# Adaptive choreographies

Ivan Lanese
Computer Science Department
University of Bologna/INRIA
Italy

# Choreographies

- Allow to describe the behavior of a distributed communicating system at the very abstract level

- Based on interactions of the form $a \xrightarrow{o} b$

- Composed using different operators (sequence, parallel, ...)

- Possibly extended with other information (data, ...)

- Very similar to global types in multiparty session types

# Choreography projection

- Allow to automatically derive from a choreography a description of the behavior of each participant
- When interacting, the participant behave as specified by the choreography
  - Requires some conditions on the choreography
- The projection of $a \xrightarrow{o} b; b \xrightarrow{o'} c$ is $[\bar{o}]_a \ || \ [o; \bar{o'}]_b \ || \ [o']_c$
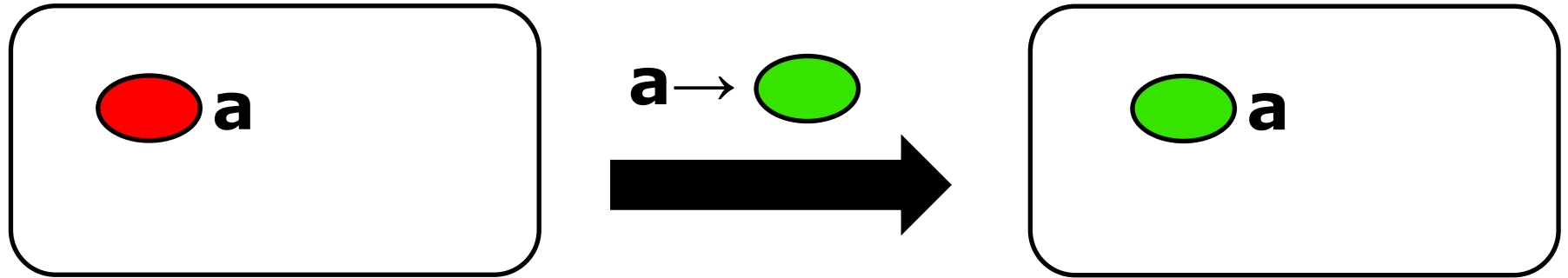
# Adaptation

- Systems should live for long periods of time in ever changing environments

- Users can change their minds

- The system should adapt to satisfy new requirements
  - Adaptation at runtime
  - Adaptation details not known when the system has been designed or even started

- Lots of works on adaptation

- Very little guarentees on what happens after adaptation
  - Which properties does the system satisfy?
  - Is it still safe?

# Our approach

- Extend choreographies with adaptation scopes
  - Part of the choreography that may be adapted
- Adaptation is specified by adaptation rules
  - Include the new code for the scope
  - Include information on when and where the rule can be applied
- The effect of the rules at choreography level is clear
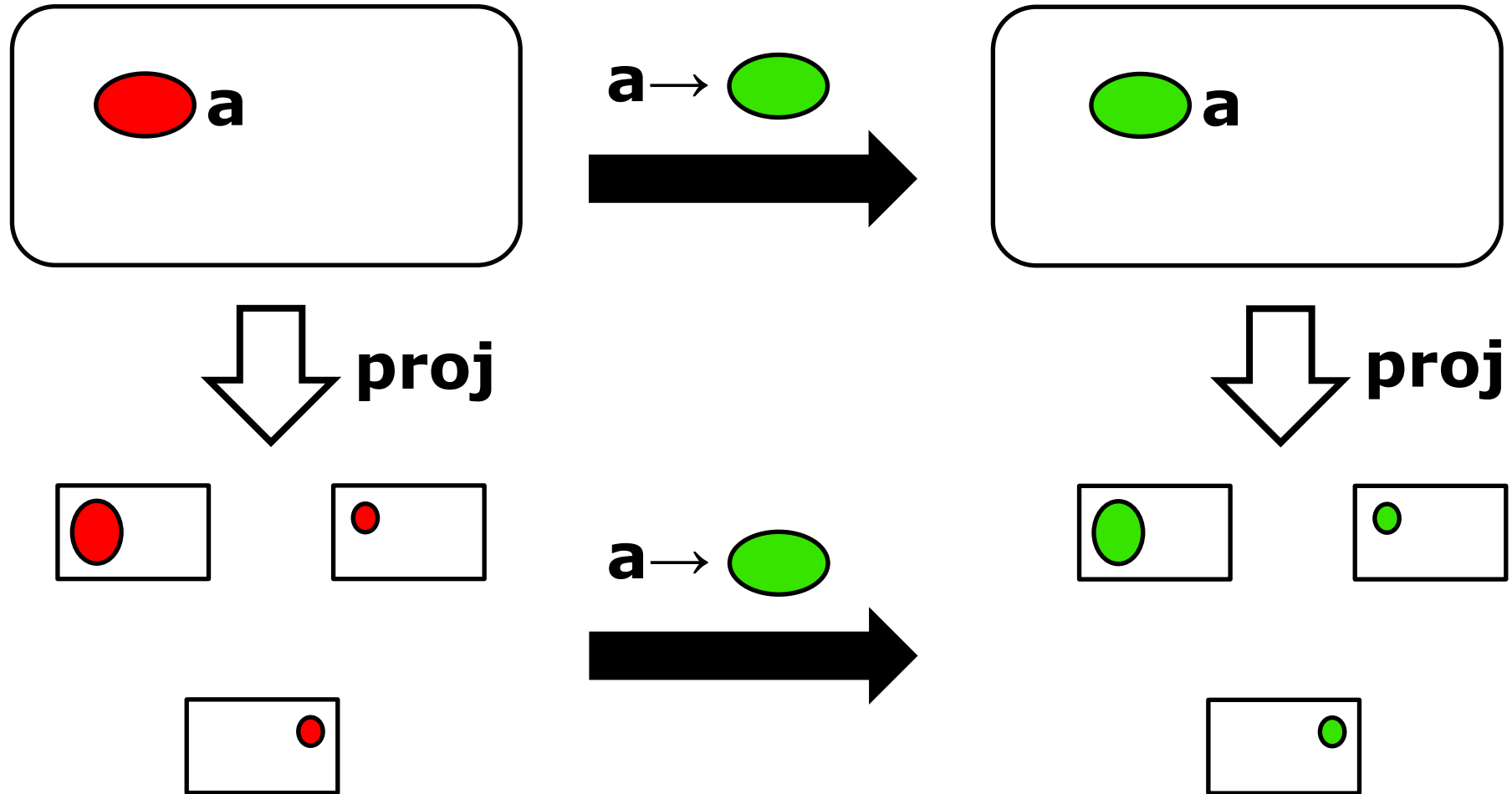- The projection of the rules is applied to the projected system

