



(19)  
Bundesrepublik Deutschland  
Deutsches Patent- und Markenamt

(10) **DE 697 37 170 T2** 2007.11.08

(12) **Übersetzung der europäischen Patentschrift**

(97) **EP 0 817 012 B1**

(51) Int Cl.<sup>8</sup>: **G06F 9/44** (2006.01)

(21) Deutsches Aktenzeichen: **697 37 170.0**

(96) Europäisches Aktenzeichen: **97 304 365.6**

(96) Europäischer Anmeldetag: **20.06.1997**

(97) Erstveröffentlichung durch das EPA: **07.01.1998**

(97) Veröffentlichungstag

der Patenterteilung beim EPA: **03.01.2007**

(47) Veröffentlichungstag im Patentblatt: **08.11.2007**

(30) Unionspriorität:

**9613767            06.07.1996    GB**

(84) Benannte Vertragsstaaten:

**DE, GB**

(73) Patentinhaber:

**International Business Machines Corp., Armonk,  
N.Y., US**

(72) Erfinder:

**Trainer, Ray, Warwick, Warwickshire, GB;  
Donnelly, Vanessa, Styvechale, Coventry, CV3  
5RN, GB**

(74) Vertreter:

**Duscher, R., Dipl.-Phys. Dr.rer.nat., Pat.-Ass.,  
70176 Stuttgart**

(54) Bezeichnung: **Auswahl von Befehlen in einem Rechnersystem**

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

## Beschreibung

### Gebiet der Erfindung

**[0001]** Die vorliegende Erfindung bezieht sich im Allgemeinen auf Computersysteme mit Benutzerschnittstellen, über die ein Benutzer Operationen auswählen kann, die von dem Computersystem ausgeführt werden sollen, und im Besonderen auf die Mechanismen, mit denen der Benutzer derartige Operationen aufrufen kann.

### Hintergrundinformationen

**[0002]** Benutzerschnittstellen gestatten dem Computerbenutzer die Interaktion oder Kommunikation mit dem Computersystem. Benutzerschnittstellen werden üblicherweise mit einem Anzeigebildschirm und einer vom Benutzer gesteuerten Eingabeeinheit wie beispielsweise einer Tastatur, einer Maus, einem Mikrofon, einem Lichtstift oder Ähnlichem realisiert. Der Anzeigebildschirm zeigt dem Benutzer Informationen und Daten an, und der Benutzer verwendet die Eingabeeinheit, um Befehle einzugeben und dem Computersystem Daten bereitzustellen.

**[0003]** In den letzten Jahren wünschen und benötigen immer mehr Menschen die Leistungsfähigkeit des Computers für die Erledigung ihrer täglichen Arbeit. Im Allgemeinen wollen diese Benutzer jedoch nicht, dass ihnen die Kenntnis bestimmter Befehle, Operatoren, Syntaxregeln usw. abverlangt wird, und es wurden daher mit großem Aufwand Benutzerschnittstellen entwickelt, welche die Notwendigkeit von derart weit reichenden Kenntnissen überflüssig machen. Die heutzutage üblichste Form der Benutzerschnittstelle ist die so genannte grafische Benutzeroberfläche (Graphical User Interface, GUI). Eine GUI stellt dem Benutzer üblicherweise Fenster und Symbole bereit. Fenster enthalten in der Regel eine Menüleiste, eine Symbolleiste und einen Client-Bereich. Der Client-Bereich kann üblicherweise eine Anzahl von Symbolen beinhalten, bei denen es sich um kleine, symbolhafte Darstellungen von Einheiten (Anwendungen, Ordner usw.) handelt, mit denen der Benutzer arbeitet.

**[0004]** Damit der Benutzer bestimmte Operationen auf dem Computersystem aufrufen kann, muss ihm die Benutzeroberfläche Auswahlmechanismen bereitstellen. In einer typischen Benutzeroberflächenumgebung kann dies durch die Definition von „Aktionen“ erreicht werden, die der Benutzer über „Ansichten“ dieser Aktionen auswählen kann, die ihm über die Benutzeroberfläche wie beispielsweise die in WO-A-94/15281 beschriebene Oberfläche bereitgestellt werden. Im Rahmen dieser Patentanmeldung kann eine Aktion als ein vom Benutzer veranlassenes Ereignis definiert werden, das eine Operation aufruft. Die „Ansichten“, mit denen eine Aktion auf der Benutzeroberfläche dargestellt wird, kann verschiedene visuelle und auch nichtvisuelle Formen annehmen.

zeroberfläche dargestellt wird, kann verschiedene visuelle und auch nichtvisuelle Formen annehmen.

**[0005]** Typische Beispiele visueller Ansichten, die in GUIs verwendet werden, könnten ein Wort oder eine Wortverbindung in einer Menüleiste (wie in [Fig. 5A](#) gezeigt, in der die Aktion „Print“ („Drucken“) aus einer Wort-Ansicht ausgewählt wird), eine grafische Darstellung wie z.B. ein Symbol oder ein Pixelmuster in einer Symbolleiste (wie in [Fig. 5B](#) gezeigt, in der die Aktion „Print“ („Drucken“) aus einer Ansicht eines Drucken-Pixelmusters ausgewählt wird) oder auch anderweitige visuelle Bedienelemente sein, welche die Benutzerauswahl oder -eingabe ermöglichen. Beispiele für derartige anderweitige Bedienelemente sind ein Kombinationslistenfeld, das die Auswahl einer Schriftart ermöglicht (wie in [Fig. 5C](#) dargestellt), oder ein Eingabefeld, das die Festlegung der Eigenschaft einer Zeichenfolge wie beispielsweise die Festlegung eines Verzeichnisnamens oder die Festlegung eines Symbolleistennamens (wie in [Fig. 5D](#) dargestellt) ermöglicht. Diese Bedienelemente können an einer beliebigen Stelle der Benutzeroberfläche einschließlich einer Symbolleiste oder einer Fensteranordnung angezeigt werden. Weitere Beispiele für visuelle Ansichten sind in [Fig. 5E](#) dargestellt, in der die Drucken-Aktion aus einem Kontextmenü ausgewählt werden kann, das angezeigt wird, indem die rechte Maustaste gedrückt wird, während sich der Mauszeiger über dem Fensterbereich befindet, sowie in [Fig. 5F](#), in der die Drucken-Aktion aus einer Schaltflächenansicht in einer Fensteranordnung ausgewählt werden kann.

**[0006]** Beispiele für nichtvisuelle Ansichten wären Schnellzugriffsdaten wie beispielsweise „Strg+D“ zur Auswahl einer Drucken-Aktion, Sprachmusterdaten um zu ermitteln, ob die Aktion anhand einer Spracheingabe ausgewählt wurde, oder Berührungsdaten um zu ermitteln, ob eine Eingabeberührung (wie z.B. die Berührung mit einem Stift auf einem Tablett) eine Auswahl einer bestimmten Aktion darstellt. Im Rahmen der vorliegenden Patentanmeldung werden alle derartigen Mechanismen für die Auswahl von Aktionen (ob visuell oder nichtvisuell) als „Ansichten“ bezeichnet.

**[0007]** Bei der Entwicklung von Anwendungen ist zurzeit ein beträchtlicher Aufwand seitens des Entwicklers erforderlich, um die notwendige Funktionalität bereitzustellen, die dem Benutzer die Auswahl von Aktionen ermöglicht. Außerdem gestaltet es sich als problematisch, eine anwendungsübergreifende Einheitlichkeit von Benutzeroberfläche und Funktionen für mehrere Anwendungen zu erzielen, die Ansichten von Aktionen anzeigen, welche vom Benutzer hinsichtlich ihrer Funktion als identisch empfunden werden.

**[0008]** Ein Grund für den mit dem Ermöglichen die-

ser Aktionen verbundenen hohen Zeitaufwand für den Entwickler besteht darin, dass es innerhalb einer Anwendung eine ganze Reihe von Stellen gibt, an denen der Benutzer ein und dieselbe Aktion auswählen kann, da es an verschiedenen Stellen der Benutzeroberfläche unter Umständen viele Ansichten einer bestimmten Aktion gibt, die dem Benutzer zur Auswahl zur Verfügung steht. Offensichtliche Ansichten von Aktionen sind diejenigen in Menüleisten, Symbolleisten und Kontextmenüs. Allerdings gibt es jedoch auch viele Stellen innerhalb einer Anwendung, wo dem Benutzer eine Schaltfläche angezeigt wird, mit der sich ein Dialogfeld oder Fenster öffnen lässt. Diese „Aktion“ könnte in einem anderen Zusammenhang auch über eine Menüleiste oder ein Kontextmenü verfügbar sein.

**[0009]** Einige Aktionen öffnen Dialogfelder, z.B. das Dialogfeld „Datei öffnen“, während andere dies zwar nicht tun, jedoch trotzdem eine Operation durchführen, wie dies beispielsweise bei den Aktionen für das Ausschneiden, Kopieren und Einfügen der Fall ist. Das gemeinsame Merkmal dieser Operationen besteht darin, dass sie dem Benutzer in Aktionsansichten zur Verfügung gestellt werden. So kann ein Benutzer beispielsweise eine Drucken-Option in einem Auswahlménü der Menüleiste, eine Symbolleisten-Schaltfläche in einer Symbolleiste, einen Eintrag in einem Kontextmenü oder auch eine Schaltfläche in einem Dialogfeld auswählen. Dabei findet im Wesentlichen ein und dieselbe Operation des Öffnens eines Drucken-Dialogfelds statt, wobei jedoch zur Erzielung desselben Ergebnisses unterschiedliche visuelle Auswahlmechanismen eingesetzt werden.

**[0010]** Ein zweiter Aspekt, der den Aufwand für den Entwickler steigert, besteht darin, dass es eine Reihe unterschiedlicher Eingabemechanismen gibt, die der Benutzer zur Auswahl derselben Aktion verwenden kann. Dabei ist der Standardmechanismus des Zeigens und Klickens mit der Maus hinreichend bekannt und der übliche Mechanismus zur Auswahl von visuellen Ansichten, die über die GUI angezeigt werden. Allerdings verfügen eine Reihe von Aktionen über Schnellzugriffsoptionen, welche die Auswahl über ein bestimmtes Zeichen auf der Tastatur (z.B. „D“ für die Aktion „Drucken“) ermöglichen. Darüber hinaus ermöglichen Anwendungen mit Spracherkennung dem Benutzer das Sprechen von Befehlen, so dass der Benutzer beispielsweise „Drucken“ sagt, um das Drucken-Dialogfeld zu öffnen, und wenn eine Stiftberührung zum Öffnen eines Drucken-Dialogfelds definiert wurde, fügt die Stift-Aktivierung eine weitere Ansicht hinzu, die dem Benutzer zur Auswahl der Aktion zur Verfügung stehen kann.

**[0011]** Daneben gibt es Situationen, in denen geringfügige Aktionen auf der Benutzeroberfläche so häufig vorkommen, dass ein und derselbe Code in der gesamten Anwendung immer wieder geschrie-

ben werden muss. Ein Beispiel hierfür ist eine Anwendung mit einer Anzahl von dialogfeldartigen Fenstern, bei denen der Benutzer das Fenster schließen kann, ohne Änderungen zu übernehmen. Die visuelle Darstellung dieser Funktion kann beispielsweise eine Schaltfläche „Abbrechen“ sein, wobei üblicherweise jedoch auch ein Tastaturmechanismus vorhanden ist, um dasselbe Ergebnis zu erzielen, indem z.B. die „Entf“-Taste gedrückt wird. Herkömmlicherweise muss der Entwickler zur Realisierung dieser Funktion drei Dinge codieren: a) eine Schaltfläche in einem Dialogfeld bereitstellen; b) die Entfernen-Schnellzugriffstaste in die Tabelle der Schnellzugriffstasten aufnehmen; und c) die Abbrechen-Prozedur in den Anwendungscode aufnehmen. Diese drei Schritte werden im Normalfall jedes Mal wiederholt, wenn die Abbrechen-Funktion für ein einzelnes Fenster benötigt wird.

