EFFIZIENTE DATENANALYSE

Hannes Hahne *
Frank Schulze *

Abstract Die Fähigkeit zur Analyse großer Datenmengen sowie das extrahieren wichtiger Erkenntnisse daraus, sind in der modernen Unternehmenswelt ein entscheidender Wettbewerbsvorteil geworden. Umso wichtiger ist es, dabei vor allem nachvollziehbar, reproduzierbar und effizient vorzugehen.

Der Beitrag stellt mit dem Instrument der skriptbasierten Datenanalyse eine Möglichkeit vor, um diesen Anforderungen gerecht zu werden.

1 WARUM DATEN ANALYSIEREN?

Daten sind das Abbild von Geschäftsprozessen in IT-Systemen. Sie sind die Grundlage jeder planerischen Tätigkeit sowie der Überwachung, Steuerung und Prognose aller Aktivitäten, welche der Leistungserstellung im Unternehmen dienen.

Im Unternehmensalltag besteht die Herausforderung oft darin, aus der Vielfalt von Unternehmensdaten jene zu vereinnahmen, zu bereinigen, zu analysieren und repräsentativ darzustellen, welche zur Beantwortung einer spezifischen Frage notwendig sind.

Hinzu kommt, dass Analyse- und Planungsprojekte i. d. R. von Personengruppen getragen werden, oft in wechselnder Zusammensetzung und über unterschiedlich lange Zeiträume hinweg. Die zur Projektbearbeitung nötigen Daten werden zudem im zeitlichen Verlauf meist ergänzt oder aktualisiert. Straffe Terminpläne lassen wenig Raum für aussagekräftige Dokumentation – die Nachvollziehbarkeit des Analyseganges, auch nach Projektende, muss dennoch gewährleistet bleiben.

*TU Dresden, Professur für Technische Logistik

2 WIE DATEN ANALYSIEREN?

Die Datenanalyse ist kein neues Arbeitsgebiet. Seit langem steht eine Vielzahl leistungsfähiger Softwareinstrumente zur Verfügung, um Daten zu archivieren (z. B. *tar*), zu komprimieren (z. B. *gzip*), umzuformen (z. B. *awk*), zu analysieren (z. B. *SQLite* oder *R*) und zu visualisieren (z. B. *Gnuplot* oder *R*).

2.1 SKRIPTE

Eine Methode, welche sich bewährt und als robust gegenüber den o. g. Herausforderungen erwiesen hat, ist das skriptbasierte Analysieren von Daten. Dabei werden alle auf der Originaldatenbasis durchzuführenden Analyseoperationen in Textform als einfache Programme (Skripte) niedergeschrieben. Dies erfordert selbstverständlich eine hinreichende Kenntnis der Befehlssätze der genannten (im Allgemeinen hervorragend dokumentierten) Softwarewerkzeuge.

Der wesentliche Vorteil dieser Methode ist die implizite Dokumentation jedes Analyseschritts durch die Skripte. Jeder Untersuchungsschritt wird so nachvollziehbar gemacht – gute Quelltexte sind oft selbsterklärend, weiterführende Informationen zu getroffenen Annahmen und Entscheidungen werden an den relevanten Stellen in Freitext-Kommentaren untergebracht. Solche Skripte erlauben Projektneulingen und selbst programmieraversen Personen einen guten Zugang und die Möglichkeit zur produktiven Mitarbeit. Der erzeugte Quellcode dient zugleich als Report über die geleistete Analysearbeit gegenüber dem Auftraggeber.

Die dargelegte Methodik hat einen weiteren wichtigen Vorzug. Aufgrund der ausschließlich in Skripten beschriebenen Bearbeitungsschritte werden die Originaldaten nicht manipuliert. Der Output der Analyse wird in neuen Dateien oder Datenbanken abgelegt. Somit bleiben die (Roh-) Datenquelle, die Datenanalyse (Skripte) und die Ergebnisdarstellung separiert. Dies führt dazu, dass nach einer Änderung der Basisdaten (z. B. durch eine Aktualisierung) keine oder nur wenige Anpassungen an den Skripten notwendig sind, um aktuelle Analyseergebnisse zu erhalten. Die Datenbasis verbleibt im

Originalzustand, so dass alle Projektbeteiligten stets einen gemeinsamen Bezugspunkt haben.

2.2 ABLAUF

Grundsätzlich ist vorab zu definieren, welche Daten zur Problemlösung benötigt werden. Stehen diese Daten zur Verfügung, schließen sich die nachfolgende Schritte an:

1. Datenvereinnahmung: Die Daten werden in tabellarischer und textbasierter Form als sequentielle Dateien bereitgestellt, so dass sie von den zur Analyse eingesetzten Werkzeugen eingelesen werden können. Diese Dateien bilden die Schnittstelle zwischen der Unternehmens-IT und dem Datenanalyseprojekt.

Da es sich oft um größere Mengen von Rohdaten handelt, werden sie komprimiert und erst bei Bedarf entpackt (z. B. mit *gzip*).

2. Datenbereinigung: Meist enthalten die vereinnahmten Datensätze Anomalien, welche die Analyse behindern oder verfälschen. Exemplarisch seien hier Codierungsfehler für Umlaute, unzulässige Trennzeichen oder Zeilenumbrüche, unnötige Dezimalstellen, falsche Datentypkonvertierungen für Datum und Uhrzeit oder überflüssige/redundante Informationen genannt. Dieser Schritt kann je nach Datenqualität einen größeren Zeitraum in Anspruch nehmen. Im Ergebnis liegen die tatsächlich für die Analyse benötigten Daten in strukturierter Form (meist als relationale Datenbank) vor.

Für die effiziente Verarbeitung großer Textdateien ist *awk* besonders geeignet.

3. Datenanalyse: Die vereinnahmten und bereinigten Datensätze werden mit Blick auf das Projektziel ausgewertet. Grundsätzlich stellt dieser Schritt die größten Anforderungen an den Analysten. Er ist getrieben von den (Ausgangs-) Fragestellungen und wird beeinflusst von den Erkenntnissen, die sich im Laufe der Analyse ergeben.

Die Aggregation von Daten erfolgt meist in Form von *SQL*-Queries. Eine schlanke und zugleich performante Datenbank-Engine ist

SQLite. Für komplexe statistische Analysen eignet sich *R* besonders gut.

4. Ergebnispräsentation: Zum Erkenntnisgewinn und als Ausgangspunkt für Diskussionen und Entscheidungen werden die Analyseergebnisse schließlich in tabellarischer Form zusammengestellt und zumeist in Form von Diagrammen visualisiert. Dies ist ein wichtiger kreativer Schritt, bei dem für einen bestimmten Sachverhalt bzw. eine Erkenntnis eine adäquate grafische Repräsentation entwickelt wird. Komplexe Grafiken lassen sich sehr gut mit *Gnuplot* (oder *R*) erstellen.