An algebra for spatio-temporal information generation

Edzer Pebesma, Simon Scheider, Benedikt Gräler, Christoph Stasch, Matthias Hinz
(Münster, Utrecht, Bochum, 52°North, Münster)

ifgi
Institute for Geoinformatics
University of Münster

EGU General Assembly, Apr 21, 2016

INTERNATIONAL JOURNAL OF GEOGRAPHICAL INFORMATION SCIENCE, 2016
http://dx.doi.org/10.1080/13658816.2016.1151520

Taylor & Francis
Taylor & Francis Group

Modeling spatiotemporal information generation

Simon Scheider[a,d], Benedikt Gräler[b], Edzer Pebesma[b] and Christoph Stasch[b,c]

[a]Departement Bau, Umwelt und Geomatik, Institut für Kartographie und Geoinformation, ETH Zürich, Zürich, Switzerland; [b]Fachbereich Geowissenschaften, Institute for Geoinformatics, University of Münster, Münster, Germany; [c]52°North Initiative for Geospatial Open Source Software GmbH, Münster, Germany; [d]Human Geography and Spatial Planning, Universiteit Utrecht, Utrecht, The Netherlands

ABSTRACT

Maintaining knowledge about the provenance of datasets, that is, about how they were obtained, is crucial for their further use. Contrary to what the overused metaphors of 'data mining' and 'big data' are implying, it is hardly possible to use data in a meaningful way if information about sources and types of conversions is discarded in the process of data gathering. A generative model of spatiotemporal information could not only help automating the description of derivation processes but also assessing the scope of a dataset's future use by exploring possible transformations. Even

ARTICLE HISTORY
Received 1 September 2015
Accepted 2 February 2016

KEYWORDS
Spatiotemporal data types; data generation; provenance model; algebra

PDF

European Commission - Press release

European Cloud Initiative to give Europe a global lead in the data-driven economy

Brussels, 19 April 2016

The Commission today presented its blueprint for cloud-based services and world-class data infrastructure to ensure science, business and public services reap benefits of big data revolution.

Europe is the largest producer of scientific data in the world, but insufficient and fragmented infrastructure means this 'big data' is not being exploited to its full potential. By bolstering and interconnecting existing research infrastructure, the Commission plans to create a new **European Open Science Cloud** that will offer Europe's 1.7 million researchers and 70 million science and technology professionals a virtual environment to store, share and re-use their data across disciplines and borders. This will be underpinned by the **European Data Infrastructure**, deploying the high-bandwidth networks, large scale storage facilities and super-computer capacity necessary to effectively access and process large datasets stored in the cloud. This world-class infrastructure will ensure Europe participates in the global race for high performance computing in line with its economic and knowledge potential.

Focusing initially on the scientific community - in Europe and among its global partners -, the user base will over time be enlarged to the public sector and to industry. This initiative is part of a package of measures to strengthen Europe's position in data-driven innovation, to improve competitiveness and cohesion and to help create a Digital Single Market in Europe (press release).

Carlos **Moedas**, Commissioner for Research, Science and Innovation, said: "*Our goal is to create a European Open Science Cloud to make science more efficient and productive and let millions of researchers share and analyse research data in a trusted environment across technologies, disciplines and borders. We listened to the scientific community's plea for an infrastructure for Open Science and with this comprehensive plan we can get down to work. The benefits of open data for Europe's science, economy and society will be enormous.*"

Günther H. **Oettinger**, Commissioner for the Digital Economy and Society, said:"*The European Cloud Initiative will unlock the value of big data by providing world-class supercomputing capability, high-speed connectivity and leading-edge data and software services for science, industry and the public sector. With this initiative, our ambition is to be in the global top-three in high performance computing by 2020. We will also be looking into the potential of quantum technologies which hold the promise to solve computational problems beyond current supercomputers.*"

How do we generate data?

1. by observing (often: measuring)
2. by deriving:
   - from observed data,
   - or from derived data

Questions we asked ourselves:

A is dataset x equivalent to dataset y, and a proper input to derivation z?

B how can we advertise derived dataset (e.g., inform to which derivation is it a proper input)?

James Frew's laws on metadata:

1. scientists don't write metadata
2. any scientist can be forced to write bad metadata

Can, somehow, scientists be relieved from the task, but still questions **A** and **B** be answered?

How do we generate data?