**[0012]** Somit muss der Anwendungsentwickler heutzutage eine erhebliche Menge an Code schreiben, um diese verschiedenen Szenarien zu berücksichtigen, und wenn eine neue Technologie wie z.B. Sprache eingeführt wird, ist zusätzliche Arbeit seitens des Entwicklers notwendig, um dem Benutzer den neuen Auswahlmechanismus zur Verfügung zu stellen. Da der Entwickler dabei eine Reihe von Permutationen berücksichtigen muss, kommt es vor, dass bestimmte Auswahlmechanismen in Produkten nicht aktiviert sind. Dies führt zu Problemen mit der produktübergreifenden Nutzung, da eine Anwendung auf eine bestimmte Art und Weise funktioniert, während eine andere dies nicht tut. Dazu kommt, dass die verschiedenen Ansichten derselben Aktionen in der Regel auf verschiedene Arten aufgebaut sind und im Nachhinein nur eine geringe oder überhaupt keine Anpassung durch den Benutzer ermöglichen.

**[0013]** Aus diesen Erörterungen wird deutlich, dass es zurzeit für den Entwickler mit sehr viel Arbeit verbunden ist, eine vollständige Unterstützung der verschiedenen benötigten Ansichten von Aktionen, seien sie visuell oder nichtvisuell, zu erzielen. Es ist daher eine Aufgabe der vorliegenden Erfindung, ein System und Verfahren bereitzustellen, das die oben erwähnten Probleme behebt.

#### Zusammenfassung der Erfindung

**[0014]** Gemäß einem ersten Aspekt stellt die vorliegende Erfindung ein Computersystem mit einer Benutzeroberfläche bereit, über die ein Benutzer eine Aktion auswählen kann, um eine gewünschte Operation aufzurufen, wobei die Aktion mittels einer Vielzahl von Ansichten ausgewählt werden kann, die über die Benutzeroberfläche bereitgestellt werden, wobei das System durch Folgendes gekennzeichnet ist: ein Mittel für das Bereitstellen eines einzelnen Aktionsobjekts, das für eine Aktion steht, wobei das Aktionsobjekt für jede der Ansichten, die zur Darstellung

der Aktion verwendet werden können, die Attribute, welche zur Bereitstellung der Ansicht erforderlich sind, sowie einen Bezeichner definiert, um die Operation, die bei Auswahl der Aktion aufgerufen werden soll, kenntlich zu machen; ein Erkennungsmittel, mit dem erkannt wird, wann eine Benutzerauswahl einer Ansicht erfolgt ist, und mit dem das zugehörige Aktionsobjekt von der Auswahl unterrichtet wird; wobei das Aktionsobjekt auf das Erkennungsmittel anspricht, um den Bezeichner zum Erzeugen eines Befehls zu verwenden, um die gewünschte Operation aufzurufen.

**[0015]** Bei bevorzugten Ausführungsformen kann das Computersystem mit einer Anzeigeeinheit verbunden werden, um eine grafische Benutzeroberfläche anzuzeigen, und wobei mindestens eine der Ansichten, die zur Darstellung der Aktion verwendet werden können, eine visuelle Ansicht ist, die über die grafische Benutzeroberfläche angezeigt werden kann, wobei die in dem Aktionsobjekt für die mindestens eine der Ansichten definierten Attribute die Daten beinhalten, die für das Erzeugen der visuellen Ansicht erforderlich sind.

**[0016]** Gemäß einem zweiten Aspekt stellt die vorliegende Erfindung ein Verfahren für den Betrieb eines Computersystems mit einer Benutzeroberfläche bereit, über die ein Benutzer eine Aktion auswählen kann, um eine gewünschte Operation aufzurufen, wobei die Aktion mittels einer Vielzahl von Ansichten ausgewählt werden kann, die über die Benutzeroberfläche bereitgestellt werden, wobei das Verfahren durch die folgenden Schritte gekennzeichnet ist: (a) Bereitstellen eines einzelnen Aktionsobjekts zur Darstellung einer Aktion, wobei das Aktionsobjekt für jede der Ansichten, die zur Darstellung der Aktion verwendet werden können, die Attribute, welche zur Bereitstellung der Ansicht erforderlich sind, sowie einen Bezeichner definiert, um die Operation, die bei Auswahl der Aktion aufgerufen werden soll, kenntlich zu machen, wobei das Aktionsobjekt auf eine derartige Auswahl anspricht, um den Bezeichner zum Aufrufen der gewünschten Operation zu verwenden; (b) Erkennen des Auftretens einer Benutzerauswahl einer Ansicht; und (c) als Reaktion auf eine Erkennung einer Benutzerauswahl einer Ansicht, Unterrichten des zugehörigen Aktionsobjekts von der Auswahl und somit Aufrufen der gewünschten Operation.

**[0017]** Gemäß einem dritten Aspekt der vorliegenden Erfindung wird ein Programmprodukt zur Verwendung in einem Computersystem mit einer Benutzeroberfläche bereitgestellt, über die ein Benutzer eine Aktion auswählen kann, um eine gewünschte Operation aufzurufen, wobei die Aktion mittels einer Vielzahl von Ansichten ausgewählt werden kann, die über die Benutzeroberfläche bereitgestellt werden, wobei das Computersystem weiter über einen Prozessor zur Ausführung einer Anwendung sowie über

einen Speicher zur Speicherung von Daten, die von der Anwendung verwendet werden, verfügt, wobei das Programmprodukt so aufgebaut ist, dass es die folgenden Funktionen bereitstellt, wenn es in den Speicher des Computersystems geladen ist: ein Mittel für die Bereitstellung eines einzelnen Aktionsobjekts, das für eine Aktion steht, wobei das Aktionsobjekt für jede der Ansichten, die zur Darstellung der Aktion verwendet werden können, die Attribute, welche zur Bereitstellung der Ansicht erforderlich sind, sowie einen Bezeichner definiert, um die Operation, die bei Auswahl der Aktion aufgerufen werden soll, kenntlich zu machen; ein Erkennungsmittel, mit dem erkannt wird, wann eine Benutzerauswahl einer Ansicht erfolgt ist, und mit dem das zugehörige Aktionsobjekt von der Auswahl unterrichtet wird; wobei das Aktionsobjekt auf das Erkennungsmittel anspricht, um den Bezeichner zum Erzeugen eines Befehls zu verwenden, um die gewünschte Operation aufzurufen.

**[0018]** Gemäß einem vierten Aspekt stellt die vorliegende Erfindung ein Programmprodukt zur Verwendung in einem Computersystem bereit, um eine Anwendung zu entwickeln, wobei die Anwendung zur Verwendung auf einem System mit einer Benutzeroberfläche gedacht ist, über die ein Benutzer eine Aktion auswählen kann, um eine gewünschte Operation aufzurufen, wobei die Aktion mittels einer Vielzahl von Ansichten ausgewählt werden kann, die über die Benutzeroberfläche bereitgestellt werden, wobei das Programmprodukt so aufgebaut ist, dass es die folgenden Funktionen bereitstellt, wenn es in das zur Entwicklung der Anwendung verwendete Computersystem geladen ist: ein Erzeugungsmittel zum Erzeugen eines einzelnen Aktionsobjekts zur Darstellung einer Aktion, wobei das Aktionsobjekt für jede der Ansichten, die zur Darstellung der Aktion verwendet werden können, die Attribute, welche zur Bereitstellung der Ansicht erforderlich sind, sowie einen Bezeichner definiert, um die Operation, die bei Auswahl der Aktion aufgerufen werden soll, kenntlich zu machen, wobei das Aktionsobjekt auf eine derartige Auswahl anspricht, um den Bezeichner zum Aufrufen der gewünschten Operation zu verwenden.

**[0019]** Gemäß der Erfindung wird ein einziger Mechanismus für die Verarbeitung von Aktionen bereitgestellt. Dieser Mechanismus hat die Form eines OOP-Objekts, das für jede verfügbare Ansicht, die zur Darstellung der Aktion verwendet werden kann, die Attribute, welche zur Bereitstellung der Ansicht notwendig sind, sowie einen Bezeichner definiert, um die Operation, die bei Auswahl der Aktion aufgerufen werden soll, kenntlich zu machen.

**[0020]** Die objektorientierte Programmierung (OOP) ist eine bestimmte Art der Softwareentwicklung, bei der erforderliche Funktionen mit Hilfe von „Nachrichten“ realisiert werden, die an „Objekte“ gesendet wer-

den. Ein „Objekt“ ist ein Softwarepaket, das eine Zusammenstellung von Daten und zugehörigen Softwareprozeduren (häufig auch als „Methoden“ bezeichnet) enthält. Üblicherweise werden Objekte als Instanzen einer bestimmten „Objektklasse“ erzeugt, wobei die Klasse eine Vorlage für die Definition der Prozeduren und Daten für einen bestimmten Objekttyp ist. Eine „Nachricht“ ist ein an ein Objekt gesendetes Signal, mit dem das Objekt aufgefordert wird, eine seiner Prozeduren auszuführen. Somit veranlasst eine an ein Objekt gesendete Nachricht, dass eine Prozedur aufgerufen wird, um die benötigte Funktion zu realisieren.

**[0021]** Im Rahmen der vorliegenden Patentanmeldung wird das für die Verarbeitung von Aktionen bereitgestellte OOP-Objekt als ein „Aktionsobjekt“ bezeichnet. Bei bevorzugten Ausführungsformen definiert dieses Aktionsobjekt für jeden verfügbaren Benutzerauswahlmechanismus (bzw. „Ansicht“), die zur Darstellung der Aktion verwendet werden kann, (a) die Attribute, die entweder zum Erzeugen einer visuellen Darstellung der Aktion bei der visuellen Benutzerauswahl oder aber zum Erzeugen anderweitiger Darstellungen, welche eine Auswahl über nichtvisuelle Mechanismen ermöglichen, notwendig sind, sowie (b) einen Bezeichner, der die Operation, die bei Auswahl der Aktion aufgerufen werden soll, kenntlich macht. Dieser Bezeichner besteht vorzugsweise aus einem eindeutigen Bezeichner für das ausgewählte Aktionsobjekt und einem Mechanismus für die Anwendungsrückfrage, der automatisch aufgerufen wird, wenn die Aktion ausgewählt wird, um die Ausführung der gewünschten Operation zu veranlassen. Durch eine derartige Vorgehensweise kann eine Anwendung einen einzelnen Einstiegspunkt definieren, der bei Auswahl einer Aktion definiert werden soll, ohne dass bekannt sein muss, welcher Eingabemechanismus den Aufruf erzeugt hat (z.B. Maus, Tastatur, Spracheingabe usw.) oder in welcher Ansicht die Aktion ausgewählt wurde.

