RIMapperWMS

An SVG Client built in OGC Web Mapping Service layers

Barend Köbben

International Institute for Geo-information Science and Earth Observation (ITC)

Department of Geo-information Processing

Overview

- **Why SVG for a Web Mapping Service?**
- **Why a built-in GUI?**
- **Past: Predecessor projects**
- **Present**
  - Principles
  - Technicalities
- **Future: Outlook**

What is a Web Mapping Service?

- A web service interface specification by the Open Geospatial Consortium (OGC)
- OGC delivers spatial interface specifications for Open Web Services (OWS) & related Encodings:
  - Geographic Markup Language (GML)
  - Web Catalog Service
  - Web Feature Service
  - Web Coverage Service
  - **Web Mapping Service**
  - Styled Layer Descriptor
  - Web Map Context Document

What is a Web Mapping Service?

"Standardized interface for the creation of super-imposed map-like views of geographic information"

- **Delivers map graphics from standardised URL requests**
- **WMS is actually the most mature and widest adopted OWS specification (numerous open source, as well as commercial solutions)**

Why Scalable Vector Graphics for a WMS?

SVG is XML-based vector graphics

- High quality (carto)graphics & attribute info
- low-bandwidth well suited for mobile applications

Many WMS exist, some with (limited) SVG

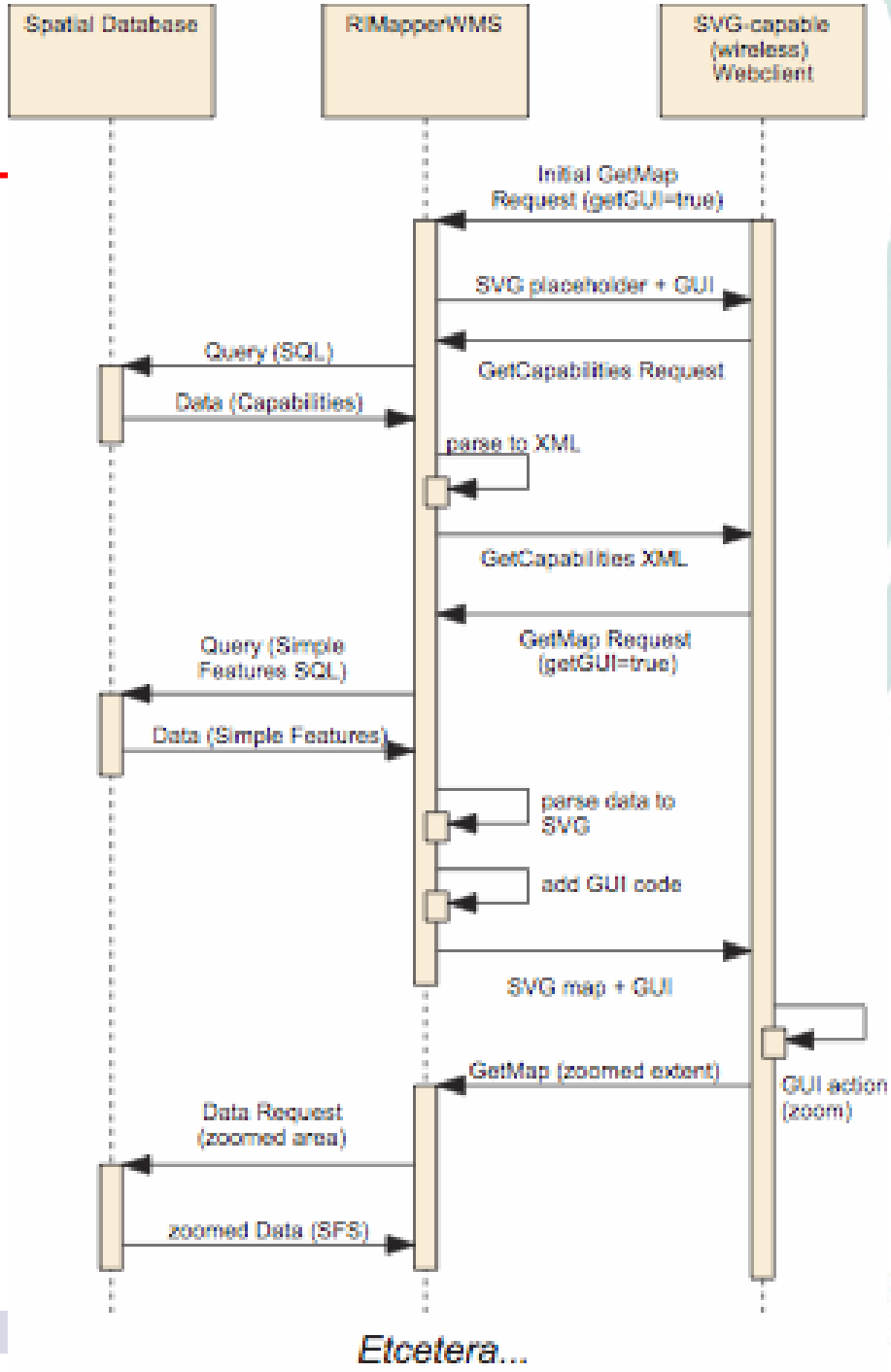
- All treat SVG as 'static graphics format' only
- SVG also can hold attribute data
- SVG also can provide animation
- SVG also can provide application logic

  ➔Can support built-in Graphical User Interface (GUI)

Why a built-in GUI?

