GWT
The Google Web Toolkit

Jeppe R. Thomsen
jenslyn@cs.aau.dk

Aalborg University
Department of Computer Science

January 22, 2010

Content Of The Thesis

Two novel privacy-aware proximity detection approaches:

- FriendLocator (FL)
  - Article of 18 pages. Submitted to SSTD 2009
  - Article of 6 pages. Accepted to SSTD 2009
- VicinityLocator (VL)
  - Article is not yet submitted

Privacy-Aware Proximity Detection Services

Functions:

- Track objects in the real-time.
  - Positioning technology, e.g. GPS, Wi-Fi
  - Wireless communication, e.g. GPRS, 3G
- Determines if two objects are close to each other.
  - Notifies users
  - Generates event
- Guarantees that object exact location is secured.
  - Prevent crime possibilities

Applications:

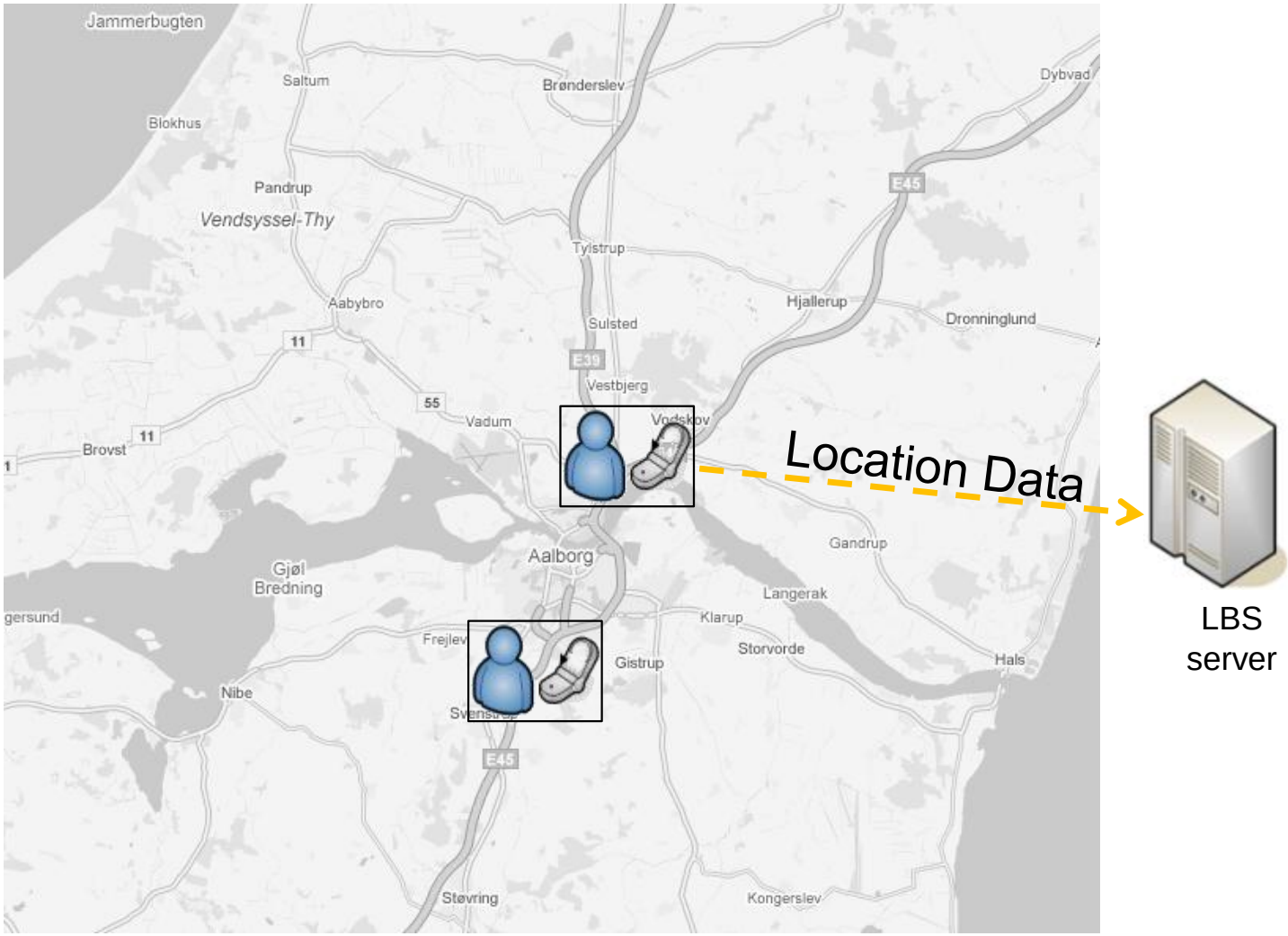
- Nearby friends identification in mobile social networks
- Collision detection
- Advertising

Example of the Proximity Detection

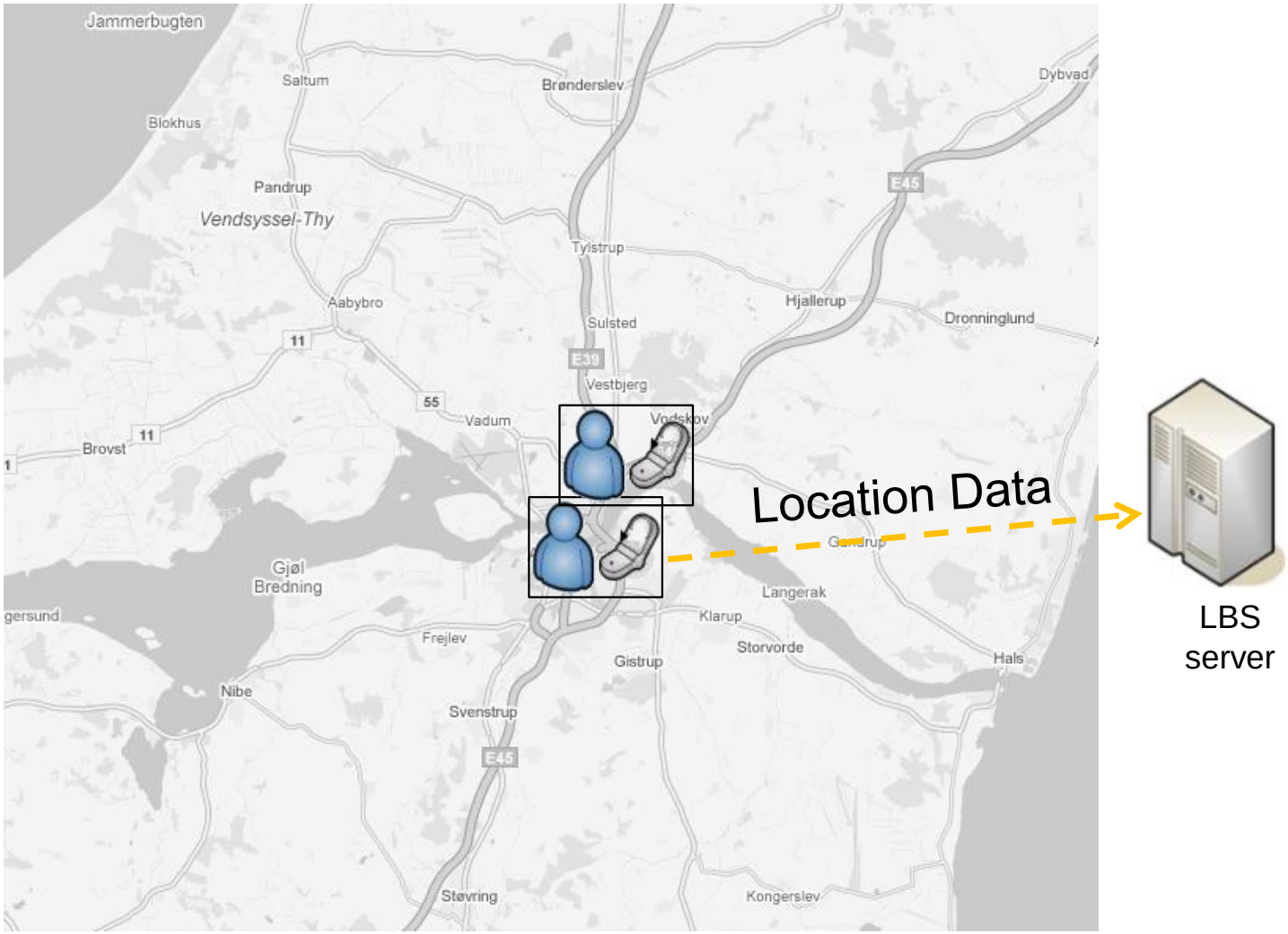
LBS server

Example of the Proximity Detection

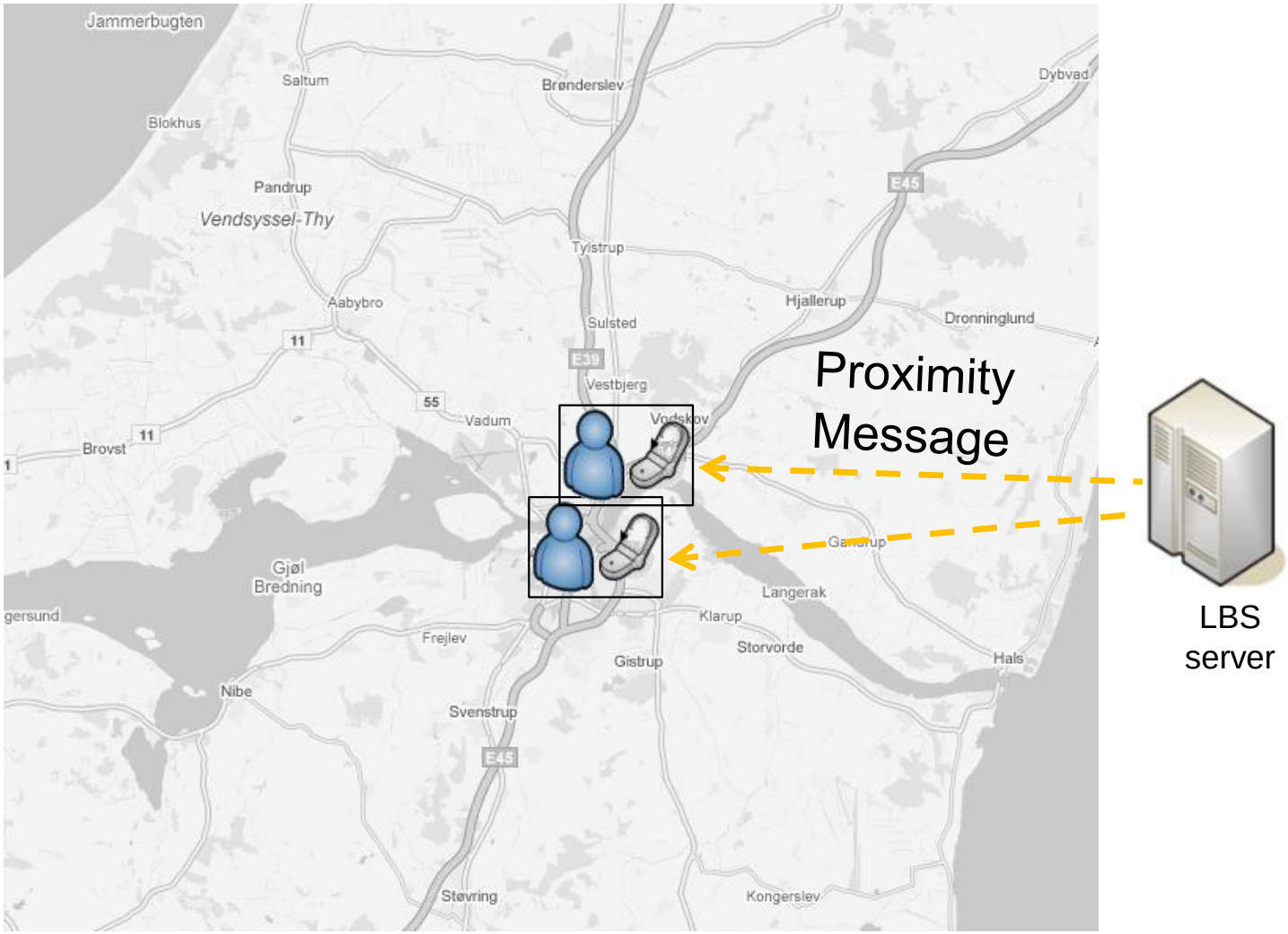
Example of the Proximity Detection



Example of the Proximity Detection



Example of the Proximity Detection



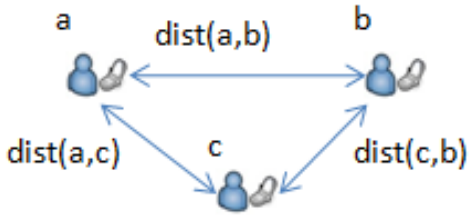
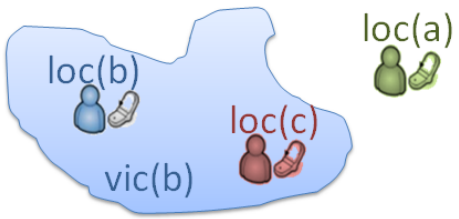
Our vs. Existing Approaches

FL&VL versus existing privacy-aware proximity detection approaches:

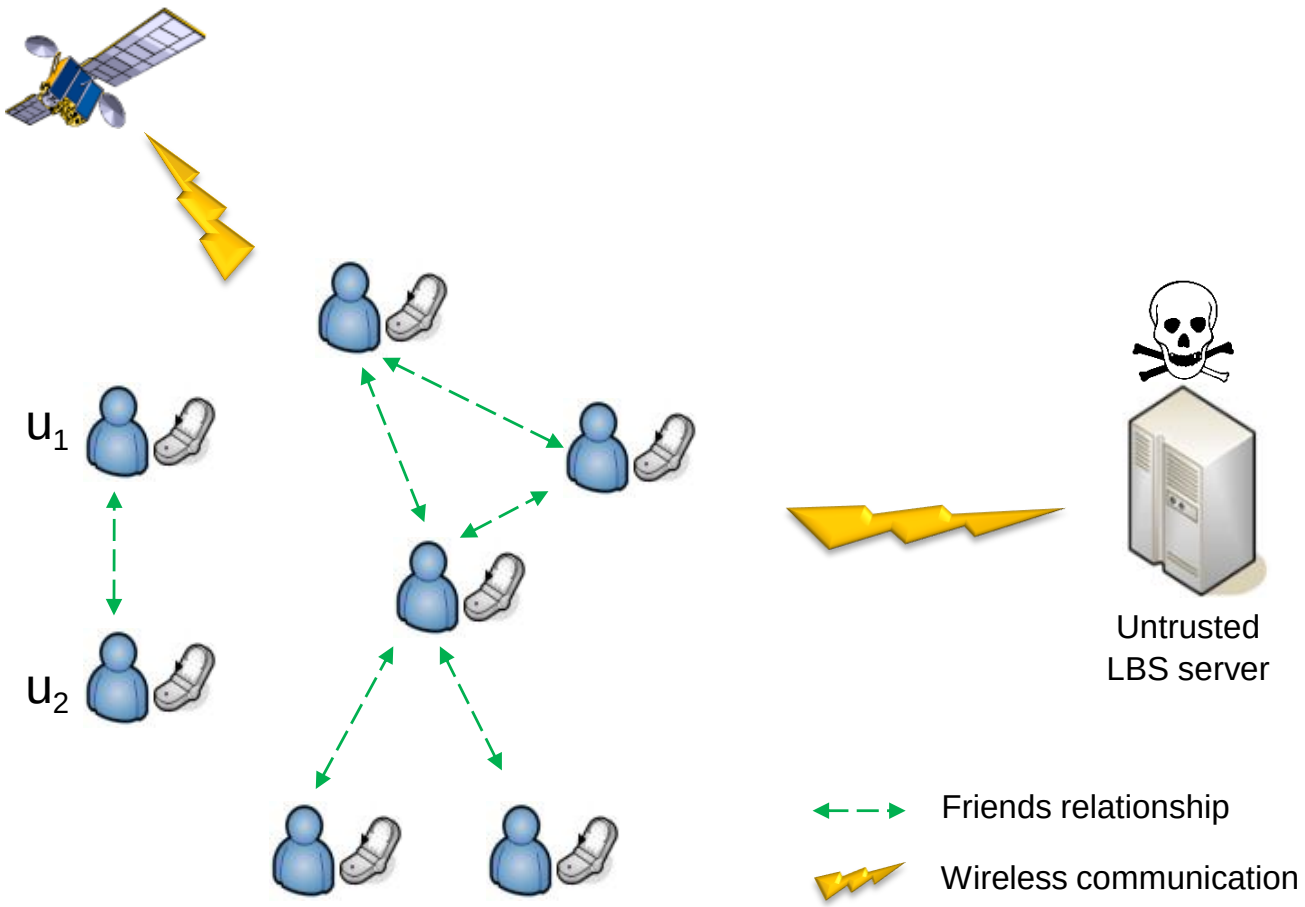
Feature	**FL&VL**	**Solution 1**	**Solution 2**
P2P comm.			+
Client-server comm.	+	+	+
Strength of privacy	strong	weak	average
User settings flexibility	high	average	average

FL vs. VL

Differences between FRIENDLOCATOR and VICINITYLOCATOR:

Feature	**FriendLocator**	**VicinityLocator**
Notion of proximity	$dist(a,b) < \epsilon$ 	$loc(a) \in vic(b)$ 
User settings	$\forall(a,b) : \epsilon_{a,b}$	$\forall a : vic(a)$, $\lambda(a)$,
Quality of service (precision)	Depends on $\epsilon_{a,b}$	$min(\lambda(a), \lambda(b))$
Communication is traded for	ϵ and precision	Precision

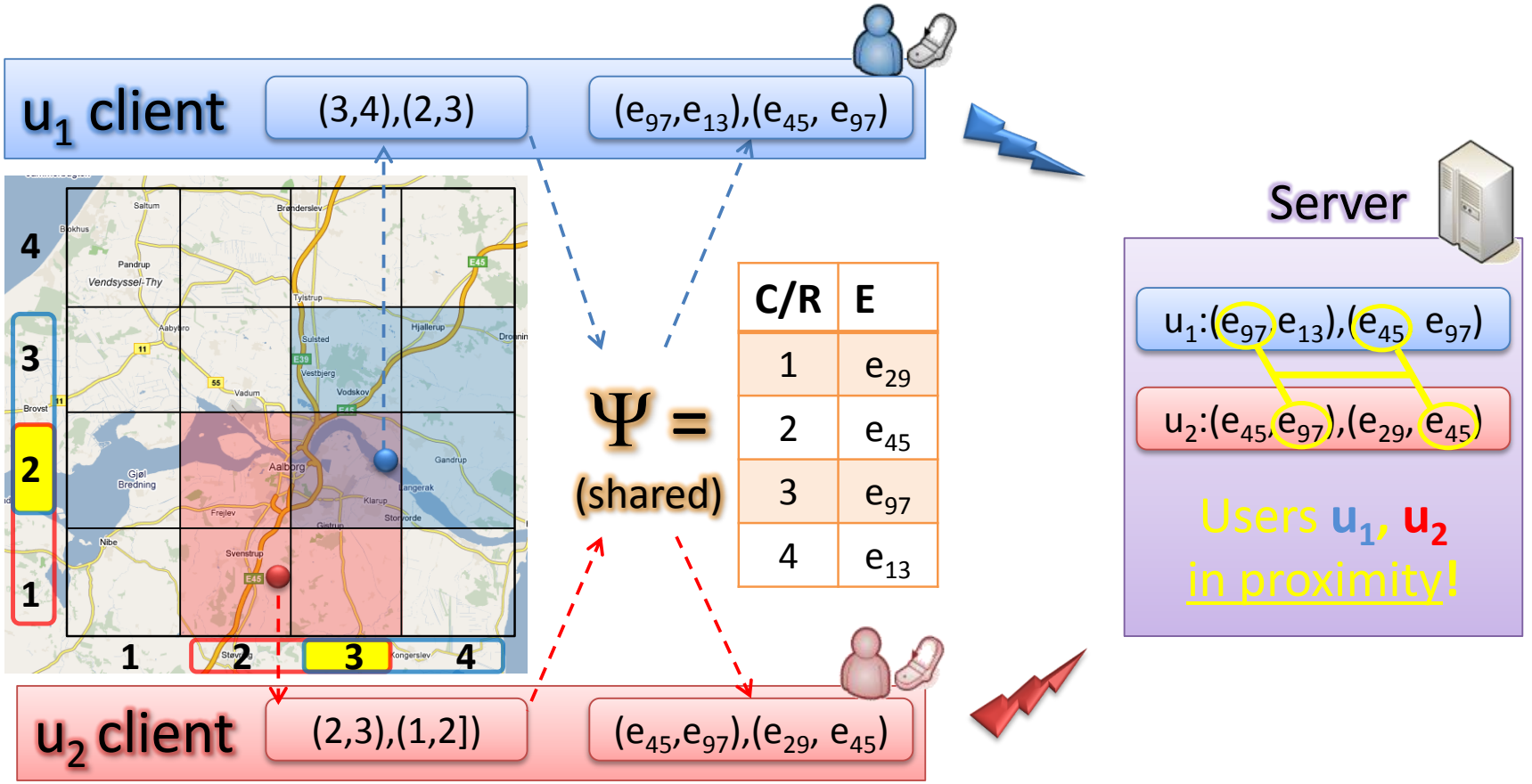
Problem setting



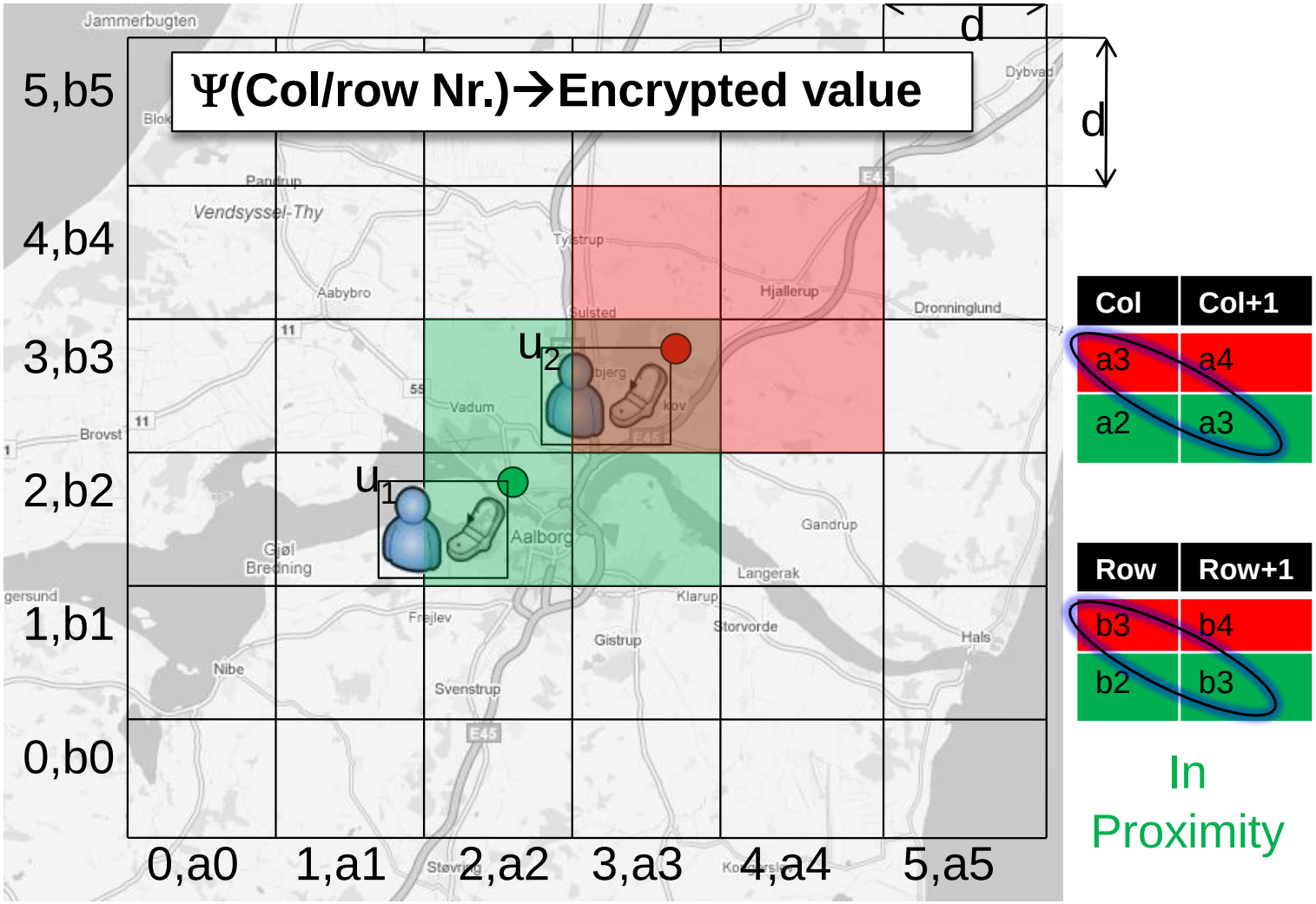
FriendLocator

FriendLocator - A Location Privacy Aware Friend Locator

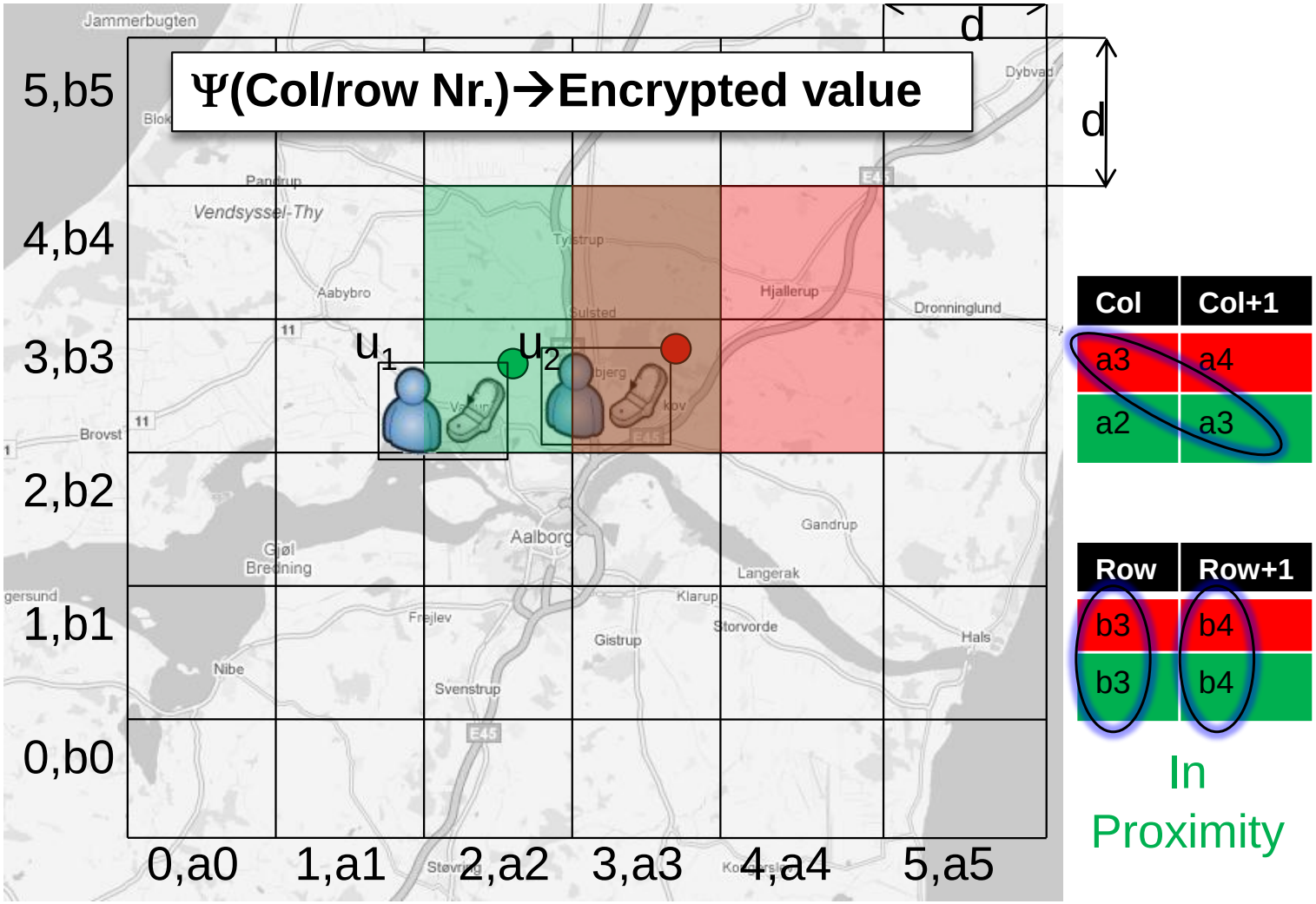
Core idea of FriendLocator



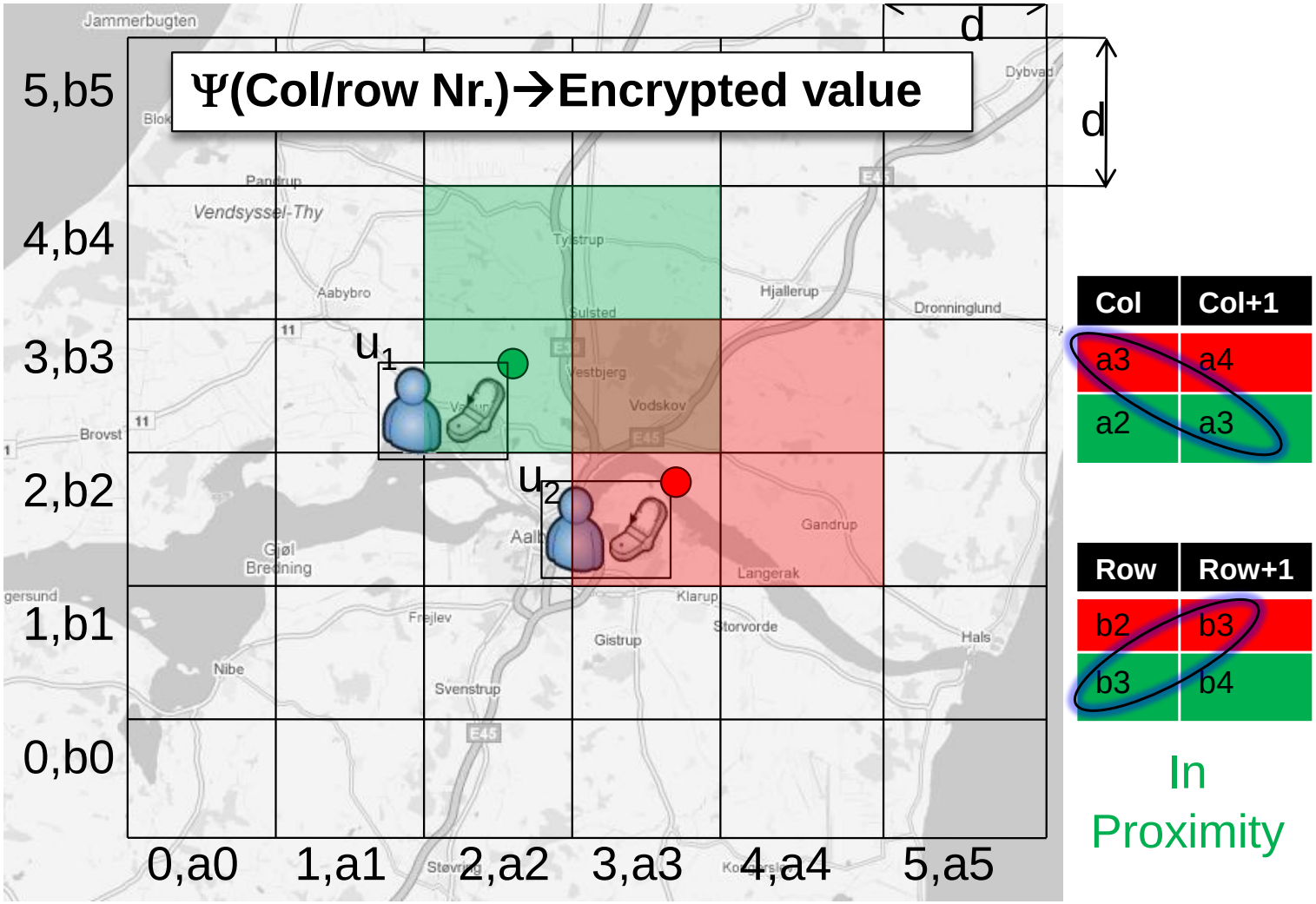
Core idea of FriendLocator



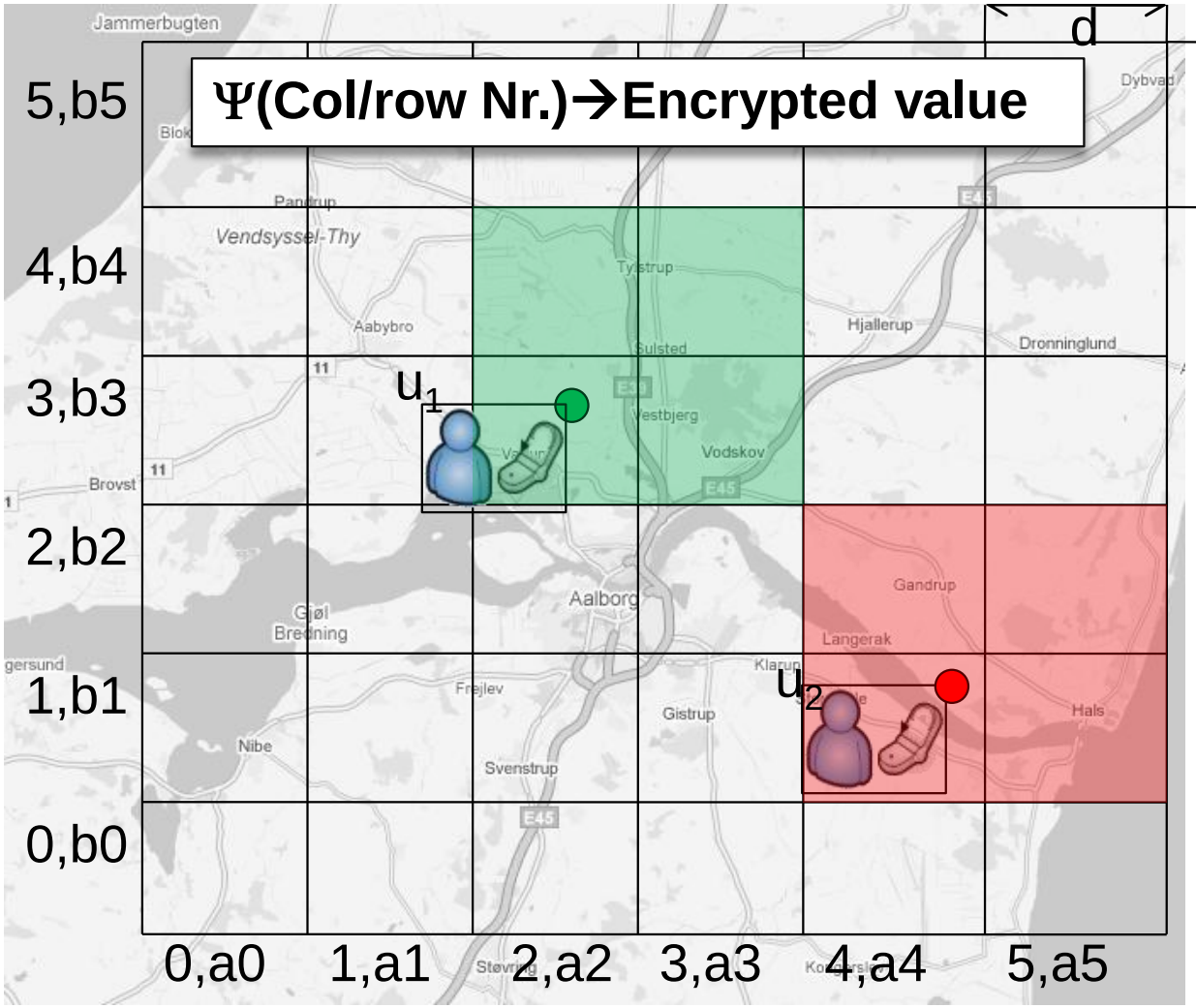
Core idea of FriendLocator



Core idea of FriendLocator



Core idea of FriendLocator



Col	Col+1
a3	a4
a2	a3

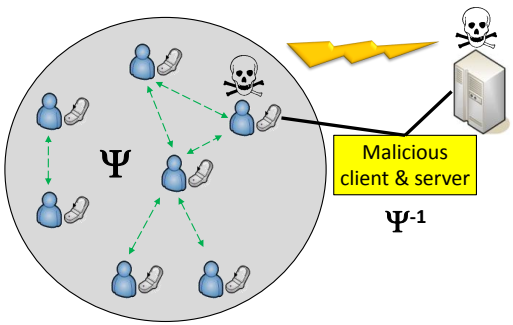
Row	Row+1
b1	b2
b3	b4

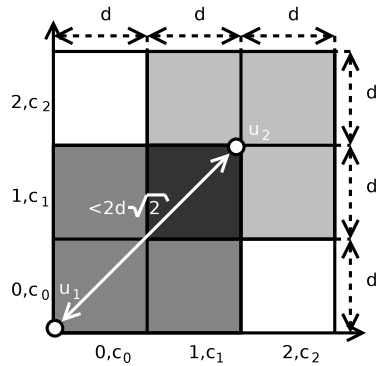
In Separation

Limitation and extensions of the idea

