

# Standardisierung von Coverage-Mehrwertdiensten in der OGC

Prof. Dr. Peter Baumann

International University Bremen  
rasdaman GmbH

# Gliederung

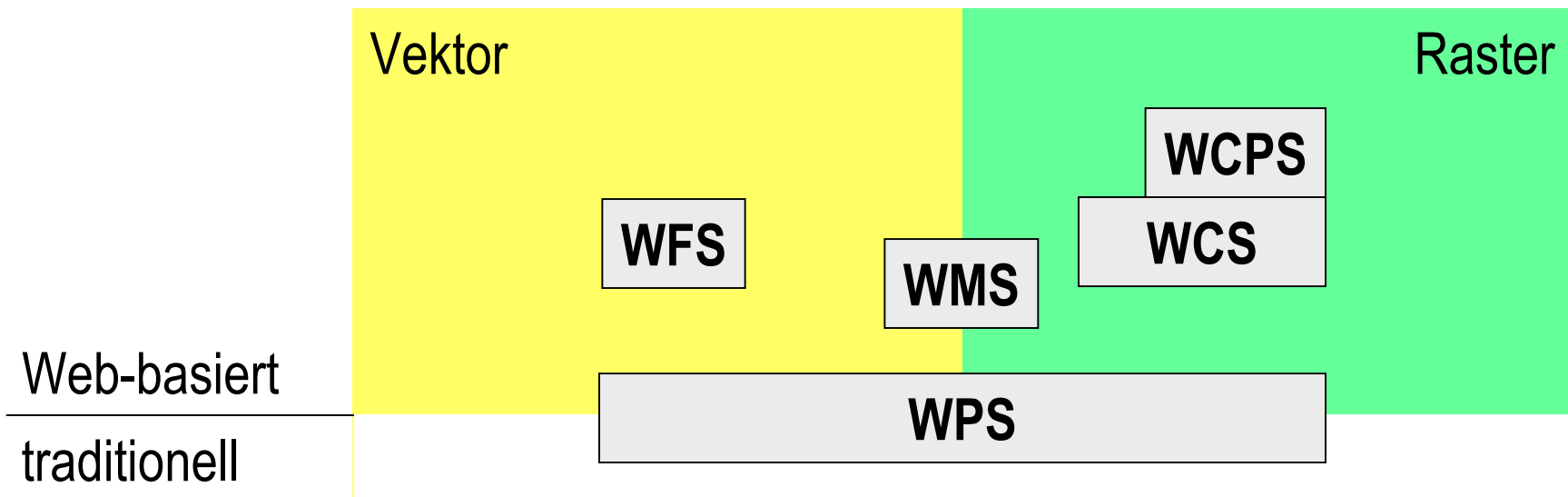
- Mehrwert-Dienste für Raster
- OGC: Web Coverage Service (WCS)
- OGC: Web Coverage Processing Service (WCPS)
- Implementierungsaspekte
- Status und Ausblick
  
- *Disclaimer:*  
*meine Ansichten, nicht notwendig die von OGC oder IUB*

# Mehrwert-Dienste für Raster



- Rasterdienste mehr und mehr Bestandteil von Geo-Diensten
  - ...nicht erst seit GoogleEarth
  - "2-D, 3-D image services next great wave in geo databases" - X. Lopez, Director Spatial, Oracle
  
- Wichtigstes Standardisierungsgremium für offene Geodienste:  
**Open GeoSpatial Consortium (OGC, [www.opengis.org](http://www.opengis.org))**
  - Liaison mit ISO, Oasis, IUGS, ...
  - Familie modularer Standards (i.w.: Service Implementation Specifications), einheitlich basierend auf OWS Common
  - Raster-relevant: WMS, **WCS**, **WCPS**, WPS
  
- Arbeitsgruppen für Rasterdienste:
  - WCS.RWG (Web Coverage Service Revision Working Group)
  - Coverage WG

