Large Scale Pattern of Life Simulation for Real-Time Applications

Nick Giannias, CAE

Co-authors: Evan Harris, CAE, Roland Geraerts, Utrecht University, Stijn Herfst, uCrowds, Aidan Hobson-Sayers, Alistair Thorpe, Hadean Supercomputing

Overview

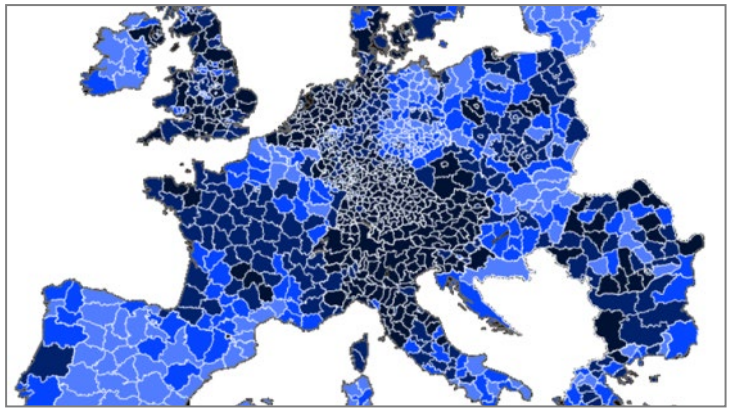
- Overview
- Why pattern of life?
- Related work
- Use cases and architecture
- Population generation and crowd simulation
- Scalability platform
- Visualization
- Conclusion

Why Pattern of Life?

- Develop urban warfighting capability
  - JFCOM Urban Resolve (2003-2007)
  - Massively distributed urban simulation
  - 1M buildings, 120K civilians, 1100 enemy combatants, 250 vehicles and sensors controlled by human-in-the-loop operators
- Study emergence of riots (Pires & Crooks)
  - Kenyan election riots
  - Agent-based population model
  - Incorporating physical, emotional, cognitive, social elements

