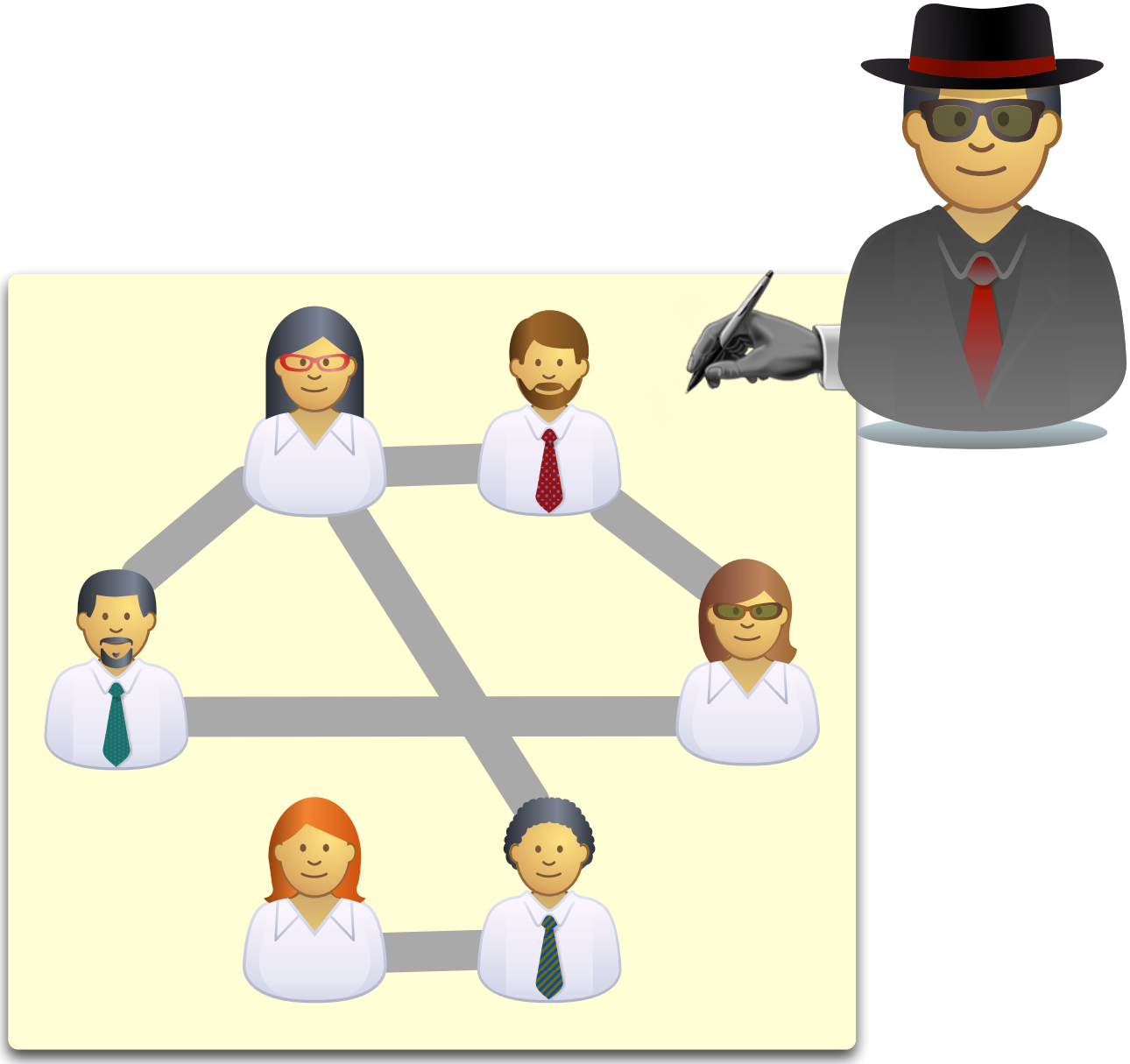
Charters for Self-Evolving Communities