# Querbeziehungen zwischen OGC-Stds



# Web Coverage Service (WCS)



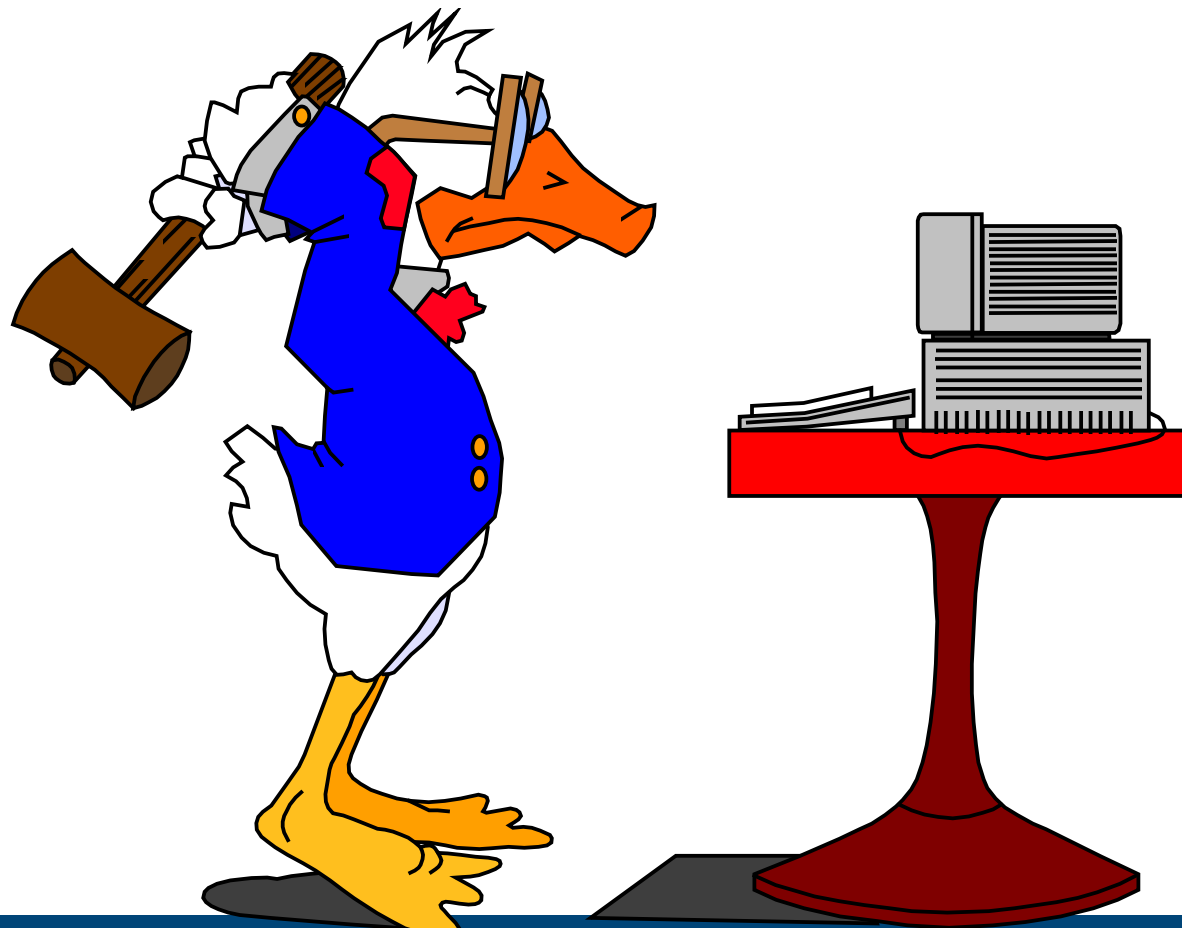
- **Coverage** = 2/3/4-D "space-varying phenomenon"
  - Prinzipiell sehr allgemein, praktisch: Raster
  - Array + ID + CRS + Nullwert(e) + Interpolationsmethode(n)
  
- **Abgrenzung:**
  - WMS "portrays spatial data to return static maps (rendered as **pictures**)"
  - WCS: "provides **data + descriptions**; rich syntax; returns data with its **original semantics** (instead of pictures) which may be interpreted, extrapolated, etc. - and not just portrayed."
  
