



MINISTERO DELLO SVILUPPO ECONOMICO
DIREZIONE GENERALE PER LA LOTTA ALLA CONTRAFFAZIONE
UFFICIO ITALIANO BREVETTI E MARCHI

DOMANDA NUMERO	102000900813981
Data Deposito	14/01/2000
Data Pubblicazione	14/07/2001

Priorità	09/236,859
Nazione Priorità	US
Data Deposito Priorità	

Sezione	Classe	Sottoclasse	Gruppo	Sottogruppo
G	06	F		

Titolo

SISTEMA E PROCEDIMENTO PER DARE INIZIO AL FUNZIONAMENTO DI UN SISTEMA ELABORATORE.

DESCRIZIONE dell'invenzione industriale dal titolo:

"Sistema e procedimento per dare inizio al funzionamento di un sistema elaboratore",

di: DELL USA, L.P., nazionalità statunitense, One Dell Way, Round Rock, Texas 78682-2244 (U.S.A.)

Inventori designati : BEELITZ, Alan, E. ; AMBERG, Richard, D.

Depositata il: 14 gennaio 2000

TO 2000A000032

TESTO DELLA DESCRIZIONE

Le descrizioni in questo contesto sono relative in generale a sistemi di elaborazione di informazioni ed in particolare a un sistema ed un procedimento per dare inizio al funzionamento di un sistema di elaboratore.

La presente domanda è relativa alla Domanda di Brevetto statunitense anch'essa pendente con numero di serie 09/198.731, depositata il 24 novembre 1998, intitolata COMPUTER SYSTEM AND METHOD FOR PREPARING A COMPUTER-READABLE MEDIUM, che cita Alan E.Beelitz come inventore. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di Brevetto statunitense anch'essa pendente con numero

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

di serie 09/198.007, depositata il 24 novembre 1998, intitolata COMPUTER SYSTEM AND METHOD FOR ACCESSING A COMPUTER-READABLE MEDIUM, che cita Alan E.Beelitz come inventore. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di brevetto statunitense anch'essa pendente con numero di serie 08/951.135, depositata il 15 ottobre 1997, intitolata, MODIFIABLE PARTITION BOOT RECORD FOR A COMPUTER MEMORY DEVICE, che cita Alan Beelitz come inventore. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di brevetto statunitense anch'essa pendente con numero di serie 08/984.386, depositata il 3 dicembre 1997, intitolata SYSTEM AND METHOD FOR CHANGING PARTITION MAPPINGS TO LOGICAL DRIVERS IN A COMPUTER MEMORY, che cita Alan Beelitz come inventore. La domanda pendente è qui allegata per riferimento nella sua interezza, ed assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di brevetto statunitense anch'essa pendente con numero

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

di serie 08/950.545 depositata il 15 ottobre 1997, intitolata SYSTEM AND METHOD FOR UPDATING PARTITION MAPPINGS TO LOGICAL DRIVERS IN A COMPUTER MEMORY DEVICE, che cita Alan Beelitz come inventore. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di brevetto statunitense anch'essa pendente con numero di serie 08/947.138 depositata l'8 ottobre 1997, intitolata METHOD FOR SIMULATING A COMPUTER STORAGE DEVICE, che cita Alan Beelitz come inventore. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di brevetto statunitense anch'essa pendente con numero di serie 08/951.137, depositata il 15 ottobre 1997, intitolata SYSTEM AND METHOD FOR UTILIZING A RAM DISK, che cita Alan Beelitz come inventore. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di brevetto statunitense anch'essa pendente con numero

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

di serie 09/012.196, depositata il 23 gennaio 1998, intitolata SYSTEM AND METHOD FOR PREPARING A COMPUTER MEMORY che cita Alan Beelitz come inventore. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di brevetto statunitense anch'essa pendente con numero di serie 09/198.007, depositata il 24 novembre 1998, intitolata COMPUTER SYSTEM AND METHOD FOR ACCESSING A COMPUTER-READABLE MEDIUM, che cita Alan Beelitz come inventore. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di brevetto statunitense anch'essa pendente con numero di serie 09/198.731, depositata il 24 novembre 1998, intitolata COMPUTER SYSTEM AND METHOD FOR PREPARING A COMPUTER-READABLE MEDIUM, che cita Alan Beelitz come inventore. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di brevetto statunitense anch'essa pendente con numero di serie 08/920.773, depositata il 29 agosto 1997,

BUZZI, NOTARO &
ANTONELLI D'OUX
s.r.l.

intitolata SOFTWARE INSTALLATION AND TESTING FOR A BUILD-TO-ORDER COMPUTER SYSTEM che cita Richard D. Amberg, Roger W. Wong e Michael A. Brundridge come inventori. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di brevetto statunitense anch'essa pendente con numero di serie 08/919.959, depositata il 29 agosto 1997, intitolata SOFTWARE INSTALLATION AND TESTING FOR A BUILD-TO-ORDER COMPUTER SYSTEM, che cita Richard D. Amberg, Roger W. Wong e Michael A. Brundridge come inventori. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

La presente domanda è relativa alla Domanda di brevetto statunitense anch'essa pendente con numero di serie 08/921.438, depositata il 29 agosto 1997, intitolata DATABASE FOR FACILITATING SOFTWARE INSTALLATION AND TESTING FOR A BUILD-TO-ORDER COMPUTER SYSTEM, che cita Richard D. Amberg, Roger W. Wong e Michael A. Brundridge come inventori. La domanda pendente è qui allegata per riferimento nella sua interezza, ed è assegnata al titolare della presente domanda.

BUZZI, NOTARO &
ANTONIELLO D'OUX
s.r.l.

L'inizio del funzionamento di un sistema di elaboratore può essere chiamato "booting" (inizializzazione) (oppure "rebooting" (reinizializzazione)) del sistema di elaboratore. Il sistema di elaboratore viene inizializzato in risposta ad un evento ("evento di boot"). Tale evento di boot può essere, per esempio, un utente che "accende" il sistema di elaboratore (ad esempio l'utente che genera l'applicazione di energia elettrica ad un sistema di elaboratore accendendo/spegnendo un pulsante sul sistema di elaboratore). In alternativa, tale evento di boot può essere la ricezione da parte del sistema di elaboratore di un comando per eseguire inizialmente il software di sistema operativo. Per esempio, il sistema di elaboratore può ricevere tale comando dall'utente (ad esempio attraverso un dispositivo di ingresso), oppure da un applicativo di elaboratore eseguito dal sistema di elaboratore, oppure da un altro sistema di elaboratore (ad esempio attraverso una rete).

In risposta ad un evento di boot, il sistema di elaboratore dà inizio all'esecuzione di software (ad esempio software di sistema operativo quale "Microsoft Windows"). Il sistema di elaboratore può

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

leggere il software da un supporto leggibile da elaboratore, quale un disco rigido o un dischetto floppy. Secondo una tecnica precedente, il sistema di elaboratore legge il software da (cioè inizializza da): (a) il dischetto floppy se tale dischetto è presente all'interno del sistema, come per esempio se un dischetto floppy è fisicamente residente nell'unità A: del sistema di elaboratore; oppure (b) il disco rigido se non è presente alcun dischetto floppy all'interno del sistema.