**[0022]** Zusätzlich definiert das Aktionsobjekt bei bevorzugten Ausführungsformen den momentanen Status der Aktion, z.B. „Aktiviert“, „Deaktiviert“, „Dauerbetrieb“ oder „Nicht definiert“. Der Status einer Aktion beschreibt ihren sichtbaren Zustand in zugehörigen visuellen Aktionsansichten sowie, ob die Aktion in nichtvisuellen Ansichten auswählbar ist. Der Status „Aktiviert“ gibt an, dass die Aktion zurzeit verfügbar, jedoch nicht ausgewählt ist. Der Status „Dauerbetrieb“ gibt an, dass für die Aktion ein Dauerbetrieb möglich ist und sie somit auf weitere Interaktionen innerhalb einer Anwendung angewendet werden kann, ohne dass eine ausdrückliche Neuauswahl notwendig ist. Ein Beispiel hierfür ist die Aktion „Fett“ in einem Textverarbeitungsprogramm. Wenn für diese Aktion der Dauerbetrieb ausgewählt wurde, erhält der neue Text bei einer weiteren Dateneingabe durch den Benutzer das Attribut „Fett“. Ein Status „Deakti-

viert“ gibt an, dass die Aktion zurzeit nicht verfügbar ist. Ein Status „Nicht definiert“ gibt an, dass die momentane Benutzerauswahl keine bestimmte Bedeutung für die Aktion hat. Ein Beispiel hierfür – wiederum mit Blick auf ein Textverarbeitungsprogramm – besteht darin, dass der Benutzer eine Anzahl von Sätzen auswählt, die durchweg verschiedene Schriftarten aufweisen. Die Aktion „Schrift auswählen“ würde normalerweise eine visuelle Rückmeldung zur momentan verwendeten Schriftart geben; in diesem Fall lautet der Status der Aktion jedoch „Nicht definiert“, da die Aktion nicht in der Lage ist, die korrekte Rückmeldung an den Benutzer zu ermitteln.

**[0023]** Durch das Entfernen dieser Funktionalität aus dem betreffenden Auswahl- und Eingabemechanismus, in dem sie bei Systemen nach dem Stand der Technik üblicherweise bereitgestellt wird, ist das einzelne Aktionsobjekt in der Lage, alle Ansichten, über die es momentan verfügbar ist, von einer etwaigen Statusänderung zu unterrichten. Auf diese Weise kommt der Anwendungscode mit weniger Logik für die Sicherstellung konsistenter Ansichten aus. Die Funktionsweise von Standardmethoden für eine Rahmenstruktur für die Benachrichtigung zwischen Objekten ist in der Technik hinlänglich bekannt. So werden beim IBM Produkt Visual Age C++ beispielsweise Klassen bereitgestellt, die diese Funktion ausführen. Andere Klassen können dieses Verhalten übernehmen, indem sie die standardmäßige OO-Methode der Vererbung von Beobachter- und Benachrichtiger-Klassen verwenden.

**[0024]** In GUI-Umgebungen werden die Auswahlmechanismen für einen Satz von Aktionen häufig auf eine bestimmte Art und Weise zusammengefasst. Meist werden Aktionssätze in visueller Form als Menüleisten, Symbolleisten und Aufklappmenüs dargestellt. Es können jedoch auch nichtvisuelle Darstellungen bereitgestellt werden, wie beispielsweise Schnellzugriffstabellen, die eine Tastaturlauswahl für einen Satz von Aktionen ermöglichen. So kann eine Symbolleiste z.B. eine Anzahl grafischer Schaltflächen enthalten, die jeweils für eine Ansicht einer bestimmten Aktion stehen, und die Auswahl einer dieser Schaltflächen führt dazu, dass die konkrete Aktion, für welche die Schaltfläche steht, die benötigte Operation aufruft. Im Rahmen der vorliegenden Patentanmeldung werden derartige Zusammenstellungen von Ansichten als „gemeinsame Ansichten“ bezeichnet, unabhängig davon, ob es sich um visuelle Zusammenstellungen (z.B. Menüleiste, Symbolleiste) oder nichtvisuelle Zusammenstellungen (z.B. Schnellzugriffstabelle) handelt.

**[0025]** Dabei ist zu beachten, dass gemeinsame Ansichten innerhalb von anderen gemeinsamen Ansichten verschachtelt sein können. So kann eine Menüleiste beispielsweise insofern als eine gemeinsame Ansicht betrachtet werden, als sie Ansichten für

eine Anzahl unterschiedlicher Aktionen enthält. Einige der Einträge in der Menüleiste (z.B. das Wort „Datei“) können wiederum gemeinsame Ansichten darstellen, da nach ihrer Auswahl eine andere gemeinsame Ansicht angezeigt werden kann. So kann bei Auswahl des Wortes „Datei“ eine gemeinsame Ansicht in Form einer Liste angezeigt werden. Die Wörter in dieser Liste sind Ansichten bestimmter Aktionen, wobei „Öffnen“ z.B. eine Ansicht der Aktion „Öffnen“ ist, mit der eine Datei geöffnet wird.

**[0026]** Bei bevorzugten Ausführungsformen der vorliegenden Erfindung wird ein einzelner Mechanismus für die Verarbeitung der Erstellung gemeinsamer Ansichten bereitgestellt. Dieser Mechanismus hat die Form eines OOP-Objekts und wird als eine Menüdefinition bezeichnet, die vorzugsweise Folgendes beinhaltet: (a) einen Satz von Einträgen, von denen jeder einzelne entweder für ein Aktionsobjekt, ein Trennzeichen oder einen Verweis auf eine andere Instanz einer Menüdefinition steht; und (b) die Attribute, die erforderlich sind, um eine Darstellung einer gemeinsamen Ansicht von Aktionen (visuell und nichtvisuell) bereitzustellen. Vorzugsweise legt die Menüdefinition auch den Kontext fest, in dem die gemeinsame Ansicht von Aktionen angezeigt oder für eine Auswahl aktiviert wird.

**[0027]** Bei bevorzugten Ausführungsformen der vorliegenden Erfindung umfasst das System ein Erstellungsmittel für das Erstellen einer gemeinsamen Ansicht von Aktionen aus einer Menüdefinition, welche die gemeinsame Ansicht definiert, wobei die Menüdefinition über Einträge verfügt, die Aktionsobjekte und/oder andere Menüdefinitionen angeben, welche durch die gemeinsame Ansicht dargestellt werden sollen, und wobei das Erstellungsmittel so aufgebaut ist, dass es auf diese Aktionsobjekte und/oder Menüdefinitionen verweist, um die gemeinsame Ansicht zu erstellen.

**[0028]** Da die Menüdefinitionen auf Aktionsobjekte verweisen und da alle Attribute, die für das Erzeugen einer Aktionsansicht notwendig sind, in diesem Aktionsobjekt enthalten sind, können Menüdefinitionen für mehrere spezifische gemeinsame Ansichten verwendet werden. So kann eine bestimmte Menüdefinition beispielsweise für das Erzeugen einer textgebundenen Menüleiste oder einer entsprechenden grafischen Symbolleiste verwendet werden, da in beiden Fällen auf das Aktionsobjekt für einen bestimmten Eintrag in der gemeinsamen Ansicht verwiesen wird, und das Aktionsobjekt kann dann – je nach Art der erstellten gemeinsamen Ansicht – entweder den betreffenden Text (z.B. „Drucken“) oder eine Symbolform (z.B. ein Druckersymbol) bereitstellen.

**[0029]** Wie bereits erwähnt, kann eine Menüdefinition bei bevorzugten Ausführungsformen Kontextda-

ten enthalten, die angeben, unter welchen Umständen gemeinsame Ansichten, für die die Menüdefinition steht, angezeigt oder aktiviert werden sollen. Eine Menüdefinition und somit auch die ihr zugehörige „gemeinsame Ansicht“ ist standardmäßig immer aktiviert bzw. wird immer angezeigt, allerdings kann ein Aufklapp- oder Kontextmenü über mindestens einen ihm zugehörigen Kontext verfügen, so dass es nur dann angezeigt wird, wenn sein eigener Kontextstatus „wahr“ ist.

**[0030]** Kontext kann auf eine beliebige Menüdefinition oder gemeinsame Ansicht anwendbar sein. So kann eine Menüleiste beispielsweise eingebettete Menüdefinitionen enthalten, die unabhängig vom Anwendungskontext immer angezeigt werden (z.B. das Auswahlmenü „Datei“, das Auswahlmenü „Bearbeiten“ usw.), sowie einige Einträge, die nur auf bestimmte Kontexte anwendbar sind. Eine Menüleiste kann beispielsweise eine eingebettete Menüdefinition beinhalten, die einen Satz von Aktionen für die Verarbeitung von Grafikbildern bereitstellen. Dieser Satz von Aktionen kann nur dann anwendbar sein, wenn ein Grafikobjekt vom Benutzer ausgewählt wird, so dass die Menüleiste – wenn kein Grafikbild ausgewählt wird – die eingebettete Menüdefinition nicht anzeigt. Die Menüleiste kann diese Entscheidung treffen, indem sie den Kontext der eingebetteten Menüdefinitionen mit dem momentanen Anwendungskontext vergleicht.

**[0031]** Bei bevorzugten Ausführungsformen wird eine Aktionspaletten-Ansicht (**280**) bereitgestellt, die auf der Anzeigeeinheit (**55**) angezeigt werden kann, um eine Liste der verfügbaren Aktionen bereitzustellen. Auf diese Weise kann der Benutzer die Attribute einer Aktion direkt bearbeiten. Außerdem verfügt die Palettenansicht vorzugsweise über eine Funktion für das Ziehen und Ablegen, mit welcher der Benutzer bequem Aktionen in visuelle gemeinsame Ansichten aufnehmen kann.

#### Kurze Beschreibung der Zeichnungen

**[0032]** Die vorliegende Erfindung wird nun beispielhaft und mit Blick auf eine bevorzugte Ausführungsform weiter beschrieben, wie sie aus den beigefügten Zeichnungen hervorgeht, wobei:

**[0033]** [Fig. 1](#) ein Blockschaubild ist, das ein System gemäß der bevorzugten Ausführungsform der vorliegenden Erfindung darstellt;

**[0034]** [Fig. 2](#) eine schematische Darstellung ist, die ein System gemäß der bevorzugten Ausführungsform der vorliegenden Erfindung weiter darstellt;

**[0035]** [Fig. 3](#) eine Objektdarstellung ist, welche die Objekte und Beziehungen darstellt, die für das Erzeugen von gemeinsamen Ansichten unter Verwendung

von Aktionsobjekten und Menüdefinitionen gemäß der bevorzugten Ausführungsform der vorliegenden Erfindung notwendig sind; und

[0036] **Fig. 4** eine Darstellung ist, die den Steuerungsfluss zeigt, wenn ein Benutzer eine Auswahl in einer gemeinsamen Ansicht erzeugt, die zum Aufruf der zugehörigen Operation führt.

[0037] Die **Fig. 5A** bis **Fig. 5F** zeigen verschiedene Beispiele von visuellen Ansichten, die für die Darstellung von Aktionen verwendet werden.

Beschreibung einer bevorzugten Ausführungsform

[0038] Das System der bevorzugten Ausführungsform wird mit Blick auf **Fig. 1** erörtert, bei der es sich um ein Blockschaubild handelt, das die Hauptfunktionselemente des Systems zeigt. **Fig. 1** zeigt ein typisches Computersystem **1** mit einem darauf installierten Betriebssystem **35**, einer Zentraleinheit **2**, einem Speicher **15**, einer Speichereinheit **8**, die als Teil des Systems eingebaut sein kann oder auf die entfernt zugegriffen werden kann, sowie einer Anzeigeeinheit **55**, die mit dem Computersystem verbunden ist.

[0039] Der Prozessor **2** ist so aufgebaut, dass er Anzeigedaten an das Anzeigemittel **50** zur Anzeige auf der Anzeigeeinheit **55** weiterleitet. Wie der Fachmann weiß, ist die Funktionsweise des Anzeigemittels in der Technik allgemein bekannt und muss darum hier nicht ausführlicher behandelt werden. Es genügt der Verweis, dass es die verschiedenen Software- und Hardwarekomponenten beinhaltet, mit denen Daten an die Anzeigeeinheit **55** übertragen werden. Beim IBM OS/2-Betriebssystem kommt hierfür eine OS/2-Schicht mit der Bezeichnung „Presentation Manager“ zum Einsatz.

