



dblint A Tool for Automated Analysis of Database Design

Benjamin Krogh & Andreas Weisberg

Aalborg Universitet

Center for Data-intensive Systems

DAPUG 2012

Indhold

- Om os
- Motivation
- Hvad er DBLint
 - Regler og score
 - System arkitektur og databasemodel
- Hvad kan DBLint
 - Tilpasning og udvidbarhed
- Demo



Om os

- Benjamin Krogh
 - M.Sc. Software
 - Ph.d. stud. i optimering af vejinfrastruktur.

- Andreas Weisberg
 - M.Sc. Software
 - Scientific Assistant på projekt omkring large-scale data analysis



Motivation (1)

- At opretholde høj kvalitet af ens database under udvikling er en stor udfordring.
- Der opstår ofte problemer i takt med at skemaet udvides.
- Analyse af "real-world" skemaer viser et klart behov for assistance til sikring af høj kvalitet.

Motivation (2)

```
CREATE TABLE address (  
  id_address      int(10) NOT NULL AUTO_INCREMENT,  
  id_country      int(10) NOT NULL,  
  id_state        int(10) DEFAULT NULL,  
  id_customer     int(10) NOT NULL DEFAULT '0',  
  id_manufacturer int(10) NOT NULL DEFAULT '0',  
  id_supplier     int(10) NOT NULL DEFAULT '0',  
  ... ,  
  KEY address_customer (id_customer),  
  KEY id_address      (id_country),  
  KEY id_address1     (id_state),  
  KEY id_address2     (id_customer),  
  KEY id_address3     (id_manufacturer),  
  KEY id_address4     (id_supplier),  
  ... )
```



Motivation (3)

```
CREATE TABLE c_country (  
  c_country_id  varchar(32) NOT NULL,  
  ad_client_id  varchar(32) NOT NULL,  
  ad_org_id     varchar(32) NOT NULL,  
  ... ,  
  CONSTRAINT c_country_key PRIMARY KEY (c_country_id),  
  ... ,  
  CONSTRAINT c_country_c_country FOREIGN KEY (c_country_id)  
    REFERENCES c_country (c_country_id),  
  ... )
```

openbravo[®]
opening ERP's future!



DBLint

- Et værktøj til automatiseret test af OLTP database designs.
- Features:
 - DBMS uafhængigt (supporterer fem DBMS'er)
 - Konfigurerbart og udvidbart
 - Hurtigt "click-and-go"
 - Kommer med 46 design regler

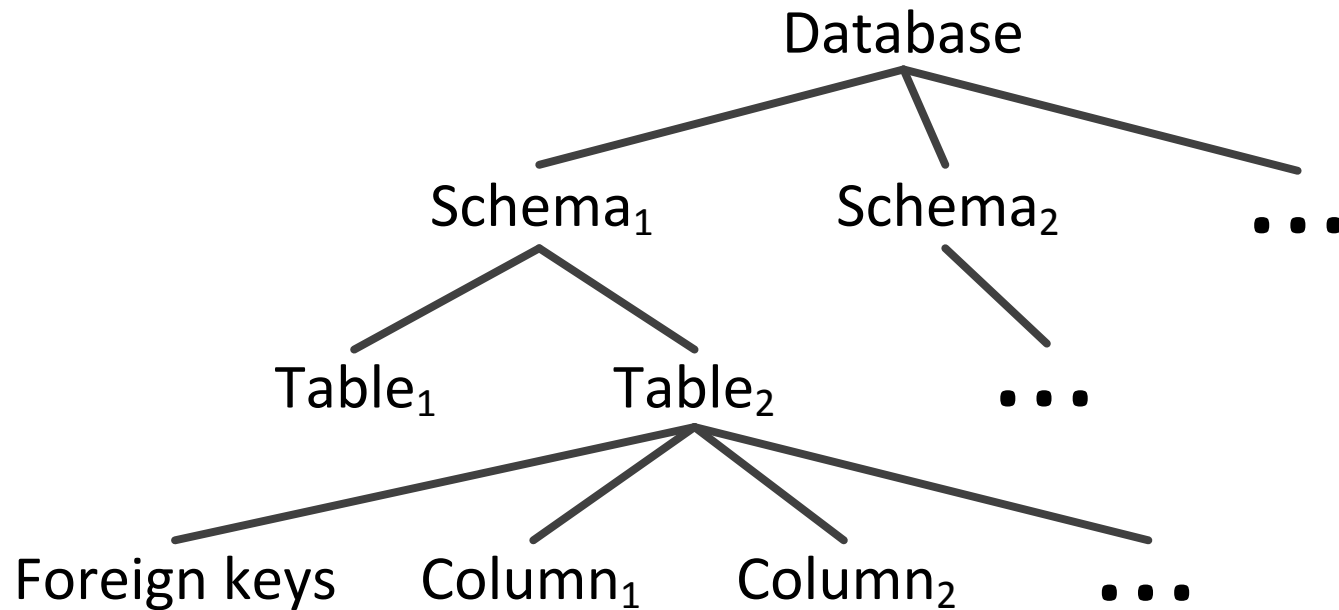


Regler og score (1)

