



MOPSI



UNIVERSITY OF
EASTERN FINLAND

API for Clustering Geo-References Data on Maps: 1M Objects in 1s

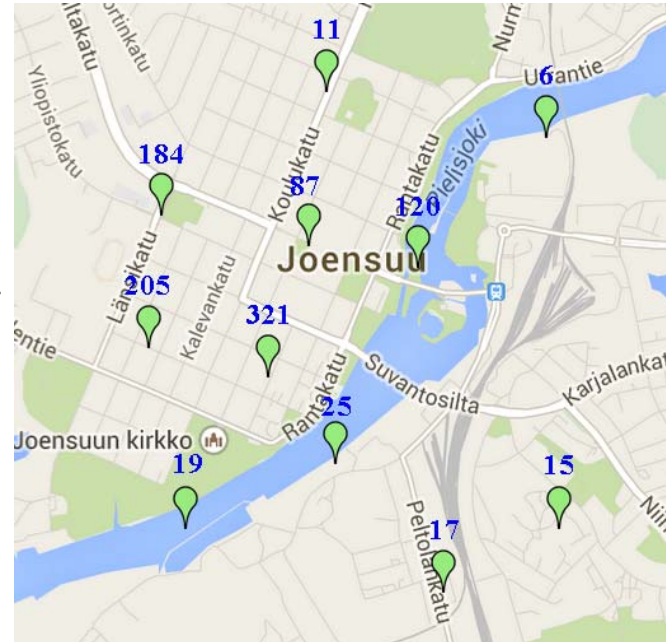
Mohammad Rezaei, Pasi Fränti
Speech and Image Processing Unit
School of Computing

23.1.2015

TOO MANY MARKERS



Clustering

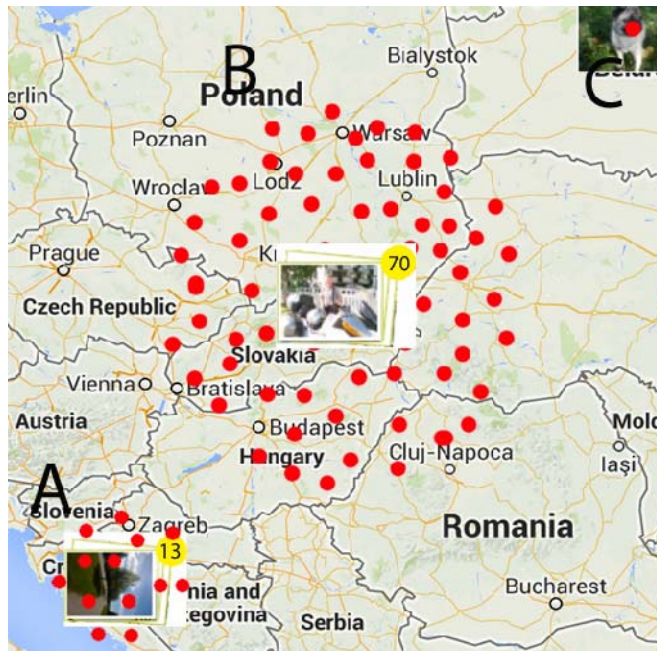


Problems:

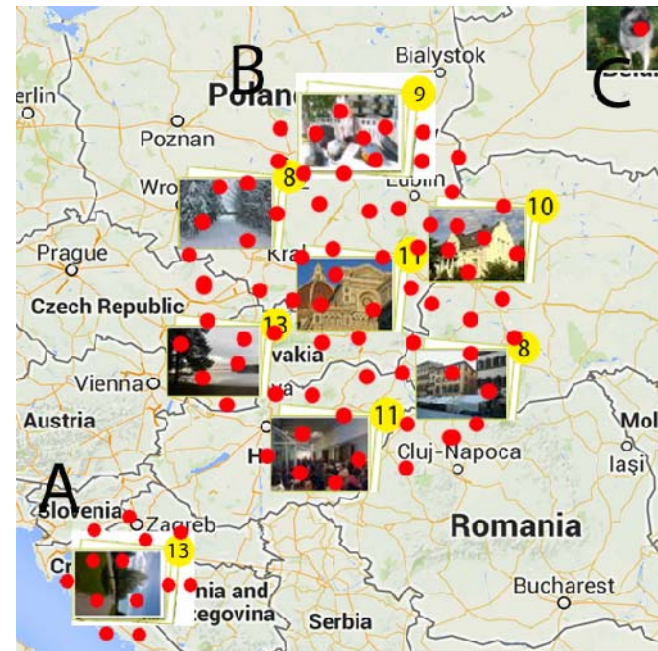
- Interaction
- Covering Map

DIFFERENCE WITH NORMAL DATA CLUSTERING

Normal clustering



Clutter removal

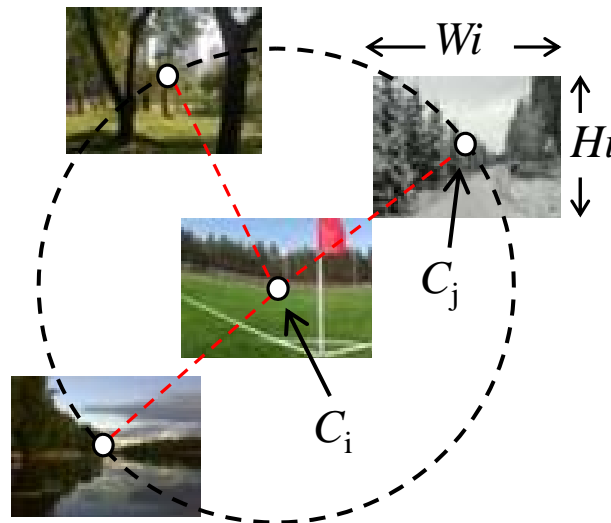


CLUSTERING PROBLEM

Cluster representative: icon with the size (W_I, H_I)

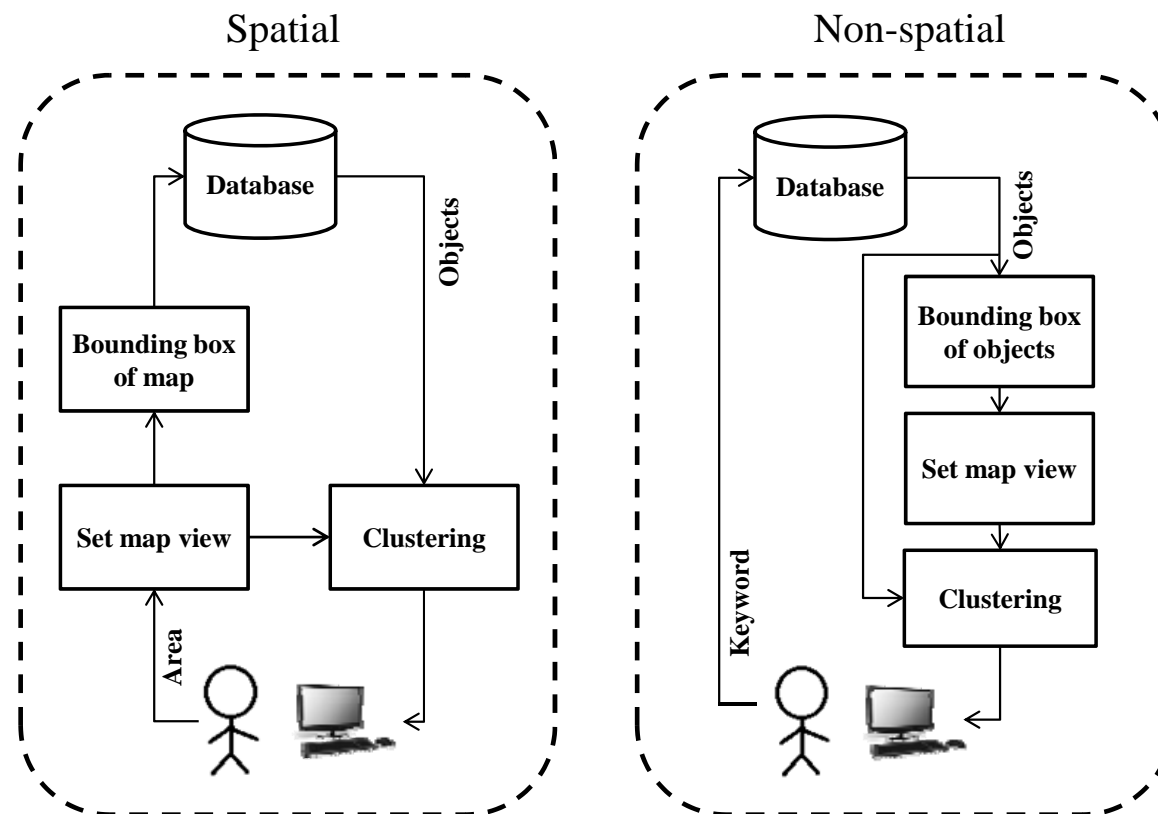
There can be as many as clusters without overlap of their representatives:

$$\|C_i - C_j\| \geq T \times \sqrt{W_i^2 + H_i^2}, T \geq 1$$



NON-SPATIAL AND SPATIAL QUERIES

- **Non-Spatial:** query for other attributes of the objects rather than location
- **Spatial:** query for the objects in a specified region, given a set of data