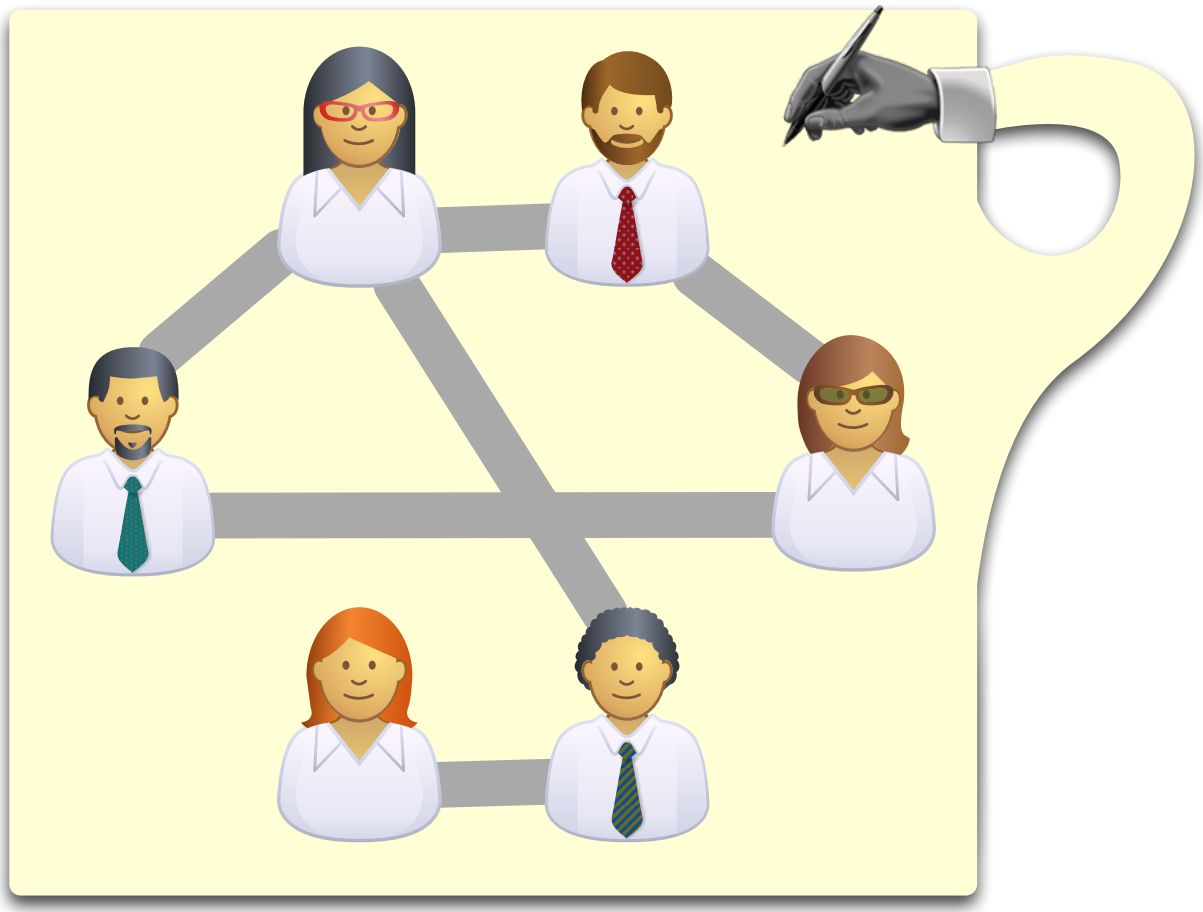
Nardine Osman and Carles Sierra and Marco Schorlemmer

