

21 Prozesse und Kommunikation

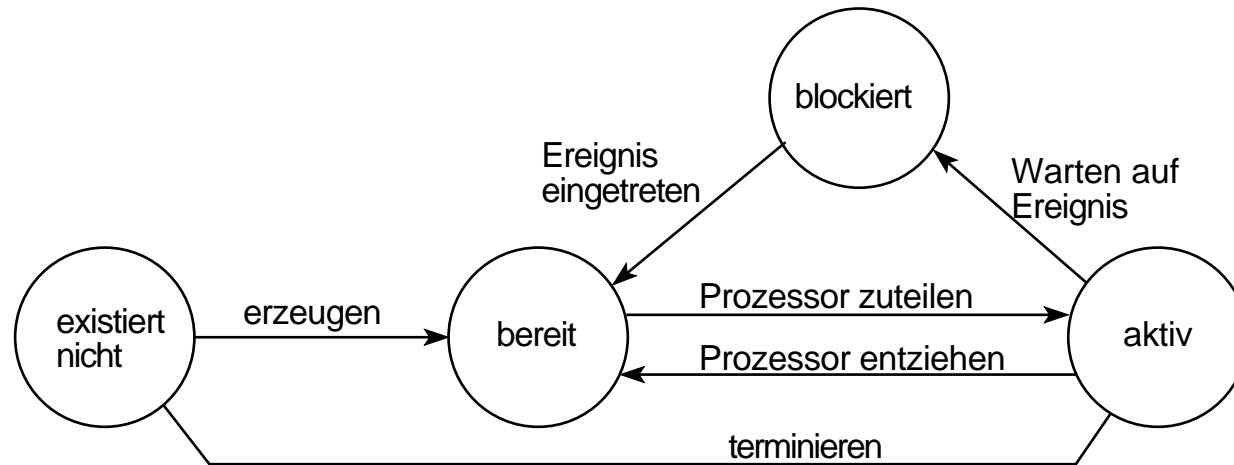
21.1 Einführung

- Gleichzeitige, koordinierte Bearbeitung mehrerer Aufgaben
- Geographische Verteilung
- ⇒ mehrere nebenläufige Prozesse

Prozess (process) – Eine durch ein Programm gegebene Folge von Aktionen, die sich in Bearbeitung befindet.

Nebenläufigkeit (concurrency) – Die parallele oder zeitlich verzahnte Bearbeitung mehrerer Aufgaben.

Mögliche Prozesszustände



- **Statische** Prozesse: alle Prozesse bei Systemstart erzeugt
- **Dynamische** Prozesse: Erzeugung/Terminierung im laufenden Betrieb

21.2 Prinzipien der Prozesskommunikation

Prozesse müssen miteinander **kommunizieren**:

- **Austausch von Information**
 - über **gemeinsame Speicher**
 - über **Nachrichten**
- **Synchronisation** des Arbeitsfortschritts
 - **Semaphore**
 - **Monitore**

Kommunikation über

gemeinsame Speicher

einfach und schnell:

- kein Kopieren
- Prozessumschaltung nur zwischen Beteiligten
- keine Dienstprozesse notwendig

Kommunikationspartner für Protokolle selbst verantwortlich

nicht verteilbar

Nachrichten

langsamer:

- Kopieren erforderlich
- zusätzliche Prozessumschaltungen
- Dienstprozess(e) erforderlich

vorhandene Protokolle nutzbar

problemlos verteilbar

21.3 Typische Kommunikationsverfahren

21.3.1 Kanal/Mailbox

- Aufbau eines realen oder virtuellen Kanals zwischen Sender und Empfänger
- **Asynchrone, gepufferte Übermittlung nicht adressierter Nachrichten (Mailbox-Paradigma)**
- Kommunikationssystem stellt Dienste bereit für
 - Kanaleröffnung / -schließung
 - Versand
 - Benachrichtigung
 - Empfang

21.3.2 Globaler Versand (Broadcast)

- Alle Knoten sind über einen **Bus** oder Ring **verbunden**
- Senderknoten setzt Nachricht ab
- Nachricht enthält Ziel- und/oder Quellenadresse
- **Kommunikationssystem**
 - **verteilt Nachricht an alle**
 - **verwaltet das Medium** und verhindert Kollisionen oder löst sie auf
- Jeder **Knoten** betreibt einen **Mithörprozess**, der
 - die Adressen jeder Nachricht **mithört**
 - bei Zieladressierung die für den Knoten bestimmten Nachrichten **empfängt**
 - bei Quellenadressierung aufgrund einer Adressentabelle die für ihn interessanten Nachrichten **empfängt**

