

# WP Datenbankdesign



## Kapitel 7: Geo-Datenbanksysteme

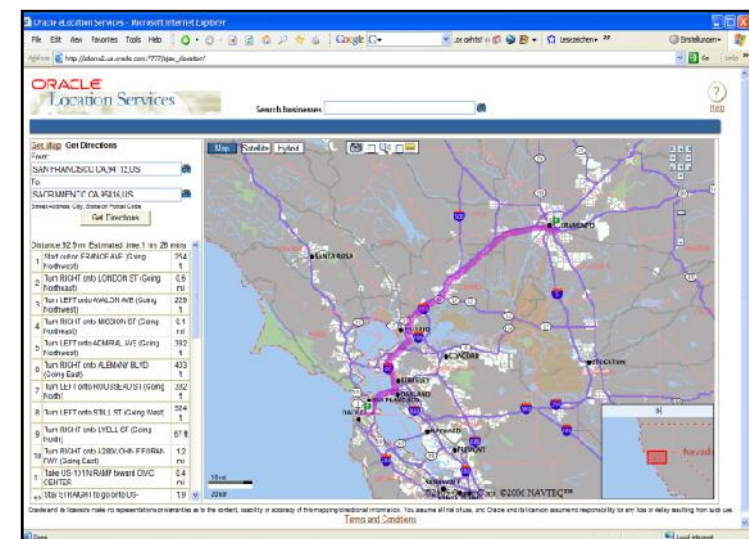
- **Geo-Informationssysteme**
- **Der Datentyp SDO-Geometry**
- **Räumliche SQL-Abfragen**
- **Architektur**



**Informationssystem mit Verarbeitung und Darstellung räumlicher Daten.**

## Anwendungen

1. Facility Management
2. Vermessungswesen, ...
3. Umweltschutz
4. Telekommunikation
5. Verkehr
6. Marketing
7. Logistik
8. Versicherungswirtschaft



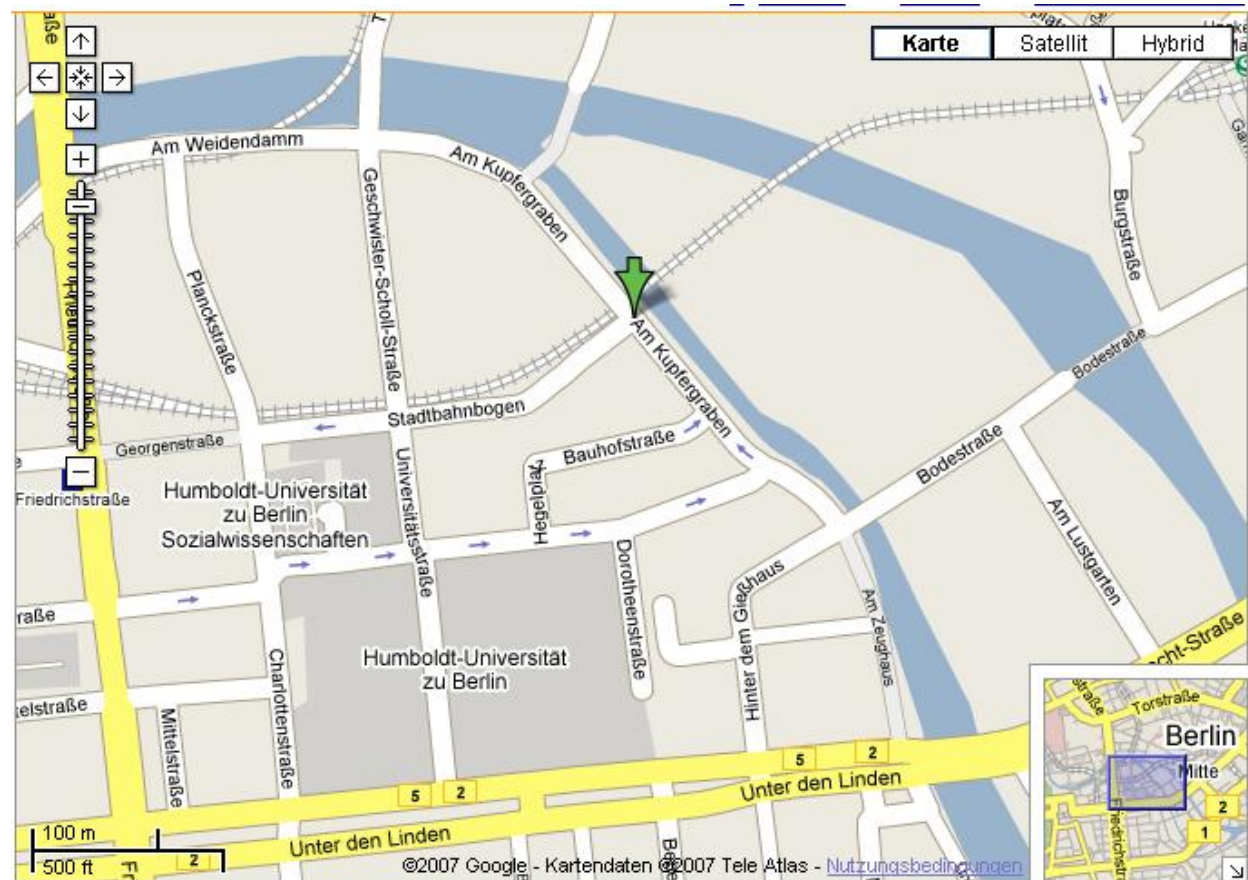
# Beispiel: Geo-Marketing



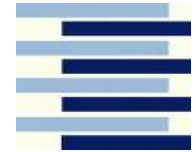
Beschreibungen eines geographischen Marktgebiets anhand detaillierter Consumer- und/oder Business-Daten. Räumliche Analysen verknüpfen unternehmenseigene Daten mit Geodaten und Marktinformationen.

