



**TECHNISCHE  
UNIVERSITÄT  
DRESDEN**

Umweltwissenschaften, FR Geowissenschaften, Professur f. Geoinformationssysteme

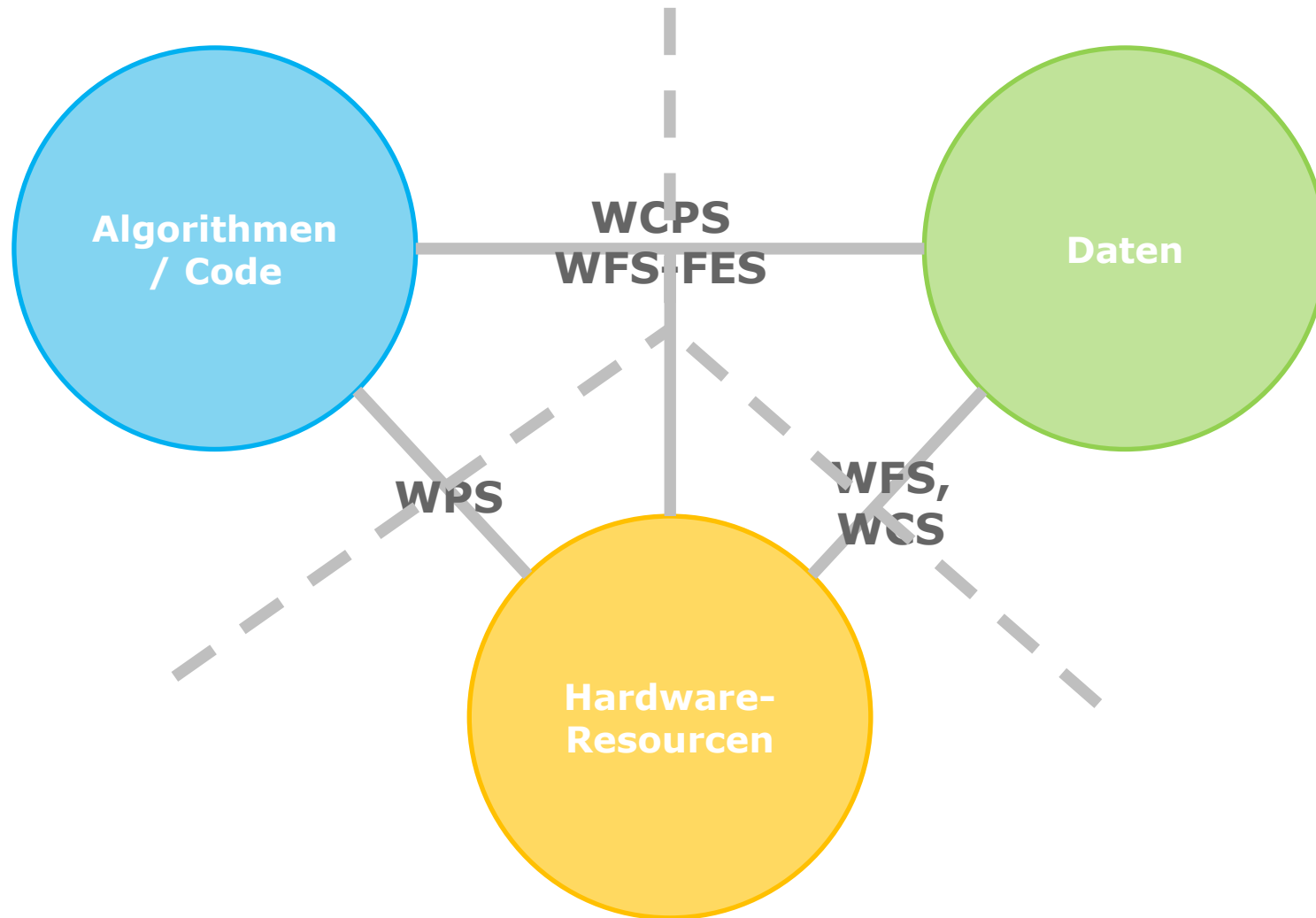
# **Geoprocessing Appstore – Ein Marktplatz für Prozessierungs- und Analysefunktionen**