Artificial Intelligence Research Institute (IIIA-CSIC), Barcelona, Spain

Motivation



Motivation



Motivation

e-communities to evolve like human communities

adapt to the evolving aspiration/ needs of community members

adapt to members' character traits

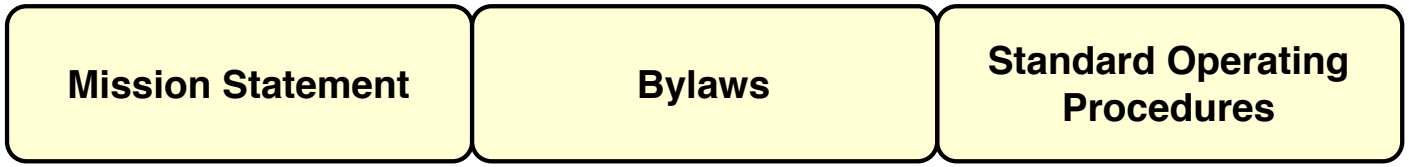
adapt to a community's environmental influences

self-evolving IT tools to replace the numerous rigid systems

empower community members

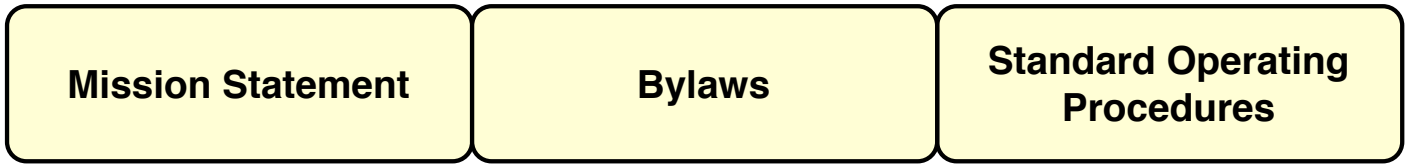
Proposal

Defining human communities:

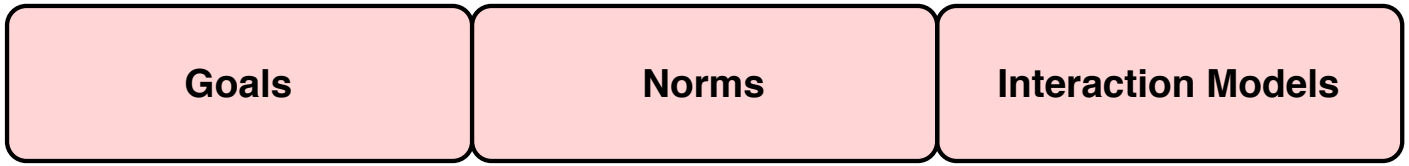


Proposal

Defining human communities:

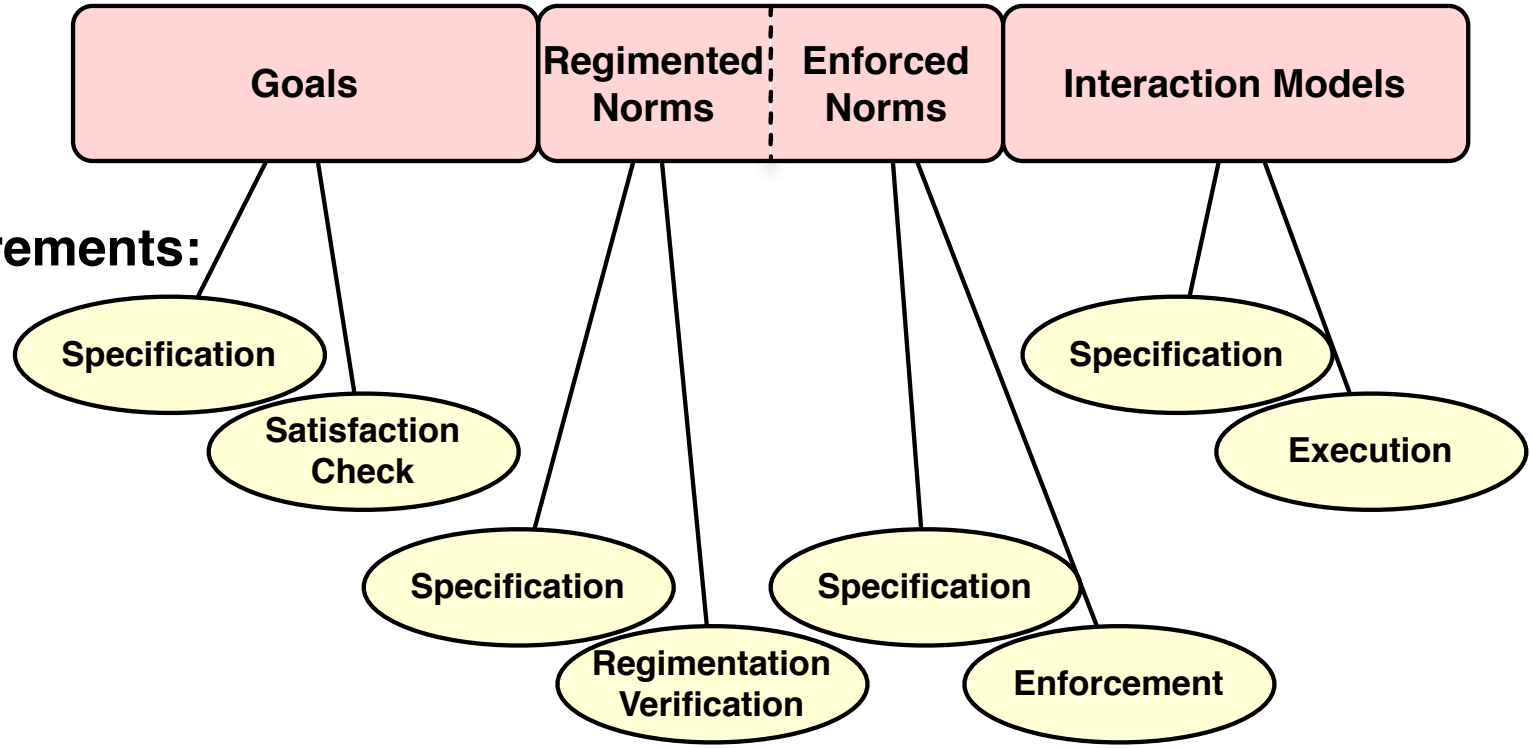


Charters for defining self-evolving IT communities:



Proposal

Charters for defining self-evolving IT communities:



Real World Application

Helpful Communities: The uHelp app

Goals Example

uHelp's Goals:

G1. To ensure the community's needs for services are being addressed

G2. To ensure the satisfaction of requesters (quality of service)

uHelp's Goals*

$$\langle G1, \exists R' \subset R \cdot (\forall r \in R' \cdot \exists m \in M \cdot volunteer(r) = m \wedge majority(R', R)) \rangle$$

$$\langle G2, \exists R' \subset R \cdot (\forall r \in R' \cdot \exists m \in M \cdot volunteer(r) = m \wedge pstvRate(m, r) \wedge majority(R', R)) \rangle$$

* Specified in first order logic

Goals' Syntax

$$\langle GId, GSpecification \rangle$$

Norms Example

uHelp's Regimented Norms:*

RN.3. Volunteers can live outside uHelp community area and join uHelp to join in activities and help those who live in the uHelp community area.

uHelp's Regimented Norms*

$$\langle RN.3, permissible, member(V), volunteer(V, Task), live\_outside\_uhelp\_area(V)\rangle$$

* Specified in first Prolog style

Regimented Norms' Syntax

$$\langle NormId, NormType, Agents, Action, Condition\rangle$$

* Norms copied from Camden's Time Bank community rules.

Norms Example

uHelp's Enforced Norms:

EN.1. Volunteers are penalised by losing credit if they do not fulfil their duties on time.

uHelp's Enforced Norms*

$$\langle EN.1, obligatory, volunteer(V), fulfil\_duty(V, Task), assigned\_duty(Task, V), gain\_points(Task), lose\_points(Task), deadline(Task)\rangle$$