## Zum Beispiel:

- Umsatz
- Einwohnerzahl
- Einwohner nach Altersklassen
- Kaufkraft
- Dominierendes Milieu
- Zahlungsmoral



# Was wissen wir über dieses Viertel?



Die GfK kennt Einkommen und Kaufkraft für jeden Straßenabschnitt in Deutschland. Hier einige Zahlen: **Einwohner** : 571      **Haushalte**: 313

## Haushaltstypen:

Haushalte mit Kindern:	<b>29 %</b>	(Bundesdurchschnitt 33 %)
Single-Haushalte:	<b>40 %</b>	(37 %)
Mehrpersonen-Haushalte ohne Kinder:	<b>30 %</b>	(30 %)

## Monatliches Nettoeinkommen der Haushalte:

bis 1.100 €	<b>16 %</b>	(23 %)
1.100 € bis unter 1.500 €	<b>12 %</b>	(18 %)
1.500 € bis unter 2.000 €	<b>13 %</b>	(17 %)
2.000 € bis unter 2.500 €	<b>19 %</b>	(16 %)
2.500 € bis unter 4.000 €	<b>24 %</b>	(17 %)
4.000 € und mehr	<b>16 %</b>	( 9 %)

Quelle:

ZeitWissen,  
4/2006

Ges. Kaufkraft je Einwohner und Jahr: **16.686 €** (17.087 €)

# Wo kommen die Daten her?



- Gesellschaft für Konsumforschung (Bsp. [www.martviewer.com](http://www.martviewer.com))  
regelmäßige Umfragen bei repräsentativen Haushalten  
Straßenbegehungen
- Schober Konzern pro Jahr 600.000 Lifestyle Fragebögen mit 120 Fragen  
(infas, [www.infas-geodaten.de](http://www.infas-geodaten.de))
- Otto-Versand
- Creditreform
- Kraftfahrt-Bundesamt
- Landesvermessungsämter
- Deutsche Post

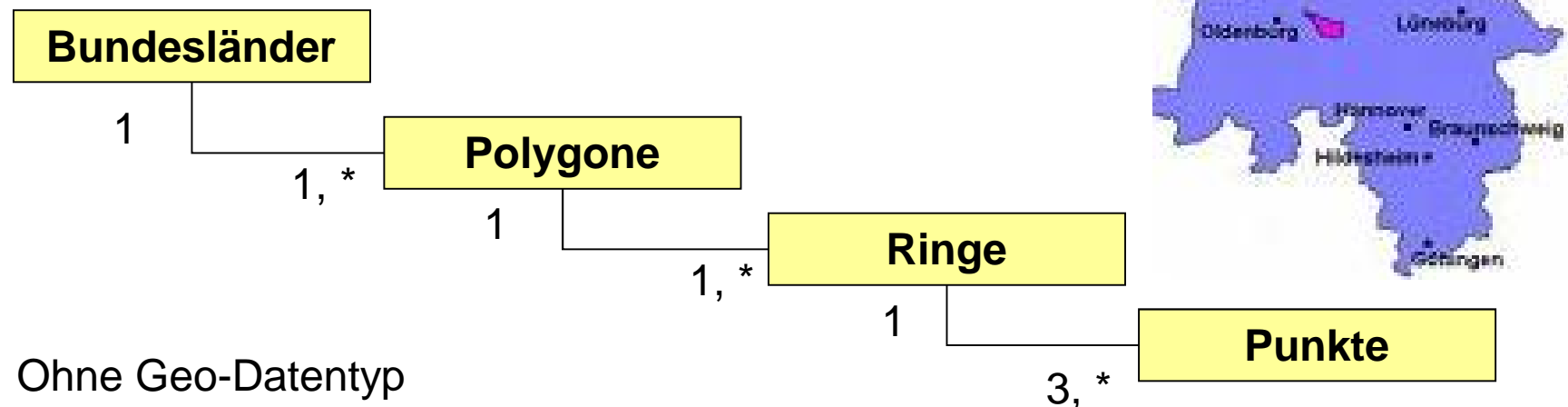
Jedes Gebiet wird einem von 10 definierten sozialen Milieus zugeordnet.



# Anforderungen an ein Geo-Datenbanksystem



- Geometrische Datentypen ( objektrelationale DBMS)
- Methoden zum Aufruf geometrischer Funktionen
- Gleichartiger, gemeinsamer Zugriff auf strukturierte relationale und geometrische Daten (Attribute)
- Verwendung von Standards aus dem Bereich GIS



# Der Oracle Datentyp SDO-Geometry



Tabelle Gemeinde

ID	NAME	Lokation
NUMBER	VARCHAR2	SDO_GEOMETRY

SDO_GTYPE	NUMBER
SDO_SRID	NUMBER
SDO_POINT	SDO_POINT_TYPE
SDO_ELEM_INFO	SDO_ELEM_INFO_ARRAY
SDO_ORDINATES	SDO_ORDINATE_ARRAY



# Bedeutung der Attribute



<b>GType</b>	Geographic type D00T Dimension $D \in \{2, 3, 4\}$ Form/Typ $T \in \{1=\text{Punkt}, 2=\text{Linienzug}, 3=\text{Polygon}, \dots\}$
<b>SRID</b>	Spatial Reference System z.B. 8307 für geodätisches Koordinatensystem lt. WGS 84
<b>Point</b>	Punkt (X, Y, Z) z. B. X=Längen-, Y=Breitengrad
<b>Elem_Info</b>	Element-Info-Array Offset, Elementtyp (vgl. T), Interpretation z. B. (1, 2, 2)
<b>Ordinates</b>	VArray mit Koordinatenwerten

