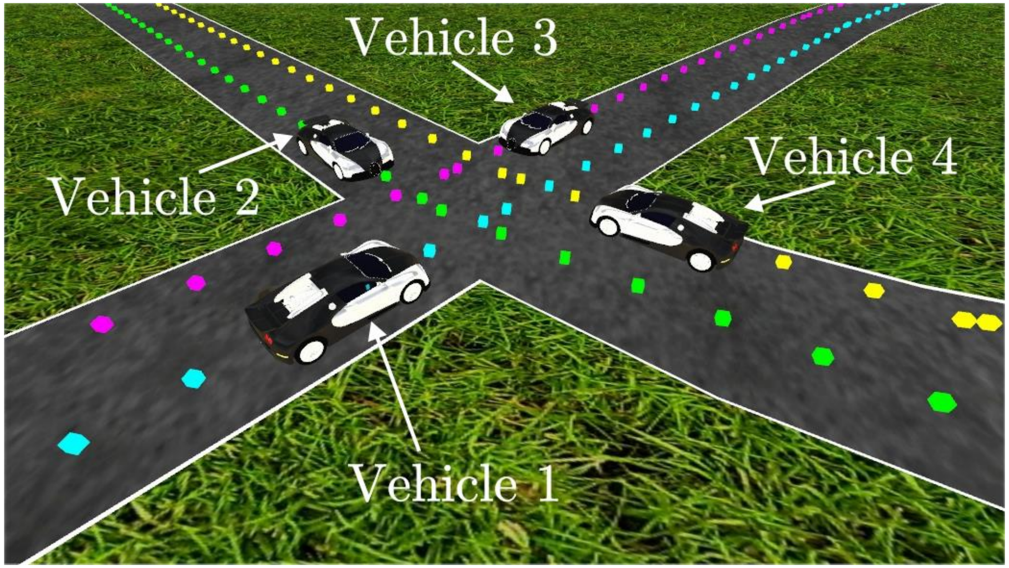
Software architectures of distributed multi-user simulation of autonomous driving vehicles