21.3.3 Fernaufruf (Remote Procedure Call RPC / Remote Method Invocation RMI)

- Analogon zu lokalem Prozedur- /Methodenaufruf
- Softwaretechnisch sehr **einfach** und **klar** (verbirgt die Verteilung gegenüber den benutzenden Komponenten weitgehend)
- Kommunikationssystem übernimmt Benachrichtigung des Empfängers, Übertragung der Parameter und der Ergebnisse
- Sender wartet auf Ergebnis (**synchrone Kommunikation**)
- Kommunikationssystem stellt Dienste bereit zum Aufrufen, sich Aufrufen lassen und einen Namensdienst (wer ist aufrufbar)
- Adressat eines RPC/RMI muss ständig verfügbar sein – Problem des Ausfalls muss gelöst werden

21.3.4 World Wide Web

Drei Kommunikationstypen:

- **Synchroner Dateitransfer** :Klassisches Laden einer Seite von einem über eine URL adressierten Server
- **Synchroner Dienstaufwurf** (analog RPC)
 - Bearbeitung einer Anfrage (Suchmaschinen, Auskunftsdienste, ...)
 - Versand der Anfrage zum Server, Bearbeitung auf Server, Rückversand von Ergebnissen
 - zum Beispiel mit CGI-BIN, Java Servlets, ASP
- **Dynamisches Laden von verteiltem Programmcode**
 - Dynamisches Laden eines Programms von einem Server
 - Lokales Ausführen dieses Programms in geschützter Umgebung
 - Das Programm kann (unsichtbar für Klient) über das Netz kommunizieren
 - zum Beispiel Java-Applets

21.4 Bestimmung der Prozesse

- **Wieviele** Prozesse
- **Statische** oder **dynamische** Erzeugung
- Zuordnung von **Funktionalität**: Verteilung der Module auf Prozesse
- Bestimmung der **Kommunikationsbedürfnisse**, Festlegung der Kommunikationsart(n)
- Geographische **Verteilung**
- Zuordnung von **Ressourcen**

Mögliches Vorgehen

- Bestimmung **unabhängiger externer Akteure**
- **Ein Prozess pro Akteur**
- Alle Module zuordnen, die zur Erzeugung der vom Akteur verlangten Systemreaktionen erforderlich sind
- Bei Redundanz **gemeinsame Leistungen in Dienstleistungsprozesse** verlagern
- **Weitere Prozesse** in besonderen Situationen:
 - Aufgaben unterschiedlicher Dauer und Dringlichkeit
 - Aufgabe für einen einzigen Prozess zu umfangreich
 - Fehlertoleranz gefordert

- **Kommunikationsbedürfnisse** resultieren aus der Verteilung der Module auf Prozesse
- Wahl der **Kommunikationsmittel**:
 - Passend zur Art des Kommunikationsbedürfnisses
 - Abgestimmt auf die Möglichkeiten der verwendeten Plattform

21.5 Ressourcenzuordnung

Zu treffende **Allokationsentscheidungen**

- Prozesse → Prozessoren
- Daten → Speicher
- Kommunikation → Medien, Kanäle

- **Module** möglichst als möglichst **als Ganzes** auf Prozesse verteilen
- Möglichst **schwach gekoppelte** Prozesse bilden
- **Prozesse** möglichst **als Ganzes** geographisch verteilen
- Entstehenden **Kommunikationsbedarf** berücksichtigen
- **Machbarkeit** prüfen (Leistung? Kommunikationsverbindungen?)

Kriterien

- ☆ **Leistungsanforderungen**: erfüllt
- ☆ **Datenverkehr**: Volumen akzeptabel
- ☆ Notwendige / entstehende **Redundanzen**: geprüft und entschieden
- ☆ **Flexibilität / Ausbaubarkeit**
- ☆ **Geheimnisprinzip**: gewährleistet
- ☆ **Konformität** zwischen logischer und physischer Struktur: wo möglich

21.6 Logische vs. physische Systemstruktur

Logische Systemstruktur: Module, Komponenten, Prozesse,...

Physische Systemstruktur:

- ☆ Gliederung der Software in Lieferungs- oder Verwaltungspakete
- ☆ Abhängig von
 - ◇ Entwicklungs- /Lieferstrategie
 - ◇ Möglichkeiten/Erfordernissen der verwendeten Programmiersprache
 - ◇ Einbindung vorhandener Software