Limitations of the core idea:

1. Intercepted Ψ opens location privacy leakage possibility
2. A proximity detection distance is fixed $(2d\sqrt{2})$





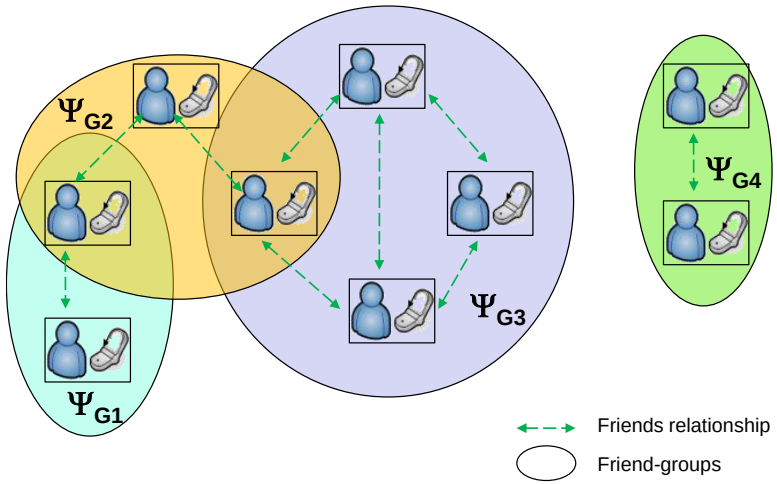
Extensions, supported by FriendLocator:

1. Grouping of friends
2. Incremental Proximity Detection Approach

Grouping of friends

Solution:

- Users are grouped into friend-groups
- A distinct Ψ is assigned for every friend-group



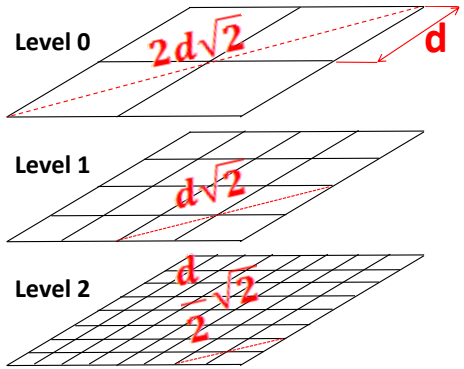
Consequences:

- When user becomes malicious, users from common groups are endangered

Incremental Proximity Detection Approach

The solution:

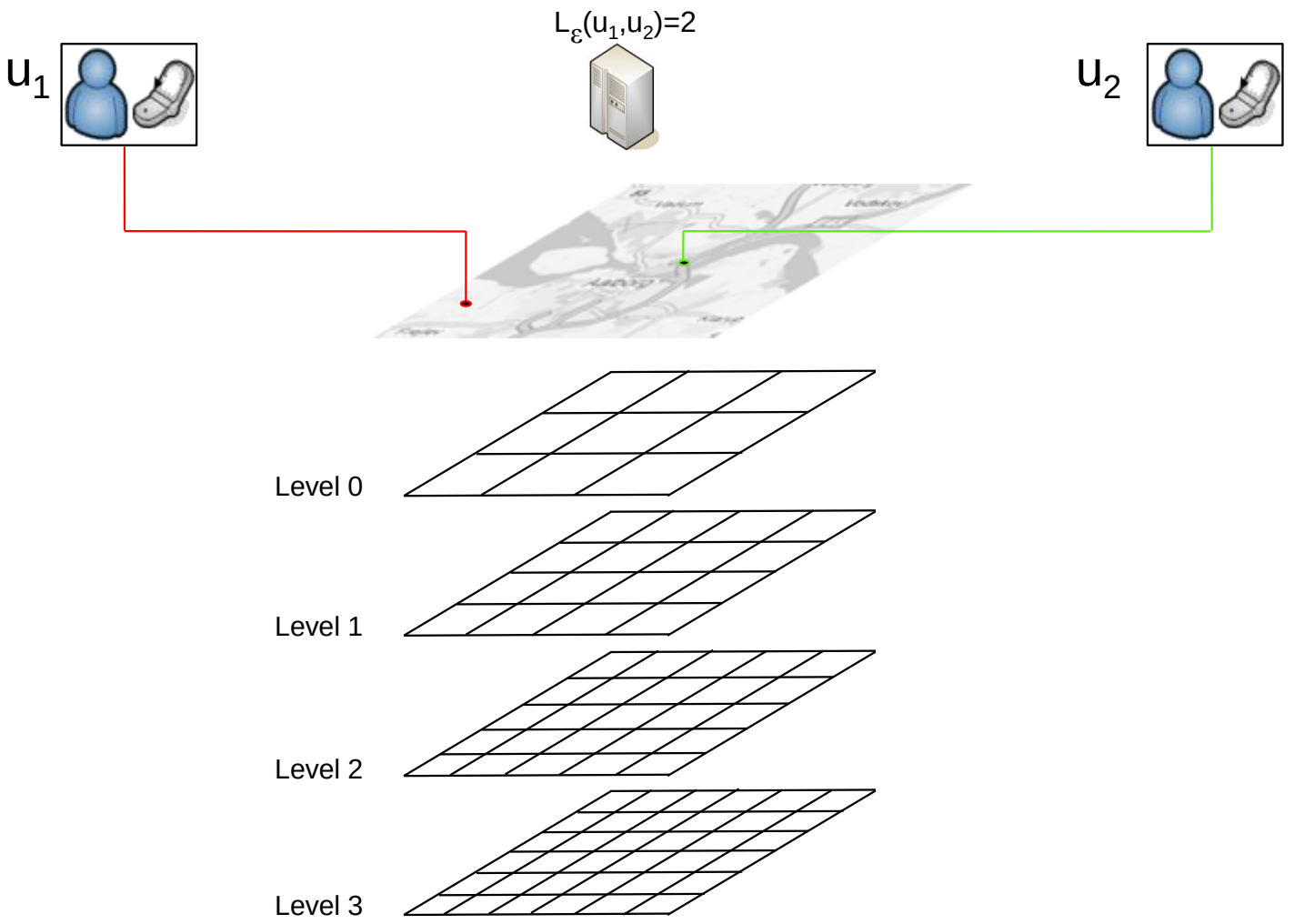
- A list of grids with decreasing cell size is assigned for every friend-group



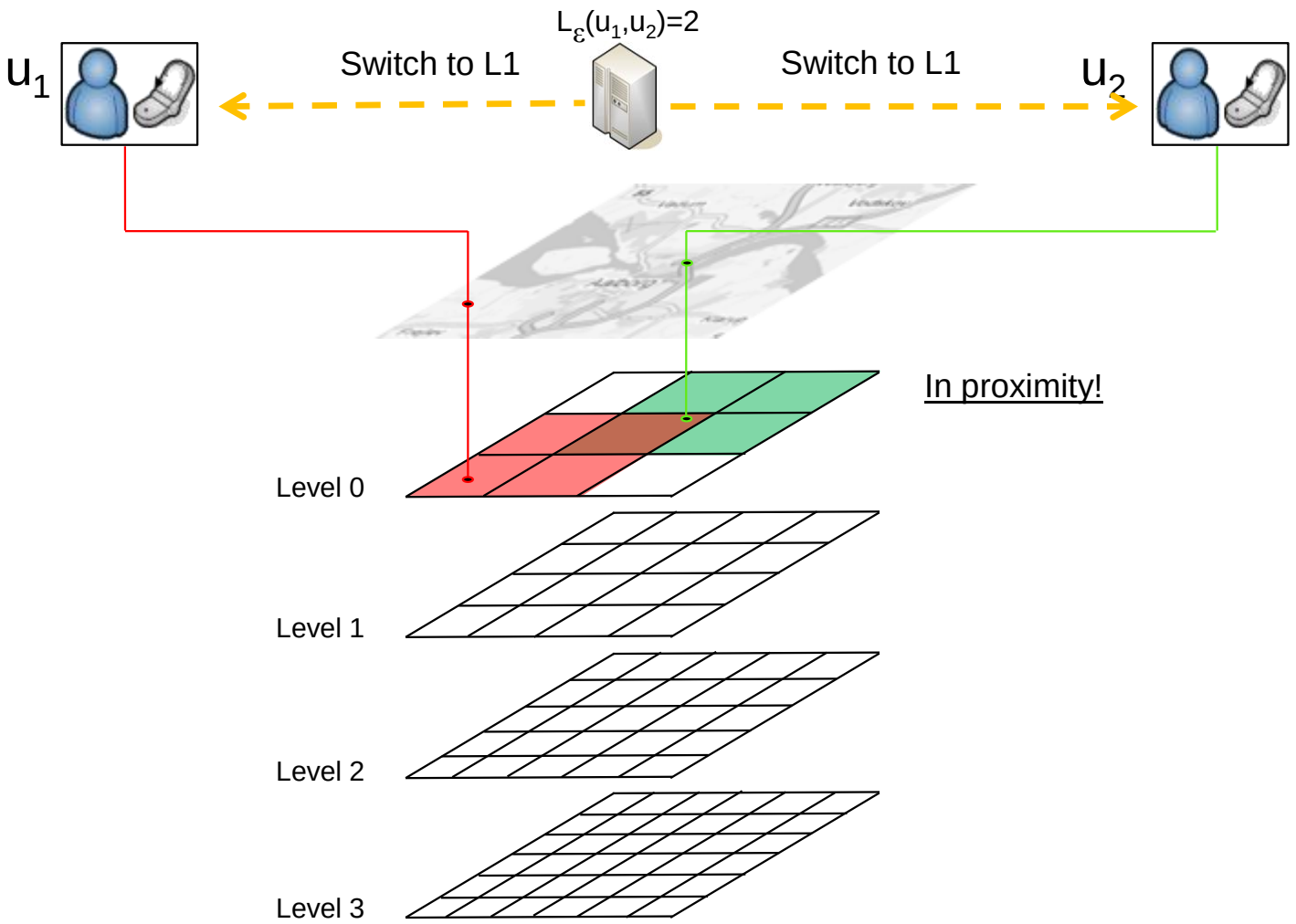
Consequences:

- Any pair of friends in friend-group will be able to choose a preferred proximity distance

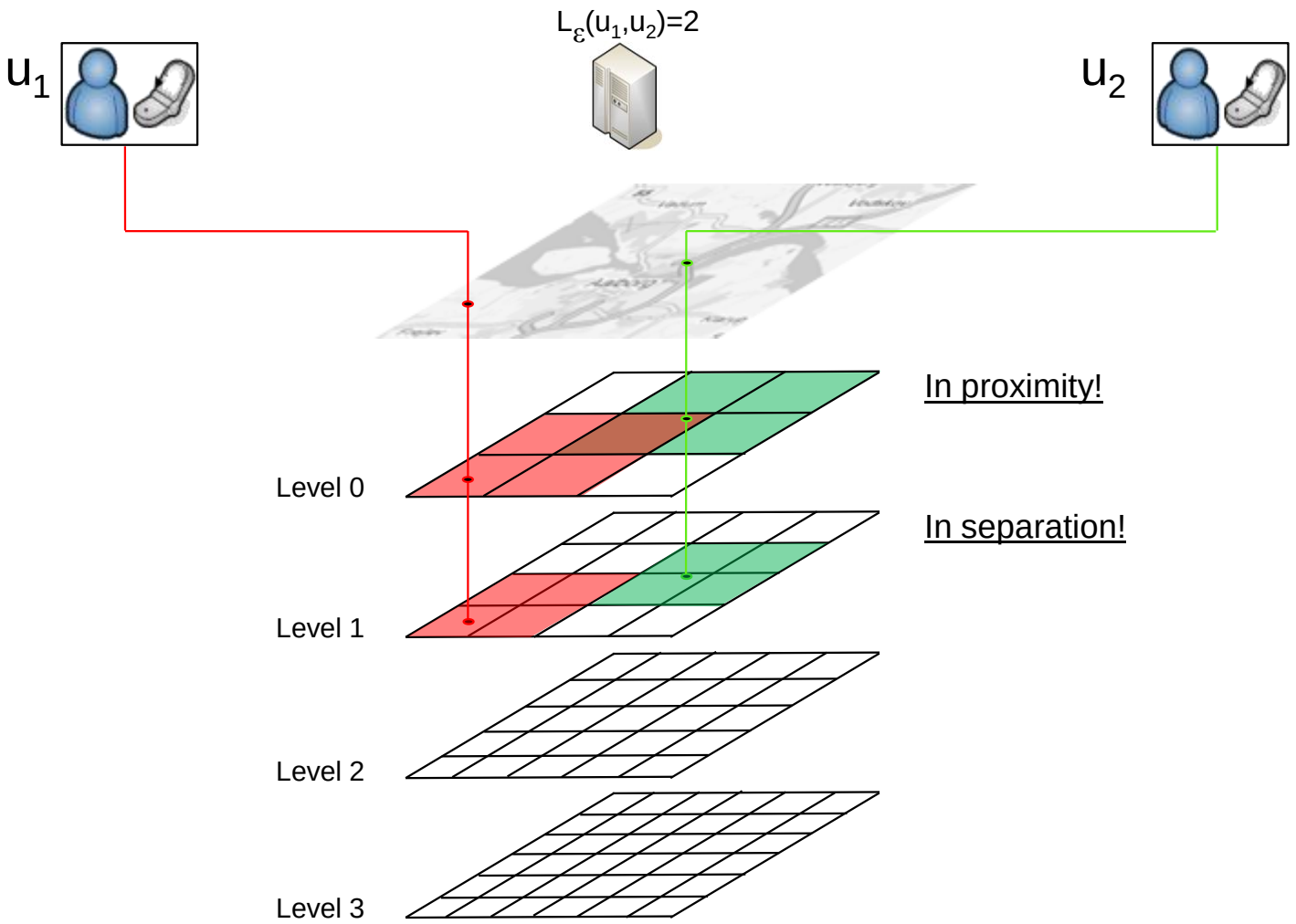
Incremental Proximity Detection Approach



