Using the DBV model to maintain versions of multi-scale geospatial data

Laboratory of Information Systems (LIS)

Institute of Computing – UNICAMP – Brazil

João Sávio C. Longo

Luís Theodoro O. Camargo

Claudia Bauzer Medeiros

André Santanchè

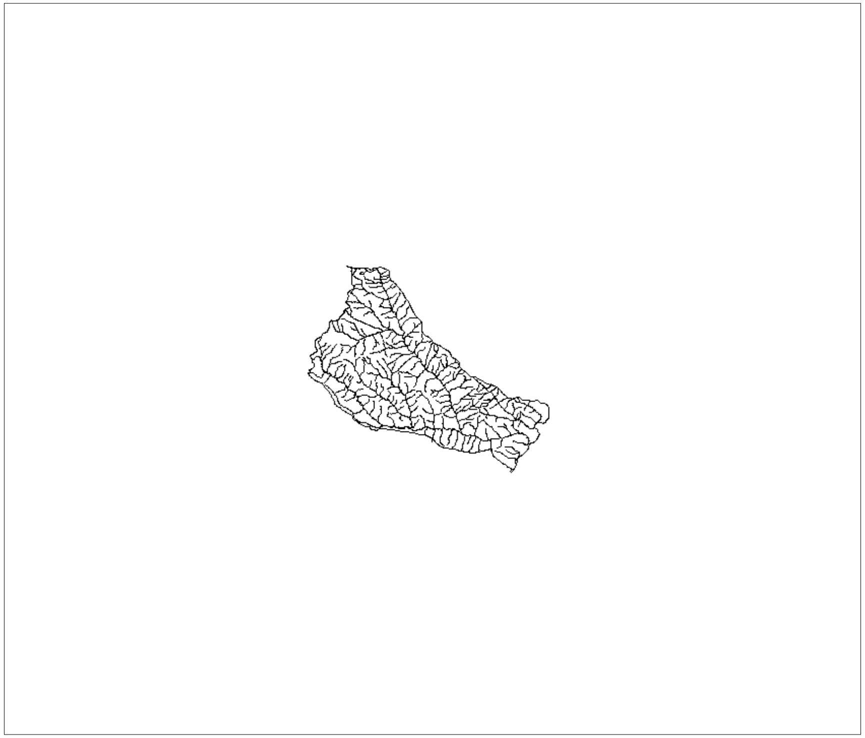
Outline

1. Motivation and goal
2. Central idea – extend the DBV model
3. Our approach
4. Conclusions and future work

Motivation and goal

Multi-scale data

Scale 1:250k



Scale 1:1M

Problems attacked

1) How to manage multi-scale data?
2) How to trace real world evolution through time?
3) How to represent alternative scenarios?

Current approaches

- Generalization algorithms (computation centric)
  - real time computation – e.g. agents
  - costly if there are continuous updates to the data
- Multi-representation databases (MRDB) (data centric)
  - data structures to store and link multi-representation data
  - costly if there are many scales to store
- Hybrid – store some scales, compute others
  - more flexible, but still lots of issues to solve

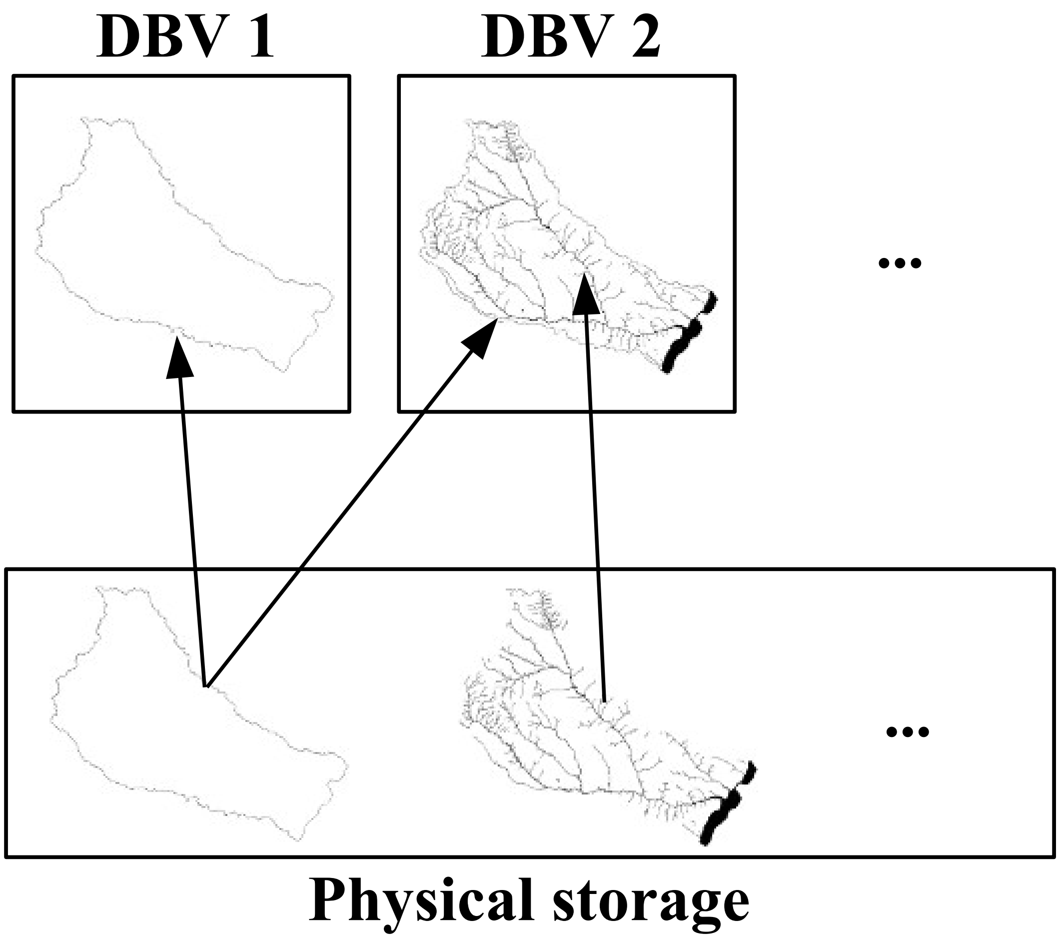
Goal

Propose a data-centric model that supports generalization plus MRDBs to:

- manage multi-scale data
- trace real world evolution
- represent alternative scenarios for a given scale

Central idea – extend the DBV model

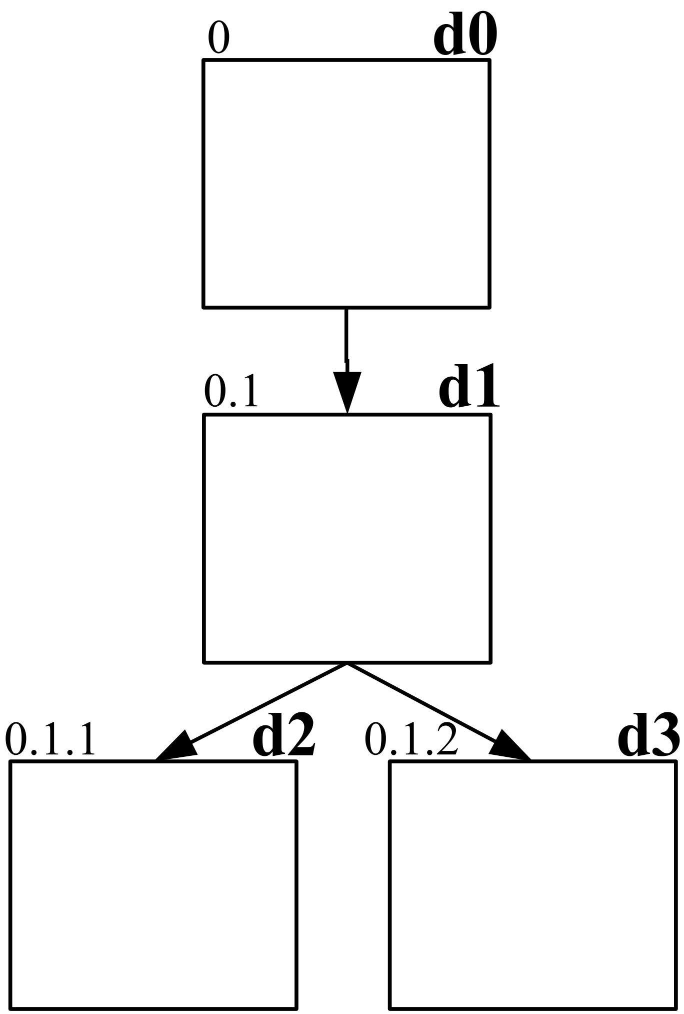
DBV model



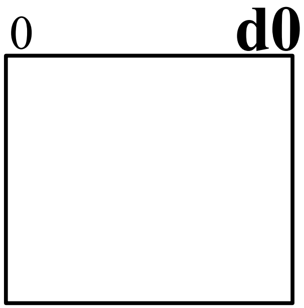
DBV model

- DBV (Database Version)
  - represents a possible state or version of the database
- Not only temporal versioning
  - any stored modification can be a version
- Save storage space
  - only the changes must be stored
  - unchanged data are shared from previous DBVs

