



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

DOMANDA NUMERO	101998900699260
Data Deposito	21/08/1998
Data Pubblicazione	21/02/2000

Priorità	08/919959
Nazione Priorità	US
Data Deposito Priorità	

Sezione	Classe	Sottoclasse	Gruppo	Sottogruppo
G	06	F		

Titolo

PROCEDIMENTO PER L'INSTALLAZIONE E/O LA VERIFICA DEL SOFTWARE PER UN SISTEMA DI COMPUTER PRODOTTO SU ORDINAZIONE, E SISTEMA DI COMPUTER PER L'ATTUAZIONE DEL PROCEDIMENTO.

DESCRIZIONE dell'invenzione industriale dal titolo:
"Procedimento per l'installazione e/o la verifica
del software per un sistema di computer prodotto su
ordinazione, e sistema di computer per l'attuazione
del procedimento"

di: DELL COMPUTER CORPORATION, nazionalità
statunitense, One Dell Way, Round Rock, Texas
78682-2244 (USA)

Inventori designati: AMBERG, Richard D.; WONG,
Roger W.; BRUNDRIDGE, Michael A.

Depositata il: 21 AGOSTO 1998

T O 98A 000722

TESTO DELLA DESCRIZIONE

SFONDO DELL'INVENZIONE

Riferimento incrociato a domande correlate

La presente domanda è relativa alla domanda di
brevetto statunitense col numero di serie _____
anch'essa pendente, n. di protocollo N-5612 US,
depositata lo stesso giorno della presente domanda,
intitolata Software Installation And Testing For A
Build-To-Order Computer System e che nomina Richard
D. Amberg, Roger W. Wong e Michael A. Brundridge
come inventori, la domanda essendo qui incorporata
per riferimento nella sua interezza.

La presente domanda è relativa alla domanda di
brevetto statunitense col numero di serie _____

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

anch'essa pendente, n. di protocollo M-5217 US, depositata lo stesso giorno della presente domanda, intitolata Software Installation And Testing For A Build-To-Order Computer System e che nomina Richard D. Amberg, Roger W. Wong, and Michael A. Brundridge come inventori, l'applicazione essendo qui incorporata per riferimento nella sua interezza.

Campo dell'invenzione

La presente forma di attuazione è relativa alla diagnostica di sistemi di computer e più in particolare ad un procedimento per sequenziare le fasi di installazione e/o test di software per un sistema di computer.

Descrizione della tecnica nota

I sistemi di personal computer in generale e i sistemi di personal computer IBM compatibili in particolare hanno ottenuto un utilizzo diffuso per fornire potenza di calcolo a molti segmenti della società. Un sistema di personal computer può essere definito di solito come microcomputer da tavolo, a torre o portatile che comprende una unità di sistema avente un processore di sistema ed una memoria volatile e non volatile associata, un monitor di visualizzazione, una tastiera, una o più unità a dischetti, un dispositivo di memorizzazione

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

a disco fisso ed una stampante opzionale.

E' noto installare il software ed eseguire test su sistemi di computer prima che essi siano spediti a aziende o singoli clienti. L'obiettivo dell'installazione di software e del relativo test è quello di produrre in modo efficiente un sistema di computer affidabile ed utile che possa essere inviato ad aziende e singoli privo di errore e pronto al funzionamento. In generale, i test rilevano e analizzano errori che si verificano sia nelle porzioni hardware sia in quelle software del sistema di computer. Un elenco parziale di test hardware del sistema di computer potrebbe comprendere diagnostica su componenti hardware quali un processore, una memoria, un dispositivo di memorizzazione a disco, un dispositivo audio, un dispositivo grafico, una tastiera, un mouse ed una stampante. L'installazione di software comprende spesso il caricamento di un pacchetto desiderato di software sul sistema di calcolatore, la preparazione di variabili di ambiente appropriato per il computer e la preparazione di file di inizializzazione appropriati per il software caricato. I test software comprendono spesso l'accertamento che una versione desiderata di

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

software è stata installata sul sistema di computer e che sono presenti driver appropriati sul sistema di computer.

E' noto nell'industria installare software e testare sistemi di computer durante la fabbricazione eseguendo una procedura fissa prima che essi siano spediti ai clienti. Per esempio, si crea un dischetto contenente certi test diagnostici per un certo tipo di sistema di computer. Il dischetto comprende file batch lunghi e spesso complicati che dirigono il processo di installazione e di diagnostica del software. Il dischetto contiene inoltre tutti i file eseguibili per eseguire test sul sistema di computer acquistato.

Ogni sistema di computer costruito è dotato di una rispettiva copia di questo dischetto. Questi dischetti accompagnano i sistemi di computer costruiti in fabbrica durante il processo di produzione, test venendo eseguiti sul rispettivo sistema di computer secondo l'ordine inerente nel file batch. Se occorre effettuare una modifica al processo, il file batch viene modificato in modo corrispondente aggiungendo o togliendo porzioni dal codice batch. Tale variazione nel file batch dà come risultato una corrispondente variazione dei

SUZZI NOTARO &
ANTONIELLI D'OUX
s.r.l.

parametri di test (compresa la sequenza nella quale i test sono eseguiti) di ogni successivo sistema di computer fabbricato, dato che ogni sistema di computer condivide la stessa procedura diagnostica del file batch.

Mentre soluzioni diagnostiche di questo tipo hanno presentato un certo grado di utilità nell'aumentare l'affidabilità dei sistemi di computer prima della spedizione, rimane spazio per miglioramenti. Per esempio, dato che i test continuano a diventare sempre più complicati e approfonditi, i file batch e i file eseguibili dei test diagnostici spesso superano le capacità di immagazzinaggio di un dischetto. Inoltre, è spesso difficile o impossibile personalizzare le procedure di test ed installazione di software per un singolo sistema di computer costruito secondo un ordine oppure per una certa famiglia di sistemi di computer senza modificare i test per altri sistemi o famiglie. Inoltre, è difficile o impossibile modificare l'ordine dell'installazione software o di test per un singolo sistema di computer costruito secondo un ordine o per una certa famiglia di sistemi di computer senza modificare l'ordine per altri sistemi e famiglie. Infine, la

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

natura spesso complicata delle attuali strutture di file batch talvolta rende difficile per i costruttori risolvere i problemi o mantenere le procedure di test ed installazione di software rapidamente in modo efficiente. In maniera corrispondente, sarebbe desiderabile stabilire un procedimento perfezionato per installare software e testare sistemi di computer prima che essi siano spediti ai clienti.

L'invenzione

Un procedimento per installare e/o testare software per un sistema di computer costruito secondo un ordine comprende la lettura di una pluralità di descrittori di componenti da un file leggibile da computer. Almeno un descrittore di componente descrive un rispettivo componente del sistema di computer. Una pluralità di fasi sono reperite da una banca dati, almeno una fase essendo associata ad un rispettivo descrittore di componente. Una fase comprende inoltre un rispettivo numero di sequenza. La pluralità di fasi sono sequenziate in un predeterminato ordine secondo i numeri di sequenza per fornire una sequenza di fasi. La sequenza di fasi comprende comandi per installare e/o testare il software sul

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

sistema di computer.

In forme di attuazione preferite, una prima sequenza di fasi di un primo sistema di computer può essere modificata indipendentemente da una seconda sequenza di fasi di un secondo sistema di computer. In aggiunta, la banca dati può essere configurata in modo da associare una prima sequenza di fasi ad una prima famiglia di sistemi di computer ed una seconda sequenza di fasi ad una seconda famiglia di sistemi di computer. La prima sequenza di fasi può essere modificata indipendentemente dalla seconda sequenza di fasi.

Secondo un altro aspetto, un procedimento per installare e/o testare software comprende la ricezione di un ordine per un sistema di computer, il sistema di computer da fabbricare che deve comprendere una pluralità di componenti. L'ordine è convertito in un record di descrittore di sistema leggibile da computer che è descrittivo della pluralità di componenti. Il record di descrittore di sistema è letto con un computer. Una pluralità di fasi sono reperite da una banca dati, una fase essendo associata ad un rispettivo componente. Una fase comprende inoltre un rispettivo numero di sequenza e fase. La pluralità di fasi sono

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

sequenziate in un predeterminato ordine secondo i rispettivi numeri di sequenza e i numeri di fase per fornire una sequenza di fasi. La sequenza di fasi comprende comandi per installare e/o testare software sul sistema di computer durante le fasi di fabbricazione. Le fasi di fabbricazione corrispondono a rispettivi numeri di fase.

Secondo un altro ulteriore aspetto, un procedimento per sequenziare una installazione software comprende la ricezione di un ordine per un sistema di computer, il sistema di computer da fabbricare che deve comprendere una pluralità di componenti. L'ordine è convertito in un record di descrittore di sistema leggibile da computer che è descrittivo della pluralità di componenti. Una modifica del record di descrittore di sistema è permessa utilizzando un patch di descrittore di sistema. Il record di descrittore di sistema è letto con un computer. Si crea una pluralità di oggetti derivati corrispondenti alla pluralità di componenti. Con gli oggetti derivati, si reperiscono una pluralità di fasi da una banca dati. Ogni fase è associata ad un rispettivo componente e comprende un rispettivo numero di sequenza e un numero di fase. La pluralità di fasi

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

sono sequenziate in un ordine predeterminato secondo i rispettivi numeri di sequenza e numeri di fase per fornire una sequenza di fasi. Una modifica alla sequenza di fasi è permessa utilizzando un patch di sequenza di fasi. La sequenza di fasi è scritta in un file di testo leggibile da computer. Il file di testo comprende comandi per installare il software sul sistema di computer durante le fasi di fabbricazione, la fase di fabbricazione corrispondendo a rispettivi numeri di fase.

Preferibilmente, la sequenza di fasi è atta a fornire comandi ripetibili per una lunghezza di tempo definita. Si preferisce inoltre che la frequenza di fasi sia adattata a fornire comandi ripetibili per un numero di interazioni definite.

Il metodo descritto fornisce così un sequenziamento effettivo dell'installazione di software e del test di computer che consente una risoluzione immediata degli errori e una personalizzazione dei sistemi di computer costruiti secondo gli ordini. Il progetto modulare del sequenziamento consente in modo vantaggioso una manutenzione elementare di un sistema di test e la rapida creazione di fasi per nuovi sistemi e famiglie di computer.

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

Questi ed altri scopi, caratteristiche e vantaggi dell'invenzione saranno ulteriormente descritti e risulteranno immediatamente più chiari da una revisione della descrizione dettagliata delle forme di attuazione preferite che segue.

BREVE DESCRIZIONE DEI DISEGNI

La Figura 1 è un diagramma schematico che illustra l'installazione e il test di software.

La Figura 2 è un diagramma schematico dell'installazione e del test di software secondo un'altra forma di attuazione.

La Figura 3A è un diagramma di flusso per convertire un ordine di computer in un record di descrittore di sistema secondo la presente invenzione.

La Figura 3B illustra una porzione di un ordine di computer di esempio, un file di Record Assembly di Base (BAR), e un record di descrittore di sistema.

La Figura 4 è un diagramma di flusso per creare e fornire una sequenza di fasi.

