

## Die wichtigsten Begriffe der Dateiverwaltung

### Datensatz, Datensatzfeld:

Ein Datensatz ist eine Menge zusammengehöriger Informationen (zum Beispiel Daten eines Artikels: Artikelnummer, Preis etc.). Die einzelnen Elemente des Datensatzes werden Felder genannt (Bild 1).

### Datei:

Eine Menge gleichartiger Datensätze (zum Beispiel alle Artikel, die Sie führen) wird als Datei bezeichnet (Bild 1).

### Dateiverwaltung:

Programm, das eine Datei verwalten kann, das heißt, das in der Lage ist, vom Benutzer vorgenommene Anfragen zu beantworten («Welche Adresse hat Herr Gustav Werner?») und den Änderungsdienst zu erledigen (Eintragen neuer Datensätze, Ändern von Datensätzen, Löschen von Datensätzen).

### Benutzerschnittstelle:

Unter Benutzerschnittstelle versteht man die Art und Weise, in der die Kommunikation zwischen Dateiverwaltung und Benutzer erfolgt, in der dieser zum Beispiel Anfragen an das Programm stellt, oder den sortierten Ausdruck der Datei veranlaßt. Die zwei gebräuchlichsten Arten der Benutzerschnittstelle sind die Menüsteuerung und die Steuerung mit Hilfe von Abfragesprachen.

### Menüsteuerung:

Die Benutzung geschieht interaktiv, im Dialog mit dem Programm.

### Abfragesprachen:

Der Benutzer formuliert seine Wünsche an die Dateiverwaltung mit Hilfe einer eigens dafür zur Verfügung gestellten Sprache; er schreibt gewissermaßen ein Abfrageprogramm.

### Datenstrukturen:

Man unterscheidet logische und physische Datenstrukturen. Logische Datenstrukturen definieren die Beziehung der einzelnen Elemente einer Datei zueinander, physische Datenstrukturen entscheiden über die Art und Weise, in der die Daten auf einem Speichermedium abgelegt werden.

### Sequentieller Zugriff:

Zugriffsart, bei der auf die Daten einer Datei nur nacheinander in der Reihenfolge der Abspeicherung zugegriffen werden kann. Um den x-ten Datensatz zu lesen, müssen erst die Datensätze 1, 2, ..., x-1 gelesen werden. Häufig verwechselt werden »sequentieller Zugriff« und »sequentielle Datei«: eine sequentielle

Datei ist eine Datei, deren einzelne Elemente sich lückenlos hintereinander auf dem Speichermedium befinden (im Gegensatz beispielsweise zur Hashing-Datei). Die Art der Datei erlaubt jedoch noch keine Schlußfolgerung auf die Zugriffsart.

### Direktzugriff:

Durch Angabe des Ortes, an dem sich ein Datensatz befindet (Floppy: Spur und Sektor) kann auf diesen direkt zugegriffen werden (Bild 2). Eine Sonderform des Direktzugriffs ist der relative Zugriff: Hierbei wird das Speichermedium in »Records« unterteilt, deren Länge beim Aufbau der Datei festgelegt wird und die der maximalen Datensatzlänge entspricht. Jeder Datensatz wird in einem solchen Record abgespeichert. Unter Angabe der jeweiligen

Recordnummer kann auf jeden Datensatz der Datei direkt zugegriffen werden.

### Schlüssel/Index:

Eine Datei wird nach einem oder mehreren Schlüsseln geordnet. Ein solcher Schlüssel ist meist ein Feld des Datensatzes, zum Beispiel »Name« oder »Artikelnummer«.

### Schlüssel-/Indexdatei:

Eine Datei, die zu jedem Datensatz das Schlüsselkriterium enthält, nach dem die Datensätze geordnet sind, sowie einen Zeiger auf den eigentlichen Datensatz in der Datensatzdatei.