[0040] Bei der bevorzugten Ausführungsform der Erfindung wird auf der Anzeigeeinheit **55** eine Standard-GUI angezeigt, die dem Benutzer eine Anzahl von Fenstern und Symbolen anzeigt. Eine Software-Anwendung **7** ist einem oder mehreren dieser Fenster der GUI zugehörig. Damit der Benutzer mit der Anwendung **7** interagieren und auf diese Weise bestimmte Operationen aufrufen kann, stellt die GUI vorzugsweise visuelle Ansichten bestimmter Aktionen bereit, die der Benutzer auswählen kann. Zusätzlich hierzu können bestimmte nichtvisuelle Ansichten wie z.B. Schnellzugriffsdaten in einer Schnellzugriffstabelle dem Benutzer die Auswahl von Tasten auf der Tastatur ermöglichen, um so Aktionen und damit die entsprechende Operation aufzurufen.

[0041] Der Benutzer wählt die Ansichten über eine geeignete Eingabeeinheit aus, die mit dem Computersystem **1** verbunden ist. Beispiele für derartige Eingabeeinheiten sind eine Maus **70**, eine Tastatur **80**, ein Mikrofon **90** oder auch ein Stift **100**. Wie der

Fachmann weiß, verwendet der Benutzer für visuelle Ansichten wie Schaltflächen, Menüleisten und Symbolleisten in der Regel die Maus **70**. Er kann Aktionen nach Wunsch jedoch auch über Tasteneingaben auf der Tastatur **80** (z.B. Schnellzugriffstasten), Berührungseingaben über den Stift oder Spracheingaben über das Mikrofon auswählen. Unabhängig von der Art seiner Erzeugung wird das Signal vom Eingabemittel **3** empfangen und anschließend zur Analyse an das Erkennungsmittel **4** weitergeleitet. Dabei ist dem Fachmann bewusst, dass die Funktion des Eingabemittels **3** üblicherweise durch das Betriebssystem/die GUI des Systems bereitgestellt wird; ein Beispiel hierfür ist der Maustreibercode, wenn die Eingabeeinheit eine Maus ist. Das Erkennungsmittel **4** ist vorzugsweise ein vom Prozessor **2** gesteuerter Softwareprozess, um zu erkennen, ob das Eingangssignal einer Auswahl einer bestimmten Ansicht einer Aktion entspricht. Bei der bevorzugten Ausführungsform wird die Art und Weise, in der diese Erkennung stattfindet, als Teil einer Aktionsrahmenstruktur bereitgestellt, die mit Blick auf die **Fig. 2** bis **Fig. 4** noch ausführlicher erläutert wird. Dabei muss die Funktion allerdings nicht zwangsweise als Teil der Aktionsrahmenstruktur bereitgestellt werden, sondern das Erkennungsmittel **4** kann – wie der Fachmann weiß – auch an anderer Stelle bereitgestellt werden.

[0042] Nachdem das Erkennungsmittel **4** erkannt hat, dass eine Benutzerauswahl einer Ansicht erfolgt ist, wird das zugehörige Objekt **20** im Speicher **15** benachrichtigt. Danach ruft das Objekt unter der Steuerung durch den Prozessor **2** eine angegebene Operation auf, die in dem Objekt definiert ist. Wenn die Ansicht beispielsweise einem Drucken-Aktionsobjekt entspricht, definiert das Drucken-Aktionsobjekt eine Drucken-Operation, die bei Auswahl der Drucken-Aktion aufgerufen wird.

[0043] Damit Objekte im Speicher **15** erzeugt werden können, wird ein Erzeugungsmittel **5** bereitgestellt. Hiermit kann der Anwendungsentwickler Objekte definieren, die von der Anwendung **7** verwendet werden sollen. Bei der bevorzugten Ausführungsform wird diese Funktion als Teil der Aktionsrahmenstruktur bereitgestellt, die mit Blick auf die **Fig. 2** und **Fig. 3** noch ausführlicher erläutert wird. Dabei ist eine Aktionsrahmenstruktur jedoch nicht zwingend erforderlich. Alternativ kann der Entwickler – wie dem Fachmann bekannt ist – unter Verwendung standardmäßiger OOP-Methoden seine eigenen Objekte erzeugen. Durch die Definition von Objektklassen, die standardmäßige Ansichtsattribute und -prozeduren bereitstellen, welche für Objekte in der Klasse von Belang sind, kann der Entwickler danach problemlos mehrere Instanzen direkt aus diesen Klassen erzeugen.

[0044] Das System der bevorzugten Ausführungsform umfasst ferner ein Erstellungsmittel **6**, mit dem

unter Steuerung durch den Prozessor **2** gemeinsame Ansichten von Menüdefinitionsobjekten **20** erstellt werden, die im Speicher **15** gespeichert sind. Menüdefinitionsobjekte werden wie Aktionsobjekte durch das Erzeugungsmittel **5** erzeugt, und das Erstellungsmittel **6** dient anschließend zum Erstellen gemeinsamer Ansichten wie beispielsweise Symbolleisten, Menüleisten und Schnellzugriffstabellen aus diesen Ansichten. Bei bevorzugten Ausführungsformen ist das Erstellungsmittel **6** ebenfalls Teil der Aktionsrahmenstruktur, wobei dem Fachmann jedoch klar sein dürfte, dass diese Funktion auch an anderer Stelle bereitgestellt werden kann. Es muss lediglich eine Funktion bereitgestellt werden, mit der sich die gemeinsame Ansicht aus den Definitionen im Menüdefinitionsobjekt erstellen lässt. Der Prozess der Erstellung gemeinsamer Ansichten wird weiter unten mit Blick auf [Fig. 3](#) näher erläutert.

**[0045]** Im Folgenden wird das System der bevorzugten Ausführungsform ausführlicher und mit Blick auf [Fig. 2](#) erläutert, wobei es sich um ein Blockschaubild handelt, das die Hauptfunktionselemente des Systems zeigt. Wie bereits erwähnt, wird das System bei der bevorzugten Ausführungsform mit einer Aktionsrahmenstruktur **10** bereitgestellt, die einen Satz von Basis-Objekten **20** umfasst, die mit Anwendungscode logisch miteinander „verdrahtet“ werden können, um so eine Aufgabe auszuführen. Die Aktionsrahmenstruktur **10** verwendet das zugrunde liegende Dateisystem **30** des Betriebssystems, um die Ansichtsattribute, die den Ansichten der in der Rahmenstruktur **10** vorhandenen Objekte zugehörig sind, aus dem Systemspeicher zu laden und im Speicher **35** zu speichern.

**[0046]** Nachdem die von einer bestimmten Anwendung benötigten Menüdefinitionen und Aktionsobjekte erstellt wurden, werden mit dem Anzeigeverwaltungssystem **40** des Betriebssystems (im Falle des IBM OS/2-Betriebssystems vom „Presentation Manager“ bereitgestellt) die notwendigen Signale für die Anzeigschnittstelle **50** erzeugt, um die Ansichten dieser Objekte auf der Anzeigeeinheit **55** anzuzeigen. Üblicherweise werden diese Ansichten auf einer GUI **60** mit Fensteranzeige angezeigt. Angesichts der Einführung neuer Ein- und Ausgabemechanismen wie z.B. Sprache dürfte jedoch klar sein, dass zusätzlich zu oder anstelle der Anzeigeeinheit **55** auch andere Ausgabeeinheiten wie beispielsweise Sprachausgabeeinheiten verwendet werden können.

**[0047]** Über die GUI **60** kann der Benutzer mit dem System interagieren, um Auswahlmöglichkeiten wahrzunehmen. Der Benutzer kann die Auswahl auf mehrere Arten vornehmen, unter anderem mit einer standardmäßigen Zeigereinheit wie z.B. einer Maus **70**, einer Tastatur **80**, einer Spracheingabeeinheit **90**, einer Stifteingabeeinheit **100** oder über einen Berührungsbildschirm. Unabhängig davon, wie der Benut-

zer die Auswahl vornimmt, erzeugen die Einheiten-treiber ein Ereignis, das von dem Ereignisverwaltungssystem **110** des Betriebssystems verarbeitet wird. Beim IBM OS/2-Betriebssystem wird die Ereignisverarbeitungs-funktion durch das Betriebssystem und den Presentation Manager verarbeitet, indem ein standardmäßiges Steuerungs- und Benachrichtigungssystem mit Fensteranzeige bereitgestellt wird.

**[0048]** Wie weiter unten ausführlicher erörtert wird, kann das Auswahlereignis auf eines der Objekte **20** innerhalb der Aktionsrahmenstruktur **10** abgebildet werden, und dies verursacht wiederum den Aufruf einer Routine, welche die angeforderte Aufgabe unabhängig davon ausführt, wie und wo das Auswahlereignis erzeugt wird.

**[0049]** Im Folgenden werden die Basis-Objekte innerhalb der Aktionsrahmenstruktur **10** mit Blick auf [Fig. 3](#) erläutert, bei der es sich um eine Objektdarstellung handelt, die zeigt, wie gemäß der bevorzugten Ausführungsform der vorliegenden Erfindung Objekte erstellt werden, um mehrere Ansichten zu ermöglichen, und wie ferner über Beziehungen zwischen Objekten gemeinsame Ansichten erzeugt werden können.

**[0050]** Im Rahmen der bevorzugten Ausführungsform wird ein Basis-Aktionsobjekt **200** erzeugt, das alle Ansichtsattribute enthält, die einer bestimmten Aktion zugehörig sind. Bei der bevorzugten Ausführungsform stellt die Aktionsrahmenstruktur **10** einen durch die Anwendung erweiterbaren Satz von Ansichtsattributen **212** für gängige Aktionstypen **210** bereit, der im zugrunde liegenden Dateisystem **30** gespeichert wird. Auf diesen einzelnen Satz von Aktionstypen können mehrere Objekte oder Anwendungen innerhalb des Systems zugreifen, so dass sich die Anpassung der Aktionstyp-Daten auf jedes beliebige Objekt bzw. jede beliebige Anwendung auswirkt, die diese gängigen Aktionstypen verwendet.

**[0051]** Zu gängigen Aktionstypen zählen Aufgaben wie „Ausschneiden“, „Kopieren“, „Einfügen“ und „Drucken“, und für jeden Aktionstyp sind alle Ansichtsattribute gespeichert, die benötigt werden, um Ansichten der Aktion darzustellen. Derartige Ansichtsattribute beinhalten üblicherweise die folgenden Informationen:

- Menütext und Schaltflächentext (z.B. „Drucken“)
- Grafik (z.B. Drucken-Pixelmuster oder -Symbol)
- Schnellzugriffsdaten (z.B. „Strg+D“)
- Mnemotechnische Tastaturbelegung (z.B. „D“ für die Menü-Navigation)
- Textsensitive Hilfe (z.B. „Druckt das ausgewählte Objekt“)
- Stilattribute (geben an, ob die Aktion für den Dauerbetrieb verfügbar ist oder ob sie dem Benutzer innerhalb einer Aktionspalette (einer Liste der verfügbaren Aktionen) angezeigt wird)

- Kategorie (gibt den Typ der Aktion an, z.B. Drucken-Aktion)
- Sprachmuster (sofern definiert)
- Berührung (sofern definiert)

**[0052]** Da die Aktionstyp-Attribute **212** für das Erzeugen von Instanzen eines Aktionsobjekts **200** verwendet werden, können diese Daten als klassenweite Daten für einen bestimmten Aktionstyp betrachtet werden. Dies bedeutet, dass es lediglich einen Satz von Daten für einen Aktionstyp **210** „Drucken“ gibt, auch wenn eine Anwendung über eine Reihe von Drucken-Aktionsobjekten **200** verfügt, die beispielsweise für den Druckvorgang auf verschiedenen Druckern stehen.

