Università di Genova

LLM-Assisted Interoperability for Autonomous Systems Simulation:

From Local SOMs to a Shared FOM



Marco Gotelli[1], Filippo Ghisi[1], Xhulia Sina[1] and Massimo Pedemonte[2]

[1] University of Genoa, Via Opera Pia 15, 16145 Genova, Italy
[2] Simulation Team, Via Magliotto 2, 17100 Savona, Italy

MESAS 2026 — Modelling and Simulation for Autonomous Systems
PVA EXPO Prague, Prague, Czech Republic · 21–23 October 2026



Connectivity is only the first step



RTI transports the information · participants must still agree on its meaning, representation and use

Position frames

UAV altitude above ground or sea level · UGV local x, y · UUV depth below the surface

Speed quantities

airspeed · ground speed · speed through water: different quantities, often in different units

Operational states

propulsion active ≠ moving: a hovering UAV, a UUV holding station

HLA High Level Architecture

RTI Run-Time Infrastructure

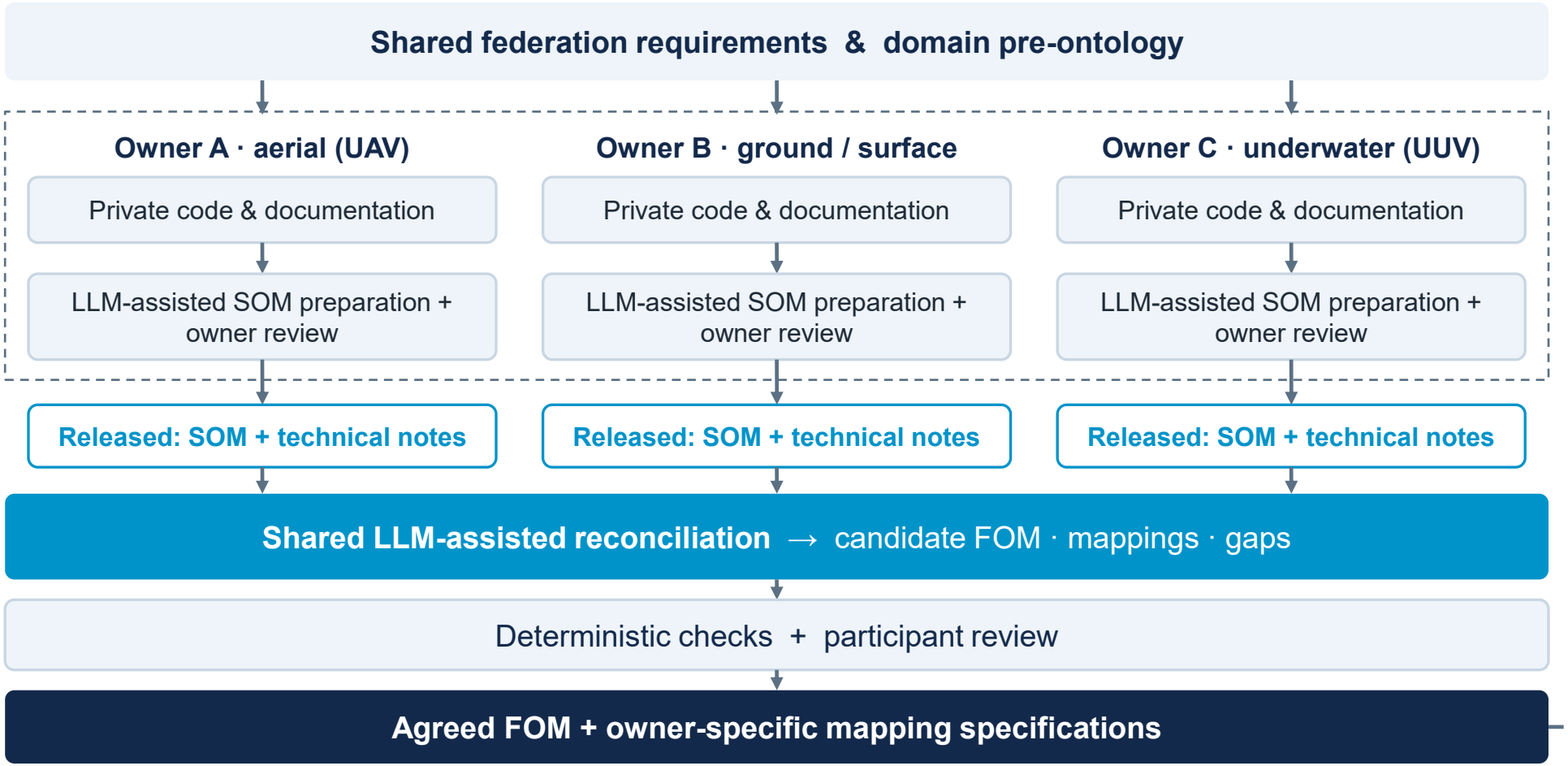
SOM what one simulator can publish / consume