CLUSTERING STRATEGIES

Client-side clustering

- All results (1M) are sent to client (bandwidth)
- Clustering is performed on client

Server-side clustering

- Only summary of clusters are sent to client
- Spatial query: pre-clustering of entire data
- Non-spatial query: need on-demand clustering

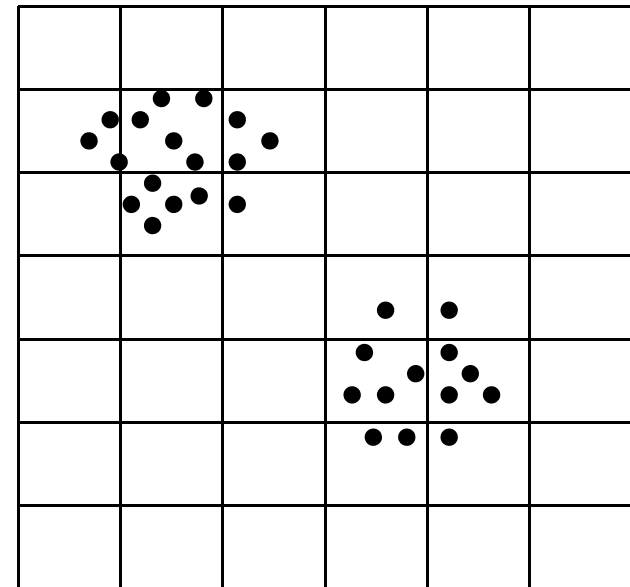
MARKER CLUSTERING API

- Grid-based clustering
- Server-side approach

GRID-BASED CLUSTERING

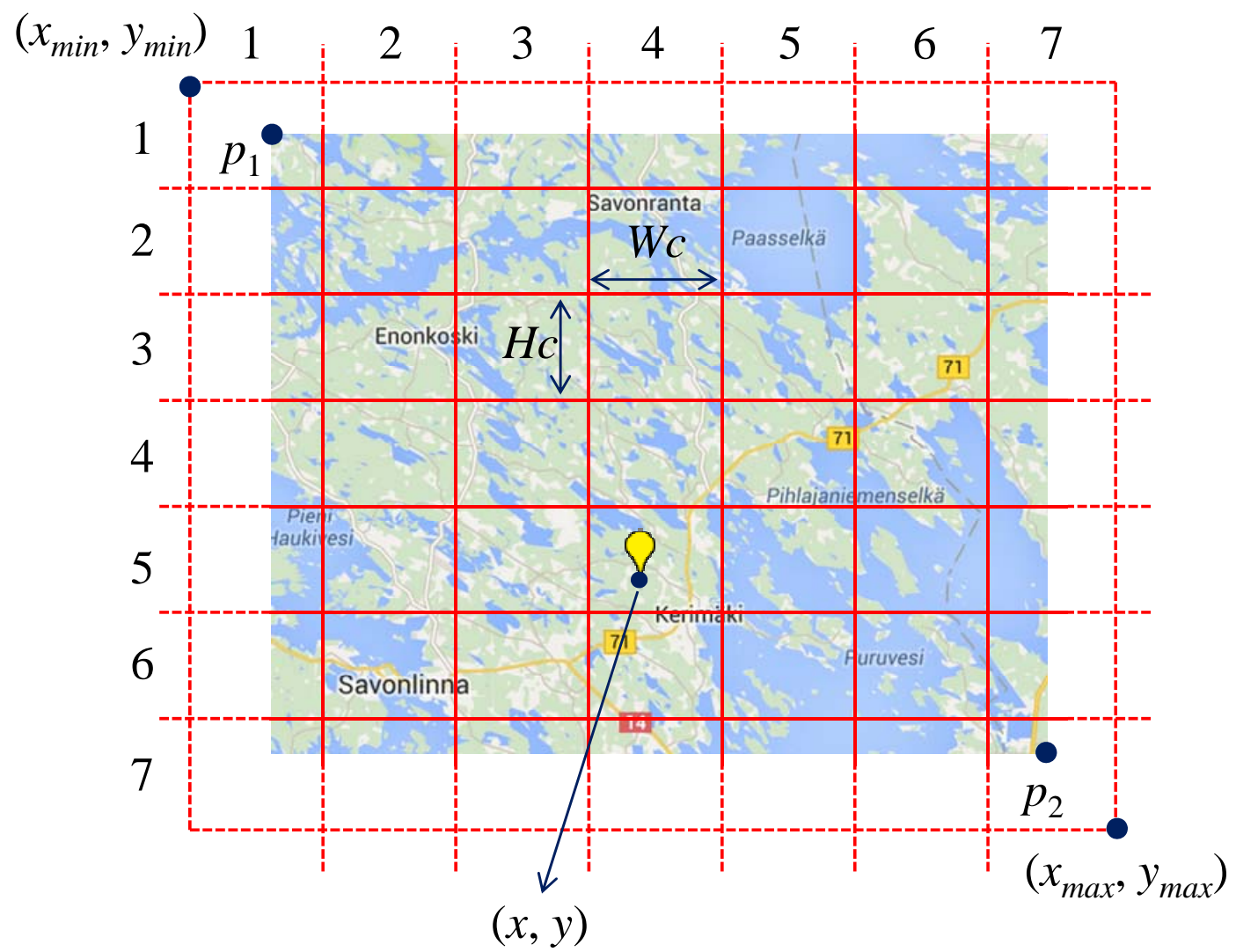
Three steps

1. Grid construction
2. Assign objects to cells
3. Merge neighbors based on a criteria to form final clusters