**[0053]** Zusätzlich zu Aktionstyp-Attributen enthält das Aktionsobjekt **200** selbst Daten, die für eine bestimmte Instanz einer Aktion von Belang sind. So weist z.B. jedes Aktionsobjekt Folgendes auf

- einen eindeutigen Bezeichner **214** (wird für die Ereignisweiterleitung von einer Benutzerauswahl verwendet)
- einen momentanen Status **216** (gibt an, ob die Aktionsinstanz verfügbar ist, sich im Dauerbetrieb befindet, nicht verfügbar ist oder den Status „Nicht definiert“ aufweist)
- eine Rückfragefunktion **218** (gibt den durch die Anwendung definierten Code an, der infolge der Auswahl der Aktion aufgerufen wird)

**[0054]** Bei der bevorzugten Ausführungsform muss der Entwickler, der eine gängige Aktion wie beispielsweise „Drucken“ erzeugen möchte, lediglich die Drucken-Operation **220** erzeugen und einen Zeiger **218** für diese Operation auf das Aktionsobjekt bereitstellen. Die Drucken-Operation wird dann aufgerufen, sobald die Aktion ausgewählt wird. Die Drucken-Operation kann auf einer Aktionstyp-Ebene **220** bereitgestellt werden, so dass, falls mehrere Drucken-Aktionsobjekte im System vorhanden sind, diese bei einer Auswahl durchweg dieselbe Drucken-Funktion **220** aufrufen. Alternativ hierzu kann die Drucken-Funktion auf einer Instanzen-Ebene definiert werden, so dass für jede Drucken-Aktionsinstanz eine andere Rückfragefunktion gegeben ist.

**[0055]** Die Festlegung und Abfrage von Aktionstyp-Attributen **212** erfolgt über das Aktionsobjekt **200**. Aus diesem Grund unterstützt das Aktionsobjekt bestimmte „Methoden“, um die Fähigkeit zur Durchführung bestimmter Funktionen bereitzustellen, z.B.:

- Festlegen und Abfragen der Attribute des Aktionsobjekts
- Speichern des momentan Status der Aktion, d.h. ob die Aktion hervorgehoben, verfügbar oder im Dauerbetrieb ist
- Benachrichtigen etwaiger „Beobachter“, sobald Attribute geändert wurden; wenn alle Ansichtsattribute an derselben Stelle abgelegt und aktua-

lisiert werden, kann das Aktionsobjekt bei einer Änderung der Attribute oder des Status einer Aktion alle Ansichten, die es momentan anzeigen, benachrichtigen, so dass sie ihre Anzeige auf den neuesten Stand bringen können.

- Ermöglichen einer Speicherung unterschiedlicher Typen von Ansichtsdefinitionen für die Aktion, d.h. einer Symbolleistenansicht, einer Entwurfsansicht
- Ermöglichen das Erzeugen eines eindeutigen Ereignisbezeichners, so dass alle Auswahl- und Eingabemechanismen für die Aktion ein einziges Ereignis für die Anwendung erzeugen, unabhängig davon, wie der Benutzer den Aufruf erzeugt hat.
- Speichern und Wiederherstellen der Aktionsattribute

**[0056]** Aus der obigen Beschreibung wird deutlich, dass durch die Bereitstellung eines derartigen Objekts gängige Aktionen erzeugt werden können, die alle gängigerweise definierten visuellen und nichtvisuellen Ansichtsattribute enthalten. Da das Aktionsobjekt nicht die tatsächliche, von einer ausgewählten Aktion durchgeführte Operation verarbeitet, kann eine Anwendung die Mechanismen für die Auswahlauswahl problemlos übernehmen, und die tatsächliche, auf die Anwendungsdomäne anwendbare Operation kann dann von der Anwendung definiert werden.

**[0057]** Die obige Verfahrensweise verringert den Arbeitsaufwand seitens des Entwicklers, Aktionen zur Verfügung zu stellen, die vom Benutzer ausgewählt werden können. Außerdem löst sie die meisten der Probleme, die sich aus der mangelnden Übereinstimmung von Oberfläche und Funktionalität gängiger Aktionen in unterschiedlichen Anwendungen ergeben, da sie einen Großteil der Arbeit, die üblicherweise benötigt wird, um eine Einheitlichkeit der Ansichten bei verschiedenen Aktionsansichten aufrechtzuerhalten, überflüssig macht, da diese Aufgabe durch das Zusammenspiel zwischen Aktionsobjekt und Aktionsansichten übernommen wird. Wenn auf der Anwendungsebene etwaige Ansichten einer Aktion definiert sind, können diese als ein Beobachter des Aktionsobjekts registriert werden, so dass die Ansicht benachrichtigt werden kann, wenn sich die Attribute oder der Status einer Aktion ändert. Es obliegt dann der durch die Anwendung definierten Ansicht, ihre Darstellung bei Bedarf zu ändern.

**[0058]** Die obige Beschreibung hat erläutert, wie die Aktionsobjekte aufgebaut sind und wie dies das einfache Erzeugen konsistenter Ansichten derartiger Objekte ermöglicht. Im Folgenden wird erörtert, wie das System der bevorzugten Ausführungsform gemeinsame Ansichten verarbeitet. Damit gemeinsame Ansichten bei gleichzeitiger Beibehaltung der Oberflächen- und Auswahlkonsistenz erzeugt werden

können, wird eine baumartige Struktur erzeugt, die Gruppen von Aktionsinstanzen enthält. Hierfür wird ein Menüdefinitionsobjekt **230** erzeugt, das diese Struktur erhält, indem es einen einfachen Satz von Menüeinträgen **232** speichert, der entweder für eine Aktionsinstanz wie beispielsweise das Drucken-Aktionsobjekt **200**, ein Trennzeichen, mit dem innerhalb einer visuellen Anzeige zwischen Gruppen unterschieden wird, oder aber für eine andere Menüdefinitionsinstanz steht, die wiederum einen anderen einfachen Satz von Menüeinträgen enthält. Auf diese Weise wird eine baumartige Struktur erzielt, indem Menüdefinitionsinstanzen in andere Menüdefinitionsinstanzen eingebettet werden.

**[0059]** Bei der bevorzugten Ausführungsform wird ein grundlegendes Menüdefinitionsobjekt **230** erzeugt, das alle Ansichtsattribute enthält, die einer angegebenen Menüdefinition zugehörig sind. Dabei stellt die Aktionsrahmenstruktur **10** vorzugsweise einen vorgegebenen Satz von Ansichtsattributen **242** für gängige Menüdefinitionstypen **240** bereit, die im zugrunde liegenden Dateisystem **30** gespeichert sind. Im Gegensatz zu den standardmäßigen Aktionstyp-Definitionen **210** lassen sich die Menütyp-Definitionen **240** nicht durch Objekte oder Anwendungen anpassen, da sich Inhalt und Reihenfolge in Menüs auf einzelne Anwendungen beziehen und nicht auf mehrere Anwendungen anwendbar sind. Somit stellt das Menüdefinitions-Objekt die Anpassungsfunktion von Menüs sowie die Lade- und Speicherfunktionalität bereit, die für das Laden und Speichern einzelner Menüinhalte und -reihenfolgen notwendig ist.

**[0060]** Gängige Menütypen beinhalten üblicherweise ein Standard-Menü, ein Datei-Menü, ein Bearbeiten-Menü und ein Hilfe-Menü. Für jeden Menütyp sind sämtliche Ansichtsattribute vorhanden, die für das Erstellen gemeinsamer Ansichten benötigt werden. Diese Daten beinhalten zumeist die folgenden Informationen:

- Menütext (z.B. „Datei“)
- Grafik (z.B. ein Datei-Pixelmuster oder -Symbol)
- Mnemotechnische Tastaturbelegung (z.B. „D“ für die Menü-Navigation)
- Textsensitive Hilfe (z.B. „Aktionen für die Dateibearbeitung“)
- Kontext (z.B. „Immer“, d.h. dieses Menü wird unabhängig von der Anwendung bzw. vom Objekt-kontext angezeigt)
- Einträge (z.B. NEUER Aktionstyp, Aktionstyp ÖFFNEN, TRENNZEICHEN, Aktionstyp SPEICHERN, Aktionstyp SPEICHERN UNTER, TRENNZEICHEN, Aktionstyp DRUCKEN)

**[0061]** Bei der bevorzugten Ausführungsform muss der Entwickler, der eine gängige Menüdefinition wie beispielsweise eine Menüdefinition „Datei“ erzeugen möchte, lediglich sicherstellen, dass die Aktionstypen

in der Menüeintragsliste des Datei-Menütyps **240** bereits erzeugt wurden und dass eine Rückfragefunktion registriert wurde. Wenn dies nicht der Fall ist, ist die Aktion nicht in der Menüeintragsliste der Menüdefinitionsinstanz enthalten. Dies bringt gewisse Vorteile mit sich, da gängige Menüdefinitionen für das Erzeugen von Anwendungsmenüs verwendet werden können, ohne dass die Anwendung hierfür alle Aktionen innerhalb des Satzes unterstützen muss.

**[0062]** Menütyp-Attribute **242** werden über das Menüdefinitionsobjekt **230** festgelegt und abgefragt. Aus diesem Grund unterstützt das Menüdefinitionsobjekt **230** bestimmte „Methoden“, um die Fähigkeit zur Durchführung bestimmter Funktionen bereitzustellen, z.B.:

- Festlegen und Abfragen der Attribute des Menüdefinitionsobjekts
- Benachrichtigen etwaiger „Beobachter“, sobald Attribute geändert wurden
- Speichern und Wiederherstellen von Menüdefinitionsattributen

**[0063]** Durch die Bereitstellung von standardmäßigen Aktions- und Menütypen werden die Konsistenz der Benutzeroberfläche und die Entwicklungsproduktivität erhöht. Dabei dürfte dem Fachmann jedoch klar sein, dass das Konzept von Aktionsobjekten und Menüdefinitionen auch ohne die Verwendung vordefinierter Aktionstypen **210** und Menütypen **240** realisiert werden kann. Indem alle Ansichtsattribute für eine Drucken-Aktion direkt innerhalb eines einzigen Aktionsobjekts **200** (anstelle durch den Verweis auf einen Drucken-Aktionstyp) definiert werden, lassen sich beispielsweise dennoch mehrere Ansichten der Drucken-Aktion erzeugen und mehrere Auswahlmechanismen unterstützen. Außerdem kann eine beliebige Anzahl von Menüdefinitionen dennoch ein und dasselbe Aktionsobjekt enthalten, und eine beliebige Anzahl von „gemeinsamen Ansichten“ kann aus ein und derselben Menüdefinition erzeugt werden.

**[0064]** Nachdem eine Menüdefinitionsinstanz **230** erzeugt wurde, lassen sich aus einer einzigen Menüdefinition standardmäßige gemeinsame Ansichten erzeugen. Diese Ansichten beinhalten eine Standard-Menüleiste **250**, eine Standard-Symboleiste **260**, ein Kontext- oder Aufklappmenü sowie eine Schnellzugriffstabelle **270** (die Schnellzugriffstabelle **270** wird ausführlicher in Zusammenhang mit [Fig. 4](#) erläutert). Andere standardmäßige gemeinsame Ansichten können eine Sprachtabelle und eine Berührungstabelle beinhalten.

