SYBASE
Das Relationale Datenbankbetriebssystem für On-line-Anwendungen

Mit Beginn des Wintersemesters verfügt die Humboldt-Universität über eine Campus-Lizenz für das relationale Datenbankbetriebssystem SYBASE. Es kann im Rahmen der Campus-Lizenz in Lehre und Forschung eingesetzt werden.
SYBASE ist ein Produkt des im Jahre 1984 in Kalifornien gegründeten gleichnamigen Unternehmens. Die Gründer des Unternehmens, die über eine mehr als 20-jährige Erfahrung mit relationalen Datenbankbetriebssystemen verfügen, brachten eine Vielzahl fortschrittlicher Lösungsansätze in die Entwicklung des RDBMS ein. Im Ergebnis präsentierte SYBASE dem Markt eine Technologie, die sich richtungsweisend für die Zukunft relationaler Datenbankbetriebssysteme erweisen wird: eine offene Client-Server-Architektur.
Mit dieser Struktur können das Datenbankmanagement und die Transaktionssteuerung unabhängig von Client-Anwendungen und den Benutzerschnittstellen ausgeführt werden. Das Ergebnis ist ein erheblicher Gewinn an Produktivität in der Anwendungsentwicklung und der Zuverlässigkeit von Anwendungen. SYBASE Client und Server können so auf transparente Weise innerhalb eines Netzwerks von Mikrocomputern, Minicomputern und Großrechnern miteinander kommunizieren.

SYBASE besteht aus zwei Produktfamilien:

- Der SYBASE SQL-Server übernimmt das Datenmanagement und die Transaktionssteuerung unabhängig von den Client-Anwendungen und den Benutzerschnittstellen.

- Das SQL-Toolset umfaßt einen sehr benutzerfreundlichen Satz von Werkzeugen zur Entwicklung und zum Betrieb von On-line-Anwendungen.

Im Folgenden werden die beiden Produktfamilien in kurzer Form vorgestellt.

SQL-Server

Der programmierbare Server ist das Herzstück des RDBMS. Er verwaltet die Datenbank, schützt die Datenintegrität und sorgt für Datensicherheit. Er ist die entscheidende Komponente, die es gestattet, Datenmanagement und Transaktionsverarbeitung von den Funktionen der Anwendungen und Benutzerschnittstellen getrennt zu verwalten. Dies wird durch **Stored Procedures** und **Triggers** erreicht. Stored Procedures sind häufig benutzte Verarbeitungsroutinen, Transaktionen und parametrisierte Abfragen, die bereits vorkompiliert im Server abgelegt werden. Sie werden vom Client aufgerufen und auf dem Server abgearbeitet. Diese Arbeitsweise reduziert die Netzwerkbelastung erheblich, da die Ausführung mehrerer einzelner SQL-Anweisungen entfällt. Da Stored Procedures bereits vorkompiliert sind, erfolgt deren Abarbeitung wesentlich schneller als die Abarbeitung einzelner SQL-Anweisungen. Eine besondere Art von Stored Procedures, die Trigger, werden zur Wahrung der feld- und tabellenübergreifenden Integrität benutzt. Trigger sind Prozeduren, die bei bestimmten Feldbedingungen automatisch ablaufen und im Anschluß an eine Änderung Folgemodifikationen aktivieren. Damit wird die Integrität der Daten in der Datenbank sichergestellt. Stored Procedures und Trigger gestatten, unternehmensweite "business rules" und Transaktionen direkt im Server abzulegen und bei Bedarf allen Client-Anwendungen zur Verfügung zu stellen. Damit werden die Kosten für die Anwendungsentwicklung beträchtlich verringert. Ebenso reduziert sich der Wartungsaufwand in großem Umfang, da "business rules" und Transaktionen nur an einer Stelle, nämlich im Server, geändert werden müssen. Zusätzlich erhöht sich die Zuverlässigkeit der Datenbank dadurch, daß die Datenintegrität unabhängig von den Anwendungen implementiert wird. Für eine größtmögliche Sicherheit sind "rules" und Transaktionen ebenso wie die Datenbanktabellen gegen unberechtigten Zugriff geschützt.