1. by observing (often: measuring)
2. by deriving:
   - from observed data,
   - or from derived data

Questions we asked ourselves:

A is dataset x equivalent to dataset y, and a proper input to derivation z?

B how can we advertise derived dataset (e.g., inform to which derivation is it a proper input)?

James Frew's laws on metadata:

1. scientists don't write metadata
2. any scientist can be forced to write bad metadata

Can, somehow, scientists be relieved from the task, but still questions **A** and **B** be answered?

How do we generate data?

1. by observing (often: measuring)
2. by deriving:
   - from observed data,
   - or from derived data

Questions we asked ourselves:

A is dataset x equivalent to dataset y, and a proper input to derivation z?

B how can we advertise derived dataset (e.g., inform to which derivation is it a proper input)?

James Frew's laws on metadata:

1. scientists don't write metadata
2. any scientist can be forced to write bad metadata

Can, somehow, scientists be relieved from the task, but still questions **A** and **B** be answered?

How do we generate data?

1. by observing (often: measuring)
2. by deriving:
   - from observed data,
   - or from derived data

Questions we asked ourselves:

A is dataset x equivalent to dataset y, and a proper input to derivation z?

B how can we advertise derived dataset (e.g., inform to which derivation is it a proper input)?

James Frew's laws on metadata:

1. scientists don't write metadata
2. any scientist can be forced to write bad metadata

Can, somehow, scientists be relieved from the task, but still questions **A** and **B** be answered?

How do scientists communicate data generation?

```
library("RandomFields")

##########################################
##  SECTION 4: UNCONDITIONAL SIMULATION  ##
##########################################

RFoptions(seed = 0, height = 4)
## seed=0:  *ANY* simulation will have the random seed 0; set
##          RFoptions(seed=NA) to make them all random again

## Fig. 1: linear model of coregionalization
M1 <- c(0.9, 0.6)
M2 <- c(sqrt(0.19), 0.8)
model <- RMmatrix(M = M1, RMwhittle(nu = 0.3)) +
         RMmatrix(M = M2, RMwhittle(nu = 2))
x <- y <- seq(-10, 10, 0.2)
simu <- RFsimulate(model, x, y)
plot(simu)

## Fig. 2: Wackernagel's delay model
model <- RMdelay(RMstable(alpha = 1.9, scale = 2), s = c(4, 4))
simu <- RFsimulate(model, x, y)
plot(simu, zlim = 'joint')

## Fig. 3: extended Wackernagel's delay model
model <- RMdelay(RMstable(alpha = 1.9, scale = 2), s = c(0, 4)) +
         RMdelay(RMstable(alpha = 1.9, scale = 2), s = c(4, 0))
simu <- RFsimulate(model, x, y)
plot(simu, zlim = 'joint')
# ToPDF("delay")
plot(model, dim = 2, xlim = c(-5, 5), main = "Covariance function",
     cex = 1.5, col = "brown")

## Fig. 4: latent dimension model
## MARGIN.slices has the effect of choosing the third dimension
##                as latend dimension
```

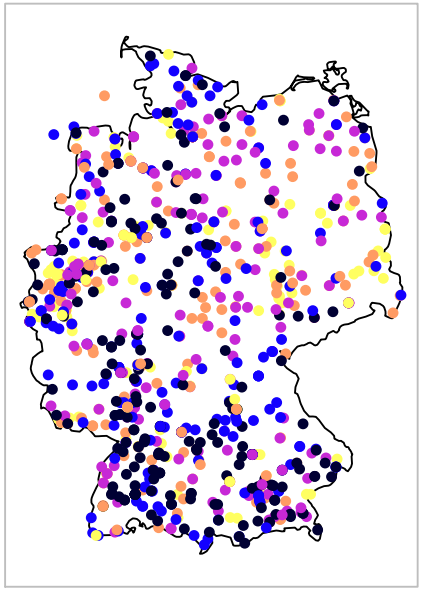
Algebra

basic notions:

- basic types with reference systems
- data generation types (functions)
- data derivation

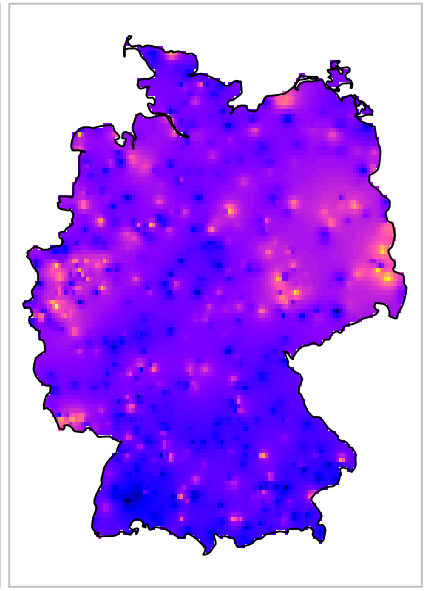
Are phenomena discrete, or continuous?