Derivation tree

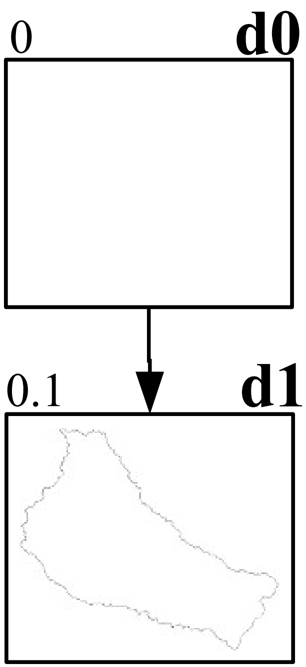


Example



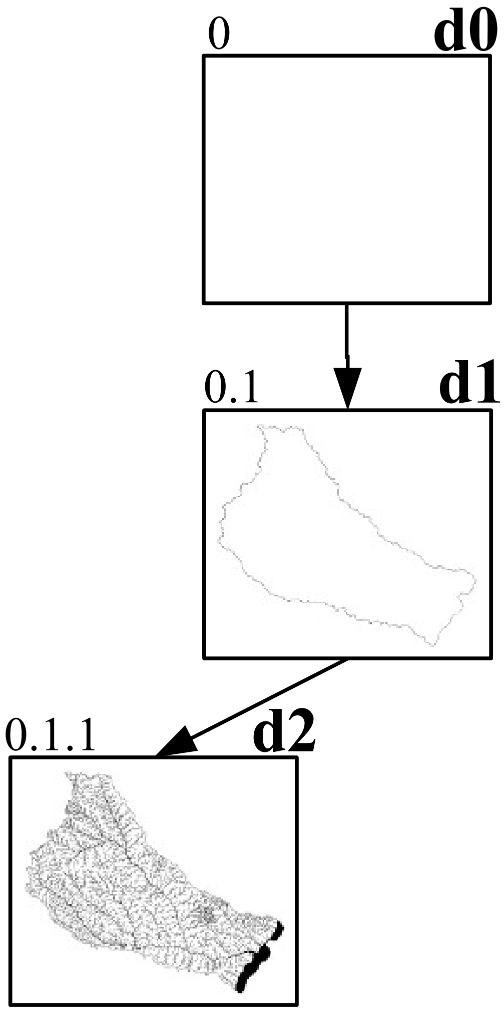
DBV	Data
d0	-

Example



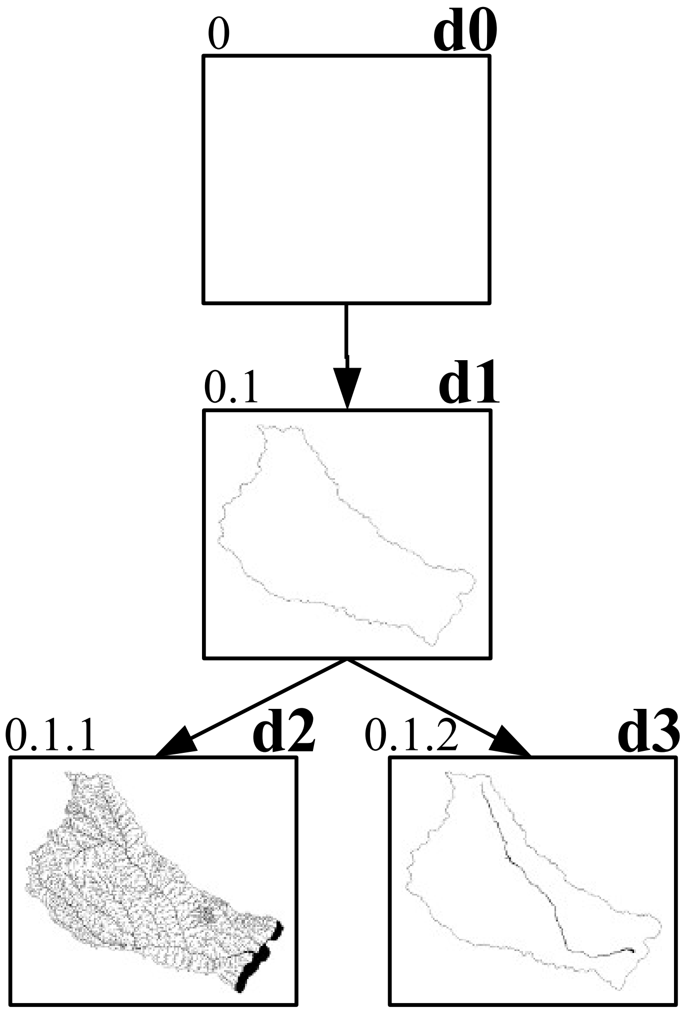
DBV	Data
d0	-
d1	- watershed polygons

Example



DBV	Data
d0	-
d1	- watershed polygons
d2	- watershed polygons - rivers

Example



DBV	Data
d0	-
d1	- watershed polygons
d2	- watershed polygons - rivers
d3	- watershed polygons - main river

Our approach

Overview

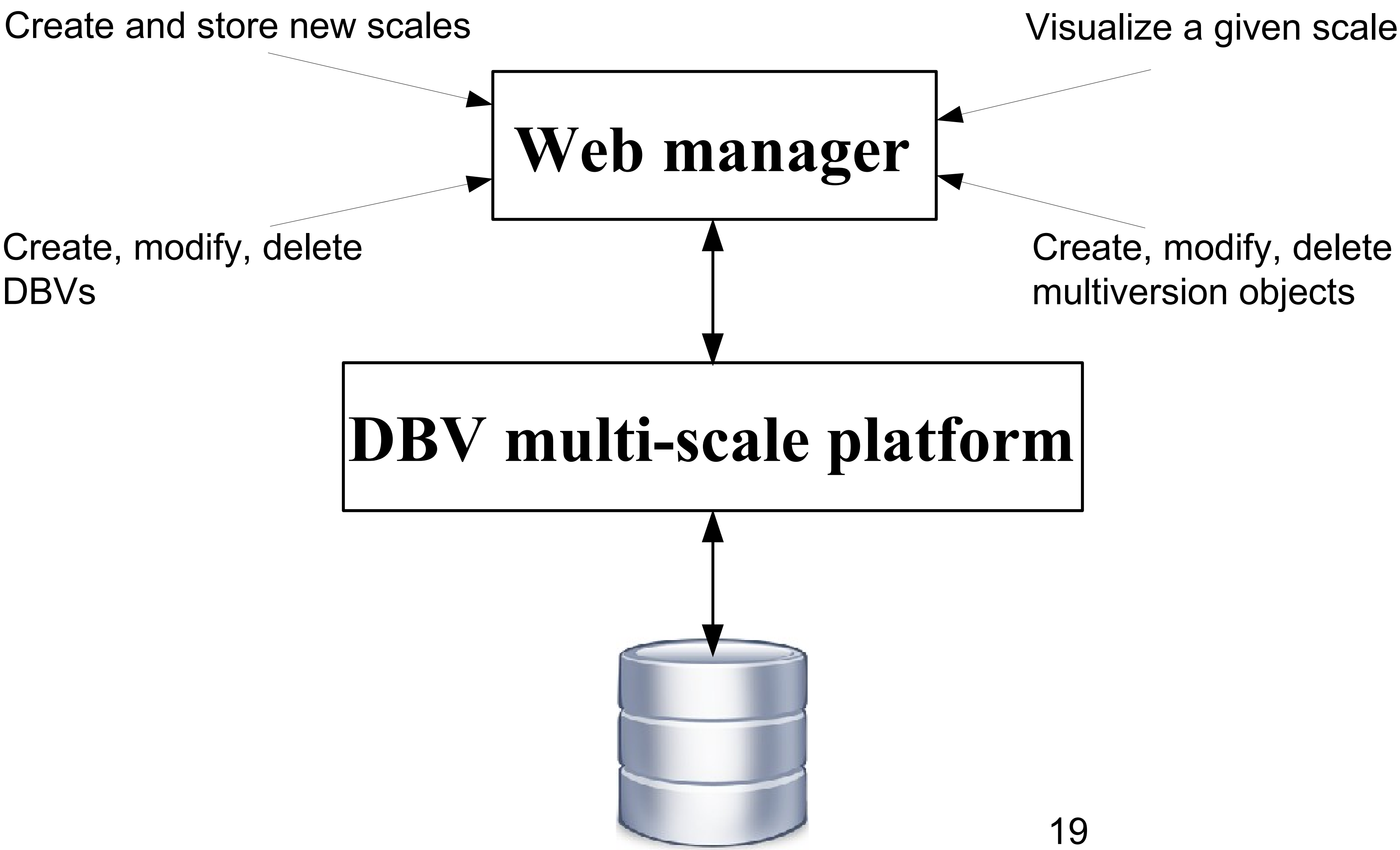
It is an extension of the DBV model, in which:

- each scale has its own derivation tree
- all trees grow and shrink together
- scenario – set of DBVs with same version stamp

DBV multi-scale platform

- Open source
- http://code.google.com/p/dbv-ms-api
  - plataform
  - sample project
  - web manager

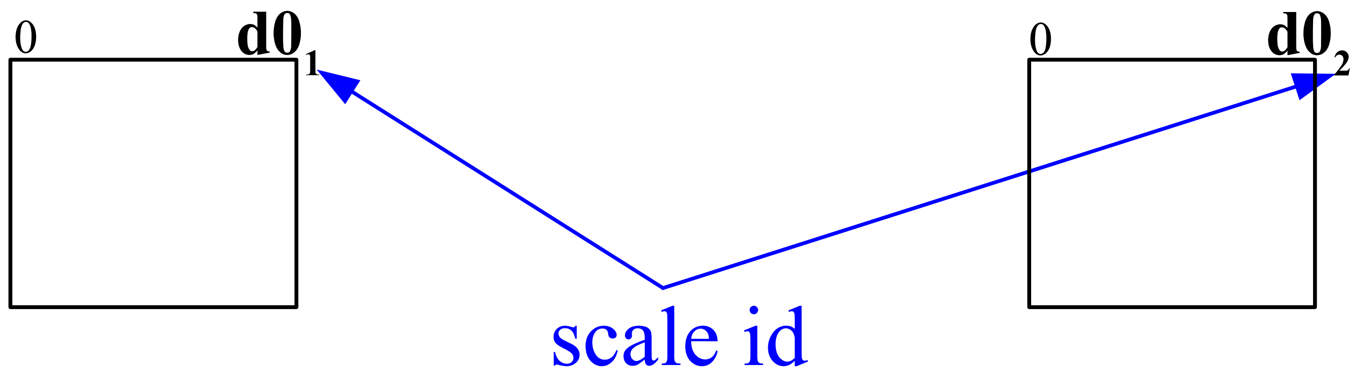
General architecture



Case study

- Watershed multi-scale data – Rio Pardo/Brazil
  - 1:250k
  - 1:1M
- Content – 5641 geometric objects:
  - watershed polygons
  - rivers