Der SQL-Server basiert auf einer **Multi-Threaded-Architektur**, die ihren eigenen Systemkern und einen SQL-Task Manager umfaßt. Der Server ist so optimiert, daß er Mehrbenutzerfunktionen übernehmen kann, die traditionell vom Betriebssystem übernommen werden, etwa "scheduling", "task switching", "disk caching" oder "locking". Somit ist der SQL-Server ein Datenbankbetriebssystem, das oberhalb des eigentlichen Betriebssystems läuft. Dadurch wird nahezu die gesamte ineffiziente Systemlast des Betriebssystems, die andere RDBMS in Betrieb und Performance stark beschränkt, beseitigt. Darüber hinaus ist der SQL-Server in der Lage, innerhalb von nur einem MByte zusätzlichem Hauptspeicher dreißig Benutzer zu verwalten, was die Anforderungen an die Hardware drastisch verringert. Dadurch wird es möglich, einen größeren Cache-Bereich zu konfigurieren, um die Zahl der Plattenzugriffe zu minimieren.

Die Arbeit mit Mehrprozessorsystemen wird durch die "**Virtual Server**" Architektur von SYBASE voll unterstützt. Dies wird durch dynamisches und transparentes "load balancing" auf der Ebene von SQL-Anweisungen erreicht, wodurch nur ein SQL-Server Prozeß auf einem Prozessor läuft. Ein im System integrierter SQL-Control-Monitor erlaubt eine einfache, effiziente und verläßliche Ablaufsteuerung aller Anwendungen.

Der SQL-Server läßt sich in zwei verschiedenen Konfigurationen betreiben. Im sogenannten **Host-Modus** befinden sich Client und Server auf der gleichen Maschine. Im **Netz-Modus** befinden sich Client und Server auf verschiedenen Maschinen.

Der SQL-Server sichert eine hohe Verfügbarkeit der Daten und Anwendungen während der Systemwartung und bei Hardwareausfällen. Er erlaubt die Abwicklung von **Wartungsarbeiten im On-line-Betrieb** durch den Einsatz der vom System zur Verfügung gestellten Stored Procedures, während der Betrieb der Anwendungen weiterläuft. Diese in Transact-SQL geschriebenen Stored Procedures führen sämtliche Wartungsarbeiten als parallele SQL-Transaktionen aus, ohne die Steuerung für derartige Aufgaben an das Betriebssystem abzugeben.

Für den Fall eines Systemausfalls (Stromausfall, Netzwerkzusammenbruch) stellt der SQL-Server durch das **"write-ahead"-Transaktions-Logging** sowie ein **physisches Logging** eine schnelle Wiedergewinnung der Daten sicher. Durch die "Write-Ahead" Log-Daten können die Daten nach Systemzusammenbrüchen vollständig wiederhergestellt werden. Hierzu wird jede Transaktion in der Log-Datei verzeichnet, bevor sie auf die Speicherplatten geschrieben wird. So können alle Änderungen in der Datenbank wiederholt nachgefahren werden. Physisches Logging bedeutet, daß Offset-Position und "byte stream" protokolliert werden, nicht aber der logische Satz. Damit wird die Wiederherstellung der Daten beschleunigt. Darüber hinaus ist der SQL-Server in der Lage, auftretende Hardwarefehler durch ein **maschinenunabhängiges Plattenspiegelungsverfahren** und den durch das RDBMS kontrollierten **dynamischen Prozessorwechsel bei Totalausfall eines Rechners** im Netz abzufangen.

Durch **Open Server** können SYBASE-fremde Datenbanksysteme über die Client/Server-Schnittstellen in eine SYBASE-Umgebung integriert werden. Aus der Sicht der Anwendung wirken sie dann wie ein SQL-Server von SYBASE. Damit können bereits existierende Datenbanksysteme weiter genutzt werden. Andererseits können SYBASE-fremde Werkzeuge und Anwendungsprogramme durch **Open Client** über die gleiche Schnittstelle auf SYBASE-Datenbanken zugreifen.

SYBASE ist ein relationales Datenbankbetriebssystem, welches verteilte Anwendungen und Datenbanken in heterogenen Netzwerken erlaubt. Die Verteilung wird dabei in doppelter Hinsicht realisiert. Der verteilte Zugriff ermöglicht es verschiedenen Anwendungen auf unterschiedlichen Rechnern, mit dem SQL-Server zu kommunizieren. Das Konzept der verteilten Datenbanken ist dadurch gekennzeichnet, daß eine Datenbank auf mehrere Rechner im Netzwerk verteilt ist.