CO_2 emissions of power plants

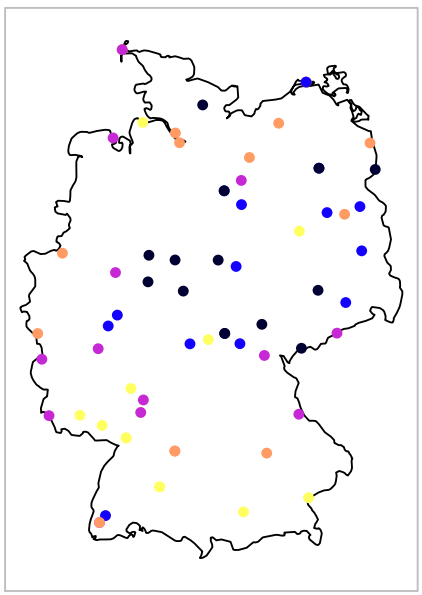


Sum of CO_2 emissions

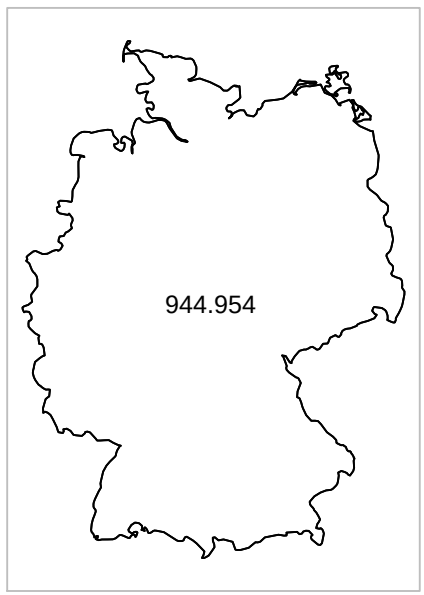
Interpolated CO_2 emissions



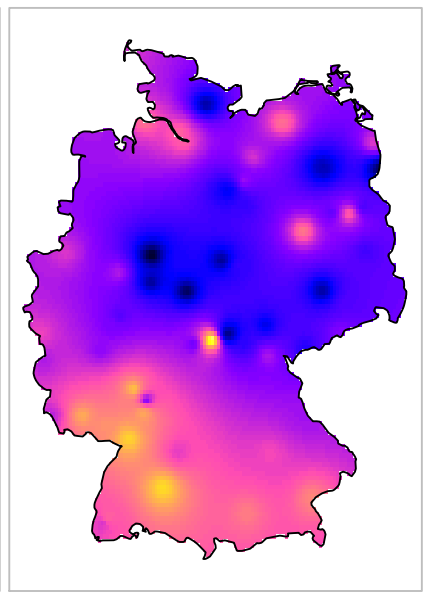
PM_{10} measurements



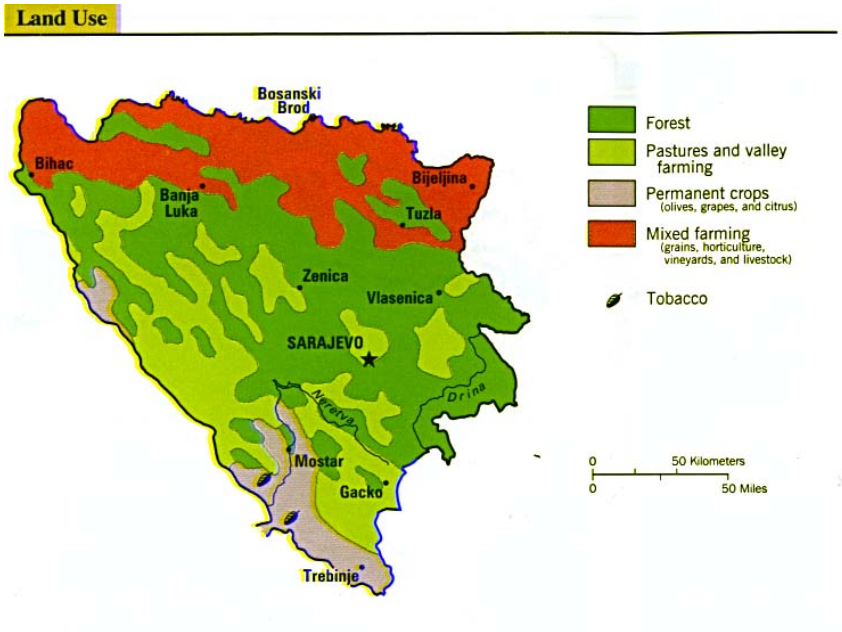
Sum of PM_{10} measurements

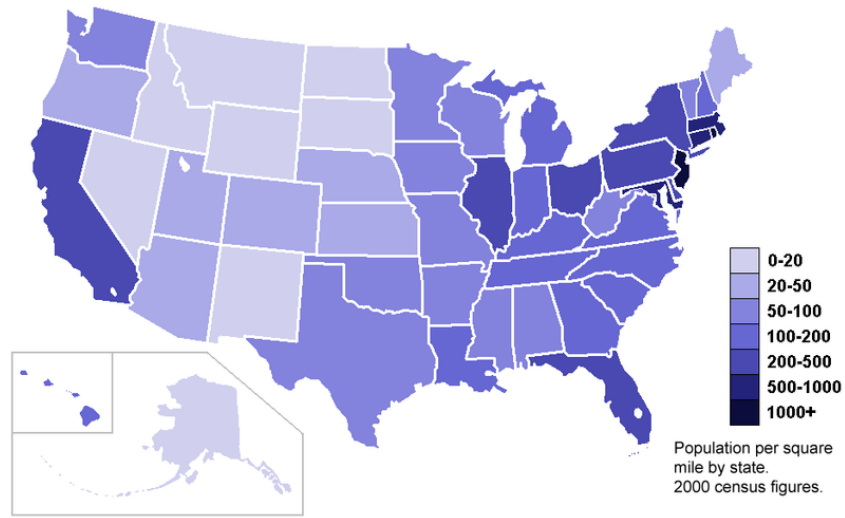