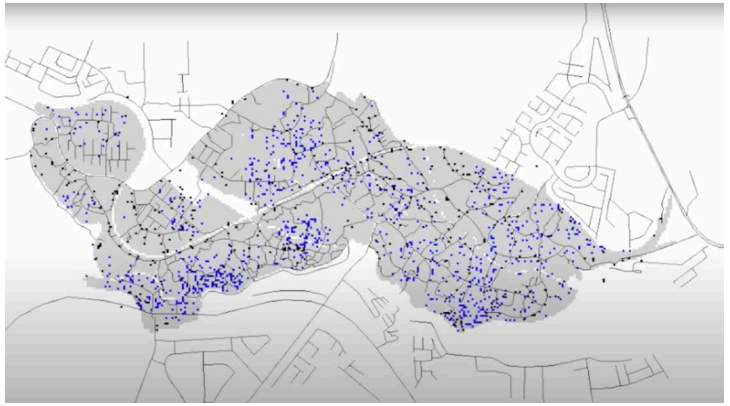
Related Work

- 3 broad categories of pattern of life



Non real-time analytic

Migration studies



Real-time, low update rate

Urban planning

Real-time, 3D interactive

Video games

Demonstrator Use Cases

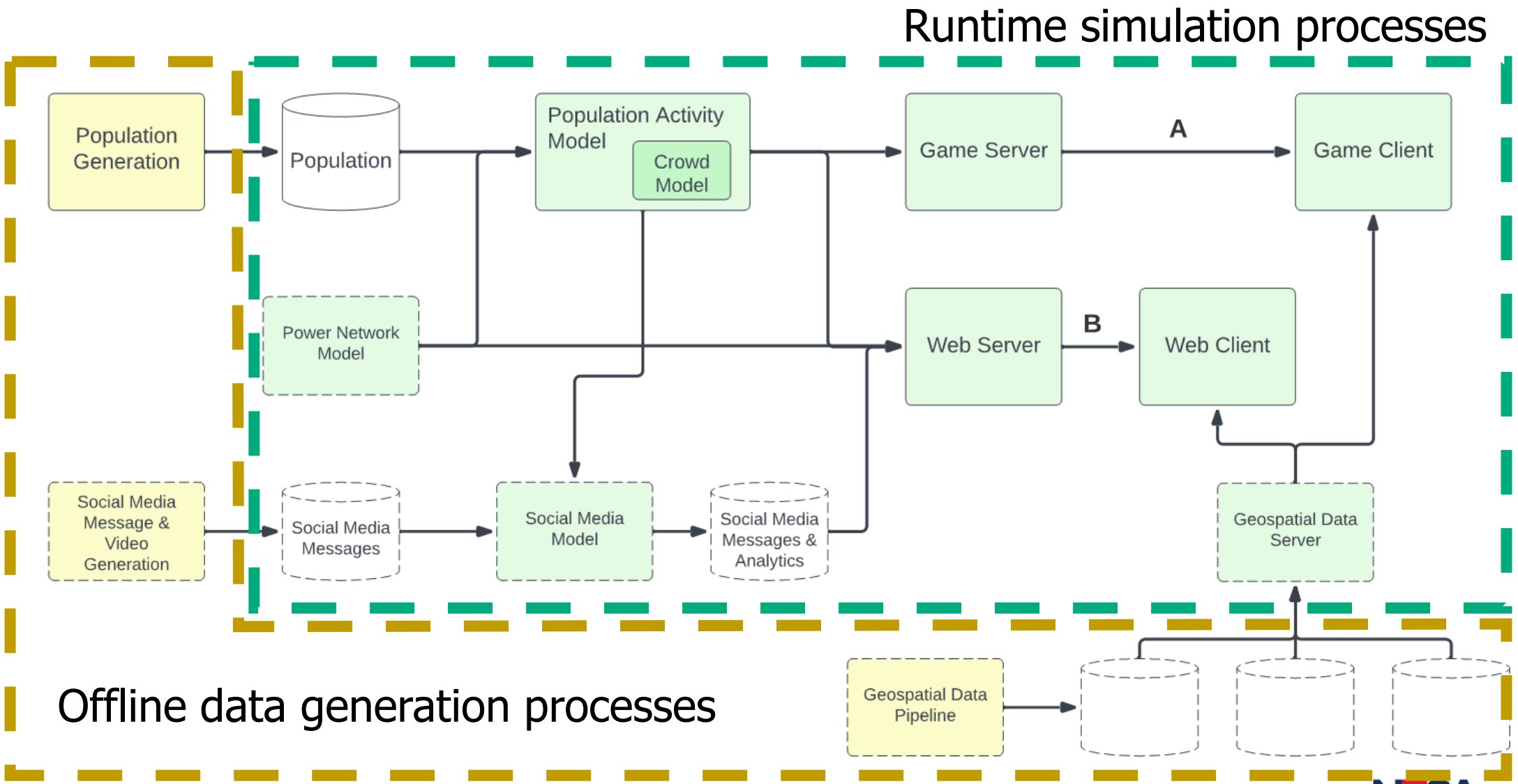
Civil

- Civil resilience scenario involving a cyberattack on the London power network causing a disruption in the patterns of life

Military

- Military course of action with ethnically motivated paramilitary operations and civil unrest in the Baltics