- **Request-Struktur:**
  - **GetCapabilities:** Dienstbeschreibung + Übersicht über angebotene Coverages
  - **DescribeCoverage:** volle Coverage-Beschreibung
  - **GetCoverage:** liefert Coverage
  
- **Anfrageformulierung in KVP, XML**

# Aktuell: WCS 1.1



- Generell: einfachere Verwendung, Beseitigung von Inkonsistenzen
  - Klarere Dienstedefinition
  - Viele erläuternde Beispiele und Kommentare
  - Abarbeitungsmodell
  - Voreinstellungen ← umfassend vs einfach (range, CRS!)
  - Derzeit 64+120 Seiten, 9 Anhänge
- CRS Change Requests für OWS Common
- Range structure: Zelle ("Pixel") = n-D Array über atomaren Werten
  - Interpolation pro Zell-Komponente
- SOAP
- Coverage in Antwort oder serverseitig zwischengespeichert
- Hierarchisches Coverage-Modell
  - Derzeit keine Semantik hinterlegt → Applikations-Profile
- **Künftig** (v1.2+) angedacht u.a.:
  - Aktualisierungsoperationen; WSDL; Test suite spec

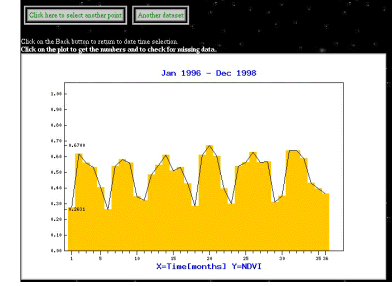
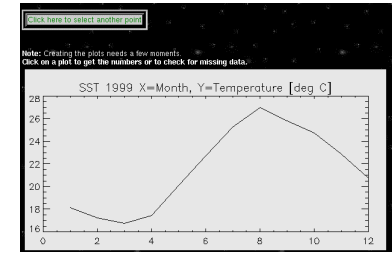
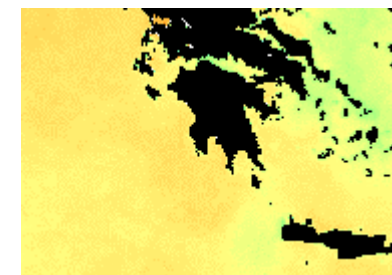
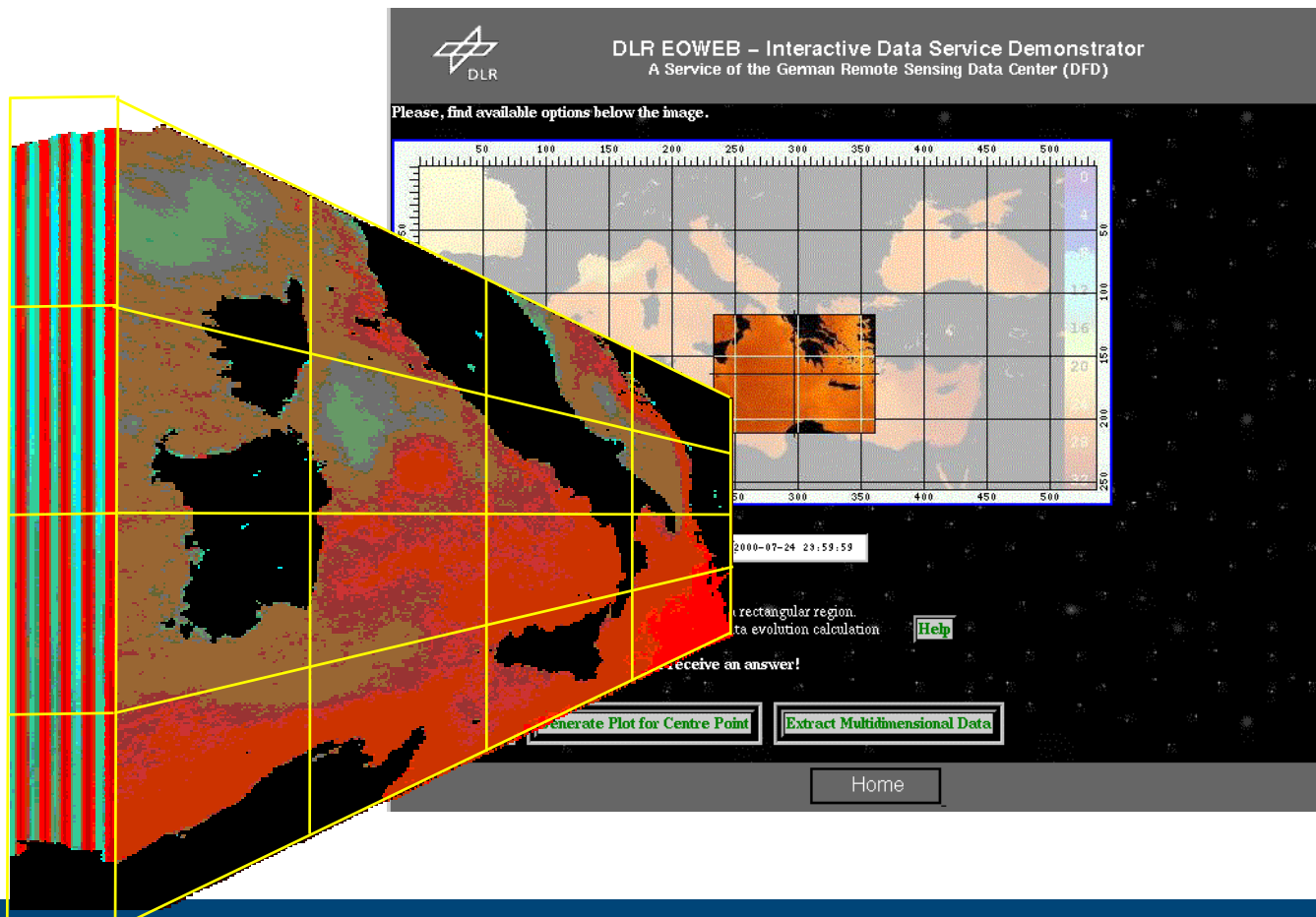
# Demo