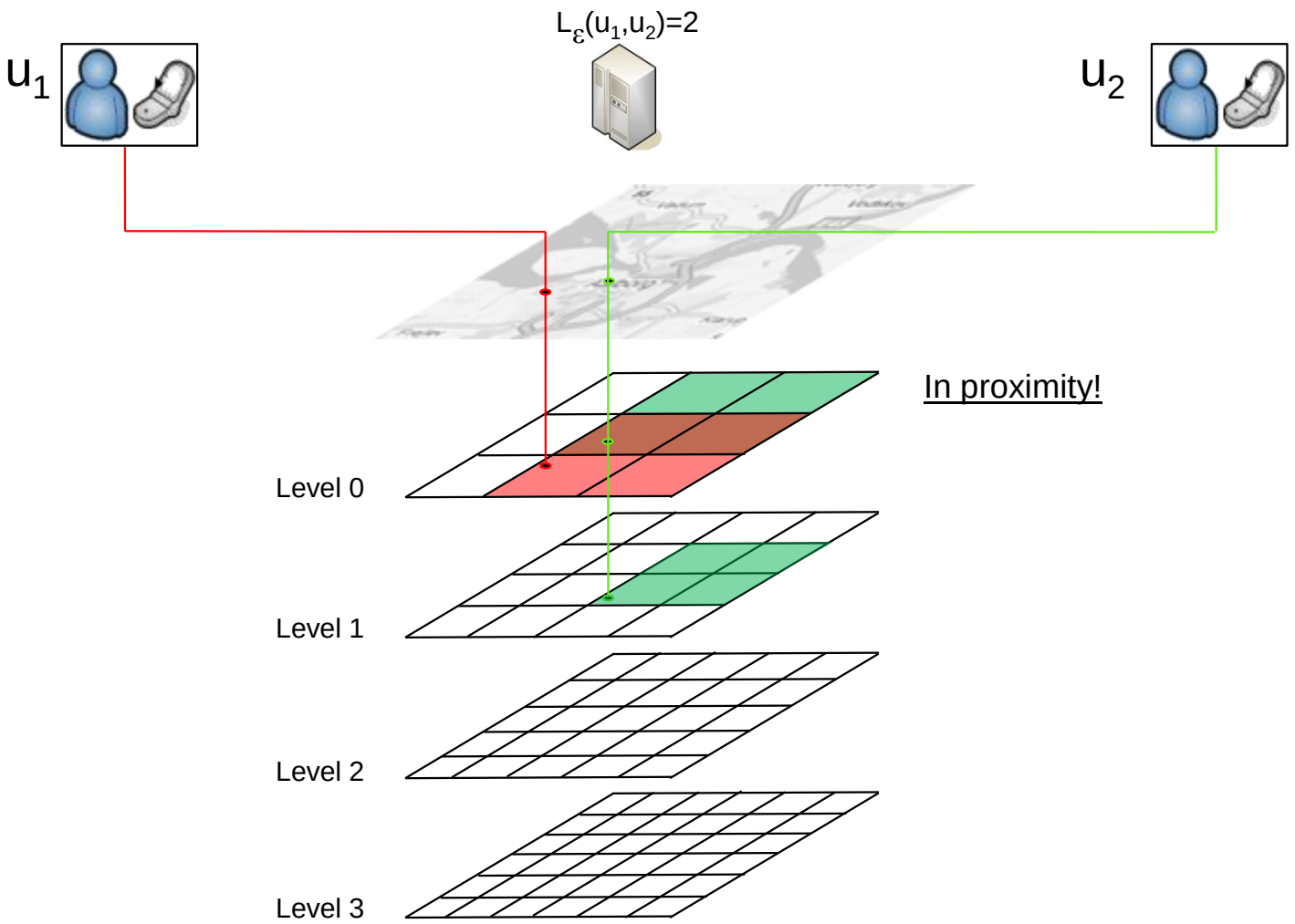
Incremental Proximity Detection Approach



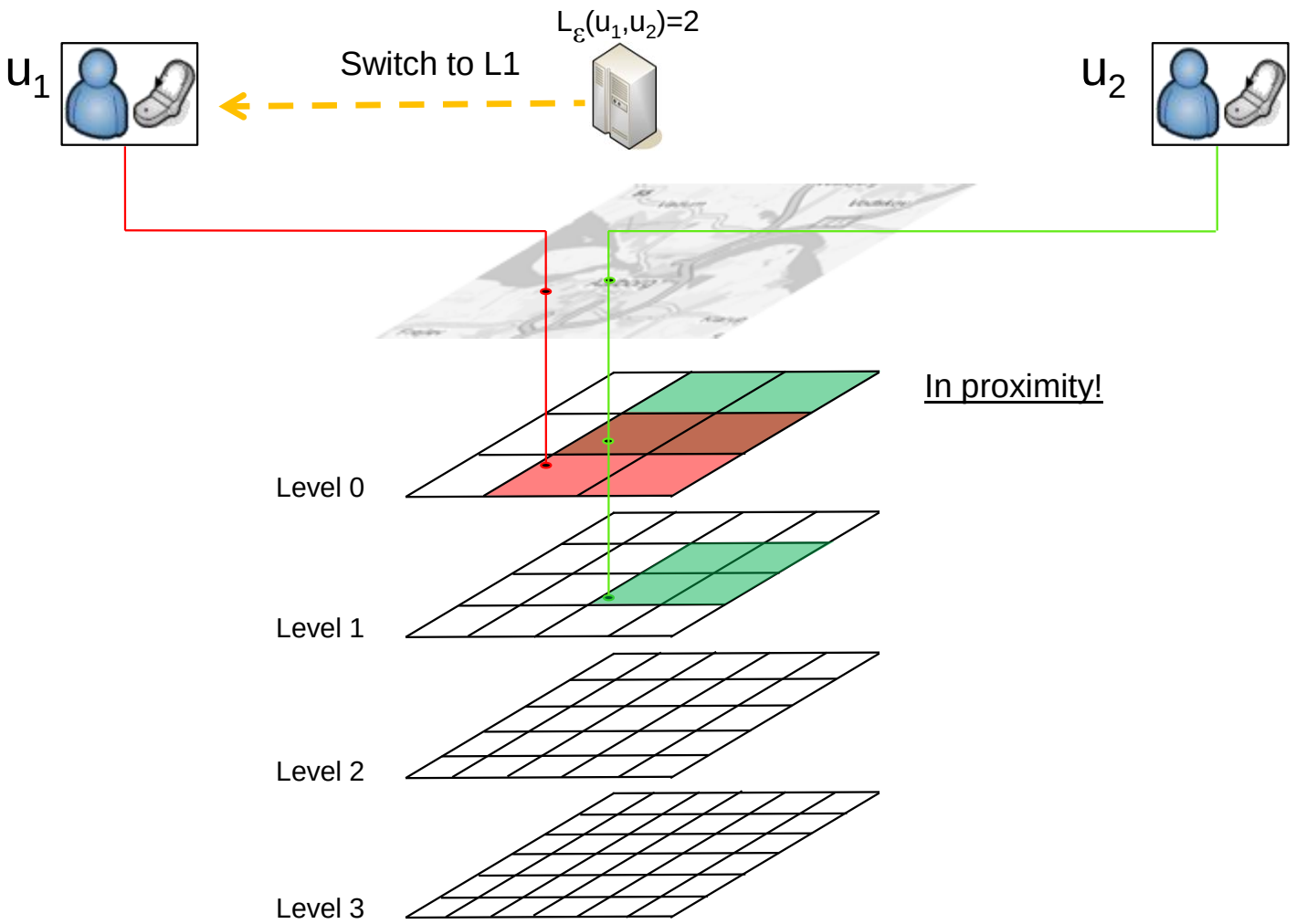
Incremental Proximity Detection Approach



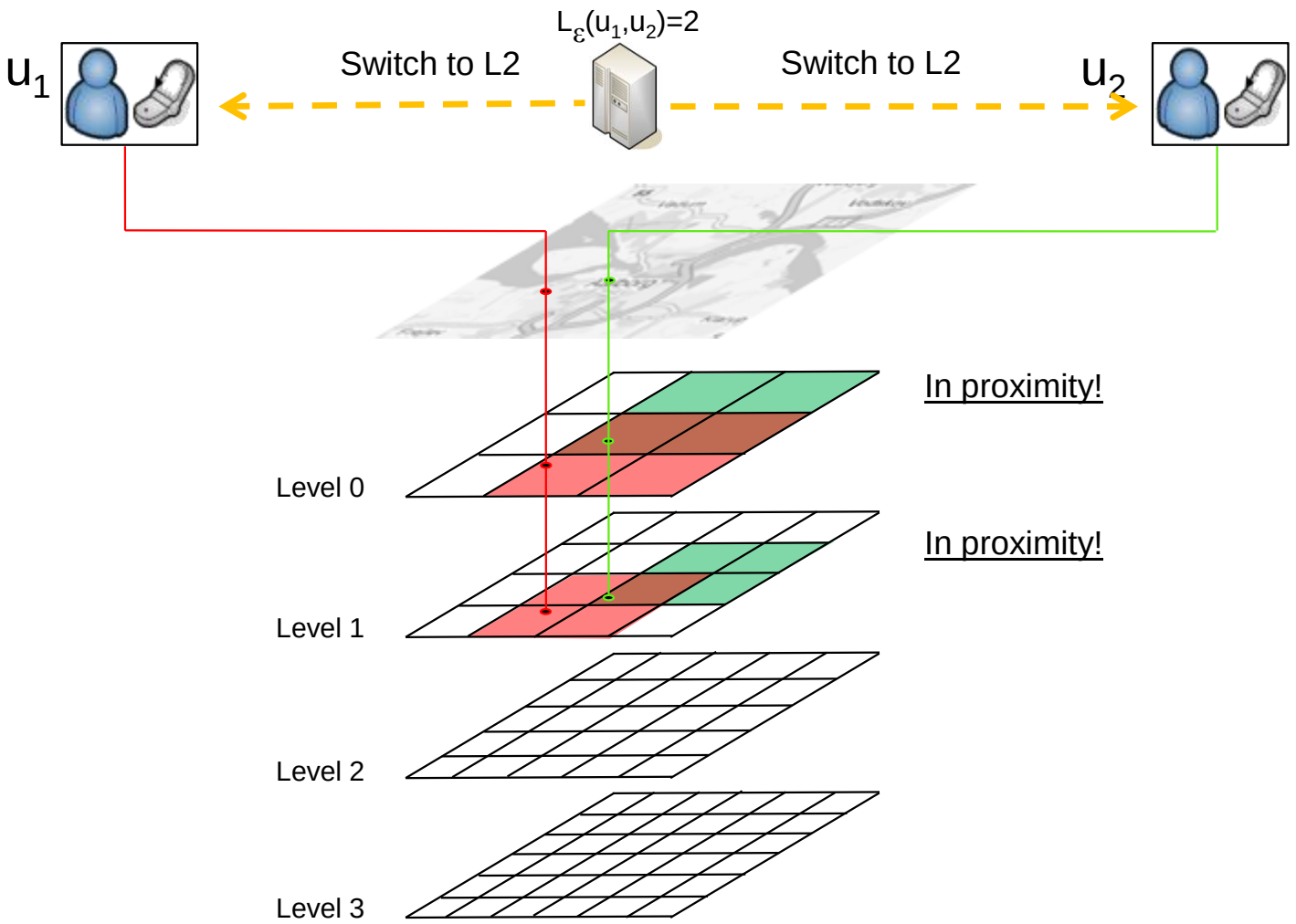
Incremental Proximity Detection Approach



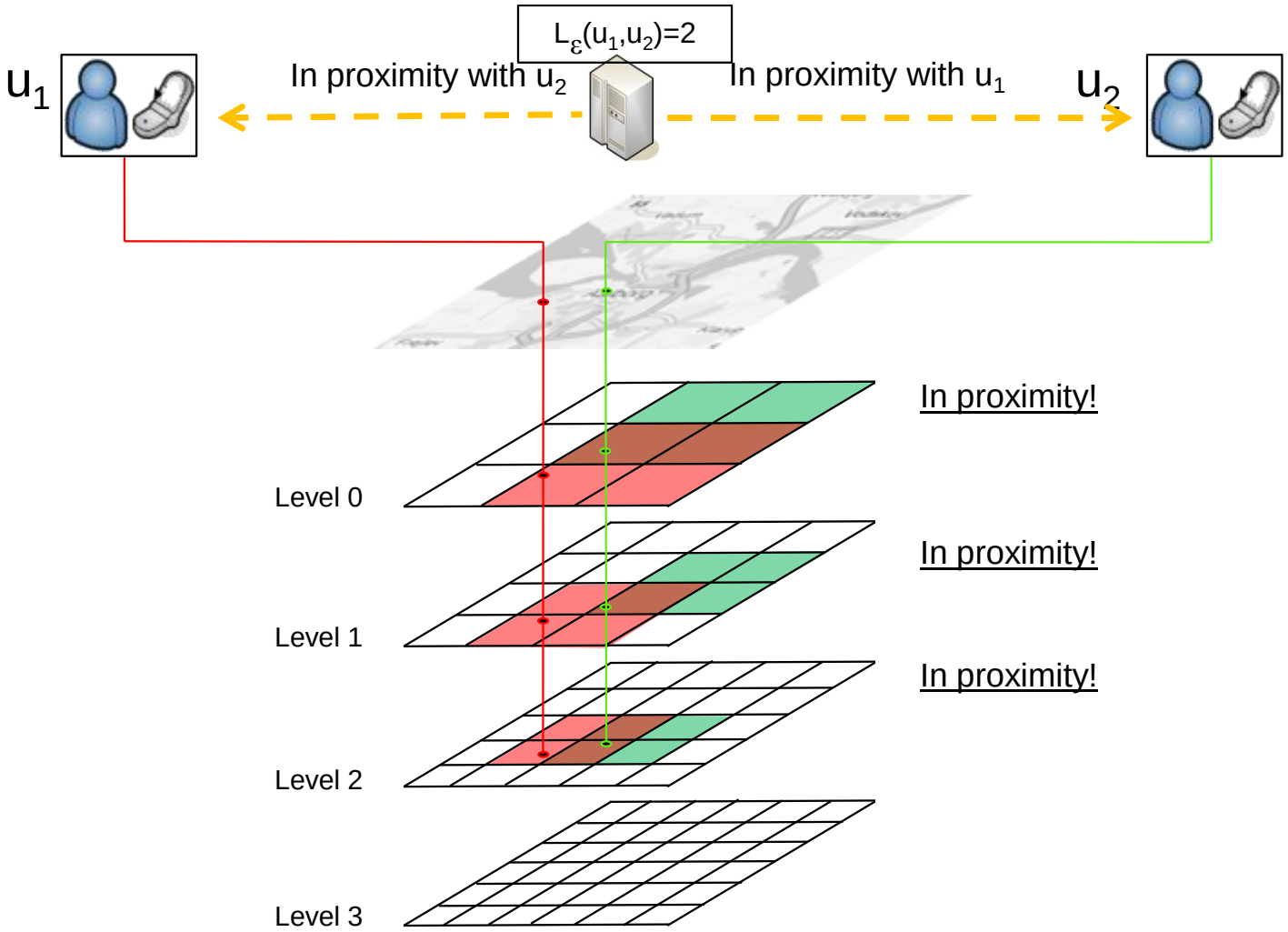
Incremental Proximity Detection Approach



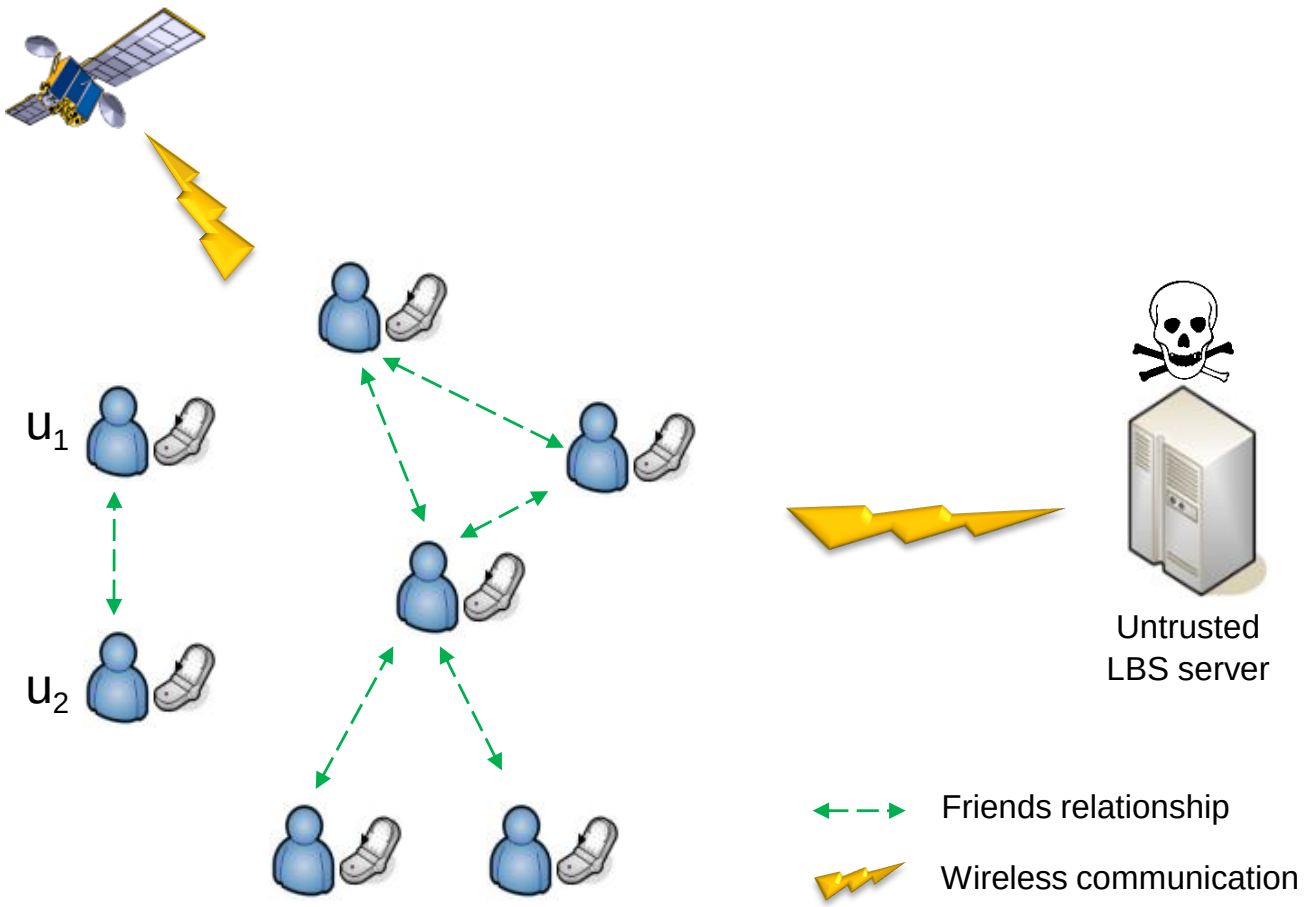
Incremental Proximity Detection Approach



