



Rasterdaten in
Secondo

Das
Betreuerteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

Rasterdaten in Secondo

Aufgabenbeschreibung Phase 2

Das Betreuerteam

Fakultät für Mathematik und Informatik
Datenbanksysteme für neue Anwendungen



FernUniversität in Hagen

15.November 2012

©2012 FernUniversität in Hagen



Outline

Rasterdaten in
Secondo

Das
Betreuerteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- 1 Einführung
- 2 Datentypen
- 3 Operationen
 - Import
 - Konvertierung
 - Einschränkung
 - Rasterinfos
 - Compose
 - Manipulation
 - Verschmelzen
- 4 Visualisierung



Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Werte können vor der räumlichen Position abhängen
z.B. Höhe der Erdoberfläche im Bezug auf den Meeresspiegel
- Solche Werte können auch zeitlich variieren
z.B. Temperaturen der letzten Woche in Deutschland
- Möglich variierende Werte sind bool, int, real, string
- Namensgebung sT bzw. msT mit
 $T \in \{\underline{bool}, \underline{int}, \underline{real}, \underline{string}\}$
- Operationen sind notwendig
- SECONDO unterstützt solche Darstellungen/Operationen bislang nicht
- Ändern Sie dies



- Werte können vor der räumlichen Position abhängen
z.B. Höhe der Erdoberfläche im Bezug auf den Meeresspiegel
- Solche Werte können auch zeitlich variieren
z.B. Temperaturen der letzten Woche in Deutschland
- Möglich variierende Werte sind bool, int, real, string
- Namensgebung sT bzw. msT mit
 $T \in \{\underline{bool}, \underline{int}, \underline{real}, \underline{string}\}$
- Operationen sind notwendig
- SECONDO unterstützt solche Darstellungen/Operationen bislang nicht
- Ändern Sie dies . . . dann gibt es den Praktikumsschein



Darstellung räumlicher Abhängigkeiten

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Raum wird durch ein gleichmäßiges quadratisches Gitter partitioniert
- Jeder Gitterzelle wird ein konstanter Wert zugewiesen
- Gitter können sehr groß werden (nicht geeignet als Attributdatentyp)
- Gitter ist definiert durch seinen Ursprung sowie die Kantenlänge – somit werden 3 reelle Zahlen benötigt
- Gitterdefinition erfolgt durch Datentyp grid2
- Zu einer Zelle gehören deren innere Punkte sowie der linke und der untere Rand ¹
- Gitterzellen können undefinierte Werte enthalten

¹Wichtig für die eindeutige Zuordnung eines Punktes zu einer Zelle



Beispiel räumlich variierender Integer (sint)

- Rasterdaten in
 Secondo
- Das
 Betreuersteam
- Einführung
- Datentypen
- Operationen
 - Import
 - Konvertierung
 - Einschränkung
 - Rasterinfos
 - Compose
 - Manipulation
 - Verschmelzen
- Visualisierung

47360	47872	47360	47872	47872	48128	48640	48384	47616	47104	47104	47360	47360
46592	45824	45056	44800	45056	46080	47104	47360	47104	46848	46848	47104	46848
42752	41984	42240	43008	43520	43776	44544	45568	46336	46592	46592	46592	46848
40704	41216	41728	42240	42496	43264	43776	44288	45312	46080	45568	44800	44544
40192	41216	41728	42240	42496	43008	43264	43264	43008	43008	43008	42752	42240
40192	40960	41728	42240	42496	42240	41984	41728	41216	40960	40704	40448	40192
40192	41216	41728	41984	41472	40960	40448	40192	39936	39424	38912	38656	38656
40960	41472	41216	40960	40704	40192	39936	39680	39168	38400	38144	37888	37632
40448	40448	40192	39936	39424	38912	38912	38144	37632	37120	36864	36352	36096



