

# Kolloquium Masterarbeit

Untersuchung quelloffener verteilter geografischer  
Informationssysteme zur Verarbeitung agrartechnischer  
Kennzahlen

Kurt Junghanns, B.Sc.  
([kjungha@htwk-leipzig.de](mailto:kjungha@htwk-leipzig.de))

14. Juli 2015

- 1 Einleitung
- 2 Methodisches Vorgehen
- 3 Ausgangsszenario
- 4 Systemauswahl
- 5 Untersuchung von Postgres-XL
- 6 Tests
- 7 Fazit

## Betreuer:

M. Sc. Volkmar Herbst

Prof. Dr. rer. nat. Thomas Riechert

## Unternehmen:

Agri Con GmbH

<http://agricon.de>

*Precision Farming*



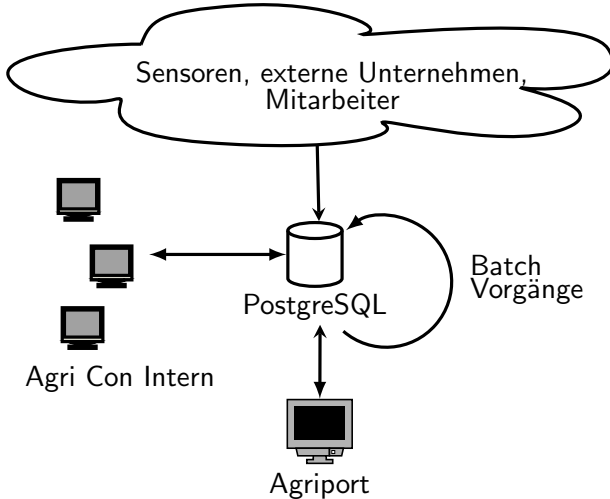


Abbildung : Aktueller Stand bei Agri Con

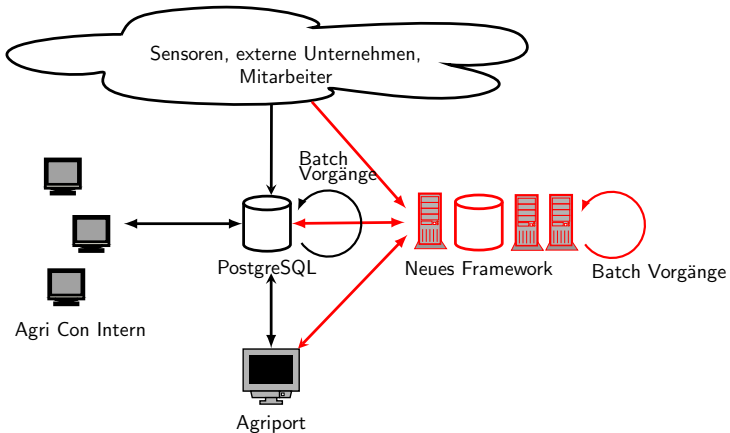


Abbildung : Ziel-Installation



Abbildung : Relevante GIS

## Anforderungen:

- große Datenmengen
- räumliche Funktionen
- geringe Laufzeiten

## Anforderungen bedingen Untersuchung alternativer Datenhaltung:

- verteiltes System
- NoSQL

*Untersuchung quelloffener verteilter geografischer Informationssysteme zur Verarbeitung agrartechnischer Kennzahlen:*

- 1 Untersuchung bestehender Frameworks anhand von Qualitätsmerkmalen
- 2 Auswahl eines Frameworks
- 3 Entwurf eines Prototypen
- 4 Prototypische Implementierung



## Softwarequalitätsmerkmale:

Funktionsumfang, Fehlertoleranz, Dokumentation, Zeitverhalten, Analysier- und Modifizierbarkeit.

## Qualitätsmetriken:

Richtigkeit, Interoperabilität, Funktionsumfang, Fehlertoleranz, Dokumentation, Zeitverhalten und Modifizierbarkeit.

## Testfälle:

Funktions- und Leistungstests.

## Stand der Forschung

- Vergleiche mit Oracle Spatial<sup>1</sup> und Neo4J<sup>2</sup> sind vorhanden
- Zur räumlichen Datenverarbeitung sind Standards des OGC<sup>3</sup> ausschlaggebend
- Semantic Web momentan ungeeignet
- Steigerung der Leistungsfähigkeit von PostgreSQL existiert thematisch als Handlungsempfehlung<sup>4</sup> und nicht in Form von wissenschaftlichen Dokumenten

---

<sup>1</sup> AHLERS, Jöhrn: Untersuchung von Techniken verteilter Datenbanksysteme zur Speicherung und Abfrage von räumlichen Daten