Incremental Proximity Detection Approach



Problem setting



VicinityLocator

VicinityLocator

-

Flexible Proximity Detection In Mobile Social Networks

Motivation

FriendLocator

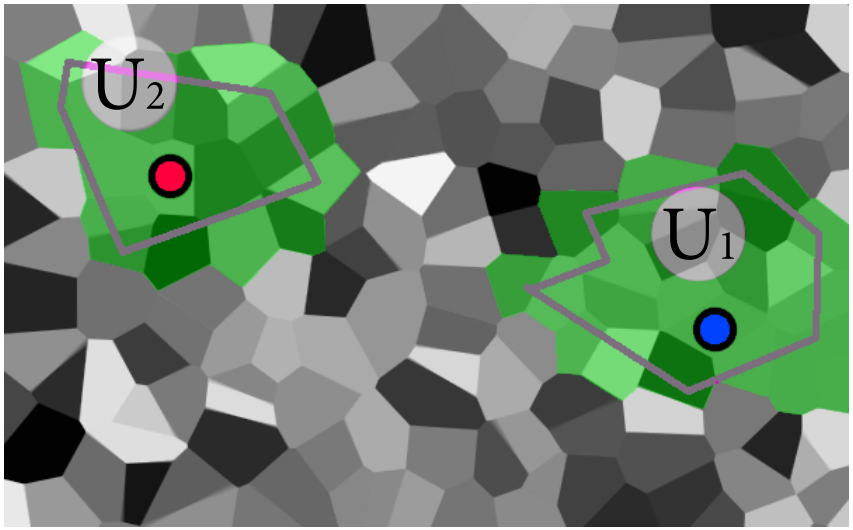
- the Proximity Detection Guaranties is to weak $(2d\sqrt{2})$
- is too inflexible - privacy & precision not independent.

Want solution independent of how the space it partitioned

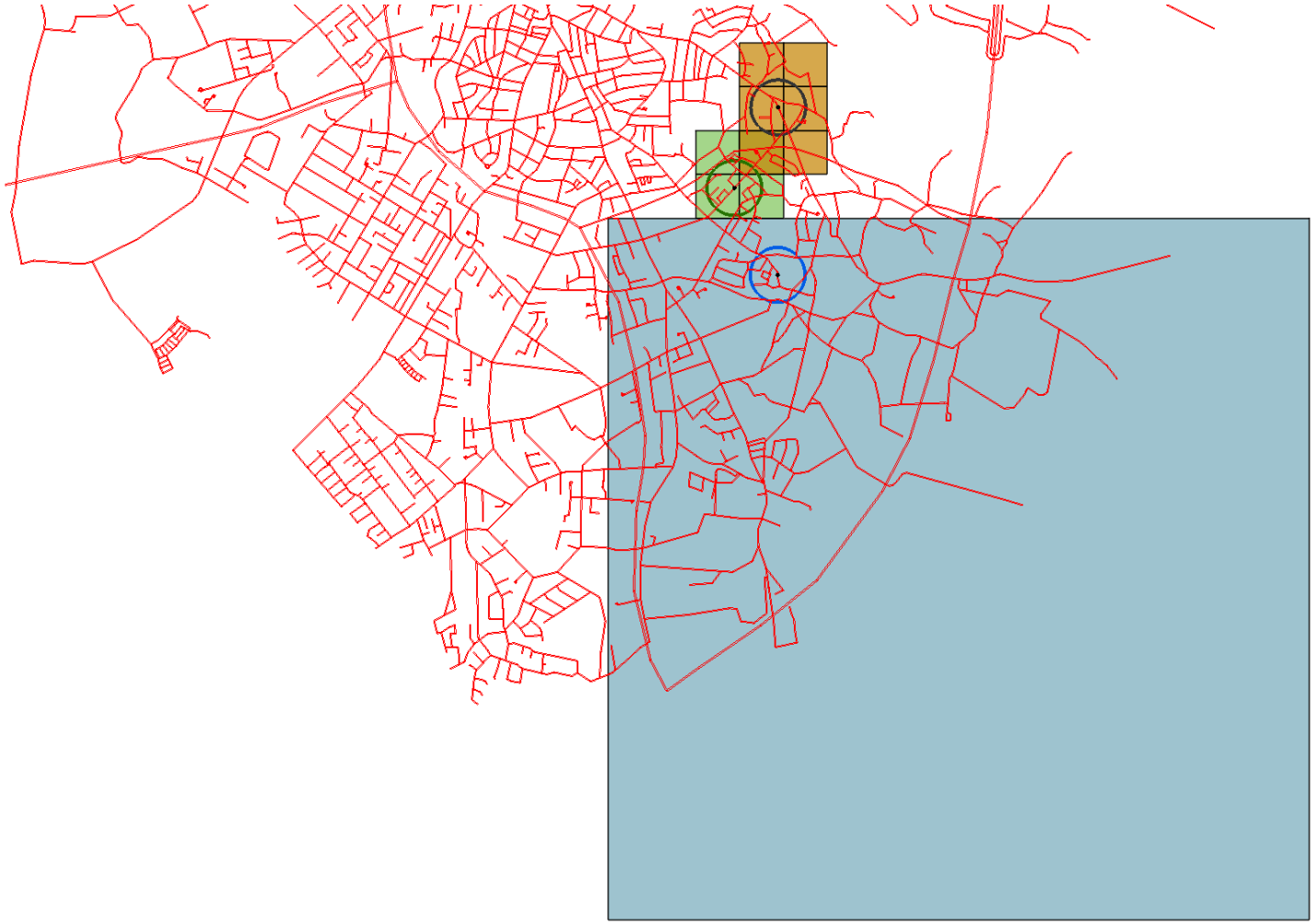
Improvements

New features and improvements in VicinityLocator

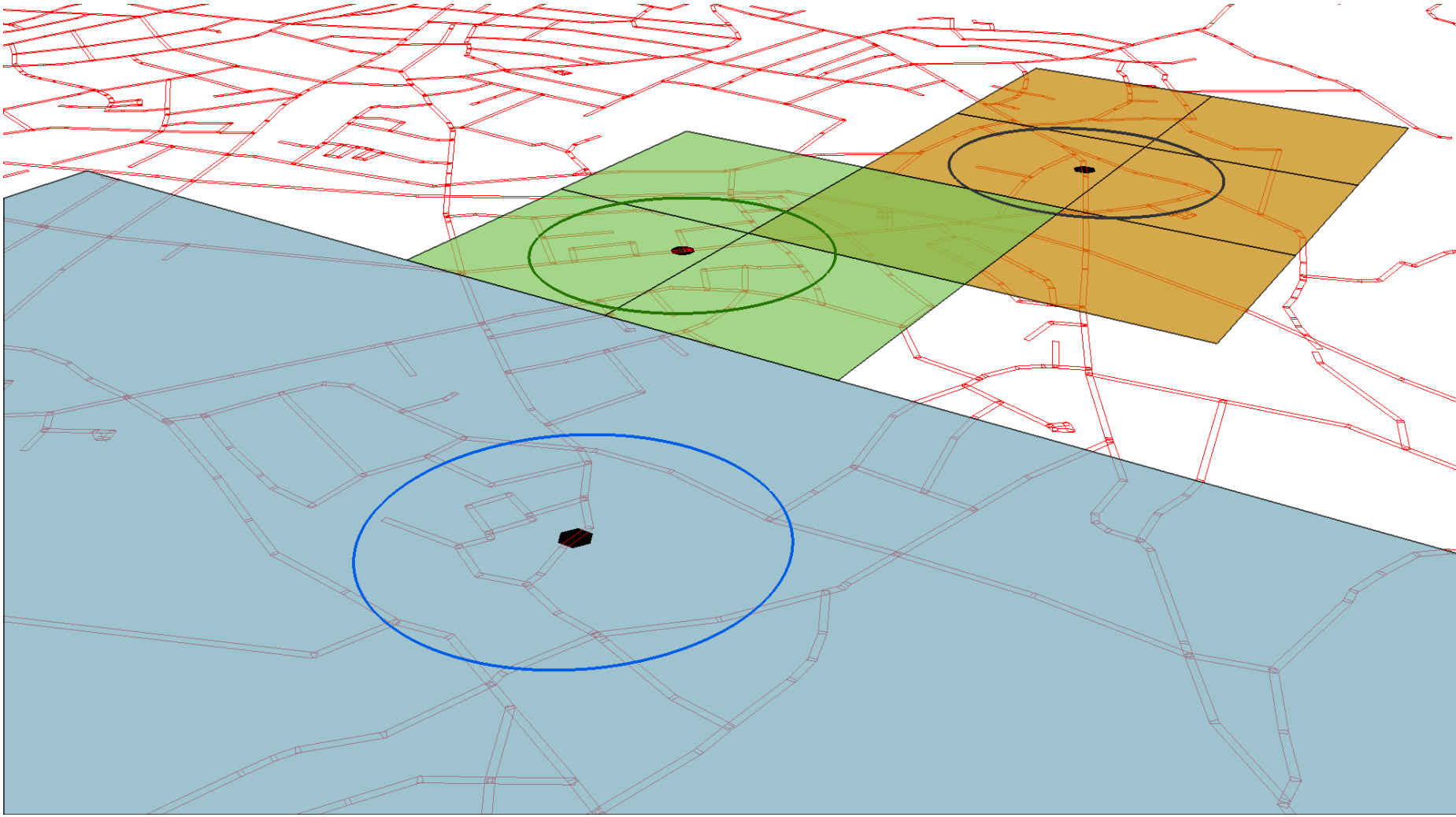
- Granules (and rasterasation)
- Better guaranties
- Ajustable vicinity