Petyo Ilov
Master Thesis
Chair of Software Engineering
RWTH Aachen

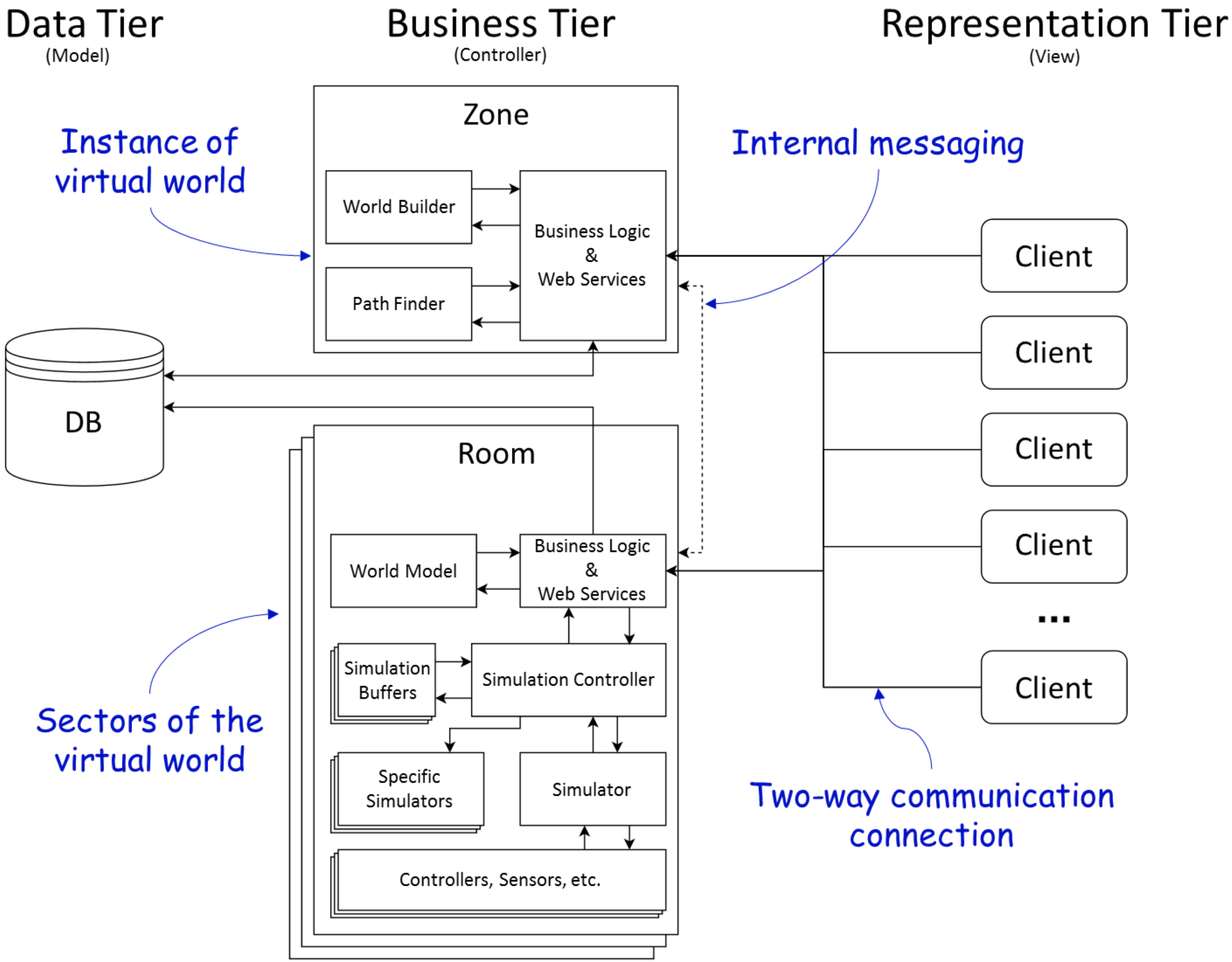
Simulator: Overview

- Aggregate of components:
  - sensors, controllers, simulator, co-simulators, etc.



- Provides extension points for additional simulators
- Simulation and visualization split in 2 separete applications

