non esposta con una lettera e che coincide con la prima partizione):
    - bootsect /nt52 all
c. Imprimere nel MBR il boot sector utilizzato da Windows Vista (versione: NT60) in poi sulla partizione attiva identificata da "C:":
    - bootsect /nt60 c:

Ripristino dei file di boot di Windows Vista, Windows 7 e Windows Server 2008/2008-R2

Tramite l'utility BcdBoot.exe è possibile ripristinare il file bootmgr che svolge le funzioni di boot manager e la struttura del database contenente la configurazione di boot (BCD) (ad uno stato di default o "fabbrica") per i sistemi operativi Windows Vista, Windows 7, Windows Server 2008/2008-R2.

Sintassi del comando BcdBoot.exe:

Bcdboot - Bcd boot file creation and repair tool.

The bcdboot.exe command-line tool is used to copy critical boot files to the system partition and to create a new system BCD store.

bcdboot <source> [/l <locale>] [/s <volume-letter>] [/v]
[/m [{OS Loader ID}]]

source — *Specifies the location of the windows system root.*

/l — *Specifies an optional locale parameter to use when initializing the BCD store. The default is US English.*

/s — *Specifies an optional volume letter parameter to designate the target system partition where boot environment files are copied. The default is the system partition identified by the firmware.*

/v — *Enables verbose mode.*

/m — *If an OS loader GUID is provided, this option merges the given loader object with the system template to produce a bootable entry. Otherwise, only global objects are merged.*

Examples: bcdboot c:\windows /l en-us
bcdboot c:\windows /s h:
bcdboot c:\windows /m {d58d10c6-df53-11dc-878f-00064f4f4e08}

Di seguito alcuni esempi di utilizzo:

1. bcdboot c:\windows /s c:

   in tal caso la partizione di sistema e di boot coincidono, ed il sistema operativo è installato nella cartella c:\windows (i.e.: identifica la cosiddetta "Windows System Root").

2. bcdboot v:\windows /s c: /v

   dove:

   - v:\windows identifica la cartella di installazione di Windows 7. Da notare che nel caso di un sistema operativo Win7/WS08-R2 installato su un disco VHD, "v:" potrebbe essere la lettera utilizzata per esporre il disco VHD montato tramite l'utility DiskPart.exe (da un prompt di comando avviato con privilegi amministrativi) oppure tramite la console grafica DiskMgmt.msc.
   - c: identifica la partizione attiva contenente i file di boot.

In caso di sistema operativo Windows 7 (o Windows Server 2008 R2) installato su disco virtuale VHD, è necessario montare prima il disco tramite DiskPart.exe oppure DiskMgmt.msc, ed esporlo mediante una lettera di unità.

Sommario

Questo capitolo analizza il processo di *boot-chaining* dei sistemi operativi in generale, anche se l'attenzione è focalizzata principalmente sui sistemi operativi Microsoft pre e post Windows Vista. Inoltre, viene affrontata anche la problematica della gestione del database che controlla il processo di boot dei sistemi operativi Microsoft: boot.ini per i computer con sistema operativo precedente a Vista e BCD (*Boot Configuration Data*) per i sistemi operativi Windows Vista, Windows Server 2008 e successivi. In fine, vengono presentate alcune delle novità riguardanti il nuovo boot manager di Windows 7 e Windows Server 2008 R2, come ad esempio la possibilità di effettuare il boot di un sistema operativo installato direttamente su un disco virtuale VHD. Per finire vengono presi in considerazione alcuni casi emblematici di danneggiamento e ripristino della struttura di boot (MBR, Boot Sector e file di boot).