### Index-sequentielle Datei:

Dateiform, bei der außer der Datensatzdatei noch eine Schlüsseldatei existiert. Da diese Schlüsseldatei von jedem Datensatz nur einen Teil enthält (zum Beispiel den Namen), ist sie klein im Vergleich zur Datensatzdatei und kann dadurch in den Computerspeicher geladen und in

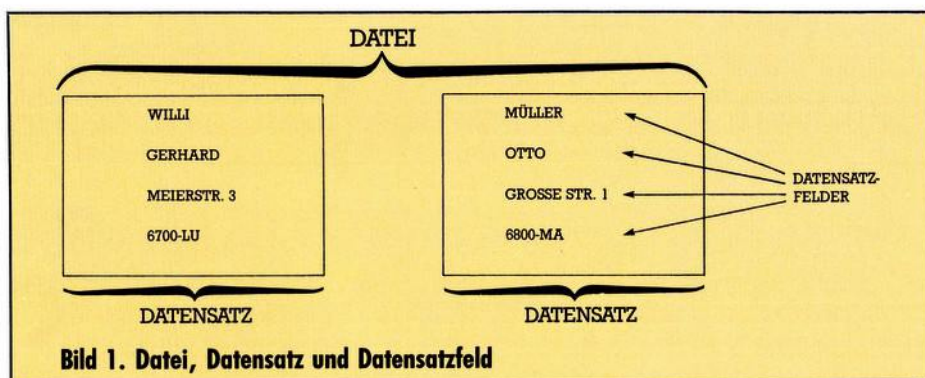


Bild 1. Datei, Datensatz und Datensatzfeld

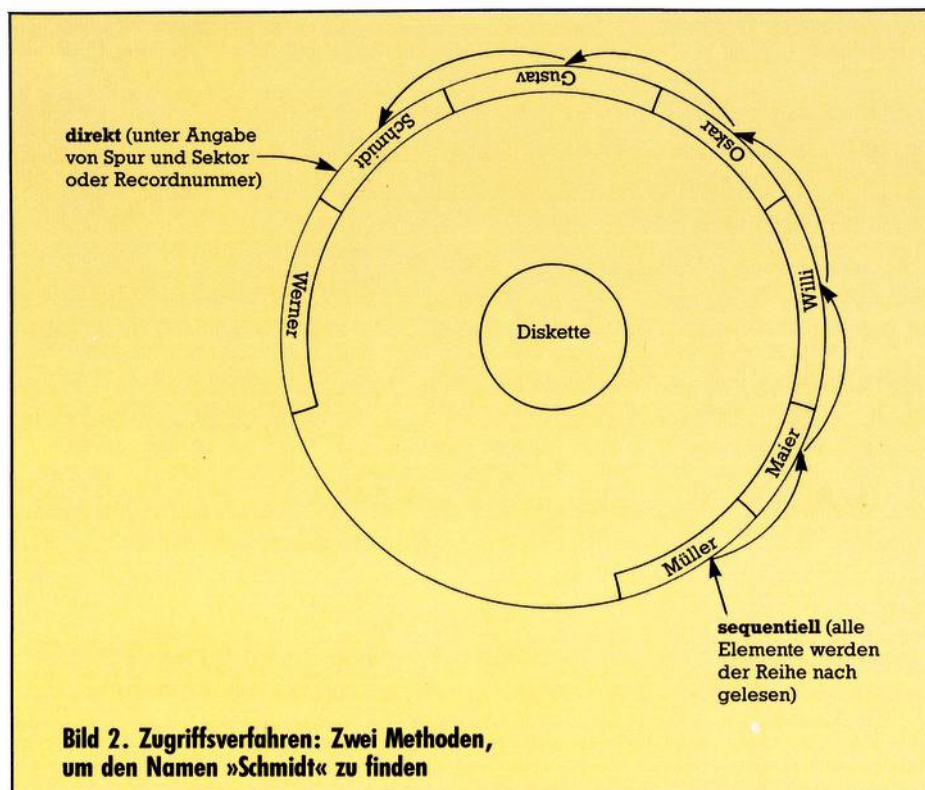


Bild 2. Zugriffsverfahren: Zwei Methoden, um den Namen »Schmidt« zu finden

diesem erheblich schneller als auf einem externen Speichermedium durchsucht werden (Bild 3). Wenn die Suche nach einem Datensatz nicht nach dem Schlüssel erfolgt, geht der Vorteil der schnellen Suche über die Indexdatei verloren; die Datensatzdatei muß sequentiell durchsucht werden (außer wenn das Dateiverwaltungsprogramm die Verwendung mehrerer Schlüssel gestattet).