Interpolated PM_{10} measurements



How do values refer to regions?





value = constant

value = aggregate

Basic Reference System types

Basic reference system types and simple derivations thereof. Each type needs to go along with its reference system (RS).

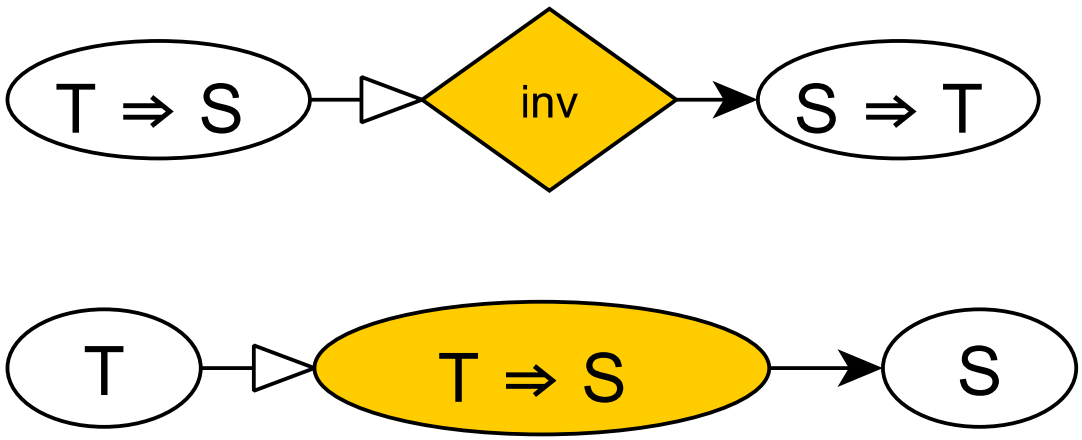
$\mathcal{P}$ denotes the power set (set of all subsets).