La Figura 5 è un diagramma di flusso più dettagliato per creare una sequenza di fasi.

La Figura 6 illustra la struttura di una banca dati.

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

La Figura 7 è un diagramma di flusso per modificare un record di descrittore di sistema e una sequenza di fasi.

La Figura 8 illustra un esempio di un file di fasi prima di essere eseguito.

La Figura 9 illustra il file di fasi esemplificative dopo essere stato eseguito.

La Figura 10 è un diagramma di flusso del funzionamento di un programma per eseguire una sequenza di fasi.

La Figura 11 è un diagramma di flusso più dettagliato del funzionamento del programma di Figura 10 per eseguire una sequenza di fasi.

DESCRIZIONE DETTAGLIATA

Quanto segue evidenzia una descrizione dettagliata del modo migliore e contemplato per eseguire l'invenzione. La descrizione vuole essere illustrativa e non deve essere presa come limitativa. Nelle figure, gli elementi analoghi o simili possono essere designati dallo stesso numero di riferimento. Nella descrizione, un modulo è definito come comando o insieme di comandi.

La Figura 1 è un diagramma schematico di un sistema di installazione e test di software 90. Durante il funzionamento, si colloca un ordine 92

per acquistare un sistema di computer obiettivo 160 costruito secondo l'ordine. Il sistema obiettivo 160 deve essere fabbricato in modo da contenere una pluralità di componenti hardware e software. Per esempio, il sistema obiettivo 160 potrebbe comprendere una certa marca di unità di disco rigido, un tipo particolare di monitor, una certa marca di processore, ed una particolare versione di un sistema operativo. Prima che il sistema obiettivo 160 sia spedito al cliente, la pluralità di componenti vengono installati e testati. Tale installazione e test del software garantisce in modo vantaggioso un sistema di calcolatore funzionante e affidabile che è pronto a funzionare quando viene ricevuto.

Poiché famiglie diverse di sistemi di computer e singoli componenti di computer diversi richiedono fasi di installazione e test di software diverse, è necessario determinare quali test devono essere eseguiti sul sistema obiettivo 160 e in quale ordine questi test devono essere eseguiti in modo da ottenere un processo di installazione e test di software efficace. Il dispositivo di realizzazione di fasi 140 è un sistema di computer configurato in modo da sequenziare le fasi di installazione e test

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

di software da eseguire sul sistema obiettivo 160. Per sequenziare le fasi di installazione e/o test di software, il dispositivo di realizzazione di fasi 140, e più in particolare il programma di sequenziamento 204 che risiede sul dispositivo di realizzazione di fasi 140, legge dapprima una pluralità di descrittori di componenti dal file di descrittori 96. Il file di descrittori 96 è fornito convertendo un ordine 92, che corrisponde ad un sistema di computer desiderato avente componenti desiderati, in un formato leggibile da computer tramite il modulo di conversione 94.

I descrittori di componenti sono descrizioni leggibili dal computer dei componenti del sistema obiettivo 160 i quali componenti sono definiti dall'ordine 92. Nella forma di attuazione preferita, i descrittori di componenti sono compresi in un file di descrittori chiamato record di descrittore di sistema che è un file leggibile da computer contenente un listato dei componenti, componenti hardware e/o software, da installare sul sistema obiettivo 160. Avendo letto la pluralità di descrittori di componenti, il programma di sequenziamento 204 reperisce una pluralità di fasi di installazione e/o test di software

BUZZI NOTARO &
ANTONELLI D'OULD
s.r.l.

corrispondenti ai descrittori di componenti dalla banca dati 100 sulla connessione di rete 110. La connessione di rete 110 può essere qualsiasi connessione di rete ben nota nella tecnica, quale una rete di area locale, una intranet, o l'internet. Le informazioni contenute nella banca dati 100 possono essere aggiornate tramite una modifica illustrata dalla freccia 130.

Avendo reperito le fasi di installazione e/o test di software appropriate per il sistema obiettivo 160, il programma di sequenziamento 204 sequenzia le fasi in un ordine predeterminato secondo numeri di sequenza corrispondenti ad ogni fase. Avendo sequenziate le fasi richieste per il sistema obiettivo 160, il programma di sequenziamento 204 scrive una serie di file di uscita sul disco di fasi 150. Nella forma di attuazione evidenziata in Figura 1, i file di uscita comprendono file di test contenenti linee di comando appropriate per eseguire le fasi di installazione e/o test di software appropriate sul sistema obiettivo 160. L'esecuzione è effettuata nell'ordine predeterminato secondo i numeri di sequenza corrispondenti ad ogni fase. Il disco di fasi 150 accompagna il sistema obiettivo 160 in

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

fabbrica dove i test sono eseguiti direttamente dal disco di fasi 150 oppure, in alternativa, dal file server 190, connesso al sistema obiettivo 160 tramite la connessione di rete 180. Preferibilmente, la connessione di rete 180 è un dispositivo di rete generico collegato in una corrispondente porta di rete del sistema di computer obiettivo. A seguito dell'esecuzione delle fasi di installazione e test di software, i risultati dell'installazione dei test sono registrati nuovamente nel file server 190 sulla connessione di rete 180.

La Figura 2 è un diagramma schematico del sistema di installazione e test di software 192 conforme ad un'altra forma di attuazione della presente invenzione. Un cliente colloca un ordine 92 per acquistare un sistema di computer obiettivo 160 costruito secondo un ordine. Il sistema obiettivo 160 deve essere fabbricato in modo da contenere una pluralità di componenti i quali componenti possono comprendere componenti sia hardware sia/o software. Prima che il sistema obiettivo 160 sia spedito al cliente, la pluralità di componenti sono installati e testati. Tale installazione e test garantisce in modo vantaggioso

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

un sistema di computer funzionante ed affidabile che è pronto a funzionare dopo essere ricevuto dal cliente.

Per sequenziare le fasi di installazione e test di software, il programma di sequenziamento 204 legge una pluralità di descrittori di componenti dal file di descrittori 96. L'ordine 92 è convertito nel file di descrittore 96 tramite il modulo di conversione 94. I descrittori di componenti sono descrizioni leggibili da computer dei componenti del sistema obiettivo 160. Nella forma di attuazione preferita, i descrittori di componenti sono compresi in un file di descrittori chiamato record di descrittore di sistema, un file leggibile da computer comprendente un listato di ogni componente, sia hardware sia software, da installare sul sistema obiettivo 160. Il record di descrittore di sistema può essere memorizzato direttamente sul file server 202. Il programma di sequenziamento 204 reperisce una pluralità di fasi di installazione e/o test di software corrispondenti ai descrittori di componenti dalla banca dati 100. Avendo reperito le fasi di installazione e/o test di software appropriate per il sistema obiettivo 160, il programma di

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

sequenziamento 204 sequenzia le fasi in un predeterminato ordine secondo i numeri di sequenza corrispondenti ad ogni fase. Avendo sequenziate le fasi richieste per il sistema obiettivo 160, il programma di sequenziamento 204 dirige l'esecuzione delle fasi di installazione e test di software sul sistema obiettivo 160 nell'ordine predeterminato tramite le connessioni di rete 195 e 180. Si desidera che la connessione di rete 200 sia un dispositivo di rete generico collegato ad una corrispondente porta del sistema obiettivo 160. La rete 195 può essere qualsiasi connessione di comunicazione ben nota nella tecnica. A seguito dell'esecuzione delle fasi di installazione e/o test di software, i risultati dell'installazione e dei test sono registrati di nuovo nel file server 202 sulla connessione di rete 200 o memorizzati all'interno di una banca dati appropriata. Come appare chiaro dall'illustrazione, non c'è alcuna necessità di separare il sistema di computer di realizzazione di fasi 140 di Figura 1. In aggiunta, il disco di fasi 150 non è necessario. Piuttosto, è necessario soltanto il disco di boot 220, che è configurato per inizializzare il sistema obiettivo 160, per accompagnare il sistema obiettivo 160 in

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

fabbrica.

Avendo descritto in generale i sistemi di installazione e test di software, si orienterà ora l'attenzione alla descrizione del funzionamento dei sistemi evidenziati nelle Figure 1 e 2 in maggior dettaglio.

La Figura 3A illustra il processo preferito in cui un ordine per un sistema di computer viene convertito in un record di descrittore di sistema leggibile da computer. Più specificamente, nell'elemento 300, si riceve un ordine per un sistema di computer obiettivo. Questo ordine può essere in una qualsiasi di innumerevoli forme. Per esempio, sono possibili formati di ordinamento diversi così come meccanismi di consegna di ordini diversi. Per esempio, gli ordini per un sistema di computer obiettivo possono essere collocati per telefono, tramite posta, o su reti di computer (ad esempio su internet). Indipendentemente dai mezzi di assunzione o dalla forma dell'ordine, l'ordine comprende il tipo di sistema di computer obiettivo che un cliente desidera acquistare e, eventualmente, un listato esplicito dei particolari componenti che il cliente desidera che il sistema di computer obiettivo comprenda. Dopo che si riceve

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

l'ordine, il controllo opera una transizione sul modulo di trasmissione 310 durante la quale l'ordine di sistema di computer obiettivo viene trasmesso su una rete di computer ad un sistema di fabbricazione (non illustrato) che produce il sistema di computer obiettivo. L'ordine di sistema di computer obiettivo è fornito inoltre al sistema di installazione e test di software dove viene concatenato in un programma di conversione nel modulo 320. La rete di computer utilizzata nel modulo 310 può essere di qualsiasi tipo ben noto nella tecnica.

Il programma di conversione converte l'ordine di sistema di computer obiettivo in un record utile per il processo di fabbricazione. Più specificamente, il programma di conversione converte l'ordine di computer dapprima in un record chiamato file BAR nel modulo 330. Preferibilmente il file BAR contiene un identificatore unico che identifica il sistema di computer obiettivo specifico in fabbricazione. Il file BAR contiene inoltre un listato dettagliato dei componenti, che possono comprendere sia hardware sia software, da inserire nel sistema obiettivo. Inoltre si desidera che il file BAR contenga numeri categorici

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

specifici del costruttore o altri identificatori utili per ogni componente. Infine, il file BAR può contenere informazioni specifiche del cliente quali nome, indirizzo e numero di telefono.

A seguito della creazione del file BAR NEL MODULO 330, si crea un record di descrittore di sistema nel modulo 340. Un record di descrittore di sistema, nella forma di attuazione preferita, è un file leggibile da computer che è descrittivo dei componenti hardware e software da inserire nel sistema di computer obiettivo. In una forma di attuazione preferita, il record di descrittore di sistema contiene un elenco di componenti del sistema obiettivo in un formato comprendente etichette hardware, etichette software, etichette di informazioni e commenti. Un'etichetta hardware identifica al programma di sequenziamento 204 che le informazioni che seguono l'etichetta sono relative ad un componente hardware. Analogamente, l'etichetta software identifica le informazioni che seguono l'etichetta come relative ad un componente software. L'etichetta di informazioni indica che seguono informazioni generali. I commenti consentono di inserire varie scritte nel record di descrittore di sistema che sono ignorate dal

SUZZI, NOTARO &
ANTONIELLI D'OUX
S.r.l.

programma di sequenziamento 204. Si desidera che il record di descrittore di sistema sia un file di testo che sia leggibile dalle persone e facile da capire. Tale file consente vantaggiosamente una risoluzione facile dei problemi e la manutenzione del processo di installazione e test. Si potrà apprezzare che il record di descrittore di sistema potrebbe essere qualsiasi elenco di identificatori unici che corrispondono ad un insieme unico di token, per esempio in un esempio semplice, il record di descrittore di sistema può essere un elenco di numeri categorici.