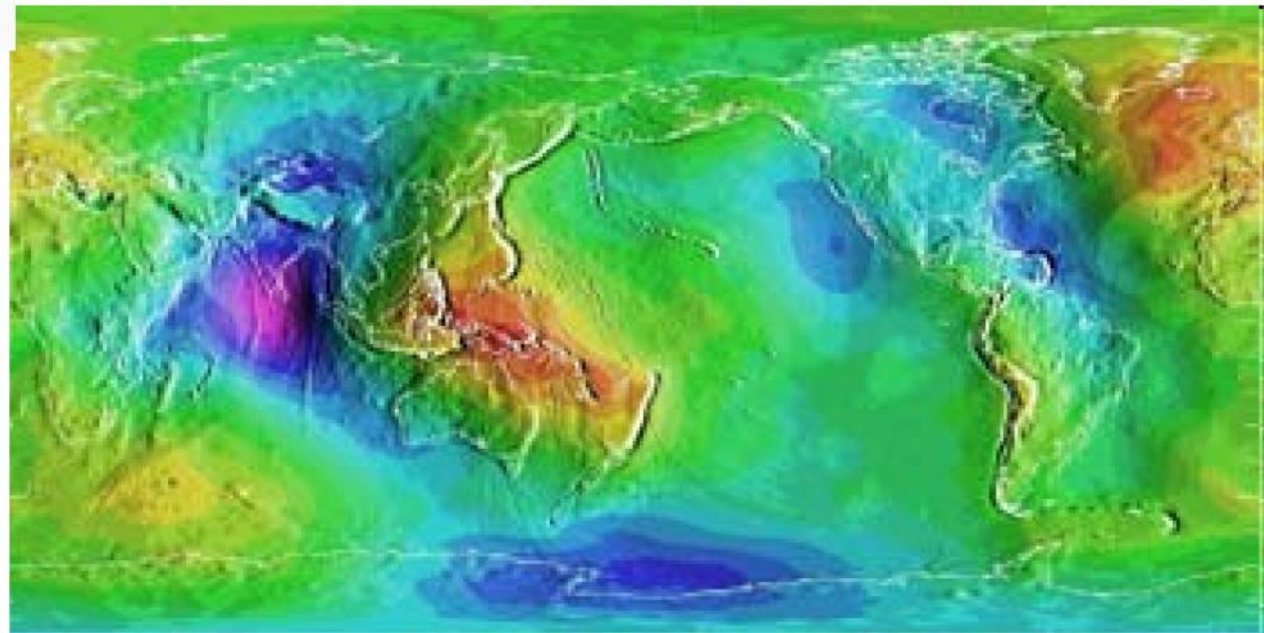
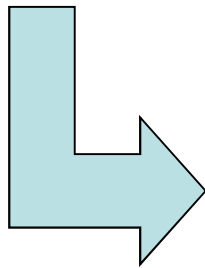
verbunden mit  
Kreusbögen



# Geodätische Koordinatensysteme



180 .. - 180 ° östliche/westliche Länge  
90 .. - 90 ° nördliche/südliche Breite



# Oracle Spatial und Oracle MAPS

## Hands-On Workshops



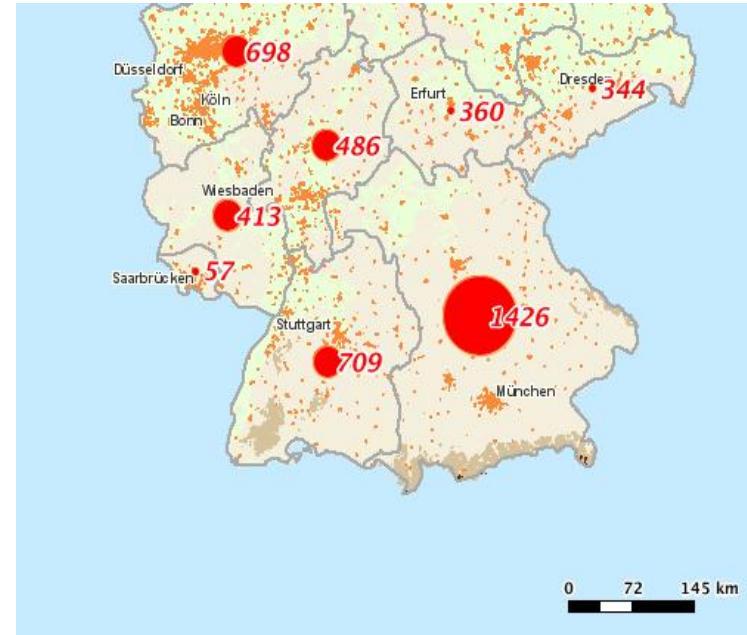
**Bernhard Fischer-Wasels**  
([bernhard.fischer-wasels@oracle.com](mailto:bernhard.fischer-wasels@oracle.com))