Implementierung von sint, sreal, sbool

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Gitter können sehr groß werden
- Aufteilung des Gitters in Teilgitter
- Die Größe eines Teilgitters entspricht einer Seite (z.B. 4 kB) (abzüglich einiger Verwaltungsinformation)
- Jedes Teilgitter enthält mindestens einen definierten Wert, andere Teilgitter werden nicht gespeichert.
- Ein Teilgitter wird in einem Record eines SMI-Keyed-Files mit Records fester Größe gespeichert. Als Schlüssel dienen die Zellindexe.
- Teilgittergröße für sint, 32 bit-Integer, 4 kB Seitengröße: 8 Byte als Key, $(4096 - 8) / 4 = 1022$ Integer speicherbar \Rightarrow Teilgitter: 31×31
- Verwendung von NaN als undefinierter Wert (Speicherplatz)



Besonderheit sstring

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Werte im Verhältnis zu bool , real und int sehr groß
- Häufig nur wenige verschiedene Werte (z.B. Land Use)
- Daher
 - Mögliche Stringwerte werden in einer separaten Struktur gespeichert
 - Zelle enthält einen Verweis in diese Struktur (Integer)

1	1	1
1	2	2
5	5	5

1	Acker
2	Straße
3	Gebäude
4	Wiese
5	Wald
.	
.	
.	



Weitere Felder in den Datenstrukturen

Rasterdaten in
Secondo

Das
Betreuerteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- R-Baum: indiziert vorhandene Teilgitter
- bbox: Minimal begrenzendes Rechteck des Gitters
- min/max: minimaler und maximaler auftretender Wert
- ...

- Verwenden Sie einen Cache zum Zugriff auf die Teilgitter, um häufigen Zugriff auf die Festplatte zu vermeiden



Räumlich und zeitlich abhängige Werte

Rasterdaten in
Secondo

Das
Betreuerteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Ebenfalls durch Gitter dargestellt
- Gitter hat nun 3 Dimensionen (3. Dimension = Zeit)
- Einteilung der Zeitdimension gleichmäßig, jedoch unabhängig von räumlicher Zellgröße
- Gitterdefinition erfolgt über grid3 (zusätzlich zu grid2: duration Wert)
- Zelle kann auch hier nur konstanten Wert enthalten
- Implementierung analog zu rein räumlich abhängigen Typen (einschließlich Sonderbehandlung für string)



Import von Rasterdaten

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Gitter sollen aus (Mengen von) Dateien² erstellt werden können
- Zu unterstützende Formate:

HGT eine Datei stellt den Bereich eines Breiten- und eines Längengrads dar

Dateiname kodiert den Bereich, z.B.
N51E007

Datei selbst enthält nur noch Integer
Zahlen

ESRI GRID Beschreibung siehe Webseite in der
Aufgabenstellung

ESRI ASCII raster

²dargestellt als Strom von Dateinamen



Konvertierung von und in räumliche Datentypen

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Räumliche Datentypen line und region aus SpatialAlgebra
- Region kann zusammen mit Gitterdefinition in sbool konvertiert werden
- Zelle erhält Wert `true`, wenn ihr Mittelpunkt im Gebiet liegt
- Ebenso kann ein sbool in eine Region umgewandelt werden
basiert auf Zyklensuche
- line kann auch in eine sbool umgewandelt werden
Zelle erhält Wert true, wenn Zelle von Linie geschnitten wird



Operator *fromregion*

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

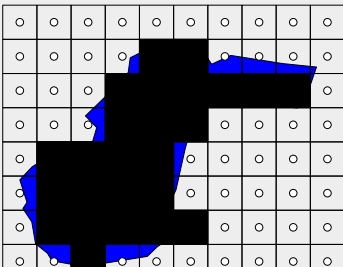
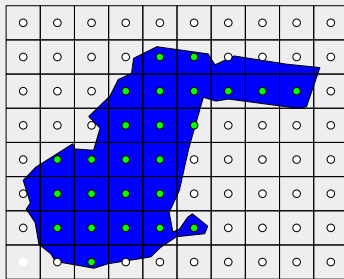
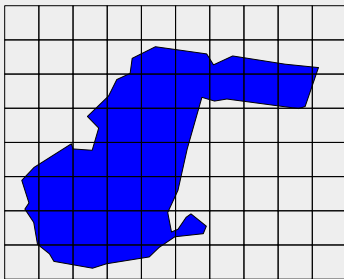
Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung





Rasterdaten in
Secondo

Das
Betreuerteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Es ist nicht effizient, jede Zelle, die das MBR der Region schneidet zu prüfen
- Idee: Verwendung des ScanLine-Algorithmus (Computergraphik)
- Durchlaufe Zellen zeilenweise
- Grenze der Region schneidet Zelle: prüfen auch nachfolgende Zelle prüfen (falls zwei Segmente in Zelle liegen)
- Umschalten zwischen `true` und `false`



Operator *fromline*

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

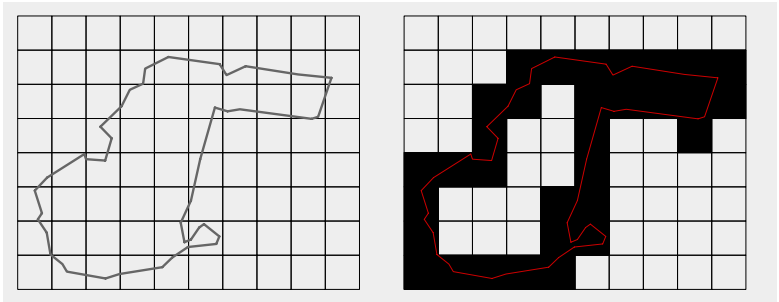
Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung



- Prüfe nicht jede Zelle auf Schnitt mit Linie
- Stattdessen markiere Zellen, die von Segmenten geschnitten werden mit `true`



Rasterdaten in
Secondo

Das
Betreuerteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Zellränder mit verschiedenen Werten in beiden angrenzenden Zellen bilden Segmente der Grenze der Region
- Segmente sind soweit wie möglich zusammenzufassen
- Baue zunächst Linie, die Grenze der Region enthält
- Class `Line` enthält Funktion `Transform` für Umwandlung in *region*



Der Operator *s2ms*

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- schwierig zeitlich abhängige Rasterdaten zu finden, aber z.B. DWD stellt Rasterdaten für verschiedene Zeitperioden bereit
- Operator *s2ms* wandelt rein räumliches Raster in *ms* Objekt um
- Neben Rasterobjekt erhält der Operator
 - *duration* : legt Zeitauflösung des Gitters fest
 - Zeitintervall (in Form zweier *instant*-Objekte): legt die Gültigkeit fest
- Jede Gitterzelle des Ergebnisses, die mindestens zur Hälfte vom Argumentzeitintervall überdeckt wird, erhält den Wert der ursprünglichen Gitterzelle; andere Zellen sind undefiniert.



Gitter beschränken

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

atlocation Liefert den Wert eines Rasterobjekts an einem gegebenen Punkt ³ (dieser Wert ist ggf. ein moving Typ)

atrangle Schränkt ein Rasterobjekt auf ein geg. Rechteck ein ⁴
bei auch zeitlich abhängigen Werten, kann optional noch eine Einschränkung in der Zeitdimension erfolgen (zwei Zeitpunkte als weitere Argumente)

³Ausrechnen des Zellindex

⁴Suche über R-Baum



Gitter zeitlich beschränken

Rasterdaten in
Secondo

Das
Betreuerteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- atinstant* Schränkt ein räumlich und zeitlich abhängiges Rasterobjekt auf einen Zeitpunkt ein
Das Ergebnis ist ein Paar bestehend aus diesem Zeitpunkt und einem rein räumlich abhängigen Rasterobjekt
- atperiods* Schränkt ein Rasterobjekt auf diejenigen Zellen ein, deren Zeitintervalle eine geg. Menge von Zeitintervallen schneiden