Civil Demonstrator Architecture

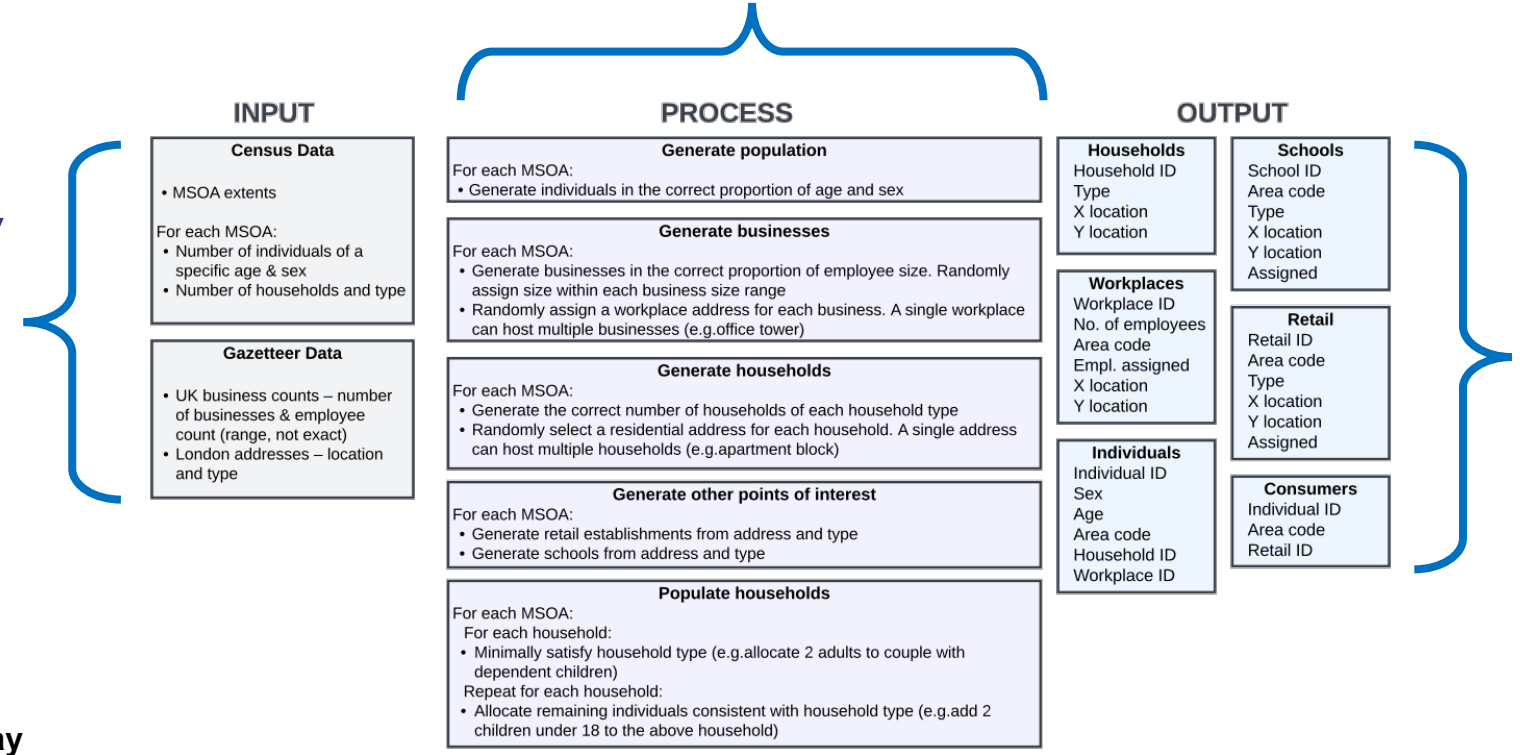


Population Generation

- Multiple existing population generation models available, methodology is similar

Population generation, points of interest, static associations (home ↔ work, home ↔ school)

Census data, location databases, etc



Runtime databases used at initialization and by activity model

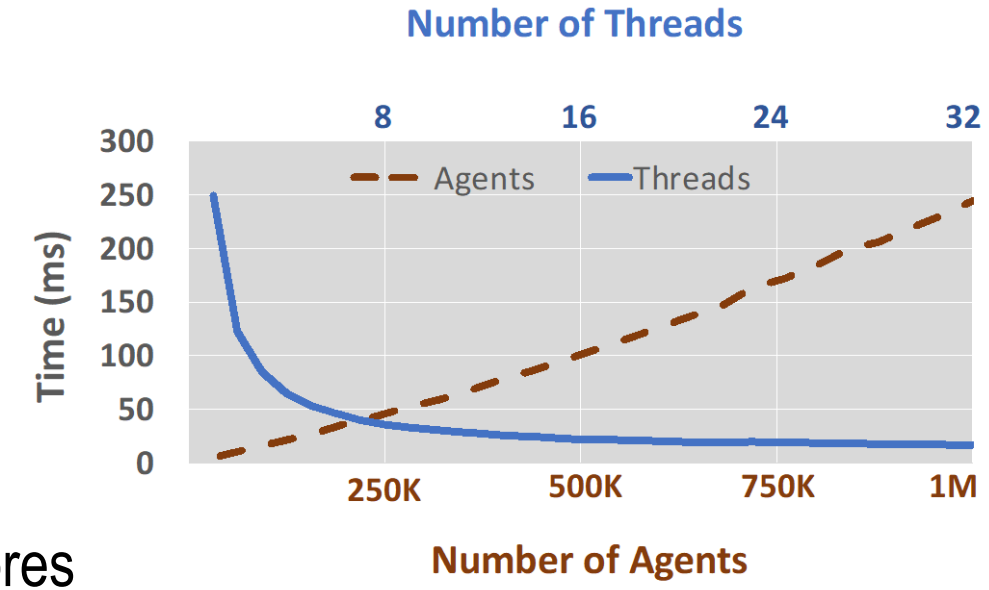
Population Activity Model

- Pattern of life centers around home ↔ work commuting on preset schedules
  - School, shopping, leisure activities not implemented yet
- Short distance commute = walking, long distance = public transit
  - Vehicle activity not modeled; commuting by car not implemented
- Comparison with Silverman's pattern of life characteristics:

