

Eine Flut von 64 Jobs

Heute mag man über die Zahl von 64 Jobs lächeln, aber anno 1976 war sie ein Problem. Und - das habe ich bereits damals eingesehen – war alles andere als eine Glanzleistung.

Die Idee war ganz einfach. Um Approximationen für bestimmte Funktionen zu erstellen, benötigte ich sehr genaue Funktionswerte an bestimmten Stützstellen. Genau das kostete viel Rechenzeit.

Die Rechenzeit pro Job konnte man zwar beliebig angeben, aber es gab Schwellwerte, die steuerten, wann ein Job anlaufen durfte. T10 hieß maximal 8 Sekunden CPU-Zeit, T20 maximal 16 Sekunden, T100 maximal 64 Sekunden. Das RRZN hat Jobs mit T10 bevorzugt behandelt, weil sie eben schnell wieder verschwunden waren.

T10 half mit aber nicht. Selbst mit T20 musste ich den Kuchen aufteilen. Es dauerte aber ziemlich lange, bis ein Job mit T20 anlief. Die Ausgabeliste war klein und harmlos, weil ich die Funktionswerte in permanenten Dateien gespeichert habe. In den Jobs musste ich auch nicht vorgeben, ab welchem Argument die Rechnung beginnen sollte. Jeder Job hat die letzte Datei eingelesen und damit auch das zuletzt benutzte Argument. Ich war sogar noch viel gemeiner. Jeder Job hat gerechnet bis zum Time Limit. Dann brach er ab. Den Abbruch habe ich mit einer EXIT-Karte abgefangen und dann die erstellte Datei katalogisiert. Trotzdem war es ein langwieriges Verfahren.

Erschwerend kam hinzu, dass auch die Zahl der T10-Job im RRZN sehr stark angestiegen war. Dadurch dauerte es noch länger, bis ein T20-Job anlief.

Es gab etwas, das sich „abhängige Jobs“ nannte. Jeder Job wurde über den Namen einer Kette identifiziert und hatte einen Haltezähler. Wenn dieser Zähler von anderen Jobs dieser Kette runtergezählt wurde (Steuerkarte TRANSF) und den Wert 0 erreichte, dann lief er an. Den Kopf dieser Kette musste man zuletzt einlesen, denn er hatte den Haltezähler 0 und lief sofort an. Wenn er an die TRANSF-Karte kam, musste der erste abhängige Job bereits im System sein.

Wegen der Wartezeiten bei den T20-Jobs war ich ziemlich genervt. Die Flut von Funktionswerten war nicht Ziel meiner Diplomarbeit. Sie waren lediglich das Mittel, um Approximationen zu erstellen.

In meinem Frust habe ich zwei Jobketten eingelesen, jede mit 32 Jobs. 64 Jobs von einem einzelnen Benutzer - das war im Sommer 1976 zu viel. Das Operating hat mit viel Mühe diese 64 Jobs aus der Input Queue entfernt und auf Band gesichert. Danach gab es für mich die berechnete Schelte. Reumütig bat ich darum, die 64 Jobs zu löschen. War nicht sofort möglich. Sie mussten erst wieder von der Sicherung in die Input Queue geladen werden. Nur dort konnte man sie löschen.

Die Idee an sich war nicht schlecht. Dass aber 64 Jobs eines einzelnen Benutzers in der Input Queue die Maschine derartig überfordern würden, war mir damals nicht bewusst. Ich bitte auch nach 41 Jahren noch um Entschuldigung dafür. Trotz meines Fehlverhaltens durfte ich aber den Rechner weiterhin benutzen.

Vielleicht tröstet es: Die ca. 2 MB an Datenkarten besitze ich heute noch, und zwar auf dem PC, an dem ich diese Texte erfasse. Immerhin sind es Funktionswerte mit ca. 28 Dezimalstellen. Später habe ich auf einem PC entsprechende Werte für andere Besselfunktionen berechnet, die nicht zum Thema meiner Diplomarbeit gehörten.

Und noch etwas habe ich dauerhaft gelernt. Wenn Mangel herrscht und man ihn nicht wirklich beseitigen kann, dann kann man ihn nur noch verwalten. Und das bedeutet, dass immer irgendjemand benachteiligt ist.

Viele Jobketten mit mehr als 300 Jobs

Nein keine Geschichte aus dem RRZN, sondern aus meinem späteren Berufsleben, in diesem Fall aus der Zeit von 1986 bis 1998 bei einem „Keksbäcker“, wie ein externer Mitarbeiter einmal abfällig geäußert hat.

Ich war Systemprogrammierer und verantwortlich dafür, dass der Rechner lief, egal was eingelesen wurde. Das Spooling System war JES3 und kannte ebenfalls abhängige Jobs und Haltezähler. Unternehmen, die über die Kosten von Software nicht nachdenken mussten, haben sich für tägliche Flut von Jobs ein Programmsystem geleistet, das die Jobs zeit- oder ereignisgesteuert einlas. Ein solches Programmsystem (Scheduler) war aber teuer und mit jährlichen Lizenzkosten verbunden. Diese Kosten hat mein damaliger Arbeitgeber gescheut, zumal JES3 die Möglichkeit von Jobnetzen bot. Tagsüber alle Jobs einlesen, auch den Kopf einer Kette mit Haltezähler 1 versehen und Terminkarten für das Operating, wann eine Kette gestartet werden durfte. Abends gegen 19 Uhr tummelten sich dann 300 bis 400 Jobs in der Input Queue, die darauf warteten, dass sie freigegeben wurden, um bis ca. 3 Uhr nachts abgearbeitet zu sein.

Es funktionierte. Mit JES2 (von der NASA als HASP entwickelt) wäre es nicht möglich gewesen. JES2 kennt keine Jobketten. Und dass dieses mittelständische Unternehmen JES3 hatte und sich damit eher wie ein großes Unternehmen (z.B. ein Automobilhersteller unweit von Hannover) verhielt, das hatte mein damaliger Arbeitgeber vorwiegend der Eloquenz der Verkäufer des Rechnerherstellers zu verdanken.