Un problema di tale tecnica anteriore è che il sistema di elaboratore non riesce a fornire un'opzione di inizializzazione del disco rigido in una situazione in cui è presente un dischetto floppy all'interno del sistema. Tale opzione sarebbe utile nell'esecuzione del montaggio in serie e nel test di centinaia, o persino migliaia, di sistemi di elaboratore in un singolo giorno o un altro breve periodo di tempo. Per esempio tale opzione potrebbe aiutare a ridurre il numero di volte in cui un utente umano inserisce fisicamente un dischetto floppy nella (o toglie un dischetto floppy dalla) unità A: del sistema di elaboratore durante il procedimento di montaggio e test. Come parte di una operazione di assemblaggio o

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

fabbricazione di elaboratori ad elevati volumi, tale inserimento e rimozione ripetuta è inefficiente.

Di conseguenza, si è verificata la necessità di avere un sistema e un procedimento per dare inizio al funzionamento di un sistema di elaboratore, in cui sono superati vari problemi di tecniche anteriori.

L'invenzione

Una forma di attuazione, di conseguenza, prevede un sistema di elaboratore che comprende un primo e un secondo supporto leggibili da elaboratore, il secondo supporto leggibile da elaboratore memorizza informazioni che specificano uno tra il primo e il secondo supporto leggibile da elaboratore da cui si inizializza il sistema di elaboratore. Il sistema di elaboratore legge le informazioni dal secondo supporto leggibile da elaboratore e inizializza da quello specificato tra il primo e il secondo supporto leggibile da elaboratore in risposta ad un evento e alle informazioni.

Il vantaggio principale di questa forma di attuazione è che si superano vari problemi di tecniche precedenti, e il sistema di elaboratore

BUZZI, NOTARO &
ANTONIELLO D'OUX
s.r.l.

fornisce una opzione di inizializzazione dal disco rigido anche in una situazione in cui è presente un dischetto floppy all'interno del sistema.

Breve descrizione dei disegni

La Figura 1 è un diagramma a blocchi di un sistema di elaboratore secondo la forma di attuazione illustrativa.

La Figura 2 è una illustrazione concettuale di informazioni memorizzate da un disco rigido del sistema di elaboratore di Figura 1.

La Figura 3 è un primo diagramma di flusso di funzionamento del sistema di elaboratore di Figura 1.

La Figura 4 è un secondo diagramma di flusso di funzionamento del sistema di elaboratore di Figura 1.

La Figura 5 è un terzo diagramma di flusso di funzionamento del sistema di elaboratore di Figura 1.

Descrizione dettagliata

La Figura 1 è un diagramma a blocchi di un sistema di elaboratore, indicato in generale con il 100, secondo la forma di attuazione illustrativa. Il sistema 100 comprende dispositivi di ingresso 104, un dispositivo di visualizzazione 106, un

dispositivo di stampa 108 ed un elaboratore 102 per eseguire processi ed effettuare operazioni (ad esempio comunicare informazioni) in risposta ad essi come discusso ulteriormente più avanti. Nella forma di attuazione illustrativa, l'elaboratore 102 è un elaboratore personale ("PC") IBM compatibile che esegue il software di sistema operativo Microsoft Windows 95. Tutti i prodotti Microsoft qui identificati sono disponibili dalla Microsoft Corporation, One Microsoft Way, Redmond, WA 98052-6399, telefono: (425)882-8080.

L'elaboratore 102 è connesso a dispositivi di ingresso 104, un dispositivo di visualizzazione 106 ed un dispositivo di stampa 108. Il dispositivo di visualizzazione 106 è, per esempio, un tubo a raggi catodici elettronico convenzionale. Il dispositivo di stampa 108 è, per esempio, una stampante elettronica o un plotter convenzionale. Inoltre, l'elaboratore 102 comprende altoparlanti interni per inviare in uscita segnali audio. In una forma di attuazione alternativa, gli altoparlanti sono esterni all'elaboratore 102. Inoltre, il sistema 100 comprende (a) un primo supporto (o dispositivo) leggibile da elaboratore 110 che è un dischetto floppy e (b) un secondo supporto (o dispositivo)

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

leggibile da elaboratore 111 che è un disco rigido di elaboratore.

Un utente umano 112 ed un elaboratore 102 operano in associazione uno con l'altro. Per esempio, in risposta ai segnali dall'elaboratore 102, il dispositivo di visualizzazione 106 visualizza immagini visive, e l'utente 112 vede tali immagini visive. Inoltre, in risposta al segnale dall'elaboratore 102, il dispositivo di stampa 108 stampa immagini visive sulla carta, e l'utente 112 vede tali immagini visive. Inoltre, l'utente 112 attiva i dispositivi di inserimento 104 allo scopo di inviare in uscita informazioni all'elaboratore 102 e l'elaboratore 102 riceve tali informazioni dei dispositivi di ingresso 104.

I dispositivi di ingresso 104 comprendono per esempio, una tastiera elettronica convenzionale, ed un dispositivo di puntamento quale un "mouse elettronico" una sfera rotolante o una penna luminosa convenzionali. L'utente 112 attiva la tastiera per inviare in uscita informazioni di testo alfanumerico all'elaboratore 102, e l'elaboratore 102 riceve tali informazioni di testo alfanumerico dalla tastiera. L'utente 112 attiva il dispositivo di puntamento per inviare in uscita

BUZZI, NOTARO &
ANTONIELLI D'OUIX
s.r.l.

informazioni di controllo di cursore all'elaboratore 102, e l'elaboratore 102 riceve tali informazioni di controllo di cursore dal dispositivo di puntamento.

L'elaboratore 102 è intercorrelato strutturalmente e funzionalmente con ciascuno dei suoi supporti leggibili da elaboratore connessi (ad esempio i supporti 110 e 111) leggibili da elaboratore), come descritto ulteriormente qui di seguito. Per esempio, il dischetto floppy 110 memorizza (ad esempio codifica, registra o realizza) materiale descrittivo funzionale (ad esempio comprendente ma limitato a programmi di elaboratore (chiamati anche software o applicativi e strutture informative). Tale materiale descrittivo funzionale impartisce una funzionalità quando è codificato sul dischetto floppy 110. Inoltre, tale materiale descrittivo funzionale è intercorrelato strutturalmente e funzionalmente con il dischetto floppy 110.

Nell'ambito di tale materiale descrittivo funzionale, le strutture informative definiscono interrelazioni strutturali e funzionali tra tali strutture di informazioni e il dischetto floppy 110 (e altri aspetti del sistema 100). Tali

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

interrelazioni permettono di realizzare la funzionalità delle strutture informative. Inoltre, nell'ambito di tale materiale descrittivo funzionale, i programmi di elaboratore definiscono interrelazioni strutturali e funzionali tra tali programmi di elaboratore e il dischetto floppy 110 (e altri aspetti del sistema 100). Tali interrelazioni permettono di realizzare la funzionalità dei programmi di elaboratore.

Per esempio, l'elaboratore 102 legge (ad esempio carica, accede o copia) tale materiale descrittivo funzionale in un dispositivo di memoria di elaboratore (ad esempio memoria ad accesso casuale ("RAM")) dell'elaboratore 102, e l'elaboratore 102 esegue le sue operazioni (come descritto altrove in questo contesto) in risposta a tale materiale che è memorizzato in tale dispositivo di memoria. Più in particolare, l'elaboratore 102 esegue l'operazione di elaborare (ad esempio eseguire) un applicativo di elaboratore (che è memorizzato, codificato, registrato o realizzato su un supporto leggibile da elaboratore) per far sì che l'elaboratore 102 esegua operazioni aggiuntive (come descritto altrove in questo contesto). Di conseguenza, tale materiale

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

descrittivo funzionale presenta un'interrelazione funzionale con il modo in cui l'elaboratore 102 esegue i suoi processi ed effettua le sue operazioni.