	Low	Medium	High
Activities of Daily Life (ADLs)	Pre-scripted ADLs following clock Navigates on own to destinations, avoiding obstacles. ✓	Simple rules to handle a few common ADL issues in 1 area (e.g., errands, OR combat, OR crowds) ✓	Dynamically (re)sets priorities and (re)plans ADLs due to shifts in internal needs and external events ✓?
Nets/Social Skills	Reacts to nearest neighbors on a landscape ✓	Connects to agents across a single-layer network	Has relationships and connections across multi-layer nets
Cognition	Can express values & reactions passed into it from a "god" source	Limited rules to react to a domain (single-layer net)	Adaptively appraises world against its own values.

Crowd Simulation

- Use cases required large, dense crowds at real-time interactive frame rates
- Many simplifying approaches broke down under our requirements:
  - Up to 100K simultaneous pedestrians in an urban square, multiple simultaneous viewpoints
  - Accurate pedestrian evacuation dynamics
  - Data and update rates sufficient to smoothly drive 3D animated characters
- Dedicated, high performance crowd simulation
  - 10 Hz updates, validated dynamics
  - Highly optimized Navmesh
  - Near ideal performance scaling with # of agents, cores

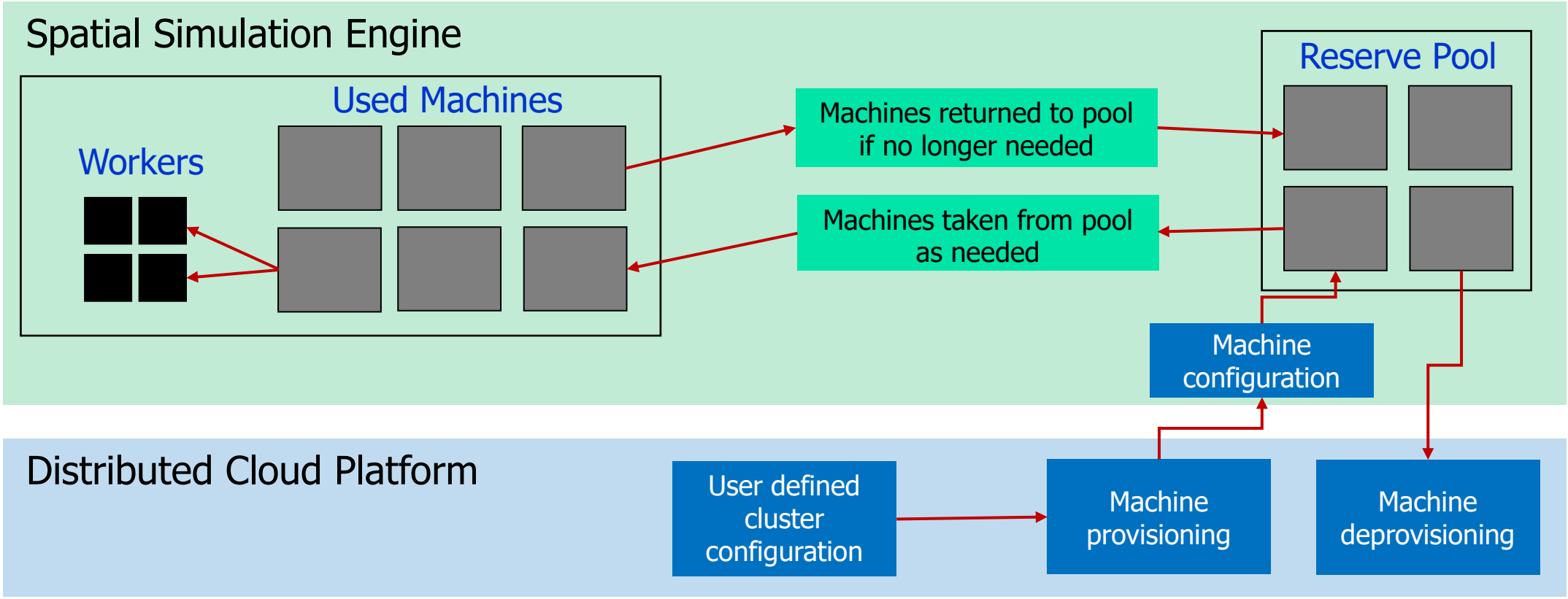


Crowd Simulation

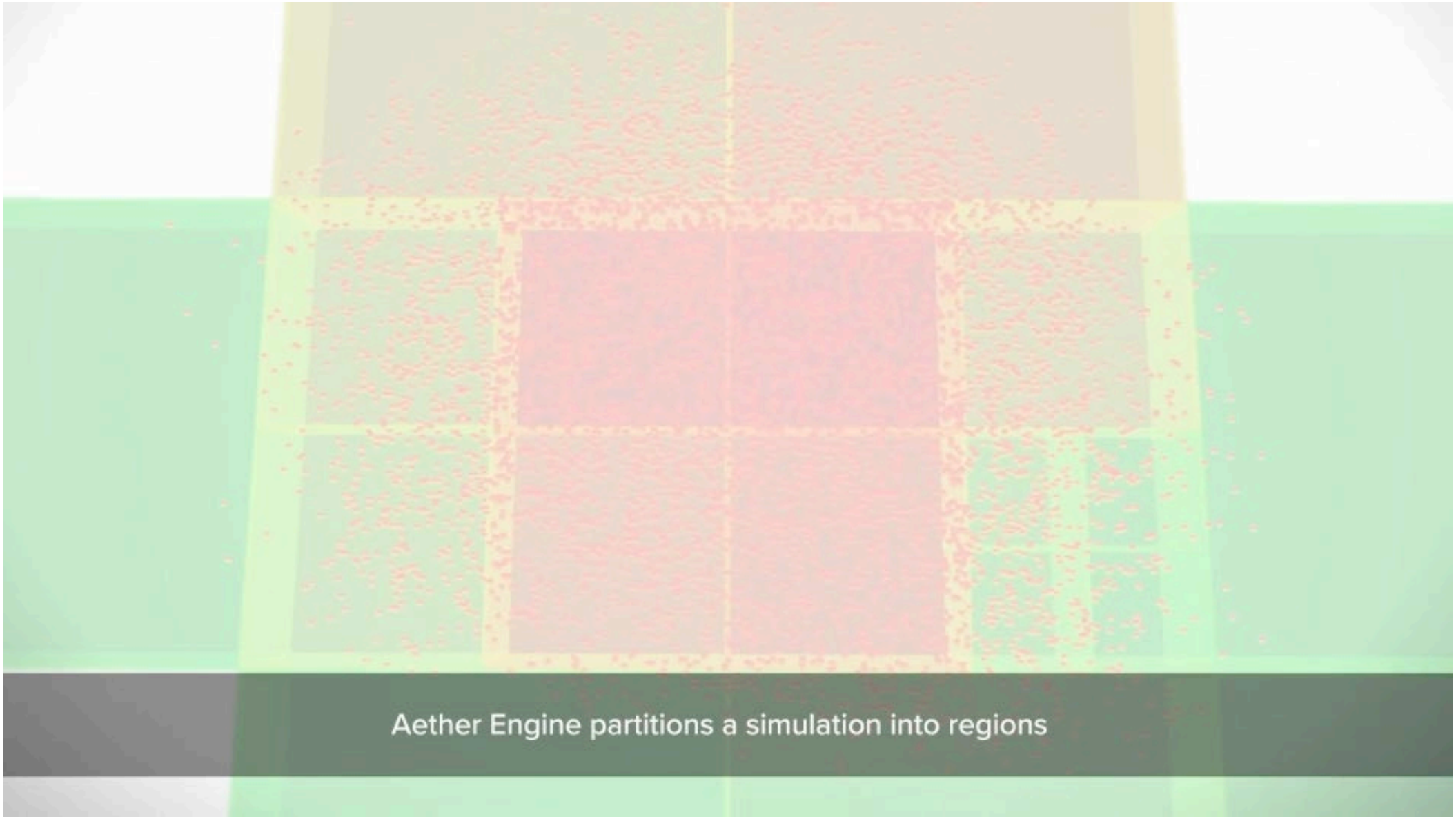
100K civilians converging on Trafalgar Square