Architecture: Overview



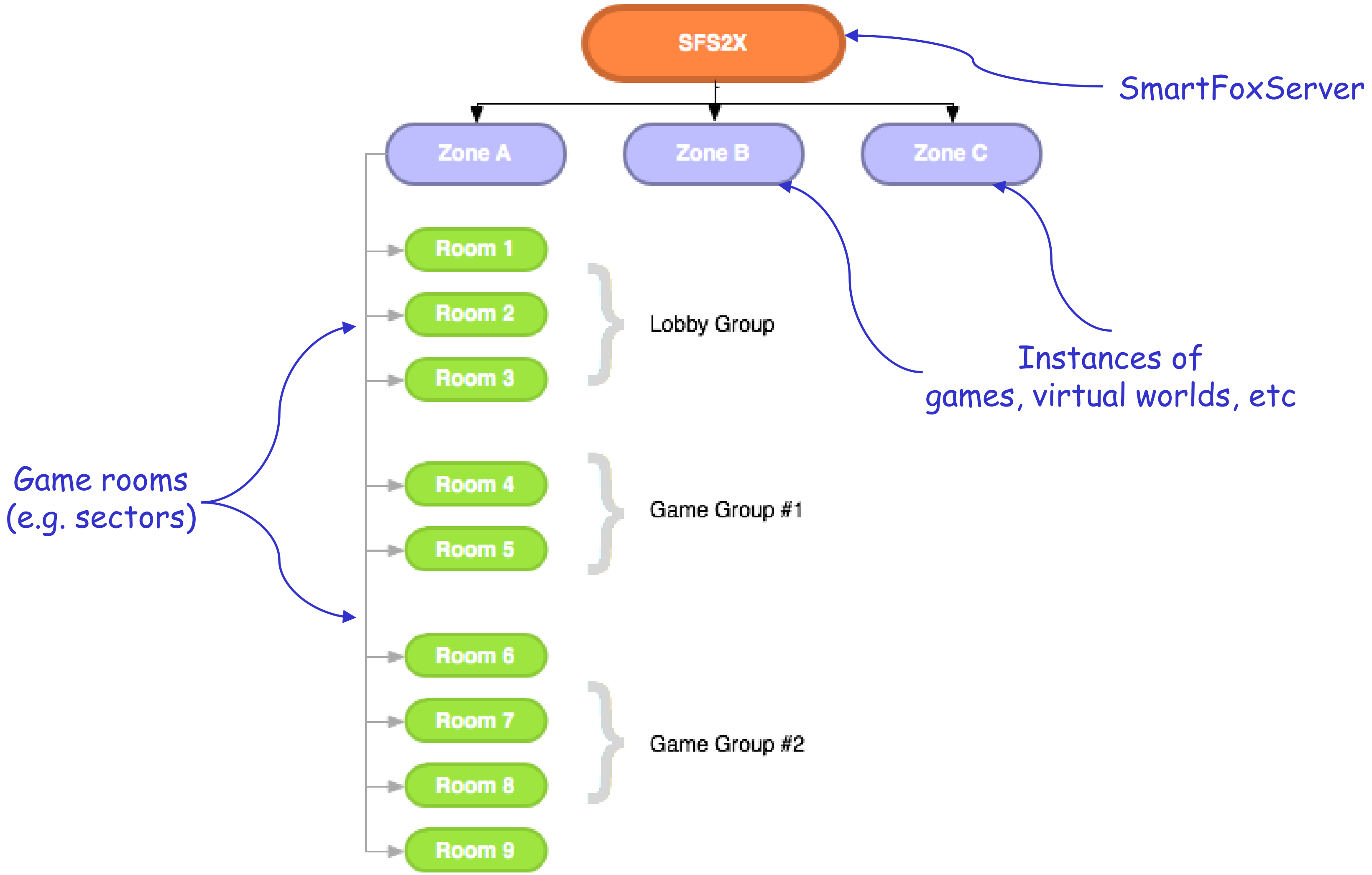
Instance of virtual world

Internal messaging

Sectors of the virtual world

Two-way communication connection

Architecture: Extension scopes



SmartFoxServer

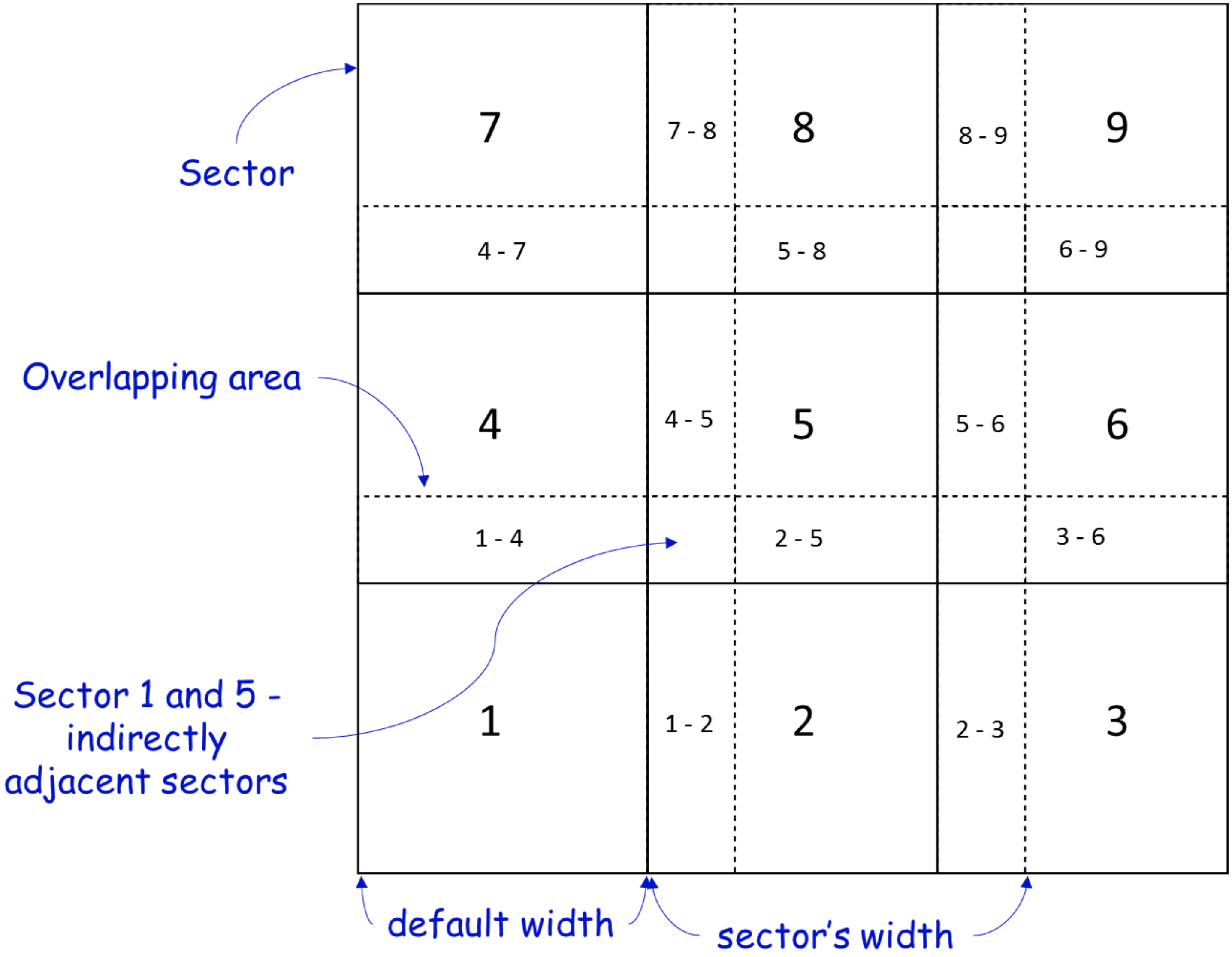
Instances of
games, virtual worlds, etc

Game rooms
(e.g. sectors)

Architecture: Benefits

- Client/Server architecture:
  - Single authority
  - Consistency
  - Security
- Three-tier architecture and MVC pattern
  - Separation of Concerns
  - Loose coupling
  - Scalability
- SmartFoxServer architecture
  - Scope management
  - Segregation
  - Hot-deployment

Map splitting approach: Overview



Sector

Overlapping area

Sector 1 and 5 - indirectly adjacent sectors

default width

sector's width

Map splitting approach: Benefits

- Handling of broad virtual world
- Improved scalability
- Segregation
- Eased synchronization, due to overlapping areas

Path finding: Prerequisites

- Preprocessing of entry/exit nodes for all sectors
- Preprocessing of path matrices for all sectors
- Querying paths for multiple sectors, source and target nodes with a single SQL query

