The International RuleML Symposium on Rule Interchange and Applications

Orlando, Florida: October 30-31, 2008

Building an Autopoietic Knowledge Structure for Natural Language Conversational Agents

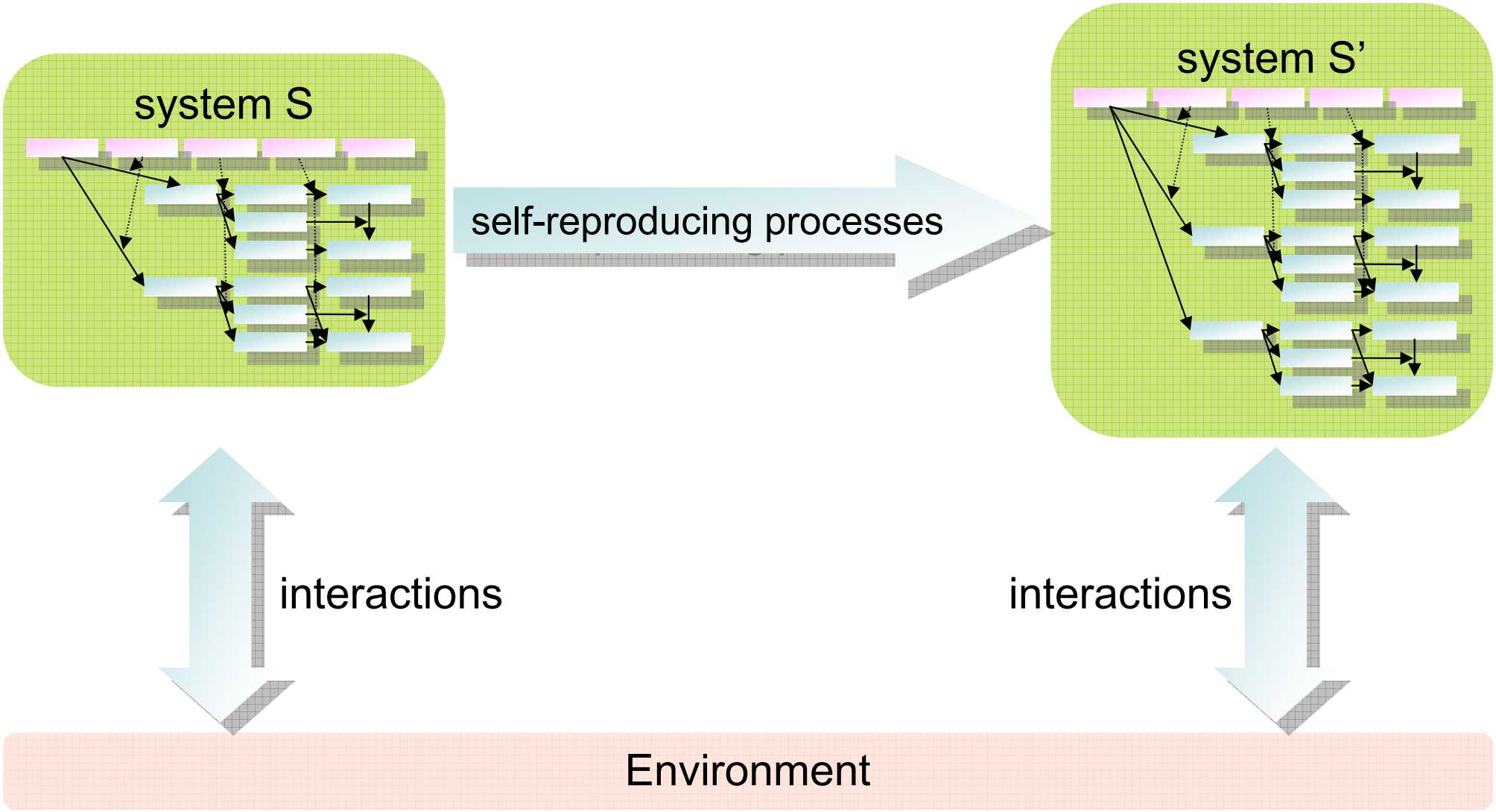
Yahoo! JAPAN Research

Kiyoshi Nitta

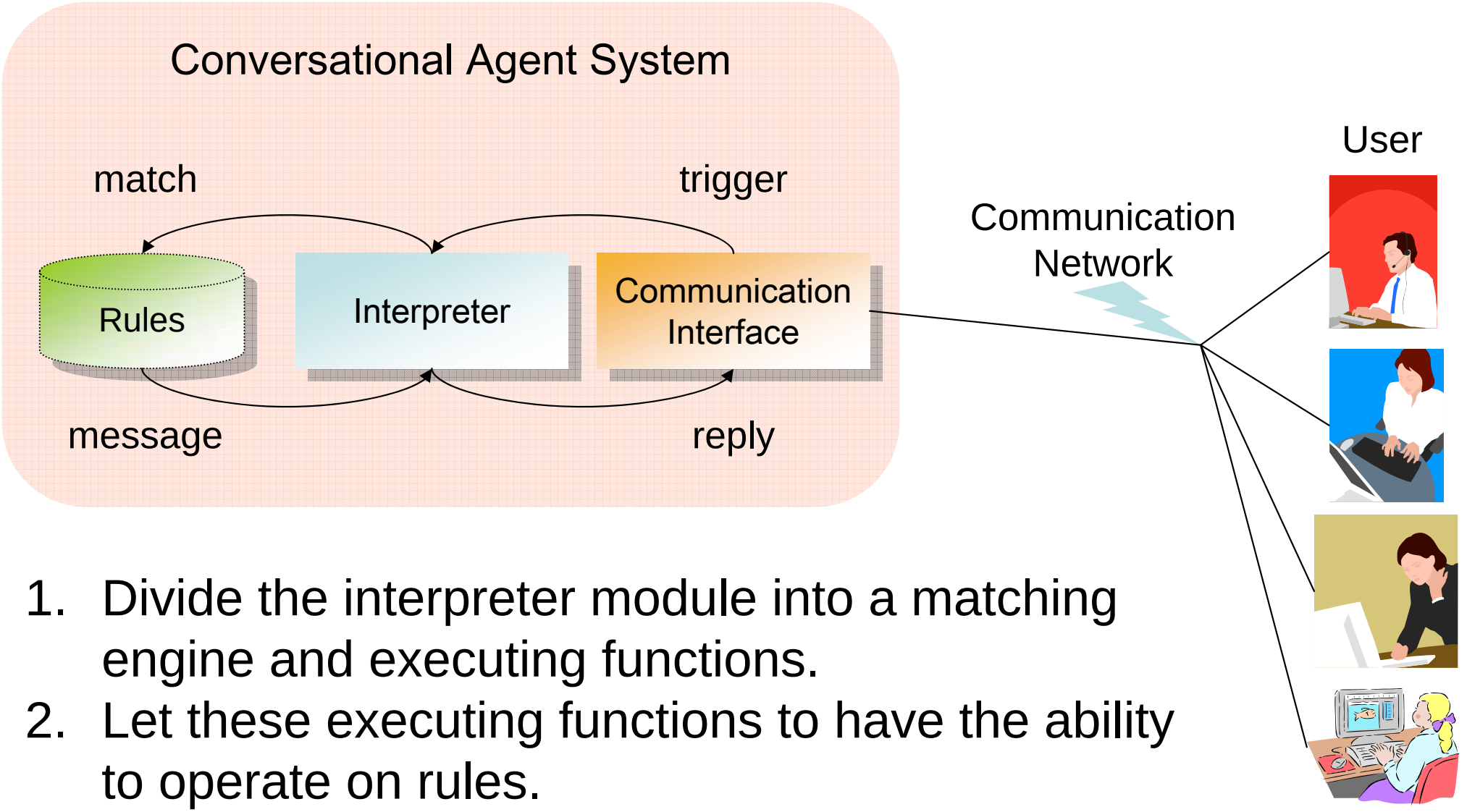
knitta@yahoo-corp.jp

(2008.10.28d)