Scalability Platform – Architecture

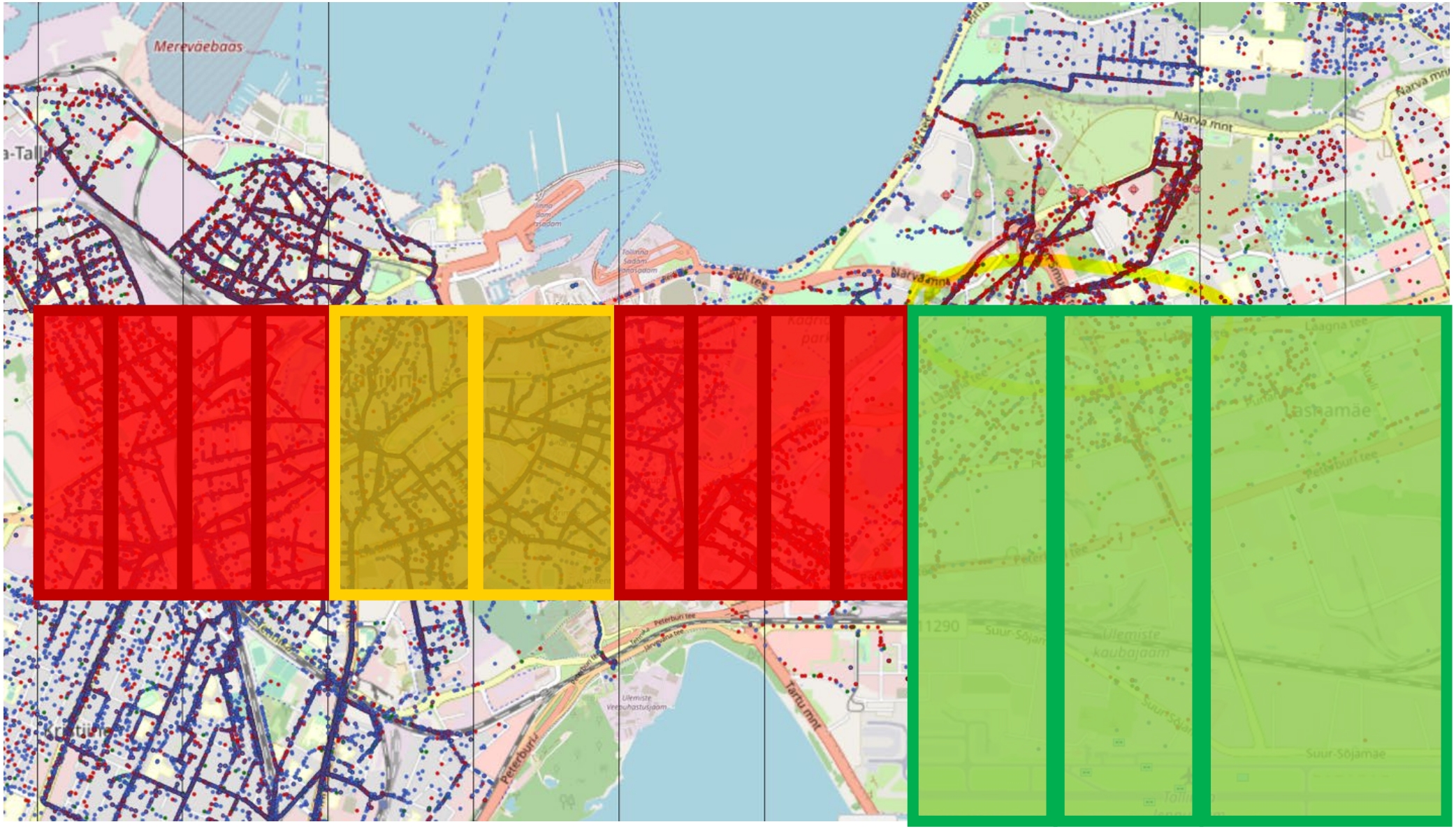
- Scaling to millions of agents requires scaling across virtual infrastructure
- Scalability platform provides machine scaling and simulation distribution/partitioning