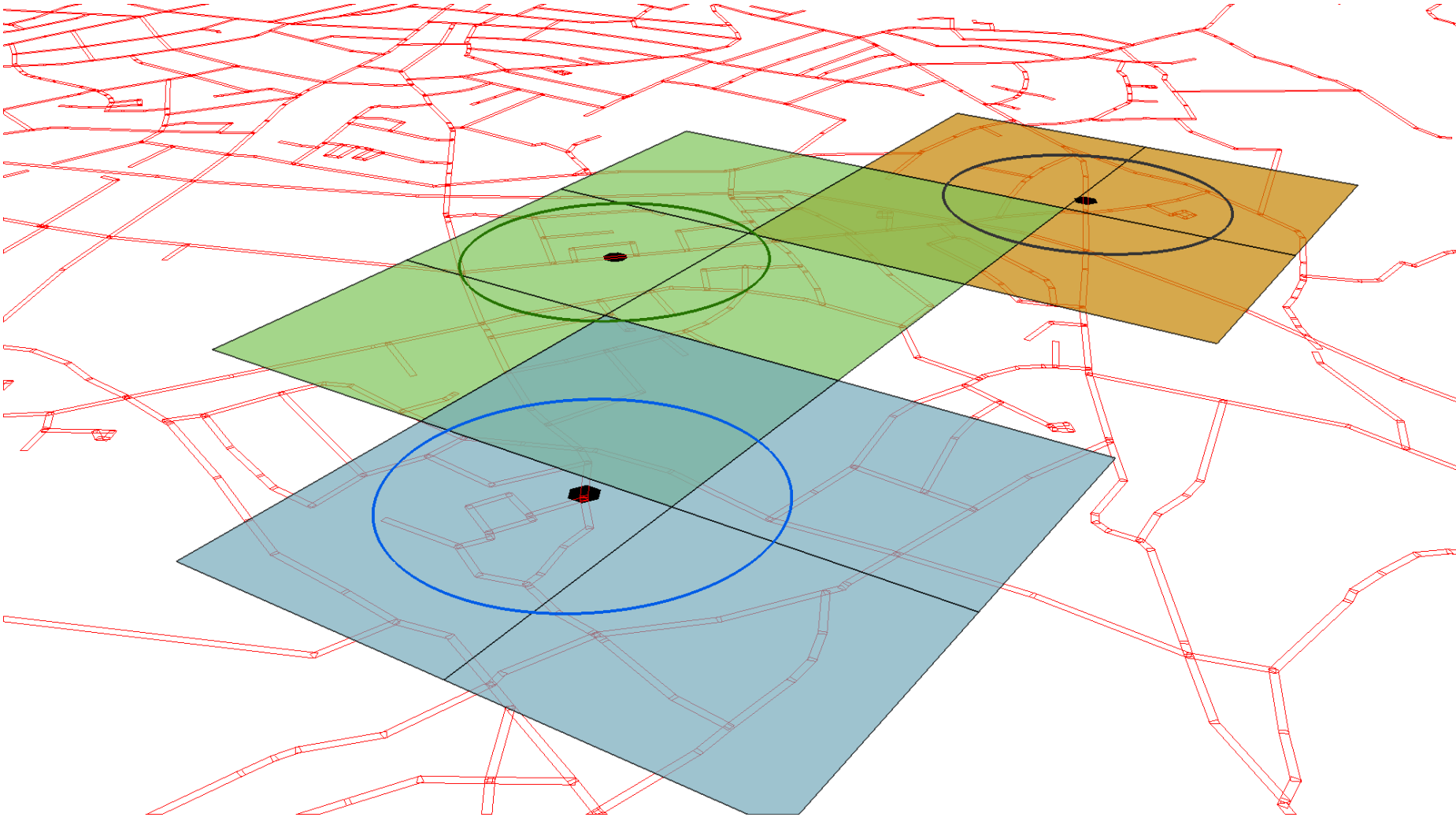
Demo of VicinityLocator



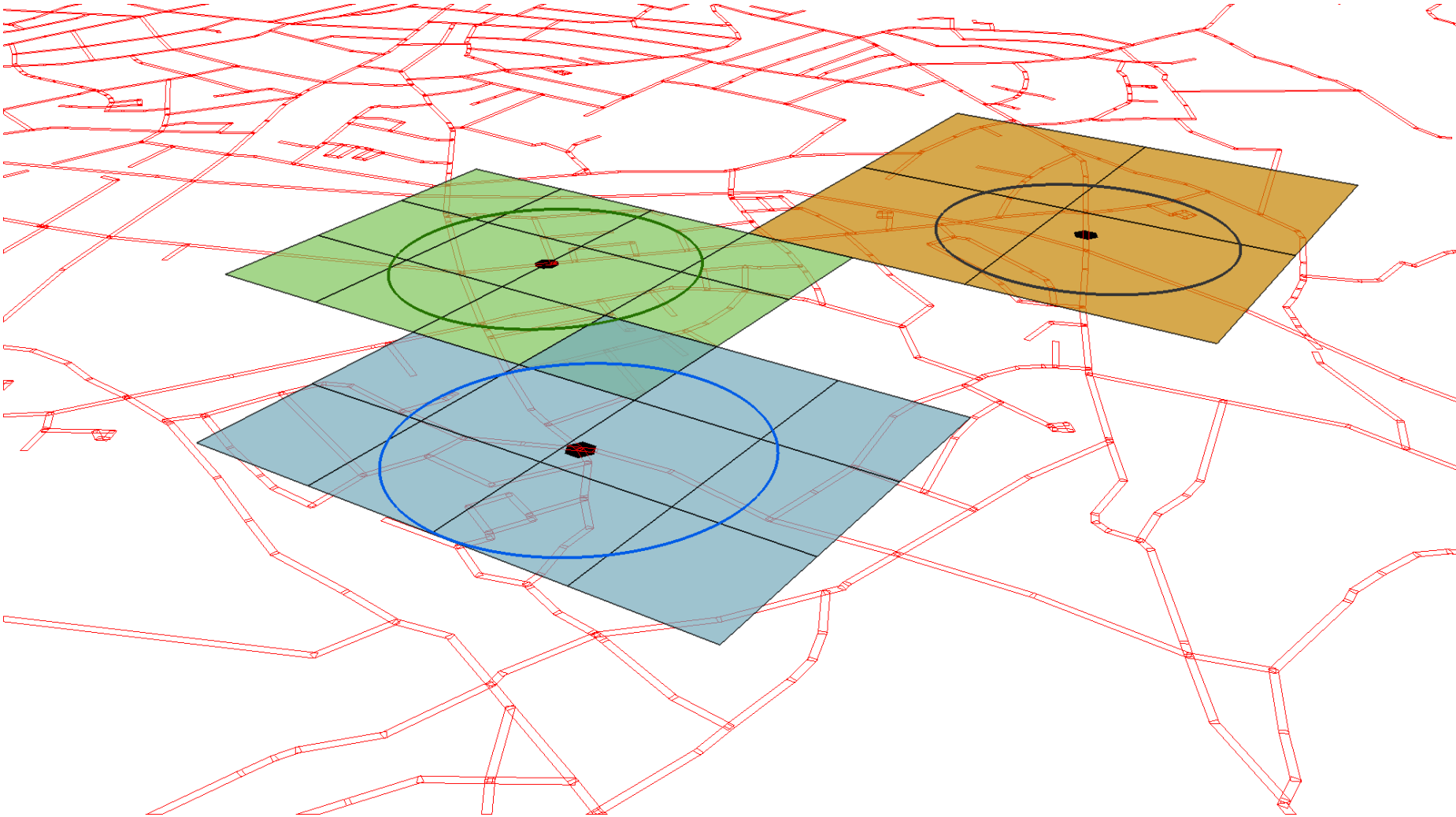
Demo of VicinityLocator



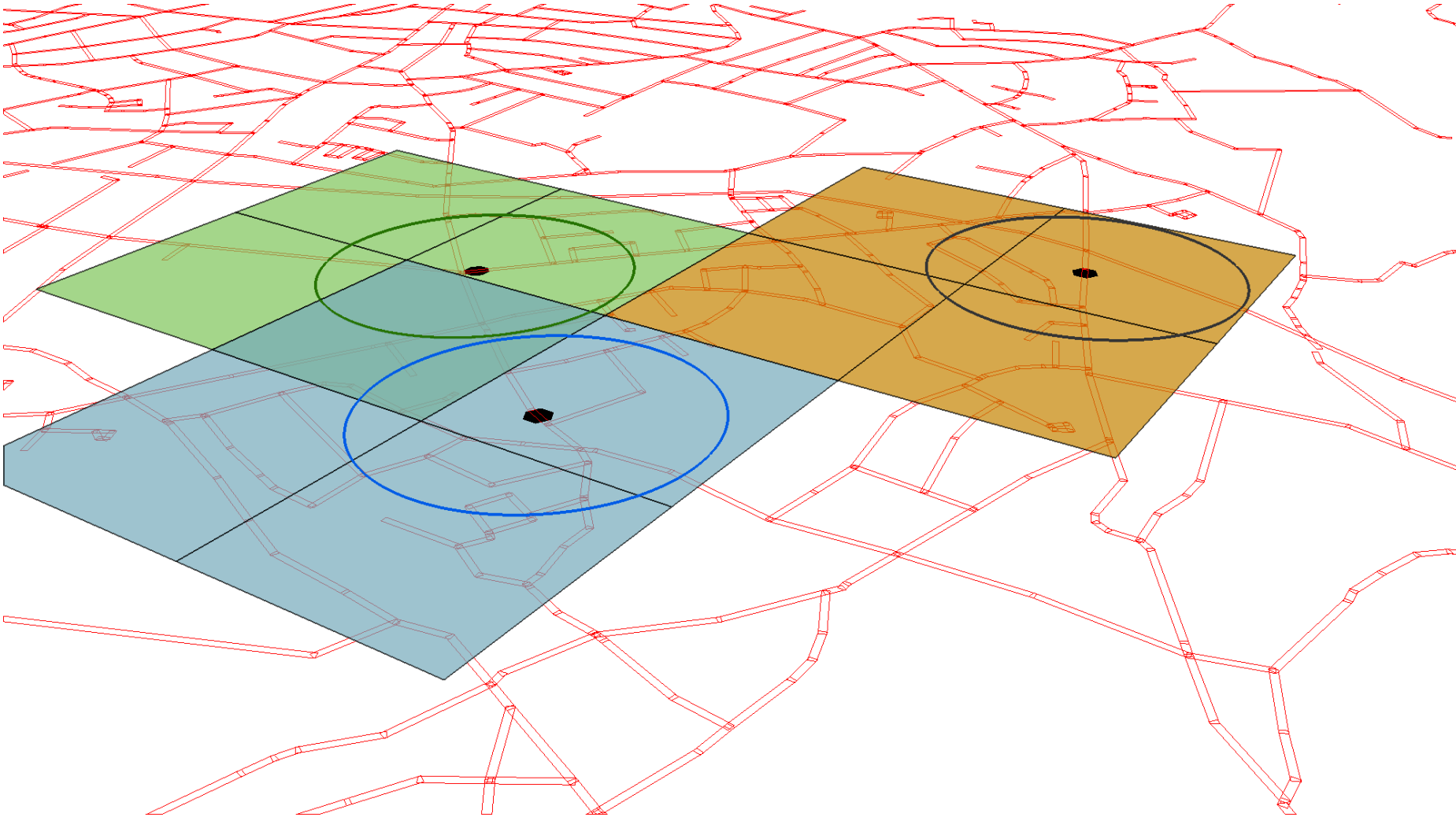
Demo of VicinityLocator



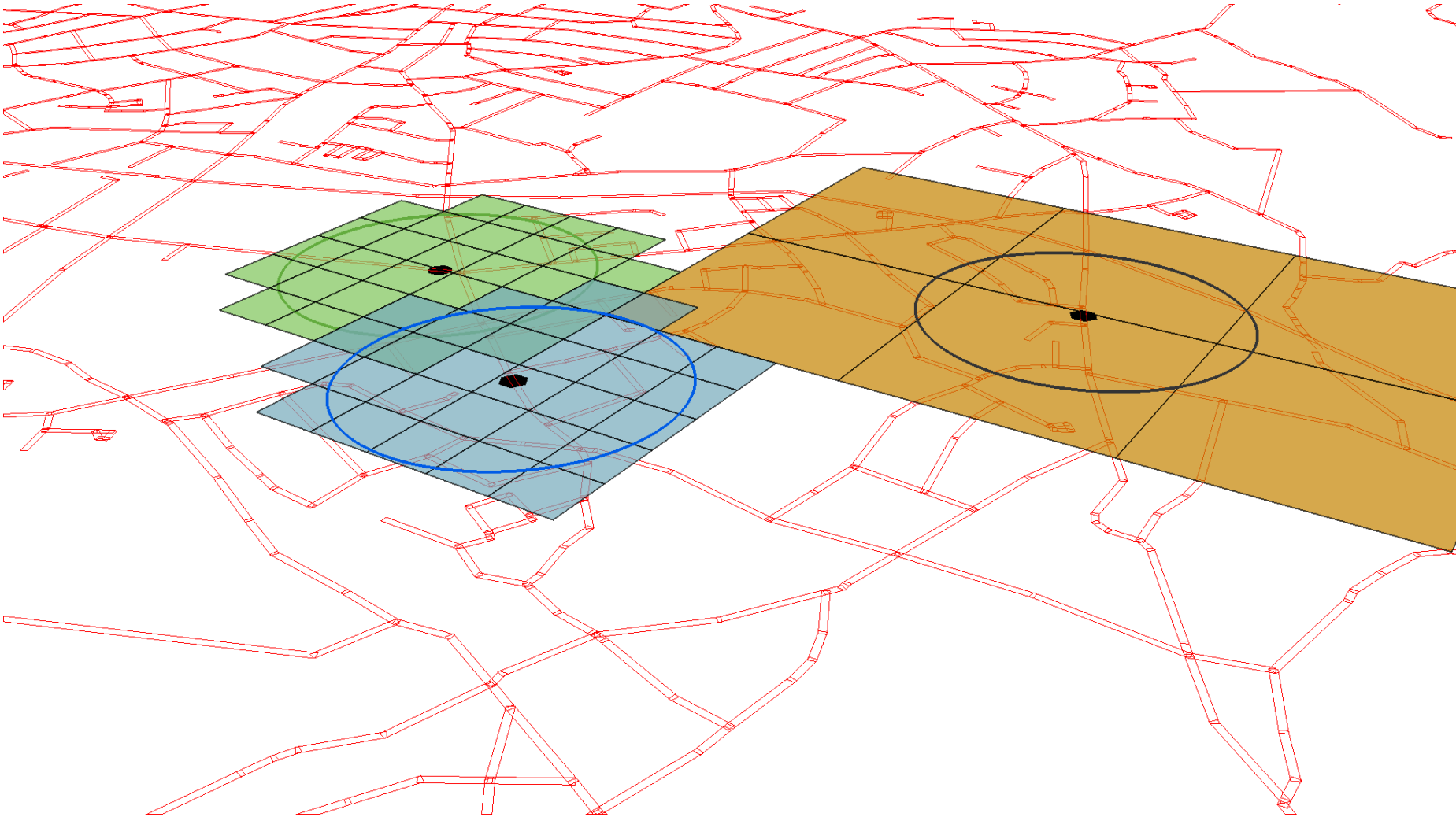
Demo of VicinityLocator



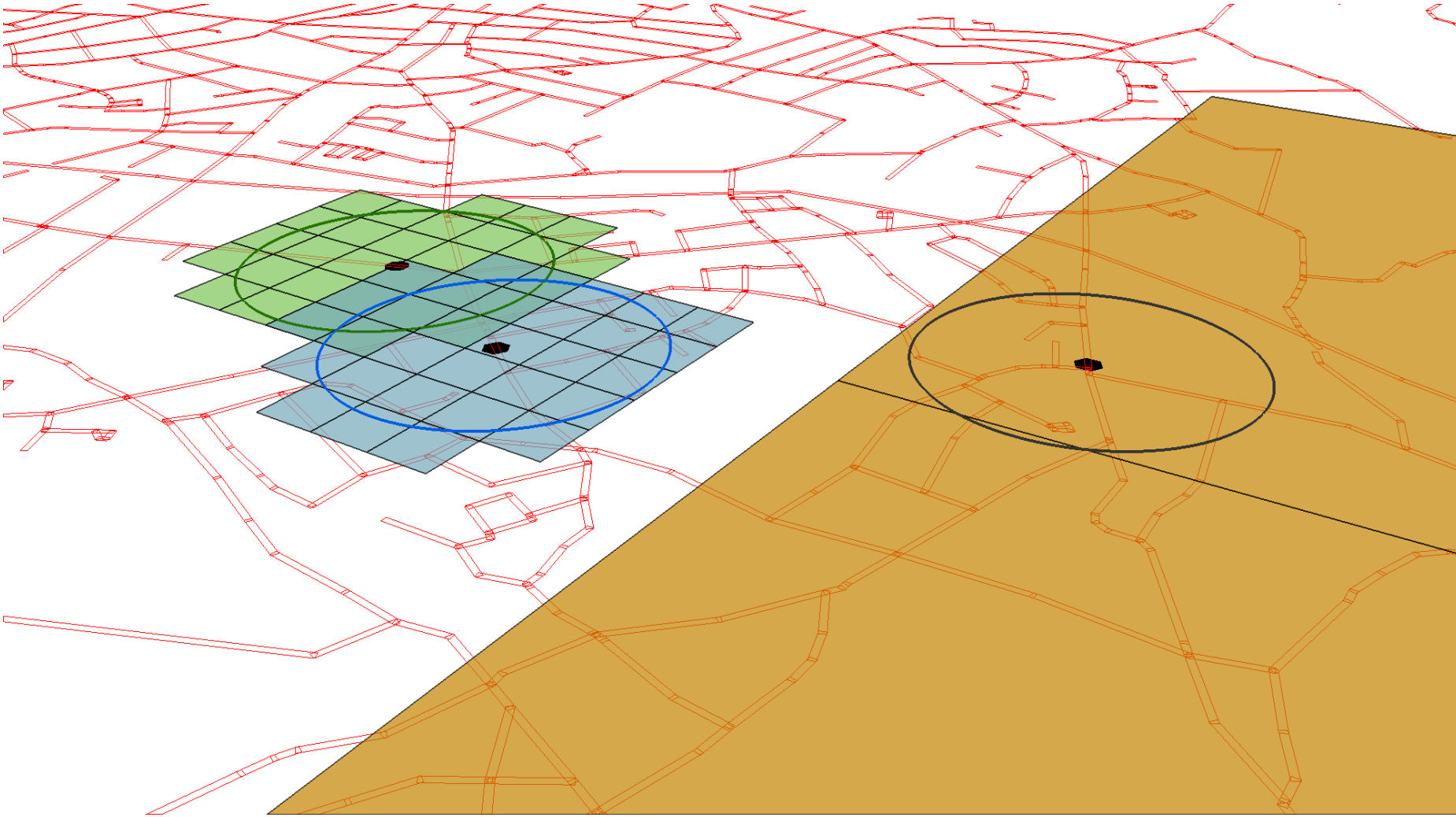
Demo of VicinityLocator



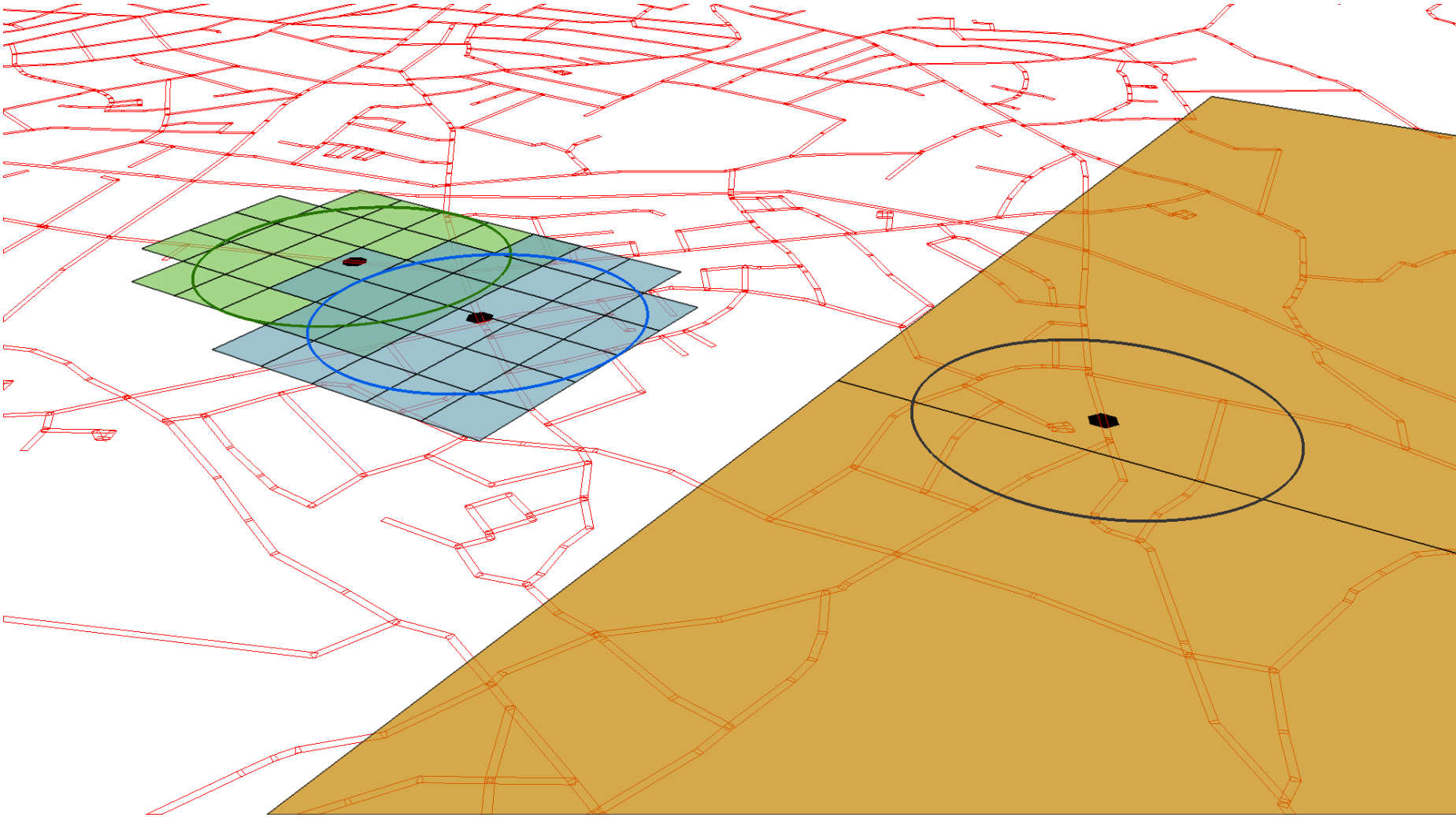
Demo of VicinityLocator



Demo of VicinityLocator