GRID CONSTRUCTION

- Fixed grid for the entire world
- Panning → all objects remain in the same cell



ASSIGNMENT

Row and column of the for an object at (x, y) :

$$row = \left\lceil \frac{y - y_{\min}}{H_c} \right\rceil$$

$$column = \left\lceil \frac{x - x_{\min}}{W_c} \right\rceil$$

Cells that contain any objects become initial clusters. From each cluster we store:

- Average location of objects (centroid)
- Number of objects (N)
- Bounding box to cover the area of objects

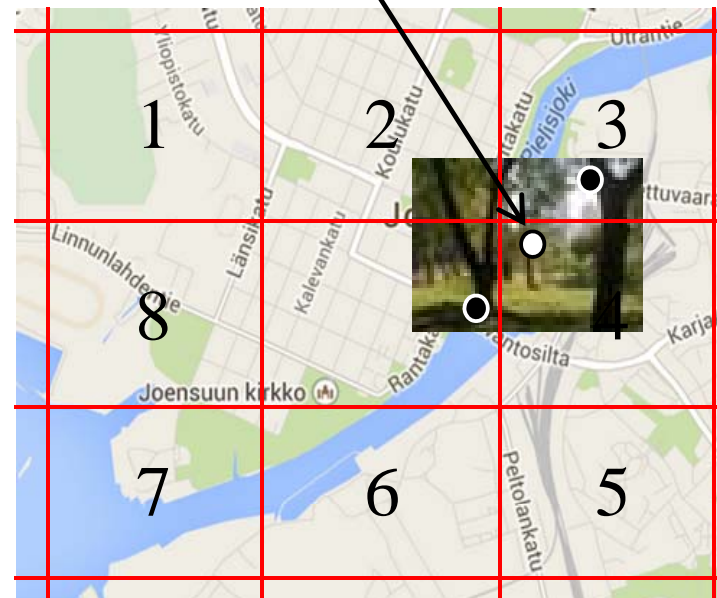
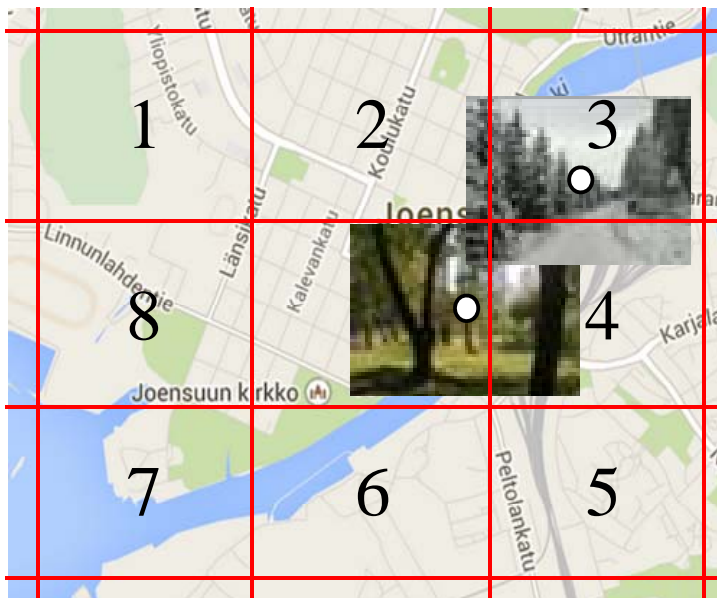
MERGING OVERLAPPING CELLS