Symbol	Definition	Meaning	Description
S		$\mathbb{R}^3$	Set of possible spatial locations with RS.
T		$\mathbb{R}$	Set of possible moments in time with RS.
D		$\mathbb{N}$	Set of possible discrete entity identifier with RS.
Q		$\mathbb{R}$	Set of possible observed values with RS.
R	S set	$\mathcal{P}(S)$	Set of regions: bounded by polygons, or collection of isolated locations and combinations thereof.
I	T set	$\mathcal{P}(T)$	Set of collections of moments in time: continuous intervals or a set of moments in time or combinations thereof.
D set	D set	$\mathcal{P}(D)$	Sets of object identifiers
Q set	Q set	$\mathcal{P}(Q)$	Sets of quality values.
bool		$\{\mathrm{T}, \mathrm{F}\}$	Boolean, also used to express predicates for selection
Extent	$R \times I$	$R \times I$	set of spatio-temporal extent as the orthogonal product of the spatial and temporal projections
Occurs	$(S \times T)$ set	$\mathcal{P}(S \times T)$	set of spatio-temporal subsets, occurrences of events and objects, but also of certain values or conditions in a field; footprint, support

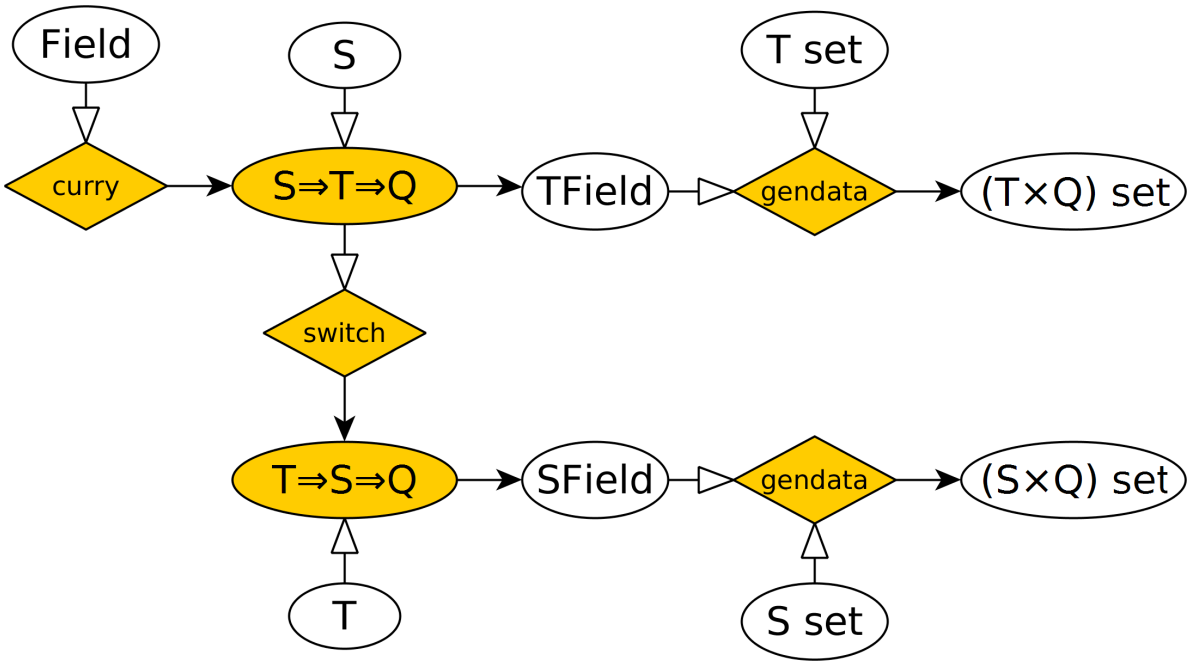
Data Generation Types

Symbol	Type definition	Description
Field	$S \times T \Rightarrow Q$	spatio-temporal field
Lattice	$R \Rightarrow I \Rightarrow Q$	spatio-temporal lattice
Event	$D \Rightarrow S \times T$	spatio-temporal events
Trajectory	$T \Rightarrow S$	trajectory
Objects	$D \Rightarrow T \Rightarrow S$	objects in time and space
LatticeT	$S \Rightarrow I \Rightarrow Q$	spatial temporal lattice
BlockEvent	$D \Rightarrow \mathrm{Extent}$	events affecting a set of locations and lasting for some time interval
RegionalTrajectory	$T \Rightarrow R$	trajectory of regions
BlockObjects	$D \Rightarrow I \Rightarrow R$	objects in space and time defined over regions and collections of moments in time
...	...	...