**[0065]** Anhand der Anpassung kann der Benutzer ein beliebiges der visuellen Attribute der Aktion ändern, d.h. den Menütext, die Symboleistengrafik, die textensitive Hilfe usw. Außerdem stellt bei bevorzugten Ausführungsformen die Aktionsrahmenstruktur **10** eine Aktionspaletten-Ansicht **280** bereit, die

eine Liste aller in dem System verfügbaren Aktionen bereitstellt, die über ein anzeigbares Attribut verfügen. Diese standardmäßige Aktionspaletten-Ansicht **280** unterstützt vorzugsweise das Ziehen und Ablegen und ermöglicht dem Benutzer die Aufnahme von Aktionen in visuelle gemeinsame Ansichten. Sie ermöglicht dem Benutzer außerdem die direkte Bearbeitung von Attributen einer Aktion. Durch die Benachrichtigung zwischen dieser Standard-Ansicht **280** und dem Basis-Aktionsobjekt **200** werden alle gemeinsamen Ansichten, die anhand einer Menüdefinition erzeugt wurden, automatisch mit den neuen Attributen aktualisiert, sobald Änderungen vorgenommen werden.

**[0066]** Die Anpassung ermöglicht dem Benutzer auch die Änderung visueller Attribute von Menüdefinitionen **230** wie z.B. des Menütextes, der Symbolleistengrafik und der textsensitiven Hilfe. Da die Menüdefinition eine dynamische hierarchisch geordnete Abfolge von Aktionen darstellt, kann der Benutzer innerhalb von visuellen gemeinsamen Ansichten Einträge löschen und hinzufügen oder sie Bedarf auch in einer neuen Reihenfolge anordnen. Jede Anpassung von Menüattributen oder Inhalten wird automatisch jeder gemeinsamen Ansicht dieses Objekts mitgeteilt. Dies verringert wiederum den Arbeitsaufwand seitens des Entwicklers für das Bereitstellen konsistenter Ansichten von Menüdefinitionen.

**[0067]** Für den Fall, dass ein Benutzer versehentlich Aktionen löscht, die eigentlich benötigt werden, steht bei bevorzugten Ausführungsformen der vollständige Satz von Aktionen stets in der standardmäßigen Aktionspaletten-Ansicht **280** zur Verfügung.

**[0068]** Durch die Verwendung des Kontextattributs in einer Menüdefinition lassen sich Menüs definieren, die dem Benutzer nur dann angezeigt werden, wenn ein bestimmter Kontextstatus „wahr“ ist. Dabei ist der momentane Kontext einem Anwendungsfenster oder Client-Bereich zugehörig und bezieht sich im Allgemeinen auf eine Benutzerauswahl. Durch die Aufnahme des momentanen Kontextes in das Fenster oder den Client-Bereich, dem eine gemeinsame Ansicht zugehörig ist (so ist z.B. eine Menüleiste immer an ein Fenster geknüpft), wird die Menüdefinition zum Beobachter dieses Fenster- oder Client-Bereichsattributs. Wenn ihr eigenes internes Kontextattribut nicht mit dem momentanen Fenster- oder Client-Bereichskontext übereinstimmt, kann sie allen in ihr enthaltenen Ansichten mitteilen, dass diese verborgen werden sollen, während sie umgekehrt bei einem übereinstimmenden Kontext den Menüs mitteilen kann, dass sie angezeigt werden sollen. Dieses Merkmal ermöglicht ein automatisches kontextgesteuertes Verhalten für jede beliebige gemeinsame Ansicht.

**[0069]** Im Folgenden wird mit Blick auf [Fig. 4](#) der

Steuerungsfluss für die Durchführung einer Aufgabe als Ergebnis einer Benutzerauswahl gemäß der bevorzugten Ausführungsform der vorliegenden Erfindung beschrieben.

**[0070]** Dem Benutzer kann eine Reihe von Eingabemechanismen für die Auswahl von Aktionen über eine grafische Benutzeroberfläche zur Verfügung stehen. [Fig. 4](#) zeigt drei verschiedene Arten für das Aufrufen einer Aktion „Print“ („Drucken“): eine Mause Auswahl eines Menüleisteneintrags **300**, eine Mause Auswahl einer Symbolleistenschaltfläche **310** und eine Tastenkombination „Strg+P“ bzw. „Strg+D“ über die Tastatur **320**.

**[0071]** Wenn die standardmäßigen gemeinsamen Ansichten der Menü- und der Symbolleiste aus einer Menüdefinitionsinstanz erstellt werden, durchlaufen sie nacheinander die Liste der Menüeinträge und erzeugen für jede Aktionsinstanz in der Liste ein Fenster. Der eindeutige Aktionsbezeichner **214** wird als Fensterbezeichner für den zugehörigen Menüeintrag oder die Symbolleistenschaltfläche verwendet, die für die Aktion steht. Wenn der Benutzer somit entweder die Symbolleistenschaltfläche oder den Menüeintrag auswählt, stimmt das Ereignis, das erzeugt und über das Systemereignis-Leitwegprogramm **330** weitergeleitet wird, genau mit der Aktionsinstanz überein, die aufgerufen werden soll, da es den eindeutigen Aktionsbezeichner **214** enthält. Das Erzeugen eines Ereignisses wird durch das standardmäßige Benachrichtigungssystem mit Fensteranzeige innerhalb einer GUI bereitgestellt. Das Systemereignis-Leitwegprogramm **330** stellt dann die Funktion dar, die durch das Erkennungsmittel **4** aus [Fig. 1](#) bereitgestellt wurde.

**[0072]** Wenn die standardmäßige gemeinsame Ansicht der Schnellzugriffstabelle **270** aus einer Menüdefinition erstellt wird, durchläuft sie ebenfalls die Menüeintragsliste auf der Suche nach Aktionsinstanzen. Für jede Aktionsinstanz in der Liste fordert das Schnellzugriffstabellen-Objekt von der Aktionsinstanz deren Schnellzugriffsdaten an und nimmt die zurückgegebene Tastenkombination mit dem eindeutigen Aktionsbezeichner **214** in seine interne Referenztable auf. Die interne Referenztable ist eine Liste der Schnellzugriffsdaten **275**. Außerdem wird eine Fenstererkennung gespeichert, welche die Beziehung zwischen dem Satz aktiver Tastenanschläge bereitstellt, die Aktionen über die Tastatur und das aktive Fenster innerhalb der GUI ermöglichen, mit dem diese Aktionen interagieren. Wenn der Benutzer eine Tastenkombination drückt, prüft die Schnellzugriffstabelle anhand ihrer Referenztable, ob für die Tastenfolge eine Aktion registriert wurde. Wenn ja, erzeugt sie unter Verwendung des eindeutigen Aktionsbezeichners **214** ein Ereignis, das wiederum an das Systemereignis-Leitwegprogramm **330** weitergeleitet wird und genau mit der Aktionsinstanz überein-

stimmt, die aufgerufen werden soll. Wenn die Aktion **200** daraufhin ein Ereignis empfängt, ruft sie die registrierte Rückfragefunktion **220** auf, um die Aufgabe aufzurufen.

**[0073]** Aus der obigen Erörterung wird deutlich, dass sich mit dem System der bevorzugten Ausführungsform ein einziges Ereignis aus mehreren Ansichten erzeugen lässt. Somit ist es für die Anwendung nicht von Belang, wie der Benutzer den Aufruf erzeugt hat. Mit dem Mechanismus der bevorzugten Ausführungsform fragen die gängigen Aktionsansichten die Attribute des Aktionsobjekts direkt ab, wobei alle notwendigen Daten an einer Stelle gespeichert sind, und bei automatischer Gleichzeitigkeit der Ansichten muss sich der Anwendungsentwickler keine Gedanken mehr darum machen, dass Aktionsansichten synchron zueinander dargestellt werden.

### Patentansprüche

1. Computersystem mit einer Benutzeroberfläche, über die ein Benutzer eine Aktion auswählen kann, um eine gewünschte Operation aufzurufen, wobei die Aktion mittels einer Vielzahl von Ansichten ausgewählt werden kann, die über die Benutzeroberfläche bereitgestellt werden, wobei das System durch Folgendes gekennzeichnet ist:

ein Mittel für die Bereitstellung eines einzelnen Aktionsobjekts, das für eine Aktion steht, wobei das Aktionsobjekt (**200**) für jede der Ansichten, die zur Darstellung der Aktion verwendet werden können, die Attribute (**212**), welche zur Bereitstellung der Ansicht erforderlich sind, sowie einen Bezeichner (**214**, **218**) definiert, um die Operation (**220**), die bei Auswahl der Aktion aufgerufen werden soll, kenntlich zu machen; ein Erkennungsmittel (**4**), mit dem erkannt wird, wann eine Benutzerauswahl einer Ansicht erfolgt ist, und mit dem das zugehörige Aktionsobjekt (**200**) von der Auswahl unterrichtet wird; wobei das Aktionsobjekt auf das Erkennungsmittel (**4**) anspricht, um den Bezeichner zur Erzeugung eines Befehls zu verwenden, um die gewünschte Operation aufzurufen.

2. Computersystem nach Anspruch 1, wobei das Computersystem mit einer Anzeigeeinheit (**55**) verbunden werden kann, um eine grafische Benutzeroberfläche anzuzeigen, und wobei mindestens eine der Ansichten, die zur Darstellung der Aktion verwendet werden können, eine visuelle Ansicht ist, die über die grafische Benutzeroberfläche angezeigt werden kann, wobei die in dem Aktionsobjekt für die mindestens eine der Ansichten definierten Attribute die Daten beinhalten, die für die Erzeugung der visuellen Ansicht erforderlich sind.

3. System nach Anspruch 1 oder Anspruch 2, das weiter ein Erstellungsmittel (**6**) für die Erstellung einer gemeinsamen Ansicht von Aktionen aus einer Menü-

definition (**230**) umfasst, welche die gemeinsame Ansicht definiert, wobei die Menüdefinition über Einträge (**232**) verfügt, die Aktionsobjekte und/oder andere Menüdefinitionen angeben, die durch die gemeinsame Ansicht dargestellt werden sollen, und wobei das Erstellungsmittel (**6**) so angeordnet ist, dass es auf diese Aktionsobjekte und/oder anderen Menüdefinitionen verweist, um die gemeinsame Ansicht zu erstellen.

4. System nach Anspruch 3, wobei die Menüdefinition weiter Kontextinformationen umfasst, die definieren, wann eine gemeinsame Ansicht für die Auswahl durch einen Benutzer angezeigt oder aktiviert werden soll.

5. System nach einem beliebigen vorangegangenen Anspruch, wobei der in dem Aktionsobjekt definierte Bezeichner einen eindeutigen Bezeichner (**214**) für das Aktionsobjekt und einen Rückrufmechanismus (**218**) umfasst, um die gewünschte Operation (**220**) bei Auswahl des Aktionsobjekts aufzurufen.

6. System nach einem beliebigen der Ansprüche 2 bis 5, das weiter eine Aktionspalettenansicht (**280**) umfasst, die auf der Anzeigeeinheit (**55**) angezeigt werden kann, um eine Liste der verfügbaren Aktionen bereitzustellen.

7. Verfahren für den Betrieb eines Computersystems mit einer Benutzeroberfläche, über die ein Benutzer eine Aktion auswählen kann, um eine gewünschte Operation aufzurufen, wobei die Aktion mittels einer Vielzahl von Ansichten ausgewählt werden kann, die über die Benutzeroberfläche bereitgestellt werden, wobei das Verfahren durch die folgenden Schritte gekennzeichnet ist:

(a) Bereitstellen eines einzelnen Aktionsobjekts (**200**) zur Darstellung einer Aktion, wobei das Aktionsobjekt für jede der Ansichten, die zur Darstellung der Aktion verwendet werden können, die Attribute (**212**), welche zur Bereitstellung der Ansicht erforderlich sind, sowie einen Bezeichner (**214**, **218**) definiert, um die Operation (**220**), die bei Auswahl der Aktion aufgerufen werden soll, kenntlich zu machen, wobei das Aktionsobjekt auf eine derartige Auswahl anspricht, um den Bezeichner zum Aufrufen der gewünschten Operation zu verwenden;

(b) Erkennen des Auftretens einer Benutzerauswahl einer Ansicht; und

(c) als Reaktion auf eine Erkennung einer Benutzerauswahl einer Ansicht, Unterrichten des zugehörigen Aktionsobjekts (**200**) von der Auswahl und somit Aufrufen der gewünschten Operation.