No need for separate client application: "output = application"

- simple WMS conformant interface to the data
- data includes built-in client-side GUI
- GUI handles the map interaction and generates further requests



Past: Predecessor projects

RIMapperWMS has "organically grown" out of a range of earlier project at ITC:

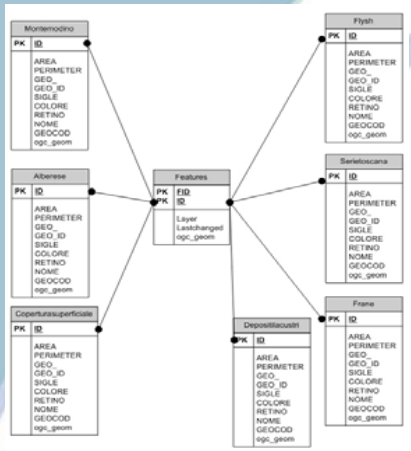
- RIMapper
- FLAVOUR (part of Wireless Campus LBS)
- Campusmapper

...all of these are under the umbrella of the SDI^{LIGHT} programme

SDI LIGHT

- Lightweight Spatial Data Infrastructure based on open standards/open source software
- testbed/playing ground at ITC
  - for research, PhD & MSc work
  - for projects & proof-of-concept applications
- server-side focus on MySQL/PostGIS, Java, open source OWS services
- client-side focus on SVG

RIMapper: Risk Inventory Mapper



Java servlets to deliver SVG output (=application)