Building scenario 0

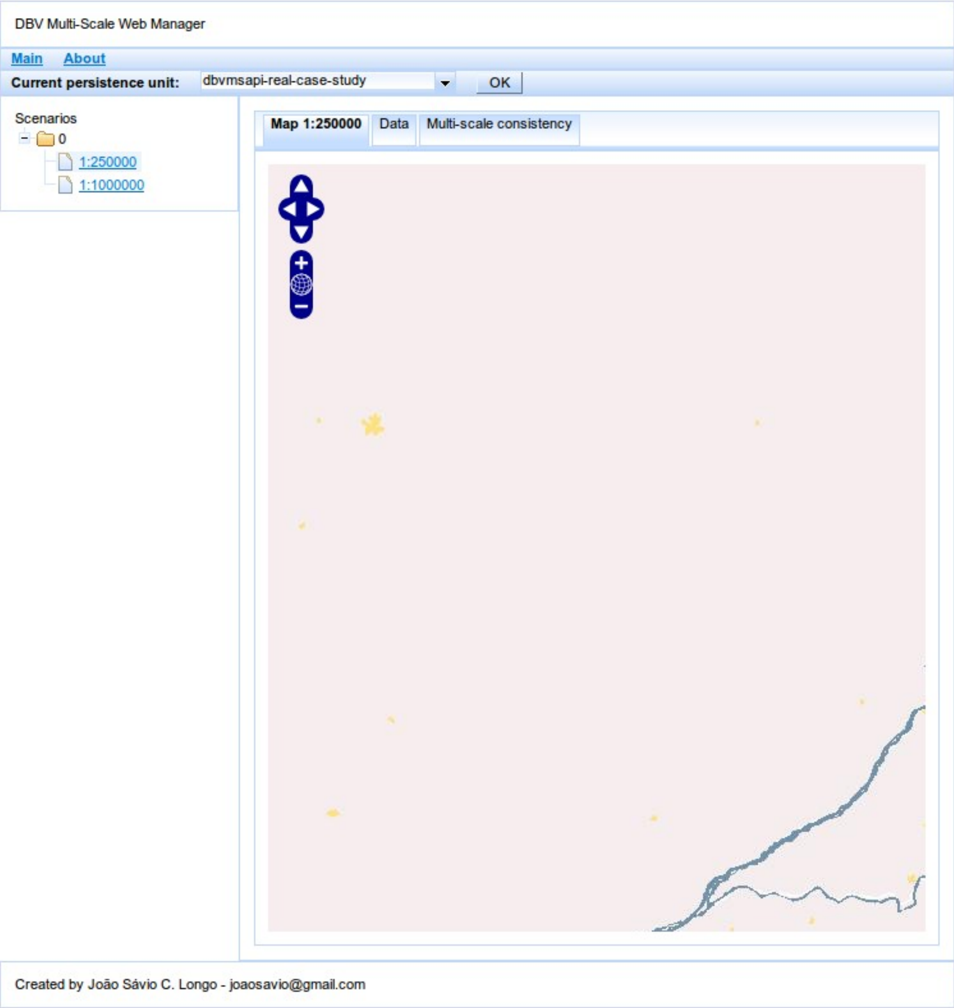


Starting from an empty database, with two possible stored scales

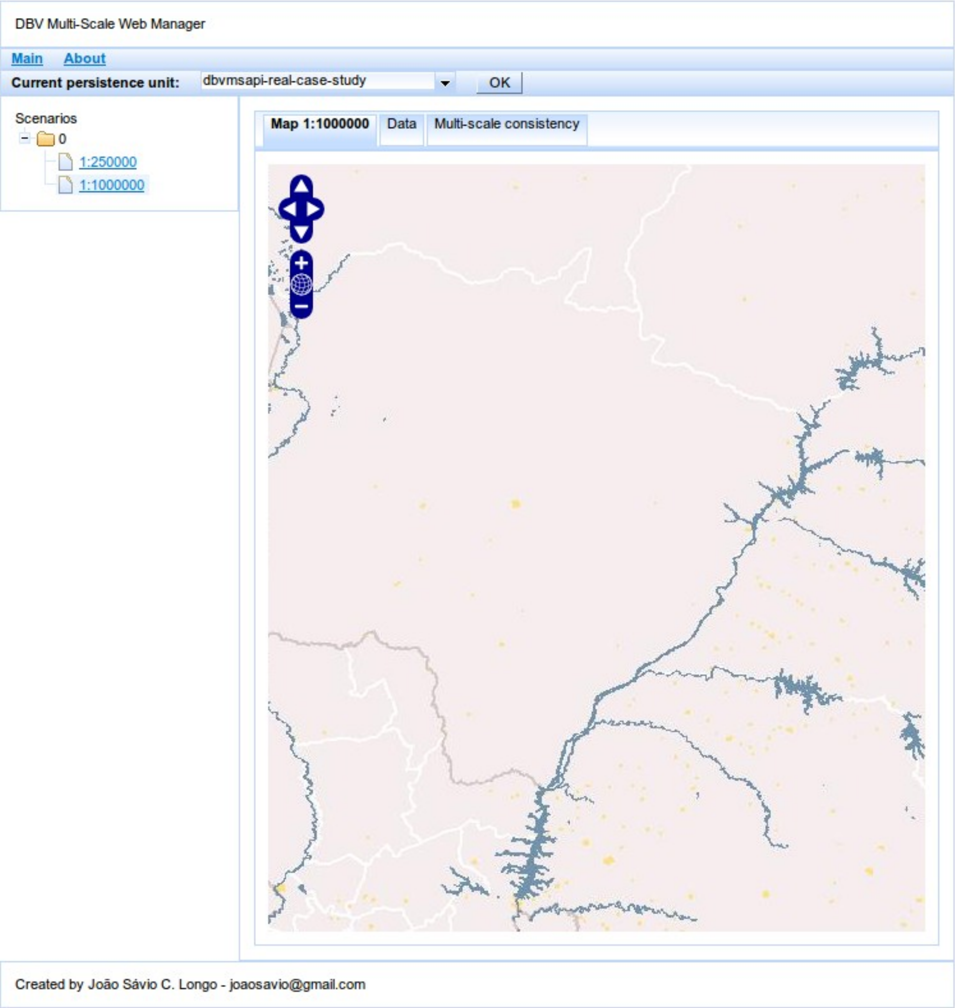
Scale 1:250k

Scale 1:1M

Building scenario 0

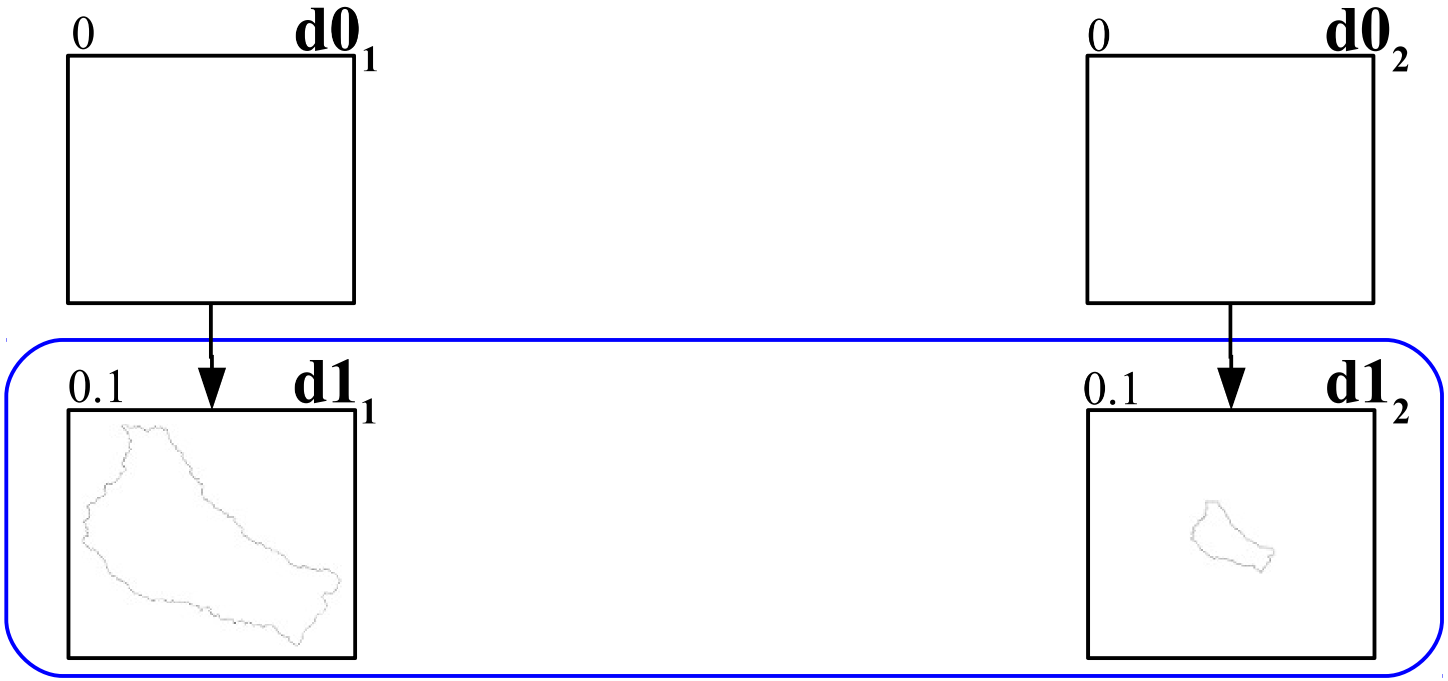


1:250k



1:1M

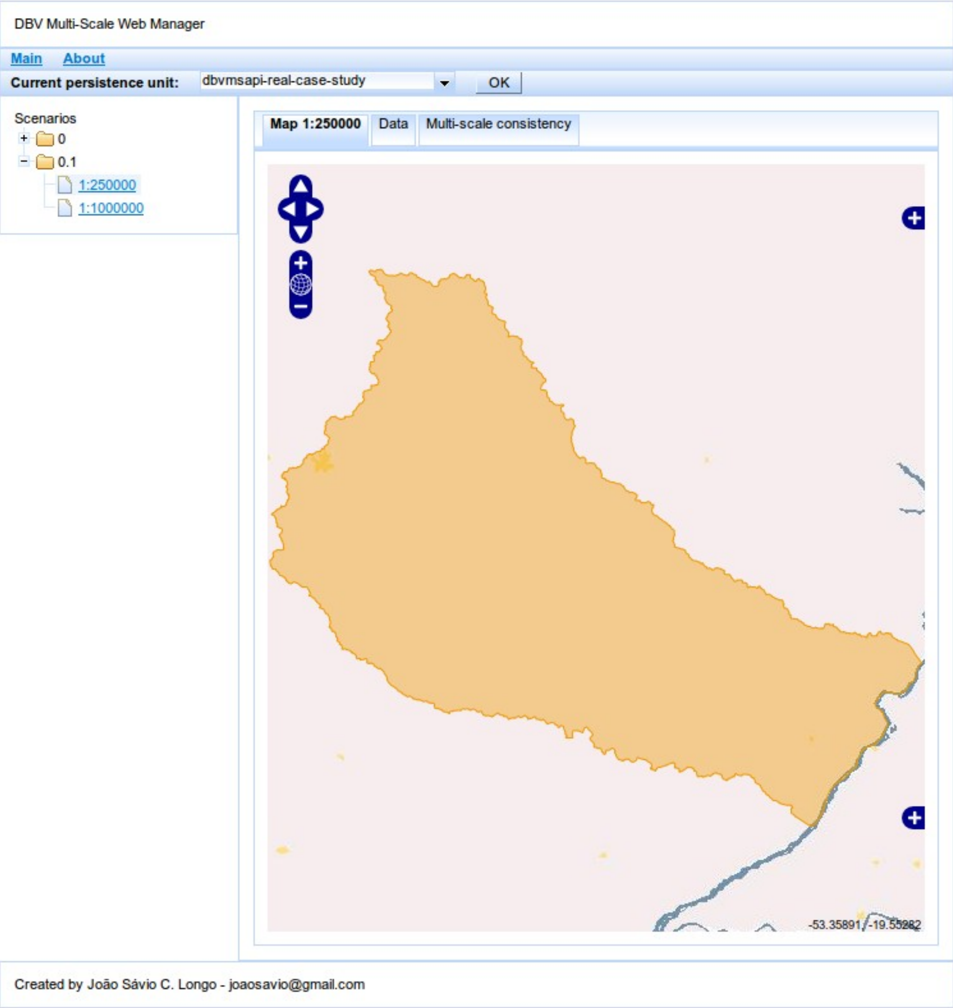
Building scenario 0.1



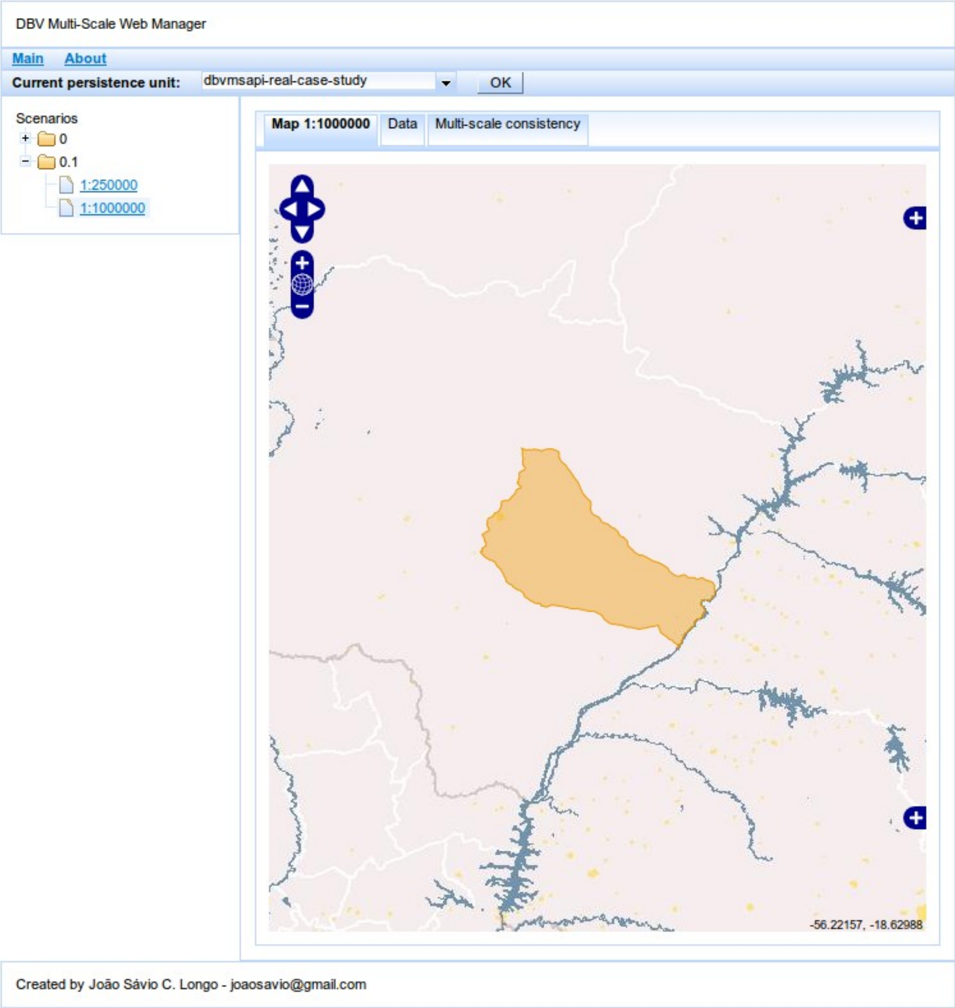
- scenario – set of DBVs, each corresponding to a given stored scale, at the same time
- trees grow and shrink together