FOM the information model the federation agrees

IEEE Std 1516.2-2010 (OMT) specifies object-model format, not domain content [1]. Domain examples illustrate the mismatch types in paper Sect. 4–5 and Table 1.

Local SOMs, one shared FOM

Proposed workflow



LOCAL
Implementation evidence stays with each owner

BOUNDARY
Only owner-released descriptions cross

Optional
local adapter generation by each owner

Redrawn from paper Fig. 1. Owner columns are conceptual multi-domain examples; the experiment covers a narrower subset (slide 6).

Matching names is not matching meanings

Illustrative example — not original simulator source

```
// C# — illustrative local representation
class PatrolBoat {
    double latitude, longitude;
    float speedKnots, headingDegrees;
    bool enginesActive;
}
```

```
// Java — independently defined representation
class Boat {
    double latitude, longitude;
    float speed, heading; // units require documentation
    boolean isMoving;
}
```

speedKnots ↔ speed

Confirm quantity and unit with the owner first; the Java unit is undocumented, so the mapping stays CONDITIONAL.

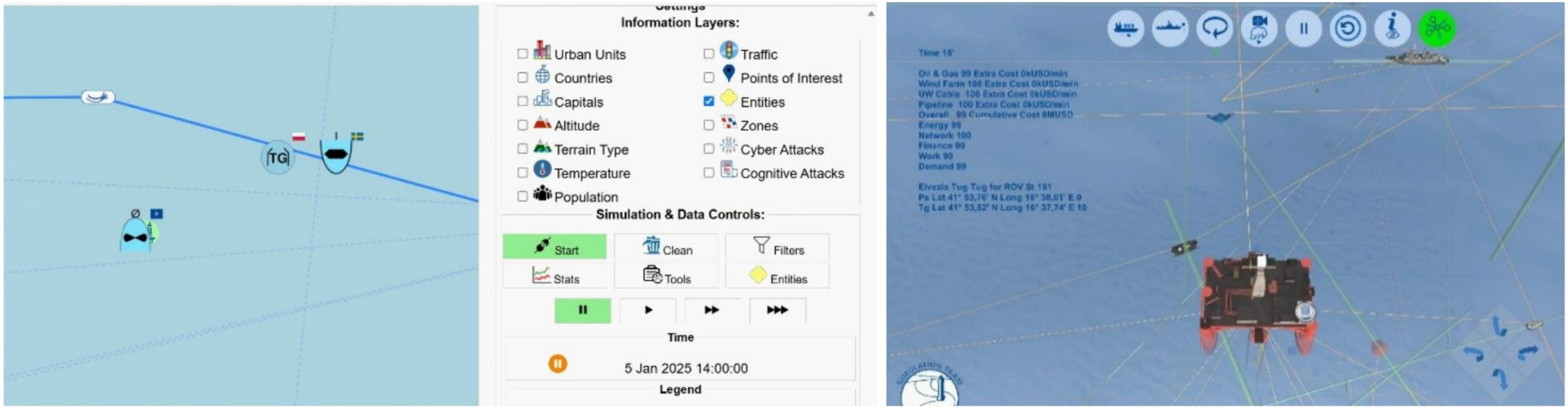
enginesActive ≠ isMoving

Engines can run while the vessel is stationary: keep two separate concepts instead of one merged state.

Code transcribed from the paper's illustrative C#/Java example (Sect. 5, Table 1); written for explanation, not extracted from DecisionTheater or CAPIAS.

Maritime case: two heterogeneous simulators

Illustrative maritime scenario



17 — **DecisionTheater**
Java representation · candidate attributes in its SOM

43 — **CAPIAS**
Unity / C# representation · candidate attributes in its SOM

Reconciliation inputs
Prepared SOMs + technical notes + federation context

Intended use: shared operational picture

Paper Fig. 2 (scenario illustration) and Sect. 6: two SOMs with 17 and 43 candidate attributes, prepared from simplified entity models.