Inoltre, il dischetto floppy 110 è un dispositivo da cui l'applicativo di elaboratore è accessibile da parte dell'elaboratore 102, e l'applicativo di elaboratore è elaborabile dall'elaboratore 102 per far sì che l'elaboratore 102 esegua tali operazioni aggiuntive. In aggiunta a leggere tale materiale descrittivo funzionale dal dischetto floppy 110, l'elaboratore 102 è in grado di leggere tale materiale descrittivo funzionale da (o attraverso) una rete di elaboratori 114 che è un altro tipo di supporto (o dispositivo) leggibile da elaboratore connesso all'elaboratore 102. Inoltre, il dispositivo di memoria (dell'elaboratore 102) è esso stesso un supporto (o dispositivo) leggibile da elaboratore.

La rete 114 comprende un elaboratore server gestore di controllo ("LCM") della rete di area locale ("LAN") di rete. Il LCM è discusso ulteriormente in questo contesto in relazione ad un NetPC. Per comunicare con (cioè inviare in uscita informazioni a, e ricevere informazioni da) la rete

BUZZI, NOTARO &
ANTONIELLI D'OLIX
s.r.l.

114 (compreso il LCM), l'elaboratore 102 comprende una scheda di interfaccia di rete ("NIC") che è un altro ulteriore tipo di supporto (o dispositivo) leggibile da elaboratore connesso all'elaboratore 102.

La Figura 2 è una illustrazione concettuale di informazioni, indicate in generale con il 200, memorizzate dal disco rigido 111. Un controllore di comando dell'elaboratore 102 invia in uscita il segnale a disco rigido 111, e il disco rigido 111 memorizza tali informazioni in risposta a tali segnali. Tali informazioni comprendono un record di boot master MBR, come illustrato in Figura 2. Inoltre, nell'esempio di Figura 2, il disco rigido 111 comprende le partizioni P1 e P2.

Nell'esempio di Figura 2, la partizione P1 è una partizione "primaria" e la partizione P2 è una partizione di "utilità" che è collocata all'interno di FR1. FR1 di informazioni 200 è sufficientemente grande da contenere la partizione P2 all'interno di FR1. Una partizione di utilità può per esempio, memorizzare programmi e informazioni che sono utili durante la costruzione, l'assemblaggio e il test del sistema 100. Tuttavia, in almeno una situazione, tali programmi e informazioni sono meno

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

utili, e persino forse non necessari, durante l'utilizzo e il funzionamento successivi del sistema 100. In tale situazione, dopo la costruzione, l'assemblaggio e il test del sistema 100: (a) la partizione di utilità stessa è meno utile, e forse neanche necessaria; (b) a confronto con la partizione di utilità, una partizione primaria è più utile durante l'uso e il funzionamento successivi del sistema 100; e (c) vantaggiosamente, il sistema 100 della forma di realizzazione illustrativa è in grado di cancellare la partizione di utilità in modo che i settori siano riallocati dalla partizione di utilità alla partizione primaria. In questa maniera, la partizione primaria del disco rigido si allarga in modo vantaggioso per ottenere un utilizzo e un funzionamento successivi più versatili del sistema 100.

Secondo l'architettura di elaboratore personale ("PC") IBM compatibile, il MBR è il primo settore (cilindro 0, testa 0, settore 1) sul disco rigido 111. Un settore è l'unità di memoria più piccola singolarmente indirizzabile sul disco rigido 111. Secondo l'architettura degli elaboratori personali ("PC") IBM compatibili, i

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

dischi rigidi hanno una dimensione di settore di 512 byte. Di conseguenza, nella forma di attuazione illustrativa, nella misura in cui l'elaboratore 102 forma (ad esempio crea o modifica) una partizione sul disco rigido 111, l'elaboratore 102 definisce la partizione lungo i limiti di cilindro. Un cilindro è una unità organizzativa sul disco rigido 111 che è uguale al numero di testate di lettura/scrittura moltiplicato per il numero di settori per traccia moltiplicato per il numero di byte per settore.

Il MBR comprende istruzioni di inizializzazione BI ed una tabella di partizione master MPT per gestire la memorizzazione su disco. Secondo l'architettura di elaboratori personali ("PC") IBM compatibili, il MPT comprende quattro inserimenti di tabelle di partizione, cioè PTE1, PTE2, PTE3 e PTE4. Un inserimento di tabella di partizione PTE_x (dove $x = 1, 2, 3$ o 4) comprende informazioni che descrivono la dimensione, la posizione e il tipo (esteso, file system di nuova tecnologia ("NTFS"), tabella di allocazione di file ("FAT") a 16 bit o 32 bit, primario o utilità) di una partizione P_x associata a tale PTE_x.

In risposta ad un evento di boot,

l'elaboratore 102 dà inizio all'esecuzione del software di sistema operativo ("OS") (ad esempio Microsoft DOS o Microsoft Windows). Tale evento di boot può essere, per esempio, l'utente 112 che "accende" l'elaboratore 102 (ad esempio l'utente 112 che genera l'applicazione di energia elettrica all'elaboratore 102 commutando un pulsante di accensione/spegnimento dell'elaboratore 102). In alternativa, tale evento di boot può essere la recezione da parte dell'elaboratore 102 di un comando per eseguire inizialmente il software di sistema operativo. Per esempio, l'elaboratore 102 può ricevere tale comando dall'utente 112 (ad esempio tramite dispositivi di ingresso 104), o da un applicativo di elaboratore eseguito dall'elaboratore 102, o da un altro elaboratore (ad esempio attraverso la rete 114).

Di conseguenza, in risposta ad un evento di boot, l'elaboratore 102 copia porzioni dell'OS da un supporto leggibile da elaboratore (ad esempio disco rigido 111 oppure rete 114) nel dispositivo di memoria dell'elaboratore 102, e l'elaboratore 102 esegue tali porzioni. Inoltre, in risposta all'esecuzione dell'OS l'elaboratore 102 copia porzioni di software applicativo da un mezzo

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

leggibile da elaboratore nel dispositivo di memoria dall'elaboratore 102, e l'elaboratore 102 esegue tali porzioni. Inoltre, in risposta all'esecuzione dell'OS l'elaboratore 102 alloca una porzione del suo dispositivo di memoria per memorizzare strutture informative.

Tali strutture informative comprendono informazioni per identificare unità logiche. Per esempio, tali strutture informative comprendono informazioni che identificano una rispettiva unità logica associata ad una partizione Px del disco rigido 111. Più in particolare, nella forma di attuazione illustrativa, tali strutture informative comprendono informazioni che designano la partizione Px tramite una lettera di unità logica (ad esempio C: oppure A:) associata. La lettera di unità logica identifica una rispettiva unità logica associata alla partizione Px.

In tale maniera, l'elaboratore 102 designa (ad esempio mappa o fa riferimento a) la partizione Px tramite la sua lettera di unità logica (ad esempio C: oppure A:) associata. Con tale designazione, la partizione Px analogamente è associata alla rispettiva unità logica che è identificata da tale lettera di unità logica associata. Tuttavia, senza

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

tale designazione, è possibile che una partizione non sia associata con alcuna unità logica.

Di conseguenza, in risposta all'esecuzione dell'OS, l'elaboratore 102 è in grado di designare la partizione primaria P1 tramite la lettera C: ed è in grado di designare la partizione di utilità P2 tramite la lettera A:. In tale maniera, l'elaboratore 102 mappa la partizione primaria P1 in una unità C: logica e mappa la partizione di utilità P2 in una unità A: virtuale. Tale unità A: è "virtuale", poiché il suo spazio di memoria effettivo è collocato sul disco rigido 111 invece del dischetto floppy 110.

