Onderzoeksrapport

“Database Cubrid onderzoeken”

Door: Luuk de Jong, Niek Hamoen, Evan van Essen, Brighton van den End

Datum: 27-05-2018

Inhoudsopgave  
1. Inleiding  
2. Oracle  
3. Cubrid  
4. Verschillen  
5. Query’s  
6. Installatie  
7. Onderhoud van Cubrid  
8. Bronnen

Inleiding

In dit verslag gaan we het hebben over hoe we Cubrid hebben gebruikt en hoe we de connectie met Java hebben gelegd enzovoort. We gaan het ook over de verschillen hebben tussen Oracle en Cubrid en welke wij zouden aanraden voor beginners zoals wij. Wij hebben ook een PoC gemaakt om te laten zien dat wij met een totaal nieuwe database om kunnen gaan.

Oracle

Oracle database is een relationele database management systeem (RDBMS) ontwikkeld door Oracle Corporation, hoofdvestiging in California met wereldwijd ruim 130.000 werknemers.

De server draait op veel verschillende besturingssysteem, waaronder Linux, Mac OS X en Windows. Oracle is een van de meest gebruikte database managers en wordt veel gebruikt door bedrijven. Veel banken maken bijvoorbeeld gebruik van Oracle.

Oracle staat bekend om zijn betrouwbaarheid. Oracle levert uitstekende prestaties, ook wanneer deze onder druk staat door zwaar belastende taken. Oracle bevat ‘flashback-technologie’, dit zorgt ervoor dat er geen gegevens verloren gaan of verwijderd worden, wanneer de server niet werkt, of een applicatie problemen heeft.

Oracle heeft de mogelijkheid om verschillende transacties in dezelfde batch te groeperen. Hiermee maakt Oracle een groot onderscheid van zijn concurrenten die elke transactie afzonderlijk uitvoeren. Dit zorgt voor een efficiëntere werking.

Hoewel Oracle DataBase zelf vrij duur is investeert Oracle Corporation in andere software, om kosten voor developers te verlagen. Zo heeft het bedrijf MySQL ontwikkelt, een open source RDBMS.

Hieronder staat een lijst van software waar Oracle Corporation onder andere in investeert.

* Berkeley DB
* Eclipse
* GlassFish
* InnoDB
* Java
* JAva ME
* Linux
* NetBeans
* OpenJDK
* PHP
* VirtualBox
* Xen

Cubrid

Cubrid is een open source op SQL gebaseerde relationele database management systeem. (RDBMS). Cubrid is ontwikkeld door Naver Corporation, hoofdvestiging in Zuid-Korea voor web applicaties. Cubrid gebruikt ook object extensies. Dit houdt dat je met hele objecten van Java zoals bijvoorbeeld een reiziger of product in Cubrid kan toevoegen.

Het gene wat Cubrid onderscheidt van de andere relationele databases is het feit dat het een 3 lagen systeem heeft. Deze 3 lagen bestaan uit de database-server, de verbinding regelaar en de applicatielaag. De database-server zorgt ervoor dat je de host kan selecteren waar je de database runt. Dit zorgt ervoor dat je eigenlijk oneindig veel storage hebt. op de database-server kunnen wel meerdere databases staan. De verbinding regelaar zorgt zoals de naam impliceert voor de verbinding tussen de database en de persoon die de database aanstuurt, aangezien de database en de aanstuurder niet perse op het zelfde machine zitten. In de applicatielaag werkt de aanstuurder en wordt geholpen door de verbindings regelaar.

Verschillen

|  |  |  |
| --- | --- | --- |
|  | **ORACLE** | **CUBRID** |
| **Max database size** | 8 EB | 2 EB |
| **Security** | ACID | ACID, MVCC |
| **Kosten** | Gratis versie  Oracle DataBase XE | Open source Volledig gratis |

Naast uiterlijk en structuur van beide software zijn er weinig opmerkelijke verschillen. Het grootste verschil is dat Cubrid open source is en volledig gratis.

Oracle DataBase XE is een gratis instap versie, die vooral gebruikt wordt door studenten, beginnende developers en kleine bedrijven

Query’s

Hier gaan we de query’s van Oracle en Cubrid vergelijken. Laten we beginnen met het command wat je waarschijnlijk als eerste uitvoert bij het maken van een nieuwe database.

