



**Funded by the
European Union**

EuropeanCity2

Project number:	101178170
Project name:	European City Squared: Computational Social Science Simulation for Democracy
Topic:	HORIZON-CL2-2024-DEMOCRACY-01-06
Type of action:	HORIZON-RIA
Starting date of action:	01/01/2025
Project duration:	36 months
Project end date:	31/12/2027
Deliverable number:	D4.2
Deliverable title:	UML Diagrams
EC document version:	Ver1
WP number:	WP4
Lead beneficiary:	1-AU
Main author(s):	Iza Romanowska, ARU
Internal reviewers:	Luis Razo Bravo, EIS, Jan Lorenz, CUB, Nikola Grunchevski, HYB
Nature of deliverable:	OTHER
Dissemination level:	PU
Delivery date from Annex 1:	M10
Actual delivery date:	M7

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Executive Agency (REA). Neither the European Union nor European Research Executive Agency (REA) can be held responsible for them.

Executive Summary

1.1 UML Diagram Overview

The presented UML class diagram depicts the European City 2 domain model architecture using standard object-oriented notation. Rectangular boxes represent a class containing attributes (data fields) and methods (functions). Lines connecting classes indicate relationships with multiplicity indicators (0.., 1.., etc.) showing cardinality constraints. Diamond-headed arrows represent aggregation relationships, while standard arrows indicate associations and dependencies.

1.2 System Architecture

The EC2 model implements a multi-agent electoral simulation system with five core entities operating in a distributed architecture.

1.2.1. Agents

At the heart of the model lies the Agent class, which serves as the primary decision-making entity. Each agent possesses individual preferences, beliefs, and attributes while maintaining connections to other agents and group memberships. The agent's behavioural repertoire includes perception, learning, forecasting, and voting capabilities, enabling sophisticated decision-making processes that mirror real-world electoral participation. These capacities may be extended further to accommodate behaviours required by emerging research needs.

Agent Class: Central autonomous entity with preference vectors, belief systems, and behavioural algorithms. Implements perception(), learn(), forecast(), and vote() methods with preference updating mechanisms. Maintains bidirectional associations with Groups and unidirectional voting relationships with Proposals.

1.2.2 Groups

The Group entity facilitates collective behaviour through membership management, aggregation mechanisms, and broadcasting capabilities. Groups can influence individual agent preferences through social dynamics, creating realistic modelling of political dynamics operating across parties, different demographics and interest groups.

Group Class: Manages collective behaviour through membership aggregation and broadcasting protocols. Implements social influence mechanisms affecting agent preferences through gravitas weighting and group-level decision propagation.

1.2.3 Polis

The Polls class provides the operational backbone for electoral processes, managing agent participation, event scheduling, and temporal dynamics. It can be conceptualised as “the world” in which agents live. The outcome metrics are also calculated here, including perceived fairness scores and trust indices. The aggregation and scoring mechanisms enable comprehensive analysis of electoral outcomes and their perceived validity.

Polls Class: Orchestrates electoral processes through agent management, temporal scheduling, and statistical aggregation. Calculates fairness metrics and trust indices using algorithmic scoring functions. Maintains composition relationship with elections and manages voting system selection.

1.2.4 Proposals

The Proposal entity represents the issues and candidates that agents evaluate and vote upon. The relationship between proposals and agents through voting mechanisms creates a decision-making pathway, while the connection to groups enables collective endorsements and campaigns.

Proposal and Voting Class: The system of proposals represents decision objects in the electoral space with attribute vectors. Receives votes from agents and maintains references to affecting Events, enabling dynamic proposal evaluation.

1.2.5. Events

The Event class orchestrates system dynamics through targeted group impacts and broadcasting capabilities. This enables modelling of campaign events, policy announcements, scandals, and other factors that influence electoral outcomes in real-world scenarios.

Event Class: Implements external shock modelling through targeted group impacts and system-wide broadcasting. Enables simulation of campaign events, policy announcements, and environmental factors affecting electoral outcomes.

1.3 Technical Implementation & System Capabilities

The model employs composition and aggregation patterns to maintain referential integrity while supporting dynamic reconfiguration. The Event-Proposal-Agent interaction chain enables complex causality modelling, while the Group-Agent bidirectional relationship supports emergent coalition formation. The Polls class serves as the primary orchestrator, implementing temporal dynamics and statistical aggregation across the agent population. Key technical features include adaptive preference learning algorithms, multi-stage electoral process simulation, trust-based fairness assessment, group influence propagation, and event-driven preference updating. The architecture supports scalable agent populations with configurable behavioural parameters and multiple concurrent electoral processes.

Abbreviations (please add relevant abbreviations used in document here)

D	Deliverable
EC	European Commission
WP	Work Package
WT	Work Task

EC2 Domain Model