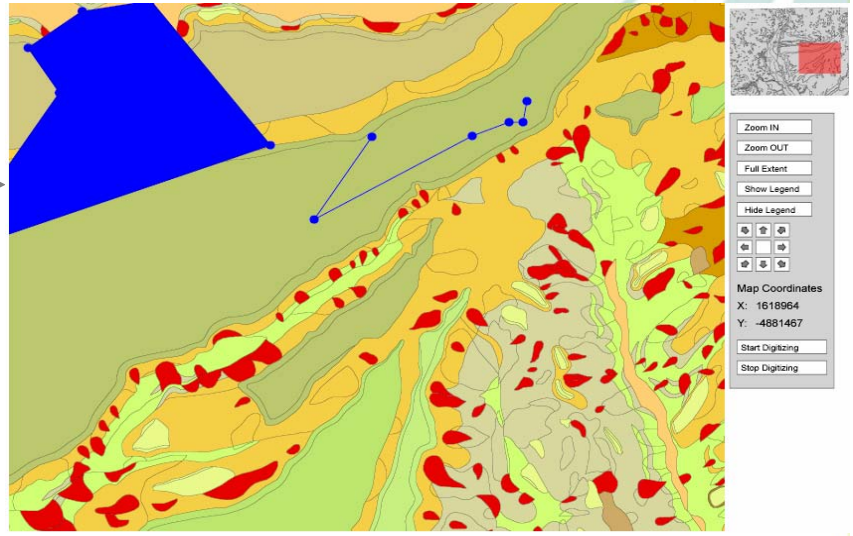
makeSVG

XML2SVG

parseXML

XML - configuration

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE RIM PUBLIC "" "/RIMapper/XML/RIM.dtd">
<RIM TYPE="SVG_STANDALONE" DB="rimapper" UN="un" PW="pw">
<HEADER>
    <FRAGMENT DBID="default" NAME="root" TYPE="SVG_ROOT"/>
    <STYLES>
    <STYLE DBID="default" NAME="defPoint" TYPE="CSS"/>
    <STYLE DBID="default" NAME="defLine" TYPE="CSS"/>
    <STYLE DBID="default" NAME="defArea" TYPE="CSS"/>
    </STYLES>
    <FRAGMENT DBID="default" NAME="init" TYPE="ECMASCRIPT"/>
    <FRAGMENT DBID="default" NAME="show" TYPE="ECMASCRIPT"/>
</HEADER>
<LAYERS>
  <LAYER DBID="default" NAME="ward" STYLETYPE="single"
         STYLE="defLine" />
  <LAYER DBID="default" NAME="river" STYLETYPE="single"
         STYLE="defArea" >
    <ACTION TYPE="simple" NAME="showRIM" SCOPE="feature"