#### Ordnung:

Die Ordnung einer Datei nach einem bestimmten Kriterium kann durch sequentielle Reihung entsprechend der Ordnung (einfache Liste), oder durch Verknüpfung der Daten über Zeiger (verkettete Listen, Baumstrukturen) geschehen.

#### Zeiger (Pointer):

Um Ordnung in eine Datei zu bringen, deren Elemente nicht in der Reihenfolge dieser Ordnung abgespeichert sind, verwendet man für

jedes Element der Datei einen oder mehrere Zeiger, die auf den Speicherort des in der Reihenfolge nächsten oder auch vorhergehenden Elements zeigen.

#### Binäre Suche:

Eine — zum Beispiel alphabetisch oder numerisch — geordnete Datei läßt sich mit Hilfe der binären Suche sehr schnell durchsuchen: Zuerst wird auf das mittlere Element der Datei zugegriffen. Ist das gesuchte Element größer, anschließend auf die Mitte der rechten Hälfte der Datei und so weiter. Bei jedem Suchschritt wird die Länge der noch zu durchsuchenden Datei halbiert.

#### Hashing:

Ein Verfahren, bei dem der Ort, an dem ein Datensatz abgespeichert wird, mit Hilfe eines geeigneten Algorithmus direkt aus dem Inhalt dieses Datensatzes ermittelt wird (zum Beispiel, indem die Quersumme der ASCII-Zeichen des ersten Da-

tenatzfeldes als Recordnummer beim relativen Zugriff verwendet wird). Hashing-Verfahren zeichnen sich durch außergewöhnlich schnellen Datenzugriff aus. Nachteilig ist, daß die Daten so abgespeichert werden, wie sie eingegeben wurden, und daher nicht geordnet sind.

#### Baumstruktur:

Eine häufig verwendete Datenstruktur, bei der jedem Datensatz mehrere Zeiger zugeordnet werden, durch die die Ordnung der Datei hergestellt wird; zum Beispiel einen Zeiger auf das »nächstgrößere« Element und einen Zeiger auf das »nächstkleinere« Element (Binärbaum, Bild 4).

#### Sortierverfahren:

Eine der häufigsten Aufgaben einer Dateiverwaltung ist die sortierte Ausgabe von Daten. Bei einer Hashing-Datei muß die Datei in einem solchen Fall unbedingt sortiert werden, da die Datensätze nach keiner erkennbaren Ordnung abgelegt sind. Aber auch bei einer geordneten Datei ist die Sortierung immer dann unumgänglich, wenn eine sortierte Ausgabe nach einem anderen Ordnungskriterium gewünscht wird als jenem, das zum Aufbau zum Beispiel der Baumstruktur verwendet wurde. Es gibt eine Vielzahl von Sortierverfahren (Sortieren durch Auswahl, durch Vergleich, durch Mischen, siehe 64er, Ausgabe 4/85 ff, Effektives Programmieren).

#### Interne Sortierverfahren:

Interne Sortierverfahren (zum Beispiel Quicksort) können immer dann angewendet werden, wenn sich die zu sortierenden Elemente im RAM befinden.

#### Externe Sortierverfahren:

Externe Sortierverfahren sind notwendig, wenn der Arbeitsspeicher (RAM) zu klein ist, um alle zu sortierenden Elemente aufzunehmen. In diesem Fall müssen die Elemente direkt auf dem externen Speicher (Floppy/Platte) sortiert werden. Meist werden mit Hilfe interner Sortierverfahren auf dem externen Speicher mehrere kleine, vorsortierte Dateien erzeugt, die dann mit entsprechenden Sortierverfahren weiterverarbeitet werden.