Eine Transaktion ist die grundlegende Arbeitseinheit, die der SQL-Server durchführt. Sie besteht aus mehreren SQL-Anweisungen, die im Server gespeicherte Informationen ändern und lesen. Bei der Arbeit in einer verteilten Umgebung kann eine einzige Transaktion Daten betreffen, die über mehrere SQL-Server verteilt sind. Verteilte Änderungen sind komplexer als verteilte Abfragen, weil eine Änderung die Datenkonsistenz auf verschiedenen Maschinen betreffen kann. Besonders kritisch gestalten sich durch einen Systemzusammenbruch unterbrochene verteilte Transaktionen, da die Datenkonsistenz nicht sichergestellt werden kann. Der SQL-Server unterstützt zwei Funktionen, um die Konsistenz der Daten bei verteilter Umgebung sicher zu stellen: **Two-Phase-Commit** und **Server-Server-Kommunikation über Remote Procedure Calls (RPCs).**

SQL-Toolset

Mit der **Client-Software,** dem **SQL-Toolset**, bietet SYBASE einen umfassenden und klar aufgebauten Satz von leistungsfähigen Werkzeugen zur Entwicklung von On-Line Anwendungen auf Workstations und/oder terminal-orientierten Minicomputern. Über eine angebotene offene Schnittstelle lassen sich darüber hinaus weitere Tools anderer Hersteller integrieren. Mit dem SQL-Toolset wird dem Anwender eine benutzerfreundliche **Window-Oberfläche** bereitgestellt, die mit leistungsstarken graphischen und objektorientierten Werkzeugen zur Erstellung von Anwendungen in Window-Technik ausgerüstet ist.
Das SQL-Toolset setzt sich aus folgenden Bestandteilen[2] zusammen:

Die **APT Workbench** besteht aus einem Paket von Application Productivity Tools (APT) zum Entwurf, Prototyping, Erstellen und zur Wartung formatgestützter Anwendungen.

APT- Edit
Ein Editor zum Entwurf und der Implementierung von Formaten und Menüs mit einer Window-Oberfläche.

[2] Nach "SYBASE First in On-Line Applications - SQL-Toolset" Broschüre der Firma SYBASE

APT-Build
Ein automatischer Code-Generator, der APT-SQL Code produziert.

APT-SQL
Eine mächtige Sprache der vierten Generation, die zur Definition von Verarbeitungs- und Darstellungslogik für Formate und ihre Objekte eingesetzt wird.

APT-Execute
Ein vollständiges Runtime-Modul für Anwendungen, die mit der APT Workbench entwickelt worden sind.

Die **Data Workbench** besteht aus einer Gruppe von benutzerfreundlichen Werkzeugen für Decision Support (Entscheidungsunterstützung) und DB-Administration. Die Data Workbench umfaßt:

VQL
Eine endbenutzerorientierte Abfragesprache (Visual Query Language). Sie unterstützt den Anwender in der Konstruktion komplexer SQL-Abfragen, ohne daß er SQL-Kenntnisse besitzen muß.

Interactive SQL
Ein Hilfsprogramm, mit dem Abfragen in ANSI-Standard SQL ausgeführt sowie Stored Procedures in Transact-SQL definiert werden können.

Report Workbench
Ein mächtiges Werkzeug zum Definieren, Ausführen und Speichern von Reports.

Data Entry
Ein Hilfsprogramm zur Auswahl und Eingabe von Daten, wobei ein automatisch erzeugter Standardbildschirm verwendet wird.

Utilities
Werkzeuge für die normale DB-Administration, den Zugang zum Data Dictionary und das Laden großer Datenmengen.

Client/Server Schnittstellen bieten SYBASE-fremden Produkten eine standardisierte Schnittstelle zu den SYBASE-Produkten. Im besonderen sind dies:

DB-Library
Eine Anwendungsprogrammier-Schnittstelle, die SYBASE-fremden Programmierwerkzeugen und Anwendungssystemen erlaubt, mit dem SQL-Server zu kommunizieren.

Open Server
Eine Anwendungsprogrammier-Schnittstelle, die den Zugang zu SYBASE-fremden Datenbasen und Anwendungen bereitstellt.

SYBASE am Rechenzentrum

Zur Unterstützung der Benutzerverwaltung wurde von der Firma SYBASE eine Verwaltungslizenz für eine SUN-Workstation erworben. Sie gestattet dem Rechenzentrum, das leistungsfähige Produkt zur Erstellung eines Verwaltungsprojekts zu nutzen und erste Erfahrungen bei der praktischen Arbeit mit SYBASE zu sammeln.

Michael Rybczak