         EVENT="onclick" PARAMS="evt, '', 'id'"/>
  </LAYER>
  <LAYER DBID="default" NAME="roads" STYLETYPE="single"
         STYLE="defArea" ATTRIBS="type" />
  <LAYER DBID="default" NAME="build" STYLETYPE="single"
         STYLE="defArea" />
</LAYERS>
<FOOTER/>
</RIM>
```



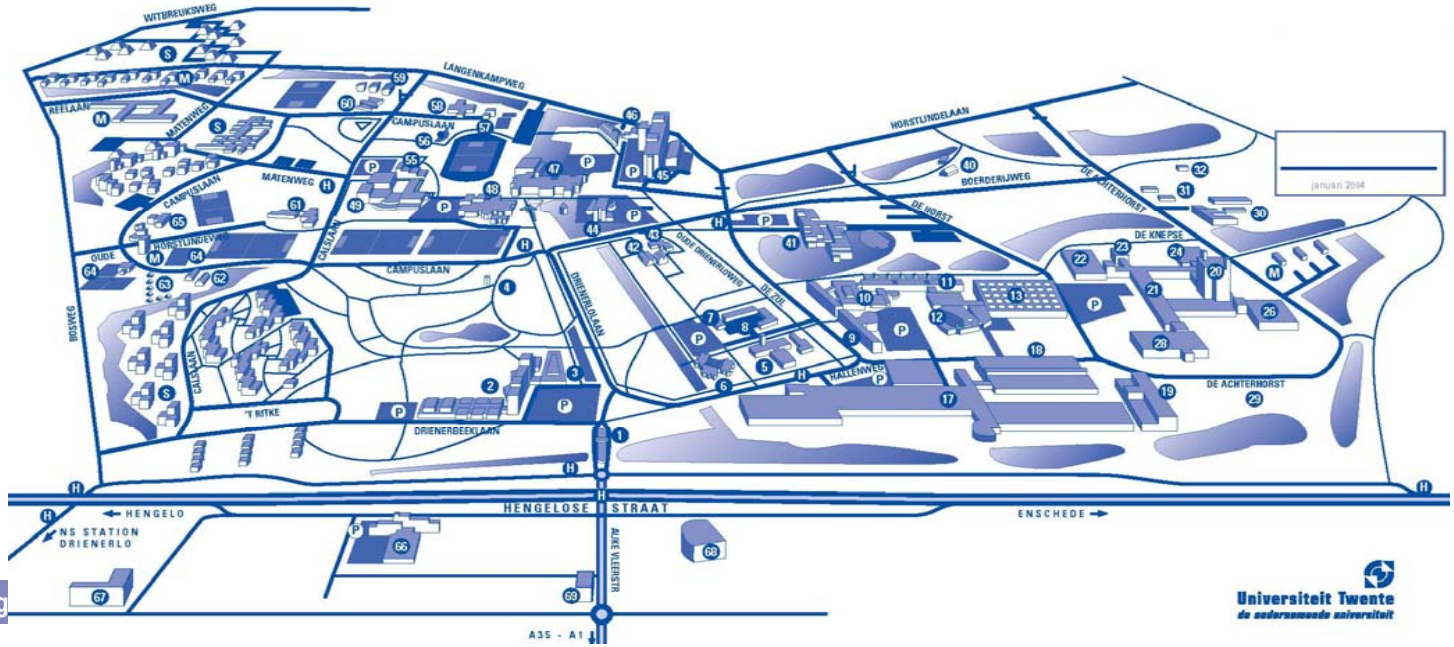
Wireless CampusLBS

- co-operation between ITC & University of Twente
- to set up *infrastructure* necessary for Campus Location Based Services, pilot at *SVGopen2005*

Europe's largest uniform hotspot

- 140 ha campus (covered in- and outdoors) + Enschede city centre (outdoors)
- 650+ individual access points

testbed for wireless and mobile applications

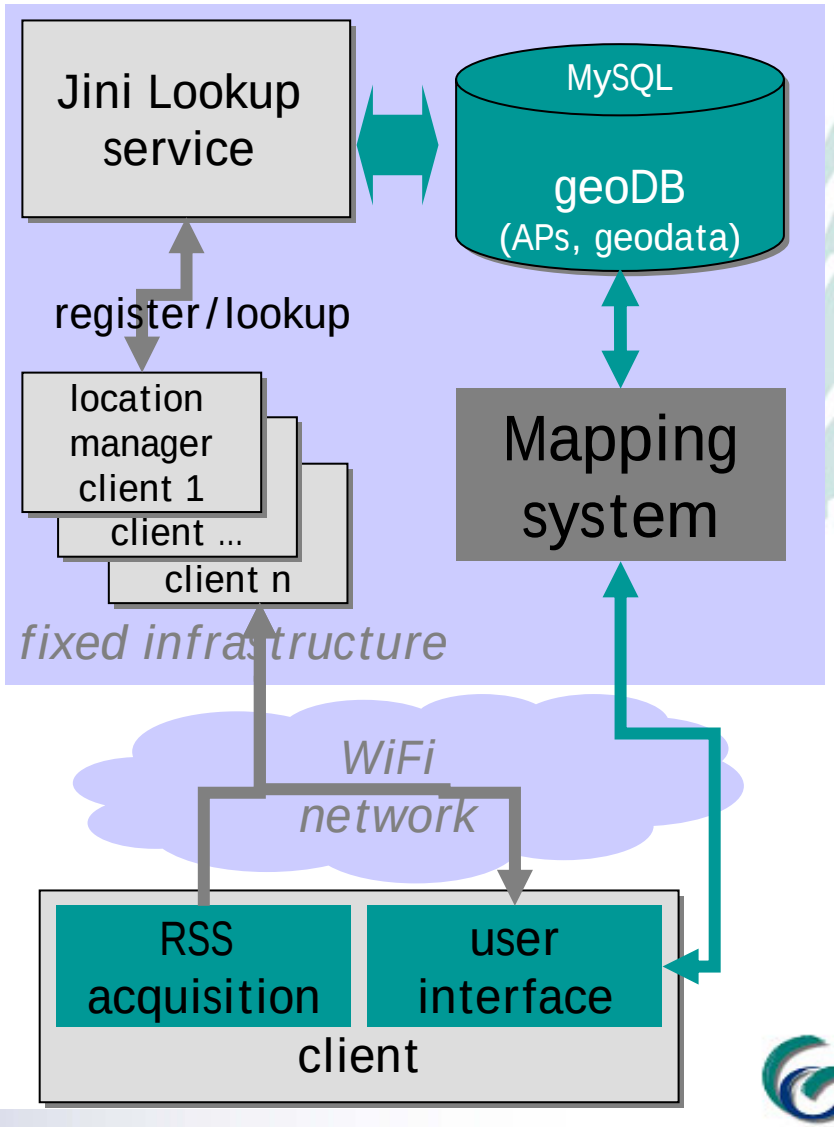


Universiteit Twente
de ondernemende universiteit

FLAVOUR prototype: architecture

Friendly Location-aware conference Assistant with priVacy Observant architectURe