<sup>2</sup> BAAS, Bart: NoSQL spatial Neo4j versus PostGIS

<sup>3</sup> Standards des OGC: <http://www.opengeospatial.org/standards/is>

<sup>4</sup> SMITH, Gregory: PostgreSQL 9.0 High Performance

DBS	License	Distributed	Spatial objects	Spatial functions	PostgreSQL interface	UMN MapServer interface	Documentation	Modifiable	HDFS
AsterixDB	Apache License 2.0	yes	yes (custom)	center, radius, distance, area, intersect and cell	no	no	good in Google Code	own datatypes, functions and indexes	possible
ESRI GIS Tools for Hadoop	Apache License 2.0	yes	yes (own specific API)	yes (union, difference, intersect, clip, cut, buffer, equals, within, contains, crosses, and touches)	no	no	just briefly	forking	yes
GeoMesa	Apache License 2.0	yes	yes (Simple Features)	yes (JTS)	no (manufacturable with GeoMesa)	no	parts of the functions, a few examples	with Simple Feature Access in Java Virtual Machine and Apache Spark are all kinds of tasks	yes
H2GIS	GPL 3	no	yes (custom, no raster)	Simple Feature Access and custom functions for H2Network	yes	no	yes (homepage)	SQL	no
Ingres	GPL or proprietary	yes (if extension is installed)	yes (custom, no raster)	Geometry Engine, Open Source	no	with MapScript	just briefly	with C and OME	no
Neo4J-spatial	GNU Affero general public license	no	yes (Simple Features)	yes (contain, cover, covered by, cross, disjoint, intersect, intersect window, overlap, touch, within and within distance)	no	no	just briefly	fork or JTS	no
Postgres-XL with PostGIS	Mozilla public license and GNU general public license	yes	yes (Simple Features and raster)	yes (Simple Feature Access and raster functions)	yes	yes	PostGIS: yes, Postgres-XL: briefly	SQL, in connection with R or Tcl or Python	no
PostgreSQL with PostGIS	GNU General Public License	no	yes (Simple Features and raster)	yes (Simple Feature Access and raster functions)	yes	yes	detailed	SQL, in connection with R	no
Rasdaman	server GPL, client LGPL, enterprise proprietary	yes	just raster	raster manipulation with rasq	yes	with Web Coverage Service or Web Processing Service	detailed wiki	own defined function in enterprise edition	no

## Abbildung : Relevante GIS nach Recherche

[https://en.wikipedia.org/wiki/Spatial\\_database](https://en.wikipedia.org/wiki/Spatial_database)

Nutzwert GeoMesa: 56



Metrik	erreichter Wert	Erfüllung in %	gewichteter Teilnutzen
Interoperabilität	7	58	17
Funktionsumfang	48	79	16
Dokumentation	4	31	11
Modifizierbarkeit	4	80	12

Tabelle : Nutzwertanalyse GeoMesa

<https://raw.githubusercontent.com/geomesa/geomesa.github.io/master/img/geomesa-2x.png>

Nutzwert Postgres-XL: 86



**Postgres-XL**

Metrik	erreichter Wert	Erfüllung in %	gewichteter Teilnutzen
Interoperabilität	12	100	30
Funktionsumfang	53	87	17
Dokumentation	9	69	24
Modifizierbarkeit	5	100	15

**Tabelle** : Nutzwertanalyse Postgres-XL

<http://www.postgres-xl.org/wp-content/uploads/2014/04/xl592x497g.jpg>



Nutzwert Rasdaman: 51

Metrik	erreichter Wert	Erfüllung in %	gewichteter Teilnutzen
Interoperabilität	7	58	17
Funktionsumfang	10	16	3
Dokumentation	8	62	22
Modifizierbarkeit	3	60	9

**Tabelle** : Nutzwertanalyse Rasdaman

[http://www.rasdaman.org/chrome/site/trac\\_logo.png](http://www.rasdaman.org/chrome/site/trac_logo.png)