Informationen über Rasterobjekte

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

getgrid Liefert das einem Rasterobjekt zugrundeliegende Gitter (*grid2* oder *grid3*)

deftime Liefert die Menge derjenigen Zeitintervalle, in denen Zellen definierte Werte besitzen

bbox Ermittelt das minimale begrenzende Rechteck der Zellen mit definierten Werten (rein räumlich)

minimum Ermittelt den minimalen definierten Wert, der im Gitter gespeichert ist

maximum Ermittelt den maximalen definierten Wert, der im Gitter gespeichert ist



- Verbindet ein rein räumlich abhängiges Rasterobjekt $\underline{s}T$ mit einem *m*point p
- Ergebnis ist ein *m* T , d.h. die Werte der Zellen während der Bewegung von p



Compose – Beispiel

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung



Hallo Secondo-Team,

ich bin Klara das Rentier
und gerade in Alaska unterwegs.
Ich habe immer meinen GPS-Tracker
dabei. Damit weiß ich stets,
wo ich mich gerade befunden habe.

Jetzt möchte ich aber
wissen, wie hoch ich wann war . . .
Könnt Ihr mir dabei helfen ?

Liebe Grüße Klara



Antwort an Klara

Rasterdaten in
Secondo

Das
Betreuerteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

Liebe Klara,

leider können wir Dir erst im Februar helfen, wenn der *compose*-Operator fertiggestellt ist. Dann können wir nämlich:

- Die Höhe der Erdoberfläche von Alaska in ein Rasterobjekt lesen.
- Den *compose* -Operator auf dieses Raster und Deine Bewegung anwenden.
- Und schon hast Du genau das, was Du möchtest.

Liebe Grüße – das Secondo-Team



Berechnung neuer Zellinhalte

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

map Wendet für ein geg. Rasterobjekt (s*T* oder ms*T*) eine gegebene Funktion ($T \rightarrow U$) für jede Zelle mit definiertem Inhalt an.

- Das Ergebnis ist vom Typ s*U* bzw. ms*U*
- Beispiel: *H* sei ein Gitter (sreal), das die Schneehöhe eines bestimmten Gebiets beschreibt
Definiere $f : \text{real} \rightarrow \text{bool}$ als $f(x) = x > 1.0$
dann liefert $H \text{ map } [f]$ die Bereiche mit einer Schneehöhe von mehr als einem Meter als sbool

- Secondo-Query:

```
query H map[ fun(x : real) x > 1.0] toregion
```

- Kurzschreibweise:

```
query H map[ . > 1.0] toregion5
```

⁵Über implizite Parameter und TypeMap-Operatoren realisierbar



- Implizite Parameter kennen Sie schon aus den Spec-Files.
- TypeMap-Operatoren
 - Besitzen kein ValueMapping (auf 0 gesetzt)
 - Erhalten im TypeMapping die Argumente des umgebenden Operators
 - Erhält hier ein $\underline{s}T$ oder $\underline{ms}T$ und weitere Argumente und sollte auf T abbilden
 - Vorhandene Beispiele: *TUPLE*, *STREAMELEM*, *STREAMELEM2*, *GROUP*



Verschmelzen zweier Raster: *map2*

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Verbindet Zellen zweier Raster mittels einer binären Funktion
- Diese wird angewandt, wenn mind. eine der beteiligten Zellen definiert ist
- Funktioniert nur, wenn Gitterdefinitionen miteinander verträglich sind
 - räumlicher Ursprung eines Gitters muss auf Zellenecke des anderen Gitters liegen (und umgekehrt)
 - räumliche Gittergröße muss gleich sein
 - wenn beide Rasterobjekte auch zeitlich abhängig sind, müssen zusätzlich die Zeitdauern für die Zellen übereinstimmen.
- Verschmelzen verschiedener Gitter über weiteren Operator möglich