Come esempio, nel designare una partizione tramite la lettera C: associata ad una rispettiva unità logica C:, l'elaboratore 102: (a) crea le strutture informative IS sotto forma di una tabella di dati di unità ("DDT") ed un blocco di parametri di unità ("DPB"); (b) modifica una struttura di directory corrente ("CDS") associata all'unità logica C: che è implementata come matrice lineare piuttosto che elenco collegato, in modo tale che l'elaboratore 102 non allochi spazio di memoria aggiuntivo per la CDS; e (c) modifica una tabella di variabili di sistema ("SysVars")

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

dell'elaboratore 102 in modo che comprende un'indicazione dell'unità logica C:.

Ogni inserimento di tabella di partizione PTE_x comprende un rispettivo indicatore "attivo". Il controllore di unità dell'elaboratore 102 in via in uscita segnali a disco rigido 111, cosicchè un massimo di uno solo dei quattro inserimenti di tabella di partizione PTE_x è contrassegnato come "attivo" in qualsiasi particolare momento (cioè soltanto un massimo di uno dei quattro indicatori "attivi" opposto ad uno stato vero di uno logico in qualsiasi momento particolare). BI comprende istruzioni per determinare se (e quale) uno degli inserimenti di tabella di partizione PTE_x è contrassegnato come "attivo". Nell'inizializzazione del disco rigido 111 (in risposta ad un evento di boot), l'elaboratore 102 legge (o "copia") ed esegue porzioni dell'OS dalla partizione attiva (cioè la partizione P_x associata con PTE_x che è contrassegnata come "attiva") del disco rigido 111.

La partizione P1 comprende un record di boot di partizione PBR1, una tabella di allocazione di file FAT1, una directory radice RD1, ed una regione di file FR1. La partizione P2 comprende un record di boot di partizione PBR2, una tabella di

BUZZI, NOTARO &
ANTONIELLI D'OLIX
s.r.l.

allocazione di file FAT2, una directory radice RD2, ed una regione di file FR2. PBRx è il record di boot di partizione di Px, cioè la partizione associata a PTE_x. Nella forma di attuazione illustrativa, un PBRx è un singolo settore di informazioni. BI ed ogni PBRx comprendono istruzioni secondo il tipo, la versione e il linguaggio del software di sistema operativo. In risposta a segnali dall'elaboratore 102, un FRx memorizza file di informazioni.

Ogni FRx è diviso in un certo numero C di grappoli di informazioni. Un particolare grappolo è designato come grappolo b, dove b è un numero intero tra 0 e C-1. Nella forma di attuazione illustrativa, i grappoli all'interno di un particolare FRx sono di dimensioni uguali, cosicchè ogni grappolo ha una dimensione uguale 2^y , dove y è un numero intero e $C \times 2^y$ è minore di o uguale alla dimensione del particolare FRx. Di conseguenza, PBRx comprende un blocco di parametro BIOS che specifica y e C per il particolare FRx. Ogni FRx ha la propria rispettiva dimensione, e ogni partizione ha i propri rispettivi valori di y e C.

Nella forma di attuazione illustrativa, ogni directory radice RDx comprende spazio per

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

memorizzare fino 512 inserimenti per RDx. Tale inserimento è associato ad un file di informazioni e comprende il nome di file del file (ad esempio PROGRAM.EXE, DATA.DAT, COMMANDS.BAT), la data, l'ora, la dimensione, gli attributi (ad esempio nascosto) e il grappolo iniziale. Il grappolo iniziale è uno dei C grappoli all'interno di FRx.

Per accedere ad una partizione P1, il processore dell'elaboratore 102 invia in uscita un comando al controllore di unità dell'elaboratore 102, e il controllore di unità riceve tale comando. Per esempio, tale comando è indirizzato alla partizione P1 specificando la lettera C: che designa la partizione P1. Prima di allocare un gruppo per memorizzare un file in risposta a tale comando indirizzato alla partizione P1, il controllore di unità verifica che il gruppo allocato sia associato ad un inserimento Disponibile all'interno di FAT1. Di conseguenza, nella misura in cui gli inserimenti sono Riservati invece di Disponibili all'interno di FAT1, il controllore di unità non alloca in questo modo i grappoli associati a tali inserimenti.

In questa maniera, effettuando il numero adatto di inserimenti "riservati" all'interno di

FAT1 delle informazione 200, il sistema 100 è in grado di collocare in modo affidabile la partizione P2 all'interno del numero adatto di grappoli di FR1. Questo avviene poiché tali grappoli associati a tali inserimenti Riservati sono inaccessibili da parte del processore dell'elaboratore 102 che invia in uscita (al controllore di unità) un comando che è indirizzato alla partizione P1 (ad esempio specificando la lettera C: che designa la partizione P1). Invece, tali grappoli sono accessibili da parte del processore dell'elaboratore 102 che invia in uscita (al controllore di unità) un comando che è indirizzato alla partizione P2 (ad esempio specificando la lettera A: che designa la partizione P2).

Nell'esempio di Figura 2, l'elaboratore 102 inizializza la tabella di partizione master MPT (del MBR) ad uno stato specifico sia per la partizione primaria P1 sia per la partizione di utilità P2. Nella forma di attuazione illustrativa, la partizione di utilità P2 è relativamente piccola (ad esempio all'incirca 10 megabyte). Inizialmente, il PTE1 è associato alla partizione primaria P1 e il PTE2 è associato alla partizione di utilità P2.

In modo notevole, in risposta ad un comando di

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

"scambio adatto", l'elaboratore 102 è in grado di eseguire un programma che scambia la designazione di unità C: logica tra la partizione primaria P1 e la partizione di utilità P2. Per esempio, se (immediatamente prima di ricevere tale comando) l'elaboratore 102 designa la partizione primaria P1 tramite la lettera C: e designa la partizione di utilità P2 tramite la lettera A:, (in risposta a tale comando) l'elaboratore 102 modifica la sua mappatura di partizioni P1 e P2, cosicchè l'elaboratore 102 invece mappa la partizione primaria P1 sull'unità A: virtuale e mappa la partizione di utilità P2 sull'unità C: logica. L'elaboratore 102 realizza tale scambio modificando le informazioni all'interno di un DPB e DDT (ad esempio per l'unità C: logica) in modo da incorporare parametri per la partizione primaria P1 oppure la partizione di utilità P2, come applicabile.

In risposta ad un opportuno comando di "cancellazione di partizione" (ad esempio un comando ricevuto dal sistema 100 dopo la costruzione, l'assemblaggio e il test del sistema 100), il sistema 100 cancella la partizione di utilità P2 tramite: (a) una modifica adatta

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

dell'inserimento di tabella di partizione PTE_x (ad esempio PTE₂ in questo esempio) associato alla partizione di utilità P₂, cosicchè la partizione P₂ non è più riconosciuta dall'elaboratore 102; (b) la modifica di tutti gli inserimenti Riservati memorizzati dal disco rigido all'interno della FAT₁ delle informazioni 200 in Disponibili; e (c) il fatto di rendere (nella MPT) la partizione primaria P₁ attiva, cosicchè l'elaboratore 102 legge (o "copia") ed esegue istruzioni da PBR₁ in risposta ad un evento di boot.

Le Figure 3, 4 e 5 sono diagrammi di flusso del funzionamento del sistema 100. Secondo un aspetto significativo della forma di attuazione illustrativa, il dischetto floppy 110 comprende un record di boot di dischetto speciale SDBR, che è il primo settore (traccia 0, testa 0, settore 1) sul dischetto floppy 110. Un settore è l'unità di memoria più piccola singolarmente indirizzabile sul dischetto floppy 110. Secondo l'architettura di elaboratori personali ("PC") IBM compatibili, i dischetti floppy hanno una dimensione di settore di 512 byte.