Two tests: model generation and a runtime exchange

1 Model-generation test

Fixed inputs
2 prepared SOMs (17 + 43) + technical notes + federation context

→

5 SFS settings
0 · 25 · 50 · 75 · 100
one GPT-6 Extra High run each

→

5 candidate FOMs
with mappings, adapter requirements and open issues

→

Portico RTI
federation creation with each candidate FOM

2 Runtime exchange test

separate from model generation; one FOM only

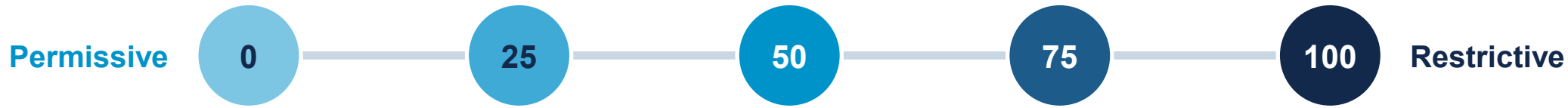
SFS = 50 FOM → **2 lightweight Java test federates**

Initial target: HLA Evolved — IEEE 1516-2010

Paper Sect. 6.2 and Sect. 8. One execution per SFS setting; SOMs, notes and federation context unchanged across runs.

SFS sets how readily concepts merge

SFS is an instruction parameter defined for this study, ranging from 0 to 100 — not an HLA standard term and not a probability.



five values tested, one run each

Toward 0

Broader abstractions and generalised attributes; more mappings left CONDITIONAL on further decisions