**Haitham Zyadeh**  
([haitham.zyadeh@oracle.com](mailto:haitham.zyadeh@oracle.com))

**Rainer Meisriemler**  
([rainer.meisriemler](mailto:rainer.meisriemler))

**Carsten Czarski**  
([carsten.czarski@oracle.com](mailto:carsten.czarski@oracle.com))

**ORACLE Deutschland GmbH**



**Quelle der folgenden Folien**

# Oracle Spatial: OGC Simple Features

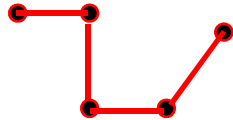
## Geometrische Elemente



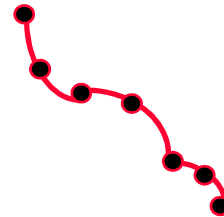
Point



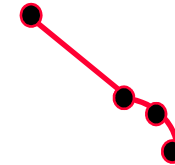
Line string



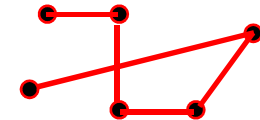
Arc line string



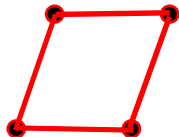
Compound line string



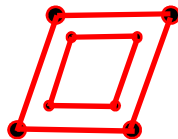
Self-crossing line strings



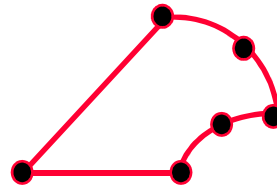
Polygon



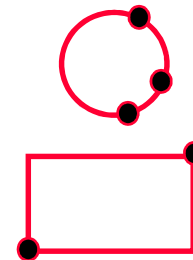
Polygon with hole



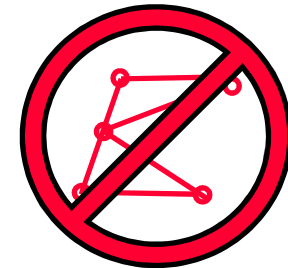
Compound polygon



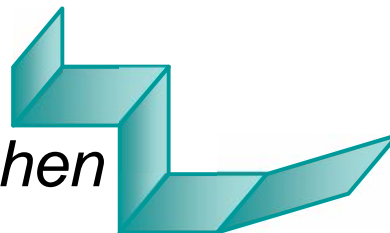
Optimized polygons



Self-crossing polygons



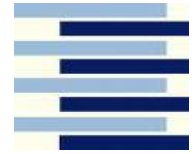
*Zusätzlich Oberflächen und Körper*



Geo-Datenbanksysteme



# Geodaten-Tabelle erstellen



## 1. Tabelle anlegen

- SDO\_GEOMETRY wie jeden anderen Datentyp verwenden
- Empfehlung: Nur eine SDO\_GEOMETRY-Spalte

```
CREATE TABLE GEO_KONTINENTE(  
    "AREA_ID"          NUMBER,  
    "AREA_NAME"        VARCHAR2(20 BYTE),  
    "FEATURE_TYPE"     CHAR(9 BYTE),  
    "GEOMETRY"         SDO_GEOMETRY  
)
```

- Geodaten können sofort gespeichert werden
- Indexerstellung allerdings noch nicht möglich: Schritt 2

# Geodaten-Tabelle erstellen - 2



## 2. Spatial-Metadaten eintragen

(Deklaration, **welche** Geodaten gespeichert werden)

- Koordinatensystem
- Mögliche räumliche Ausdehnung

```
insert into user_sdo_geom_metadata values (  
  'GEO_KONTINENTE',   
  'GEOMETRY',   
  sdo_dim_array(  
    sdo_dim_element('X',-180,180,  
    sdo_dim_element('Y',-90,  
  ),  
  8307  
)
```

Name der Tabelle

Name der Tabellenspalte

Informationen zu den

m Längengrade:

Breitengrade:

-90 = Südpol

Koordinatensystem:

8307 = WGS 84

# Geodaten-Tabelle erstellen - 3



## 3. Index erstellen

- CREATE INDEX-Kommando
- Spalte vom Typ SDO\_GEOMETRY angeben
- Indextype MDSYS.SPATIAL\_INDEX angeben

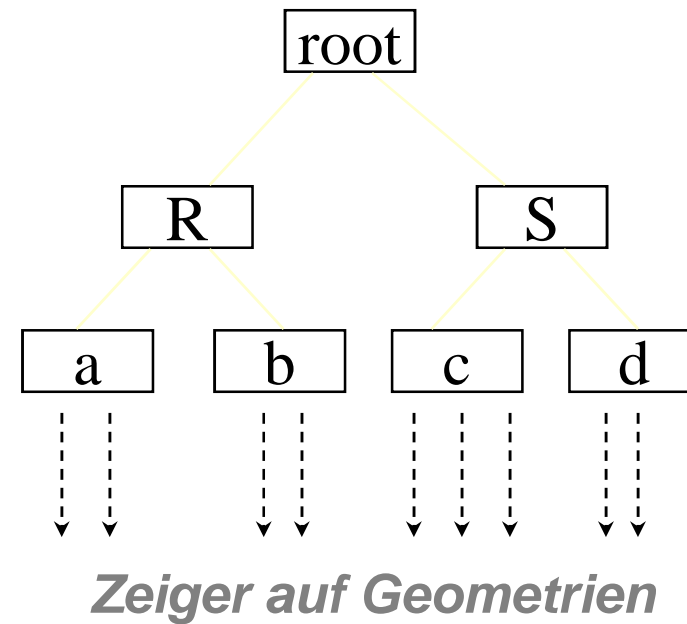
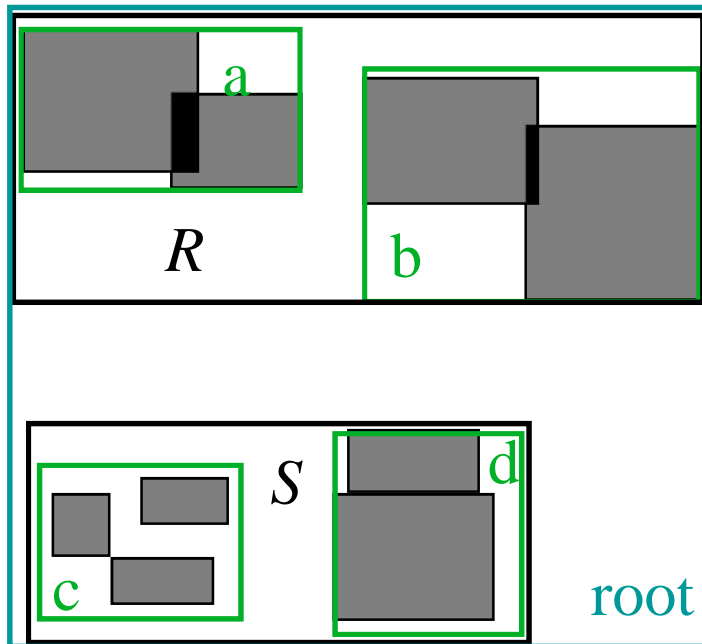
```
CREATE INDEX IDX_GEODATEN  
ON GEO_KONTINENTE(GEOMETRY)  
INDEXTYPE IS MDSYS.SPATIAL_INDEX
```

- fertig!