Matthias Müller  
Karlsruhe, Mai 2014



**DRESDEN  
concept**  
Exzellenz aus  
Wissenschaft  
und Kultur

# Verteilte Geoprozessierung - Klassifikation



# Defizite WPS in gängigen Szenarien

- Kopplung von Funktionalität und Ressourcen
- Moving Code vs Moving Data
  - Bandbreite
  - Sensible Daten
  - Overhead
- Ausfallsicherheit (stabile Netzverbindung)
- Enorme Menge denkbarer Prozesse und Datenformate

# Defizite WPS in gängigen Szenarien

- Unklare Granularität
  - Mannigfaltige Funktionalität
    - ETL / Konverter
    - Klass. GIS-Toolboxes
    - Spezialisierte Simulationsmodelle
  - Mannigfaltige Formate
- ➔ Enorme Menge denkbarer Prozesse und Datenformate:
- Skaliert hier der reine WPS-Ansatz noch?  
(Weiterentwicklung, Pflege, Dienstwartung / QoS)

# Moving-Code – ursprüngliche Motivation

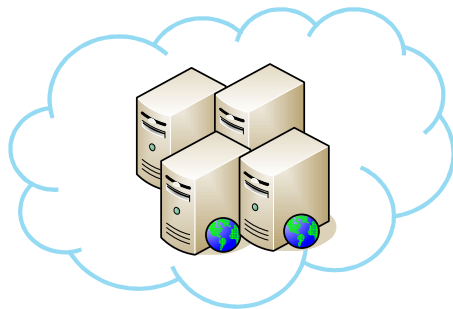
- Stellt das WPS-Konzept auf den Kopf um Prozesse näher zu den Daten bringen
  - Austauschen von Prozessen, nicht von Daten
  - Bandbreite sparen (Prozesse wandern zu den Daten)
  - Schutz sensibler Daten (hier ist keine Prozessierung auf klass. Web Services möglich)
  - Um in Umgebungen mit schlechter Konnektivität diensteorientiert zu prozessieren (interoperable Prozess-Schnittstellen)
- Entkopplung von Prozessierungsfunktionen und ausführenden Ressourcen
  - vmtl. ein Grund für die schlechte Verbreitung von WPS in der Praxis

# We like to move it, move it ...



Implementierte Algorithmen:

- wohldefiniert
- getestet
- nützlich für andere
- robust
- ... publikationswürdig



Cloud infrastructures



Workstation GIS



Web Service Backends

## Moving-Code – Aspekte

Functionality

Software  
environment

Hardware  
Requirements

Terms  
of use



# Moving-Code – Aspekte

Ausgangspunkt: Servicemodelle in Cloud-Infrastrukturen

- SaaS – Software as a Service → Algorithmische Funktionalität
- PaaS – Platform as a Service → Benötigte Softwareumgebung
- IaaS – Infrastructure as a Service → Benötigte Hardware
- Plus: Nutzungsrechte

Bestehende Standards:

|                     |  |
|---------------------|--|
| SaaS                | OGC WPS Standard: Process Descriptions           |
| PaaS                | (URI registry)                                   |
| IaaS                | OCCI (Open Cloud Computing Infrastructure Spec.) |
| Exploitation Rights | Creative Commons Modular license                 |



# Funktionale Beschreibung

- WPS-Prozessbeschreibungsmodell (ProcessDescription, ohne WPS-Protokoll)
  - Was sind die Prozessparameter?
  - Welche Bedeutung haben sie?
  - Sind es Inputs oder Outputs?
  - Mandatorisch oder optional, Kardinalität
  - Welches Datenformat wird benötigt? (z.B. GeoJSON, GeoTiff, ...)