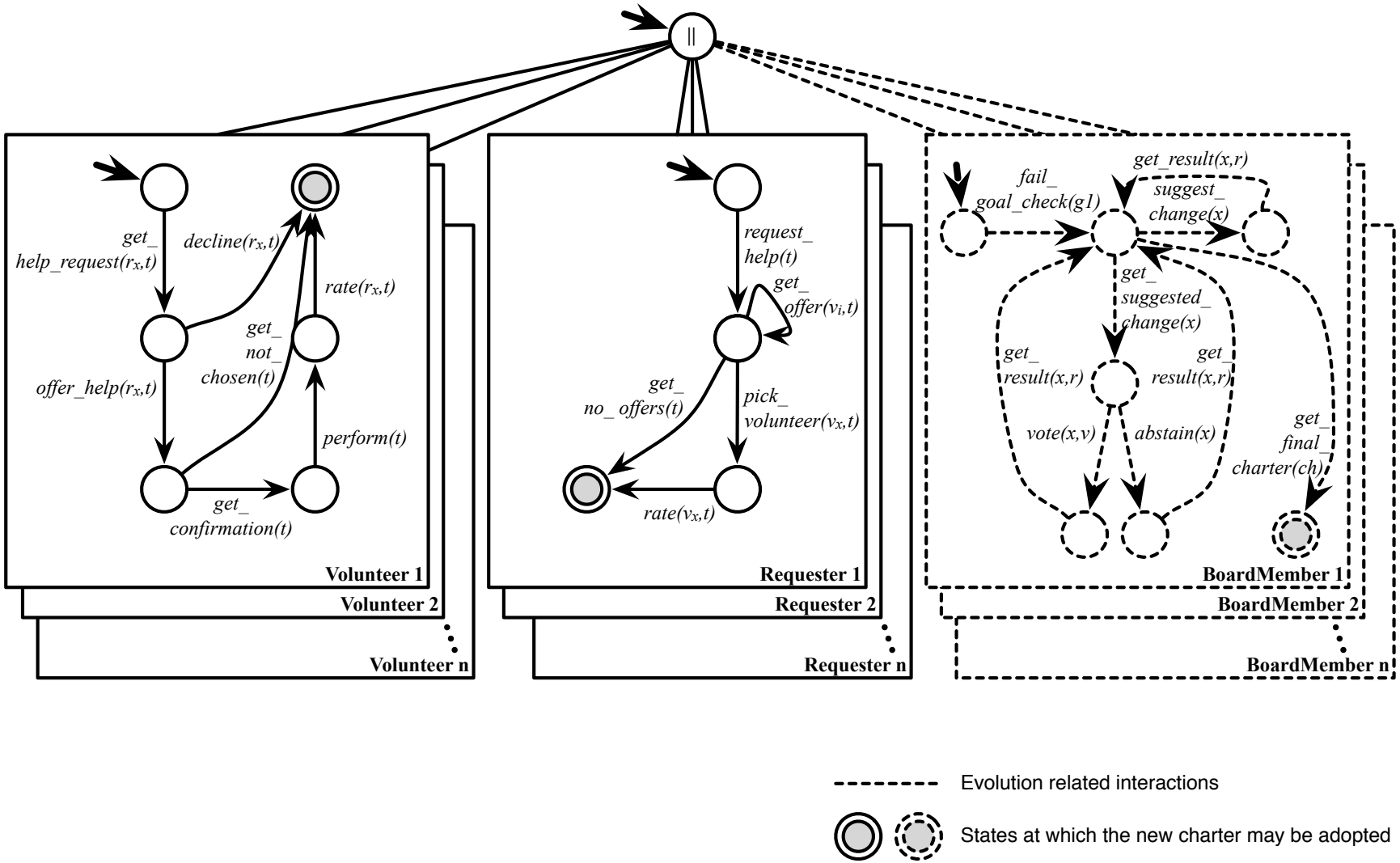
* Specified in first Prolog style

Enforced Norms' Syntax