Scalability Platform – Spatial Partitioning

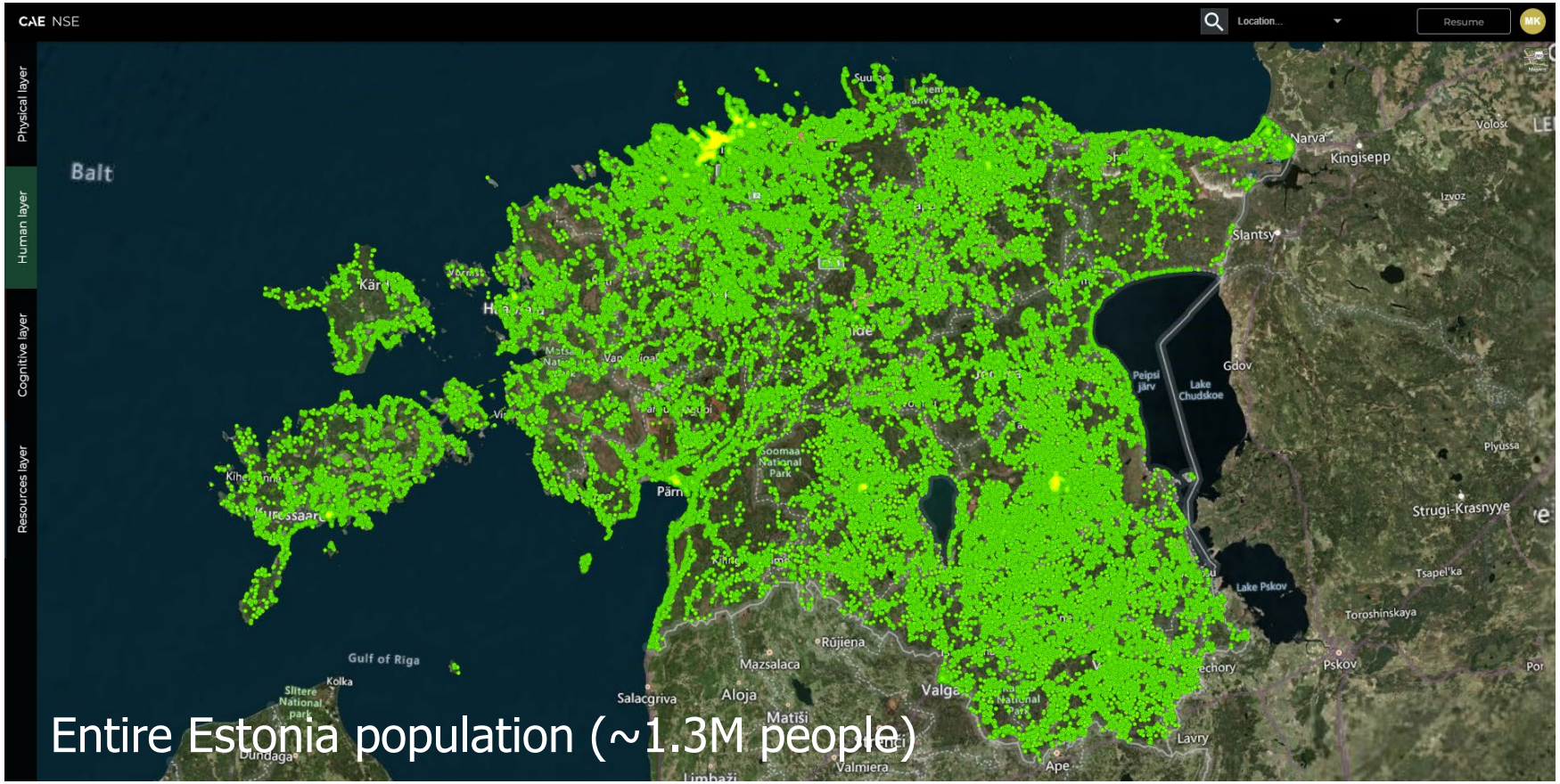


Spatial Partitioning – Tallinn Area

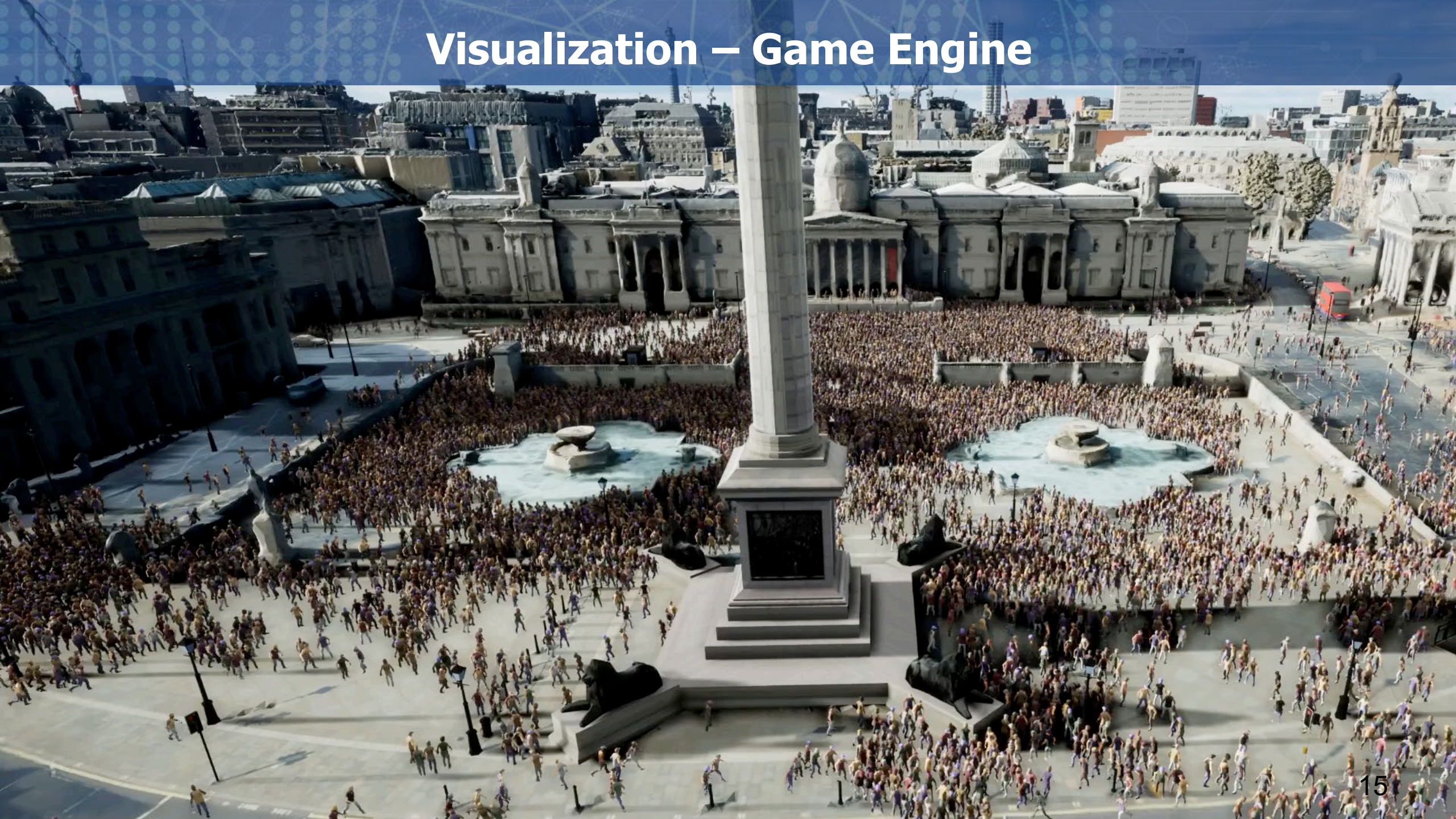


Visualization – "God's Eye View"

- "God's Eye View" presents unique challenges for large scale simulations
- Smart filtering (right data, right place) plus highly optimized network protocols



Visualization – Game Engine



Conclusions

- Large scale, real-time, interactive pattern of life can be achieved
  - Modern frameworks, simulations, game engines and the cloud make it possible
  - At a fraction of the cost and complexity of pioneering efforts like Urban Resolve
- Expect to modify some or all components to meet your use case
  - Crowd simulation was modified to be spatially distributable
  - Game engine networking and rendering pipeline were replaced and/or modified
  - God's Eye View required custom networking/filtering not provided by the scalability framework
- System integration is the majority of the effort when assembling these commercial components
- Rapid advancements are easing the path to scalability