# Untersuchung von Postgres-XL

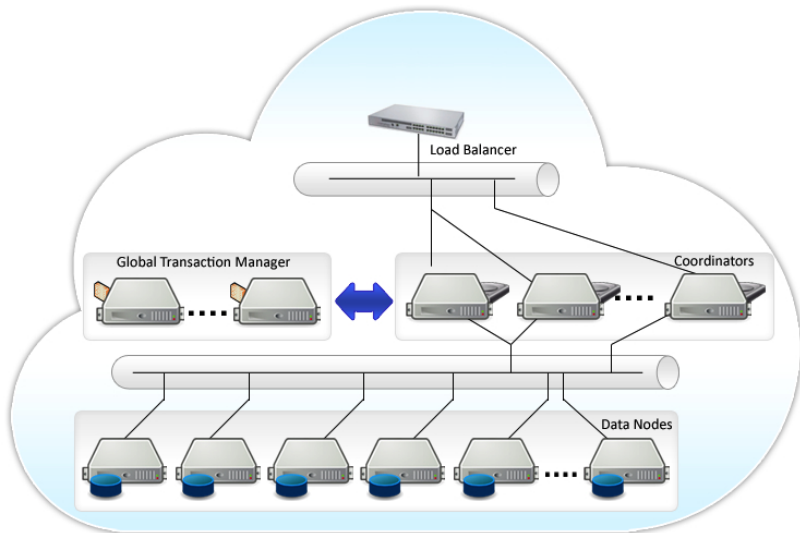


Abbildung : Aufbau von Postgres-XL

## Schnittstellen:

Erfolgt analog zu PostgreSQL mit PostGIS mit Coordinator.

## Verarbeitung:

Abhängig von der Verteilung der Daten sind ausgewählte Knoten aktiv.

Aufruf und Bibliotheken analog zu PostgreSQL mit PostGIS.



## Entwurf:

Ähnlichkeit zu Ist-Stand bedingt Übernahme von Funktionalität.

	Q3 2015	Q4 2015	Q1 2016	Q2 2016
Schemaintegration				
Normalisierung				
Anpassung der Programme				

Abbildung : Aufwandsschätzung der teilweisen Integration

## Testumgebung

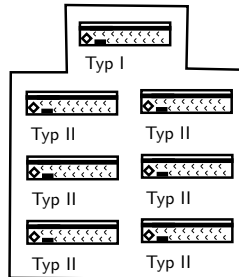
Windows System  
VMware vSphere Client



Ethernet



IBM RackServer  
VMware ESXi



Virtualisierung  
VMware ESXi

je  
Ubuntu 14.04 LTS  
2x 2,4 GHz CPU  
7 GB DDR2  
20 bzw. 100 GB

## Funktionstests

Funktionstests überprüfen grundlegende Funktionen des realen Anwendungsfalles.

6 von 7 bestanden Funktionstests.

## Funktionstests

<b>Testfall:</b>	Verschneidung von räumlichen Daten.
<b>Beschreibung:</b>	Überlagernde Vektordaten werden miteinander verschnitten.
<b>Testdaten:</b>	Ausgewählte Schläge und Teilschläge aus farm.fields.
<b>Sollergebnis:</b>	Intersection, Union, Difference und Symmetric Difference ist durchführbar und liefert das korrekte Ergebnis.
<b>Ist Ergebnis:</b>	Die Verschneidung erfolgt korrekt und kann dargestellt werden.
<b>Bestanden:</b>	Ja

Tabelle : FT05

## Leistungstests

### Testdefinition:

- Identisch für PostgreSQL und Postgres-XL
- Durchführung mit JMeter
- Messung mit Zabbix
- Mittlung der Ergebnisse

### Vorüberlegungen:

- Kostenmaß als Möglichkeit der Berechnung der theoretischen Leistung
- Anpassung des Datenbankschemas an Verteilung der Daten
- Anpassung des Query Planers
- Skalierbarkeit für verteilt arbeitende Systeme relevant

## Leistungstests - Aggregation

### Definition:

- Pro Coordinator 3 Threads
- 5 Wiederholungen
- Aggregation von 5 Datensätzen

### Ergebnis:

Postgres-XL: 3619ms

PostgreSQL: 4319ms

## Leistungstests - Aggregation

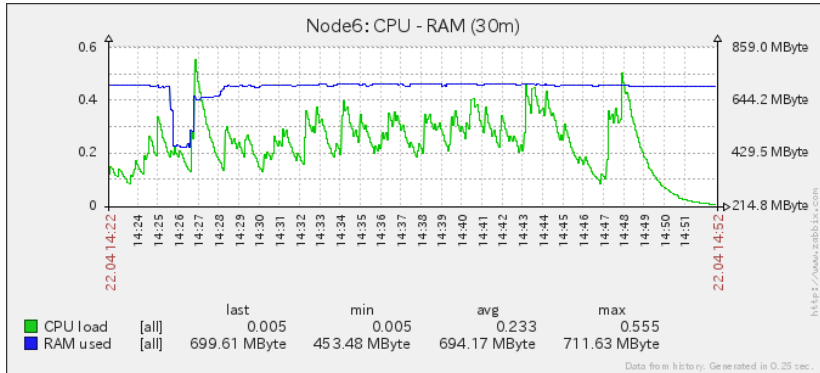


Abbildung : Auslastung des Knotens 6

Durchschnittliche Antwortzeit	Aggregation	Aggregation mit Kartendarstellung
Postgres-XL	3,6s	2,4s
PostgreSQL	4,3s	2,6s