Il SDBR memorizza informazioni per specificare se il sistema 100 inizializza dal dischetto floppy

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

110 o invece dal disco rigido 111. Durante il processo di inizializzazione, l'elaboratore 102 legge tali informazioni dal SDBR in risposta a istruzioni da un sistema di ingresso/uscita di base ("BIOS") di memoria di sola lettura ("ROM") dell'elaboratore 102. In risposta a tali informazioni dal SDBR, l'elaboratore 102 determina se inizializzare dal dischetto floppy 110 oppure invece dal disco rigido 111.

Il sistema 100 comprende software per modificare tali informazioni nel SDBR. In modo notevole, in risposta all'utente 112 che preme un tasto ALT di una tastiera di dispositivi di inserimento 104 durante il processo di inizializzazione, il sistema 100 ignora tali informazioni e inizializza in maniera convenzionale (cioè inizializza dal dischetto floppy 110 se il dischetto floppy 110 è fisicamente residente all'interno dell'unità A: del sistema 110, e inizializza dal disco rigido 111 se il dischetto floppy 110 è fisicamente assente dall'unità A: del sistema 100).

Con riferimento alla Figura 3, il funzionamento comincia in una fase 300 dove l'elaboratore 102 determina se si è verificato un

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

nuovo evento di inizializzazione. Se si è verificato un nuovo evento di inizializzazione, l'elaboratore 102 determina nella fase 302 se il dischetto floppy 110 è fisicamente residente all'interno dell'unità A: del sistema 110. Se il dischetto floppy 110 è fisicamente residente all'interno dell'unità A: del sistema 110, l'elaboratore 102 determina nella fase 304 se il SDBR memorizza informazioni per specificare che il sistema 100 inizializza dal dischetto floppy 110.

Se il SDBR memorizza informazioni per specificare che il sistema 100 inizializza dal dischetto floppy 110, l'elaboratore 102 inizializza dal dischetto floppy 110 in una fase 306. Dopo aver inizializzato dal dischetto floppy 110 nella fase 306, l'elaboratore 102 termina il processo (che varia secondo l'applicazione specifica) nella fase 308.

Se (nella fase 302) il dischetto floppy 110 è fisicamente assente dall'unità A: del sistema 110, l'elaboratore 102 inizializza dal disco rigido 111 in una fase 310. Secondo un aspetto significativo della forma di attuazione illustrativa, se (nella fase 304) il SDBR memorizza informazioni per specificare che il sistema 100 inizializza dal

BUZZI, NOTARO &
ANTONELLI D'OUX
s.r.l.

disco rigido 111 invece che dal dischetto floppy 110, l'elaboratore 102 inizializza dal disco rigido 111 nella fase 310. Dopo la fase 310, il funzionamento continua con la fase 308.

Nell'inizializzazione dal disco rigido 111, l'elaboratore 102:

- (a) legge istruzioni in un dispositivo di memoria dell'elaboratore 102: (i) da BI; e (ii) da PBRx se l'inserimento di tabella di partizione PTE_x è contrassegnato come "attivo"; e
- (b) esegue tali istruzioni in risposta a istruzioni da un sistema di ingresso/uscita di base ("BIOS") di una memoria di sola lettura ("ROM") dell'elaboratore 102.

Nell'inizializzazione dal dischetto floppy 110, l'elaboratore 102:

- (a) legge istruzioni in un dispositivo di memoria dell'elaboratore 102 da un file chiamato IO.SYS sul dischetto floppy 110; e
- (b) esegue tali istruzioni.

Nella forma di attuazione illustrativa, il sistema 100 memorizza software sull'unità A: per

BUZZI, NOTARO &
ANTONIELLI D'OUIX
s.r.l.

eseguire il processo di costruzione, assemblaggio e test. Di conseguenza, nella forma di attuazione illustrativa, il sistema 100 comprende una unità A: fisica in cui è fisicamente residente il dischetto floppy 110, e l'unità A: è associata al dischetto floppy 110. Al contrario, come discusso ulteriormente qui di seguito in relazione alla Figura 4, se l'elaboratore 102 è un NetPC (cioè un PC che non ha alcuna unità A: fisica e si basa sulla connessione alla rete 114 allo scopo di ricevere e inviare in uscita informazioni da e a un altro sistema), il sistema 100 configura il disco rigido 111 in maniera tale che l'unità A: sia associata alla partizione di utilità P2 del disco rigido 111 (cioè una unità A: virtuale).

Con riferimento alla Figura 4, se l'elaboratore 102 è un NetPC, il dischetto floppy 110 è fisicamente assente dall'unità A: del sistema 100 e il disco rigido 111 inizialmente è vuoto e non contiene informazioni di partizionamento (cioè il disco rigido 111 inizialmente non è formattato). Come illustrato in Figura 4, il funzionamento comincia in una fase 400 dove l'elaboratore 102 inizializza (cioè legge ed esegue porzioni del software di sistema operativo Microsoft DOS) dalla

sua scheda di interfaccia di rete ("NIC") che è discussa ulteriormente in precedenza in relazione alla Figura 1. Dopo aver inizializzato dalla NIC, l'elaboratore 102 comunica (in una fase 402) attraverso la rete 114 con il LCM della rete 114 il che è discusso ulteriormente in precedenza in relazione alla Figura 1.

In risposta a istruzioni dalla LCM, l'elaboratore 102 (a) inizializza (cioè scrive informazioni su) il disco rigido 111 in una fase 404 secondo l'esempio di Figura 2, (d) inizializza le strutture informative in una fase 406 per identificare unità logiche secondo l'esempio di Figura 2, e (c) designa la partizione di utilità P2 come partizione attiva in una fase 408 secondo l'esempio di Figura 2.

Dopo la fase 408, l'elaboratore 102 inizializza dalla partizione di utilità P2 del disco rigido 111 in una fase 410. Dopo la fase 410, l'elaboratore 102 mappa la partizione di utilità P2 in una unità A: virtuale in una fase 412. Dopo la fase 412 l'elaboratore 102 genera (in una fase 414) un comando di "reinizializzazione" che è un evento di boot, come discusso ulteriormente più avanti in relazione alla Fase 500 di Figura 5.

Anche se l'elaboratore 102 comprende una unità A: fisica in cui è fisicamente residente il dischetto floppy 110 (cioè anche se l'elaboratore 102 non è un NetPC), il disco rigido 111 inizialmente è vuoto e non contiene informazioni di partizionamento. Di conseguenza, indipendentemente dal fatto che l'elaboratore 102 sia un NetPC, l'elaboratore 102 della forma di attuazione illustrativa esegue l'operazione evidenziata in Figura 5. Tuttavia, l'elaboratore 102 comincia l'operazione evidenziata in Figura 5 dopo aver eseguito la fase 412 di Figura 4, in modo tale che la fase 414 è realizzata dalla fase 500. Inoltre, all'inizio dell'operazione evidenziata in Figura 5, se il dischetto floppy 110 è fisicamente residente all'interno A: del sistema 110, il SDBR del dischetto floppy 110 memorizza inizialmente informazioni per specificare che il sistema 100 inizializza dal dischetto floppy 110.