- Location Managers
  - provide client with location
  - register with:
- Jini Lookup Services:
  - 'pull' (find others, locate resources)
  - 'push' (communicate with others, conference messages)
- Client application
- Mapping System based on RIMapper



From Flavour to CampusMapper

- **Flavour mapping system based on RIMapper with addition of extent-based feature extraction**
- **useful for more than Wifi localization:**
  - basis for quickly and easily customised maps of the UT Campus

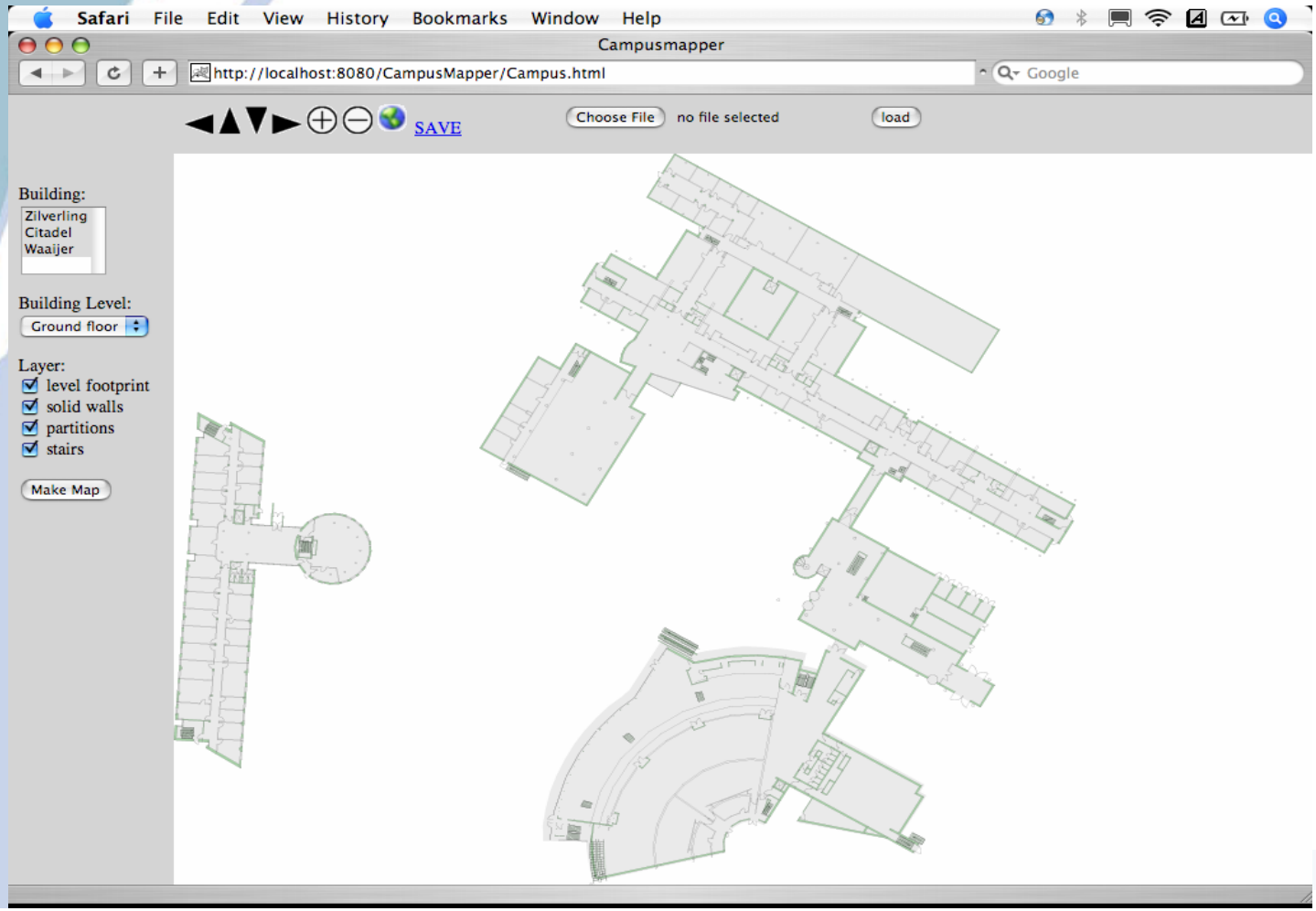
➔ **CampusMapper pilot**

  - DHTML interface generates GET/POST requests
  - JavaBeans store user/session settings

From CampusMapper to RIMapperWMS

CampusMapper already ‘almost’ an OGC WMS

- Only OGC compatible request/response missing

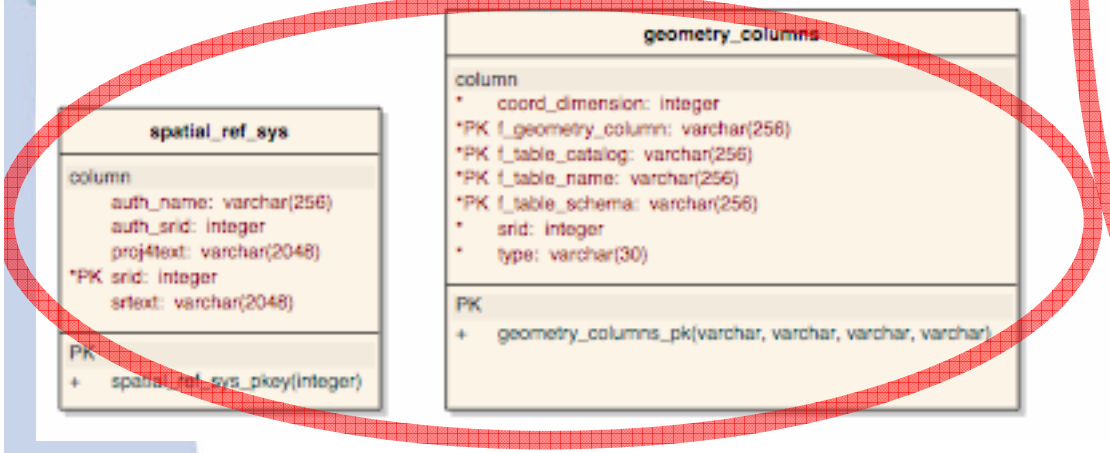


General setup of RIMapperWMS