Autopoietic Systems

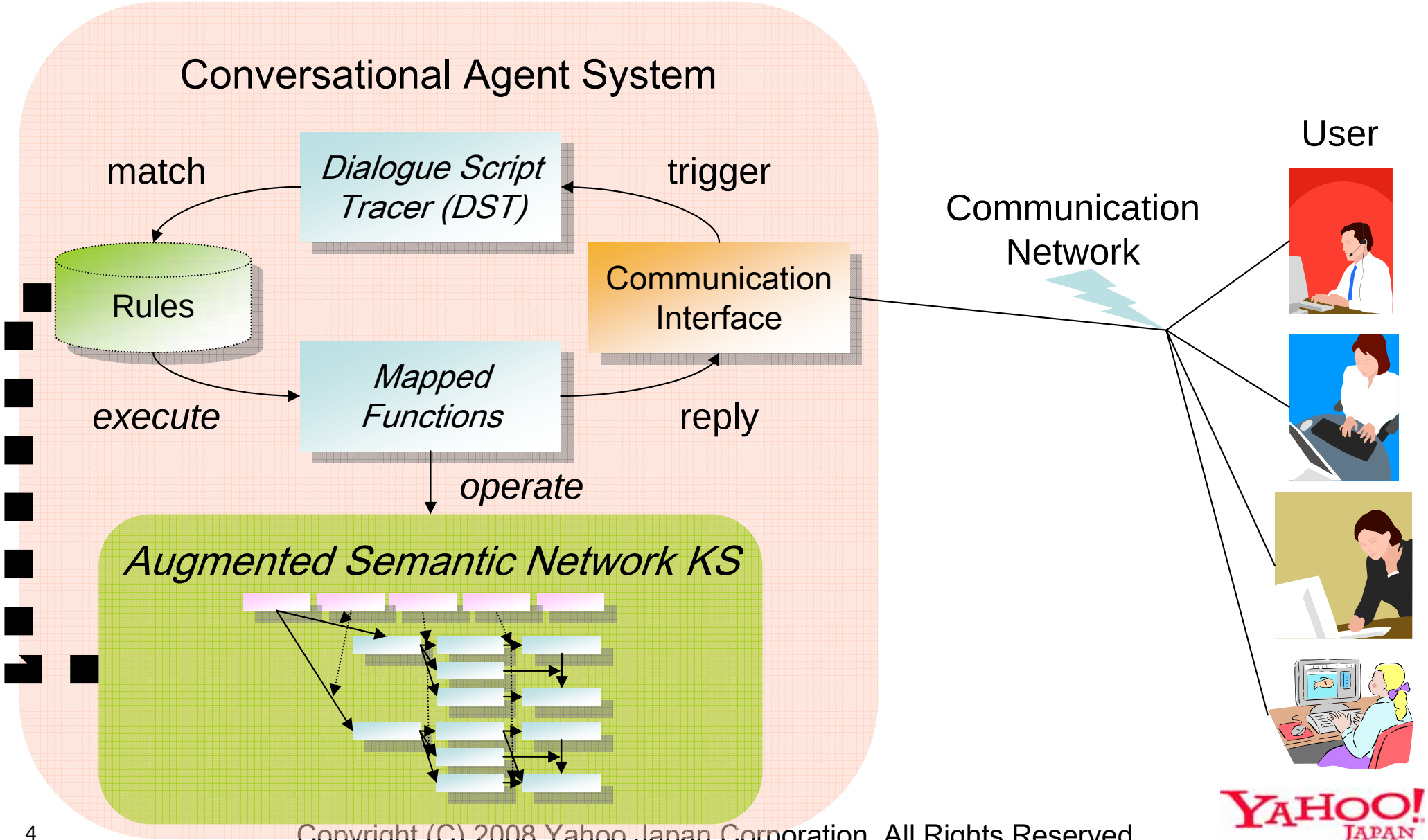


1st Step to be an Autopoietic System

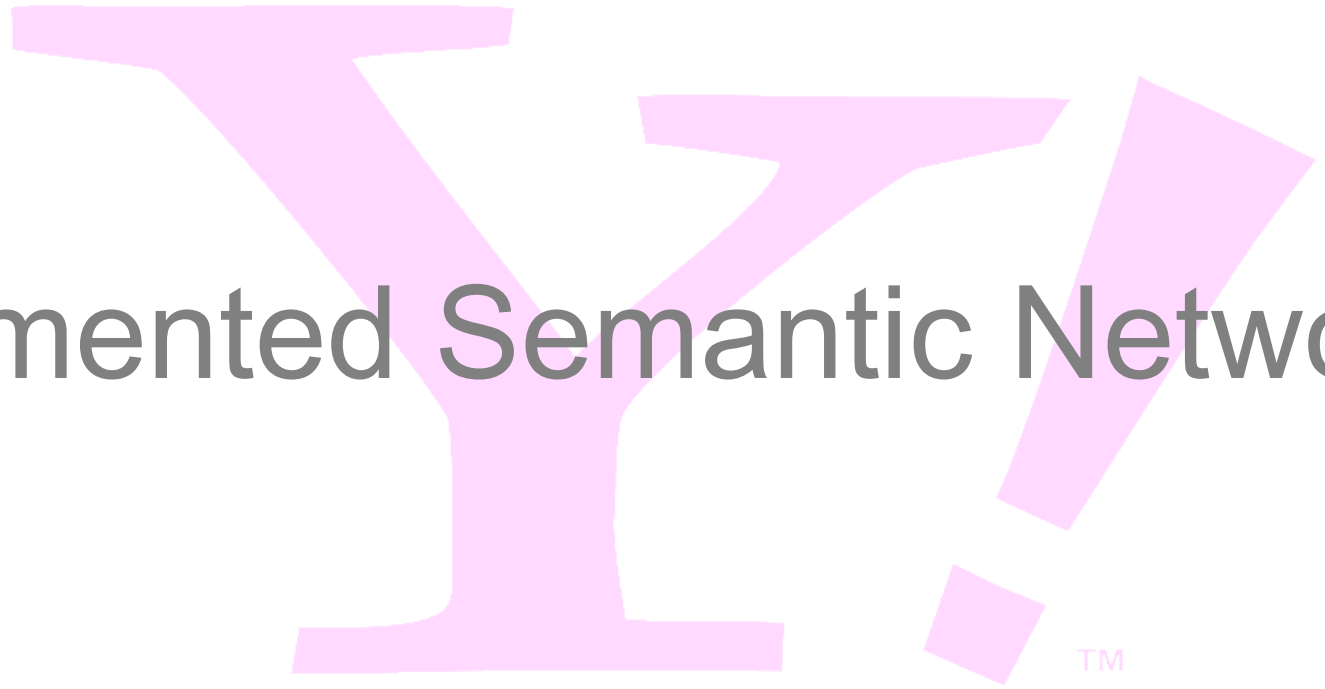


1. Divide the interpreter module into a matching engine and executing functions.
2. Let these executing functions to have the ability to operate on rules.