# WCS-basierter 3D-Dienst



DFD: [eoweb.dlr.de](http://eoweb.dlr.de) [Diedrich et al 2001]





# Warum reicht WCS nicht?

- WCS Grundlage für Raster-Retrieval
- (1) weitere serverseitige Prozessierungsfunktionalität wünschenswert
  - WCS change requests
  - Beispiel: NDVI aus multi/hyperspektralen Bilddaten
- (2) ad-hoc Addition von Funktionen nicht zielführend
  - Unbegrenzte Liste – welche Funktionen wählen?
  - Kohärenz des Gesamtkonzepts?
- Ansatz: Raster-**Anfragesprache**
  - Funktionsschachtelung → unbeschränkte Komplexität
  - Serverseitige Optimierung möglich
  - Formale Semantik → Interoperabilität, kaskadierbar

# Raum-zeitliche Semantik

- **WCPS Coverage** = WCS Coverage
  - + explizite Achsen-Semantik
  
- Koordinaten-Adressierung auf zwei Ebenen möglich:
  - **Gitter-Koordinaten**: Zelladressen ganzzahlig, keine Achsen-Semantik
  - **raum-zeitliche Koordinaten**
    - *Achsen typisiert: x / y / t / elev / other*
    - *CRS / Geokoordinaten-Operationen* → x, y  
*Zeit-Operationen* → t

# WCPS: Operationen

## ■ Operationen *lesen*, *verknüpfen*, *liefern* Coverages

- ...oder beschreibende Info
- **Schachtelbare Ausdrücke**  
über Raster-Primitiven
- **Deklarativ**  
= was zu liefern, nicht wie zu berechnen
- **Sichere Auswertung** = alle Requests  
terminieren (keine Rekursion)

## ■ Operationen:

- allg. Raster-Op.
  - *Ausschnittsbildung*
  - *werte-ändernde Op.*
- Koordinaten-Trafo
- Format-Konverter
- Hilfsfunktionen