Con riferimento alla Figura 5, il funzionamento comincia alla fase 500 dove l'elaboratore 102 inizializza dall'unità A:, che è una partizione di utilità P2 del disco rigido 111 (cioè l'unità A: virtuale) se l'elaboratore 102 è un NetPC, e che è un dischetto floppy 110 se il

dischetto floppy 110 è residente fisicamente all'interno dell'unità A: del sistema 110. Dopo aver inizializzato dall'unità A: l'elaboratore 102 esegue istruzioni (alcune ricevute dalla rete 114) per testare l'hardware del sistema 100 in una fase 502. Dopo aver testato l'hardware del sistema 100 l'elaboratore 102 determina in una fase 504 se il dischetto floppy 110 è fisicamente residente all'interno dell'unità A: del sistema 110.

Se il dischetto floppy 110 è residente fisicamente all'interno dell'unità A: del sistema 110, l'elaboratore 102 (a) inizializza il disco rigido 111 in una fase 506 secondo l'esempio di Figura 2, e (b) inizializza le strutture informative in una fase 508 per identificare unità logiche secondo l'esempio di Figura 2. Ciò avviene poiché, se l'elaboratore 102 è un NetPc, l'elaboratore 102 ha già eseguito tale inizializzazione nelle fasi 404 e 406 di Figura 4. Dopo la fase 508 o se l'elaboratore 102 determina nella fase 504 che il dischetto floppy 110 è fisicamente assente dall'unità A: del sistema 110, il funzionamento continua in una fase 510.

Nella fase 510, l'elaboratore 102 copia il software (ad esempio software di dispositivo di

comando di dispositivi, software applicativi e software di sistema operativo Microsoft Windows), dalla rete 114 sul disco rigido 111. Dopo la fase 510 l'elaboratore 1202 designa (in una fase 512) la partizione primaria T1 come partizione attiva secondo l'esempio di Figura 2. In un aspetto significativo della forma di attuazione illustrativa, dopo la fase 512: (a) se il dischetto floppy 110 è fisicamente residente all'interno dell'unità A: del sistema 110, l'elaboratore 102 memorizza (in una fase 514) informazioni nel SDBR per specificare che il sistema 100 inizializza dal disco rigido 111 invece che dal dischetto floppy 110; e (b) se il dischetto floppy 110 è fisicamente assente dall'unità A: del sistema 110, il funzionamento continua direttamente ad una fase 516.

Di conseguenza, dopo la fase 514, l'elaboratore 102 inizializza dalla partizione primaria P1 del disco del disco rigido 111 (cioè l'unità C:) nella fase 516, anche se il dischetto floppy 110 è residente fisicamente all'interno dell'unità A: del sistema 110, come discusso ulteriormente in precedenza in relazione alla Figura 3. Dopo la fase 516, l'elaboratore 102

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

esegue una procedura di test di sistema completamente integrato ("FIST") in modo sostanziale in una fase 518. Secondo un altro aspetto significativo della forma di attuazione illustrativa, dopo la fase 518: (a) se il dischetto floppy 110 è residente fisicamente all'interno dell'unità A: del sistema 110, l'elaboratore 102 memorizza (in una fase 520) informazioni nel SDBR per specificare che il sistema 100 inizializza dal dischetto floppy 110 invece che dal disco rigido 111; e (b) se il dischetto floppy 110 è fisicamente assente dall'unità A: del sistema 110, l'elaboratore 102 designa (nella fase 520) la partizione di utilità P2 come partizione attiva. Di conseguenza, dopo la fase 520 l'elaboratore 102 (a) inizializza tra unità A: in una fase 522 e (b) in una fase 524, in risposta a istruzioni dall'unità A:, legge ed esegue ulteriori istruzioni dalla rete 114 allo scopo di terminare l'esecuzione del processo di costruzione assemblaggio e test.

Quando il sistema 100 inizializza come evento pianificato in risposta ad un comando di "reinizializzazione" (ad esempio fasi 410, 414, 516, 522), il sistema 100 memorizza informazioni in un file indicatore della partizione attiva sul

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

disco rigido 111. Tali informazioni (nel file indicatore) indicano che il sistema 100 sta inizializzano come evento programmato piuttosto che come evento non programmato (ad esempio accidentale). In modo notevole, nella forma di attuazione illustrativa, l'elaboratore 102 è in grado di leggere informazioni dal disco rigido 111 più rapidamente rispetto al dischetto floppy 110, fornendo in tal modo il vantaggio di inizializzare da una unità A: virtuale sul disco rigido 111 invece dell'unità A: fisica in cui risiede fisicamente il dischetto floppy 110.

Il sistema 100 comprende hardware RPC per attivare un clock in tempo reale, un bus di sistema e la memoria del sistema 100. L'hardware RTC comprende una memoria ad accesso casuale statica ("SRAM") a semiconduttori ad ossido metallico complementari ("CMOS") con supporto a batteria non volatile per memorizzare informazioni di configurazione di elaboratore. Secondo tecniche standard industriali, il sistema 100 modifica le informazioni di configurazione di elaboratore inviando in uscita comandi adatti agli indirizzi di porta 70H e 71H. Gli indirizzi di porta 70H e 71H sono associati all'hardware RTC. Di conseguenza, in

BUZZI, NOTARO &
ANTONIELLI D'OLIX
s.r.l.

una forma di attuazione alternativa, il sistema 100 modifica opportunamente le informazioni di configurazione di elaboratore cosicchè, quando il sistema 100 viene successivamente inizializzato, il sistema 100 non riconosce che il dischetto floppy rimane presente all'interno del sistema 100.

Tuttavia, tale forma di attuazione alternativa è meno preferibile della forma di attuazione illustrativa, poiché:

(a) il formato della SRAM CMOS varia tra il venditore, aumentano in tal modo la difficoltà di implementare in pratica la forma di attuazione alternativa;

(b) la SRAM CMOS può essere difficile da azzerare in modo appropriato prima della spedizione all'utente finale;

(c) le informazioni di configurazione dell'elaboratore modificate nella SRAM CMOS possono essere incoerenti con le informazioni memorizzate nella BIOS, specialmente nell'identificazione della presenza di un dischetto floppy, in modo tale che un sistema operativo (ad esempio Microsoft Windows 95) del sistema 100 può essere progettato in modo inadeguato a tener conto di tale incoerenza, quando si legge sia dal BIOS sia dalla SRAM CMOS; e

BUZZI, NOTARO &
ANTONIELLO D'OUX
s.r.l.

(d) la forma di attuazione alternativa può essere meno pratica in una situazione in cui l'elaboratore 102 è un NetPC (cioè una PC che non ha alcun dischetto floppy e che si basa su una connessione alla rete 114 allo scopo di ricevere e inviare in uscita informazioni da e ad un altro sistema) poiché, secondo tecniche industriali messe generalmente in pratica, la SRAM CMOS dell'hardware RTC può avere uno spazio di memoria troppo piccolo per memorizzare informazioni di partizione, aumentando in tal modo la difficoltà di implementare la forma di attuazione alternativa per inizializzare da una unità A: o una unità C: virtuale.

Anche se sono state illustrate e descritte forme di attuazione illustrative, un'ampia gamma di modifiche, variazioni e sostituzione è contemplata nella precedente descrizione e, in alcuni esempi, alcune caratteristiche delle forme di attuazione possono essere impiegate senza un corrispondente uso di altre caratteristiche. Di conseguenza, è appropriato che le rivendicazioni allegate siano costruite in modo ampio e in maniera coerente con il campo di protezione delle forme di attuazione qui discusse.

BUZZI, NOTARO &
ANTONIELLI D'OULD
s.r.l.