Our Rule-Based Conversational Agent



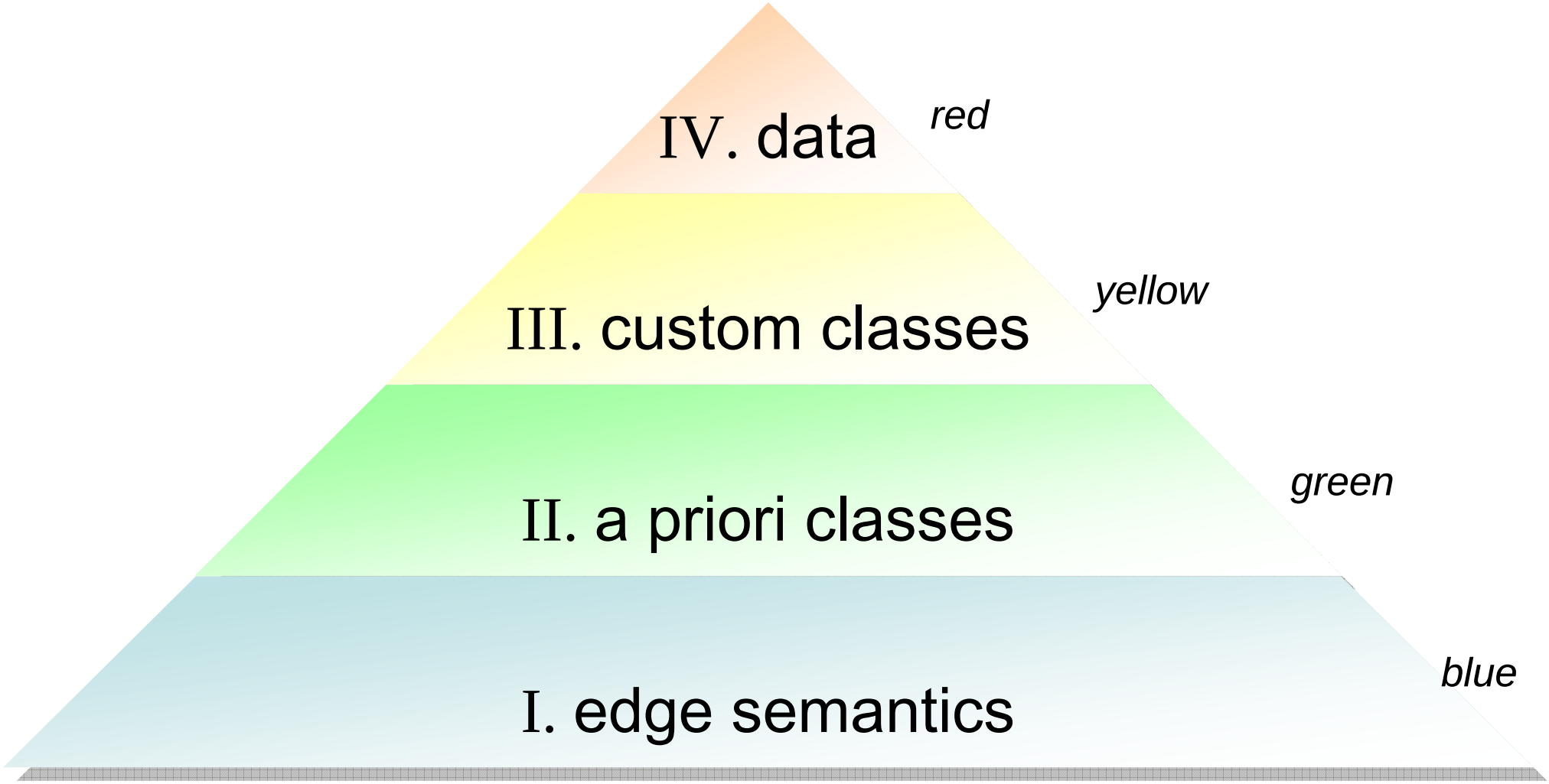
Augmented Semantic Network



Data Structure and Semantics

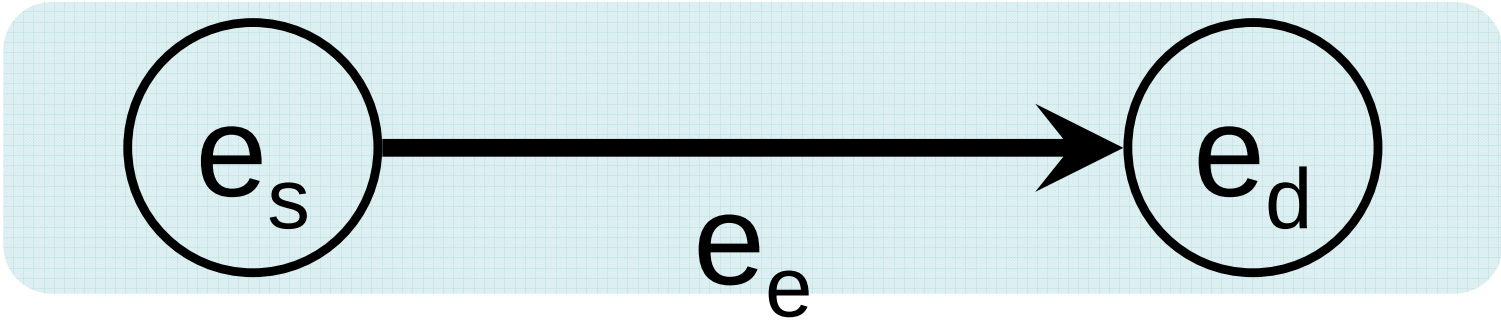
	Semantic Network (SN)	Augmented Semantic Network (ASN)
Data Structure	Directed Graph DG=(V, E) $V=\{v_1, v_2, \ldots\}$ $E=\{e_1(v_i, v_j), e_2(v_k, v_l), \ldots\}$	Augmented Directed Graph ADG=(E) $E=\{e_1(e_i, e_j), e_2(e_k, e_l), \ldots\}$
Semantics	Each semantic definition of V and E elements is defined independently.	Semantic definitions are aggregated to classes. (described below)