## ■ Mächtigkeit:

- WCS durch WCPS beschreibbar
- Statistische Operationen, Konvolution,  
Filter-Kerne, ...
- Keine Rekursion (zB Matrix-Inversion)

# WCPS: Request-Struktur

- **GetCapabilities:** i.w. WCS

- **ProcessCoverage**

- liefert 0..n Coverages (oder skalare Daten)

- Syntax: 

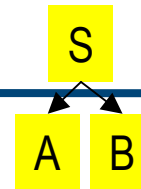
```
for var in ( coverageList )  
[ where condition(var) ]  
return processingExpr(var)
```

- Beispiele:

```
for c in ( A, B, C )  
return encode( c, "tiff" )
```

```
( tiff(A),  
  tiff(B),  
  tiff(C) )
```

```
for c in ( S )  
return max_cells( c )
```



```
( 83,  
  255 )
```

# Codierung

- WCPS **Abstract Syntax**: unabhängig von Codierung

- Bsp.: `encode( C.red + C.nir, "tiff" )`

- KVP: WCPS Abstract Syntax + Entities wo nötig

- Bsp.: `http://myhost/myservice ? SERVICE=WCPS & VERSION=0.0.1 & EXCEPTIONS=... REQUEST=ProcessCoverage & QUERY=encode(C.red%20+%20C.nir,"tiff")`

- XML : XML Schema

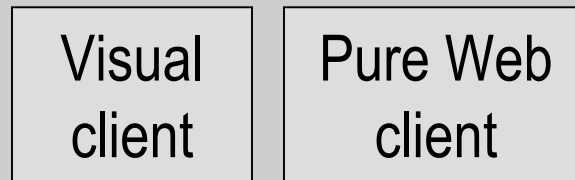
- Beispiel (verkürzt):

```
<encodedCoverageExpr format="tiff">  
  <binaryPlus>  
    <structSelection coverage=„C“ component=„red“/>  
    <structSelection coverage=„C“ component=„nir“/>  
  </binaryPlus>  
</encodedCoverageExpr>
```

# WCPS-Referenzimplementierung

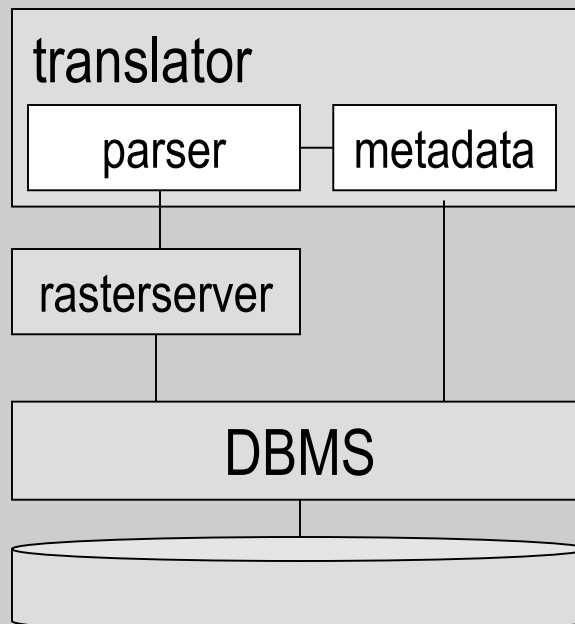


## WCPS client



Java 1.5;  
Swing, Jgraph

## WCPS server



Java 1.5;  
Tomcat, Xerces, JAXB

rasdaman

PostgreSQL

# Ex: Request-Umsetzung WCS → rasql

```
http://myServer/wcsServlet?  
  VERSION=1.0.0 & SERVICE=WCS & REQUEST=GetCoverage &  
  COVERAGE=AVHRRcube &  
  SRS=EPSG:31464 &  
  BBOX=4636000.0,5717000.0,4687000.0,5768000.0 & TIME=max &  
  WIDTH=246 & HEIGHT=300 & DEPTH=1 &  
  FORMAT=HDF-EOS &  
  EXCEPTIONS=application/vnd.ogc.se_xml
```