Checking overlap of representative icons

Consider a cell and its 8 neighbors

Merging clusters: $x = \frac{n_1 x_1 + n_2 x_2}{n_1 + n_2}$ $y = \frac{n_1 y_1 + n_2 y_2}{n_1 + n_2}$ $n = n_1 + n_2$

Centroid after merge



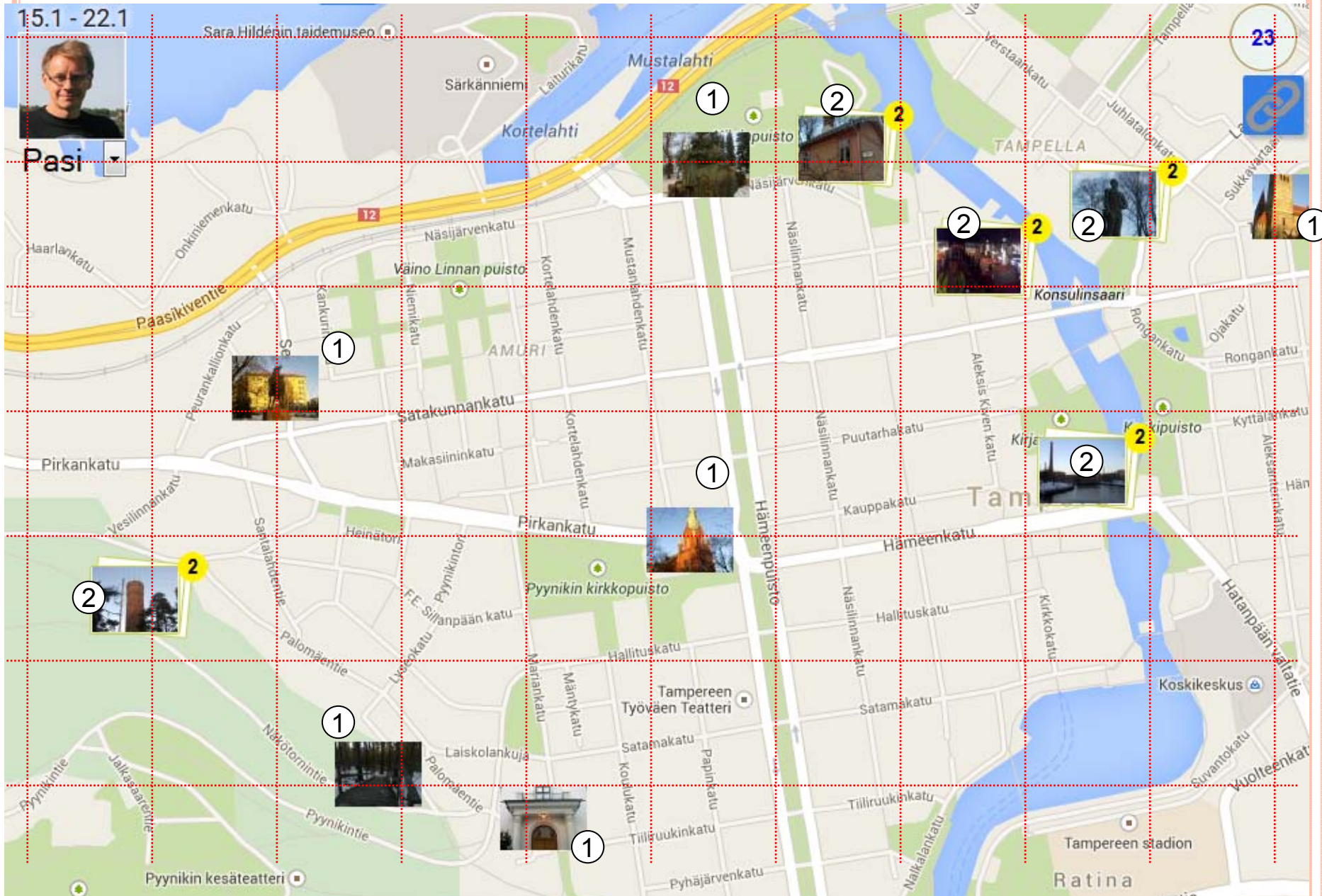