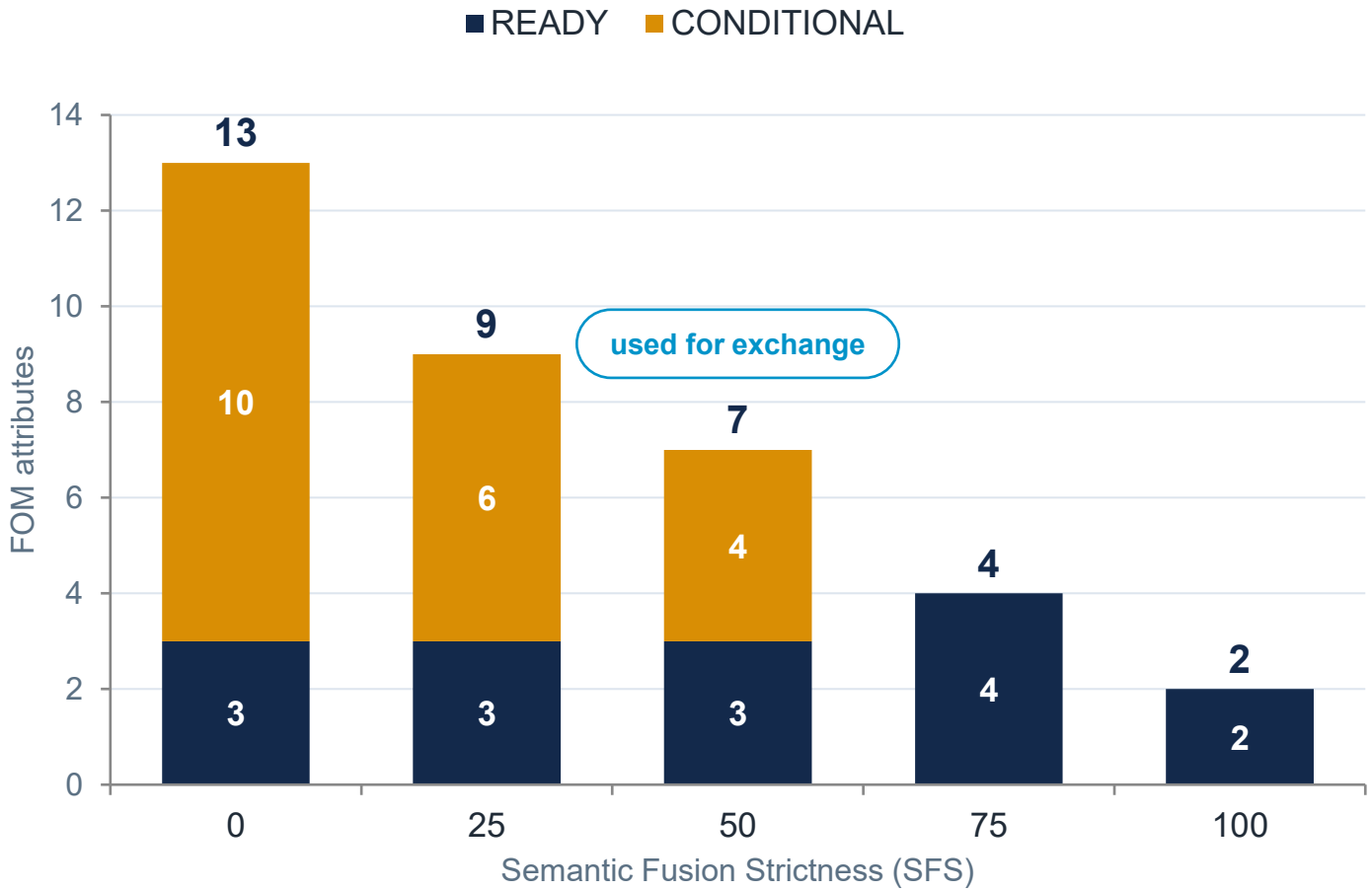
Toward 100

Stronger semantic equivalence and better-documented transformations required before merging

Trade-off: stricter fusion can exclude information the scenario actually needs.

Paper Sect. 7 (Semantic Fusion Strictness). Exact prompt wording and repeatability to be documented in follow-up work.

Stricter fusion produced smaller FOMs



13 → 2 attributes
as strictness rises from 0 to 100

Compactness ≠ success
a discarded mandatory exchange remains an unmet requirement

Five single runs
observations, not an averaged curve or statistical trend

Paper Table 3. READY = mapping specified enough for subsequent implementation and testing (not proven correct); CONDITIONAL = more decisions, information or verification needed.

The SFS = 50 candidate: seven attributes, with notes

Identity

- Federated identifier
- Display name

State

- Geographical position
- Speed

Classification

- Operational domain
- Entity classification
- Affiliation

Aggregate classification

3 READY

4 CONDITIONAL

Status reported in aggregate, not per attribute

Accompanying notes flag what each federate must adapt — examples:

Position
coordinate origin, axes, datum, possible loss of vertical information

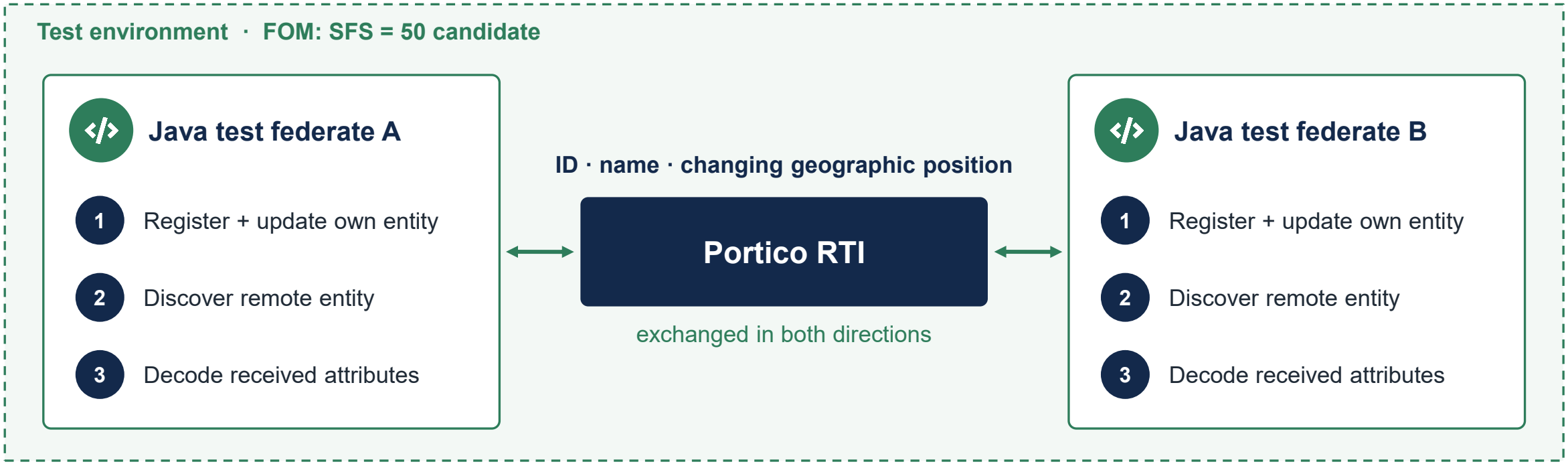
Speed
units to confirm between the two representations

