

# Evaluation - Database

## Projekt: DigitalSchoolNotes

**Projekt Team: Adler, Brinnich, Hohenwarter, Karic, Stedronsky**

**Version 3.0**

**15.12.2015**

**Status: [RELEASE]**

	Datum	Name	Unterschrift
Erstellt	29.09.2015	Niklas Hohenwarter	
Geprüft	15.12.2015	Thomas Stedronsky	
Freigegeben			
<b>Git-Pfad:</b> <i>/doc/evaluation</i> <b>Dokument:</b> <i>evaluation_db.doc</i>			

# 1 Changelog

Version	Datum	Status	Bearbeiter	Kommentar
0.1	2015-09-29	Erstellt	Selina Brinnich	Dokument erstellt
1.0	2015-10-07	Geprüft	Niklas Hohenwarter	QA
1.1	2015-10-14	Bearbeitet	Philipp Adler	Formatierung
2.0	2015-10-14	Geprüft	Thomas Stedronsky	QA
2.1	2015-12-15	Bearbeitet	Philipp Adler	Codeverzeichnis hinzugefügt
3.0	2015-12-15	Geprüft	Niklas Hohenwarter	QA

## 2 Vergleich

	MySQL	PostgreSQL	MongoDB	Couchbase
<b>Dokumentation</b>	8/10 Sehr ausführlich, etwas unübersichtlich	7/10 Teilweise zu wenig ausführlich, unübersichtlich	9/10 Sehr ausführlich mit Beispielen	10/10 Sehr ausführlich mit Screenshots u. Beispielen
<b>Lizenz</b>	0/10 2000\$ pro Jahr pro Server	10/10 Kostenfrei & kommerziell nutzbar	10/10 Kostenfrei & kommerziell nutzbar	10/10 Kostenfrei & kommerziell nutzbar
<b>Installation / Konfiguration</b>	9/10 Einfach, keine Probleme	6/10 Einige Probleme	10/10 Einfach, keine Konfiguration notwendig	5/10 Einfache Installation, sehr hohe Systeman- forderungen
<b>Community</b>	10/10 428.788 Stackoverflow Questions	8/10 53.342 Stackoverflow Questions	9/10 59.672 Stackoverflow Questions	7/10 1.734 Stackoverflow Questions
<b>Benutzerfreund- lichkeit</b>	8/10 Nur CLI, eindeutige Fehlermeldungen, Hot Swap nicht nativ (HA Proxy)	6/10 Nur CLI, Fehlermeldungen nicht sofort hilfreich	8/10 Nur CLI, eindeutige Fehlermeldungen	10/10 CLI+GUI
<b>Skalierbarkeit</b>	5/10 Cluster und Replikation, Hot Swap nicht nativ (HA Proxy)	5/10 Cluster und Replikation, Hot Swap nicht nativ (HA Proxy)	10/10 Cluster und Replikation, Hot Swap nativ	10/10 Cluster und Replikation, Hot Swap nativ
<b>Prototyp</b>	10/10 Funktionierte sofort, SQL	10/10 Funktionierte sofort, SQL	10/10 Funktionierte sofort, SQL	5/10 Sehr kompliziert, schwierig zu verstehen
<b>Summe</b>	53/70	57/70	66/70	57/70

## 3 MySQL

### 3.1 Installation

Die Installation läuft wie folgt ab[1]:

```
sudo apt-get install mysql-server mysql-client
sudo mysqladmin -u root -h localhost password 'mypassword'
sudo apt-get install php5-mysql
```

### 3.2 Konfiguration

Erstellen eines Users und setzen der Rechte[2]:

```
mysql -u root -p
CREATE USER 'eval'@'localhost' IDENTIFIED BY '1234';
GRANT ALL PRIVILEGES ON eval.* TO 'eval'@'localhost';
CREATE DATABASE eval;
FLUSH PRIVILEGES;
mysql -u eval -p
```

### 3.3 CRUD

Create:

```
CREATE TABLE `test` (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
  `name` varchar(255) default NULL,
  PRIMARY KEY (`id`)
) AUTO_INCREMENT=1;
```

**Code 1: MySQL Create**

Insert:

```
INSERT INTO `test` (`name`) VALUES ("Xandra Mcknight");
```

Delete:

```
DELETE FROM test WHERE name='Xandra Mcknight';
```

Update:

```
UPDATE test SET name='Peter Test' WHERE name='Xandra Mcknight';
```

## 4 PostgreSQL

### 4.1 Installation

Die Installation läuft wie folgt ab[3]:

```
apt-get install postgresql postgresql-client
```

### 4.2 Konfiguration

Erstellen eines Users und setzen der Rechte[3]:

```
su postgres
psql
CREATE USER mypguser WITH PASSWORD 'mypguserpass';
CREATE DATABASE mypgdatabase OWNER mypguser;
\q
vim /etc/postgresql/X.Y/main/pg_hba.conf
Change: local      all      all      peer
To: local  all      all      md5
/etc/init.d/postgresql reload
psql -d mypgdatabase -U mypguser
```