8. Verfahren nach Anspruch 7, wobei das Computersystem mit einer Anzeigeeinheit (**55**) verbunden werden kann, um eine grafische Benutzeroberfläche anzuzeigen, und wobei mindestens eine der Ansichten, die zur Darstellung der Aktion verwendet werden

können, eine visuelle Ansicht ist, die über die grafische Benutzeroberfläche angezeigt werden kann, wobei die in dem Aktionsobjekt für die mindestens eine der Ansichten definierten Attribute die Daten beinhalten, die für die Erzeugung der visuellen Ansicht erforderlich sind.

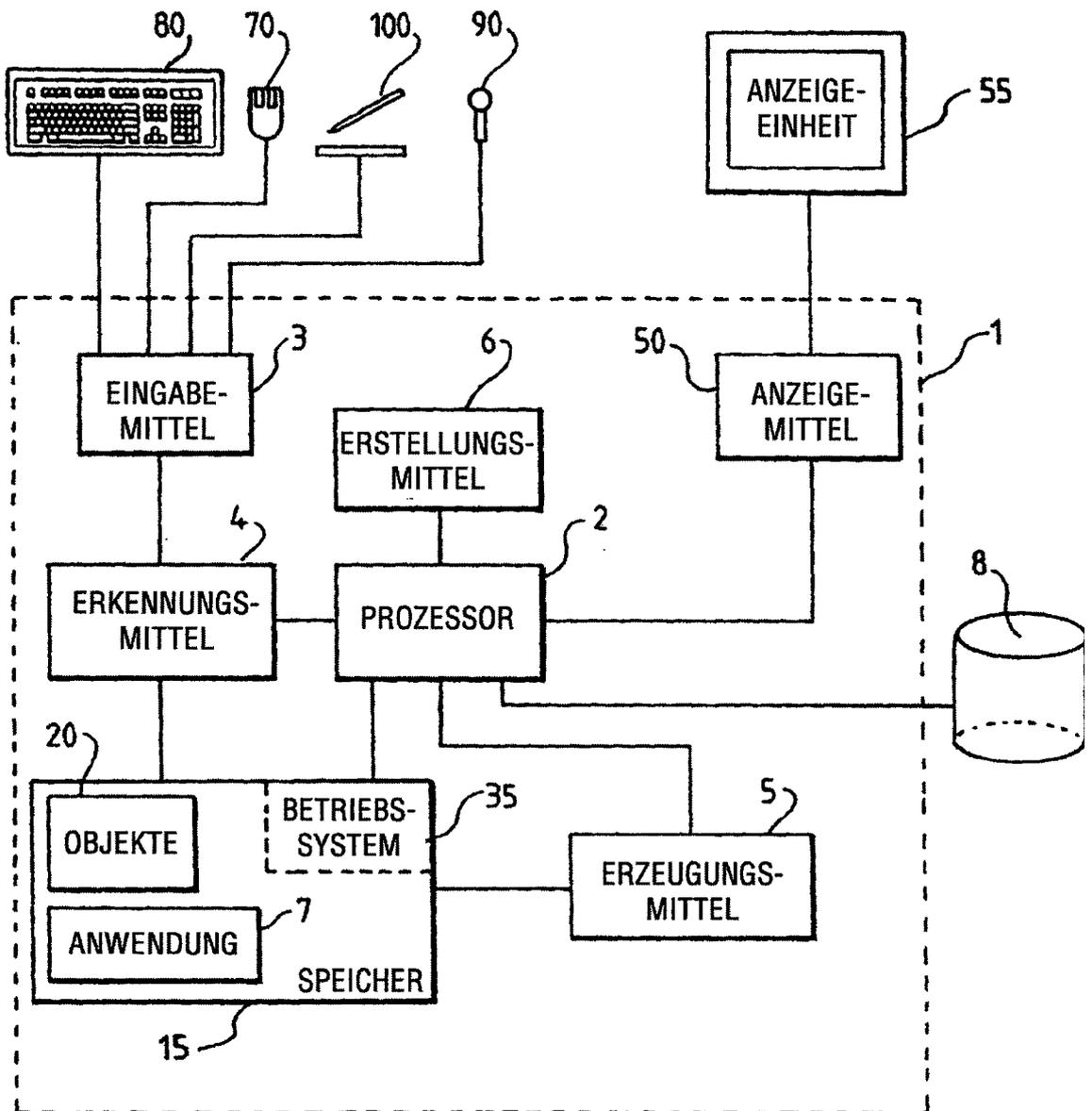
9. Verfahren nach Anspruch 7 oder Anspruch 8, das weiter den Schritt des Erstellens einer gemeinsamen Ansicht von Aktionen aus einer Menüdefinition (**230**) umfasst, welche die gemeinsame Ansicht definiert, wobei die Menüdefinition über Einträge (**232**) verfügt, die Aktionsobjekte und/oder andere Menüdefinitionen angeben, die durch die gemeinsame Ansicht dargestellt werden sollen, und wobei der Schritt des Erstellens das Verweisen auf diese Aktionsobjekte und/oder anderen Menüdefinitionen beinhaltet, um die gemeinsame Ansicht zu erstellen.

10. Verfahren nach Anspruch 9, wobei die Menüdefinition weiter Kontextinformationen umfasst, die definieren, wann eine gemeinsame Ansicht für die Auswahl durch einen Benutzer angezeigt oder aktiviert werden soll.

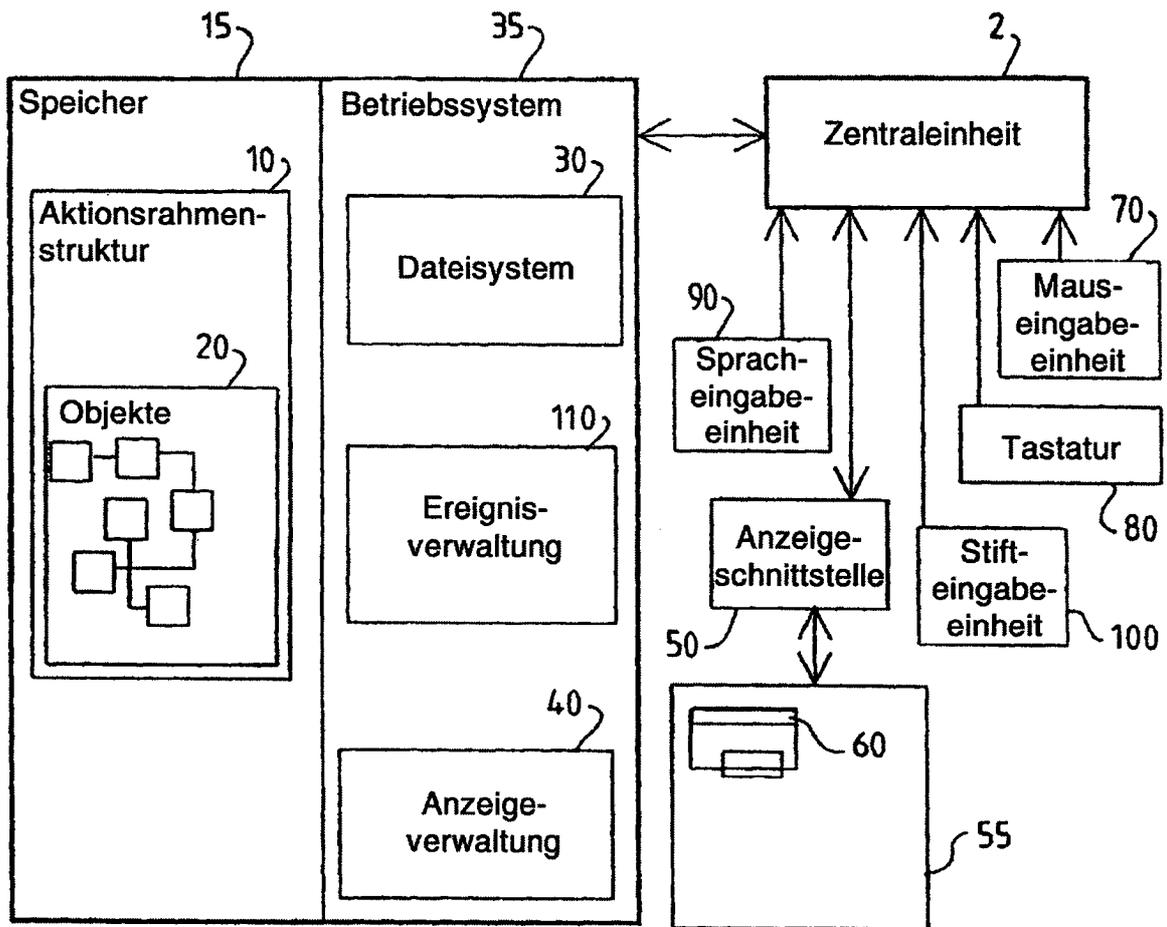
11. Verfahren nach einem beliebigen der Ansprüche 7 bis 10, das weiter den Schritt des Bereitstellens einer Aktionspalettenansicht (**280**) umfasst, die auf der Anzeigeeinheit (**55**) angezeigt werden kann, um eine Liste der verfügbaren Aktionen bereitzustellen.