La Figura 3B illustra un ordine di sistema di computer obiettivo 350 esemplificativo, un corrispondente file BAR 360, ed un corrispondente record di descrittore di sistema 370. L'ordine di sistema di computer obiettivo 350 contiene il nome di una famiglia di computer, in questo esempio illustrativo la famiglia "X". Compresi anche nell'ordine di sistema di computer obiettivo 350 vi sono tre componenti hardware esemplificativi comprendenti un processore Pentium®, un'unità a disco rigido, ed un monitor. Il file BAR 360 deriva dall'esecuzione dell'ordine di sistema di computer obiettivo 350 tramite un programma di conversione

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

come illustrato nel modulo 320 di Figura 3A. Il file BAR 360 contiene un identificatore unico per il sistema di computer obiettivo specifico all'interno della famiglia X. Il file BAR 360 comprende inoltre i numeri categorici specifici del costruttore per ciascuno dei componenti elencati nell'ordine di sistema di computer obiettivo. Inoltre, il file BAR 360 contiene un identificatore che indica la quantità desiderata di ogni componente così come una descrizione di test di ogni componente da inserire sul sistema di computer obiettivo. Il sistema 90 utilizza il file BAR 360 per creare il record di descrittore di sistema 370.

Come illustrato, il record di descrittore di sistema 370 contiene anche l'identificatore unico per il sistema di computer obiettivo specifico nell'ambito della famiglia X. Inoltre, il record di descrittore di sistema 370 contiene etichette appropriate, che indicano qui che il processore, l'unità a disco rigido e il monitor sono tutti componenti hardware, piuttosto che software. Il record di descrittore di sistema 370 descrive questi componenti in una descrizione di testo. In aggiunta, il record di descrittore di sistema 370 esemplificativo contiene un'etichetta software che

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

indica che un certo software deve essere installato o testato sul sistema di computer obiettivo appartenente alla famiglia X. Per esempio, l'etichetta software potrebbe indicare che un certo sistema operativo appropriato per il processore Pentium® sia sempre installato sull'unità a disco rigido del sistema di computer obiettivo appartenente alla famiglia X.

In Figura 4, è evidenziato il procedimento generale preferito per sequenziare le fasi di installazione e test di software. Nel modulo 400, si genera l'identificatore unico del sistema di computer obiettivo per il sistema di computer obiettivo 160. Nella forma di attuazione illustrata in Figura 1, un utente seduto al sistema di computer che realizza le fasi 140 fornisce l'identificatore unico (ad esempio l'identificatore BAR che funziona come codice di rintracciamento) nel programma di sequenziamento 204 del dispositivo di realizzazione di fasi 140. In alternativa, nella forma di attuazione di Figura 2, l'identificatore unico viene letto automaticamente nel programma di sequenziamento 204 dopo che si riceve l'ordine di sistema di computer obiettivo.

Nel modulo 410, si localizza un record di

descrittore di sistema corrispondente all'identificare BAR. Nella forma di attuazione di Figura 1, la connessione di rete 110 oppure la connessione di rete 195 localizza il record di descrittore di sistema. Nella forma di attuazione di Figura 2, la connessione di rete 195 localizza il record di descrittore di sistema. Nel modulo 420, il record di descrittore di sistema localizzato è fornito al programma di sequenziamento 204. Nella forma di attuazione di Figura 1, il programma di sequenziamento risiede sul sistema di computer di realizzazione di fasi 140 mentre nella forma di attuazione di Figura 2, il programma di sequenziamento risiede sul file server 202. Il programma di sequenziamento 204 lavora insieme alla banca dati 100 (delle Figure 1 e 2) per sequenziare le fasi di installazione e test di software per il sistema di computer obiettivo 160. Quando vengono sequenziate le fasi di installazione e test di software appropriate per il particolare sistema di computer obiettivo, il programma di sequenziamento 204 produce gli file di uscita come illustrati nel modulo 430.

Nella forma di attuazione illustrata in Figura 1, i file di uscita sono scritti preferibilmente

sul disco di fasi 150 (vedere Figura 1) in sei file separati. Questi file comprendono (1) un file di fasi, (2) un file Setenv.bat, (3) un file Qt.txt, (4) un file Et.txt, (5) un Etlast.txt, e (6) un file Ft.txt. Si desidera che il file di fasi sia un file di testo ASCII comprendente un elenco di linee di comandi appropriate per eseguire le fasi di installazione e test di software per il sistema di computer obiettivo ordinato. In una forma di attuazione preferita, il file di fasi comprende inoltre comandi che possono essere fatti girare ad anello. Più specificamente, il file di fasi consente di ripetere i comandi per un numero definito di iterazioni o per una durata di tempo definita. Tale formato consente vantaggiosamente di ripetere le fasi di installazione o test di software n maniera calcolata e predeterminata. Il file Setenv.bat stabilisce preferibilmente variabili di ambiente sul sistema di computer obiettivo. Si potrà apprezzare che in un modo di funzionamento sono necessari soltanto il file Step e il file Setenv.bat per l'installazione di test. Il file Step e il file Setenv.bat sono file di scritti di testo ASCII contenenti un elenco di linee di comandi appropriate per eseguire le fasi

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

di installazione e test per il sistema di computer obiettivo. Il file Qt.txt, Et.txt, Etlast.txt e Ft.txt sono tutti preferibilmente file di test ASCII contenenti un elenco di linee di comandi appropriato per eseguire le fasi di installazione e test di software per il sistema di computer obiettivo nelle fasi di fabbricazione di Test Rapido (Qt), Test Esteso 1 (Et), Test Esteso 2 (Etlast) e Test Finale (Ft) del sistema di computer obiettivo.

Nella forma di attuazione di Figura 2, d'altra parte, i file di uscita non sono scritti in un disco di fasi come illustrato in Figura 1. Invece, i file di uscita risiedono sul file server 202 o sul file server 190, dove sono utilizzati per dirigere l'esecuzione delle fasi di installazione e/o test di software sul sistema di computer obiettivo 160.

La Figura 5 illustra uno schema più dettagliato del funzionamento del programma di sequenziamento 204 illustrato nelle Figure 1 e 2. Nel modulo 500, si fornisce un record di descrittore di sistema corrispondente al sistema di computer obiettivo 160 al programma di sequenziamento 204. Nel modulo 510, si legge un

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

descrittore di componente dal record di descrittore di sistema. Ogni descrittore di componente descrive un rispettivo componente, hardware o software, nel sistema di computer obiettivo.

Passando alla Figura 3B, la linea del record di descrittore di sistema comprendente il processore Pentium® nel modulo 370 è un descrittore di componente esemplificativo. Nel modulo 520, il programma di sequenziamento 204 esemplifica una pluralità di oggetti derivati corrispondenti alla pluralità di componenti del sistema di computer obiettivo 160. Nella forma di attuazione preferita, questi oggetti derivati sono utilizzati per memorizzare informazioni (ottenute dalla banca dati 100) relative alle fasi di installazione e test di software che devono essere fatte girare sul sistema di computer obiettivo 160. Nel modulo 550, le fasi di installazione e test di software associate ai rispettivi componenti del sistema di computer obiettivo 160 sono reperite dalla banca dati 100 e memorizzate nell'oggetto derivato appropriato. Nella forma di attuazione di Figura 1, le fasi sono reperite tramite la connessione di rete 110 mentre nella forma di attuazione di Figura 2, le fasi possono essere reperite direttamente dal file

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

server 202. Per descrivere come le fasi sono reperite dalla banca dati 100 nella forma di attuazione preferita si richiede una descrizione della costruzione preferita di tale banca dati.

La Figura 6 illustra il progetto della banca dati 100. La banca dati 100 associa sequenze di fasi di installazione e/o test di software, in un predeterminato ordine, con famiglie di sistemi di computer. Inoltre, la banca dati 100 è configurata in modo da associare componenti di sistemi di computer con famiglie di sistemi di computer. Ancora di più, la banca dati 100 associa le fasi di installazione e/o test di software con componenti di sistemi di computer.

La banca dati 100 è preferibilmente una banca dati razionale. La banca dati 100 contiene parecchie tabelle, contenenti ciascuna attributi adatti a creare le associazioni sopra citate.

La banca dati 100 contiene la tabella Step 102, la tabella SysFamily 104, la tabella Sys_Step_Seq 106, la tabella Component 108, la tabella Sys_Comp 112, e la tabella Comp_Step 114. Nella forma di attuazione preferita, ogni tabella contiene un elenco di attributi, gli attributi sottolineati servendo come chiave primaria.

La tabella Step 102 contiene un insieme di fasi di installazione e test di software che sono condivise tra componenti diverse di tutte le famiglie di computer. Nella costruzione preferita, la tabella Step 102 ha attributi comprendenti StepID, Phase, Name, Cmd, CmdType, AfterCode e MaxInstance. StepID è un numero di identificazione unico per ogni fase di installazione o test di software. Phase indica quale fase di fabbricazione deve eseguire la fase. Per esempio Phase è un intero scelto in modo da corrispondere a quattro fasi di fabbricazione di sistemi di computer comprendenti (1) Test Rapido, (2) Test Esteso 1, (3) Test Esteso 2, e (4) Test Finale. Name è una stringa che assegna un nome che è descrittivo della fase. Cmd è una stringa che assegna una linea di comando eseguibile per eseguire la fase di installazione o test di software sul sistema obiettivo 160 (illustrato nelle Figure 1 e 2). AfterCode è un identificatore che determina se è necessaria una interruzione o una nuova inizializzazione dopo che si esegue la fase di installazione o test di software. MaxInstance è un identificatore che indica il numero massimo di volte consentite per cui si può eseguire la fase.

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

Infine, ClassID identifica un certo tipo di componente che è associato alla fase di installazione o test di software.

La tabella SysFamily 104 identifica ogni famiglia di sistemi di computer con un intero di identificazione specificato nell'attributo SysID. E' compresa anche nella tabella SysFamily una stringa che identifica il nome della famiglia.

La tabella Sys_Step_Seq 106 è una tabella relazione che contiene relazioni tra la tabella Step 102 e la tabella SysFamily 104. La tabella Sys_Step_Seq 106 comprende un intero di identificazione di famiglia specificato nel SysID di attributo per una particolare famiglia di sistemi di computer (dalla tabella SysFamily 104), un intero di identificazione di fase specificato nel StepID di attributo (dalla tabella Step 102) che identifica un particolare insieme di fasi appropriate per tale famiglia, ed un numero di sequenza. Il numero di sequenza è contenuto preferibilmente all'interno dell'attributo SeqNum che rappresenta un predeterminato ordine in cui devono essere eseguite le fasi associate ad una particolare famiglia. Gli ingegneri di test segnano i numeri di sequenza, unici all'interno di ogni

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

fase di fabbricazione, in un ordine scelto in modo che sia più efficace per un particolare sistema obiettivo. Si potrà apprezzare che si possono utilizzare altri modi di assegnare i numeri di sequenza.