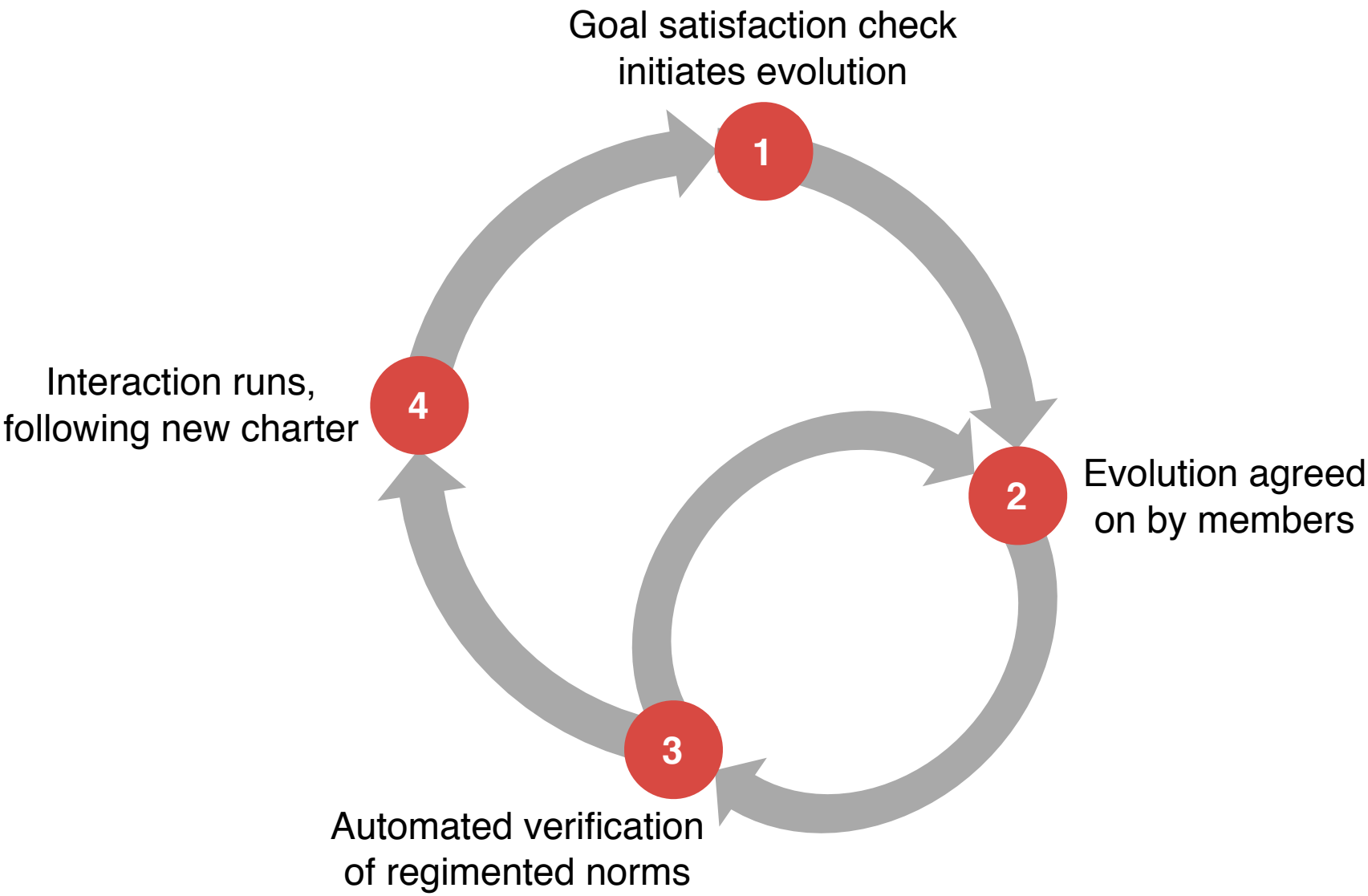
$$\langle NormId, NormType, Agents, Action, Condition, Reward, Punishment, Deadline\rangle$$