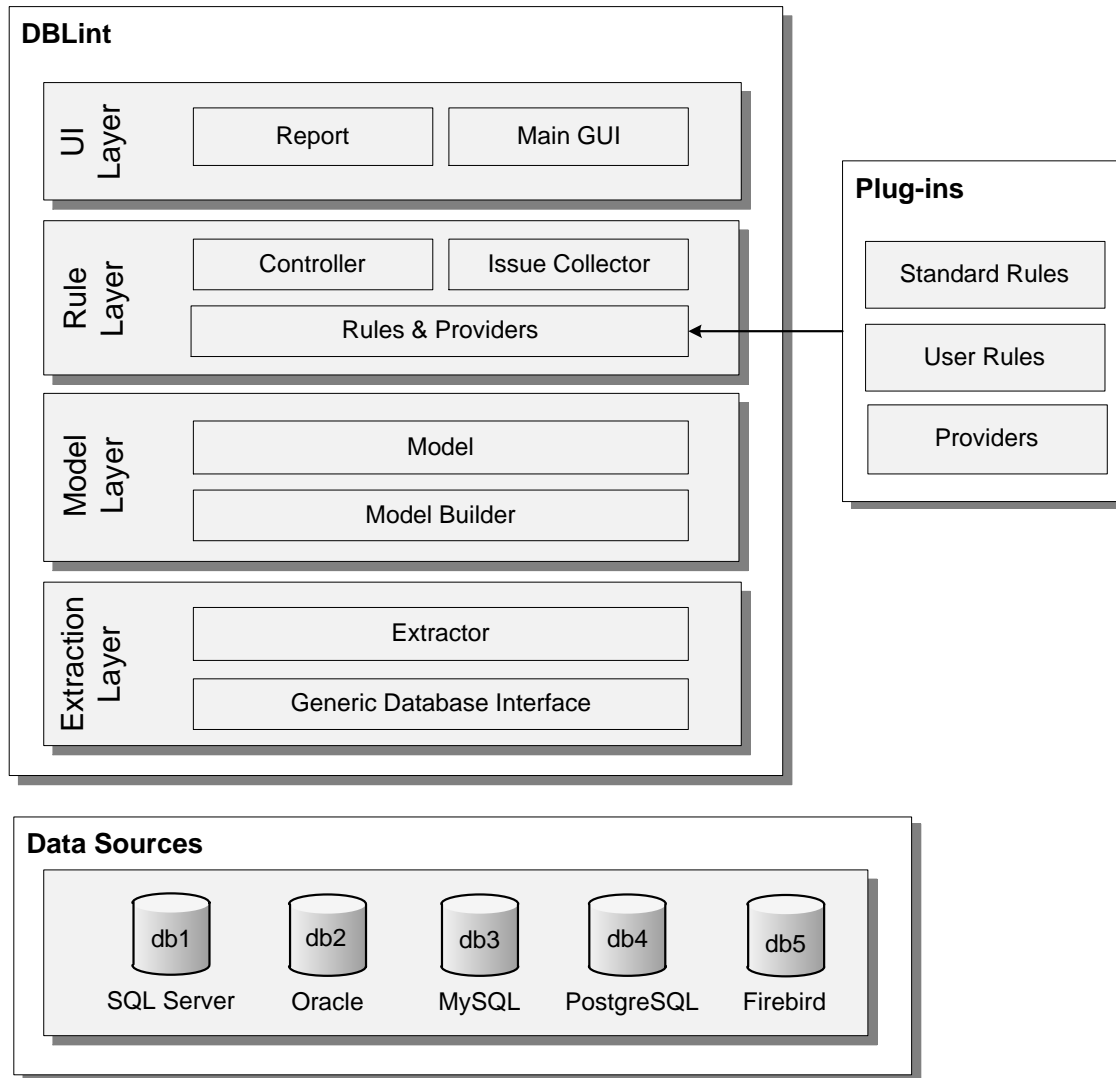
- En regel er en betingelse et database design skal overholde.
- Når en regel ikke er overholdt rapporteres der et issue.
- Systemet beregner en score, på baggrund af issues og deres alvorlighed samt placering.

Regler og score (2)

- Der gives en score til hvert element i hierarkiet



System arkitektur



Regel konstruktion

Plain SQL:

```
SELECT table_name
FROM    information_schema.tables AS p
WHERE   p.table_schema = 'public'
AND     p.table_type = 'BASE TABLE'
AND NOT EXISTS (SELECT table_name
                  FROM    information_schema.table_constraints
                  WHERE   table_schema = p.table_schema
                  AND     constraint_type = 'PRIMARY KEY'
                  AND     table_name = p.table_name)
```

In DBLint:

```
var tables = database.Tables.Where(t => t.PrimaryKey == null);
```



Regel konstruktion - data

Plain SQL

Rule1: `SELECT * FROM users WHERE A=B`

Rule2: `SELECT * FROM users WHERE C=D`

Rule3: `SELECT * FROM users WHERE E=F`

DBLint

Rule1: `foreach (var row in users.Rows) ...`

Rule2: `foreach (var row in users.Rows) ...`

Rule3: `foreach (var row in users.Rows) ...`



Hvad kan DBLint

- Kan tjekke din database for 46 generelle design regler
 - 27 metadata regler
 - 19 data regler
- Hver regel kan konfigureres
- Udvides med regler til et specifikt behov



Demo

- "Click-and-run"
- Analyse af metadata og data
- Konfigurationer og incremental run
- Specialisering af regler





dblint.codeplex.com