# Plattform

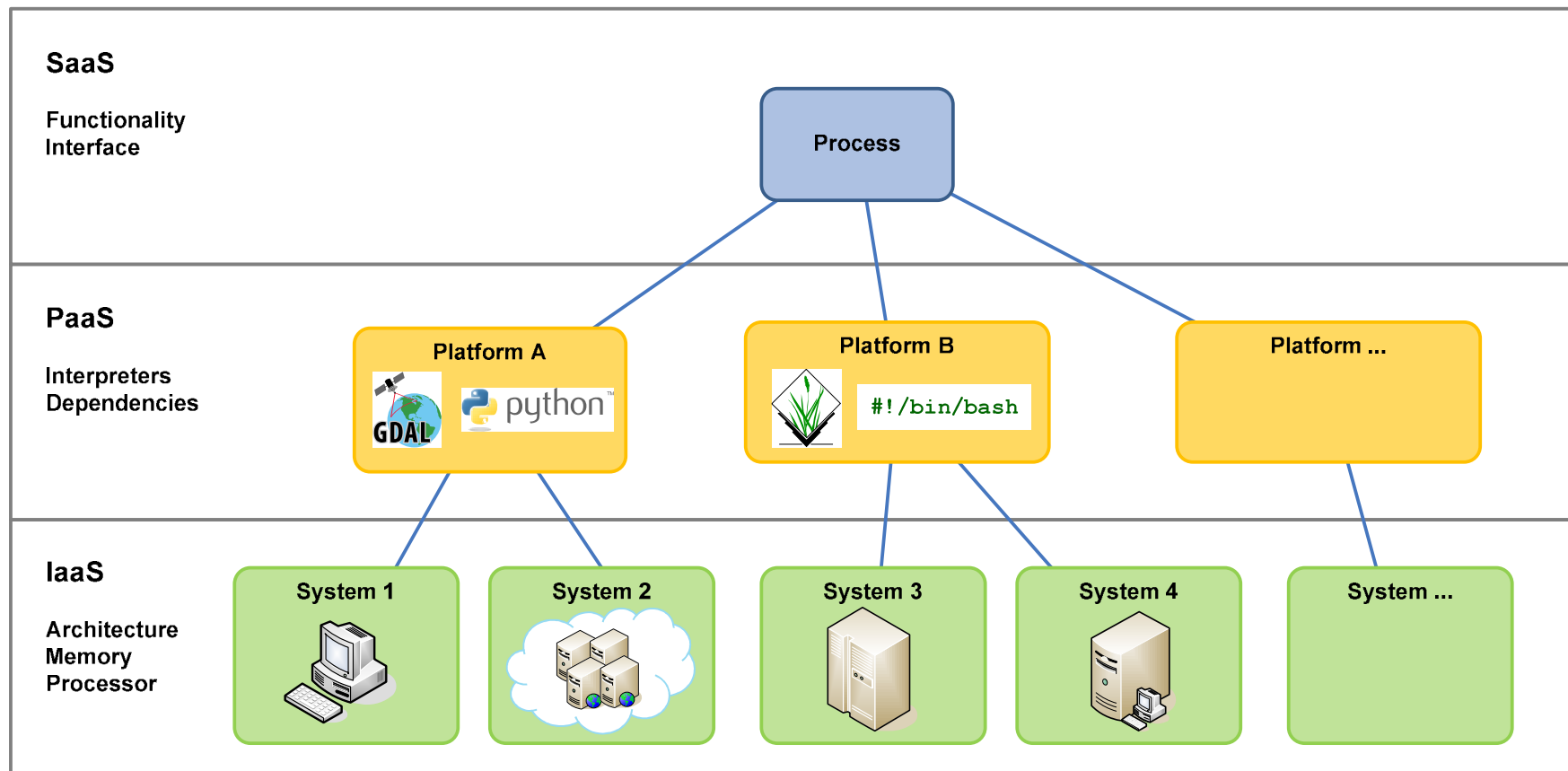
- Endliche Zahl an APIs / Softwarekomponenten im Geo-Bereich:
  - GDAL (which usually includes Numpy)
  - GRASS
  - ArcGIS
  - R
  - Matlab
  - ...
- 90% der Entwickler nutzen diese Schnittstellen für das Schreiben eigener Software
- PaaS kann über Identifikationsmechanismen entsprechende Laufzeitumgebungen bereitstellen