De create table command. In oracle begin ziet deze command er zo uit:

**CREATE TABLE Orders(**

**OrderID int NOT NULL,**

**OrderNumber int NOT NULL**

**PersonID int,**

**PRIMARY KEY (OrderID)**

**FOREIGN KEY (PersonID) REFERENCES Persons(PersonID)**

**);**

In cubrid ziet hij er zo uit:

**CREATE TABLE Orders(**

**OrderID INT NOT NULL PRIMARY KEY,**

**OrderNumber INT NOT NULL,**

**OrderDate DATE NOT NULL,**

**reizigerid INT NOT NULL,**

**CONSTRAINT FK\_REIZIGERID FOREIGN KEY (reizigerID)**

**REFERENCES Reiziger(reizigerID)**

**);**

Het grootste verschil is dat je de primary key bij cubrid bij de constraints zet. In Oracle zet je hem echter aan het einde van je command. Het verschil bij de foreign keys is dat je in cubrid voor de foreign key constraint zet en een constraint naam. Grotendeels is het in Cubrid zoals in Oracle.

Over naar alter table. Met alter table kun je meerdere dingen onder andere: het datatype aanpassen, een nieuwe column toevoegen of een constraint toevoegen. Alle 3 deze dingen worden in Cubrid en Oracle hetzelfde gedaan. Namelijk zo:

**ALTER TABLE table\_name**

**MODIFY column)name datatype;**

---------------------------------------------------------------------------------------------------------------------------

**ALTER TABLE table\_name**

**ADD column\_name datatype;**

---------------------------------------------------------------------------------------------------------------------------

**ALTER TABLE table\_name**

**ADD CONSTRAINT constraint\_name constraint\_type (column\_name);**

Insert into doen ze ook allebei op dezelfde manier namelijk zo:

**INSERT INTO table\_name (column\_name(s))**

**VALUES (value(s));**

De select queries zijn ook bij allebei hetzelfde:

**SELECT column\_name(s)**

**FROM table\_name**

**WHERE condition;**

Conclusie, beide query’s maken grotendeels gebruik van dezelfde opbouw van statements met hier en daar een kleine afwijking. Dit zorgde er wel voor dat het werken met Cubrid niet zo’n probleem was. De ov-casus was er dan ook makkelijk in te voegen. Alleen de primary keys moesten aangepast worden

Installatie

Als u Cubrid wilt gebruiken voor eigen gebruik moet je deze stappen volgen om Cubrid te kunnen gebruiken.

**Stap 1: Download de Cubrid engine en manager**

Download de Cubrid Engine en Cubrid Manager.

<https://www.cubrid.org/downloads/os-select/64-bit/engine>

<https://www.cubrid.org/downloads/os-select/64-bit/tools/manager>

**Stap 2: Starten**

Je kan nu de Cubrid manager starten.

**Let op dat je de Cubrid Workspace gebruikt die automatisch wordt aangesteld bij het opstarten.**

**Stap 3: Modes**

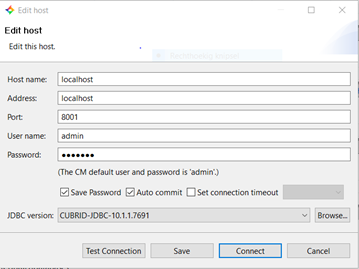
Je hebt 2 modes in Cubrid manager mode en query mode. Bij de eerste opstart wil je beginnen met de manager mode daar kan je nieuwe connecties maken. De connectie “Localhost” zou er moeten staan.