Scale 1:250k **Scale 1:1M**

Building scenario 0.1

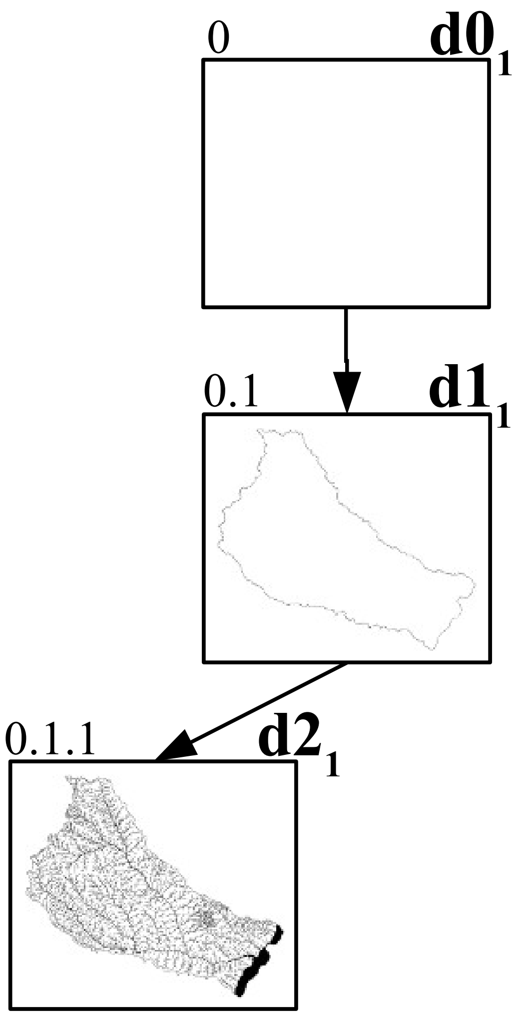


1:250k

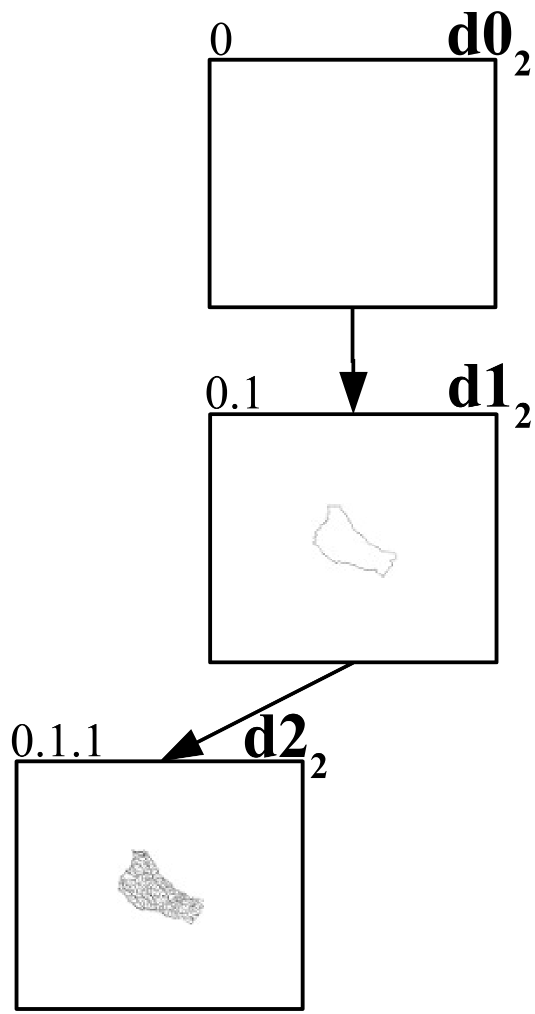


1:1M

Building scenario 0.1.1

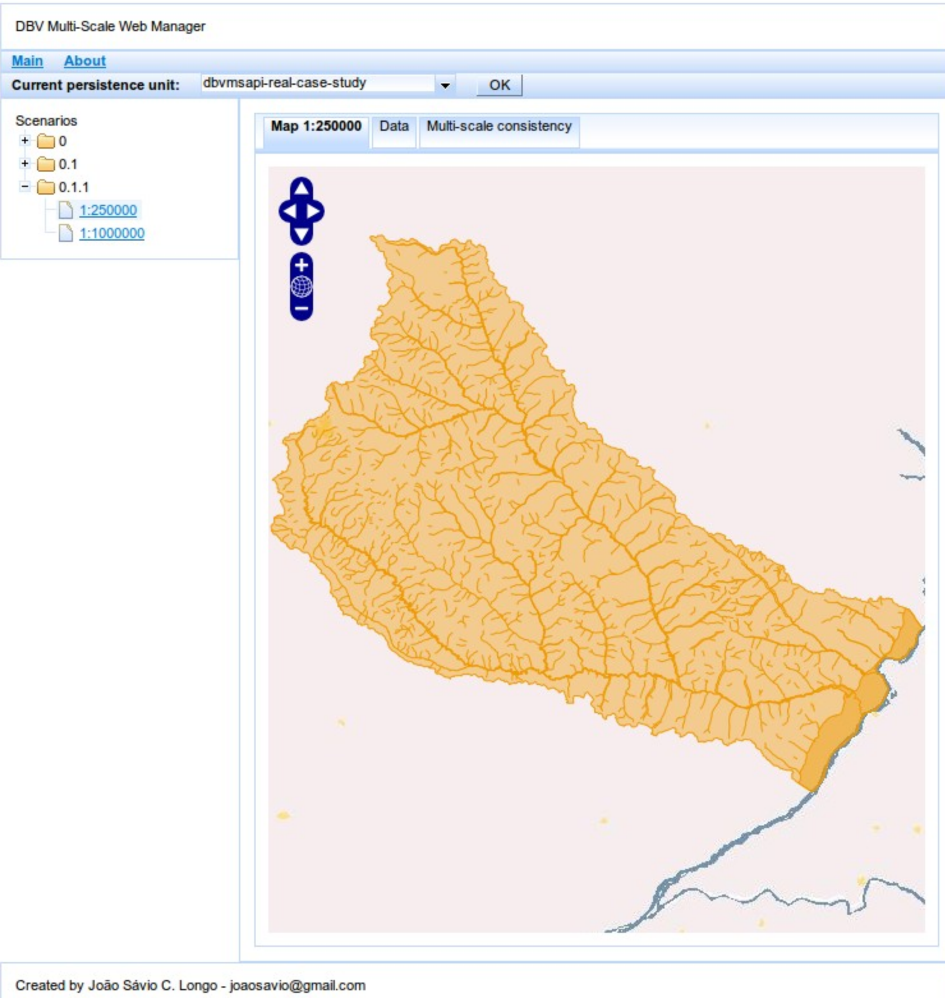


Scale 1:250k

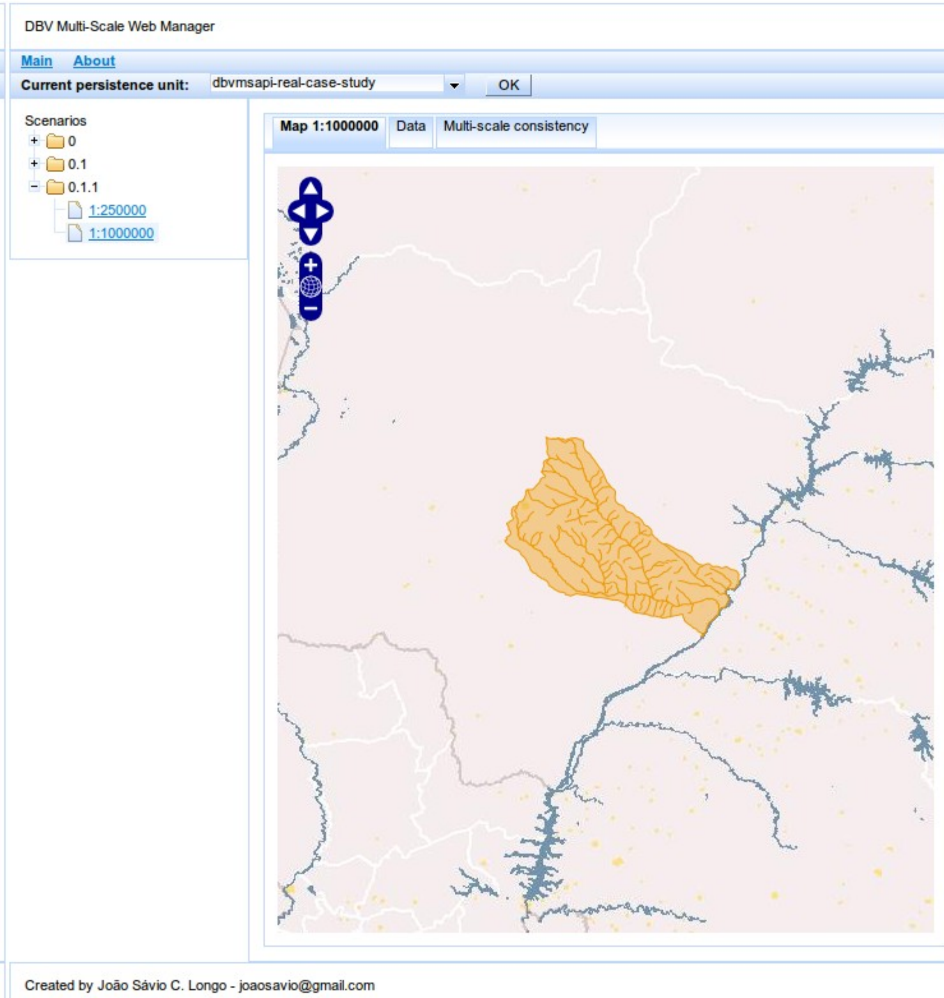


Scale 1:1M

Building scenario 0.1.1

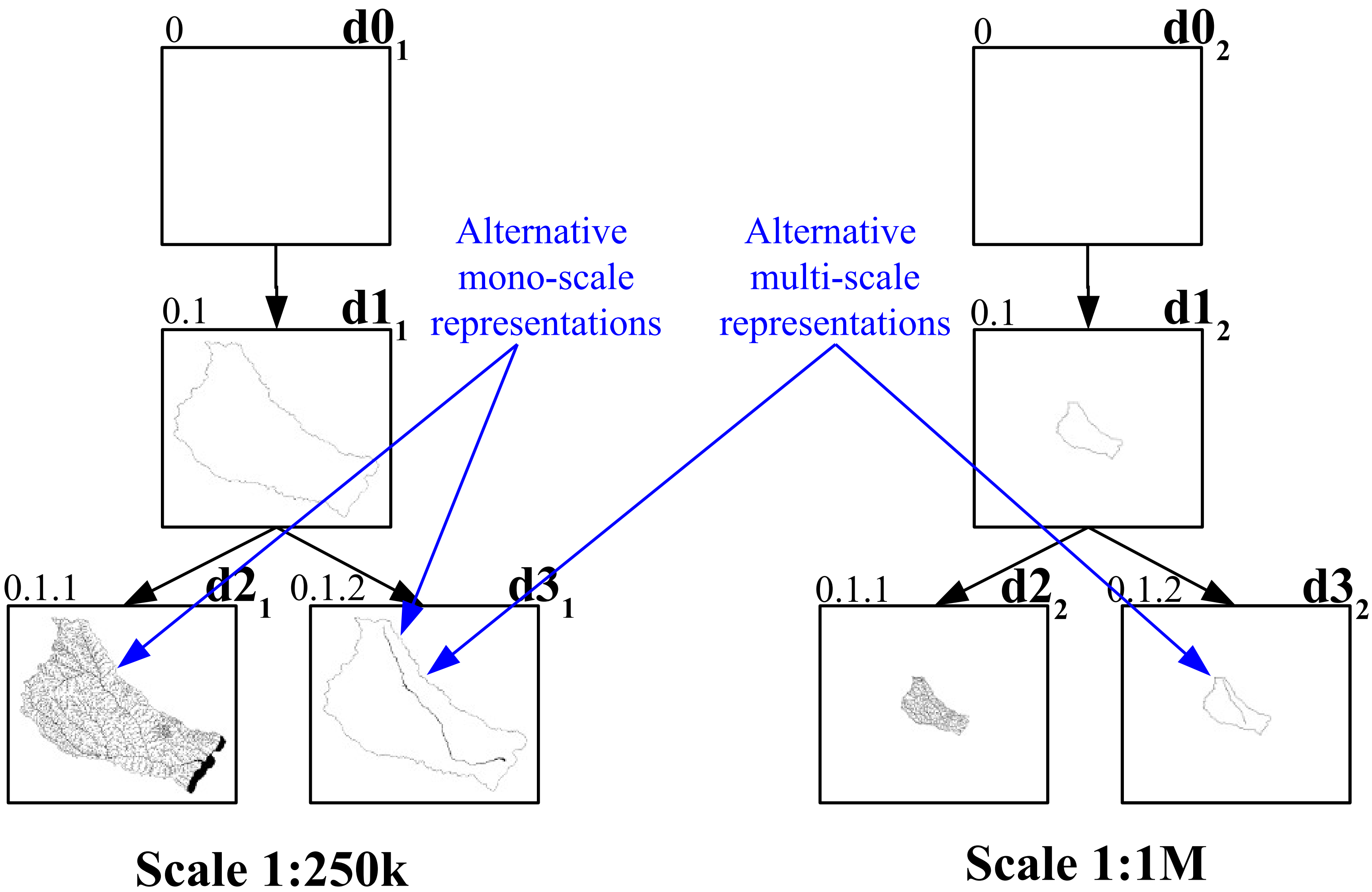


1:250k

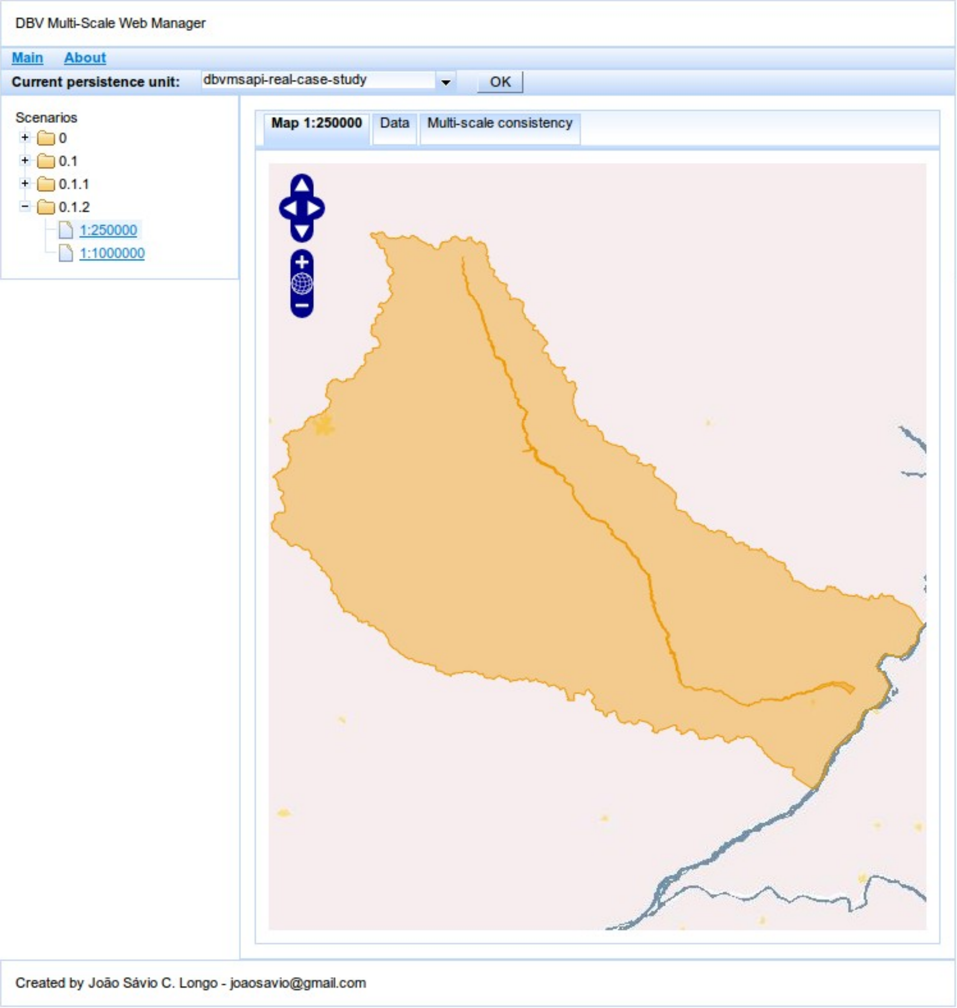


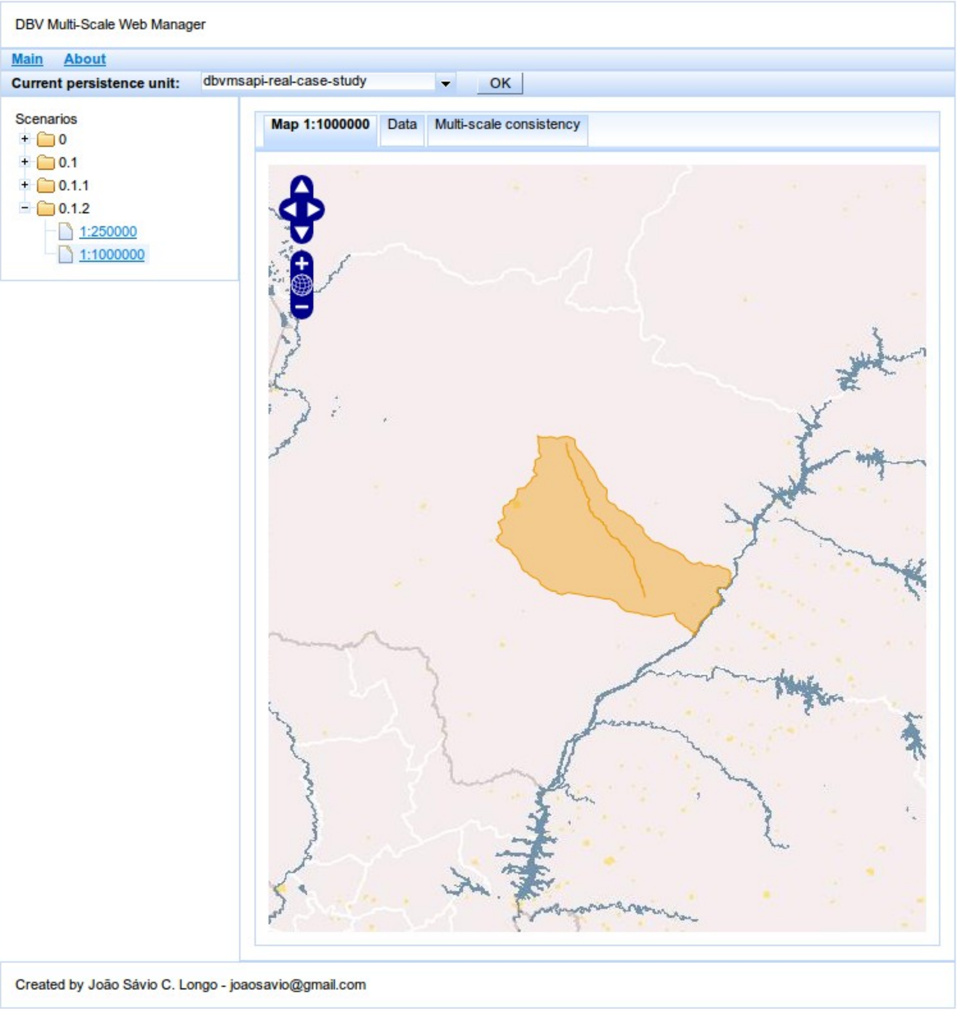