La tabella Component contiene tutti i possibili componenti che sono compresi all'interno dei sistemi di computer fabbricati. Gli attributi di questa tabella sono preferibilmente CompID che assegna un identificatore ad ogni componente, NameDesc che assegna un nome di stringa ad ogni componente, e CallsId che fa riferimento al tipo di componente (ad esempio unità a disco rigido, unità CD-ROM).

La tabella Sys_Comp 112 è una tabella relazionale contenente relazioni tra una famiglia di sistemi di computer ed un insieme di componenti che possono essere inseriti in tale famiglia. Gli attributi della tabella Sys_Comp 112 comprendono un intero di identificazione di famiglia di computer specificato nell'attributo SysID (dalla tabella SysFamily 104) ed un intero di identificazione di componente specificato nell'attributo CompID (dalla tabella Component 108).

La tabella Comp_Step 114 è una tabella

relazione contenente relazioni tra un componente ed un insieme di fasi di installazione e test di software appropriati per tale componente. Gli attributi della tabella Comp_Step 114 comprendono un intero di identificazione di componente specificato nell'attributo CompID (dalla tabella Component 108) ed un intero di identificazione di fase specificato nell'attributo StepID (dalla tabella Step 102).

Il sistema di computer obiettivo esemplificativo illustrato in Figura 3B sarà utilizzato per illustrare come si utilizza il progetto di banca dati evidenziato in precedenza per reperire le fasi di installazione e test di software. L'identificatore di famiglia di computer nel record di descrittore di sistema che identifica la famiglia X è associato al SysID corrispondente alla famiglia X nella tabella SysFamily 104. La tabella Component 108 è utilizzata per controllare se i componenti del sistema di computer obiettivo elencati nell'ordine di sistema di computer obiettivo sono legali. In altre parole, il programma di sequenziamento e la banca dati determinano se il processore, l'unità a disco rigido, il monitor e il software contenuti nel

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

record di descrittore di sistema di Figura 2B hanno corrispondenti inserimenti e corrispondenti interi specificati da CompID nella tabella Component 108. Se un componente non è legale (cioè se un componente nel record di descrittore di sistema non è contenuto nella tabella Component 108, si crea un indicatore di errore. La tabella Sys_Comp 112 è una tabella relazionale che contiene mappature dalla tabella Component 108 e dalla tabella SysFamily 104. La tabella Sys_Comp 112 contiene tutti i componenti legali che possono essere inseriti su un sistema di computer obiettivo appartenente alla famiglia X. Così, la tabella Sys_Comp 112 può essere utilizzata per controllare se tutti i componenti del sistema obiettivo sono legali. In altre parole, il programma di sequenziamento e la banca dati determinano se il processore, l'unità a disco rigido, il monitor e il software contenuti nel record di descrittore di sistema di Figura 3B hanno corrispondenti relazioni nella tabella Sys_Comp 112. Se un componente non è legale (cioè se un componente nel record di descrittore di sistema non può essere compreso su un sistema obiettivo appartenente alla famiglia X), si crea un indicatore di errore.

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

Nella tabella relazionale Sys_Step_Seq 106 risiedono le mappature dalla tabella Step 102 e dalla tabella Sys_Family 104. La tabella Sys_Step_Seq 106 contiene tutte le fasi di installazione e test di software che possono essere fatte funzionare legalmente sui sistemi di computer obiettivo appartenenti alla famiglia X. Inoltre, è in questa tabella Sys_Step_Seq 106 che i numeri di sequenza e fase sono associati ad ogni fase di installazione e test di software. Questi numeri di sequenza e fase rappresentano l'ordine appropriato in cui le fasi devono essere fatte funzionare per una particolare famiglia di sistemi di computer. Pertanto, la tabella Sys_Step_Seq 106 contiene un listato di fasi da far funzionare sui sistemi di computer obiettivo della famiglia X così come i numeri di sequenza e fase che rappresentano un predeterminato ordine in cui le fasi devono essere eseguite.

La tabella Comp-Step 114 è una tabella relazionale che contiene mappature dalla tabella Component 108 e dalla tabella Step 102. La tabella CompStep 114 contiene le fasi di installazione e test di software che devono essere eseguite per il processore, l'unità a disco rigido, il monitor e il

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

software del sistema di computer obiettivo.

Per reperire le fasi di installazione e test di software associate ai rispettivi componenti da inserire sul sistema obiettivo è necessaria l'esecuzione di una operazione di unione sulla tabella Sys_Comp 112 e sulla tabella Comp_Step 114 per ottenere un insieme intermedio che elenca le fasi da far funzionare sui componenti del sistema di computer obiettivo 160.

L'operazione di unione dà come risultato un elenco di fasi da far girare sul processore, sull'unità a disco rigido, sul monitor e sul software elencati nel record di descrittore di sistema illustrato in Figura 3B. Il risultato dell'unione della tabella Sys_Comp 112 e della tabella Comp_Step 114 viene quindi unita alla tabella Sys_Step_Seq 106 che contiene tutte le fasi per la famiglia X. Il risultato di questa operazione di unione comprende informazioni di sequenziamento sotto forma di numeri di sequenza e numeri di fase, i numeri di sequenza essendo unici all'interno di una fase particolare. Così, un unione di tre tabelle della tabella Sys_Comp 112, della tabella Comp_Step 114 e della tabella Sys_Step_Seq 106 produce le fasi di installazione

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

e test di software appropriate così come le informazioni di sequenziamento sotto forma di numeri di sequenza e fase per installare e/o testare software sul sistema di computer obiettivo 160.

Se il risultato della prima operazione di unione (l'unione della tabella Sys_Comp 112 e della tabella Comp_Step 114) è un insieme vuoto, si crea una condizione di errore, dato che un insieme vuoto segnala che un componente da inserire sul sistema obiettivo non appartiene alla famiglia elencata sul record di descrittore di sistema. Un esempio di ciò è illustrativo. Si consideri che un record di descrittore di sistema indica correttamente che un sistema di computer obiettivo appartiene alla famiglia X. Si ipotizzi, tuttavia, che il record di scrittura di sistema indichi in modo non corretto che una unità a disco rigido (unità a disco rigido Z) appartenente soltanto a sistemi obiettivo nella famiglia X debba essere inserita sul sistema obiettivo che si trova nella famiglia Y. In tal caso, la tabella Comp_Step 114 contiene le fasi associate all'unità a disco rigido Z. La tabella Sys_Comp 112 contiene i componenti associati alla famiglia Y. Così, l'unione della

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

tabella Comp_Step 114 con la tabella Sys_Comp 112 produce un insieme vuoto, dato che l'unità a disco rigido Z non è un componente associato alla famiglia Y (invece, esso è associato soltanto alla famiglia X). Come appare chiaro dal suddetto esempio, il progetto preferito della banca dati consente in modo vantaggioso di accertarsi che un sistema obiettivo di una certa famiglia contenga soltanto i componenti appropriati per tale famiglia.

Con riferimento di nuovo alla Figura 5, dopo che si reperiscono le fasi associate ai componenti da inserire nel sistema obiettivo, il programma di sequenziamento 204 prepara variabili di ambiente per il sistema di computer obiettivo nel modulo 560 leggendo il record di descrittore di sistema e creando un file di ambiente corrispondente ai componenti da inserire sul sistema obiettivo. Per esempio, il record di descrittore di sistema illustrato in Figura 3B viene letto e si potrebbe preparare una variabile di ambiente quale "set cpu=pentium" corrispondente al componente hardware di processore del record di descrittore di sistema.

Nel modulo 570 di Figura 5, la pluralità di fasi di installazione e test di software reperite,

reperate dall'unione di tre tabelle sopra descritte, vengono sequenziate nell'ordine predeterminato. Questo sequenziamento avviene secondo i rispettivi numeri di sequenza e numeri di fase per fornire una sequenza di fasi. Il sequenziamento stesso viene realizzato utilizzando uno qualsiasi tra molti algoritmi di ordinamento ben noti nella tecnica.

Nel modulo 580, il programma di sequenziamento 204 invia in uscita i file. Come citato in precedenza, i file di uscita sono scritti preferibilmente nel disco di fasi 150 (vedere Figura 1) in sei file separati nella forma di attuazione illustrata in Figura 1. Questi file comprendono (1) un file di fasi, (2) un file Setenv.bat, (3) un file Qt.txt (4), un file Et.txt (5) un file Etlast.txt e (6) un file Ft.txt. Si desidera che il file di fasi sia un file di test ASCII. In una forma di attuazione preferita, il file di fasi comprende inoltre comandi che possono essere eseguiti ad anello. Più specificamente, il file di fasi consente di ripetere i comandi per un numero definito di iterazioni o per una durata di tempo definita. Il file Setenv.bat stabilisce le variabili di ambiente sul sistema di computer

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

obiettivo. Il file di fasi contiene le fasi da eseguire rispettivamente durante le fasi di fabbricazione di Test Rapido (Qt), Test Esteso 1 (Et), Test Esteso 2 (Etlast), e Test Finale (Ft) del sistema di computer obiettivo. Nella forma di attuazione di Figura 2, d'altra parte, gli file di uscita non sono scritti in un disco di fasi come illustrato in Figura 1. Invece, gli file di uscita risiedono sul file server 202 o sul file server 190, dove possono essere utilizzati per dirigere l'esecuzione delle fasi di installazione e test di software sul sistema di computer obiettivo 160.

Passando ora alle Figure 1 e 2, la freccia 130 illustra il fatto che le modifiche possono essere effettuate sulla banca dati 100. Per esempio, se si crea una nuova famiglia di sistemi di computer, si può modificare la banca dati 100 di conseguenza. Più specificamente, alla nuova famiglia viene assegnato un identificatore di nuova famiglia in SysID della tabella SysFamily 104 e si assegna un nome alla nuova famiglia nell'attributo Name della tabella Sys_Family 104. Si aggiunge un elenco di fasi di installazione e fasi di test di software alla tabella Sys_Step_Seq 106, queste fasi rappresentando quali fasi occorre eseguire, e in

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

quale ordine predeterminato, sulla nova famiglia di sistemi di computer. Se la nuova famiglia di sistemi di computer condivide parecchie somiglianze con una famiglia esistente, è probabile che gli inserimenti per la famiglia esistente nella tabella Sys_Step_Seq 106 possono essere modificati per produrre inserimenti per la nuova famiglia. Se occorre creare qualche nuova fase per la nuova famiglia di sistemi di computer, queste fasi sono aggiunte alla tabella Step 102. Analogamente, se qualche nuovo componente accompagna la nuova famiglia di sistemi di computer, questi componenti vengono aggiunti alla tabella Component 108. La tabella Comp_Step 114 viene aggiornata per associare ogni componente dalla nuova famiglia di sistemi di computer alle fasi appropriate per la sua installazione e test di software. Se la nuova famiglia utilizza soltanto componenti già presenti nella banca dati, non occorre modificare questa tabella. Si aggiorna la tabella Sys_Comp 112 in modo che un elenco di componenti consentiti che possono essere inseriti sulla nuova famiglia si trovi nella banca dati. In particolare, occorrerebbe associare il SysID del nuovo sistema di computer al CompID di ogni componente

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

consentito. Di nuovo, si potrebbe fare ciò copiando e quindi modificando un inserimento esistente di una famiglia più vecchia di sistemi di computer.