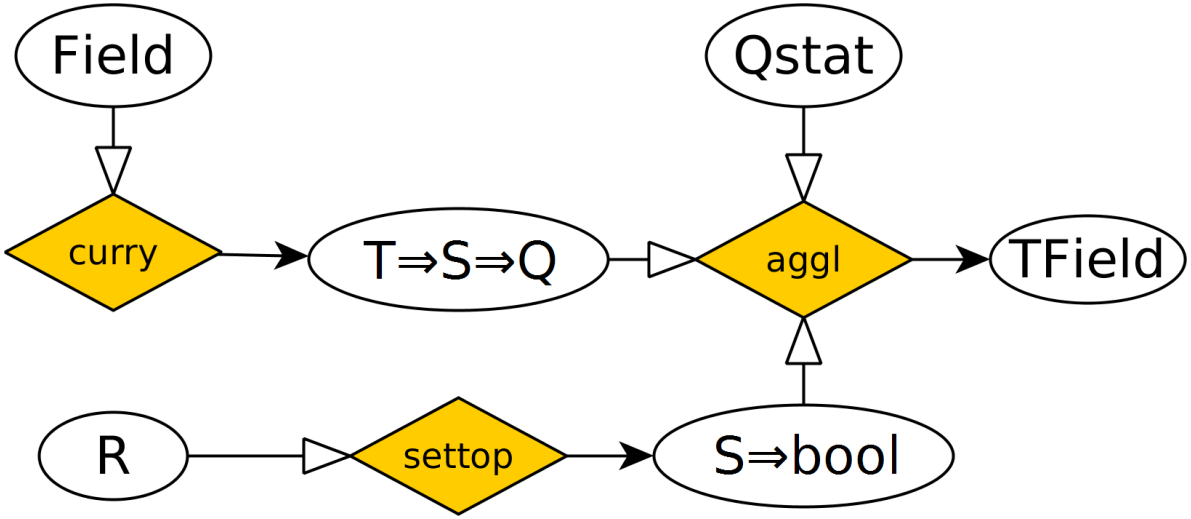
Data derivation

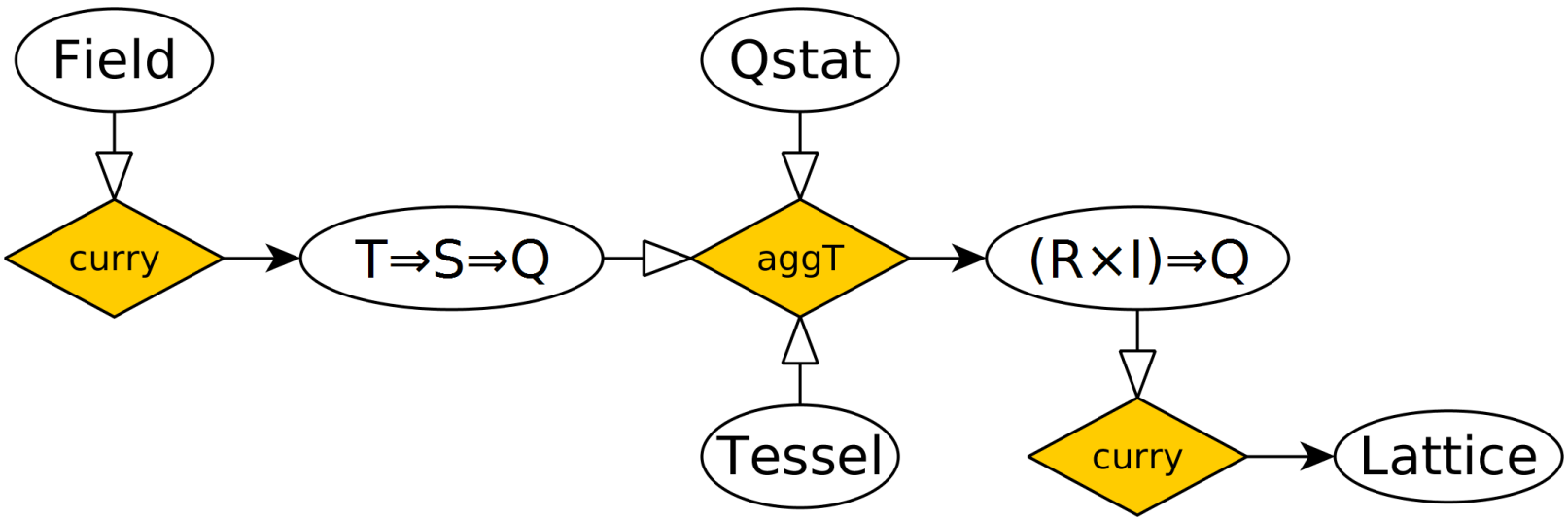