COVID BASED CLUSTERING



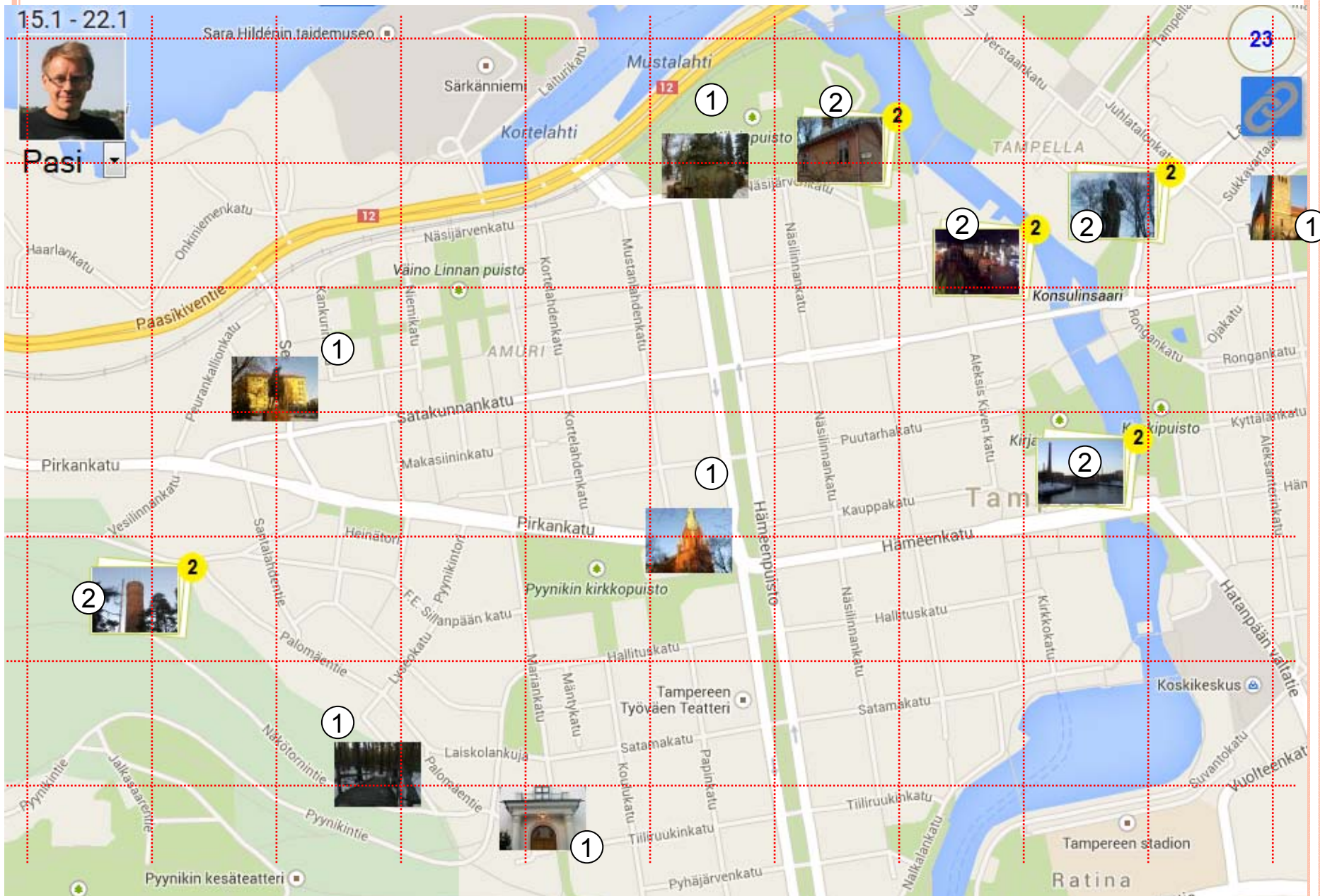
STEP 1: CONSTRUCTING GRID



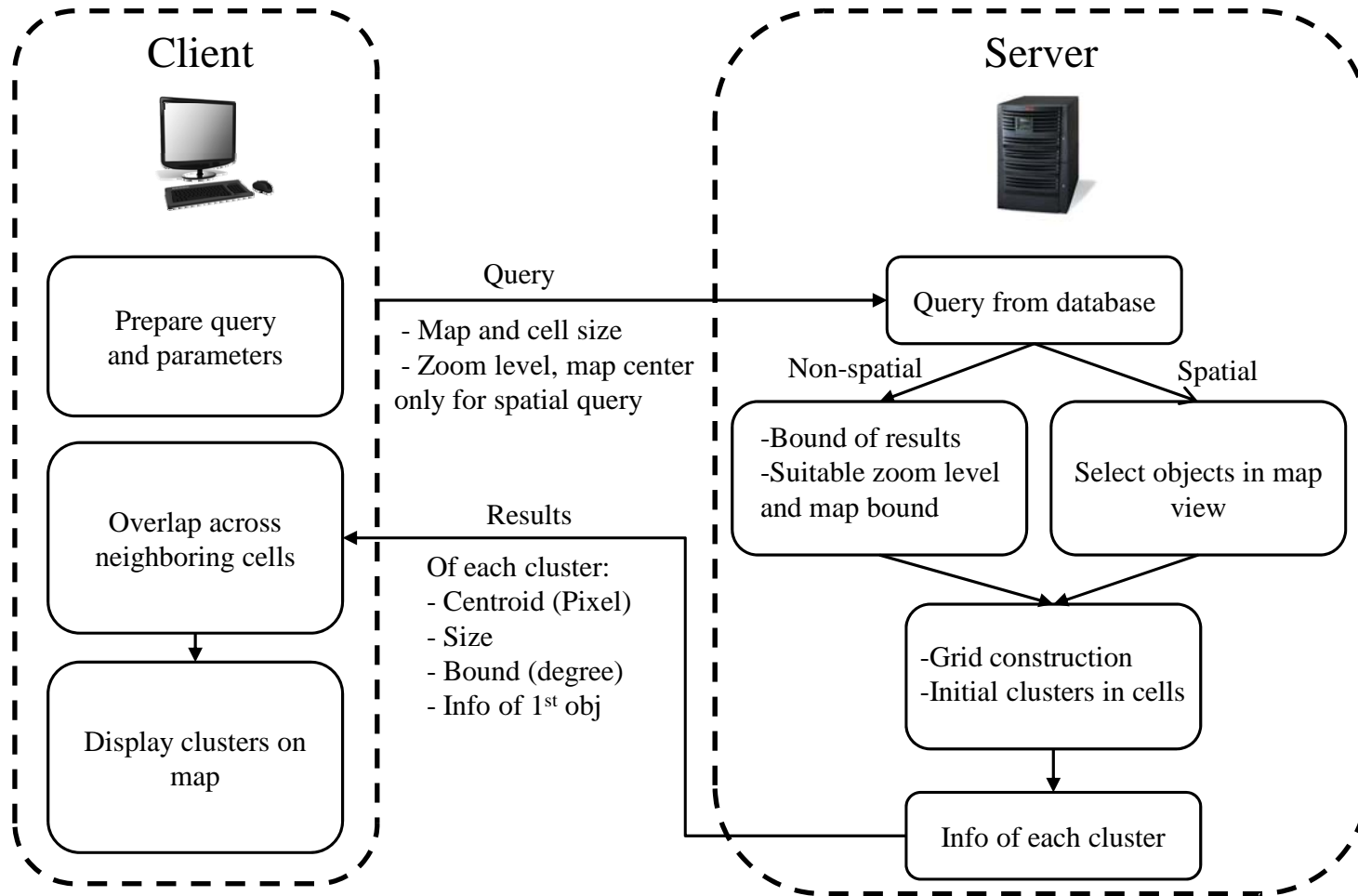
STEP 2: ASSIGN OBJECTS INTO CELLS



STEP 3: MERGNIG OVERLAPPING CELLS



SERVER-SIDE APPROACH

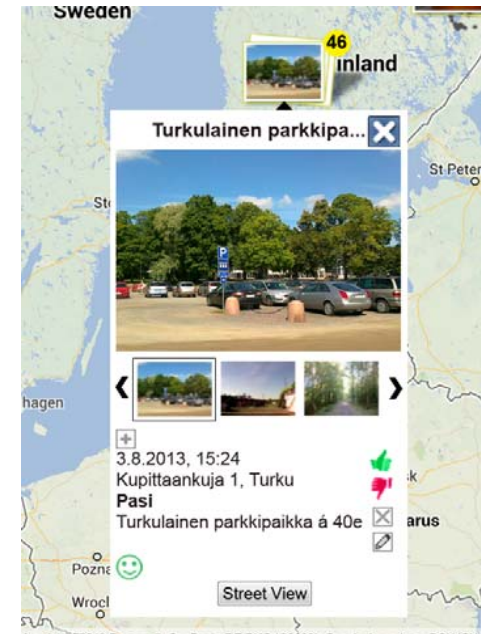
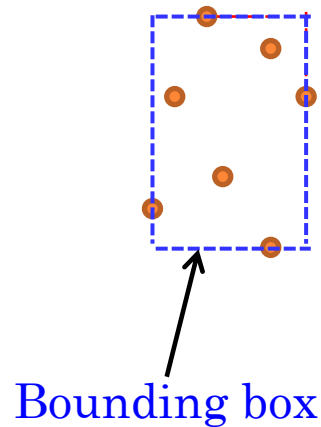


ACCESS TO OBJECTS IN THE CLUSTER

Goal: Show m^{th} object

Steps:

1. Spatial query using the bounding box of a cluster
2. Get the *id* of m^{th} object
3. Make query using *id*
4. Send the data to client