Si potrà apprezzare che nella costruzione di una banca dati secondo la forma di attuazione preferita, sono previsti certi vantaggi significativi. In particolare, il progetto modulare della banca dati consente in modo vantaggioso una facile predisposizione delle fasi di installazione e test di software per nuove famiglie di sistemi di computer. In aggiunta, le fasi di installazione e test di software e una particolare famiglia di sistemi di computer o per un particolare componente possono essere modificati in modo indipendente da altre fasi di installazione e test di software.

La Figura 7 illustra come un record di descrittore di sistema ed una sequenza di fasi possono essere aggiustate per consentire modifiche modulari in un processo di installazione e test di software in conformità con l'invenzione. Nel modulo 600, si crea un record di descrittore di sistema. Nel modulo 610, il record di descrittore di sistema viene modificato utilizzando un patch di record di descrittore di sistema. Nella forma di attuazione preferita, questo patch è modulare, consentendo di

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

creare patch per uno specifico sistema di computer obiettivo, una particolare famiglia di sistemi di computer, o per un particolare componente. Per esempio, se un costruttore volesse sostituire una marca di unità a disco rigido con un'altra per una certa famiglia di sistemi di computer in un certo giorno, si può formare un patch che modificherebbe tutti i record di descrittore di sistema contenenti unità a disco rigido da sostituire per realizzare la sostituzione nel modulo 610. Nel modulo 620, si determina una sequenza di fasi come evidenziato in precedenza. Nel modulo 630, la sequenza di fasi viene modificata utilizzando un patch di sequenza di fasi. Nella forma di attuazione preferita, questo patch è modulare, consentendo di creare patch per uno specifico sistema di computer obiettivo, una particolare famiglia di sistemi di computer o per un particolare componente. Per esempio, se un costruttore volesse far girare una fase di test prima di un'altra per un certo componente in un certo giorno, si potrebbe formare un patch che modificherebbe tutte le sequenze di fasi contenenti le fasi il cui ordine deve essere modificato e cambiare in modo corrispondente l'ordine di esecuzione nel modulo 640.

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

Si sposterà ora l'attenzione all'esecuzione della sequenza di fasi sul sistema obiettivo 160. Le fasi di installazione e test di software sono eseguite sul sistema di computer obiettivo 160 utilizzando un programma che legge, interpreta ed esegue la sequenza di fasi corrispondente al sistema di computer obiettivo. Nella forma di attuazione preferita, questo programma è chiamato Runstep ed è collocato sul disco di fasi 150 nella forma di attuazione di Figura 1 e sul file serve 202 nella forma di attuazione di Figura 2.

La Figura 8 illustra una porzione di una sequenza di fasi contenuta in un file di fasi prima che siano eseguite le fasi di installazione e test di software. Come citato in precedenza, la sequenza di fasi comprende comandi per installare il software e/o per testare il sistema di computer obiettivo costruito secondo l'ordine. In aggiunta, la sequenza di fasi nelle file di fasi consente di ripetere i comandi per un numero definito di iterazioni o per una durata di tempo definita. Inoltre, il file di fasi può contenere commenti, ignorati dal programma Runstep. Nel file di fasi, si utilizzano segni 800 per separare i campi della sequenza di fasi. L'elemento 810 sono comandi per

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

testare il sistema di computer obiettivo 160. I comandi comprendono, per esempio, un comando per testare la memoria e per testare dispositivi di una piccola interfaccia di sistema di computer (SCSI). Come si può vedere dalla figura, ogni comando può comprendere interruttori quali '-o' appropriati per il particolare ambiente di prova. L'elemento 120 è un commento che è ignorato dal programma Runstep. L'elemento 810c è un comando che viene fatto funzionare ad anello per tempo. Nella costruzione preferita, l'istruzione 'begin_time_loop' indica il punto di partenza di un anello. L'istruzione 'end_time_loop' indica il punto finale di un anello. L'istruzione 'begin_time_loop' si combina con un campo che designa la durata di tempo in cui operare le iterazioni attraverso l'anello. Qui, per esempio, il comando 810c viene eseguito per un'ora e trenta minuti. L'elemento 810d è un comando che viene eseguito ad anello secondo un certo numero di iterazioni. Nella forma di attuazione preferita, il comando 'begin_iterate_loop' istruisce il programma Runstep ad eseguire un anello iterativo. Il comando 'end_iterate_loop' segnala la fine dei comandi di anello. Qui, il comando 810d viene eseguito tre volte.

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

Quando un programma Runstep esegue la sequenza di fasi, il programma Runstep colloca informazioni di marcatura temporale nel file di fasi, consentendo in modo vantaggioso una facile risoluzione dei problemi e un rintracciamento del processo di installazione e test di software.

La Figura 9 illustra una porzione della sequenza di fasi di Figura 8 dopo che le fasi sono eseguite. Come illustrato, il programma Runstep inserisce informazioni di indicazioni temporali nella sequenza di fasi. L'elemento 830 illustra quando ha avuto inizio il test di memoria, e l'elemento 832 illustra quando ha avuto termine tale test. L'elemento 834 illustra quando ha avuto inizio l'ultima iterazione del test. L'elemento 836 e 838 illustrano quando il test scsiHD ha avuto inizio ed è terminato, rispettivamente. L'elemento 840 conferma che l'anello iterativo è stato eseguito tre volte. Infine, gli elementi 842 e 844 illustrano quando l'ultima iterazione del test scsiCD ha avuto inizio ed è terminata, rispettivamente. L'inserimento di informazioni di indicazioni temporali adiacenti al comando che è stato eseguito consente in modo vantaggioso una risoluzione dei problemi e un rintracciamento

efficienti del processo di installazione e test di software.

La Figura 10 illustra il flusso generale preferito del programma Runstep. Il programma Runstep 860 viene fatto girare in un anello con un file batch Runstep 870. Il programma Runstep 860 legge e interpreta una fase in una sequenza di fasi e scrive il comando che deve essere eseguito dalla sequenza di fasi nel file batch 870. Il file batch 870 viene quindi fatto girare, eseguendo la fase sul sistema di computer obiettivo 160. Al completamento di una fase, il controllo ritorna dal file batch al programma Runstep 860 che quindi legge e interpreta la linea successiva della sequenza di fasi.

La Figura 11 illustra un flusso più dettagliato del programma Runstep. Come illustrato nel modulo 900, il programma Runstep controlla dapprima per vedere se esiste un file chiamato Re_Run.bat. Si crea un file Re_Run.bat prima che un qualsiasi comando venga eseguito da una sequenza di fasi ed esso viene tolto dopo il completamento positivo del comando. L'esistenza di Re_Run.bat indica al programma Runstep nel modulo 900 che l'ultima passata di comando non è stata completata

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

positivamente. Così Re_Run.bat funziona come indicazione di inizio dell'esecuzione. Se esiste Re_Run.bat, ad un operatore viene chiesto nel modulo 904 se il processo di installazione per il software debba continuare o meno o se l'operatore preferisce invece eseguire la ricerca degli errori. Se un operatore sceglie di continuare, il controllo passa al modulo di esecuzione 928 dove si riesegue il file Runstep.bat. (Questa condizione è la scelta di default se non viene scelta in modo affermativo nessuna opzione). Se si sceglie l'opzione di ricerca e risoluzione di errori, si esegue la ricerca di risoluzione di errori come è ben noto nella tecnica.

Se non esiste Re_Run.bat, il programma Runstep determina che l'ultimo comando è stato completato correttamente, e il controllo viene passato al modulo 910, dove si legge una linea della sequenza di fasi, preferibilmente contenuta in file di fasi. Il programma Runstep legge la linea e determina se esiste un'indicazione temporale di inizio e di fine del modulo 912. Se esiste un'indicazione di inizio e di fine, il programma Runstep determina, nel modulo 914, se esiste soltanto un'indicazione

temporale di inizio per la linea che il programma Runstep sta leggendo. Se esiste soltanto un'indicazione temporale di inizio, il programma Runstep ipotizza nel modulo 916 che una fase di installazione o test di software sia appena finita e riempie un'indicazione temporale di fine nel modulo 918. Dopo aver riempito l'indicazione temporale di fine, il controllo ritorna al modulo 900.

Se si ha più di un semplice indicazione temporale di inizio per la linea che il programma Runstep sta leggendo, il programma Runstep determina nel modulo 906 se esiste sia una indicazione temporale di inizio sia una di fine. Se è così, il programma Runstep ipotizza nel modulo 908 che la fase sia stata eseguita e il controllo ritorna nel modulo 900. Se il programma Runstep non incontra alcuna indicazione temporale di inizio o di fine nel modulo 912, il programma Runstep riempie l'indicazione temporale di inizio nel modulo 920 e si prepara a far girare la fase sulla linea della sequenza di fasi che il programma Runstep sta leggendo.

Nel modulo 922, il programma Runstep determina se il comando da far girare è memorizzato su una

unità locale (il file di fasi controlla quale unità nel sistema è l'unità locale). L'unità locale può essere, ad esempio, il disco di fasi, un'unità a disco rigido del sistema obiettivo, un'unità RAM del sistema obiettivo, o un'unità di rete. Se il comando non è posto sull'unità locale, il programma Runstep ipotizza che il test da far girare sia contenuto su un file server da qualche parte su una rete. Il programma Runstep determina nel modulo 932 se il programma Runstep è già connesso a tale rete. Se non così, il programma Runstep, nel modulo 936, incorpora un comando nel Runstep.bat per il collegamento alla rete. Pertanto si effettua una connessione di rete prima che Runstep.bat esegua la fase sul sistema di computer obiettivo 160 sulla connessione di rete 180.

A seguito del modulo 936, il controllo viene passato al modulo 926. Se il programma Runstep è già collegato nella rete, il programma Runstep, durante il modulo 934, toglie i comandi da Runstep.bat per il collegamento alla rete, dato che una fase di collegamento aggiuntiva non è necessaria se esiste già una connessione di rete. Il controllo viene quindi passato al modulo 926. Se succede che la fase da far girare si trova sul

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

disco di fase 150, non occorre che il programma Runstep si colleghi nella rete. Così, nel modulo 924, il programma Runstep toglie i comandi da Runstep.bat per il collegamento alla rete. Il controllo viene quindi passato al modulo 926. Nel modulo 926, il programma Runstep incorpora il comando appropriato da far girare in Runstep.bat e in Re_run.bat. Il comando così incorporato viene preso dalla sequenza di fasi, contenuta preferibilmente nel file di fasi. Nel modulo 928, la fase viene eseguita facendo girare Runstep.bat e, se viene eseguito positivamente, si cancella Re_run.bat. Se la fase non viene eseguita positivamente, il file Re_Run.bat non viene cancellato e il controllo si trasferisce allo stato di guasto 929. Il controllo viene quindi restituito al modulo 900 cosicché si possa leggere un'altra linea dalla sequenza di fasi. Questo processo continua fino a quando tutte le fasi di installazione e test di software vengono completate.