Data derivation: generating field data



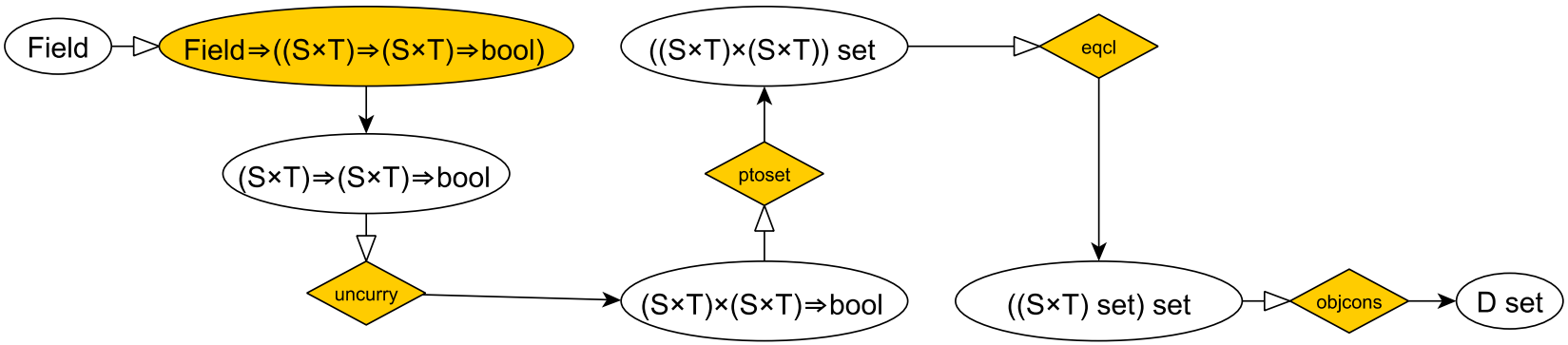
Data derivation: spatial/temporal aggregation