Affiliation
interpretation to agree between owners

Paper Sect. 8: SFS = 50 candidate FOM attributes and technical-note findings (GPT-6 Extra High run).

Basic runtime exchange through Portico

Schematic of the test set-up



All 5 candidate FOMs
accepted by Portico at federation creation

SFS = 50 FOM only
used for the reported bidirectional exchange

Paper Sect. 8. Lightweight test federates; the original DecisionTheater and CAPIAS applications were not connected in this test.

Same SFS, different model, different FOM

Exploratory · one run each · SFS = 50

GPT-6 Extra High

7
FOM attributes

4
CONDITIONAL

GLM-5.2

18
FOM attributes

15
CONDITIONAL

incl. parallel 2D and 3D position representations

Takeaway: outputs are sensitive to the model. Equal SFS values are not calibrated across models, so this calls for controlled, repeated trials — not a model ranking.

Paper Sect. 8: single exploratory GLM-5.2 run at SFS = 50 compared with the GPT-6 Extra High SFS = 50 run.

What the proof of concept shows, and what comes next

Demonstrated

- Candidate FOM synthesis from two prepared SOMs, five SFS settings
- Acceptance of all five FOMs by Portico
- Basic bidirectional exchange with the SFS = 50 FOM

Next to evaluate

- Local SOM extraction from owner code
- Semantic correctness vs. owner-validated reference mappings
- Full adapters for the original simulators
- Repeated, controlled runs
- Manual-effort baseline
- Pre-ontology contribution (ablation)

Next: 3 original simulators → 6 directed exchanges

Scope: air · ground · surface · underwater assets

HLA 4: version-specific checks (IEEE 1516-2025)

Paper Sect. 6.3 (proposed measures, Table 2), Sect. 8.1 and Sect. 9; IEEE Std 1516-2025 [15].

Take-home messages

1. **Make interface meaning explicit**
   units, frames and concepts cannot be inferred from names

2. **Generate more than a FOM**
   mapping conditions and gaps are part of the deliverable

3. **Feasibility shown, validation next**
   this proof of concept motivates full, reference-based evaluation

From local capabilities to an agreed federation interface

Questions? · marco.gotelli@unige.it

Backup: full results and status definitions

SFS	FOM attributes	READY	CONDITIONAL
0	13	3	10
25	9	3	6
50	**7**	**3**	**4**
75	4	4	0
100	2	2	0

GPT-6 Extra High, one execution per setting; READY + CONDITIONAL = FOM attributes in every row.