map2 – Beispiel

Rasterdaten in
Secondo

Das
Betreuerteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Beispiel: $H1$ und $H2$ seien Schneehöhen an verschiedenen Zeitpunkten:

```
query H1 H2 map2 [ . > .. ]
```

liefert sbool, bei denen die Höhe in $H1$ größer als die in $H2$ ist

- weiterer TypeMap Operator notwendig



Operator *matchGrid*

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Passt ein Rasterobjekt einer neuen Gitterdefinition an
- Welcher Wert soll verwendet werden, wenn mehrere alte Zellen eine neue Zelle überlappen?
- Anwendungsabhängig \Rightarrow wir überlassen diese Entscheidung dem Benutzer
- Funktion, die eine Menge von Werten zu einem Wert aggregiert
- Ggf. kann auch noch die Größe der Überlappung der Zelle als Gewicht verwendet werden (Summe aller Gewichte ist 1, nur bei numerischen Basistypen)
- Aus technischen Gründen wird diese Menge als Relation dargestellt.



matchgrid – Beispiel

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

07300	07372	07380	07372	07372	08128	08343	08384	07916	07104	07104	07380	07380	
08000	08014	08000							08018	08018	07104	08018	
02702	01864	02240			02002	02776	02844	02000	02000	02000	02000	02000	
02704	02216	02728			02486	02264	02776	02200	02000	02000	02000	02000	
02702	02216	02728			02486	02000	02264	02264	02000	02000	02702	02200	
02702	02000	02728			02486	02240	01864	02728			02000	02702	
02702	02216	02728							02422	02712	02000	02000	
02000	01472	02216	02000	02702	02702	02000	02000	02422	02422	01864	02000	02702	
02000	02000	02702	02000	02000	02712	02712	02422	02422	02702	02000	02000	02000	

- Für jede Zelle des neuen Gitters, die definierte Werte des alten Gitters schneidet:
 - Speichere alle (evtl. gewichteten) Werte des alten Gitters (schwarz), die die neue Zelle (rot) überlappen, in eine Relation (TupleBuffer)
 - Wende benutzerdefinierte Funktion auf Relation an.
 - Weise den erhaltenen Wert der neuen Gitterzelle zu
 - Ergebnistyp hängt von Funktion ab



- `query matchgrid(H, grid1, cells feed max[Elem])`
- Verwendet das Maximum der überlappten Zellen als neuen Wert
- Weiterer TypeMapOperator *CELLS* notwendig
- Bildet $\underline{s}T, \dots$ bzw. $\underline{ms}T, \dots$ auf $\underline{rel}(\underline{tuple}([Elem : T]))$ ab



Rasterdaten in
Secondo

Das
Betreuerteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Mittels Displayklassen des HöseViewers
- Verschiedene Werte in den Zellen durch Farbunterschiede oder Label (falls genug Platz) darstellen
- Auch für zeitlich abhängige Objekte
- Insgesamt 8 Displayklassen



Abschließende Hinweise

Rasterdaten in
Secondo

Das
Betreuersteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

- Achten Sie auf eine effiziente Listendarstellung (Übertragungszeiten!)
- Eine persistente Implementierung eines R-Baums finden Sie in der `RTreeAlgebra`
- Eine Hauptspeicherbasierte Implementierung eines R-Baums finden Sie in `include/MMRTree.h`
- Eine LRU-Cache-Implementierung finden Sie in `include/LRU.h`
- Die Schnittstelle zu SMI-Keyed-Files finden Sie in `include/SecondoSMI.h`
- Ein `DateTime`-Objekt (*instant*) kann man in einen `double` umwandeln und umgekehrt (Aufbau eines 3D-R-Baums)



Rasterdaten in
Secondo

Das
Betreuerteam

Einführung

Datentypen

Operationen

Import

Konvertierung

Einschränkung

Rasterinfos

Compose

Manipulation

Verschmelzen

Visualisierung

Viel Erfolg bei der Bearbeitung
der Aufgabe!