RIVENDICAZIONI

1. - Sistema di elaboratore comprendente:
un primo supporto leggibile da elaboratore;
un secondo supporto leggibile da elaboratore
per memorizzare informazioni che specificano uno
tra il primo e il secondo supporto leggibile da
elaboratore da cui si inizializza il sistema di
elaboratore; e

mezzi per leggere le informazioni dal secondo
supporto leggibile da elaboratore e inizializzare
il sistema da elaboratore da quello specifico del
primo e secondo supporto leggibile da elaboratore
in risposta ad un evento e alle informazioni.

2. - Sistema di elaboratore secondo la
rivendicazione 1, in cui il primo supporto
leggibile da elaboratore è un disco rigido.

3. - Sistema di elaboratore secondo la
rivendicazione 1, in cui il secondo supporto
leggibile da elaboratore è un dischetto floppy.

4. - Sistema di elaboratore secondo la
rivendicazione 1, in cui i mezzi danno inizio al
funzionamento del sistema di elaboratore durante
l'inizializzazione.

5. - Sistema di elaboratore secondo la
rivendicazione 4, in cui i mezzi leggono e danno

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

inizio all'esecuzione del software mentre danno inizio al funzionamento del sistema di elaboratore.

6. - Sistema di elaboratore secondo la rivendicazione 5 in cui il software è software di sistema operativo.

7. - Sistema di elaboratore secondo la rivendicazione 5 in cui i mezzi leggono il software da quello specificato tra il primo e il secondo supporto leggibile da elaboratore.

8. - Sistema di elaboratore secondo la rivendicazione 1 in cui i mezzi servono per modificare le informazioni per specificare uno diverso tra il primo e il secondo supporto leggibile da elaboratore da cui si inizializza il sistema operativo.

9. - Procedimento per un sistema di elaboratore che comprende un primo e un secondo supporto leggibile da elaboratore, comprendente le fasi di:

sul secondo supporto leggibile da elaboratore, memorizzare informazioni che specificano uno tra il primo e il secondo supporto leggibile da elaboratore da cui si inizializza il sistema di elaboratore;

leggere informazioni dal secondo supporto

BUZZI, NOTARO &
ANTONELLI D'OUX
s.r.l.

leggibile da elaboratore; e

inizializzare il sistema di elaboratore da quello specificato tra il primo e il secondo supporto leggibile da un elaboratore disposto ad un evento e alle informazioni.

10. - Procedimento secondo la rivendicazione 9 in cui il primo supporto leggibile da elaboratore è un disco rigido, e la memorizzazione delle informazioni comprende la fase di:

sul secondo supporto leggibile da elaboratore, memorizzare le informazioni che specificano uno tra il disco rigido e il secondo supporto leggibile da elaboratore da cui si inizializza il sistema da elaboratore.

11. - Procedimento secondo la rivendicazione 9 in cui il secondo supporto leggibile da elaboratore è un dischetto floppy, e la memorizzazione delle informazioni comprende le fasi di:

sul dischetto floppy, memorizzare le informazioni che specificano uno tra il primo supporto leggibile da elaboratore e il dischetto floppy da cui si inizializza il sistema di elaboratore.

12. - Procedimento secondo la rivendicazione 9 in cui l'inizializzazione del sistema di

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

elaboratore comprende la fase di:

dare inizio al funzionamento del sistema di elaboratore.

13. - Procedimento secondo la rivendicazione 12 in cui l'inizializzazione del sistema di elaboratore comprende le fasi di:

leggere e dare inizio all'esecuzione di software mentre si dà inizio al funzionamento del sistema di elaboratore.

14. - Procedimento secondo la rivendicazione 13 in cui l'inizializzazione del sistema di elaboratore comprende le fasi di:

leggere e dare inizio all'esecuzione del software di sistema operativo mentre si dà inizio al funzionamento del sistema di elaboratore.

15. - Procedimento secondo la rivendicazione 13 in cui la lettura e l'inizio dell'esecuzione del software comprendono le fasi di:

leggere il software da quello specificato tra il primo e il secondo supporto leggibile da elaboratore.

16. - Procedimento secondo la rivendicazione 9 comprendente la fase di:

modificare le informazioni per specificare uno diverso tra il primo e il secondo supporto

leggibile da elaboratore da cui si inizializza il sistema di elaboratore.

17. - Prodotto di programma di elaboratore per un sistema di elaboratore che comprende un primo e un secondo supporto leggibile da elaboratore, comprendente:

un programma di elaboratore elaborabile dal sistema di elaboratore per far sì che il sistema di elaboratore:

legga informazioni dal secondo supporto leggibile da elaboratore, le informazioni specificando uno tra il primo e il secondo supporto leggibile da elaboratore da cui si inizializza il sistema di elaboratore; e

inizializzi da quello specificato tra il primo e il secondo supporto leggibile da elaboratore in risposta ad un evento e dalle informazioni; e

un apparecchio da cui il programma di elaboratore è accessibile da parte del sistema di elaboratore.

18. - Prodotto di programma di elaboratore secondo la rivendicazione 17 in cui il primo supporto leggibile da elaboratore è un disco rigido, e il programma di elaboratore è elaborabile dal sistema di elaboratore per far sì che il

BUZZI, NOTARO &
ANTONIELLI D'OLIX
s. r. l.

sistema di elaboratore:

legga le informazioni dal secondo supporto leggibile da elaboratore, le informazioni specificando uno tra il disco rigido e il secondo supporto leggibile da elaboratore da cui si inizializza il sistema di elaboratore.

19. - Prodotto di programma di elaboratore secondo la rivendicazione 17 in cui il secondo supporto leggibile da elaboratore è un dischetto floppy, e il programma di elaboratore è elaborabile dal sistema di elaboratore per far sì che il sistema di elaboratore:

legga le informazioni dal dischetto floppy, le informazioni specificando uno tra il primo supporto leggibile da elaboratore e il dischetto floppy da cui si inizializza il sistema di elaboratore.

20. - Prodotto di programma di elaboratore secondo la rivendicazione 17 in cui il programma di elaboratore è elaborabile dal sistema di elaboratore per far sì che il sistema di elaboratore:

dia inizio al funzionamento del sistema di elaboratore durante l'inizializzazione.

21. - Prodotto di programma di elaboratore secondo la rivendicazione 20 in cui il programma di

BUZZI, NOTARO &
ANTONELLI D'OUX
s.r.l.

elaboratore è elaborabile dal sistema di elaboratore per far sì che il sistema di elaboratore:

legga e dia inizio all'esecuzione di software mentre da inizio al funzionamento del sistema di elaboratore.

22. - Prodotto di programma di elaboratore secondo la rivendicazione 21 in cui il programma di elaboratore è elaborabile dal sistema di elaboratore per far sì che il sistema di elaboratore:

legga e dia inizio all'esecuzione di software di sistema operativo mentre dà inizio al funzionamento del sistema di elaboratore.

23. - Prodotto di programma di elaboratore secondo la rivendicazione 21 in cui il programma di elaboratore è elaborabile dal sistema di elaboratore per far sì che il sistema di elaboratore:

legga il software da quello specificato tra il primo e il secondo supporto leggibile da elaboratore.

24. - Prodotto di programma di elaboratore secondo la rivendicazione 17 in cui il programma di elaboratore è elaborabile dal sistema di

BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

elaboratore per far sì che il sistema di elaboratore:

modifichi le informazioni per specificare uno diverso tra il primo e il secondo supporto leggibile da elaboratore da cui si inizializza il sistema di elaboratore.