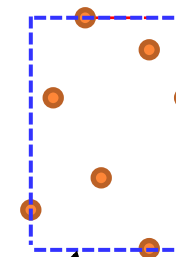
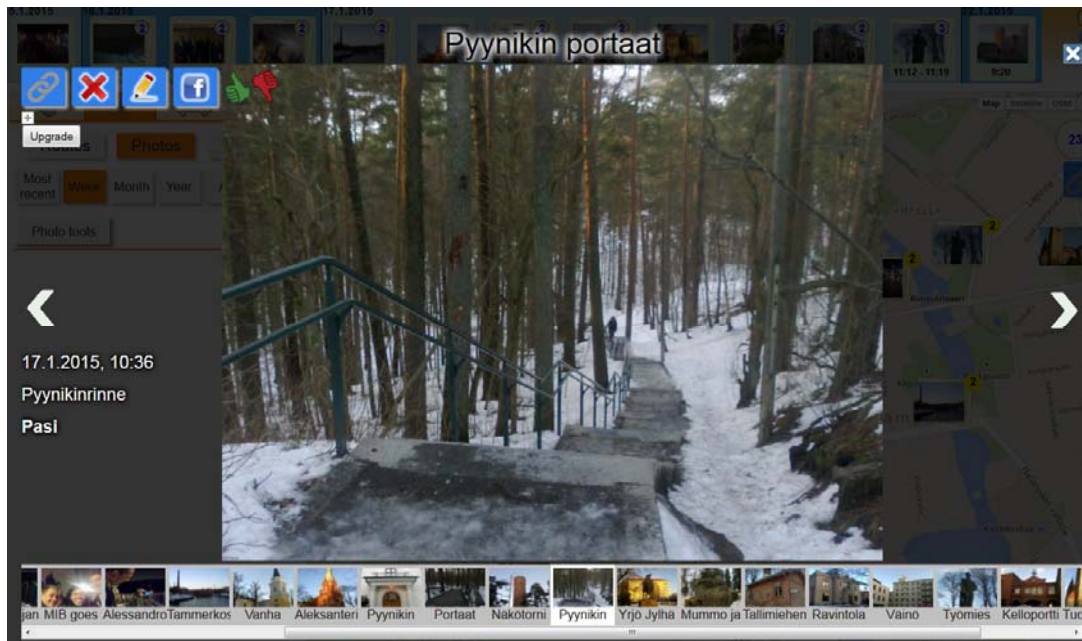
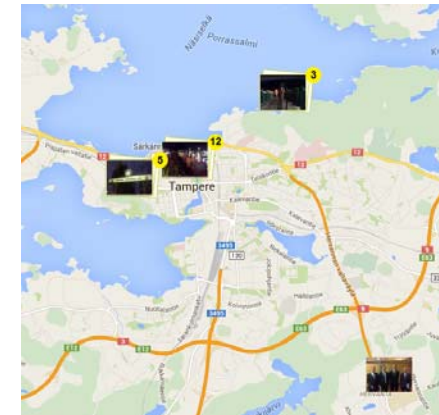


ACCESS TO OBJECTS IN CLUSTER

Goal: Show m^{th} object

Steps:

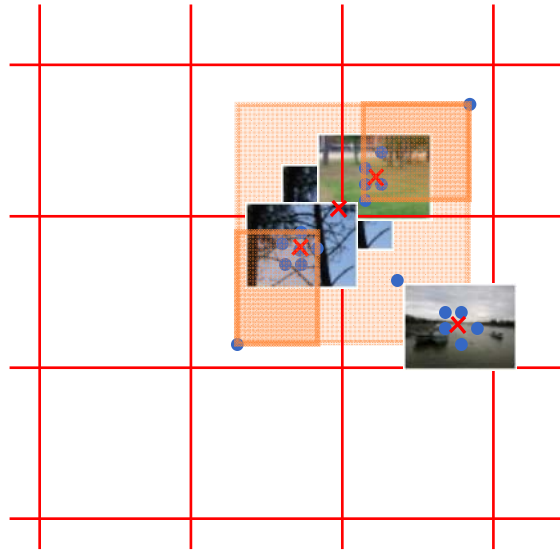
1. Spatial query using the bounding box of a cluster
2. Get the *id* of m^{th} object
3. Make query using *id*
4. Send the data to client



Bounding box

BOUNDING BOX OF MERGED CLUSTER

Issue: bounding box of a merged cluster might contain objects from other clusters



Solution: Use bounding boxes of initial clusters cells

CONCLUSION

- Freeware implementation using C code
- Very fast: 1M objects in **0.23** seconds!
- Download size is naturally limited by screen size

Demo

- http://cs.uef.fi/paikka/rezaei/markerClustering_paper/markerClustering_test/