READY

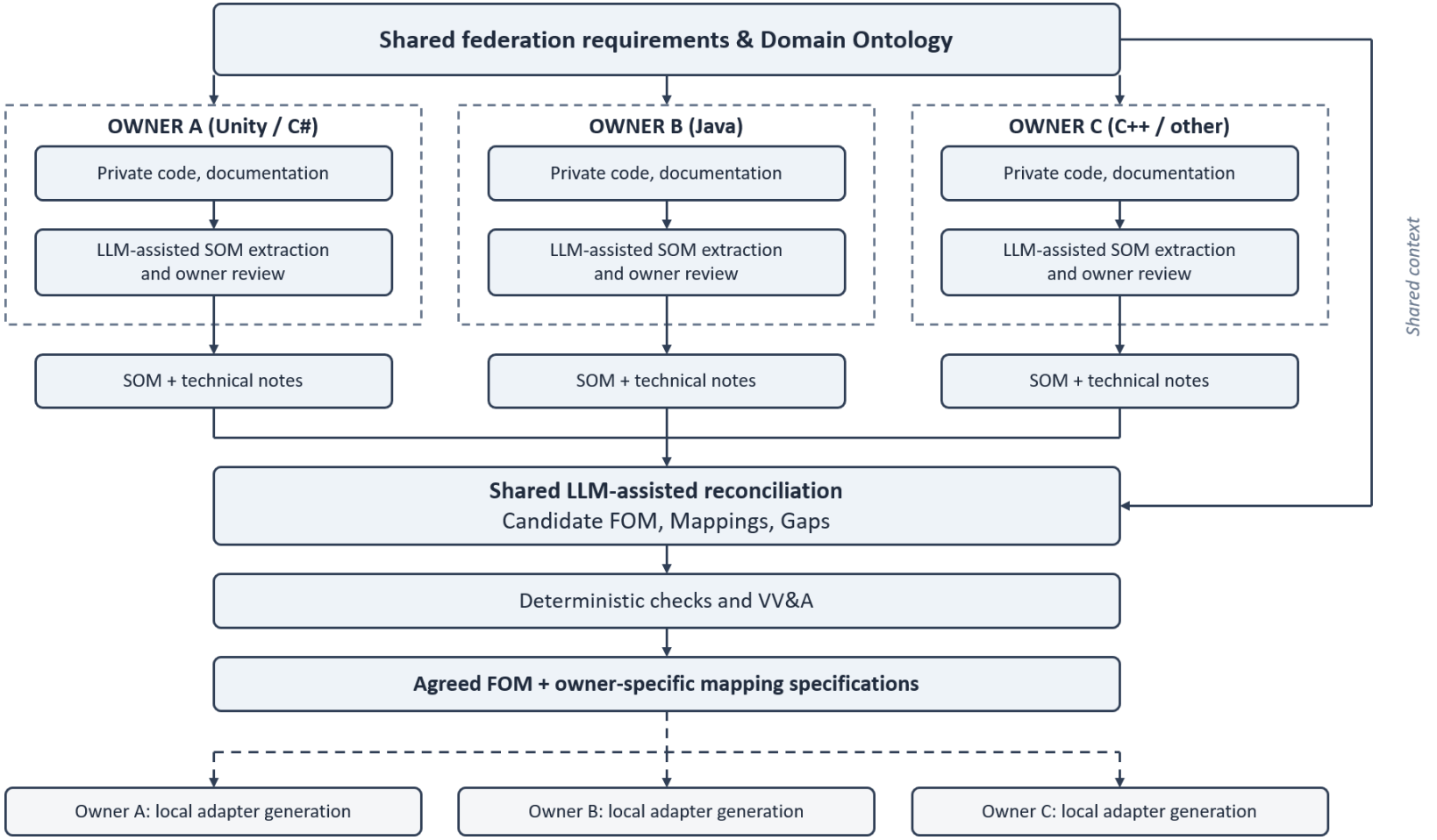
Semantic mapping specified sufficiently for subsequent implementation and testing. Not proven correct; not runtime-tested.

CONDITIONAL

Additional decisions, information or verification required before implementation.

Paper Table 3 and Sect. 8.

Backup: original architecture figure (paper Fig. 1)



Reading notes

- Three owner columns illustrate scalability; they are conceptual.
- “Deterministic checks and VV&A” marks the intended validation stage; deterministic checks alone do not establish accreditation or semantic validity.
- The LLM need not take part in runtime exchanges.

Paper Fig. 1 (original image). Slide 3 shows the same structure redrawn for legibility.

Backup: selected references

[1] IEEE (2010). IEEE Std 1516.2-2010: Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) — Object Model Template (OMT) Specification. https://standards.ieee.org/ieee/1516.2/3746/

[10] Bocciarelli, P., D'Ambrogio, A., Giglio, A., Paglia, E. (2019). Automated Generation of FOM Modules for HLA-Based Distributed Simulations. 2019 Spring Simulation Conference (SpringSim), pp. 1–12. IEEE. https://doi.org/10.23919/SpringSim.2019.8732865

[11] El Kassis, M., et al. (2024). An HLA-based automated approach for the interoperable simulation of collaborative business processes. Simulation Modelling Practice and Theory, 135, 102977. https://doi.org/10.1016/j.simpat.2024.102977

[12] Hertling, S., Paulheim, H. (2023). OLaLa: Ontology Matching with Large Language Models. K-CAP '23, pp. 131–139. ACM. https://doi.org/10.1145/3587259.3627571

[15] IEEE (2025). IEEE Std 1516-2025: Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) — Framework and Rules. https://standards.ieee.org/ieee/1516/6687/

Numbering follows the paper's reference list.