Interaction Models Example

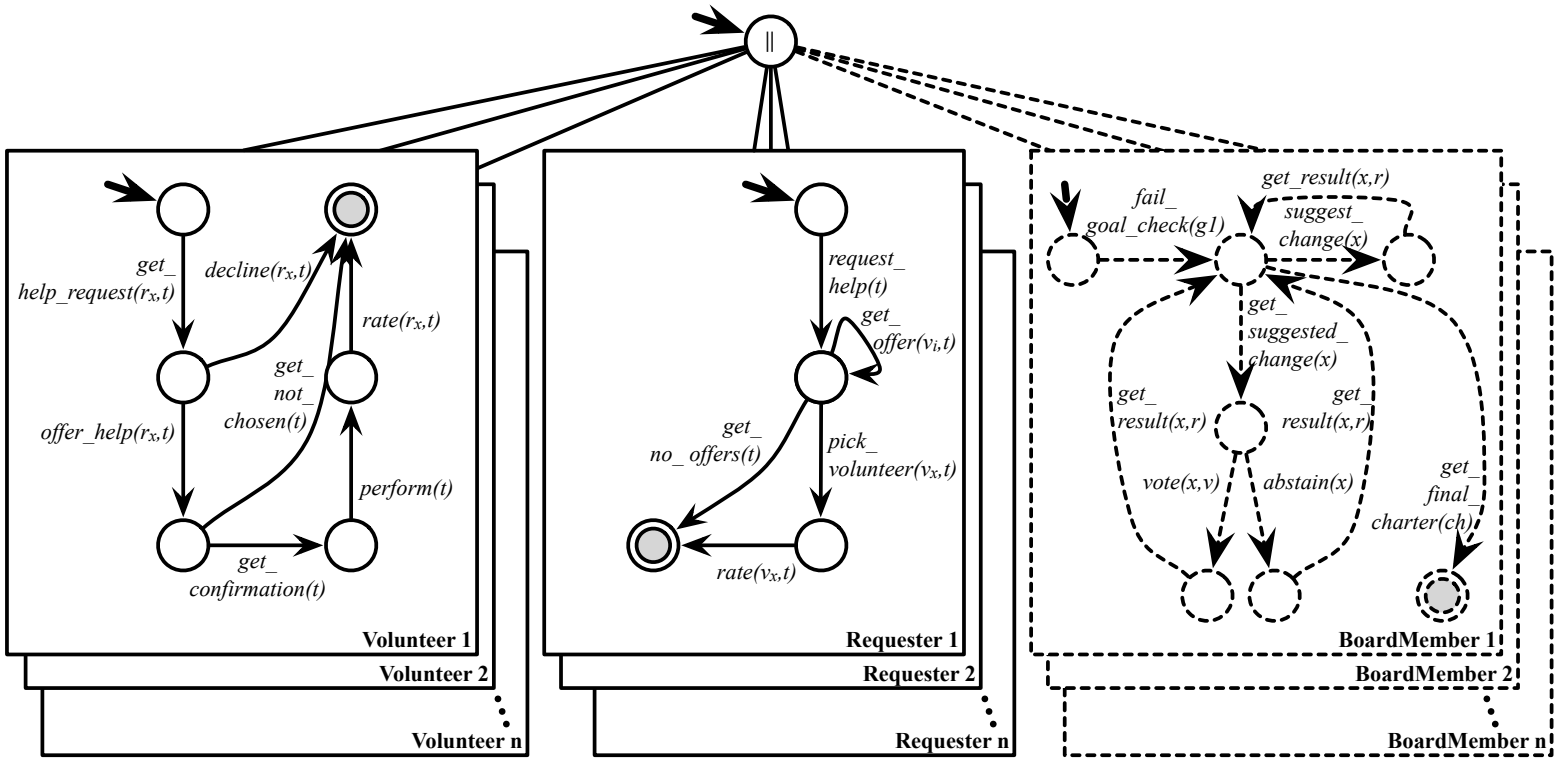
uHelp's Sample Interaction Model:



Self-Evolution Cycle



Self-Evolution Specification



Details of evolution to be specified by the charter's IP:

- When does evolution take place?
- How does the system trigger evolution?
- Which community members are allowed to suggest evolution?
- What is the minimum number of people required to discuss evolution?
- Who can suggest changes?
- How is evolution discussed and agreed upon?

Conclusion

Proposal

1. **Roadmap** for self-evolving IT communities
   - **Building blocks:** goals, norms, and interaction protocols
   - **Basic technologies:** goal satisfaction check, norm verification, norm enforcement, and interaction protocol execution
   - **Self-evolution cycle:** the interrelation between the building blocks and their technologies that helps drive self-evolution
2. **Basic approach** for each building block and its required technologies, helping build an initial prototype for self-evolving communities

Future Work

1. Consider research on emergence/self-organisation for driving evolution
2. Consider semantics as part of a charter

Thank you!