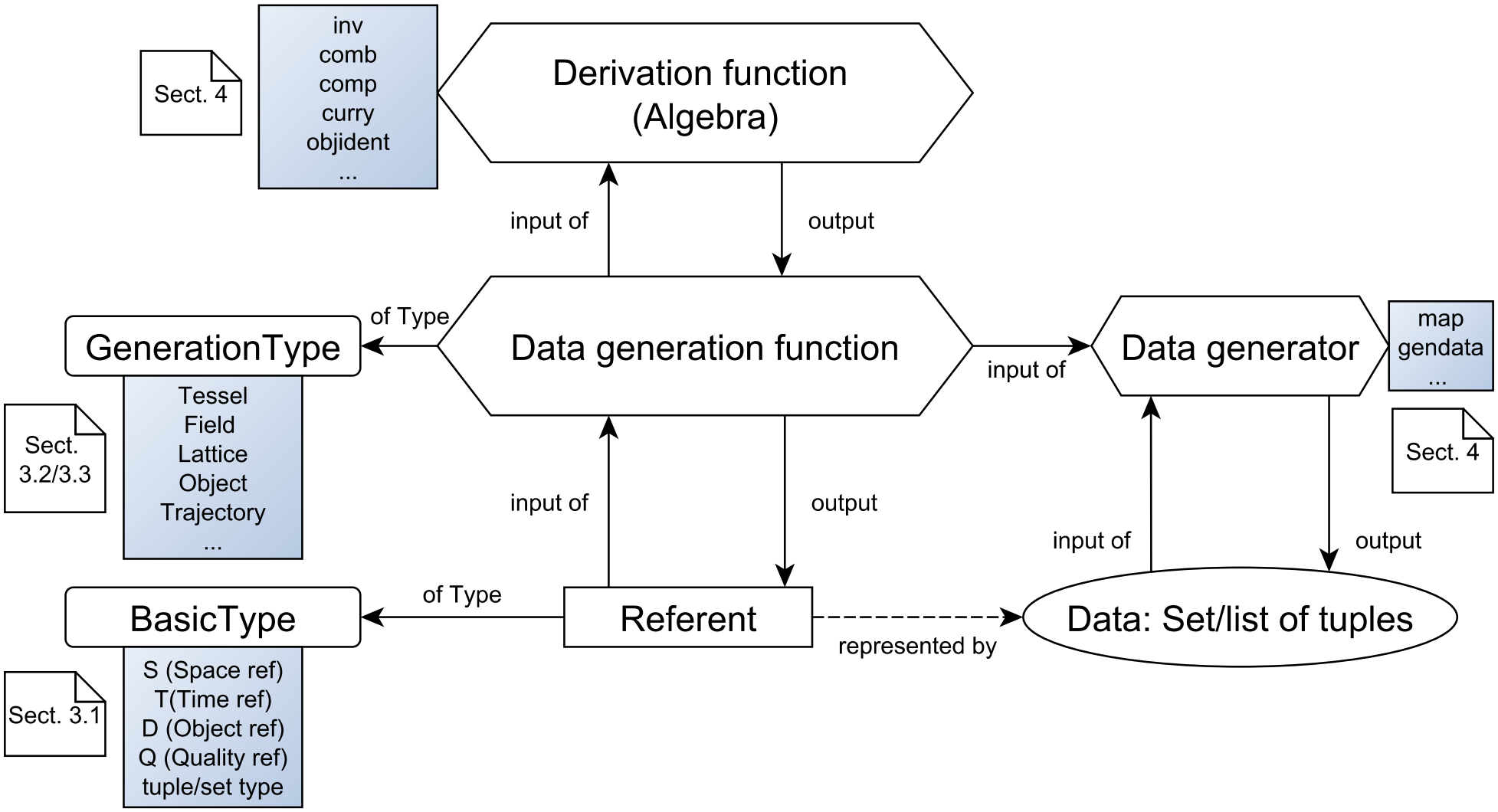




see paper for definitions of curry, aggl, aggT and settop

Data derivation: deriving objects from fields




Sect. 4
inv
comb
comp
curry
objident
...
Derivation function
(Algebra)
input of
output
GenerationType
of Type
Data generation function
input of
Data generator
map
gendata
...
Sect. 4
Sect.
3.2/3.3
Tessel
Field
Lattice
Object
Trajectory
...
input of
output
input of
output
BasicType
of Type
Referent
represented by
Data: Set/list of tuples
Sect. 3.1
S (Space ref)
T(Time ref)
D (Object ref)
Q (Quality ref)
tuple/set type

Discussion & Conclusions

- In order to be able to make meaningful inferences and type checking, we need to handle data at the level of *fields*, *events*, *objects*, *lattices*; handling them as *points*, *lines*, *polygons*, *grids* is not enough
- despite all data being discrete, our algebra distinguishes continuous from discrete phenomena
- the algebra makes explicit how values relate to {regions, time intervals} (constant? aggregate?)
- reproduction scripts convey syntax, but often little semantics
- R is written by scientists, and can be written to generate derivation graphs (w.i.p.)
- OWL DL (meaning: LOD) cannot make inference on functions of functions; higher order logic is needed for this.

Discussion & Conclusions

- In order to be able to make meaningful inferences and type checking, we need to handle data at the level of *fields, events, objects, lattices*; handling them as *points, lines, polygons, grids* is not enough
- despite all data being discrete, our algebra distinguishes continuous from discrete phenomena
- the algebra makes explicit how values relate to {regions, time intervals} (constant? aggregate?)
- reproduction scripts convey syntax, but often little semantics
- R is written by scientists, and can be written to generate derivation graphs (w.i.p.)
- OWL DL (meaning: LOD) cannot make inference on functions of functions; higher order logic is needed for this.

Discussion & Conclusions

- In order to be able to make meaningful inferences and type checking, we need to handle data at the level of *fields, events, objects, lattices*; handling them as *points, lines, polygons, grids* is not enough
- despite all data being discrete, our algebra distinguishes continuous from discrete phenomena
- the algebra makes explicit how values relate to {regions, time intervals} (constant? aggregate?)
- reproduction scripts convey syntax, but often little semantics
- R is written by scientists, and can be written to generate derivation graphs (w.i.p.)
- OWL DL (meaning: LOD) cannot make inference on functions of functions; higher order logic is needed for this.