# Spatial Index



## Aufbau des R-Tree-Index





# Geometrische Operationen



- Wie groß ist Deutschland?

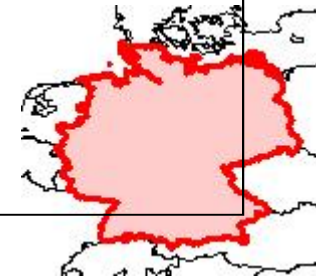
```
SQL> select sdo_geom.sdo_area(  
2     geometry, 1, 'unit=sq_km'  
3 ) "Fläche km²"  
4 from geo_staaten e  
5 where feature_name='GERMANY';
```

Einheit (km²)

Fläche km²

-----  
356644,105

1 Zeile wurde ausgewählt.



# Geometrische Operationen



- Wie lang ist die deutsche Grenze?

```
SQL> select sdo_geom.sdo_length(  
2     geometry, 1, 'unit=km'  
3 ) "Länge km"  
4 from geo_staaten e  
5 where feature_name='GERMANY';
```

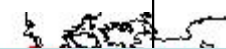
Einheit (km)

Länge km

-----

5845,59665

1 Zeile wurde ausgewählt.



Länge des Polygonzugs,  
der Deutschland  
repräsentiert; nicht  
gleich der "offiziellen"  
Länge der Grenze

# Geometrische Operationen



- Welche Bundesländer grenzen an die Niederlande?

```
SQL> select bl.feature_name
2   from geo_staaten st, geo_laender bl wh
3   st.feature_name='NETHERLANDS' and
4   sdo_relate(
5       bl.geometry, st.geometry, 'mask=
6   )= 'TRUE'
```

Es wurden keine Zeilen ausgewählt.

Join zwischen  
Die Grenze der  
Niederlande ist  
von Interesse

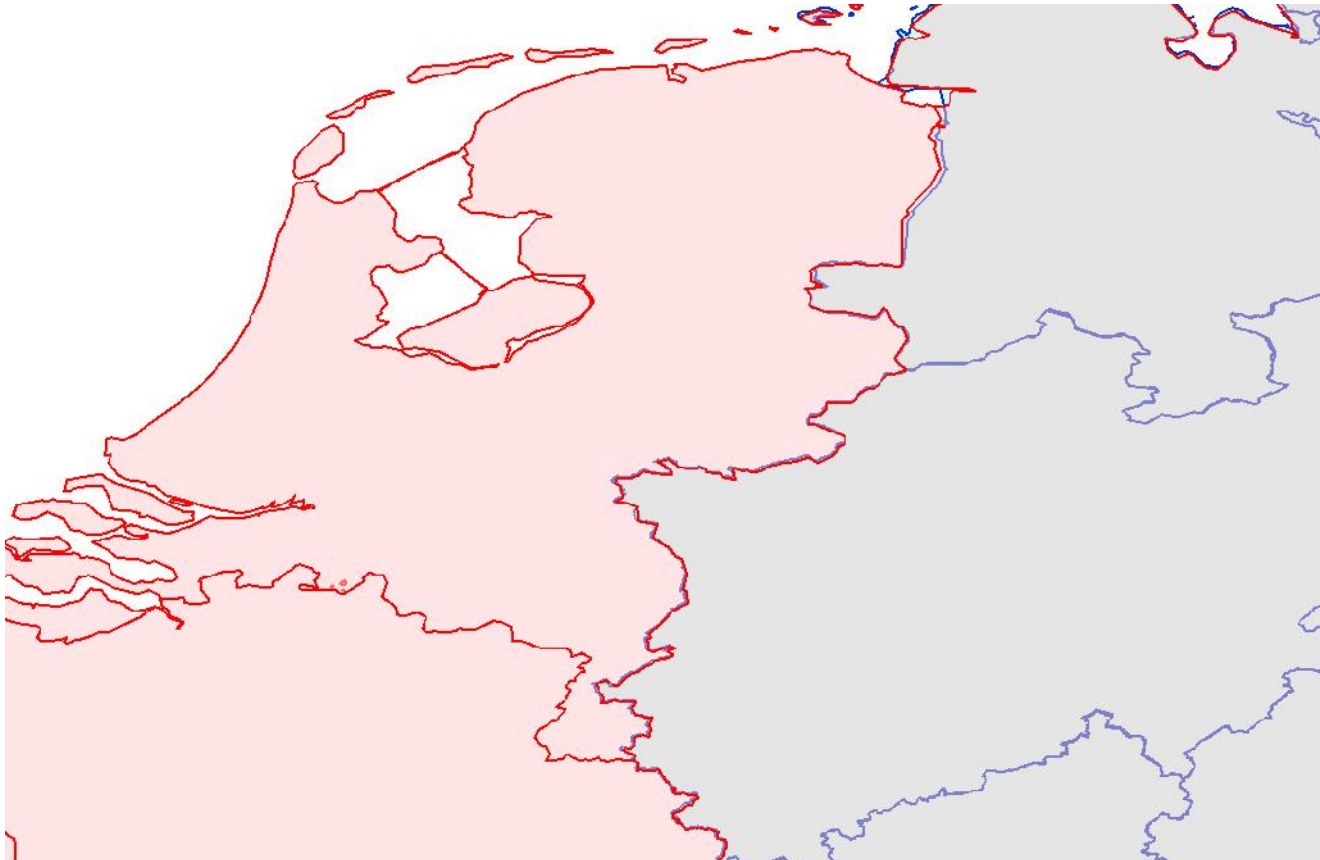
Welche Bundes-  
länder berühren  
("touch") die  
Niederlande?