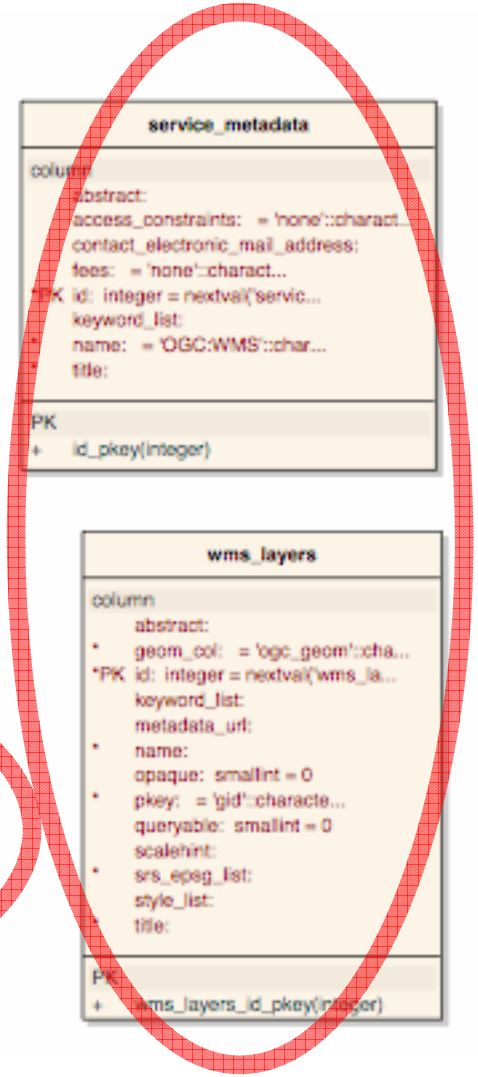
- spatial database back-end (postGIS)
  - spatial and attribute data
  - Web Mapping Service configuration
- server application (Java)
  - responds to WMS compliant requests
  - provides output in SVG (with built-in GUI)
- mobile or desktop web client
  - renders interactive & dynamic SVG maps

spatial database back-end (PostGIS)

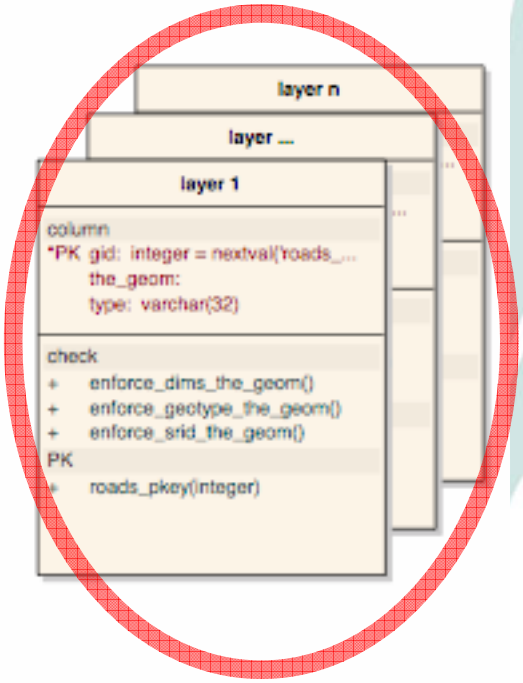
WMS styling



PostGIS spatial metadata



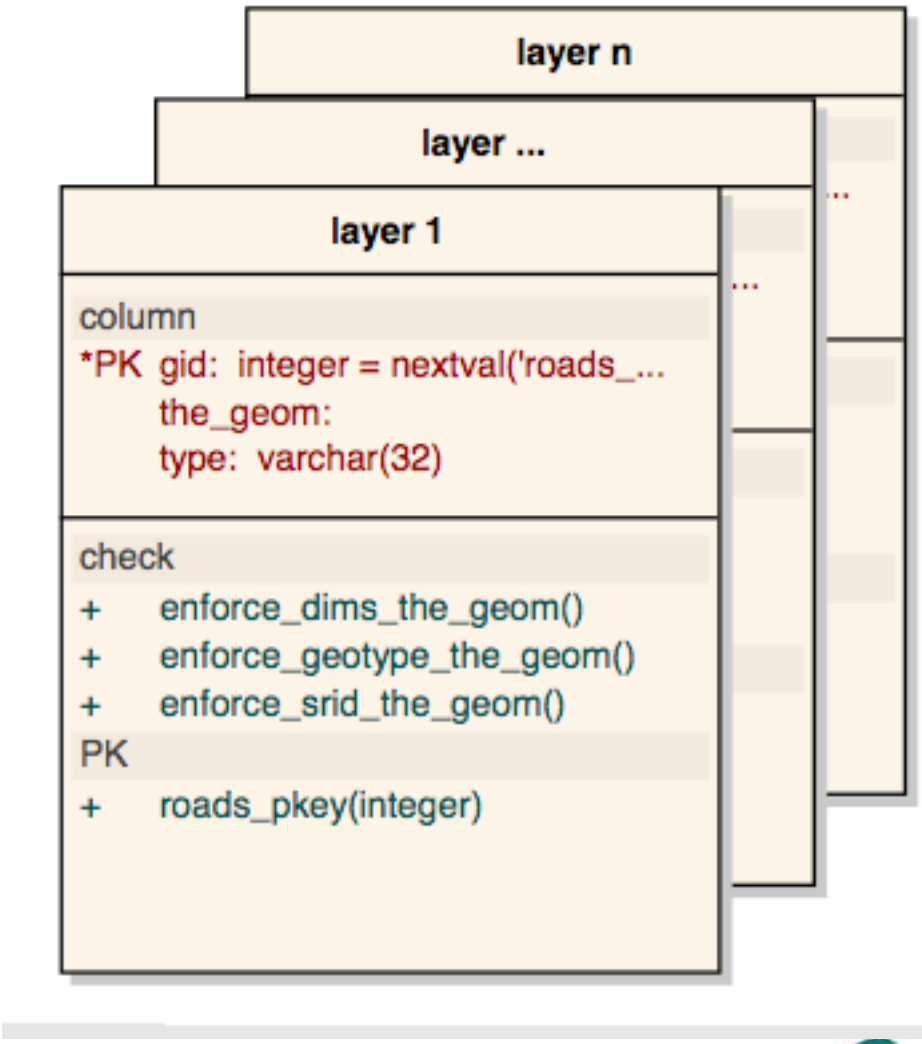
WMS metadata



spatial & attribute data per 'layer'

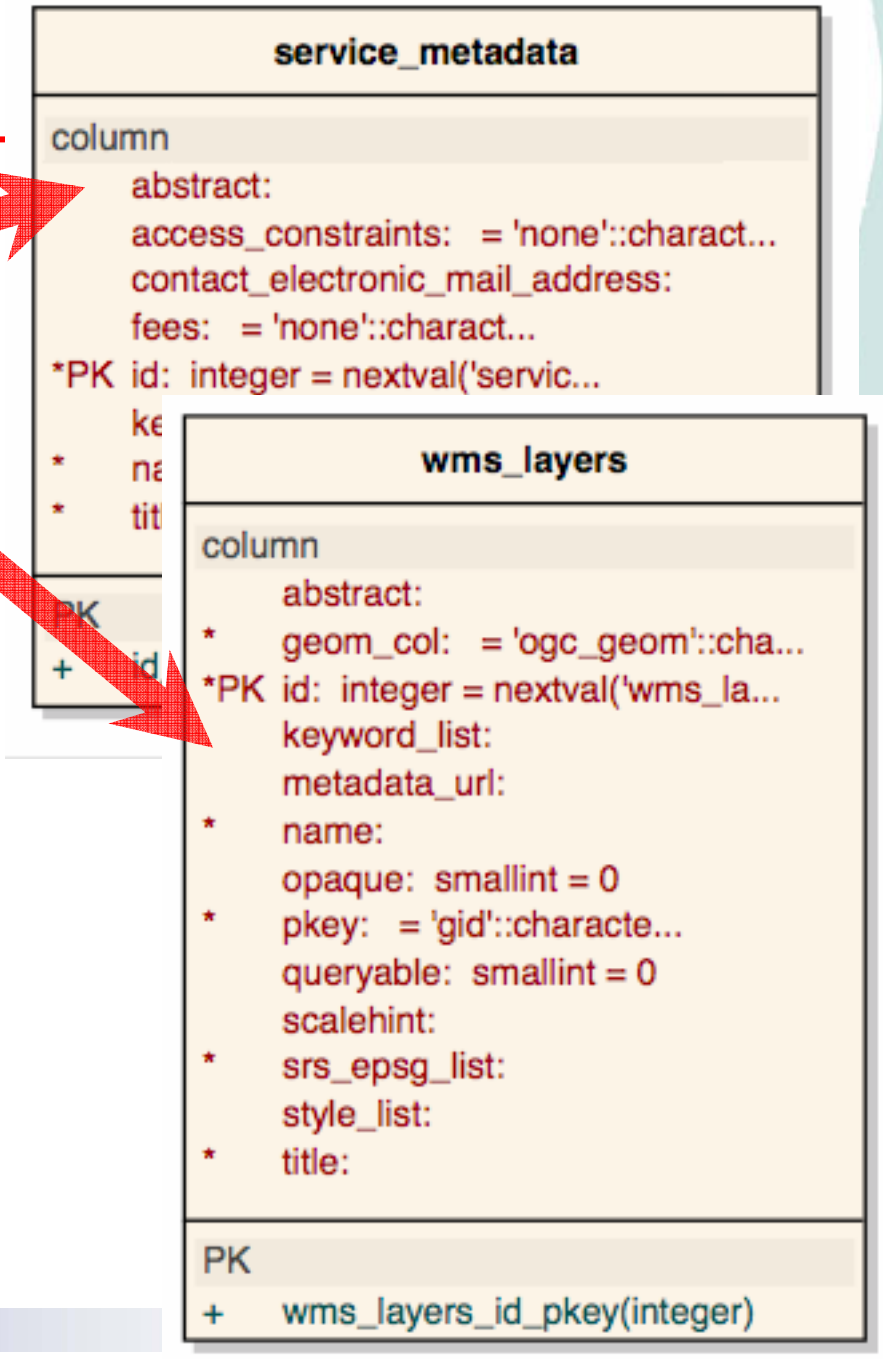
Spatial data layer tables

- Object geometries in PostGIS GEOMETRY objects
  - follows OGC Simple Features Specification
  - spatially indexed
  - (re-)projectable
- Object attributes
- Can come from many data sources (eg. shp2pgsql)



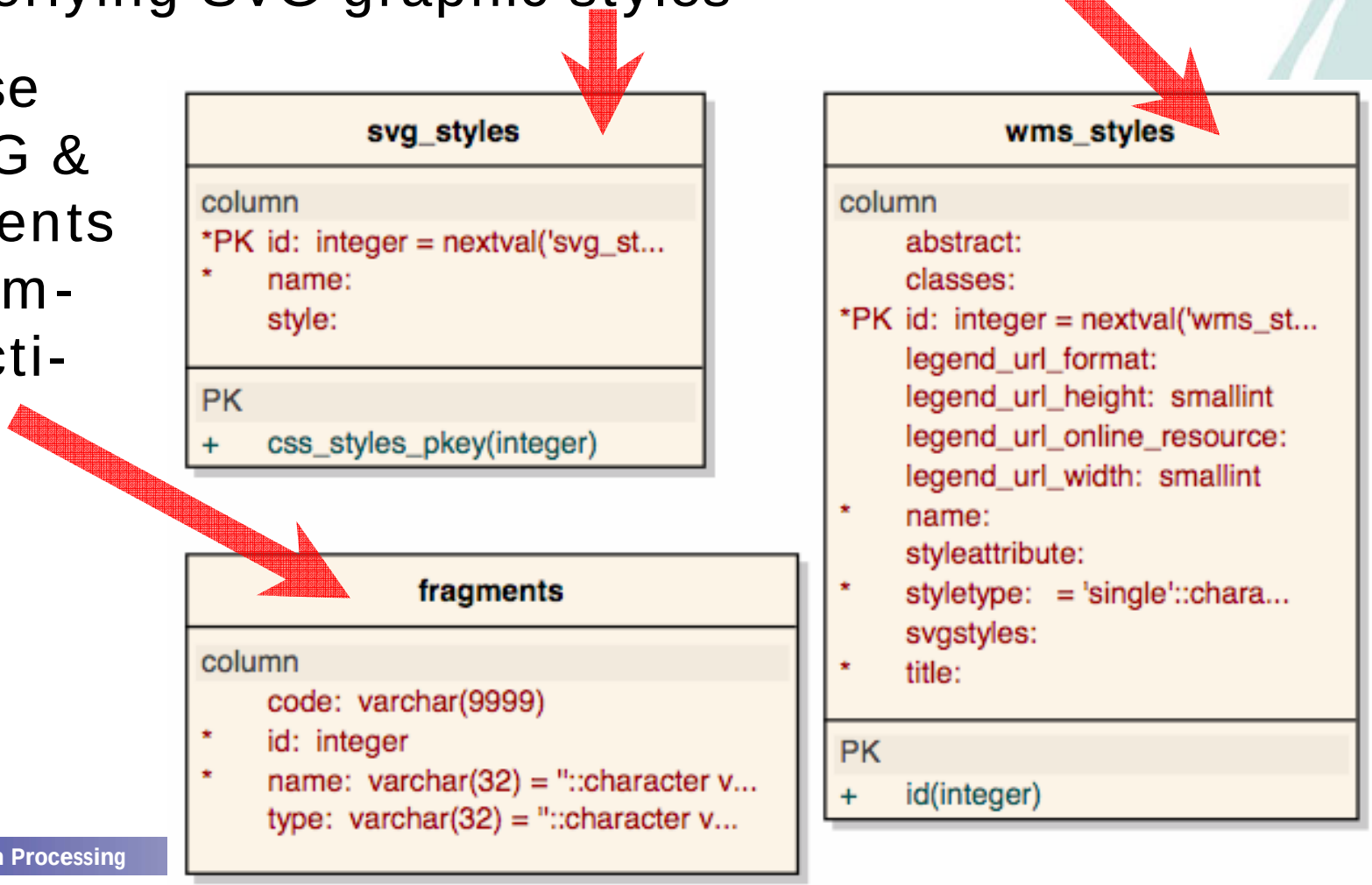