All'esecuzione della sequenza di fasi, il sistema obiettivo viene testato e si installa il software. Nella forma di attuazione di Figura 1, si può far girare un insieme selezionato di test dal

disco di fasi 150 direttamente, ma la maggior parte dei test vengono fatti girare dal file server 190 sulla connessione di rete 180. L'esecuzione di test dal file server 190 elimina vantaggiosamente le limitazioni imposte dalle capacità di memorizzazione dei dischi floppy quali il disco di fasi 150.

Nella forma di attuazione di Figura 2, le fasi vengono eseguite dal file server 190 sulla connessione di rete 180. Un disco floppy, qui il disco di boot 220, è necessario soltanto per inizializzare il sistema di computer obiettivo 160. Tale sistema semplifica vantaggiosamente il processo di installazione e test di software.

Passando ancora una volta alle Figure 1 e 2, la freccia 210 indica che i risultati dall'installazione dal test di software possono essere registrati di nuovo sul file server 190 o sul file server 202. I risultati comprendono preferibilmente il fatto se tutte le fasi sono state completate positivamente e quali tipi di guasti (se esistono) si sono incontrati. La registrazione dei risultati potrebbe comprendere il semplice salvataggio o scrittura di una versione modificata del file di fasi a seguito

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

dell'esecuzione della sequenza di fasi, dato che come discusso in precedenza il file di fasi viene marcato in modo temporale dal programma Runstep. Tale sistema consente vantaggiosamente capacità di ricerca e risoluzione degli errori migliori durante la fabbricazione dei sistemi di computer.

Mentre particolari forme di attuazione della presente invenzione sono state illustrate e descritte, risulterà ovvio agli esperti nel ramo che si possono effettuare cambiamenti e modifiche senza discostarsi dalla presente invenzione nei suoi aspetti più ampi e, pertanto, le rivendicazioni allegate servono a racchiudere dentro al loro campo di protezione tutti questi cambiamenti e modifiche come essi ricadono nell'ambito dello spirito e del campo di protezione veri della presente invenzione.

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

RIVENDICAZIONI

1. - Procedimento per installare software su un sistema di computer, il procedimento comprendendo le fasi di:

leggere una pluralità di descrittori di componenti da un file leggibile da computer, almeno un descrittore di componenti descrivendo un rispettivo componente del sistema di computer;

leggere una pluralità di fasi da una banca dati, una fase essendo associata ad un rispettivo descrittore di componenti e comprendente un rispettivo numero di sequenza; e

sequenziare una pluralità di fasi in un predeterminato ordine secondo i numeri di sequenza per fornire una sequenza di fasi, la sequenza di fasi comprendendo almeno un comando per installare software sul sistema di computer.

2. - Procedimento secondo la rivendicazione 1, in cui almeno un rispettivo componente è un componente hardware.

3. - Procedimento secondo la rivendicazione 1, in cui almeno un rispettivo componente è un componente software.

4. - Procedimento secondo la rivendicazione 1, in cui almeno una della pluralità di fasi è una

fase di installazione di software.

5. - Procedimento secondo la rivendicazione 1, comprendente inoltre la creazione di una pluralità di oggetti derivati corrispondenti alla pluralità di descrittori di componenti.

6. - Procedimento secondo la rivendicazione 1, in cui il predeterminato ordine è secondo un ordinamento sequenziale dei numeri di sequenza.

7. - Procedimento secondo la rivendicazione 1, in cui la pluralità di fasi comprendono inoltre rispettivi numeri di fase e in cui il sequenziamento della pluralità di fasi in un predeterminato ordine è inoltre secondo i numeri di fasi.

8. - Procedimento secondo la rivendicazione 1, comprendente inoltre la preparazione di variabili di ambiente corrispondenti alla pluralità di componenti.

9. - Procedimento secondo la rivendicazione 1, comprendente inoltre la scrittura della sequenza di fasi in un supporto di memorizzazione non volatile configurato in modo da accompagnare il sistema di computer durante la fabbricazione.

10. - Procedimento secondo la rivendicazione 1 comprendente inoltre la fabbricazione del sistema

SUZZI, NOTARO &
ANTONIELLI D'OUIX
S.r.l.

di computer avente la pluralità di componenti.

11. - Procedimento per sequenziare l'installazione di software per un sistema di computer obiettivo, il procedimento comprendendo le fasi di:

ricevere un ordine per il sistema di computer obiettivo, il sistema di computer obiettivo che deve comprendere una pluralità di componenti;

convertire l'ordine in un file leggibile da computer, il file essendo descrittivo della pluralità di componenti;

leggere il file;

reperire una pluralità di fasi da una banca dati secondo il file, una fase essendo associata ad un rispettivo componente e comprendendo un rispettivo numero di sequenza e un rispettivo numero di fase;

sequenziare la pluralità di fasi in un predeterminato ordine secondo i rispettivi numeri di sequenza e numeri di fase per fornire una sequenza di fasi, la sequenza di fasi comprendendo comandi per installare software sul sistema di computer obiettivo durante le fasi di fabbricazione, le fasi di fabbricazione corrispondendo a rispettivi numeri di fase.

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

12. - Procedimento secondo la rivendicazione 11, in cui si crea una condizione di errore se l'insieme intermedio è vuoto.

13. - Procedimento secondo la rivendicazione 11, comprendente inoltre la preparazione di variabili di ambiente corrispondenti alla pluralità di componenti.

14. - Procedimento secondo la rivendicazione 11, in cui la sequenza di fasi è atta a fornire comandi ripetibili per un periodo di tempo definito.

15. - Procedimento secondo la rivendicazione 11, in cui la sequenza di fasi è atta a fornire comandi ripetibili per un numero di iterazioni definito.

16. - Procedimento secondo la rivendicazione 11, comprendente inoltre la scrittura della sequenza di fasi in un dispositivo di memorizzazione non volatile configurato in modo da accompagnare il sistema di computer durante la fabbricazione.

17. - Procedimento per sequenziare l'installazione di software per un sistema di computer obiettivo appartenente ad una certa famiglia, il procedimento comprendendo le fasi di:

riceveré un ordine per il sistema di computer obiettivo, il sistema di computer obiettivo che deve comprendere una certa pluralità di componenti comprendente i componenti hardware e componenti software:

convertire l'ordine in un file leggibile da computer, il file essendo descrittivo della certa pluralità di componenti;

leggere il file;

unire una prima tabella di banca dati contenente tutti i componenti appartenenti ad una certa famiglia ad un seconda tabella di banca dati contenente tutte le fasi di installazione di software da eseguire sulla certa pluralità di componenti, in cui l'unione produce un insieme intermedio;

unire l'insieme intermedio con una terza tabella di banca dati contenente tutte le fasi di installazione di software da eseguire sulla certa famiglia, in cui l'unione produce una pluralità di fasi, ogni fase essendo associata ad un rispettivo componente da inserire sul sistema di computer obiettivo e ogni fase comprendendo un rispettivo numero di sequenza e numero di fase;

reperire la pluralità di fasi; e

sequenziare la pluralità di fasi in un predeterminato ordine secondo i rispettivi numeri di sequenza e numeri di fase per fornire una sequenza di fasi, la sequenza di fasi comprendendo comandi per installare software sul sistema di computer obiettivo durante le fasi di fabbricazione, le fasi di fabbricazione corrispondendo a rispettivi numeri di fase.

18. - Procedimento secondo la rivendicazione 17, comprendente inoltre la preparazione di variabile di ambiente corrispondenti a una certa pluralità di componenti.

19. - Procedimento secondo la rivendicazione 17, in cui la sequenza di fasi è atta a fornire comandi ripetibili per un periodo di tempo definito.

20. - Procedimento secondo la rivendicazione 17, in cui la sequenza di fasi è atta a fornire comandi ripetibili per un numero di iterazioni definito.

21. - Procedimento secondo la rivendicazione 17, comprendente inoltre scrittura della sequenza di fasi in un file di testo leggibile da computer.

22. - Procedimento per testare un sistema di computer, il procedimento comprendendo le fasi di:

leggere una pluralità di descrittori di componenti da un file leggibile da computer, almeno un descrittore di componenti descrivendo un rispettivo componente del sistema di computer;

leggere una pluralità di fasi da una banca dati, una fase essendo associata ad un rispettivo descrittore di componenti e comprendendo un rispettivo numero di sequenza; e

sequenziare la pluralità di fasi in un predeterminato ordine secondo i numeri di sequenza per fornire una sequenza di fasi, la sequenza di fasi comprendendo almeno un comando per testare il sistema di computer.

23. - Procedimento secondo la rivendicazione 22, in cui almeno un rispettivo componente è un componente hardware.

24. - Procedimento secondo la rivendicazione 22, in cui almeno un rispettivo componente è un componente software.

25. - Procedimento secondo la rivendicazione 22, in cui almeno una delle pluralità di fasi è una fase di test.

26. - Procedimento secondo la rivendicazione 22, comprendente inoltre la creazione di una pluralità di oggetti derivati corrispondenti alla

pluralità di descrittori di componenti.

27. - Procedimento secondo la rivendicazione 22, in cui il predeterminato ordine è secondo un ordinamento sequenziale dei numeri di sequenza.

28. - Procedimento secondo la rivendicazione 22, in cui la pluralità di fasi comprendono inoltre rispettivi numeri di fase e in cui il sequenziamento della pluralità di fasi in un predeterminato ordine è inoltre secondo i numeri di fase.

29. - Procedimento secondo la rivendicazione 22, comprendente inoltre la preparazione di variabili di ambiente corrispondenti alla pluralità di componenti.

30. - Procedimento secondo la rivendicazione 22, comprendente inoltre la scrittura della sequenza di fasi in un supporto di memorizzazione non volatile configurato in modo da accompagnare il sistema di computer durante la fabbricazione.

31. - Procedimento secondo la rivendicazione 22, comprendente inoltre il test del sistema di computer avente la pluralità di componenti.

32. - Sistema di computer comprendente:

un processore;

un componente accoppiato al processore; e

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

una memoria accoppiata al processore, la memoria comprendendo software installato su di essa, il software essendo installato tramite le fasi di:

leggere una pluralità di descrittori di componenti da un file leggibile da computer, almeno un descrittore di componenti descrivendo un rispettivo componente del sistema di computer;

leggere una pluralità di fasi da una banca dati, una fase essendo associata ad un rispettivo descrittore di componenti e comprendente un rispettivo numero di sequenza; e

sequenziare la pluralità di fasi in un predeterminato ordine secondo i numeri di sequenza per fornire una sequenza di fasi, la sequenza di fasi comprendendo almeno un comando per installare software sul sistema di computer.

33. - Sistema di computer secondo la rivendicazione 32, in cui almeno un rispettivo componente è un componente hardware.

34. - Sistema di computer secondo la rivendicazione 32, in cui almeno un rispettivo componente è un componente software.