# Hardwareressourcen

- Serviceorientierung bedeutet hier die robuste Ausführung von Code unter erweiterten Randbedingungen (große Datenvolumen, viele Objekte in Datensätzen, ...)
- Voraussetzung für die Verarbeitung von "Big Data"
- Anpassung / Tuning / Testing für Verarbeitung großer Datenmengen ist keine triviale Aufgabe
- Hardware ("Contracted Infrastructure") beschreibt die notwendigen Hardwareressourcen zur Laufzeit
  - Im Idealfall: obere Grenzwerte
  - Heuristiken

# Hardware-Ressourcen

- Nach dem OCCI-Standard (Open Cloud Computing Initiative)
  - `occi.compute.architecture` {x86, x64}
  - `occi.compute.cores` [n]
  - `occi.compute.speed` [GHz]
  - `occi.compute.memory` [GB]
  - `occi.storage.size` [GB]



# Nutzungsrechte

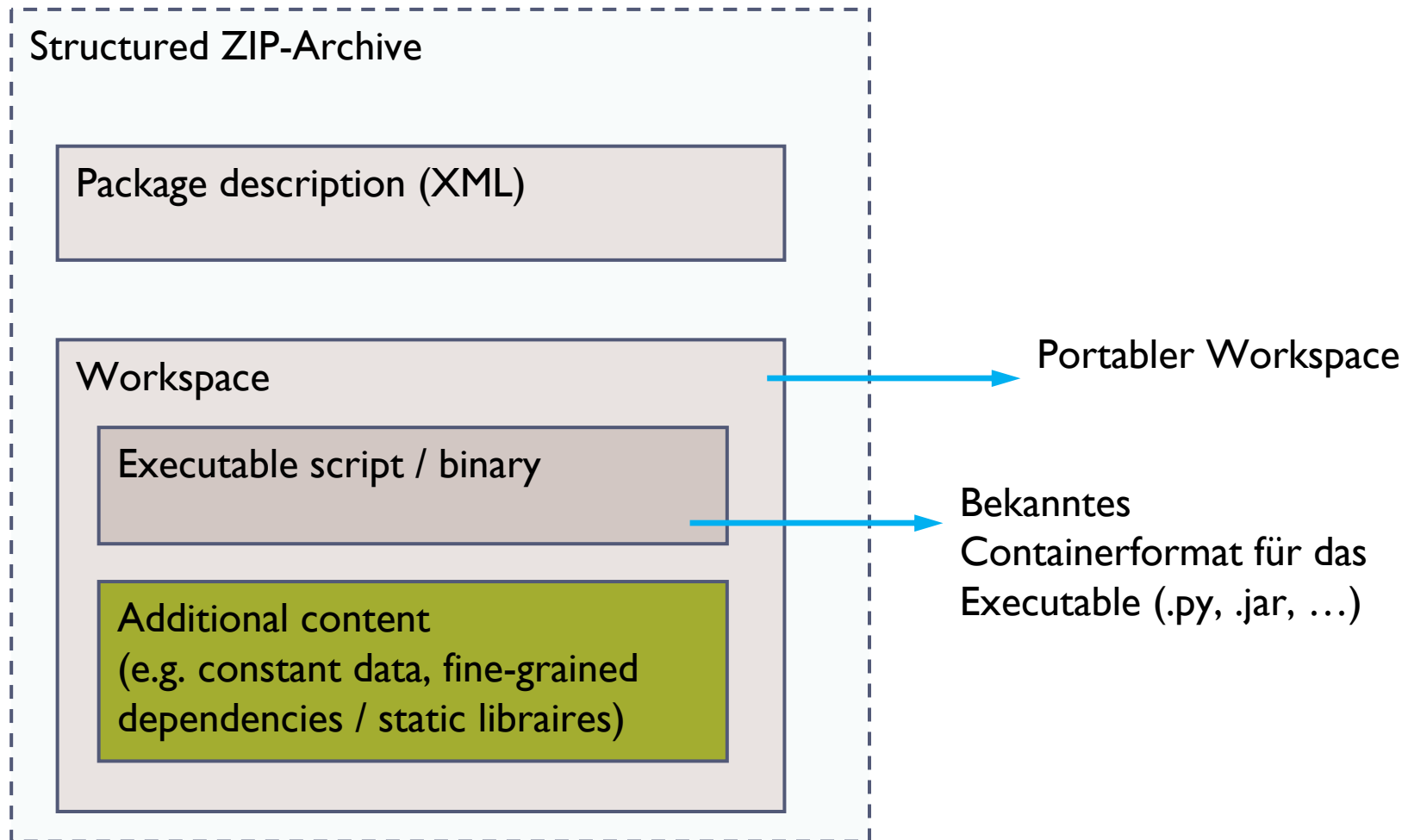
- Implementierte Algorithmen haben üblicherweise Nutzungsrechte / Lizenzen
- Creative Commons (CC) ist ein modulares, verbreitetes Lizenzmodell
- „Traditionelle“ Lizenzdokumente (GPL, BSD, Apache, MIT, ...)
- (Kommerziell?)