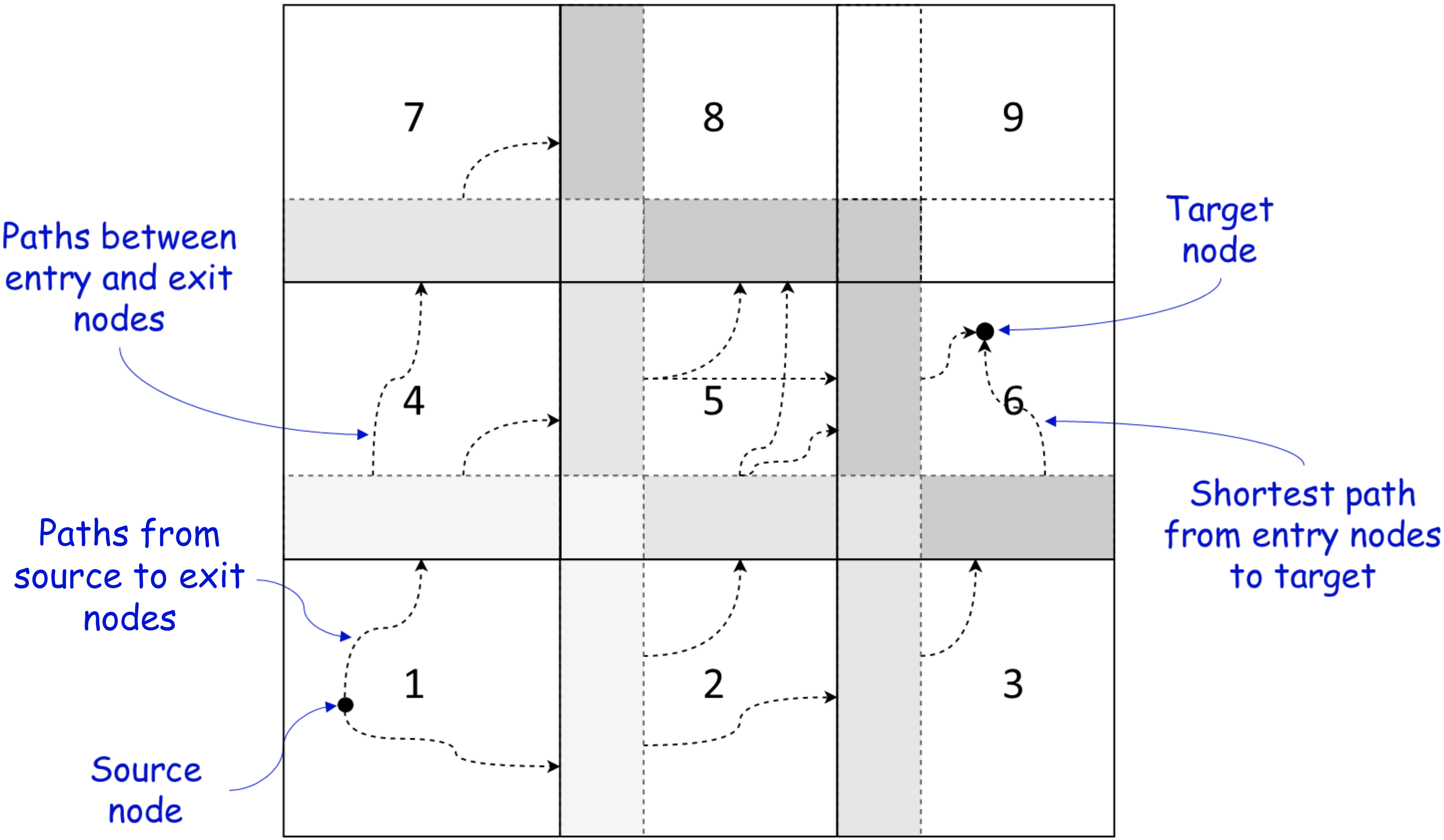
```
SELECT * FROM path WHERE sector_id IN (?)
AND source IN (?) AND target IN (?);
```

- Tracking visited sectors
- Sharing database connection for better performance

Path finding: Workflow

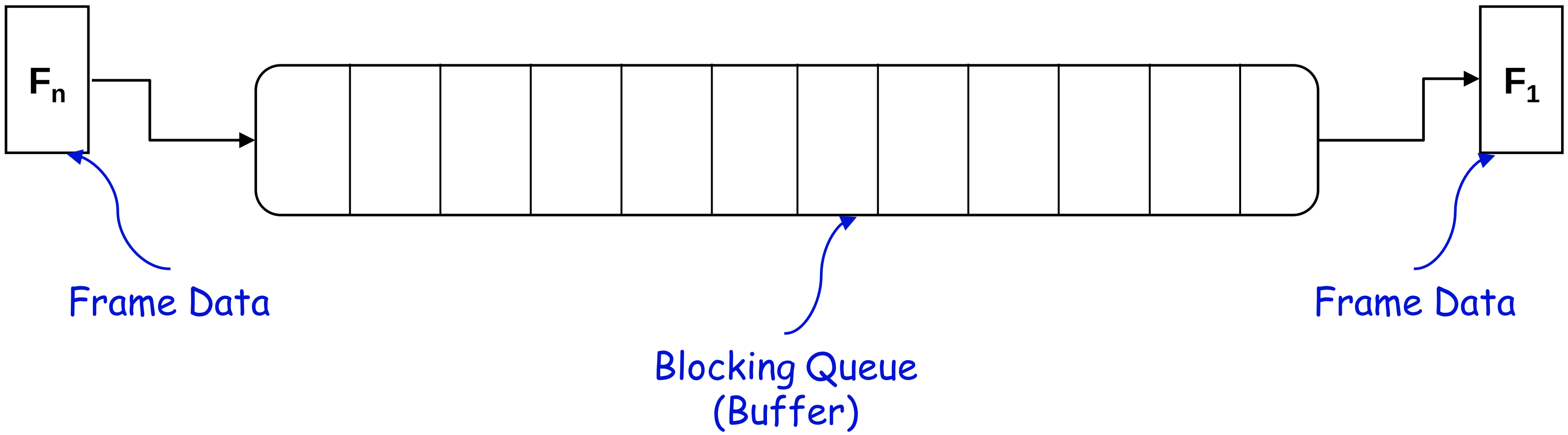
1. Search for direct paths between source and target nodes
   - If such exist, return the shortest path
2. Else find all exit nodes of current sector(s)
3. Find all paths from source node to exit nodes
4. Repeat step 1) using all the exit nodes from step 3) as source nodes
   - If there are returned paths, find the shortest path from the result of step 3), which has a target node equal to the source node of the first path of the returned list of paths.
     - Prepend this path to the list of returned paths and return that list.
   - Otherwise return empty list.