35. - Sistema di computer secondo la rivendicazione 32, in cui almeno una della

BUZZI, NOTARO &
ANTONELLI D'OUZ
S.r.l.

pluralità di fasi è una fase di installazione di software.

36. - Sistema di computer secondo la rivendicazione 32, comprendente inoltre la creazione di una pluralità di oggetti derivati corrispondenti alla pluralità di descrittori di componenti.

37. - Sistema di computer secondo la rivendicazione 32, in cui il predeterminato ordine è secondo un ordinamento sequenziale dei numeri di sequenza.

38.- Sistema di computer secondo la rivendicazione 32, in cui la pluralità di fasi comprendono inoltre rispettivi numeri di fase e in cui il sequenziamento della pluralità di fasi in un predeterminato ordine è inoltre secondo i numeri di fase.

39. - Sistema di computer secondo la rivendicazione 32, comprendente inoltre la preparazione di variabili di ambiente corrispondenti alla pluralità di componenti.

40. - Sistema di computer secondo la rivendicazione 32, comprendente inoltre la scrittura della sequenza di fasi in un supporto di memorizzazione non volatile configurato in modo da

accompagnare il sistema di computer durante la fabbricazione.

41. - Sistema di computer secondo la rivendicazione 32, comprendente inoltre una pluralità di componenti.

42. - Procedimento per testare un sistema di computer obiettivo, il procedimento comprendendo le fasi di:

ricevere un ordine per il sistema di computer obiettivo, il sistema di computer obiettivo che deve comprendere una pluralità di componenti;

convertire l'ordine in un file leggibile da computer, il file essendo descrittivo della pluralità di componenti;

leggere il file;

reperire una pluralità di fasi da una banca dati secondo il file, una fase essendo associata ad un rispettivo componente e comprendendo un rispettivo numero di sequenza e numero di fase;

sequenziare la pluralità di fasi in un predeterminato ordine secondo i rispettivi numeri di sequenza e numeri di fasi per fornire una sequenza di fasi, la sequenza di fasi comprendendo comandi per testare software sul sistema di computer obiettivo durante le fasi di

fabbricazione, le fasi di fabbricazione corrispondendo a rispettivi numeri di fase.

43. - Procedimento secondo la rivendicazione 42, in cui si crea una condizione di errore se l'insieme intermedio è vuoto.

44. - Procedimento secondo la rivendicazione 42, comprendente inoltre la preparazione di variabili di ambiente corrispondenti alla pluralità di componenti.

45. - Procedimento secondo la rivendicazione 42, in cui la sequenza di fasi è atta a fornire comandi ripetibili per un periodo di tempo definito.

46. - Procedimento secondo la rivendicazione 42, in cui la sequenza di fasi è atta a fornire comandi ripetibili per un numero di iterazioni definito.

47. - Procedimento secondo la rivendicazione 42, comprendente inoltre la scrittura della sequenza di fasi in un dispositivo di memorizzazione e non volatile configurato in modo da accompagnare il sistema di computer durante la fabbricazione.

48. - Procedimento per produrre un sistema di computer pronto all'impiego secondo una specifica

combinazione su misura di componenti hardware e/o software, detto procedimento comprendendo

(a) controllare automaticamente la compatibilità dei componenti selezionati

(b) reperire automaticamente fasi di installazione e fasi di prova per almeno uno dei componenti

(c) determinare automaticamente una sequenza di dette fasi di installazione e fasi di prova,

(d) assemblare in modo automatizzato gli specifici componenti selezionati su misura del sistema di computer inclusa l'effettuazione delle fasi di installazione e di prova nella sequenza determinata.

49. - Procedimento secondo la rivendicazione 48, comprendente una o più delle fasi di una delle rivendicazioni precedenti.

50. - Procedimento e sistema sostanzialmente come descritto ed illustrato e per gli scopi specificati.

Ing. Franco BUZZI
N. iscriz. ALBO 289
In proprio e per gli altri



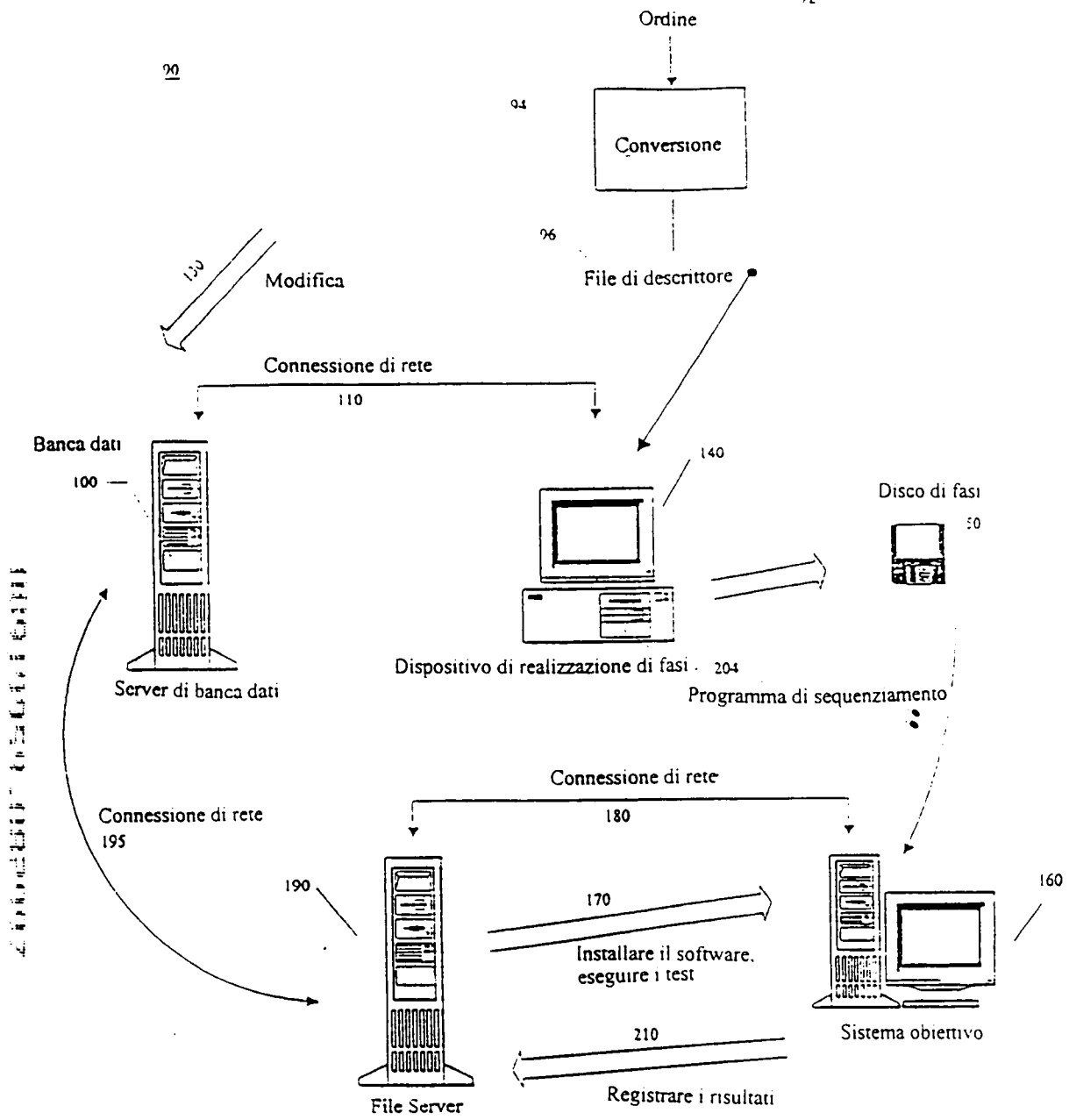


Figura 1



Ing. Franco BUZZI
N. Iscriz. ALBO 259
(a proprio e per tutti gli usi)