Drei "Schichten" der Lizenzen



<http://creativecommons.org/licenses/>


# Aufbau eines MovingCode Pakets



# Verteilungsmechanismus: Atom Feeds

**General Purpose Algorithm Repository**

Der aktuelle Feed enthält Inhalte, die häufig aktualisiert werden. Wenn Sie einen Feed abonnieren, wird dieser zu der gemeinsamen Feedliste hinzugefügt. Aktualisierte Inhalte des Feeds werden automatisch auf den Computer heruntergeladen und können in Internet Explorer und anderen Programmen angezeigt werden. [Weitere Informationen über Feeds](#)

 [Feed abonnieren](#)

## [de.tu-dresden.geo.gis.algorithms.raster.localmean](http://de.tu-dresden.geo.gis.algorithms.raster.localmean)

Donnerstag, 17. November 2011, 17:59:29

### computes the local mean for raster data (cell wise)

Computes the mean of all inputs cell by cell. Image coordinates, geo-references and projections of both rasters MUST match!

#### Parameters - Input


##### STATS\_RASTER - Statistics Rasters

collection of single band rasters to compute the mean from

#### Parameters - Output

##### LOCALMEAN - Local Mean Raster

a single band raster containing the local mean of the inputs

 [package.zip](#)

## [de.tu-dresden.geo.gis.algorithms.raster.ndvi](http://de.tu-dresden.geo.gis.algorithms.raster.ndvi)

Donnerstag, 17. November 2011, 17:59:37

### NDVI Computation

Momentan 4 / 4  
angezeigt:

• Alle 4

---

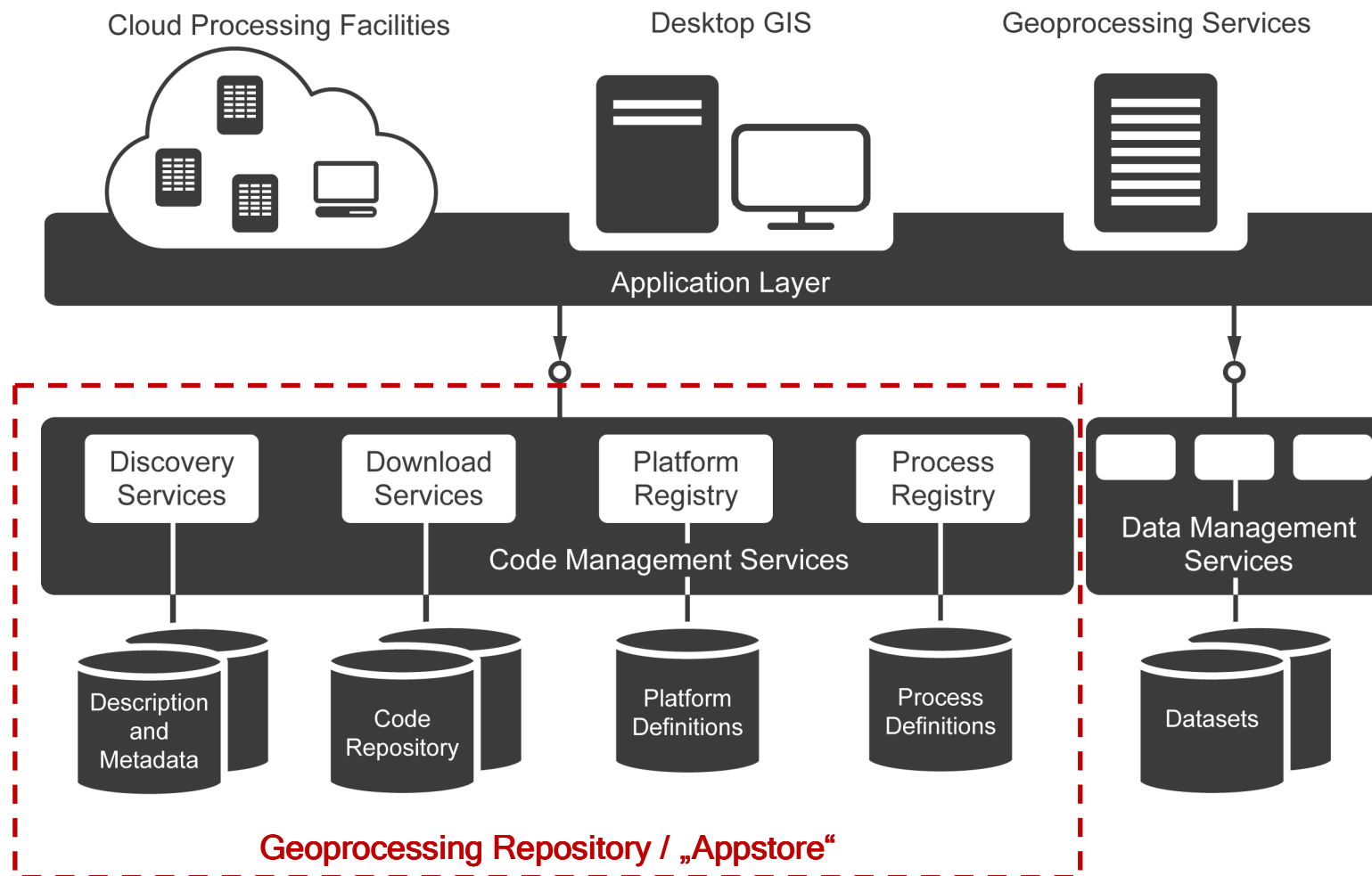
Sortieren nach:

▲ Datum  
Titel

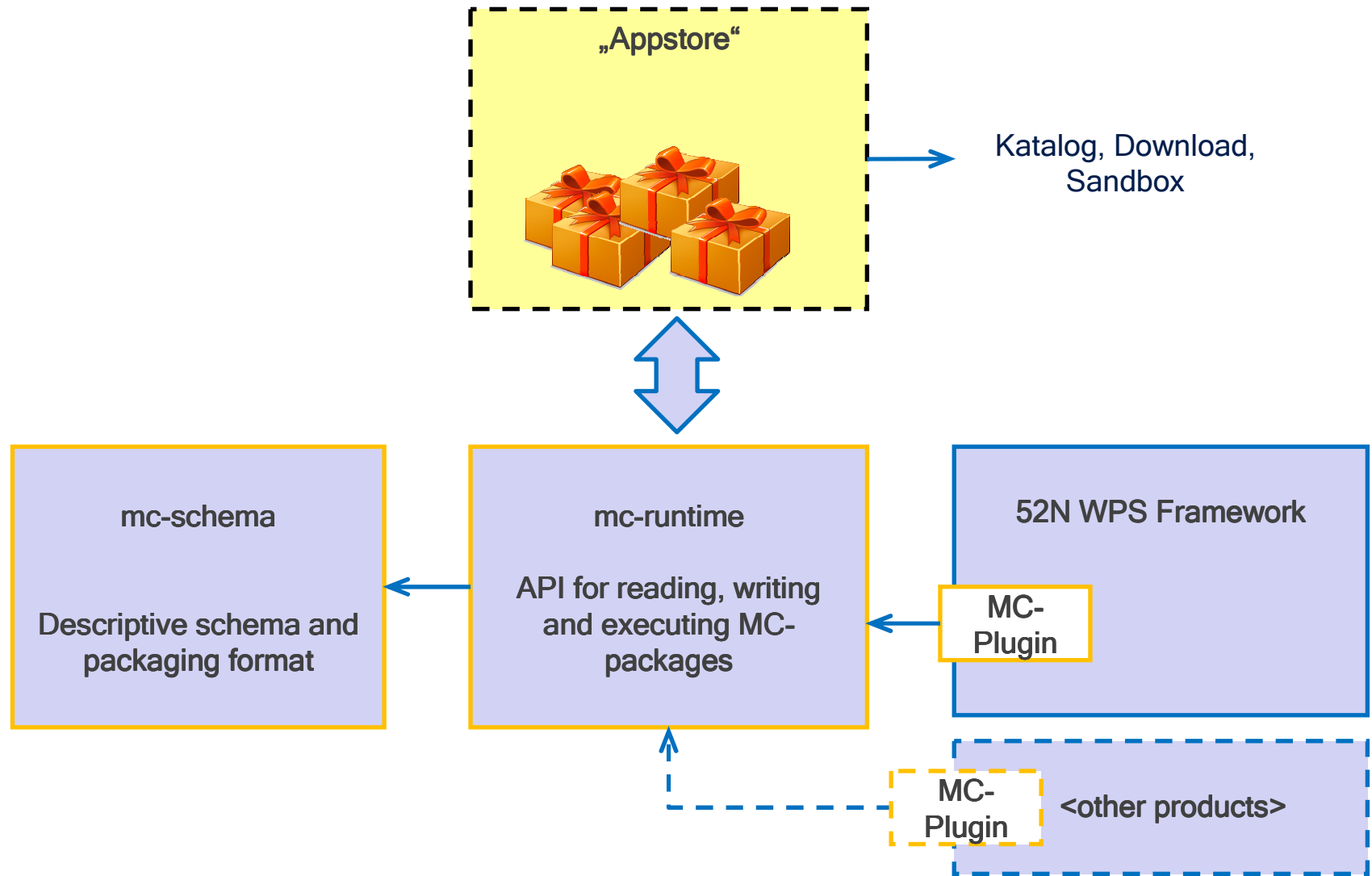
- ▶ Abonnierbar
- ▶ Human-readable
  - ▶ Web-Browser
  - ▶ RSS-Reader / Mailclient
- ▶ Machine-readable  
(über kleinere  
Protokollerweiterung)



# Architekturkonzept



# MovingCode Framework – Komponenten, automatisches Ausführen



# Moving-Code – künftige Anwendungsideen

- Basar für Geoprocessingfunktionen
  - „Source Forge“ für Entwickler im Bereich Geoprocessing
  - Appstore für potenzielle Nutzer
- Vereinheitlichung und Standardisierung von Prozessierungsfunktionen
  - Wohldefinierte Funktionalität in sinnvoller Granularität
  - Anwenderfreundliche Schnittstellen
  - Bessere Wiederverwendung einmal programmierter Verfahren und Algorithmen
- Nutzung identischer Prozessierungsfunktionen in unterschiedlichen Umgebungen

# In Zukunft: Ein Basar für Geoprocessing Tools?