*Hmmm ... das stimmt aber nicht so ganz ...*

# Geometrische Operationen



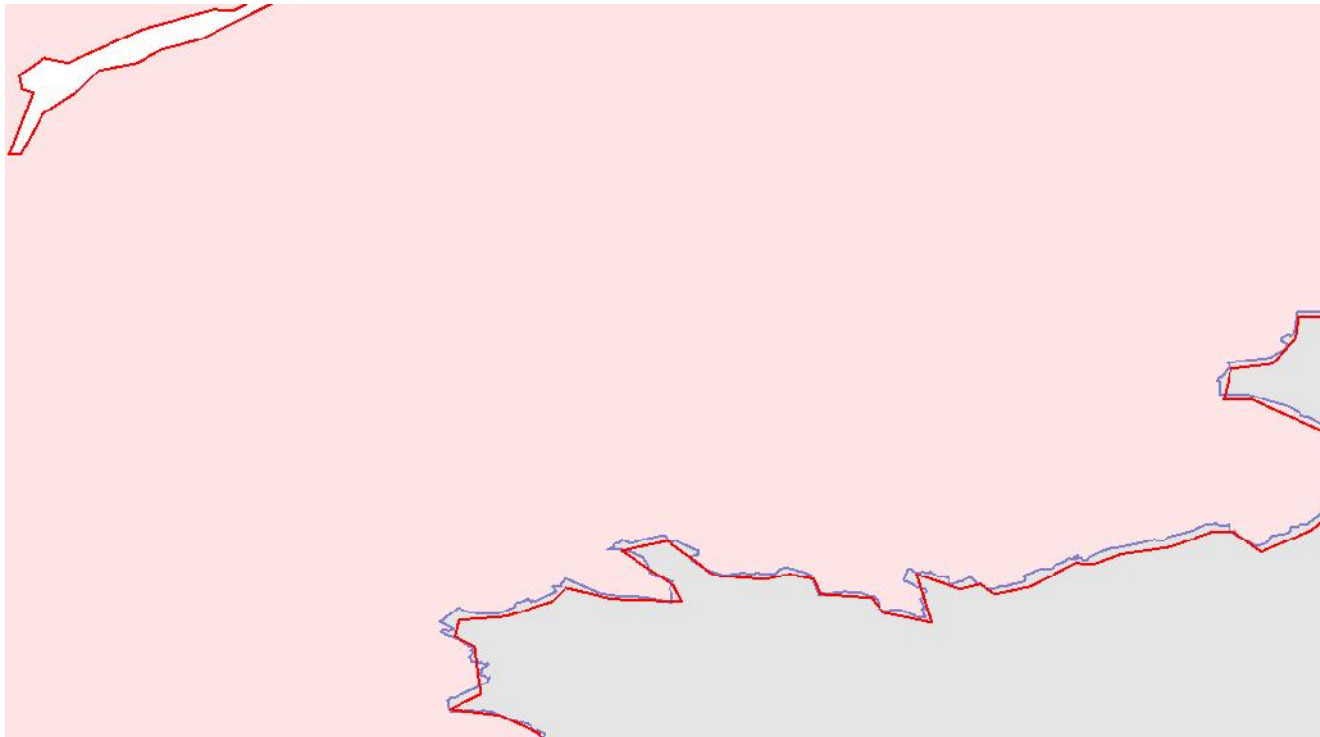
- Die Grenze im Detail ...



# Geometrische Operationen



- Die Grenze im Detail ... und vergrößert ...



***Das ist die Praxis:  
Die Daten (hier: Demodaten) sind nicht immer so genau ...***

# Geometrische Operationen



- Welche Bundesländer grenzen an die Niederlande?

```
SQL> select bl.feature_name
      2   from geo_staaten st, geo_laender bl where
      3   st.feature_name='NETHERLANDS' and
      4   sdo_relate(
      5     bl.geometry, st.geometry, 'mask=anyinteract'
      6   )= 'TRUE'
```

```
FEATURE_NAME
-----
Niedersachsen
Nordrhein-Westfalen
```

Welche Bundesländer  
hängen irgendwie  
("anyinteract") mit den  
Niederlanden zusammen?

# Kombinierte Abfragen



- Welche Kunden mit Umsatz sind in Brandenburg?

```
SQL> select name, umsatz
2   from customers c, geo_laender gl
3   where sdo_relate(
4       c.geometry, gl.geometry, 'mask=anyinteract'
5   ) = 'TRUE'
6   and gl.feature_name='Brandenburg'
7   and umsatz is not null and umsatz > 0
```

NAME	UMSATZ
-----	-----
Freie Tanke	4876
Marhoul	4798
Tankstelle Schönefeld	4598
Tankstelle Dreieck Havelland	7896
:	:



## Kombinierte Abfragen - 2



- Welche Kunden wohnen im 50 km-Umkreis um Berlin?

```
select
  c.name,                -- Name des Kunden
  c.umsatz               -- Umsatz des Kunden
from geo_orte o, customers c
where sdo_within_distance( -- Funktion für Umkreissuche
  c.geometry,
  o.geometry,
  'distance=50 unit=km'   -- Umkreis von 50km
) = 'TRUE'
and o.feature_name = 'Berlin'
and umsatz is not null    -- Nur Kunden mit Umsatz
```