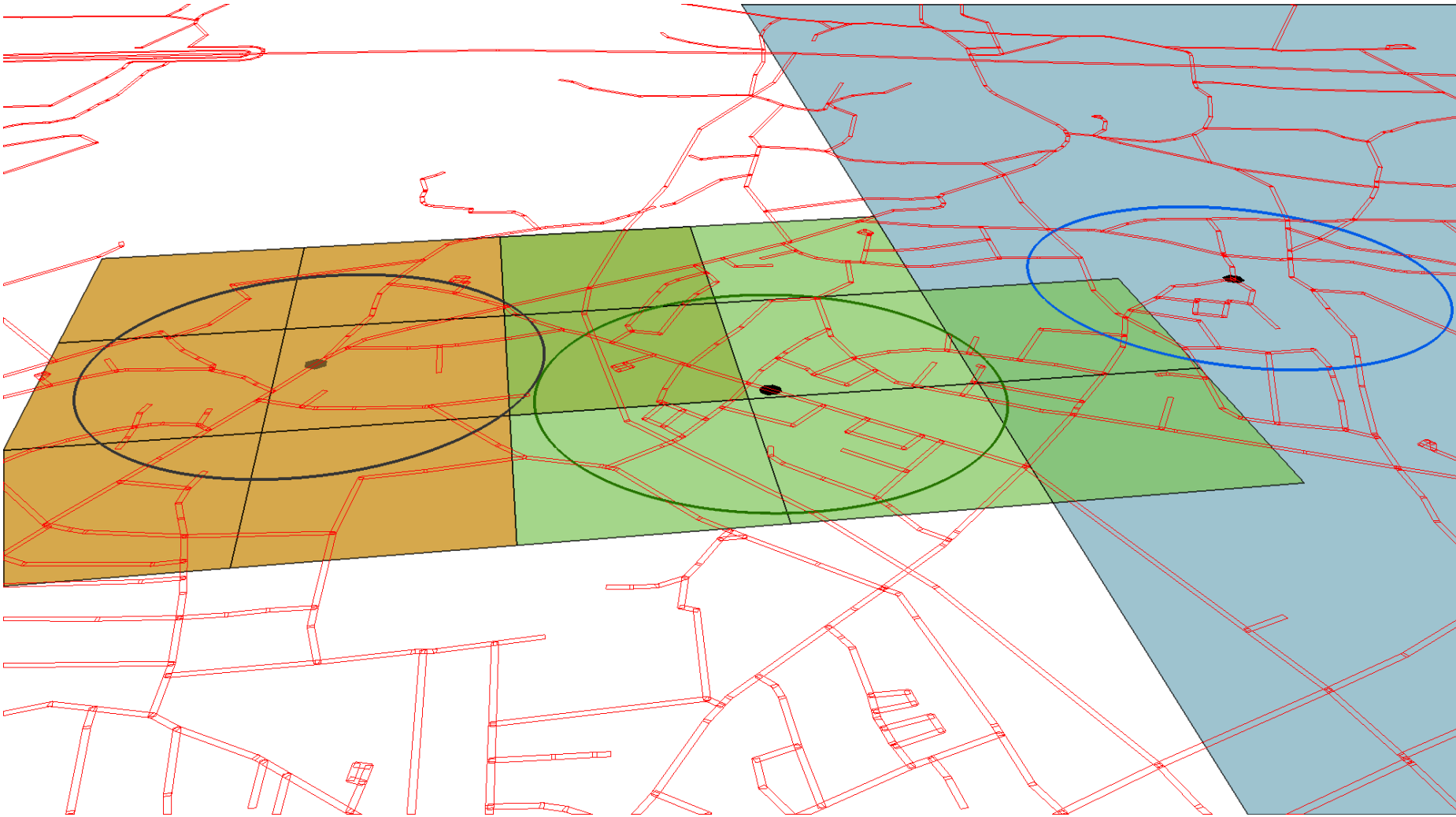


Demo of VicinityLocator

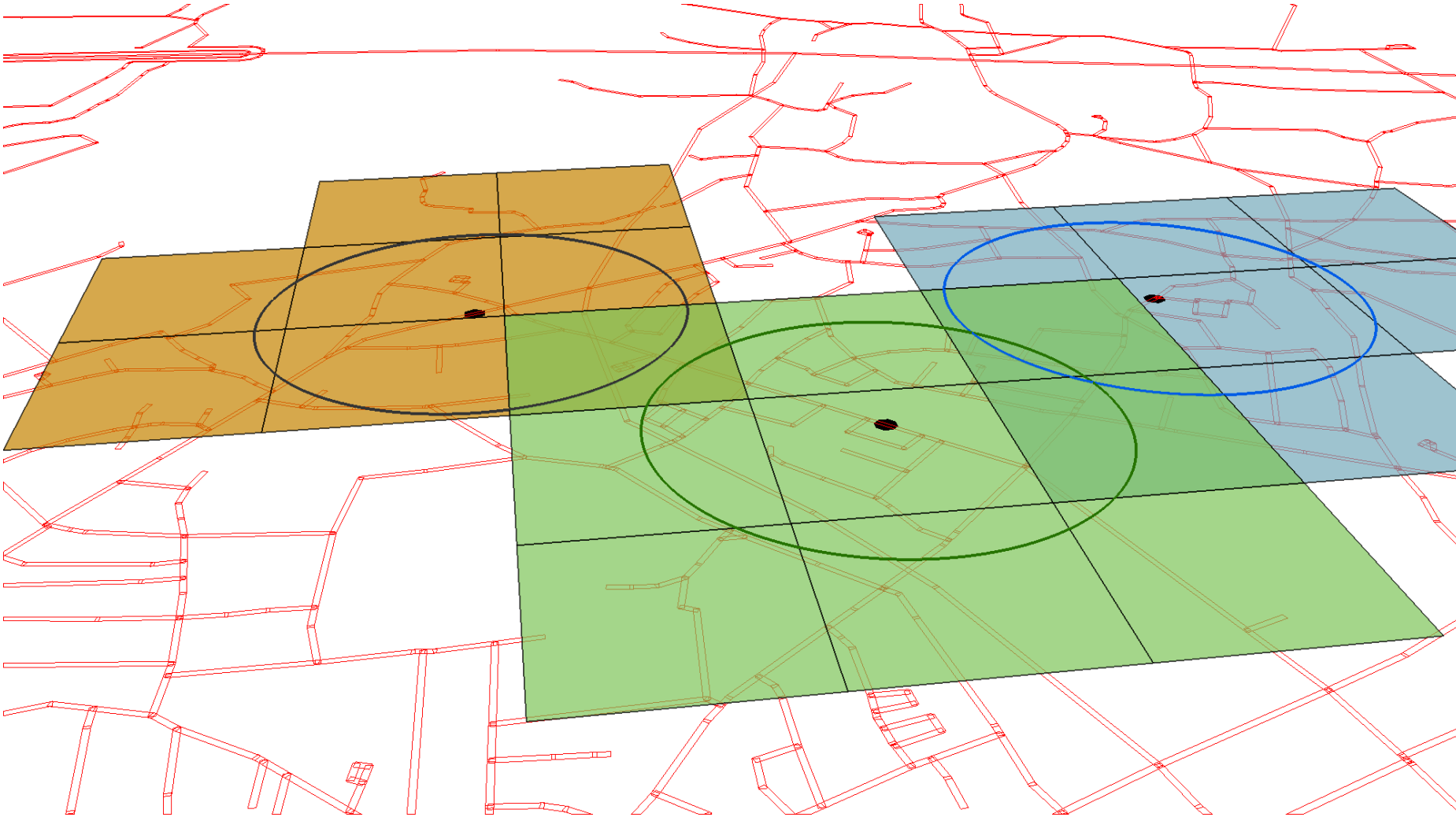


VicinityLocator with Roadnetwork Filter

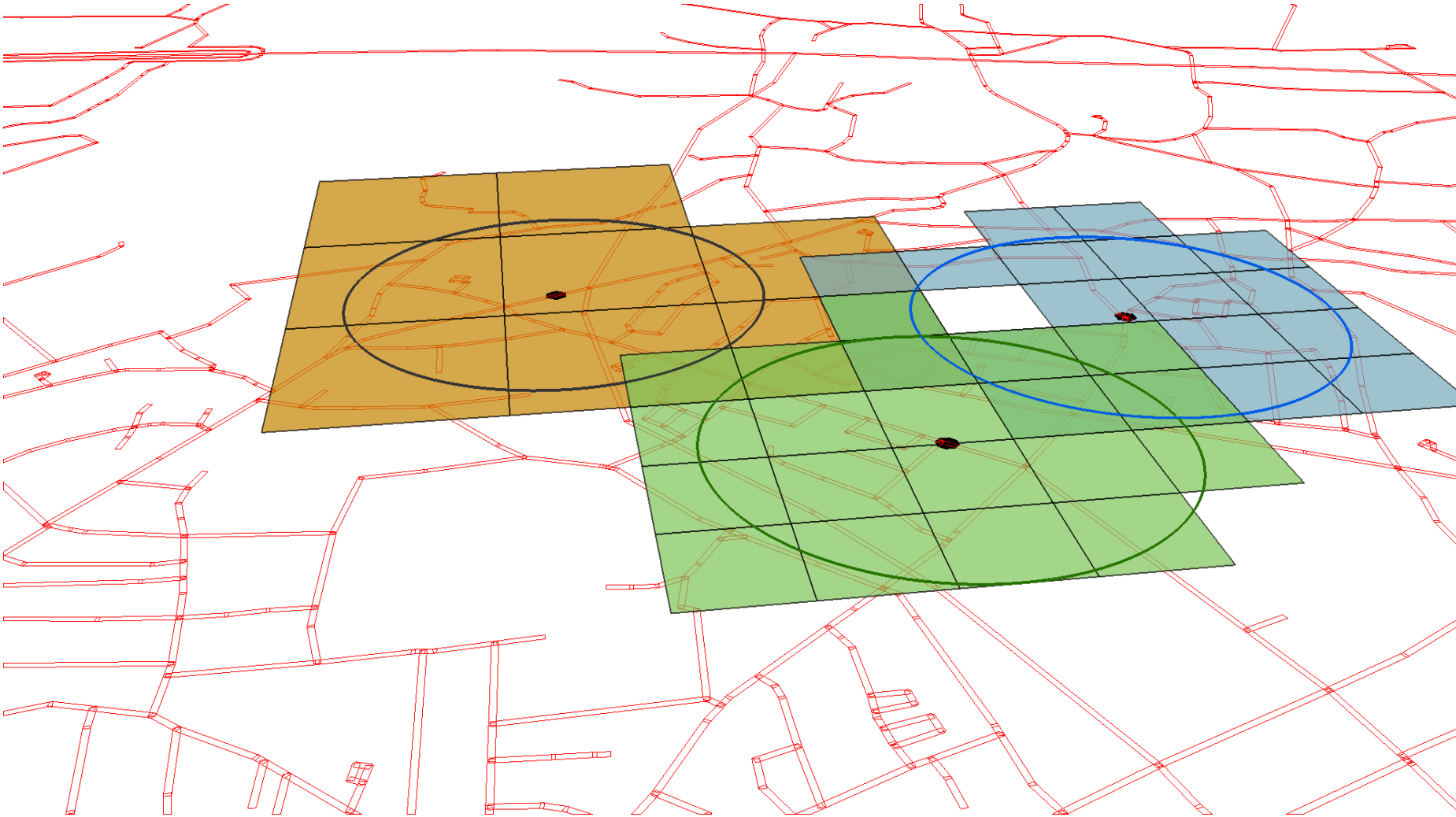
Demo of VicinityLocator with Roadnetwork Filter



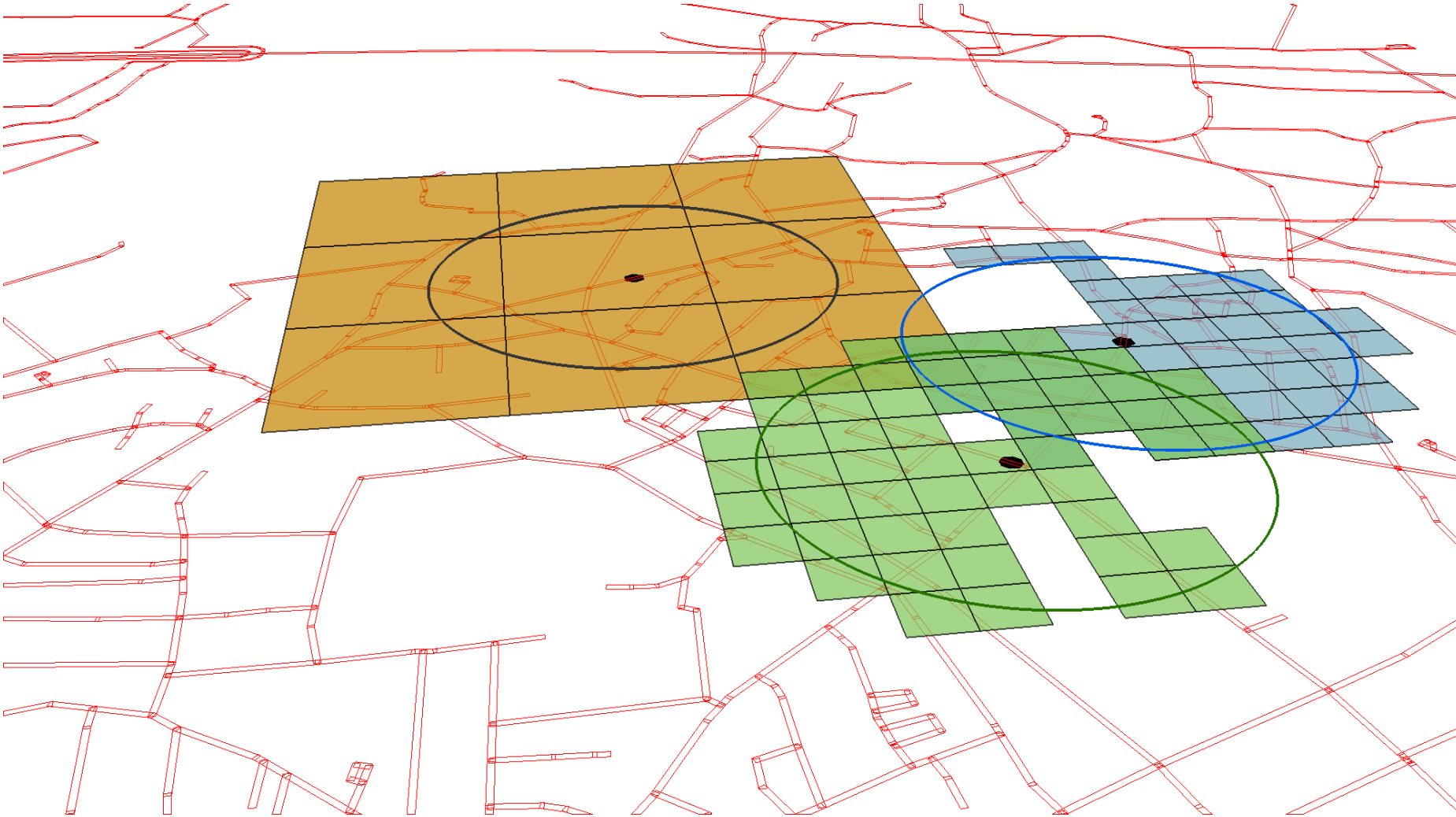
Demo of VicinityLocator with Roadnetwork Filter



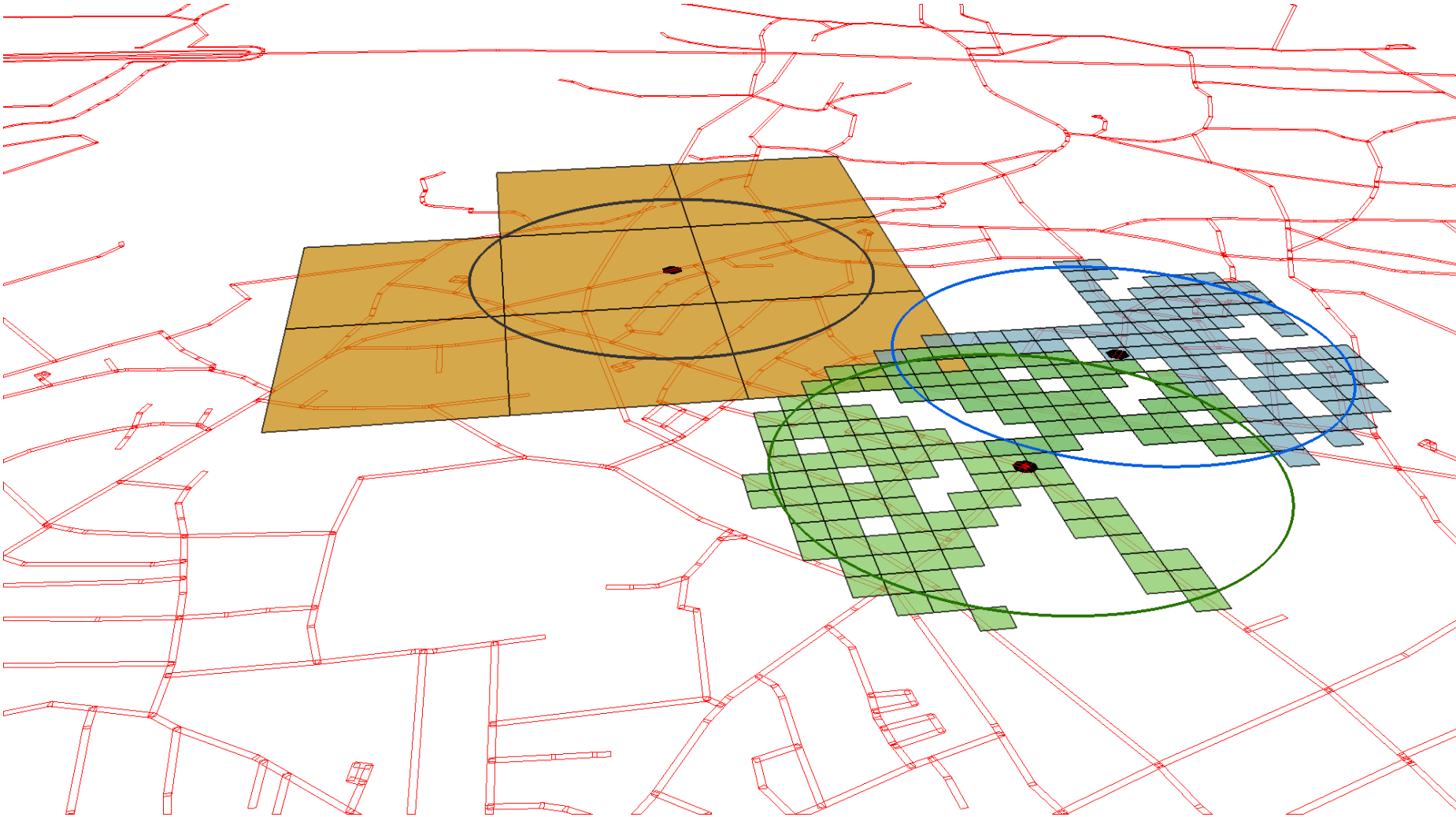
Demo of VicinityLocator with Roadnetwork Filter