# Our approach, graphically

# Our approach, graphically
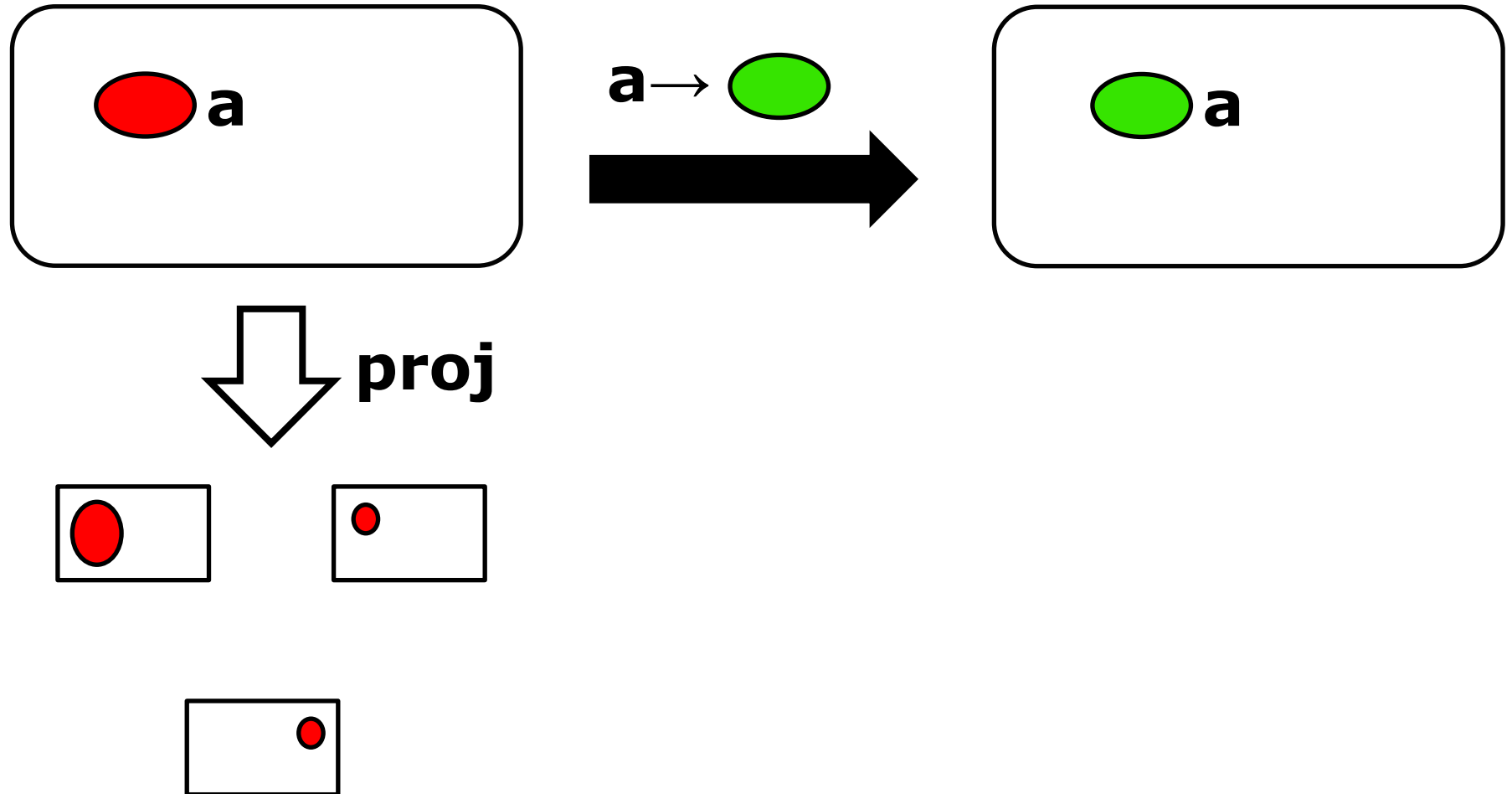
# Our approach, graphically

# Results

- We can adapt distributed communicating systems
- The adaptive system follows the behavior prescribed by the choreography under all possible adaptations
- The adapted system is deadlock free by construction

# Open issues

- Is it possible to restate/extend this approach using multiparty session types?

- What is needed to go towards an implementation?

  – More concrete language

  – Add data

- Can we specify other forms of adaptation?

# End of talk