25. - Sistema di elaboratore comprendente:

un disco rigido;

un dischetto floppy per memorizzare informazioni che specificano uno tra il disco rigido e il dischetto floppy da cui si inizializza il sistema di elaboratore;

mezzi per:

leggere le informazioni dal dischetto floppy;

inizializzare il sistema di elaboratore da quello specifico tra il disco rigido e il dischetto floppy in risposta ad un evento di inizializzazione e alle informazioni; e

modificare le informazioni per specificare uno diverso tra il disco rigido e il dischetto floppy da cui si inizializza il sistema di elaboratore;

in cui l'inizializzazione comprende le fasi di:

leggere il software di sistema operativo da quello specificato tra il disco rigido e il

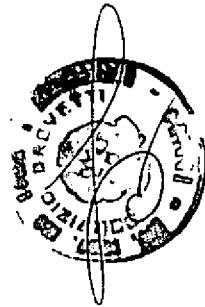
BUZZI, NOTARO &
ANTONIELLI D'OUX
s.r.l.

dischetto floppy; e

dare inizio all'esecuzione del software di sistema operativo mentre si dà inizio al funzionamento del sistema di elaboratore durante l'inizializzazione.

26. Sistema, procedimento e prodotto sostanzialmente come descritto ed illustrato.

Ing. Franco BUZZI
N. Iscriz. ALBO 257
Ha proprio e per gli atti



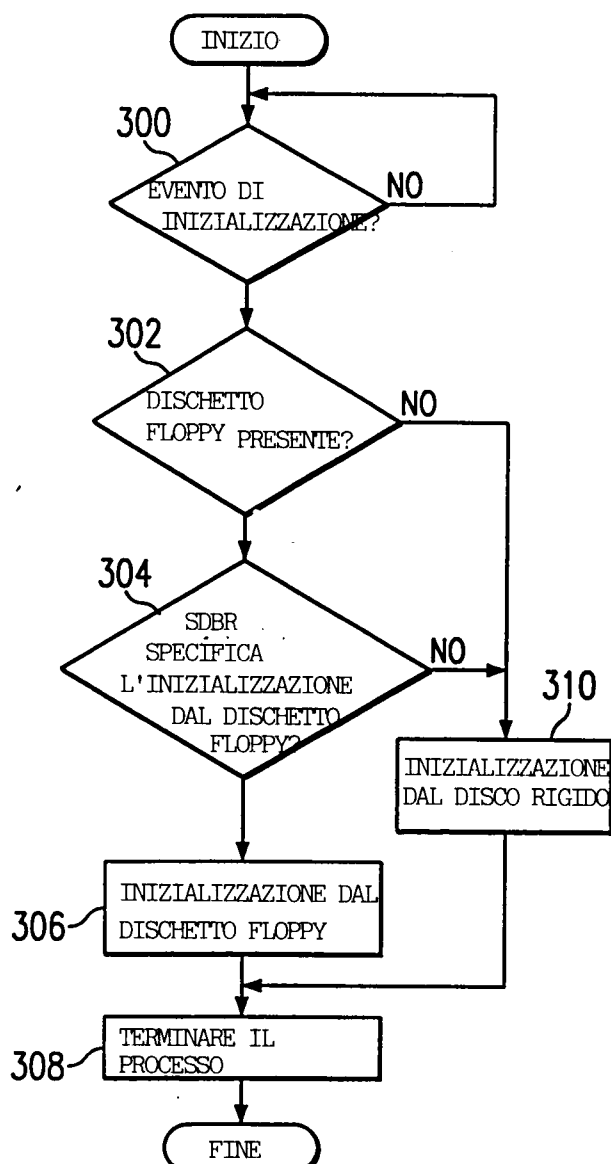


Fig. 3

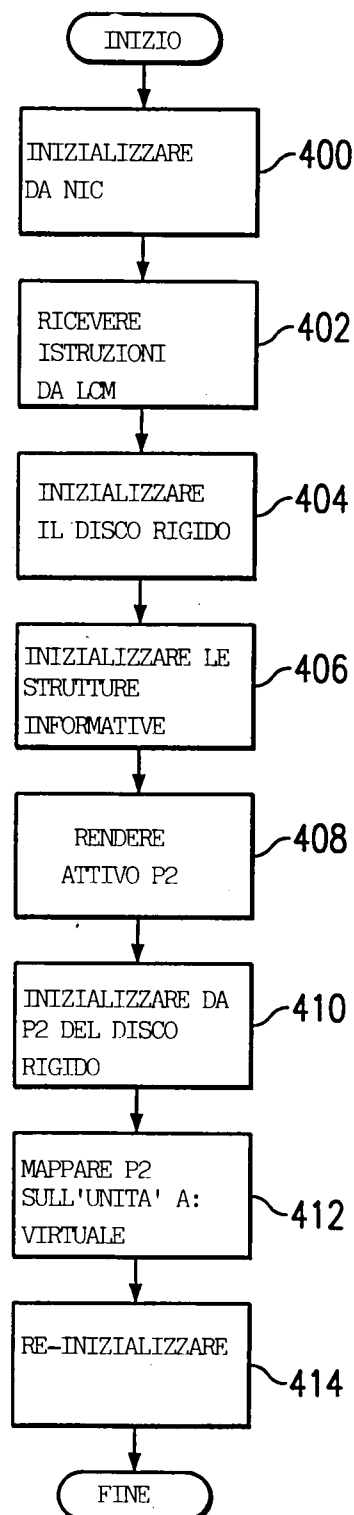


Fig. 4

Ing. Enrico P. 1991
M. 0000000032

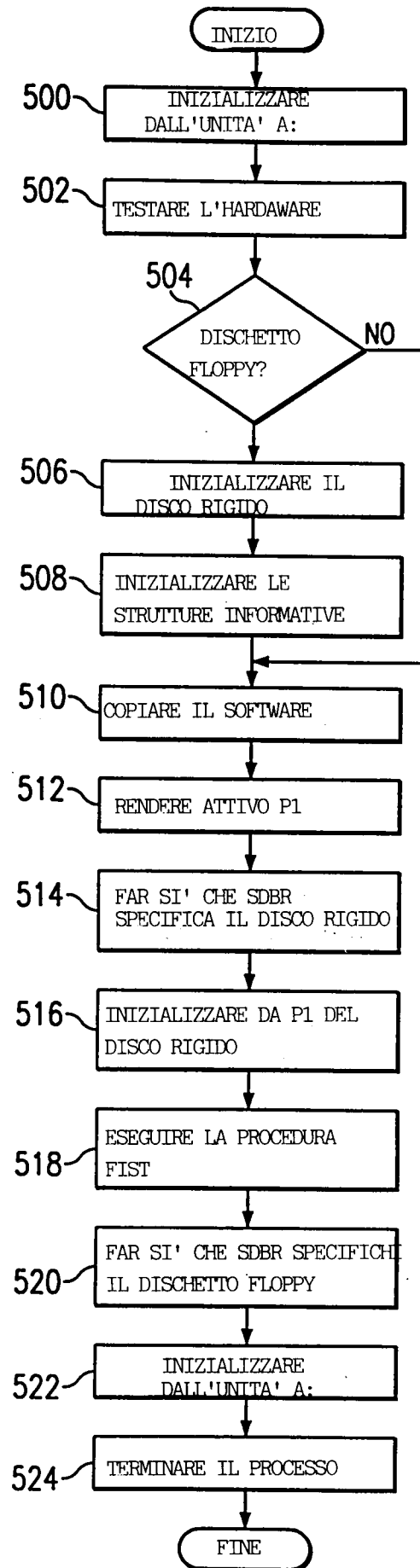


Fig. 5