ASN Semantics



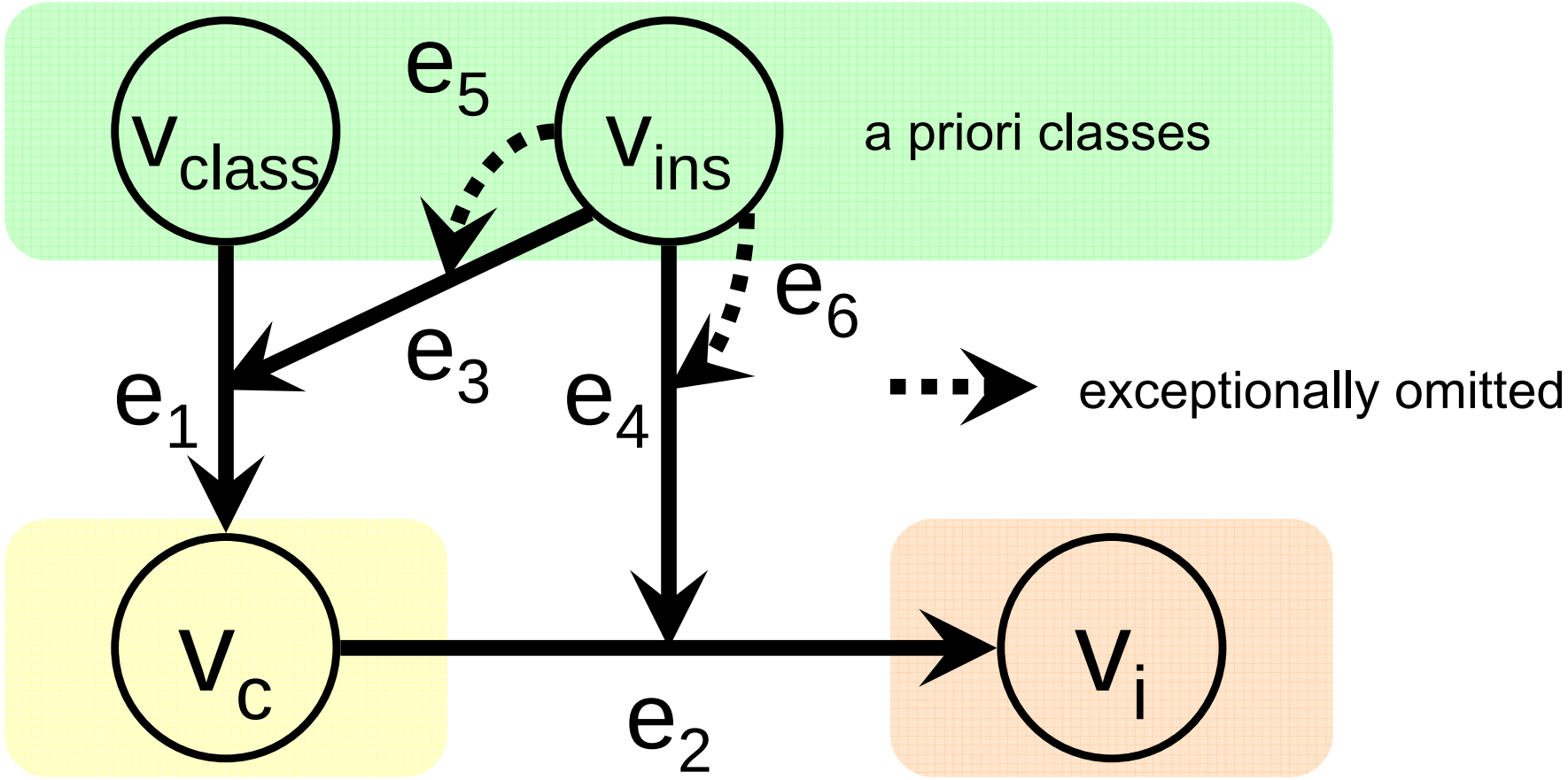
I. Edge Semantics

Edge e_e expresses the relationship from e_s to e_d.



