

JDBC Treiber für DB2/400

Für die iSeries (besser bekannt als AS/400) gibt es zwei verschiedene Implementierungen von JDBC Treibern. Einer dieser Treiber ist unter dem Namen Toolboxtreiber bekannt, der andere wird meist als native Treiber bezeichnet. Zu den verschiedenen Treibern und Versionen sind einige Legenden im Umlauf, die auch von nicht Java Programmierern weiterverbreitet werden.

Grundsätzlich gilt, dass mit Java ähnliche Zugriffszeiten zu erreichen sind, wie mit SQL in anderen Programmiersprachen. Das bedeutet, dass Einzelsatzoperationen im Bereich von Millisekunden liegen müssen. Dauern Datenbankoperationen Sekunden, dann liegt das nicht am Treiber. Erster Kandidat ist die Datenbank, das heißt es fehlen Zugriffspfade; zweiter Kandidat ist die Applikation; wenn 1000 Sätze in einen Vektor gelesen werden, um eine Combo Box zu füllen, dann ist der Sekundenbereich erreicht. Sowohl fehlende Zugriffspfade als auch die Anzahl der Datenbankoperationen lassen sich einfach und schnell mit dem Datenbank Monitor ermitteln.

Der native Treiber

Der native Treiber wird mit OS/400 mitgeliefert und gehört zu den Typ 2 Treibern. Das bedeutet, dass dieser Treiber nur unter OS/400 ausführbar ist, da er zumindest zum Teil in C implementiert ist. Gedacht ist dieser Treiber für Zugriffe auf die lokale Datenbank. Updates für diesen Treiber sind nur über den PTF Mechanismus möglich. Der Treiber ist enthalten in dem Archiv db2_classes.jar, das nur die JDBC Komponenten enthält und statisch compiliert ist.

Der Toolboxtreiber

Versionen des Toolboxtreibers:

Version	Ausgeliefert mit	Liz. Prod.	Läuft ab	Connect ab
Mod 0	V4R2	5763-JC1 V3R2M0	V3R2	V3R2
Mod 1	V4R3	5763-JC1 V3R2M1	V3R2	V3R2
Mod 2	V4R4	5769-JC1 V4R2M0	V4R2	V4R2
Mod 3	V4R5	5769-JC1 V4R5M0	V4R3	V4R2
Mod 4	V5R1	5722-JC1 V5R1M0	V4R4	V4R3
Mod 5	V5R3	5722-JC1 V5R2M0	V4R5	V4R5

Derselbe Treiber ist in mehreren Archiven enthalten:

Archiv	Inhalt	precompile	Anmerkung
jt400.jar	access, vaccess	nein	früher nicht komprimiert
jt400.zip	access, vaccess	nein	
jt400Access.zip	access	ja	seit V4R5
jt400ntv.jar	access	nein	seit V4R3 native Optimierung
jt400Native.jar	access	ja	seit V5R1 native Optimierung

Alle Archive mit dem Toolboxtreiber enthalten die kompletten Klassen der Toolbox, zum Teil auch doppelt in dem Paket vaccess, das von Visual Age benutzt wurde. Die Archive jt400ntv.jar und jt400Native.jar enthalten einen kleinen Teil an AS/400 spezifischen C Implementierungen, alle anderen sind pure Java und der JDBC Anteil gehört damit zu den Typ 4 Treibern.

Der Toolboxtreiber wird mit OS400 mitgeliefert und wird über PTF Mechanismus gepflegt, ist aber auch als jtOpen im Internet verfügbar. Die Open Source Variante ist umfassender als die Lizenz Variante und immer auf dem neueren Stand.

Treiber Auswahl

Die weit überwiegende Mehrzahl der Java Anwendungen, die DB2/400 als Datenbank benutzen laufen auf anderen Systemen, seien dies dann Linux, Wintel, AIX oder Sonstige. Selbst bei reinen iSeries Installationen ist der Datenbankserver von anderen Server-Diensten getrennt. Daraus resultiert, dass der Typ 4 Treiber am weitaus meisten benutzt wird, dann folgen mit weitem Abstand der Toolbox Treiber mit den native Optimierungen und der native Treiber (Typ2). Das hat zur Folge, dass der Support für den reinen Typ 4 Treiber am Besten und schnellsten und dieser Treiber der stabilste ist. Bei starker Gewichtung von Stabilität und Pflege ist in jedem fall zum Toolboxtreiber zu raten, mit Tendenz zur Open Source Variante.

Wie sieht es nun mit der Performance aus? Ein Stück weit wie bei anderen Treiber Mechanismen auch, bei nicht klärbaren Leistungsdefiziten macht es immer Sinn, einen Treiberdefekt auszuschließen und eine andere Treiberversion oder einen anderen Treiber auszuprobieren, zumal das eine einfache Maßnahme ist, die keinen großen Aufwand verursacht.

Immer dann, wenn die Java Anwendung auf einer AS400 läuft, ist es zu empfehlen eine der statisch kompilierten Varianten zu benutzen, eine Eigenkompilierung mit CRTJVAPGM scheidet dabei meist praktisch aus, wegen Laufzeit und Instabilität. Der Wechsel zur kompilierten Variante ist meist direkt spürbar. Ob man dann die optimierte, die native oder die Typ 4 Variante benutzt sollte man dann messtechnisch untermauern; die Unterschiede sind im praktischen Versuch oft gering. Bei diesen Konstellationen scheidet die jtOpen Variante dann aus.

Häufig werden Empfehlungen abgegeben den Zugriff auf die lokale Datenbank zu favorisieren, was aber für die Skalierbarkeit eher negativ ist. Von dieser Variante ist eher abzuraten, wenn auf dem Datenbankserver zusätzlich auch 5250 Workload vorhanden ist. Für die Trennung von Serverdiensten spricht auch, dass Hardware dann gezielter eingesetzt werden kann, bei Web Diensten erhöht sich zudem die Verletzbarkeit bei Single-Box Lösungen.

Aus Performance Gesichtspunkten kann bei Trennung von Datenbank- und Applikations-Server die Netzwerkbelastung, bei ausreichend dimensionierbarer Bandbreite, vernachlässigt werden. Skalierbarkeit und Geschwindigkeit nehmen mit der Verteilung von Serverdiensten eher zu.

Generell gilt für JDBC Performance immer, dass messen raten schlägt. Das wichtigste Instrument dafür ist für DB2/400 der Database Monitor, der alle Verarbeitungszeiten in der Datenbank bis auf Einzelzugriffs Ebene misst und protokolliert.