Tabelle : Vergleich der Testergebnisse

Mittlere CPU Auslastung	Aggregation	Aggregation mit Kartendarstellung	Verarbeitung
Postgres-XL	0,2	0,1	0,5
PostgreSQL	3,6	0,4	0,5

Tabelle : Vergleich der CPU Auslastung, GTM VM nicht berücksichtigt



## Nutzwertanalyse mit Zeitverhalten und angepasster Wertung Nutzwert: 79

<b>Metrik</b>	<b>erreichter Wert</b>	<b>Erfüllung in %</b>	<b>gewichteter Teilnutzen</b>
Interoperabilität	12	100	20
Funktionsumfang	53	87	17
Dokumentation	8	69	10
Zeitverhalten	2	67	27
Modifizierbarkeit	5	100	5

**Tabelle :** Neue Nutzwertanalyse von Postgres-XL

## Ergebnisse:

- Die Methodik ist geeignet für die Softwareauswahl
- Das Forschungsziel wurde erreicht
- Postgres-XL kann für einen produktiven Einsatz nicht empfohlen werden
- PostgreSQL mit PostGIS ist für Agri Con geeignet

**Vielen Dank für Ihre Aufmerksamkeit!**



## Anforderungen an die Technologie:

- PostgreSQL mit PostGIS zum Datenimport und -export nutzbar
- Gruppierung und Filterung mit geringer Laufzeit
- Parallele Berechnung über große Datenmengen mit geringer Laufzeit
- Räumliche Berechnungen wie Verschneidung und Overlays
- Nutzbare Schnittstelle zur Darstellung mit dem UMN MapServer

## Qualitätskriterium:

Es sind Schnittstellen zur Ein- und Ausgabe vorhanden. Dabei soll es sich um PostgreSQL Import sowie PostgreSQL und UMN Export handeln.

## Qualitätsmetrik:

Ist der Import und Export von räumlichen Daten aus PostgreSQL sowie eine Anbindungsmöglichkeit an den UMN. Statische Abbildung:

*[Datenschnittstelle und UMN Schnittstelle vorhanden,  
Datenschnittstelle vorhanden, UMN Schnittstelle vorhanden,  
keine Schnittstelle vorhanden] nach [12, 7, 5, 0]*

<b>Metrik</b>	<b>Gewichtung in %</b>
Interoperabilität	30
Funktionsumfang	20
Dokumentation	35
Modifizierbarkeit	15

**Tabelle** : Wertungsmaßstab der einzelnen Metriken

<b>Metrik</b>	<b>Gewichtung in %</b>
Interoperabilität	20
Funktionsumfang	20
Dokumentation	15
Zeitverhalten	40
Modifizierbarkeit	5

**Tabelle** : Neuer Wertungsmaßstab der einzelnen Metriken

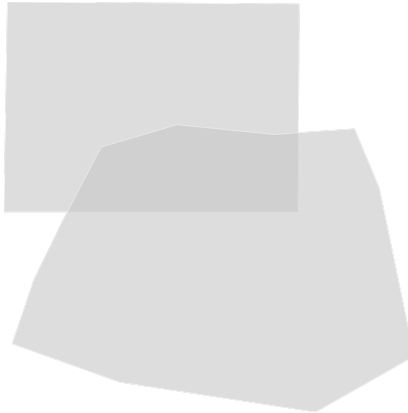
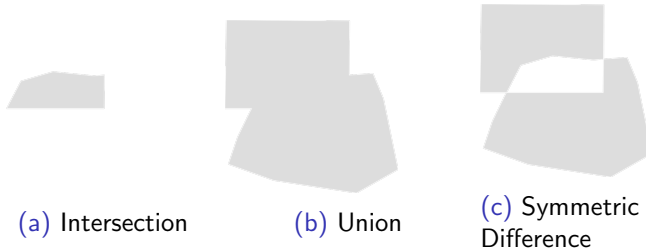
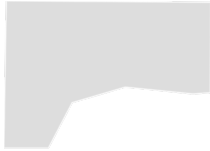


Abbildung : Kartenausschnitt mit 2 überlappenden Schlägen





**Abbildung** : Ergebnisse der Verschneidungsfunktionen Intersection, Union und Symmetric Difference



(a) Difference mit Schlag 1 zu Schlag 2



(b) Difference mit Schlag 2 zu Schlag 1

Abbildung : Ergebnisse der Verschneidungsfunktionen Difference

Interpolation mit Hilfe spezieller Algorithmen von Punkten zu Flächen.

## Probleme:

- SQL Befehl *Insert Into* ist in Plr Funktionen nicht verwendbar
- Die meisten Datensätzen konnten nicht verarbeitet werden
- Nur ein Coordinator in Testläufen verwendbar
- Keine Kontrolle der Korrektheit der Ergebnisse