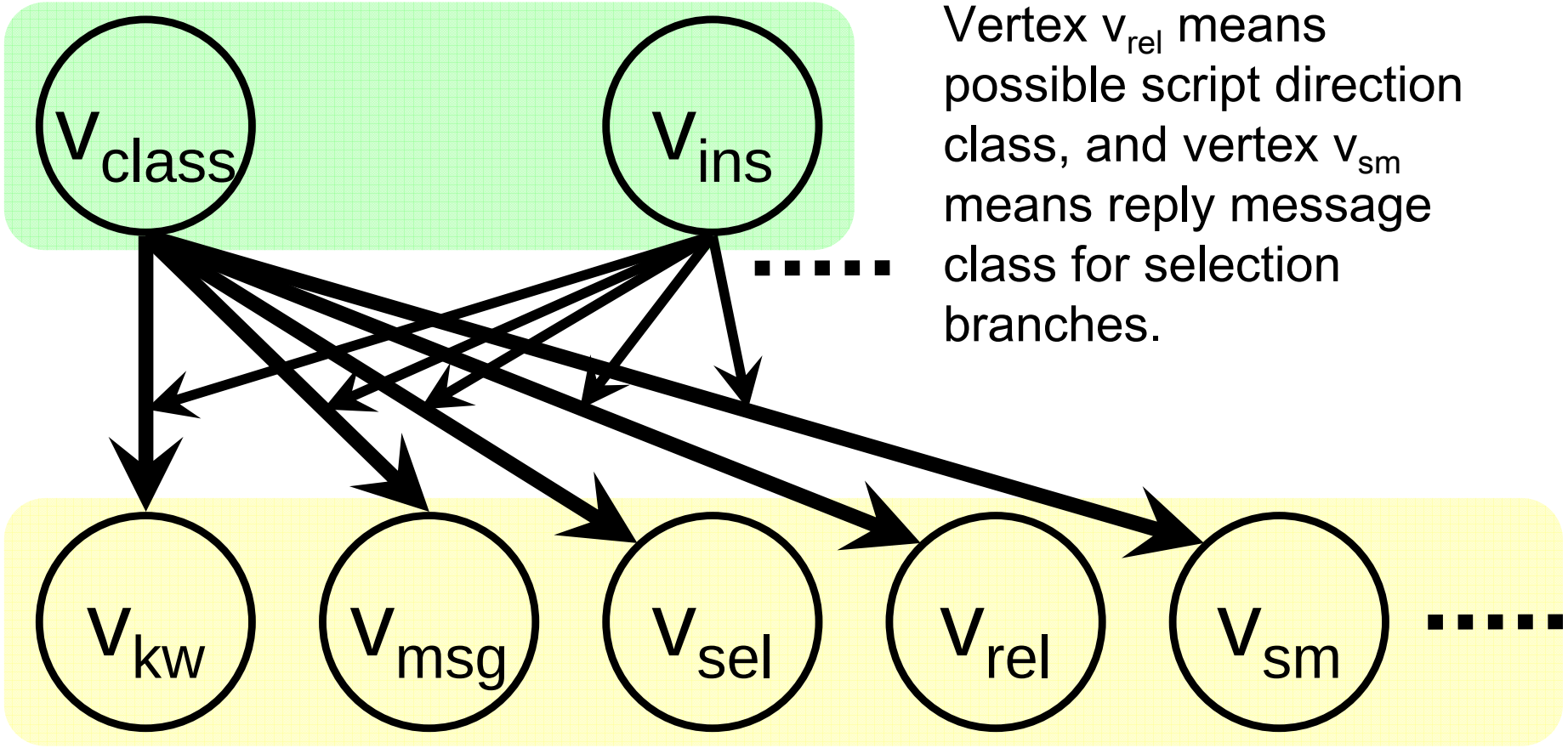
II. A priori Classes

Vertex v_{class} means that its instance vertices are classes, and vertex v_{ins} means that its instance edges are instance relations.