### 4.3 CRUD

Create:

```
CREATE TABLE "test" (
  id SERIAL PRIMARY KEY,
  name varchar(255) default NULL
);
```

**Code 2: PostgreSQL Create**

Insert:

```
INSERT INTO "test" (name) VALUES ('Quintessa Cantu');
```

Delete:

```
DELETE FROM test WHERE name='Quintessa Cantu';
```

Update:

```
UPDATE test SET name='Peter Test' WHERE name='Quintessa Cantu';
```

## 5 MongoDB

### 5.1 Installation

Die Installation läuft wie folgt ab[4]:

```
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv 7F0CEB10
echo "deb http://repo.mongodb.org/apt/debian wheezy/mongodb-org/3.0 main" |
sudo tee /etc/apt/sources.list.d/mongodb-org-3.0.list
sudo apt-get update
sudo apt-get install -y mongodb-org
sudo service mongod start
```

### 5.2 CRUD

Für einen genaueren Überblick siehe *doc/evaluation/prototype\_django\_mongodb\_20151007.pdf*

Insert[5]:

```
db.inventory.insert(
    {
        item: "ABC1",
        details: {
            model: "14Q3",
            manufacturer: "XYZ Company"
        },
        stock: [ { size: "S", qty: 25 }, { size: "M", qty: 50 } ],
        category: "clothing"
    }
)
```

**Code 3: MongoDB Insert**

Update[5]:

```
db.inventory.update(
    { item: "ABC1" },
    {
        $set: {
            category: "apparel",
            details: { model: "14Q3", manufacturer: "XYZ Company" }
        },
        $currentDate: { lastModified: true }
    }
)
```

**Code 4: MongoDB Update**

Read[5]:

```
db.inventory.find()
```

Delete[5]:

```
db.inventory.remove({})
```

## 6 Couchbase

### 6.1 Installation

Die Installation läuft wie folgt ab[6]:

```
wget http://packages.couchbase.com/releases/4.0.0-rc0/couchbase-server-community_4.0.0-rc0-ubuntu14.04_amd64.deb
dpkg -i couchbase-server-community_4.0.0-rc0-ubuntu14.04_amd64.deb
```

### 6.2 Konfiguration

CLI Programm in die Bashrc hinzufügen:

```
if [ -d "/opt/couchbase/bin" ] ; then
    export PATH="/opt/couchbase/bin:$PATH"
fi
```

**Code 5: Bashrc Eintrag**

### 6.3 CRUD

Mit den Beispieldaten vom Beersample

**Insert[7]:**

```
cbtransfer CSV_FILE http://IP:PORT -B TARGET_BUCKET -u USERNAME -p
PASSWORD
```

**Update[7]:**

```
cbtransfer FILE http://IP:PORT -B TARGET_BUCKET --destination-
operation=add/set -u USERNAME -p PASSWORD
```

**Read[7]:**

```
cbtransfer http://IP:PORT csv:FILE -b SOURCE_BUCKET -u USERNAME -p
PASSWORD
```

**Delete[7]:**

```
couchbase-cli bucket-delete -c IP:PORT --bucket=NAME -u USERNAME -p
PASSWORD
```

## 7 Endergebnis

Aufgrund der Ergebnisse der Evaluierungstabelle sowie den einzelnen Prototypen der verglichenen Produkte entscheidet das Team sich dazu **MongoDB** zu verwenden.

## 8 Codeverzeichnis

Code 1: MySQL Create.....	4
Code 2: PostgreSQL Create .....	5
Code 3: MongoDB Insert.....	6
Code 4: MongoDB Update.....	6
Code 5: Bashrc Eintrag.....	7

## 9 Quellen

- [1] Ariejan de Vroom, "Howto install MySQL on Ubuntu/Debian",  
<https://ariejan.net/2007/12/12/how-to-install-mysql-on-ubuntudebian/>, zuletzt besucht: 30.09.2015
- [2] Etel Sverdlov, "HowTo Create a New User and Grant Permissions in MySQL",  
<https://www.digitalocean.com/community/tutorials/how-to-create-a-new-user-and-grant-permissions-in-mysql>, zuletzt besucht: 30.09.2015
- [3] Debian Wiki, "PostgreSQL",  
<https://wiki.debian.org/PostgreSQL>, zuletzt besucht: 30.09.2015
- [4] MongoDB Docs, "Install MongoDB on Debian"¶,  
<http://docs.mongodb.org/manual/tutorial/install-mongodb-on-debian/>, zuletzt besucht: 30.09.2015
- [5] MongoDB Docs, "MongoDB CRUD Tutorials"¶,  
<http://docs.mongodb.org/manual/applications/crud/>, zuletzt besucht: 30.09.2015
- [6] Couchbase Docs, "Ubuntu/Debian installation",  
<http://docs.couchbase.com/admin/admin/Install/Ubuntu-install.html>, zuletzt besucht: 30.09.2015
- [7] Couchbase Docs, "CLI overview",  
<http://docs.couchbase.com/admin/admin/CLI/cli-overview.html>, zuletzt besucht: 01.10.2015