1:1M

Building scenario 0.1.2



Building scenario 0.1.2





1:250k **1:1M**

Conclusions and future work

Conclusions

1) How to manage multi-scale data?
   - DBV multi-scale platform
     - flexible MRDB – links through version stamps
2) How to trace real world evolution through time?
   - derivation tree
3) How to represent alternative scenarios?
   - creating branches
   - DBV model (in other approaches, evolution requires replication)

Future work

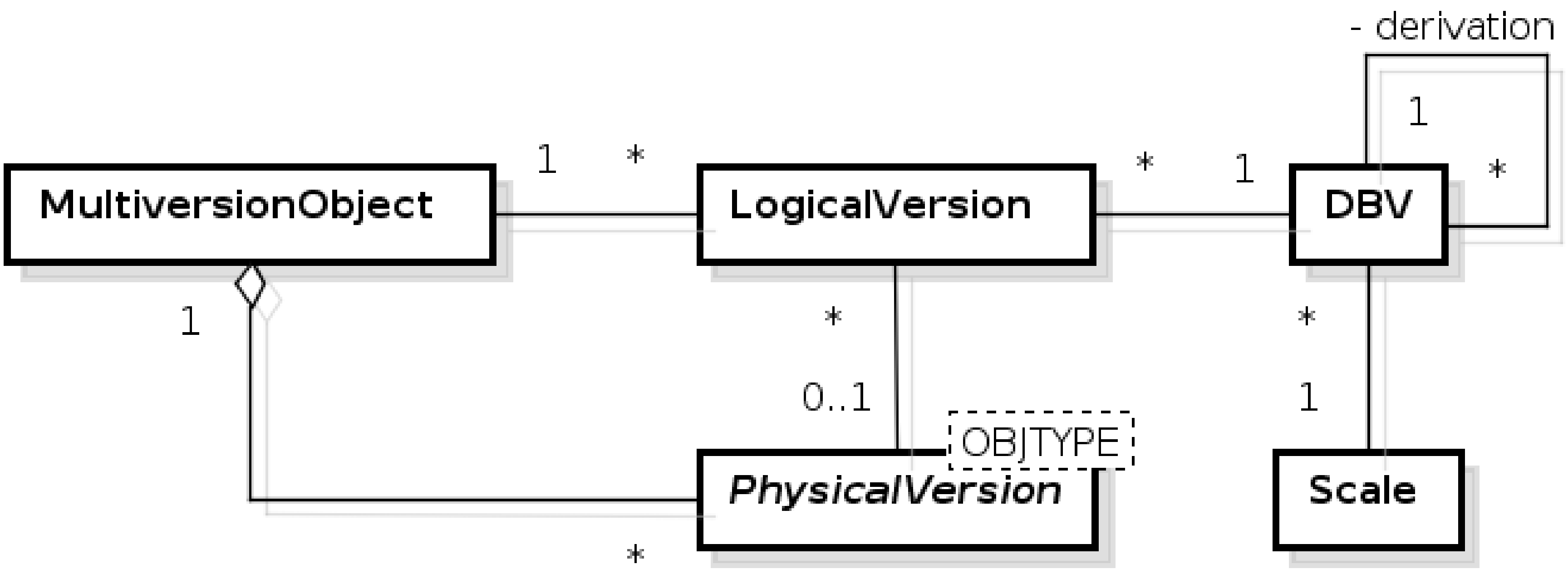
- Versioning along the temporal scale
- Specification and management of integrity constraints across scales, to determine rules for update propagation

Acknowledgments

- LIS (*Laboratory of Information Systems*)
- UNICAMP - IC
- FAPESP 2011/14280-0
- FAPESP – Microsoft Research Navscales
- Capes
- Embrapa
- INCT - Web Science
- CNPq
- MuZoo
- PRONEX

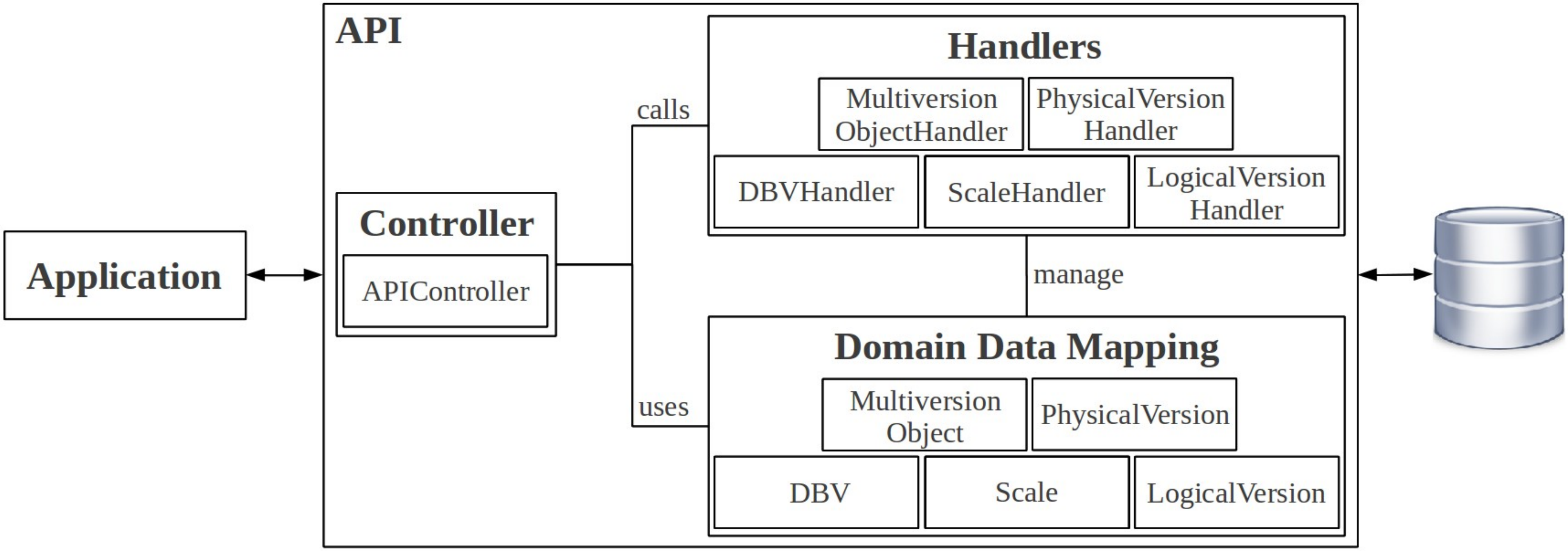
Thank you!