# Räumliche Abfragen

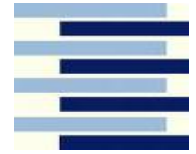


- Wie lauten die Koordinaten von Deutschland?

```
SQL> select id, x, y from
2     geo_staaten,
3     table(sdo_util.getvertices(geometry)) k
4     where feature_name='GERMANY';
```

ID	X	Y
1	14,720045	51,582406
2	14,7554651	51,603937
3	14,7570239	51,644129
:	:	:

# Geocoding mit Oracle Spatial



- Umwandlung von Adressen in Koordinaten ...  
... und rückwärts (REVERSE GEOCODING)
- PL/SQL Paket SDO\_GCDR

ID	STRASSE	PLZ	ORT
1	Riesstr. 25	80992	München
2	Marienplatz		München
3	Notkestr.	20000	Hamburg
...	...	...	...

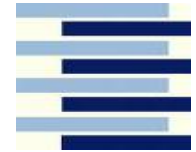
# Geocoding mit Oracle Spatial



- Umwandlung von Adressen in Koordinaten ...  
... und rückwärts (REVERSE GEOCODING)
- PL/SQL Paket SDO\_GCDR

ID	STRASSE	PLZ	ORT	LATITUDE	LONGITUDE
1	Riesstr. 25	80992	München	11.5366999	48.1800458
2	Platzl	80331	München	11.57917	48.13842
3	Notkestr.	22607	Hamburg	9.87543	53.57068
...	...	...	...	...	...

# Geodaten: Visualisierung ...?



## Kunden ...

### Aktuelle SRID

Longitude / Latitude (DHDN)

ID	Name	Länge	Breite
116	Fritz Muster	6°50'58"	49°32'20"
604	Mike Müller	6°38'42"	50°03'21"
2623	Eifel Power II	6°46'46"	50°03'34"
2637	Carsten Czarski	6°50'40"	49°59'39"
2216	Wowereit	13°17'05"	52°25'31"
4158	Klaus Weber	7°13'31"	49°59'10"
4009	Karl Hunsrück	6°57'18"	49°51'35"
5503	Gerolsteiner Sprudel GmbH	6°34'05"	50°11'26"
4990	Der Säubrenner	6°51'07"	49°58'16"
7512	Georg Meistermann	6°44'45"	49°38'37"
6552	Christian Ude	11°34'48"	48°08'56"
7760	Alfons Schommer	6°48'34"	49°30'17"
7761	Heinz Becker	7°01'44"	49°14'55"
7532	Jürgen V.	6°27'51"	49°51'08"
9252	Karlo Umsatz	7°59'53"	52°16'32"

1 - 15 Weiter ►

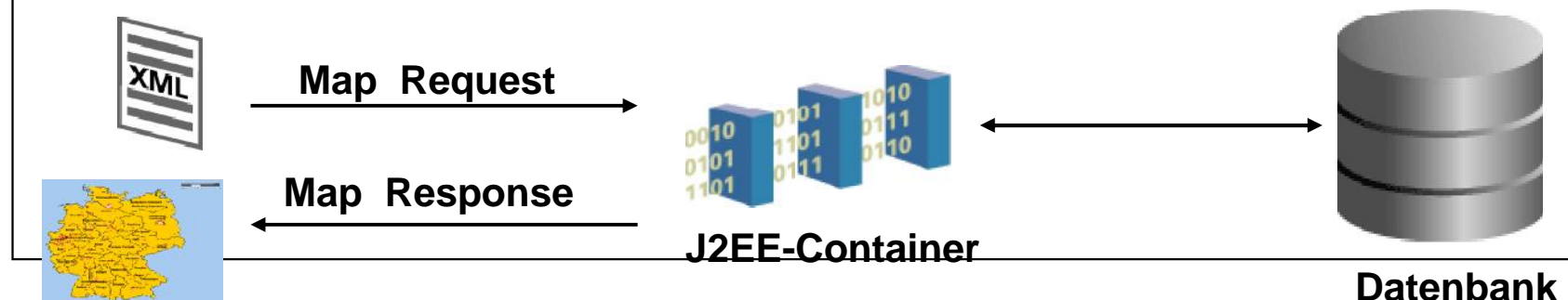


# Visualisierung der Geodaten

## Oracle MapViewer



- Karten-Rendering für ...  
Vektordaten, Rasterdaten, Netzwerke, Topologien
- mehrere Layer: Base map, features of interest (FOI)
- Architektur: J2EE
  - Lizenz: Oracle Application Server: Java Edition



# Map Viewer: Einbindung in APEX



HTMLDB\_PUBLIC\_USER | Dr

Einfache Geometrien Räumliche Abfragen Projektion Rasterdaten » Beispielanwendung Daten-Import

X,Y: 3150970, 5215651

Maus-Modus: [Zoom](#) [Kunden im Bereich](#) [Kunde bearbeiten](#) [Karte verschieben](#)

**Einstellungen**

**Oracle-Layer**
☐ Bundesländer  
☐ Reg.Bezirke  
☐ Flüsse  
☐ Autobahnen  
☐ Ballungsräume  
☒ **Kunden**

**WMS (www.ccgis.de)**
☒ Topographie  
☒ Städte  
☒ Städtenamen  
☐ Flüsse  
☐ Strassen  
☐ Bahnlinien  
☒ Bundesländer  
☒ Bundesländer (Namen)

**WMS (http://www.deegree.de)**
☐ Europa (politisch)  
☐ Europa (Flüsse)  
☐ Europa (Städte)  
☐ Osnabrück (Gewässer)  
☐ Osnabrück (Flüsse)  
☐ Osnabrück (Strassen)  
☐ Osnabrück (Grünflächen)  
☐ Osnabrück (Sehensw.)

