

Fehlertolerantes Koordinieren verteilter autonomer Dienste*

eva Kühn

Technische Universität Wien, Institut für Computersprachen
Argentinierstraße 8, 1040 Wien, Österreich
E-mail: eva@mips.complang.tuwien.ac.at

Abstract

Die Vernetzung von Rechnern führt zu neuen Einsatzbereichen und daher zu neuen Anforderungen an Programmiersprachen. Programmierung war bisher meist mit Erstellung von Softwarepaketen für einen Rechner gleichzusetzen. Sequentielle Programmiersprachen werden verwendet, die weder die Kommunikation zwischen parallelen Prozessen noch deren Synchronisation unterstützen. Neue, verteilte Programmiersprachen sind mit der Koordinierung von in einem im Netz angebotenen autonomen Diensten konfrontiert. Ähnliche Aufgaben entstehen bei der Verwaltung von Teilaufgaben in einem Projekt. Fehler und Ausfälle einzelner Komponenten sollten die Erfolgchance der Gesamtaufgabe nicht verhindern.

Wir beschreiben Techniken zur Koordination solcher Aufgaben. Die vorgeschlagenen Konzepte können in herkömmliche Programmiersprachen eingebaut werden. Wir zeigen, warum das logische Programmierparadigma besonders gut geeignet ist.

Schlüsselworte: Verteilte Sprachen, Koordination, Parallelität, logische Programmierung, Fehlertoleranz.

1 Einleitung

Netzwerke verbinden Rechner innerhalb von Institutionen oder können weltweit öffentlich zugänglich sein. Die Programmierung verteilter Rechner umfasst neben einer optimaleren Ressourcenausnutzung die Verwendung von autonomen Dienstleistungen und die Unterstützung kooperativen Arbeitens in Gruppen.

1.1 Verwendung von autonomen Dienstleistungen

In öffentlichen Netzen werden Dienste autonomer Institutionen angeboten. Dabei handelt es sich beispielsweise um den Zugriff auf Literaturdatenbanken oder „Mailboxen“, die Zurverfügungstellung hoher Rechenleistung auf Supercomputern, die Ver-

*Diese Arbeit wird vom FWF (Fonds zur Förderung der wissenschaftlichen Forschung), Projekt P9020-PHY „Multidatabase Transaction Processing“ als Kooperationsprojekt mit dem NSF (National Science Foundation) gefördert.

wendung spezialisierter Softwarepakete wie Gleichungslösern oder die Ausführung von Buchungstransaktionen bei Flugliniendatenbanken.

Man kann davon ausgehen, daß es für die meisten Problemstellungen bereits Softwarepakete gibt, deren Ankauf in vielen Fällen – etwa bei einmaliger Verwendung – nicht rentabel ist. Gewisse Dienste, wie etwa eine Geldüberweisung, müssen vom Anbieter ausgeführt werden. Dienstleistungen im Netz werden gegen Gebühren angeboten werden. Die Beschreibung und Verfügbarkeit angebotener Dienste werden von weiteren Dienstleistungsprogrammen verwaltet (Namensverwaltung). Die Koordinierung solcher Dienste unter Berücksichtigung von Kosten- und Zeitoptimierung ist eine neue Anforderung an verteilte Systeme.

Eine Aufgabe wird sich aus mehreren Diensten zusammensetzen, die bereits im Netz existieren. Existierende Systeme sind autonom und können nicht dahingehend verändert werden, daß sie nun mit anderen Systemen zusammenarbeiten – das wäre eine radikale Änderung, die ihrer ursprünglichen (sequentiellen) Entwurfsphilosophie widerspräche. Daher werden neue Werkzeuge zur Koordinierung verteilter Dienste benötigt, deren wesentliche Aufgabe die Kontrolle parallel existierender Abläufe (Prozesse) ist.

Unter der Annahme, daß Dienste autonom verwaltet werden, muß mit dem individuellen Ausfall einzelner Systeme gerechnet werden. Das sollte aber nicht zum Abbruch der gesamten Aufgabe und somit zum Verzicht auf bereits geleistete Dienste führen: Erstellte Teilaufgaben haben bereits Kosten verursacht und andere Wege müssen gefunden werden, um unter Berücksichtigung von Fehlern doch noch eine Lösung zu finden.

Heute kommerziell verwendete Betriebssystemsoftware unterstützt zwar den Zugriff auf Daten sowie den Aufruf von Programmen auf anderen Rechnern, aber es handelt sich hier nur um sehr einfache Protokolle, die die genannten semantischen und applikationsspezifischen Probleme nicht berücksichtigen. Auch herkömmliche Datenbanktransaktionen sind viel zu eingeschränkt. Deren „Alles oder Nichts“-Semantik führt unweigerlich zum Abbruch der Gesamtaufgabe (globalen Transaktion), sobald ein Teil fehlschlägt. Es wäre vielmehr wünschenswert, alternative Lösungswege für Teilaufgaben definieren zu können, denn viele äquivalente Dienste werden von verschiedenen Institutionen angeboten. Die Verwendung alternativer Dienste erhöht die Erfolgchance der globalen Transaktion. Die Spezifikation, welche Dienste bevorzugt zu verwenden sind, sollte möglich sein.

1.2 Kooperatives Arbeiten in Teams

Die Organisationsstruktur großer Projekte wird durch die technisch möglich gewordenen Kommunikationsstrukturen beeinflußt werden. Die Bearbeitung gemeinsamer Daten über gemeinsam beschreibbare „Black-Boards“, die von mehreren Mitarbeitern gleichzeitig verwendet werden, erfordert neue Formen der Koordination. Jeder Mitarbeiter muß sich auf Mitteilungen auf dem gemeinsamen Kommunikationsmedium verlassen können. Nachrichten dürfen weder verlorengehen noch zerstört werden. Das gleichzeitige Schreiben auf ein Objekt muß ausgeschlossen werden. Bei Entwurfsprojekten müssen beschlossene Planungsschritte festgehalten und ältere Versionen verwaltet werden.

Ein vom Projekt abhängiges Protokoll legt fest, ob Nachrichten von Einzelnen oder in Gruppen erarbeitet werden, bevor sie öffentlich dargestellt werden. Innerhalb von