**Stap 4: Bestaande User**

We gaan in de al bestaande localhost inloggen en een nieuw host aan te maken moet je een server hebben en dat is ons niet gelukt. Je klikt op localhost en dan krijg je een inlog scherm zoals dit:

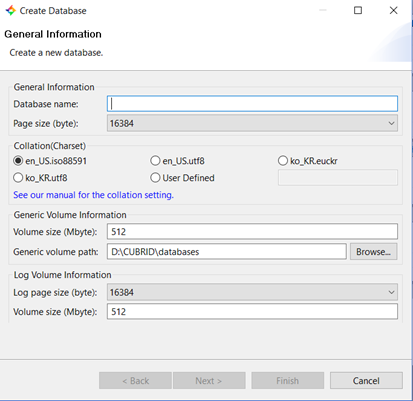
Zorg ervoor dat de JDBC versie de nieuwste is die kan je hier downloaden:

<https://www.cubrid.org/downloads/windows/64-bit/drivers/jdbc>



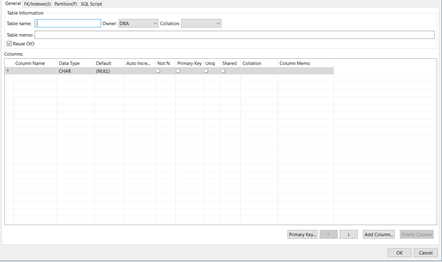
**Stap 5: Nieuwe database**

We gaan een nieuwe database aanmaken. Rechter muisknop op de databases van localhost. Je klikt dan op “create database”. Maak een wachtwoord aan dat je niet makkelijk vergeet. Volg de stappen die er zijn en daarna is je database aangemaakt.



**Stap 6: Vul je database met data.**

Als het goed is heb je de create OV-database al en vul OV database al gedownload voor data processing zelf. De create database moet je overnemen door op tabellen te klikken en daarna “Create Table”. Hier kan je de naam, data types en primary/foreign keys aangeven.



**Stap 7: Maak query’s**

Je kan nu Cubrid gebruiken en informatie er in opslaan

Onderhoud van Cubrid

Cubrid heeft is niet de eerste Database waar je aan denkt als je denkt aan databases. Een beginnende programmeur denkt eerder aan Oracle, Mysql en PostGreSql.

De eerste stabiele versie van Cubrid kwam uit in Oktober 2008 en een maand later werd het een open source project zodat iedereen erbij kan. Tot 2010 waren er nog allemaal kleine updates die de performance verbeterde van Cubrid. Daarna in Oktober 2010 was er een update waardoor er betere SQL statements geschreven kon worden met speciale extensies. Aan het einde van 2010 was er nog een belangrijke update waardoor je met JBCD Drivers met Java de persistentie kon regelen. Tot en met 2013 waren en toen nog updates waar kleine dingen werden toegevoegd maar er meer werd gefocust op de performance van Cubrid zodat de database snel alle transacties kon afhandelen. In September 2013 was er een update waardoor er extra SQL support kwam. Dit betekent dat er betere error handling bij kwam.

De laatste update 10.1 was in Juli 2017 hier werden er nog meer SQL extensies toegevoegd zodat het meer gebruiksvriendelijker werd. Juli 2017 is al bijna een jaar geleden en er is nog geen nieuwe informatie over een nieuwe update die er aan komt.

# 

# 

# 

# 

# 

# 

# 

# **Literatuurlijst**

Data Integrity [pdf]. (2017, mei). Geraadpleegd op 28 mei 2018, van <https://community.oracle.com/servlet/JiveServlet/previewBody/1014908-102-1-157429/ZFS%20-%20Data%20Integrity.pdf>

Reddy, P. (z.d.). The Advantage Of Oracle Database. Geraadpleegd op 28 mei 2018, van <https://www.techwalla.com/articles/the-advantages-of-oracle-databases>

Oracle and Open Source[pdf] (2011, juni). Geraadpleegd op 28 mei 2018, van

[www.oracle.com/us/024981.pdf](http://www.oracle.com/us/024981.pdf)

Cunningham, L. (2006, 1 maart). Oracle Database 10g Express Edition: Not Just for Learners. Geraadpleegd op 28 mei 2018, van <http://www.oracle.com/technetwork/articles/sql/cunningham-database-xe-087516.html>

[https://www.cubrid.org/](https://www.cubrid.org/downloads/os-select/64-bit/engine)

Oracle Tutorial. (z.d.). Geraadpleegd op 28 mei 2018, van <https://www.techonthenet.com/oracle/>

Introduction to Manual. (z.d.). Geraadpleegd op 28 mei 2018 van <https://www.cubrid.org/manual/en/10.0/toc.html>

[https://en.wikipedia.org/](https://en.wikipedia.org/wiki/CUBRID)

<https://www.w3schools.com>