

Multi-CPU Systeme

- Arten:
 - Multiprozessor: Ein System, ein Hauptspeicher, mehrere CPUs greifen darauf zu.
 - Multicomputer (Cluster): Eigenständige Rechner, eng gekoppelt über spezielle Verbindungen.
 - Verteiltes (Distributed) System: eigenständige Rechner, lose gekoppelt über ein Netzwerk.
- Begriffe:
 - Local: dieses System; Remote: alle anderen.
 - Server: Dienstanbieter; Client: Dienstanutzer.

Multiprozessor

- Auch Shared-Memory Multiprozessor.
- Problem: Streit um Speicherzugriff (Contention).
- Geschwindigkeit des Speicherzugriffs:
 - UMA: Unified Memory Architecture:
 - Bus; Bus + Cache; Crossbar (Kreuzverteiler, blockadefrei); Mehrstufiges Verteilnetzwerk (nicht blockadefrei).
 - NUMA: Non Unified Memory Architecture:
 - Ein Adressraum, schnellerer Zugriff zu lokalem Speicher.
 - CC-NUMA (Cache Coherent): hält Caches kohärent.

Multiprozessor OS (I)

- Asymmetrisch:
 - 1 Master CPU: führt OS aus.
 - x Slave CPUs: führen User-Prozesse aus.
- Symmetrisch (SMP):
 - Jede CPU führt gleichwertig verteiltes OS und User-Prozesse aus.
 - Ganzes OS oder Teile sind kritischer Bereich und brauchen Mutex(e) zur Synchronisation.
 - Programmierfehler können zu Deadlocks führen.

Multiprozessor OS (II)

- Synchronisation:
 - TSL (Test and Set Lock): Atomare Ausführung durch Sperre des Speicherbusses für die ganze Anweisung.
 - SW-Lösung von Peterson.
 - Aktives Warten (Spin Lock): Kann Speicherbus und andere CPUs bremsen.
- Scheduling:
 - (i) Welcher Prozess? (ii) Auf welcher CPU?
 - Affinity S.: Prozess versucht bei einer CPU zu bleiben.
 - Prozesse oft zusammengehörig (z.B. Threads).

Multicomputer

- Eigenständige Knoten mit CPU und Speicher (kein Shared-Memory); führen jeweils eine Instanz des gleichen OS aus.
- Gemeinsames Dateisystem für alle Knoten.
- Schnelles Verbindungsnetzwerk zwischen den Knoten.
- Verbindungstopologie:
 - Stern, Ring, Gitternetz, (doppelter) Torus, Würfel, Hyperwürfel.

Verbindungen

- Art:
 - Store-and-Forward Packet Switching: Daten-Pakete werden in jedem Knoten eingelesen und weitergeleitet; flexibel und effizient; zusätzliche Verzögerung durch Switching.
 - Circuit Switching: Weg muss vor dem Datentransfer aufgebaut werden.
- Primitiven:
 - Send, Receive: blockierend oder nicht blockierend.

Distributed Shared Memory

- Verteilter globaler virtueller Speicher.
- Jede Page ist in genau einem System.
- Pages werden nach Bedarf zwischen Systemen migriert. OS führt Paging aus dem Hauptspeicher eines anderen Systems durch.
- Performanceverbesserung durch Replikation von Read-Only Pages.
- Anfällig für „unechtes Teilen“.

Remote Procedure Call

- Vorgang:
 - Lokaler Prozeduraufruf wird an Stub geleitet.
 - Parameter werden verpackt und übers Netz gesendet.
 - Am Remote-System werden die Parameter entpackt und der Prozeduraufruf durchgeführt.
 - Ergebnis wird in gleicher Weise retourniert.
- Vorteil: Aufruf eines Dienstes ist transparent.
- Nachteil: Parameter marshalling (verpacken) schwierig oder unmöglich: globale Variablen.

Scheduling

- Ähnlich Multiprozessor Systemen, aber:
 - Prozessmigration ist sehr aufwändig und muss vermieden werden.
 - Es gibt keine globale Prozessliste.
 - Gleichmäßige Lastverteilung ist erwünscht (Load-Balancing).
- Verteilte Heuristik:
 - Sender-Initiiert: überlastete Systeme versuchen Prozesse zu verteilen.
 - Receiver-Initiiert: freie Systeme fordern Prozesse an.

Distributed Systems

Jedes beteiligte System:

- Ist „fully featured“ inklusive eigenem Dateisystem.
- Hat eigenes OS. (Kein einheitlicher Typ.)
- Hat eigene Administration.
- Kann irgendwo stehen. (Weltweit verteilt.)
- Ist lose verbunden über ein Netzwerk (LAN, WAN).
- Benötigt „Middleware“ um Unterschiede der Systeme auszugleichen.

Motivation

- Teilen von Ressourcen: z.B. Drucker, Dateien.
- Erhöhen der Verarbeitungsleistung: Load Sharing durch Parallelbetrieb.
- Verbessern der Zuverlässigkeit: Fail-Over auf redundante Systeme.
- Ermöglichen von Kommunikation: Austausch von Nachrichten.

Anwendungen

- Zugriff auf Remote-Computer: SSH, Telnet.
- Zugriff auf entfernte Information: Web, FTP.
- Persönliche Kommunikation: EMail, Chat.
- E-Commerce, Telemedizin, Fernlehre.

Beteiligtes OS:

- Netzwerk OS: Systeme sind einzeln sichtbar.
- Verteiltes OS: Systemgrenzen nicht erkennbar. Zugriff auf entfernte Ressourcen ist transparent.

Kommunikationsanforderungen

- Benennung von Ressourcen und Systemen und Namensauflösung (z.B. DNS).
- Wegefindung (Routing): fix, dynamisch.
- Paketierung von Nachrichten.
- Verbindungsstrategien:
 - Circuit Switching: permanente Verbindung.
 - Message Switching: Verbindung für eine Nachricht.
 - Packet Switching: jedes Paket reist individuell.
- Staubehandlung und Konfliktauflösung.

Middleware

Verteiltes System basiert auf:

- Dokumenten: Sammlung verlinkter Dokumente (Web).
- Dateisystem: Ein globales Dateisystem (NFS, AFS).
 - Up-/Download oder Remote Access.
 - Mit oder ohne gemeinsames Root-Verzeichnis.
 - Sequenzielle Konsistenz oder Session Semantik.
- Verteile Objekte: Sammlung verteilter Objekte und zugehöriger Methoden (Corba).
- Koordination: Einfügen und Entfernen von Tupeln im globalen Tupel-Raum.