WMS metadata tables

- Defines the WMS instance metadata
- Lists available layers and their:
  - projection data
  - extent
  - styles
  - etc...



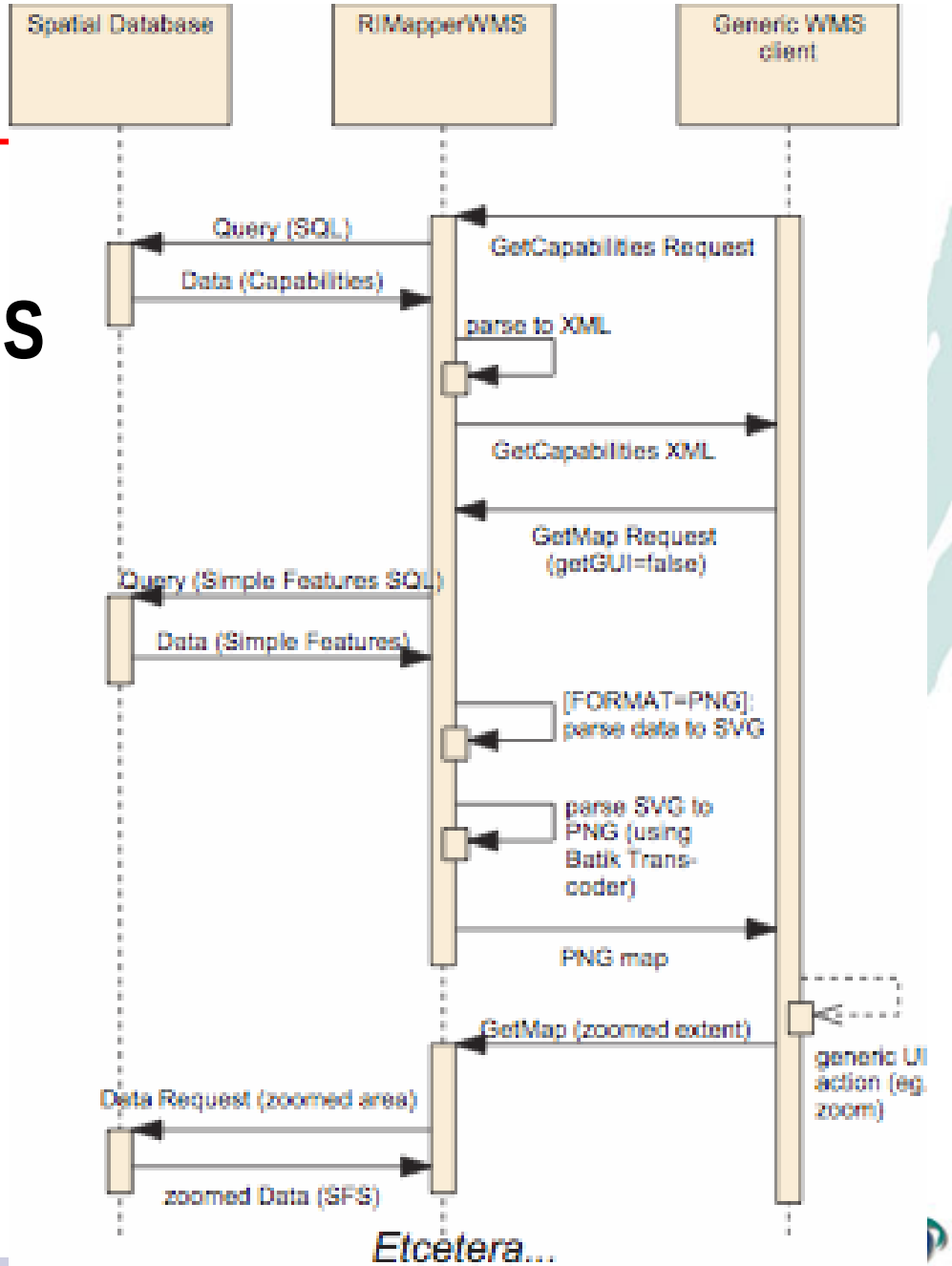
WMS styling tables

- Defines available styles from WMS perspective
- Defines underlying SVG graphic styles
- Multi-purpose table for SVG & script fragments (eg. GUI elem-ents, interacti-vity event handlers, ...)



Interoperability considerations

- **GetGUI=true would break a cascading WMS**
  - Default GetGUI=false
- **Other output formats support needed**
  - At least GIF & PNG
  - planned through Batik transcoding



Status: first public bèta released

- Adheres to OGC WMS *Basic* 1.1.1 specification
- Supports `GetCapabilities` & `GetMap` requests
- Additional vendor-specific `getGUI` capability
- Known limitations & issues:
  - GUI client very limited, need to make GUI more complete (layer switcher, attribute info, etc…) and more flexible (support more User Agents & SVG 1.2)
  - `getGUI=false` supported , but not yet output of formats other than SVG (PNG, GIF, etc…)
  - most OGC Compliance Tests pass, but no full compliance (ao. PNG or GIF output needed)
- Free, open source (*creative commons* license)

Outlook

Immediate plans:

- extending to *Queryable* WMS compliance
  - already possible to see attributes (client-side)
  - add server-side support: `GetFeatureInfo` interface
- WMS setup application for Database
- adding transcoding to other formats (PNG, GIF,...)
- performance & useability testing

and further...?

- WMS 1.3.0 support (depends on Proj4 library)
- Styled Layer Descriptor & Web Map Context
- ...?

Thank you for your attention!

<kobben@itc.nl>

http://kartoweb.itc.nl/RIMapper