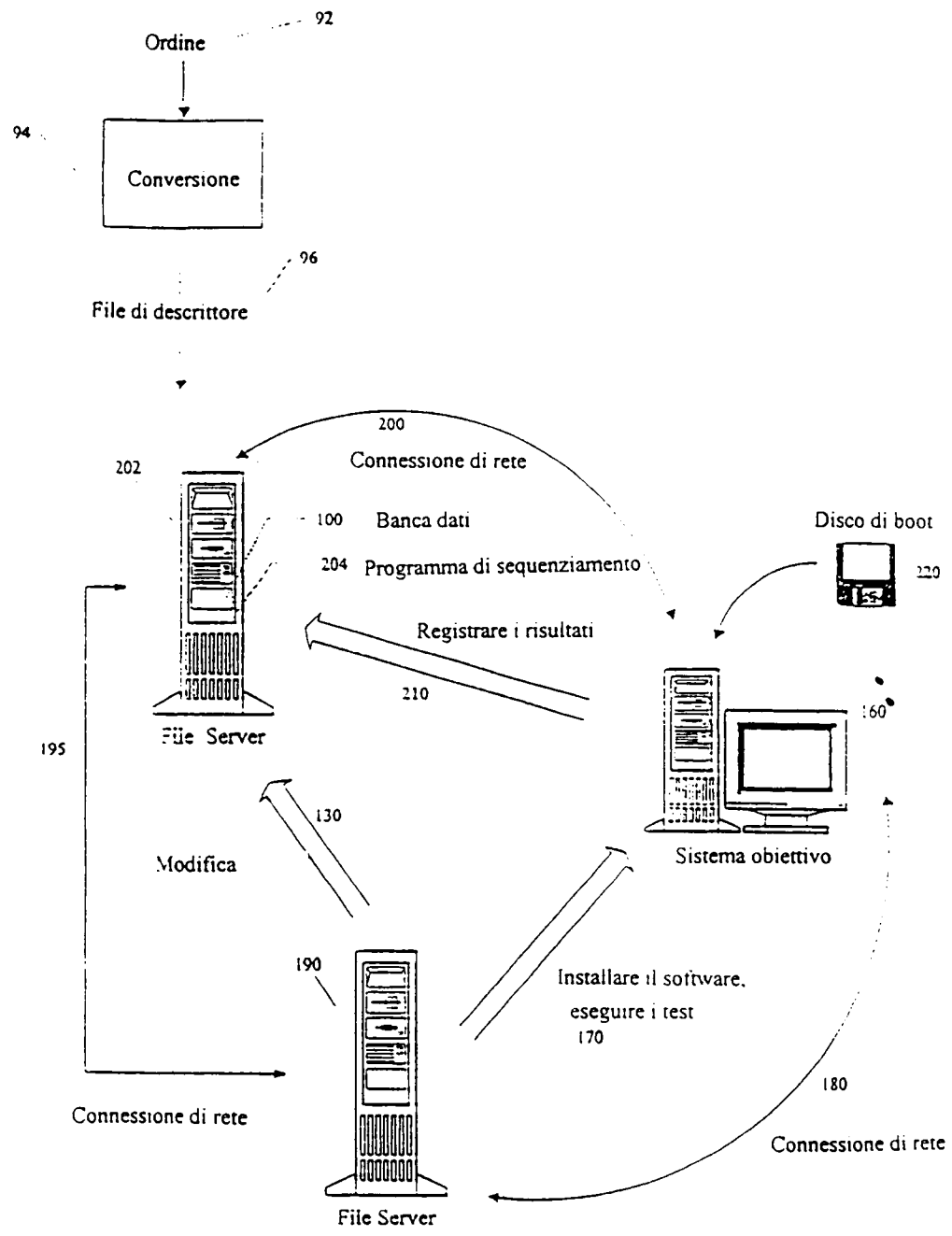



Figura 2


 Ing. Franco BUZZI
 N. Iscriz. ALBO 259
 Fa proprio e per gli altri

Progetto di banca dati

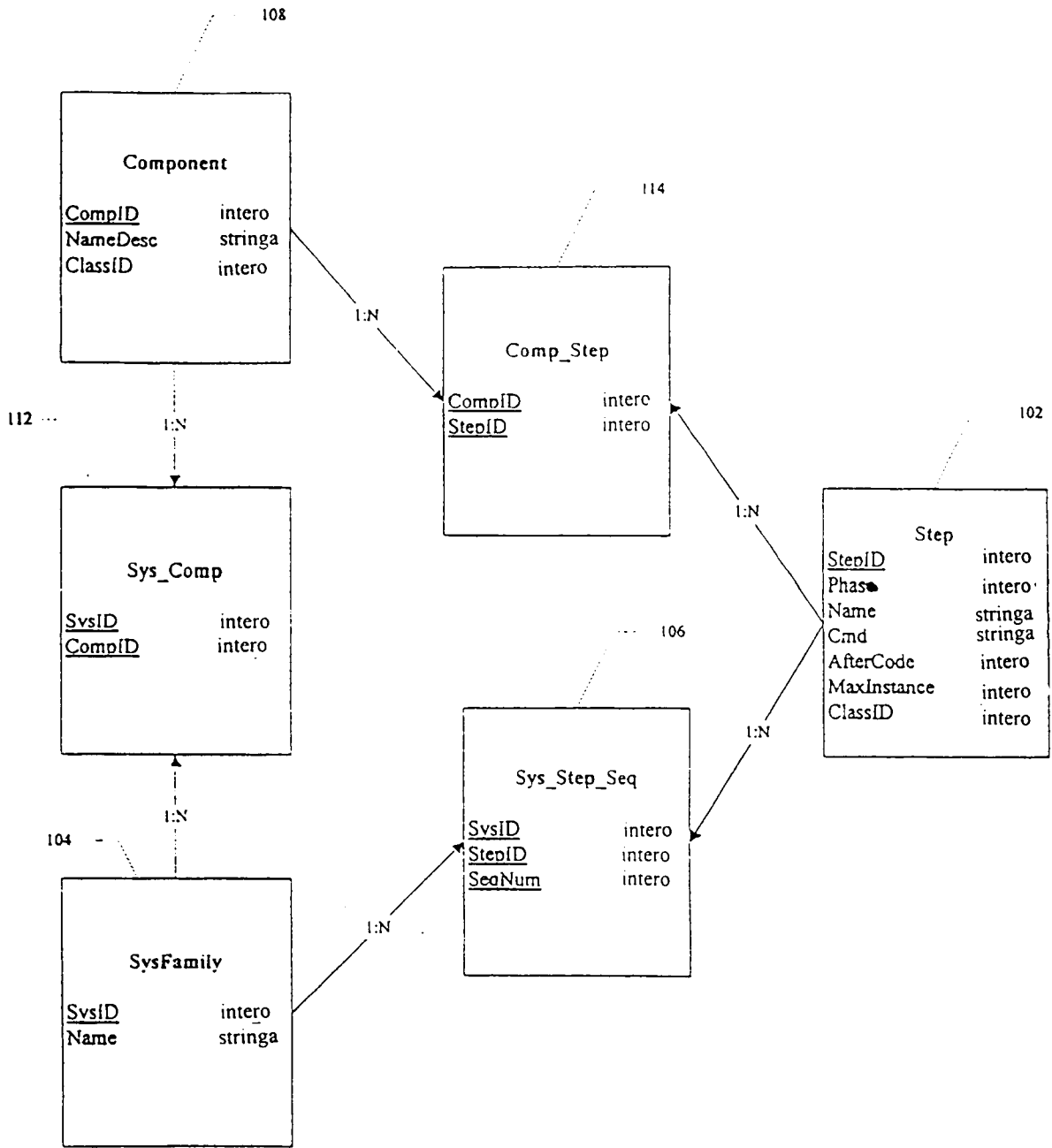


Figura 6



Ing. Franco BUZZI
P. Iscriz. Albo 259
Ma proprio e per gli altri

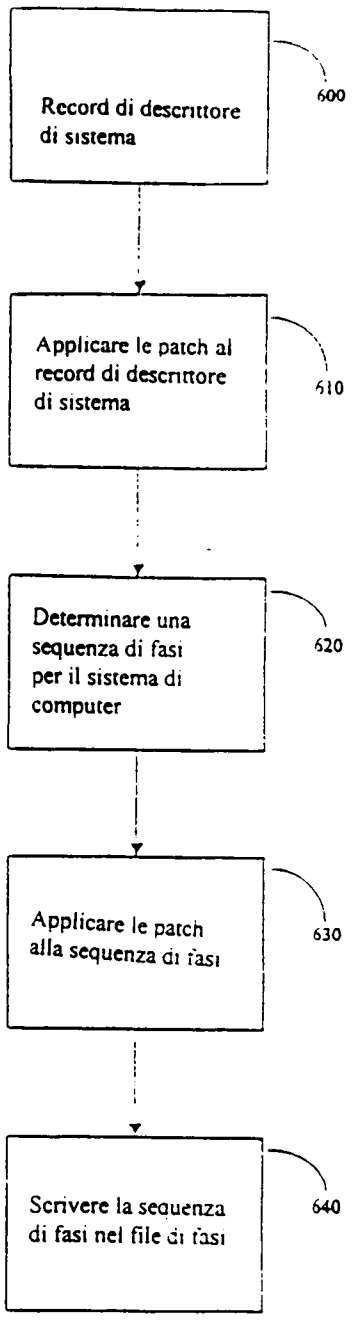


Figura 7



Ing. Franco BOZZI
N. iscriz. ALBO 259
Ho proprio e per gli altri

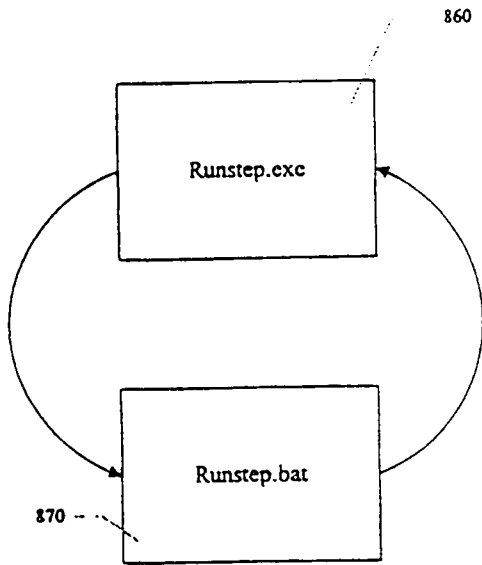


Figura 10

46628U-05661680



Ing. Franco BUZZI
N. Iscriz. ALBO 269
In proprio e per gli altri

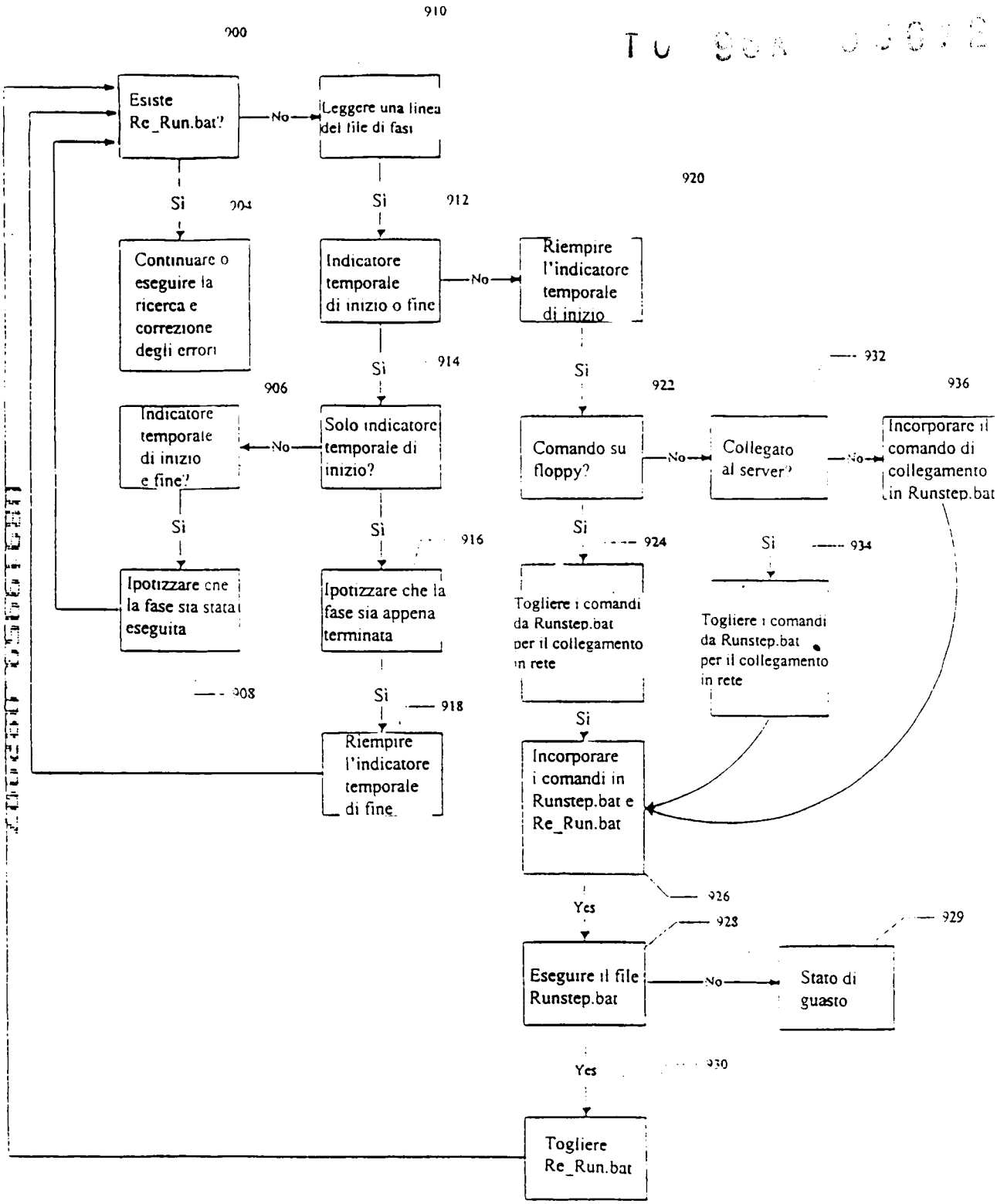


Figura 11



Ing. Franco BUZZI
 N. Iscriz. ATBO 259
la proprio e per gli altri