UML model



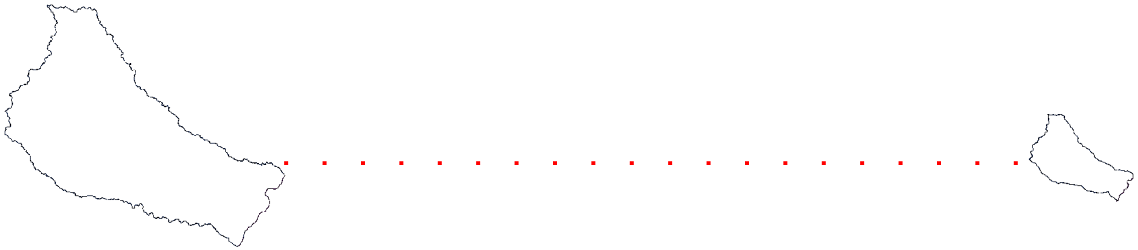
- control over the types of versioned objects

Architecture



MRDB

Link stored multi-representation data



Scale 1:250k

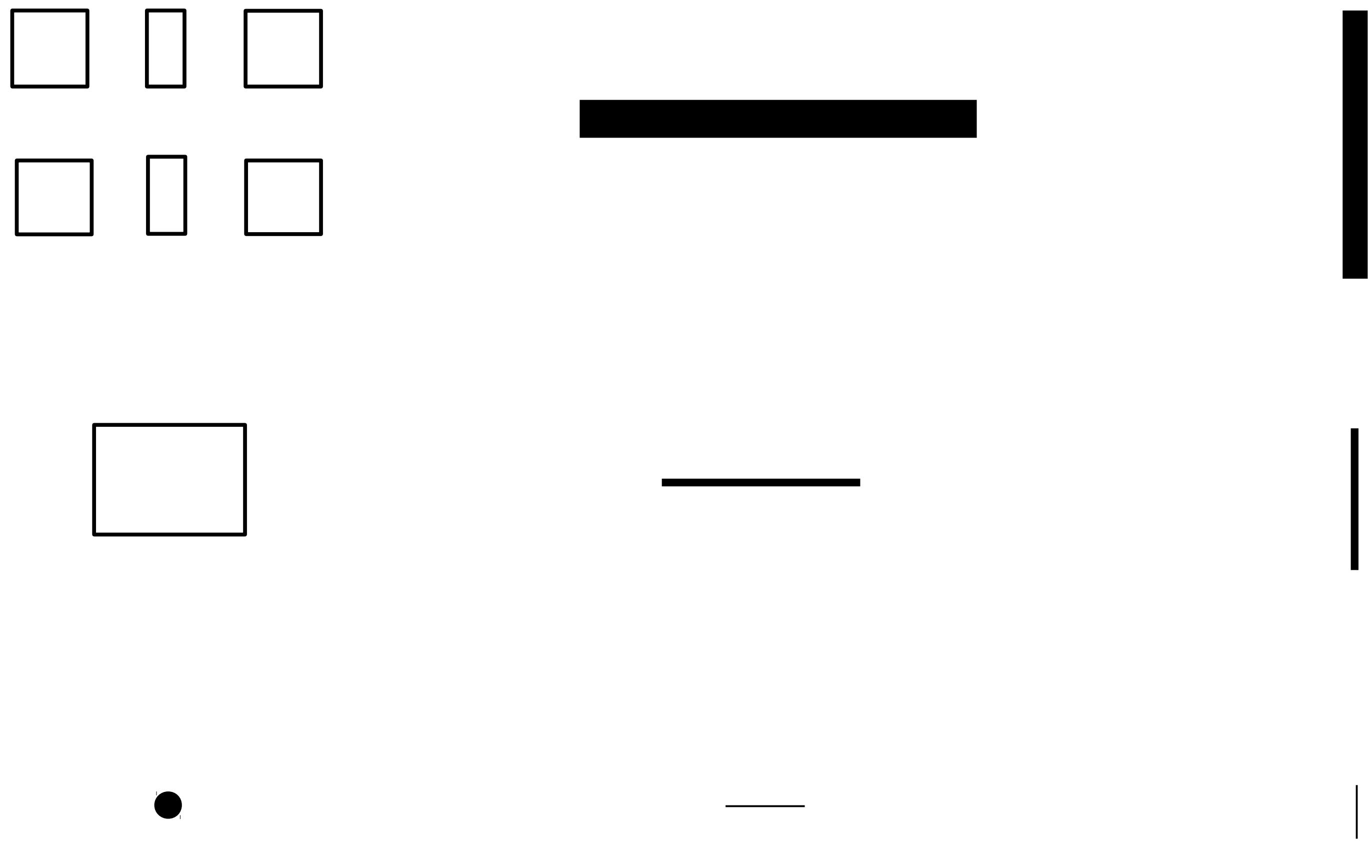
Scale 1:1M

MRDB

1:10k

1:20k

1:50k

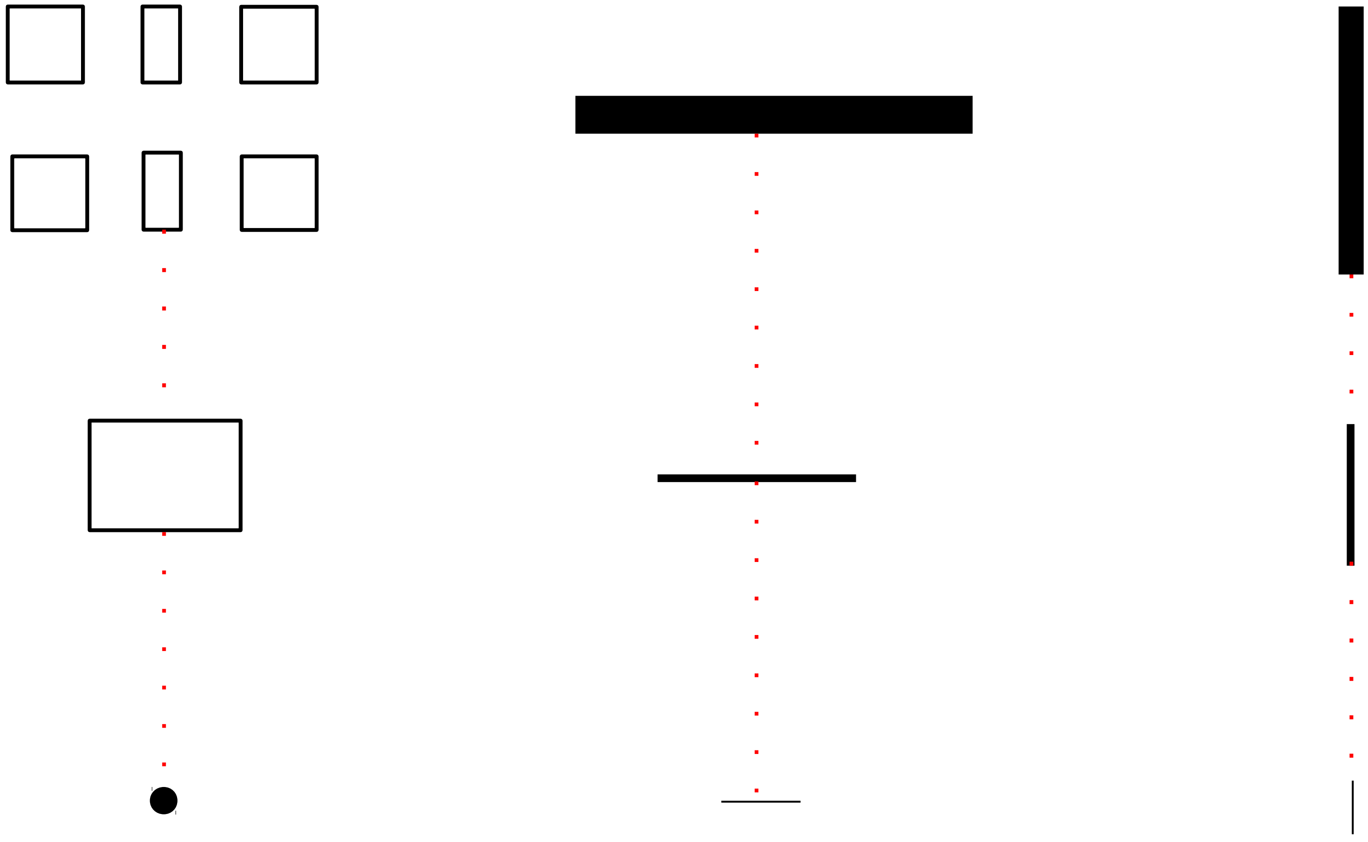


MRDB

1:10k

1:20k

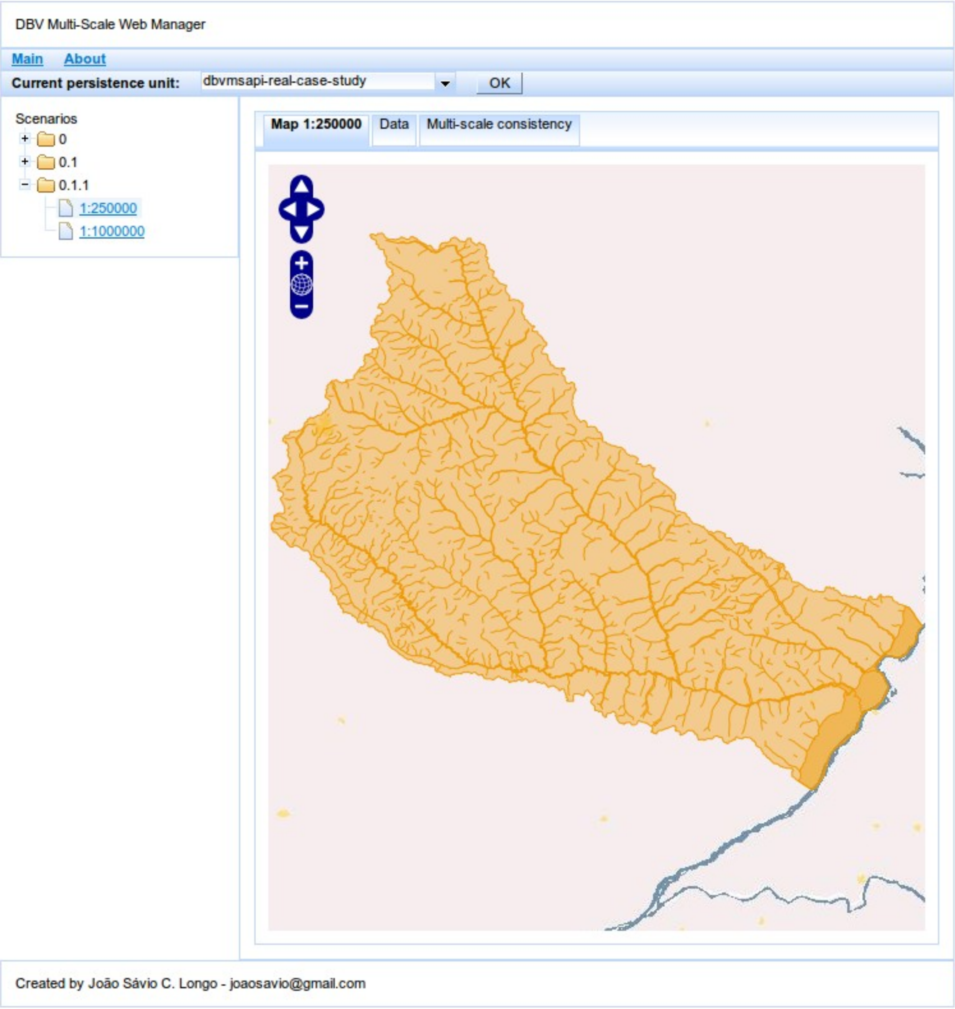
1:50k

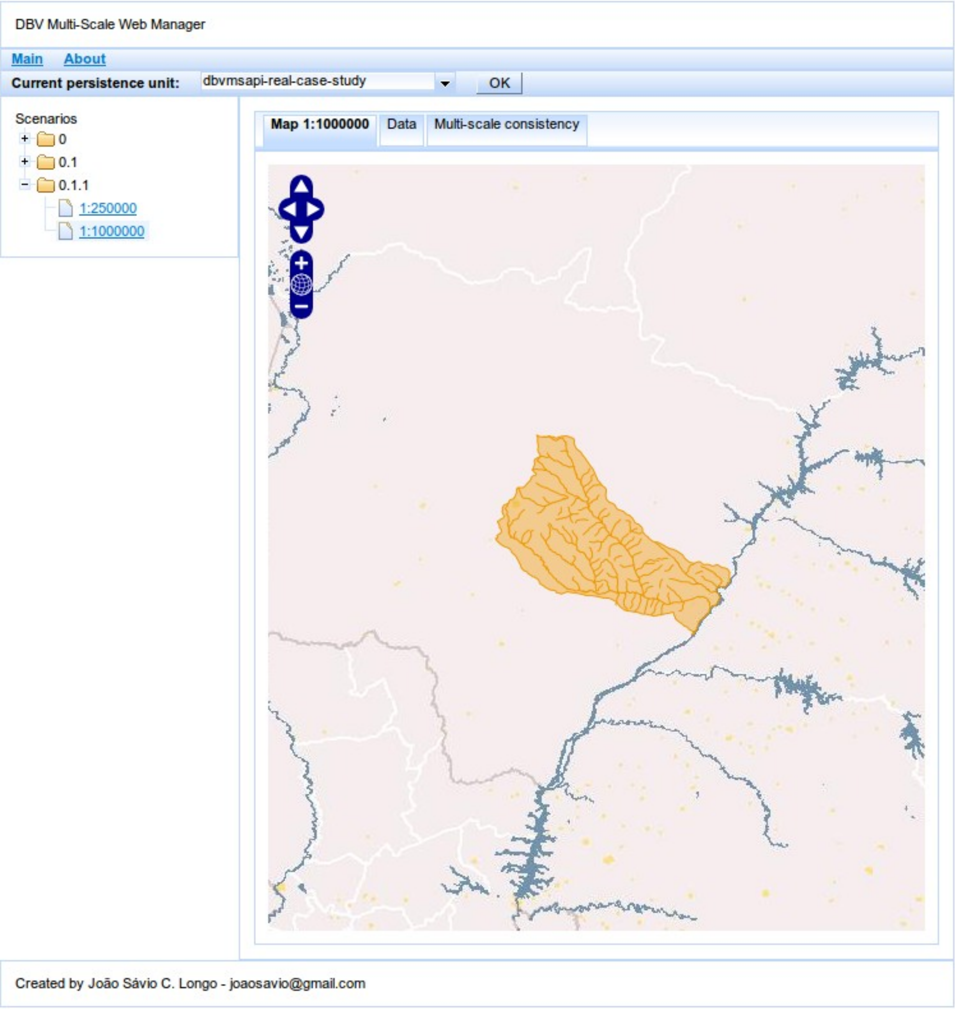


Example