**Raster Themes**
☐ BKG\_GERMANY  
☐ ADAC\_KARTEN  
☐ BKG\_GERMANY\_2500

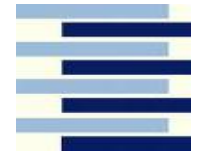
**[Karte erneuern]**

**Kunden ...**

	Name	Umsatz	Land
	Karl Hunsrück	10.000,00 €	Rheinland-Pfalz
	Guido Herges	9.001,00 €	Rheinland-Pfalz
	Klaus Weber	9.000,00 €	Rheinland-Pfalz
	Jörn Huber	4.000,00 €	Rheinland-Pfalz
	Michael Künzner	2.300,00 €	Rheinland-Pfalz

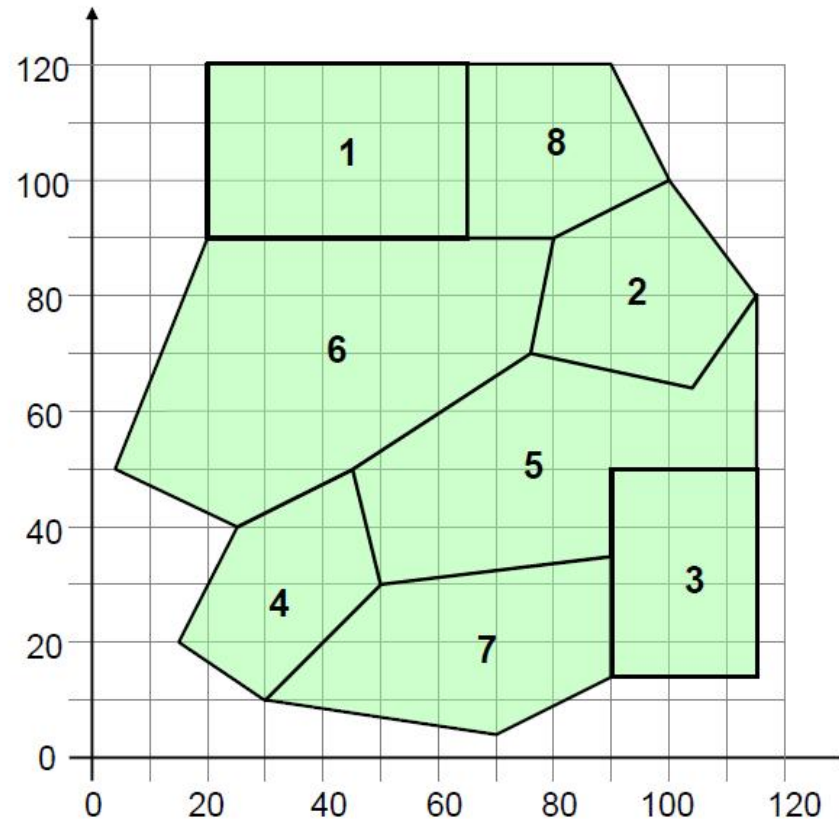
<http://blog.whitehorses.nl/2009/10/04/integrating-google-maps-in-oracle-apex/>  
<http://www.oracle.com/webfolder/technetwork/de/community/apex/tipps/geo-1/index.html>





## Beispiel

### Grundstücks- flächen



```
create TABLE grundstücke  
(id number (5) primary key, Bezeichnung varchar(29), Fläche sdo_Geometry);
```

```
insert into MDSYS.user_sdo_geom_metadata values('GRUNDSTÜCKE', 'FLÄCHE',  
mdsys.sdo_dim_array  
(mdsys.sdo_dim_element('x', 0,120,1), mdsys.sdo_dim_element('y', 0, 120,1)), null);
```

# SQL-Befehle



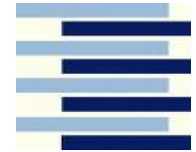
create index g\_spatial\_idx on grundstücke(Fläche) indextype is mdsys.spatial\_index;

```
INSERT INTO Grundstücke  
VALUES (1, 'meins', MDSYS.SDO_GEOMETRY (2003, NULL, NULL,  
MDSYS.SDO_ELEM_INFO_ARRAY(1, 1003, 1),  
MDSYS.SDO_ORDINATE_ARRAY(20,90,65,90,65,120,20,120,20,90)));
```

```
INSERT INTO Grundstücke  
VALUES (2, 'Konstantin', MDSYS.SDO_GEOMETRY (2003, NULL, NULL,  
MDSYS.SDO_ELEM_INFO_ARRAY(1, 1003, 1),  
MDSYS.SDO_ORDINATE_ARRAY(65,90,80,90,100,100,90,120,65,120,65,90)));
```

```
INSERT INTO Grundstücke  
VALUES (3, 'Katharina', MDSYS.SDO_GEOMETRY (2003, NULL, NULL,  
MDSYS.SDO_ELEM_INFO_ARRAY(1, 1003, 1),  
MDSYS.SDO_ORDINATE_ARRAY(20,90,5,50,25,40,45,50,75,70,80,90,20,90,20,90)));
```

# SQL-Queries



select \* from grundstücke g;

ID	BEZEICHNUNG	FLÄCHE
1	1 meins	[MDSYS.SDO_GEOMETRY]
2	2 Konstantin	[MDSYS.SDO_GEOMETRY]
3	3 Katharina	[MDSYS.SDO_GEOMETRY]

select g.bezeichnung from grundstücke m, grundstücke g where  
m.bezeichnung='meins' and sdo\_relate(g.fläche,m.fläche, 'mask=touch')= 'TRUE';

BEZEICHNUNG
1 Konstantin
2 Katharina