#### Reorganisation (update):

Dateien müssen teilweise aktualisiert werden. Dieser Vorgang wird als Reorganisation bezeichnet. Eine solche Reorganisation ist zum Beispiel immer dann nötig, wenn der Benutzer die Struktur der Daten verändern will (zum Beispiel neue Felder hinzufügt oder löscht) oder es zu Programmabstürzen kam.

(Said Baloui/gk)

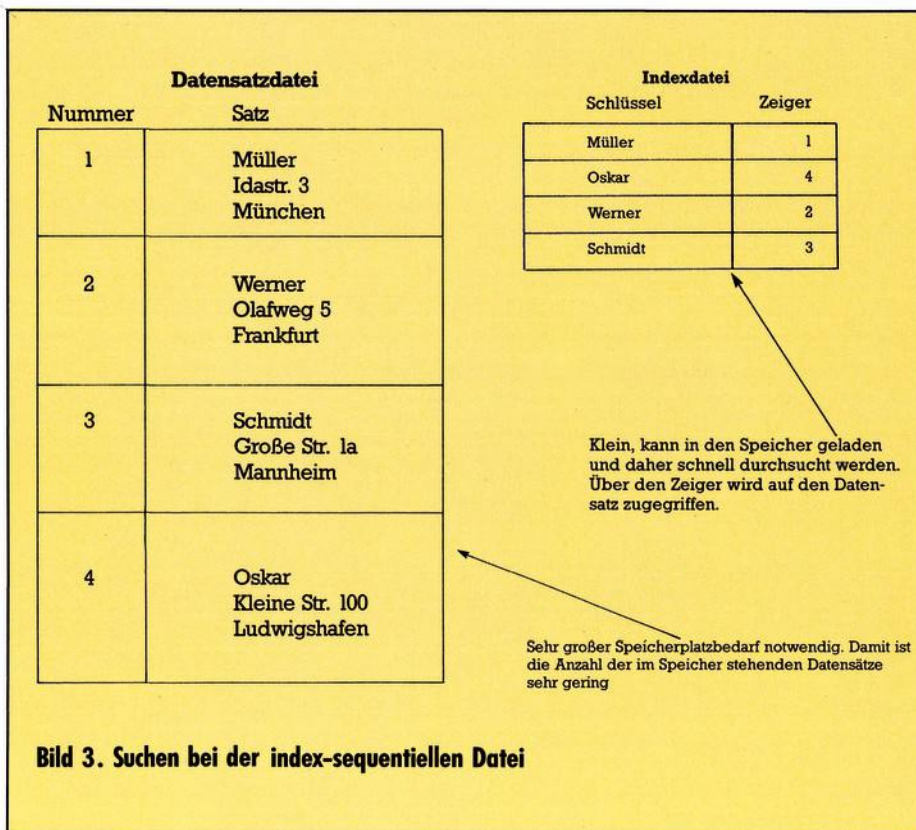


Bild 3. Suchen bei der index-sequentiellen Datei

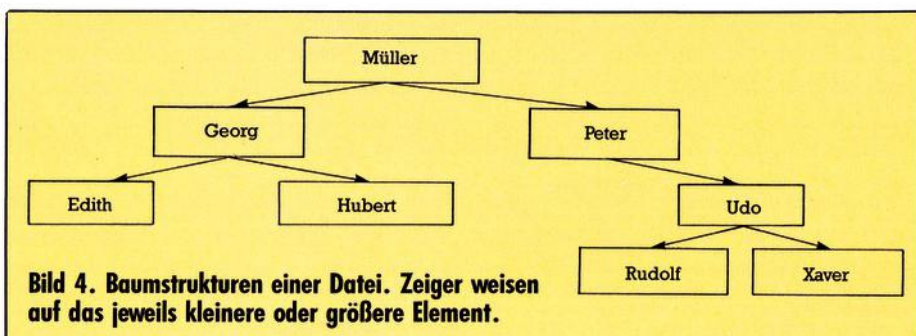


Bild 4. Baumstrukturen einer Datei. Zeiger weisen auf das jeweils kleinere oder größere Element.