Es folgen 6 Blatt Zeichnungen

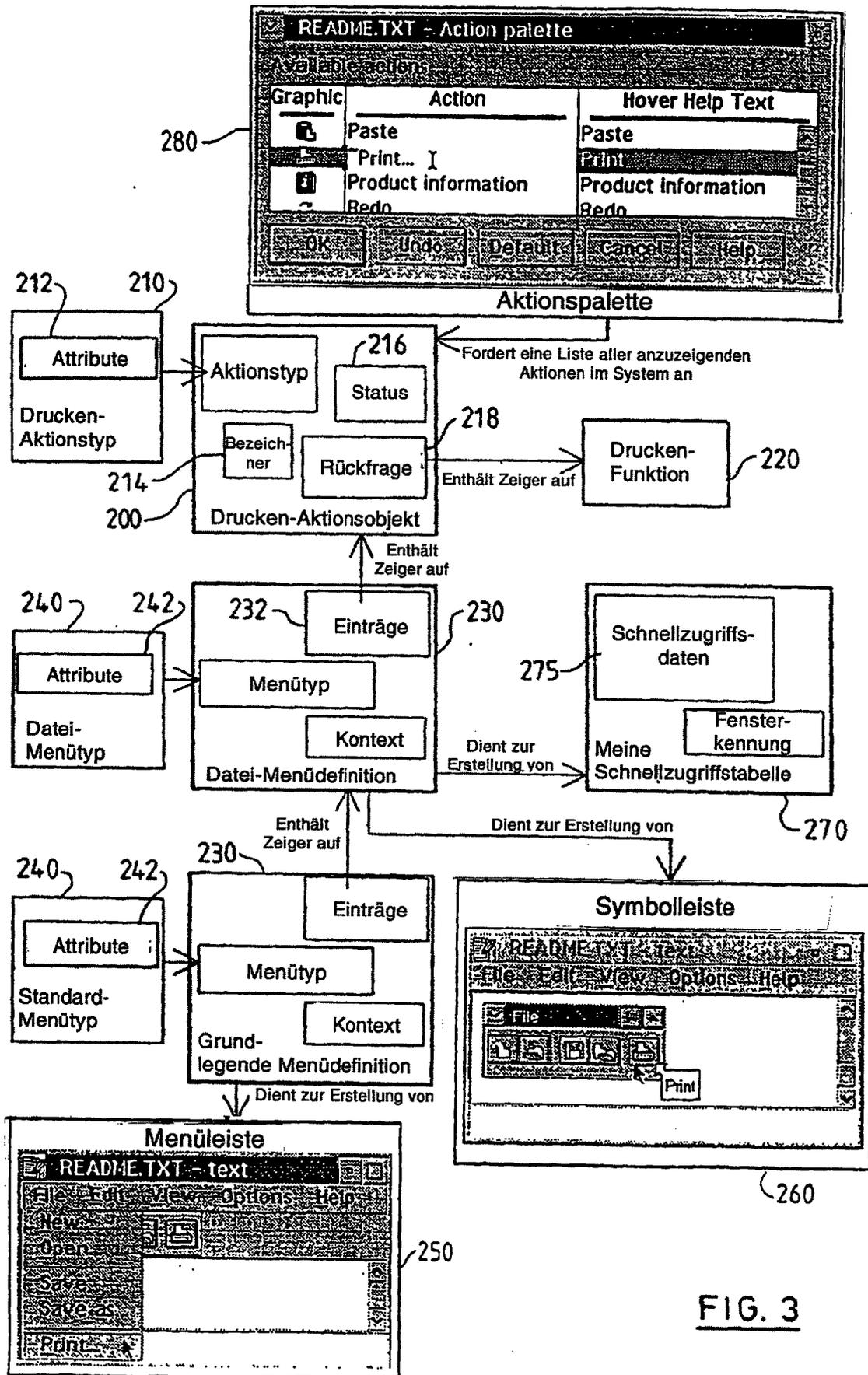
Anhängende Zeichnungen



**FIG. 1**



**FIG. 2**



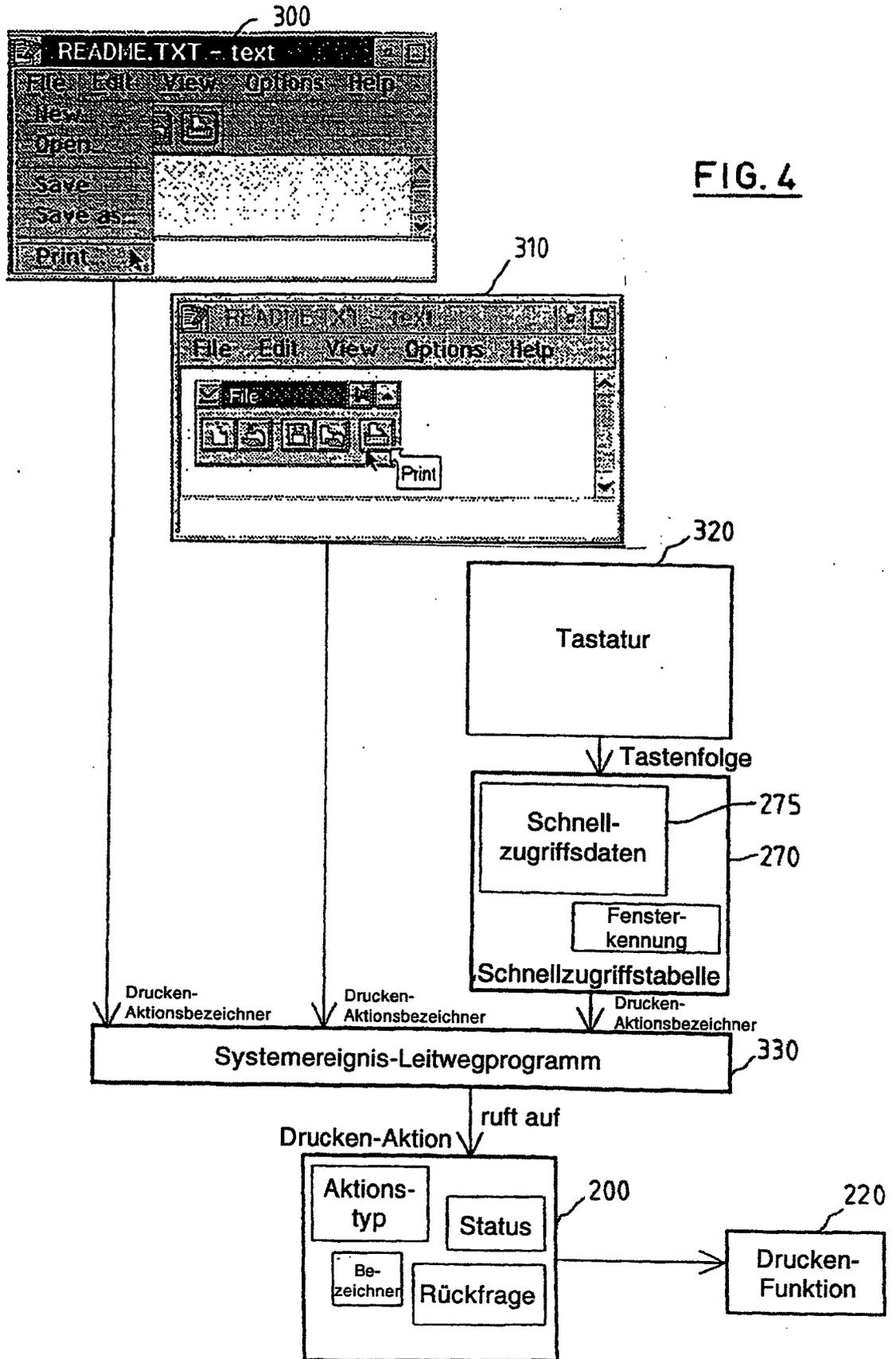


FIG. 4

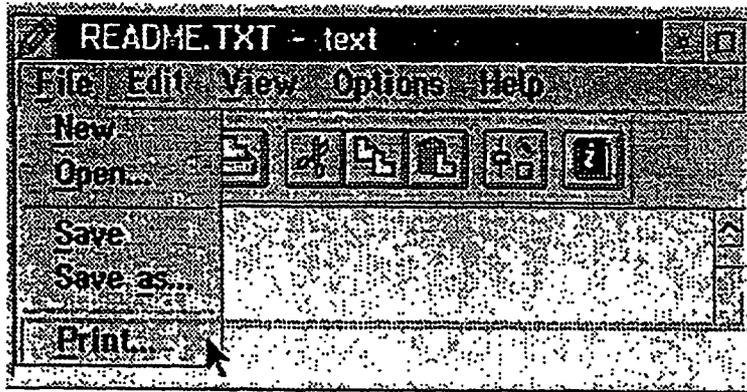


FIG. 5A

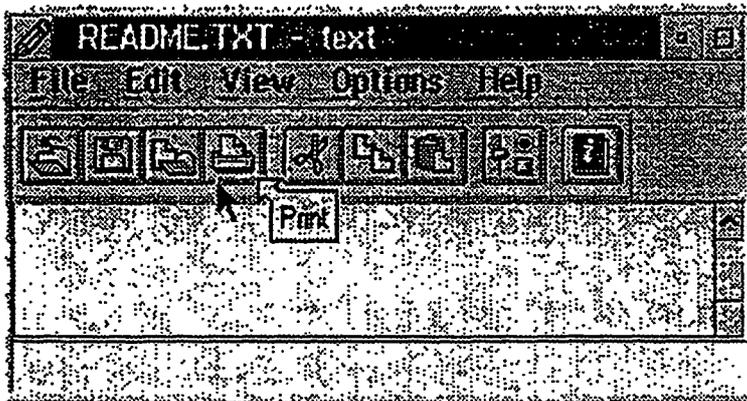


FIG. 5B

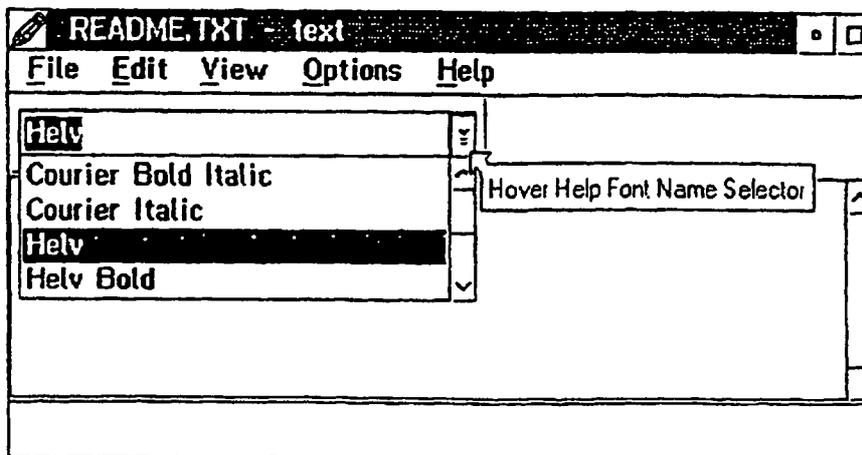


FIG. 5C



FIG. 5D

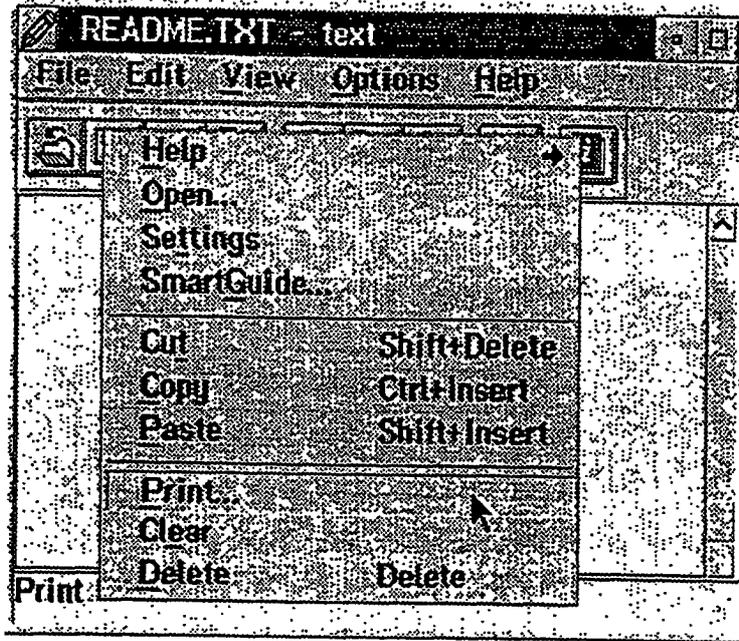


FIG. 5E

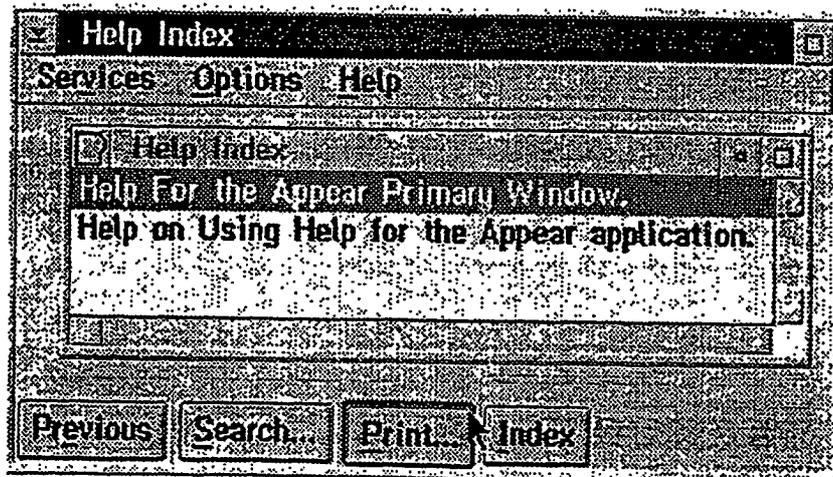


FIG. 5F