Path finding: Sample


7
8
9
4
5
6
1
2
3
Paths between entry and exit nodes
Target node
Shortest path from entry nodes to target
Paths from source to exit nodes
Source node

Simulation

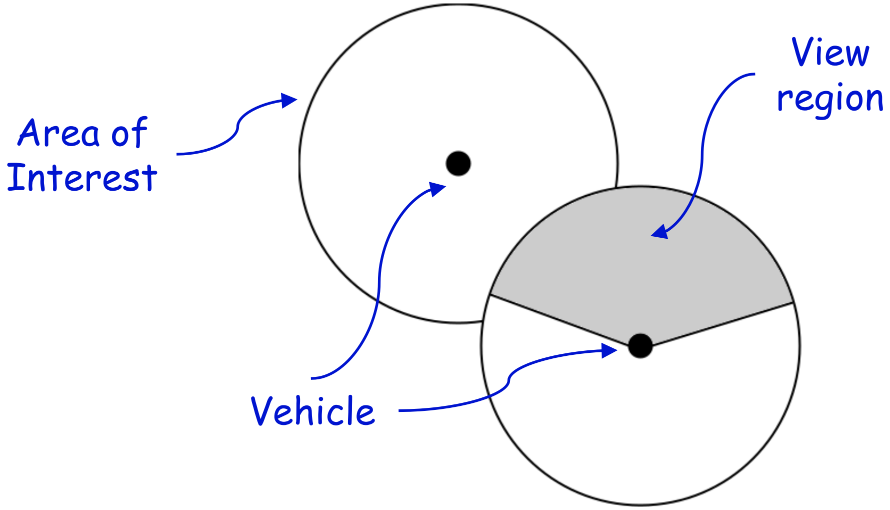
- Loose coupling between WebServices and Simulator
- Allows for additional simulators to work with the main simulator
- Mediator in means of simulation buffers
  - Independent visualization of simulation and vice versa



- Loose coupling between simulators of different sectors
- Addition/removal of vehicles during simulation
- Easy and fast to obtain virtual world data
- Persistent scenarios