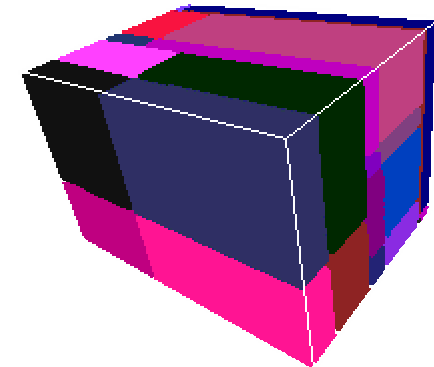
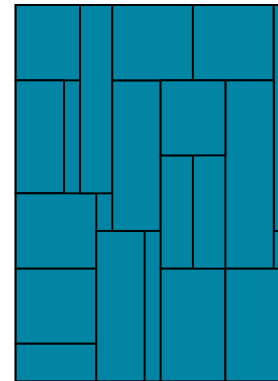


```
select hdf(scale(a[x0:x1,y0:y1,sdom(a)[2].hi],[246,300,1]))  
from AVHRRcube as a
```

- ...plus ggf. Reprojektion

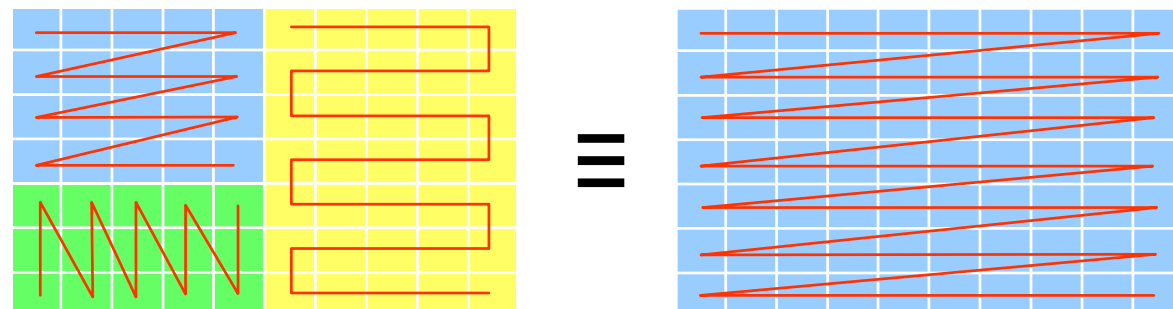
# Optimierung 1: Kachelung

- multidimensionales Objekt
  - Menge multidimensionaler **Kacheln**
    - *Optimale Partitionierung?*