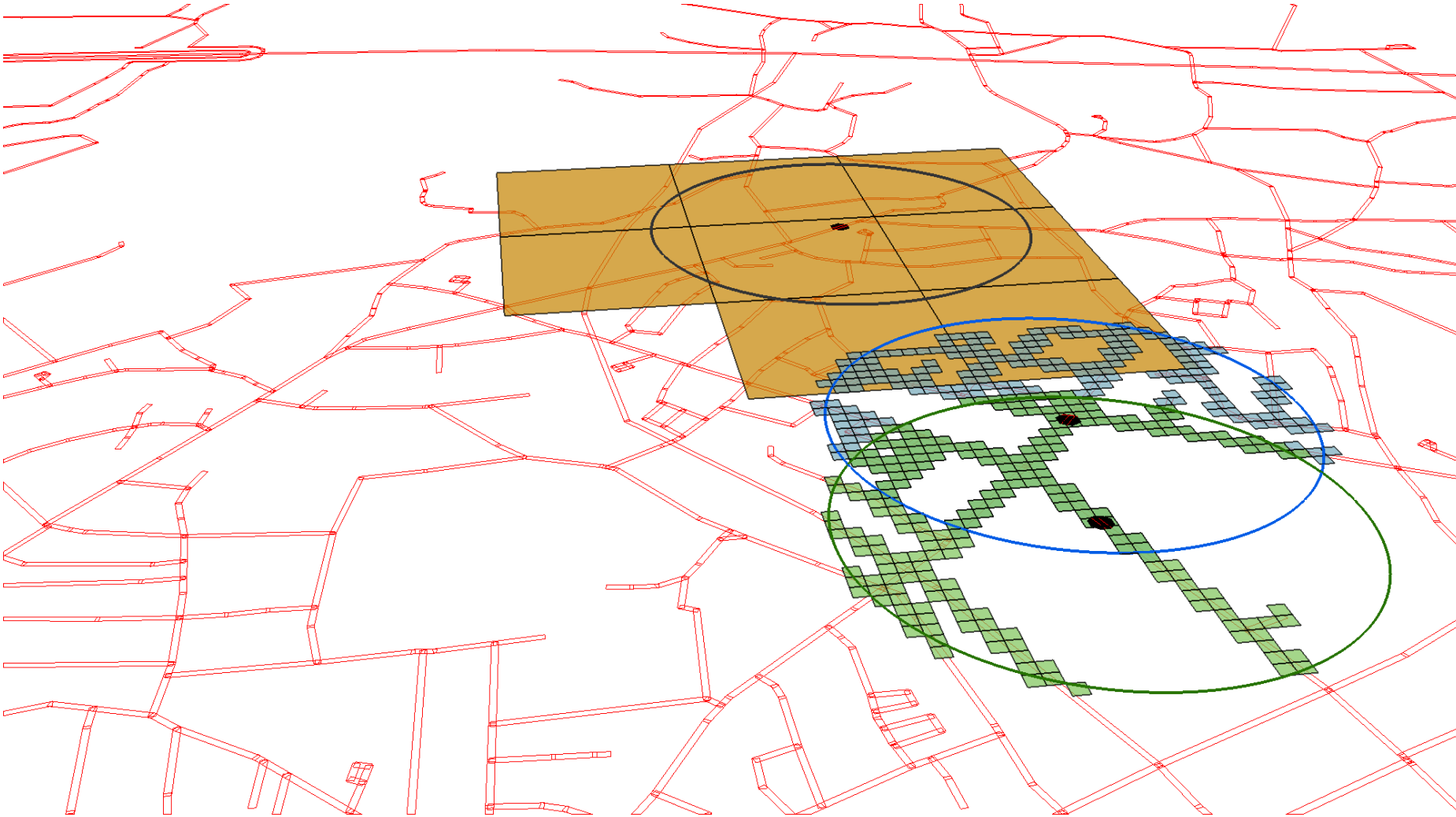
Demo of VicinityLocator with Roadnetwork Filter



Demo of VicinityLocator with Roadnetwork Filter

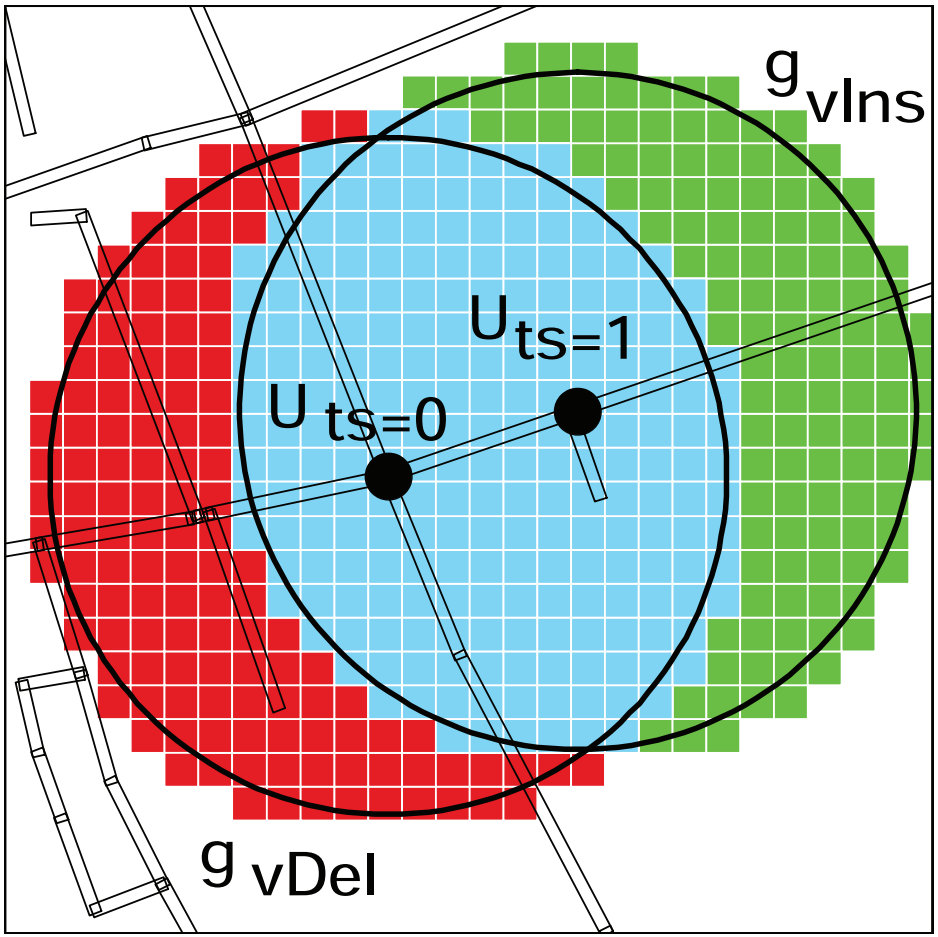


Demo of VicinityLocator with Roadnetwork Filter



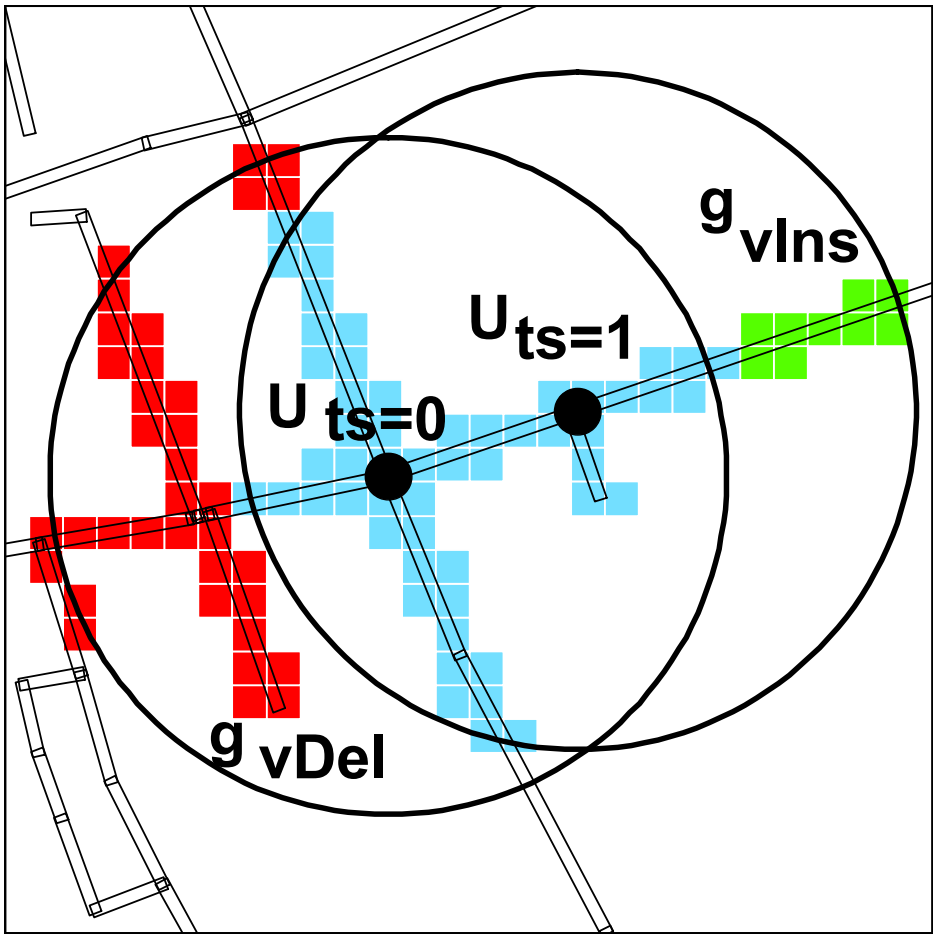
VicinityLocator with Incremental Update

- Incremental Update



VicinityLocator with Incremental Update

- Incremental Update & Roadnetwork Filter



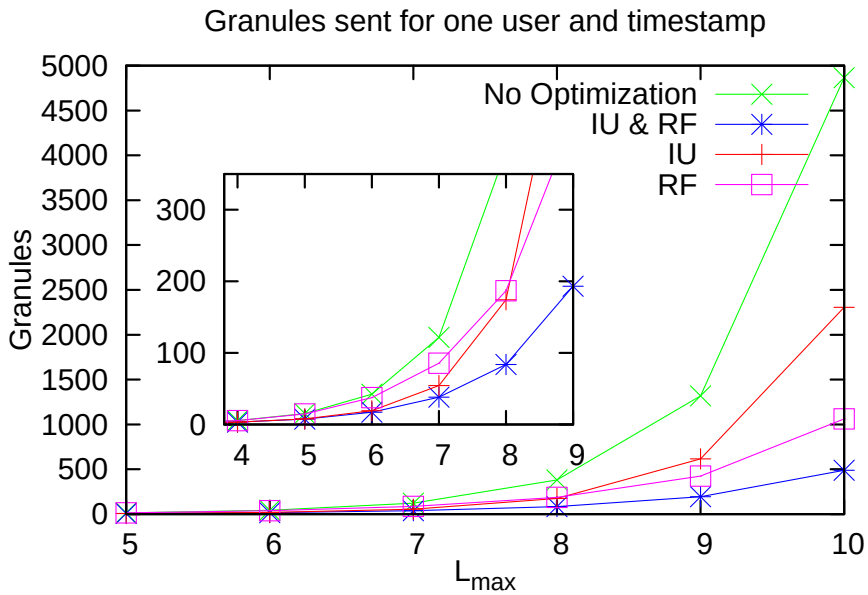
Testing Methodology

System Test Settings

- 50000 users.
- 40 timestamps per user.
- All users partitioned into disjoint groups.
- L_ϵ & L_{max}: 6
- ϵ & $B(L_{max})$: 200 units

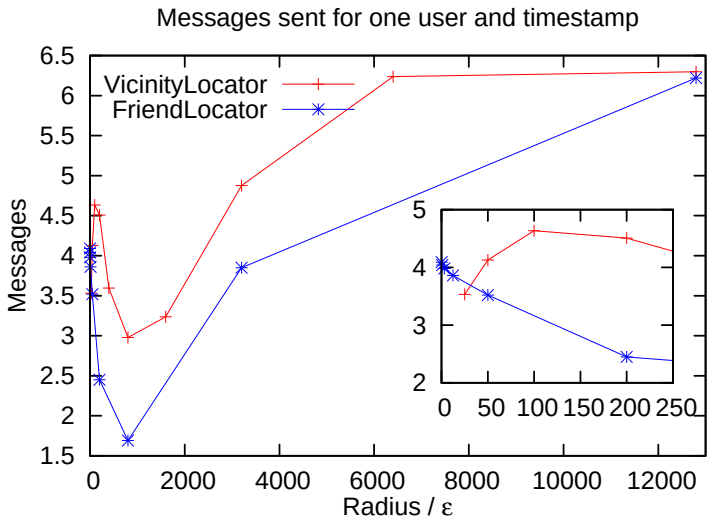
Comparative message development

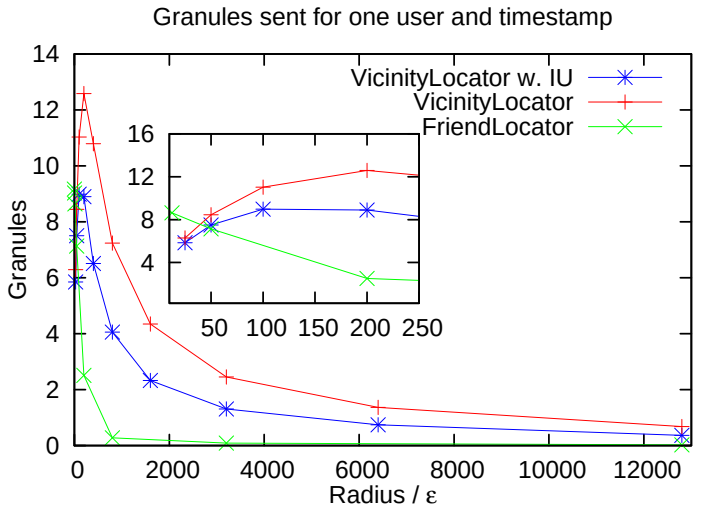
- Small groups, better message ratio
- Communication efficient
- Total messages per user, per update
  - 0.0805, 0.0935, 0.1463, 0.2663



Messages to LS

- less updates over time, on this data

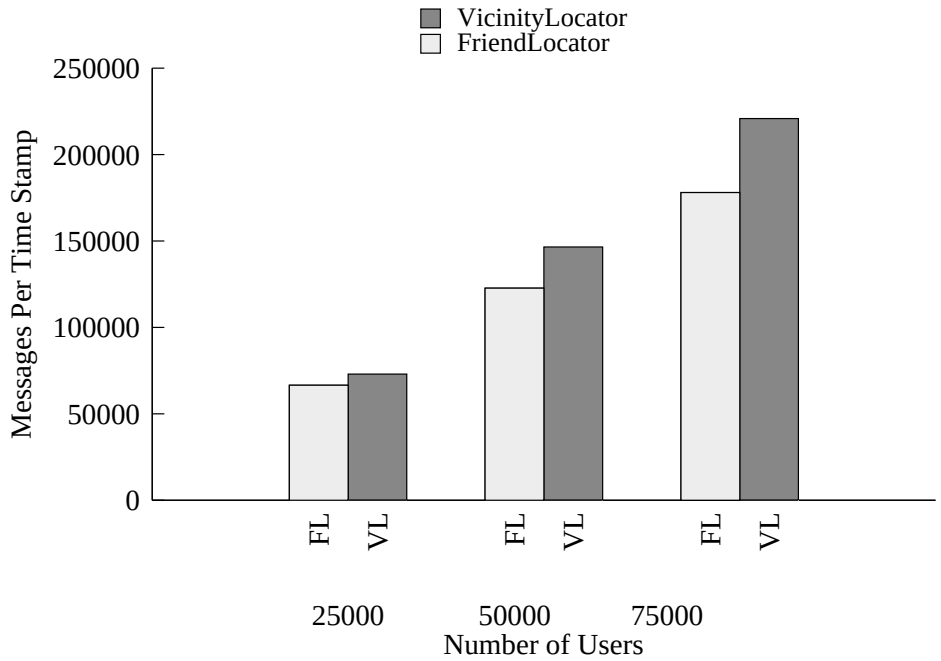




Effect of ϵ_L

- Levels calculated
- Higher ϵ_L, more communication
- Higher ϵ_L, better tracking

User count	Messages	
	FL	VL
25000	2.69	2.92
50000	2.46	2.93
75000	2.38	2.94



Conclusion

- Novel privacy preserving approaches.
- 2 Solutions developed:
  VicinityLocator & FriendLocator
- Gives strong privacy guaranties
- Both approaches strong against attacks
- Both approaches have low operating cost in terms of messages
- Messages vs. Flexibility
  - VicinityLocator offers more flexibility
  - FriendLocator has lower communication cost.

Future Work

Possible extensions
Other Directions

End of Presentation

Thank You For Listening