Future work

- Area of Interest

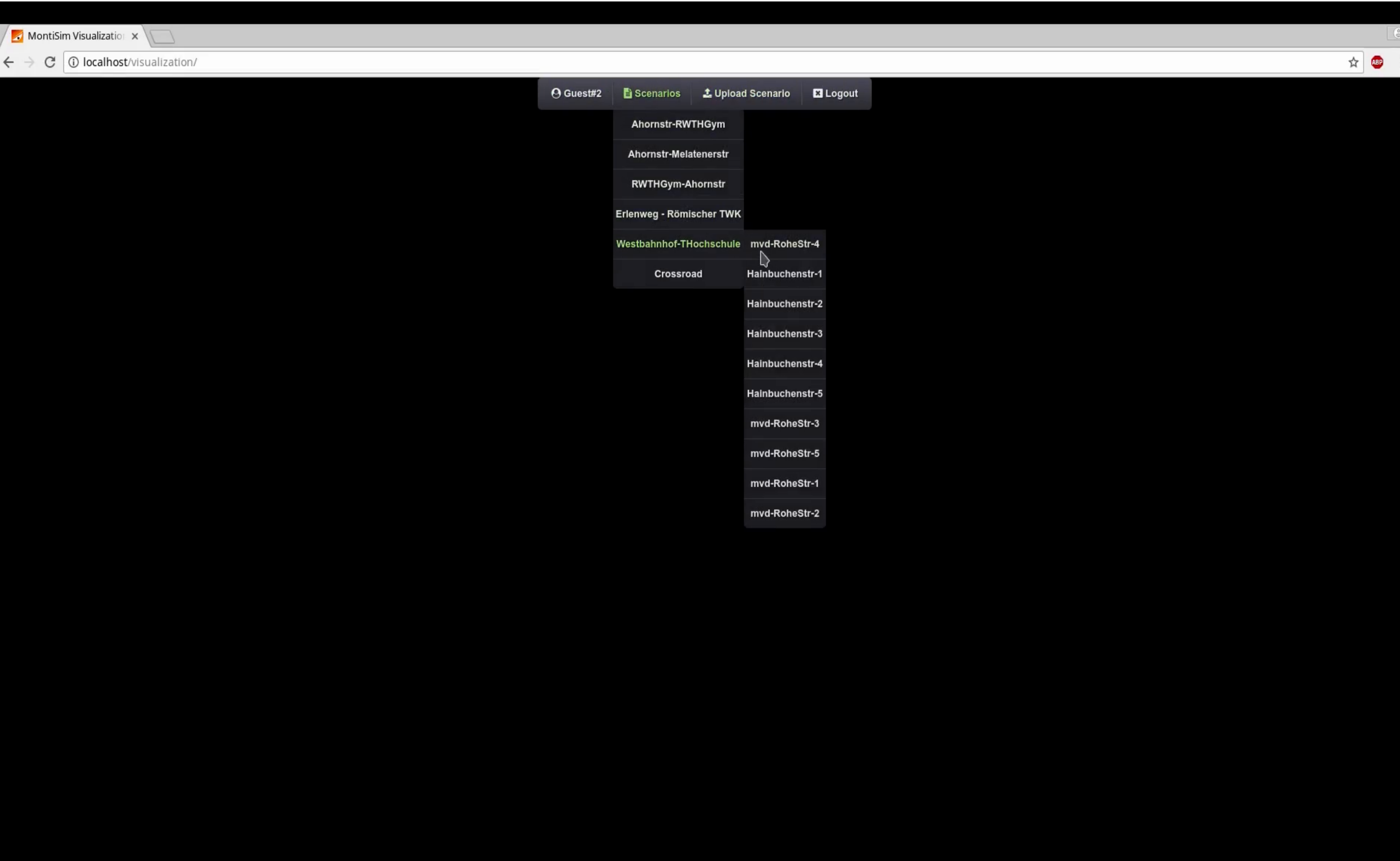


- Saving simulation frames into permanent mediator, e.g. files, database
- Visualizing simulations from archive
- Further integration for model testing

Conclusion

- Scalable architecture
- Distributed simulation
- Persistent map data and scenarios
- Optimized map data loading and path finding
- Automated deployment of code deliverables
- Allows for hot-deployment, thus automatic deployment of new versions in run-time
- Cross-platform server support
- Configurable web-server and simulator
- Hosted client application, thus easy to update
- Web-based client application, thus cross-platform

Demo


MontiSim Visualizatio ×
localhost/visualization/
Guest#2
Scenarios
Upload Scenario
Logout
Ahornstr-RWTHGym
Ahornstr-Melatenerstr
RWTHGym-Ahornstr
Erlenweg - Römischer TWK
Westbahnhof-THochschule
Crossroad
mvd-RoheStr-4
Hainbuchenstr-1
Hainbuchenstr-2
Hainbuchenstr-3
Hainbuchenstr-4
Hainbuchenstr-5
mvd-RoheStr-3
mvd-RoheStr-5
mvd-RoheStr-1
mvd-RoheStr-2

End

Thank you for your attention.