- There are many real world objects in the example:
  - watershed
  - main river
  - all other rivers with name
- Each stored geometry is a physical version of an object
- So, in a DBV **d**, a **logical version** of a multiversion object **o** is represented by a physical version **pv**

Back to scenario 0.1.1

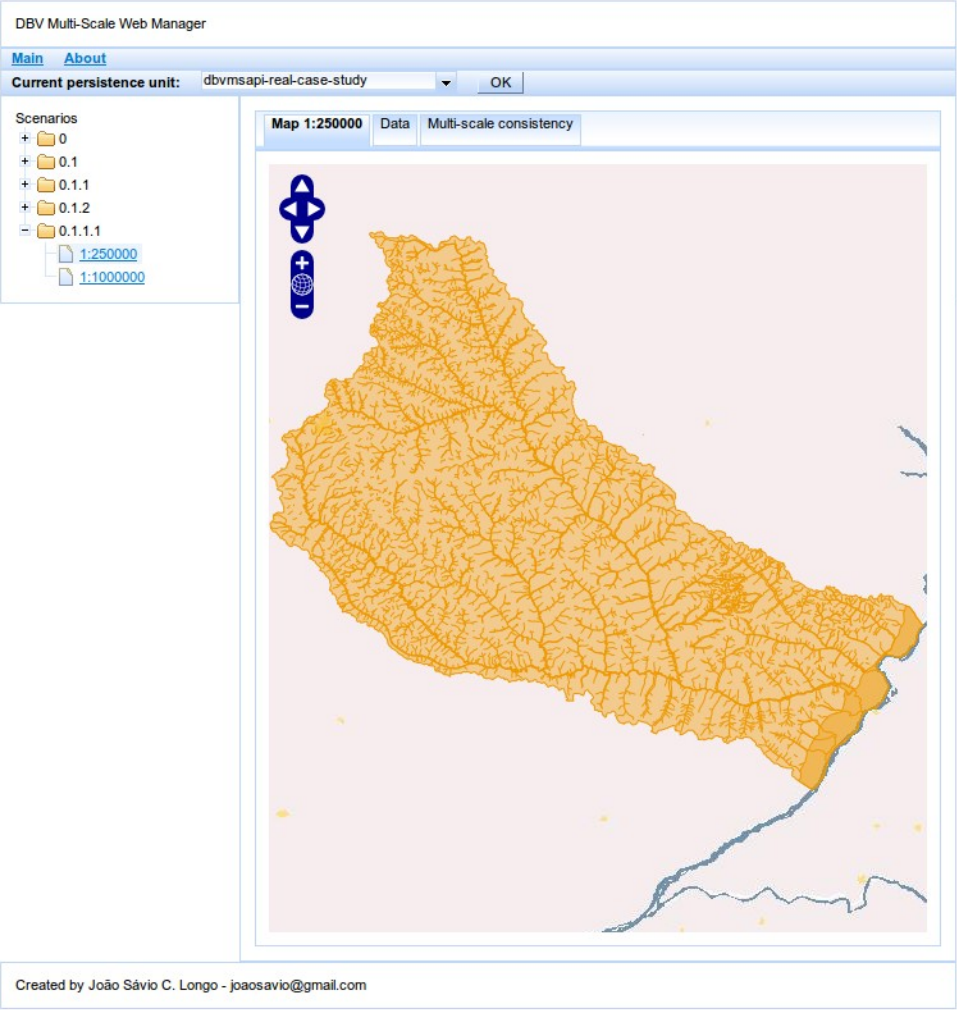


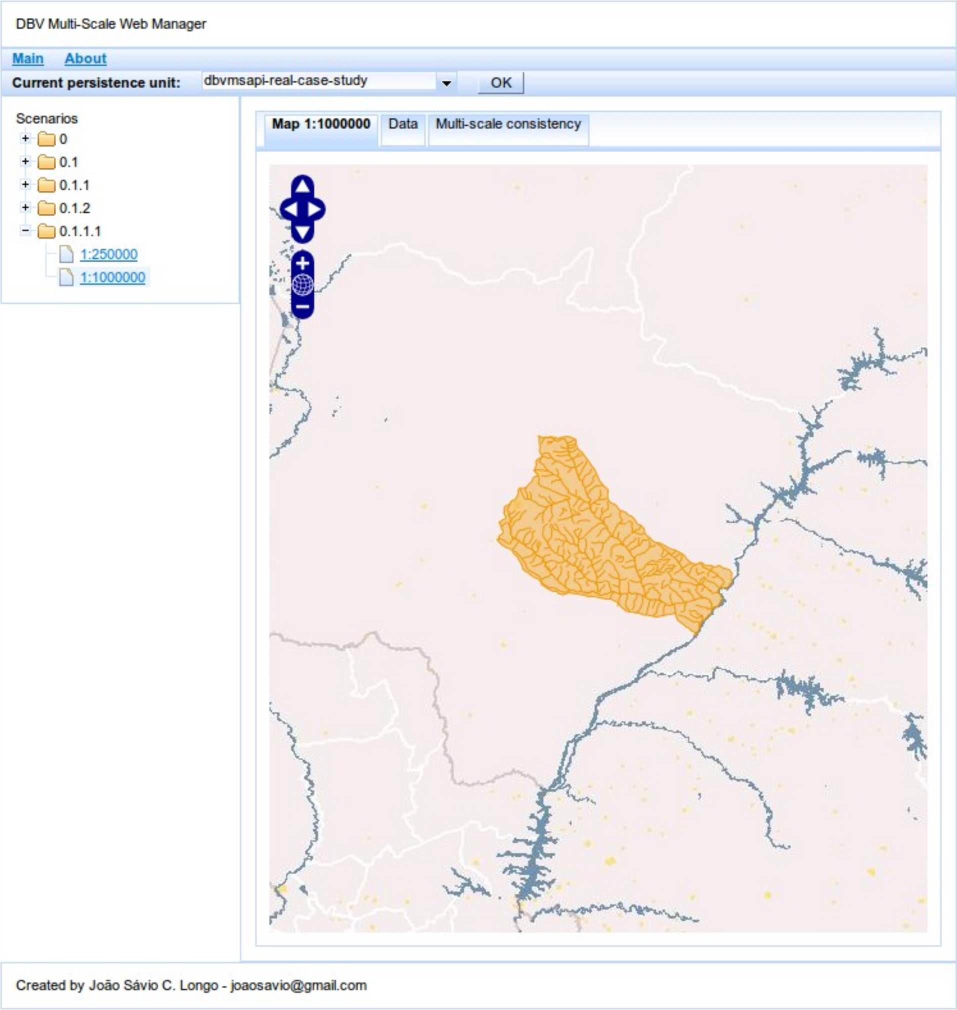


1:250k

1:1M

Scenario 0.1.1.1





1:250k **1:1M**