- Kachelbasierte Anfragebearbeitung

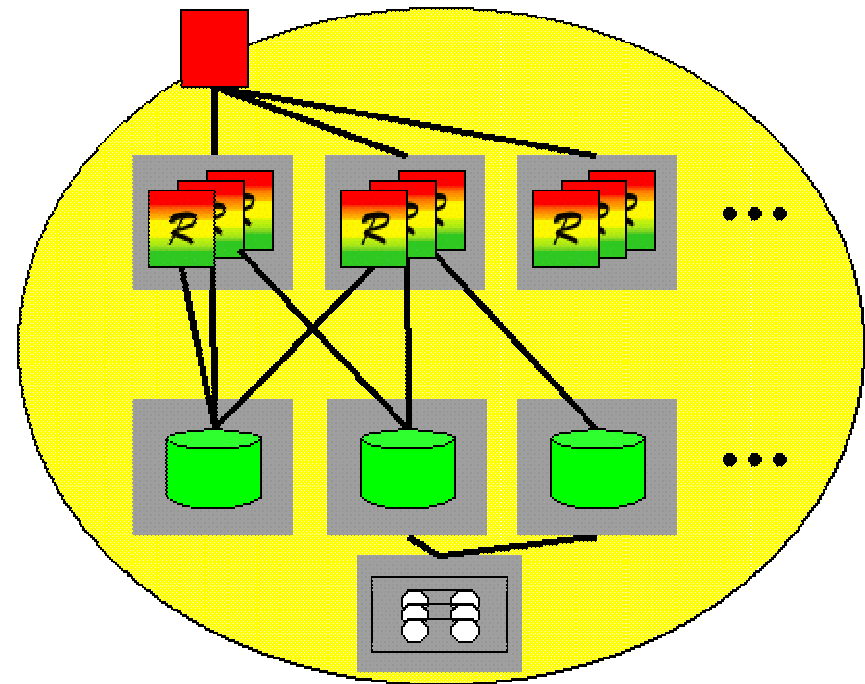
- *Optimale Auswertungsreihenfolge?*





# Optimierung 2: Parallelisierung

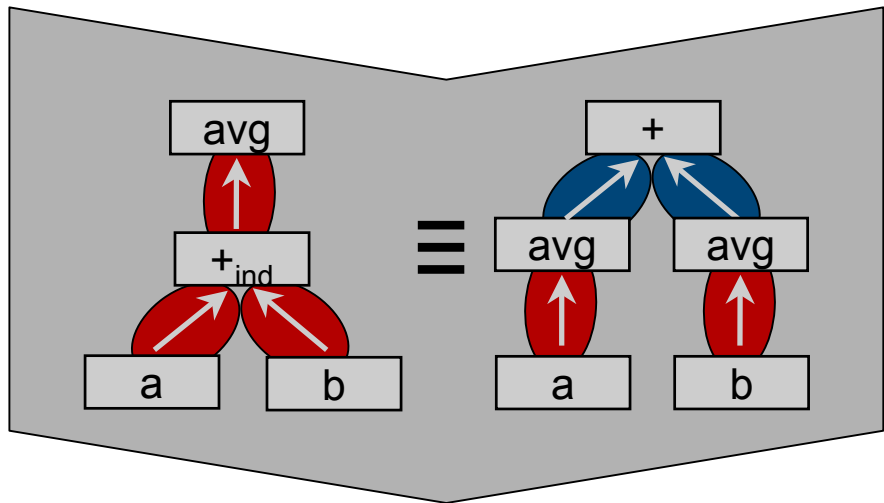
- *einfach:*  
*ein Client – ein Server-Prozess*
- *Nichttrivial:*  
*ein Client – mehrere Serverprozesse*  
*[Hahn 2003]*



# Optimierung 3: Anfrageoptimierung



```
select avg_cells( a + b )  
from a, b
```



```
select avg_cells( a )  
       + avg_cells( b )  
from a, b
```

-  Kachelstrom  
hoher Transport
-  Skalar-Strom  
geringer Transport

- *verstanden:*  
*heuristische Optimierung*
- *Nicht völlig gelöst:*  
*kostenbasierte Optimierung*

# Oracle 10g/11g

- GeoRaster
  - 2D Geo-Rasterbilder
  - Antwort auf ESRI ArcSDE 8
- Funktionalität:
  - Non-transparente Bildpyramiden
  - Ausschnittsbildung, Pixelkomponenten
  - Reprojektion?
  - Keine Optimierung erkennbar

```
declare
  g sdo_georaster;
  b blob;
begin
  select raster into g
  from uk_rasters
  where id = 4;
  dbms_lob.createTemporary(b,true);
  sdo_geor.getRasterSubset(
    georaster => g,
    pyramidlevel => 0,
    window =>
      sdo_number_array(0,0,699,899),
    bandnumbers => '0',
    rasterBlob => b);
end;
```

```
select g.green[0:699,0:899]
from uk_rasters as g
where oid(g) = 4
```

# Zusammenfassung und Ausblick



## ■ **WCPS = WCS + Raster ("Coverage")** Prozessierungssprache

- Geowissenschaften + darüber hinaus (z.B. CAD, Life Science)
- Flexible serverseitige Datenextraktion und –aufbereitung

## ■ Klare Semantikdefinition

- Erhöhte Interoperabilität
- Deklarativ, sichere Auswertung

## ■ Aktivitäten

- Verabschiedung WCS 1.1
- WCPS → Draft Standard (Dez 2006?)
- OGC's GALEON Netzwerk
- Komplettierung Implementierung WCS 1.1, WCPS
- Reale Applikationen