III. Custom Classes

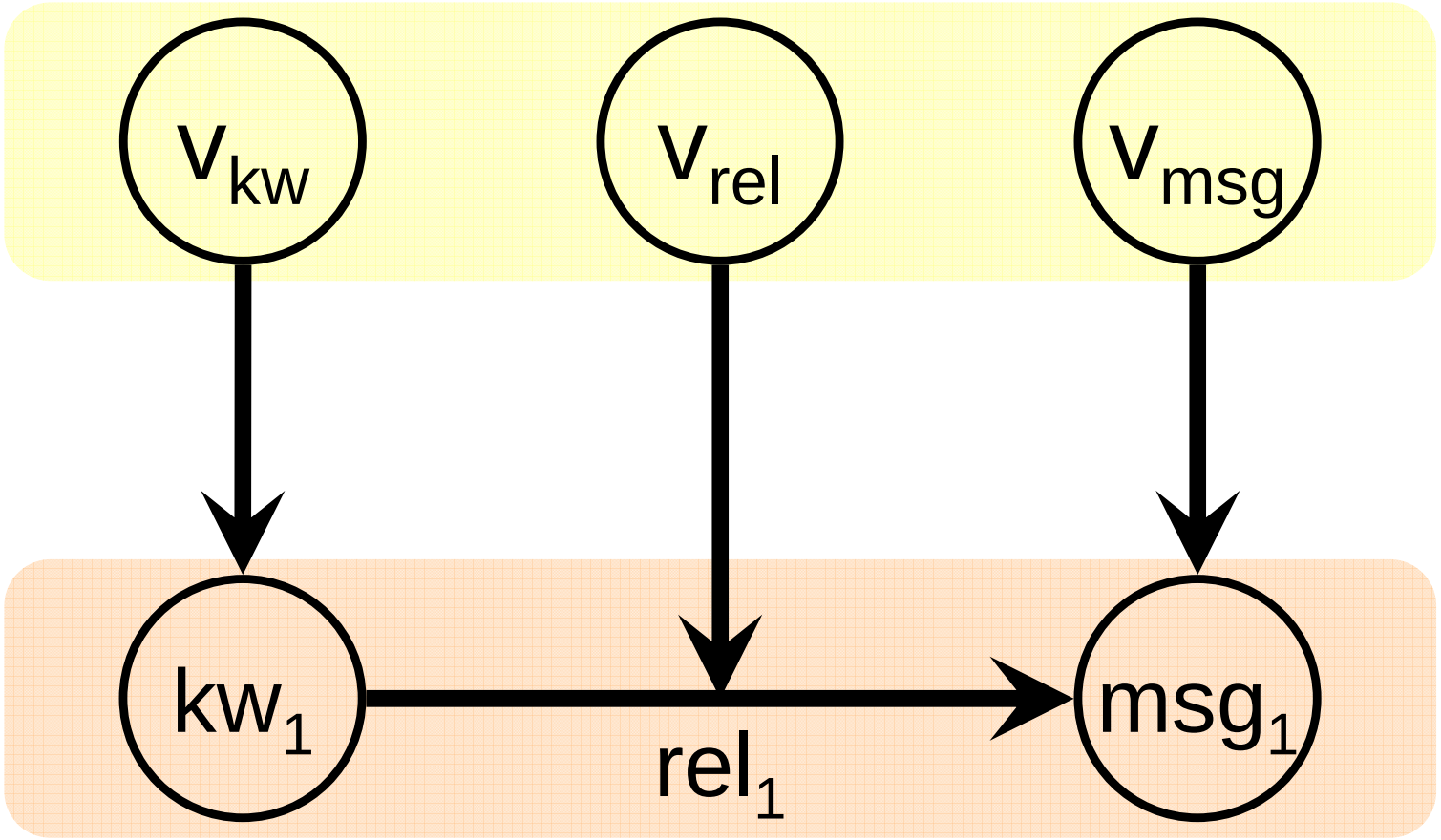
Vertices v_{kw}, v_{msg}, and v_{sel} mean trigger keyword, reply message, and selection branch classes, respectively.



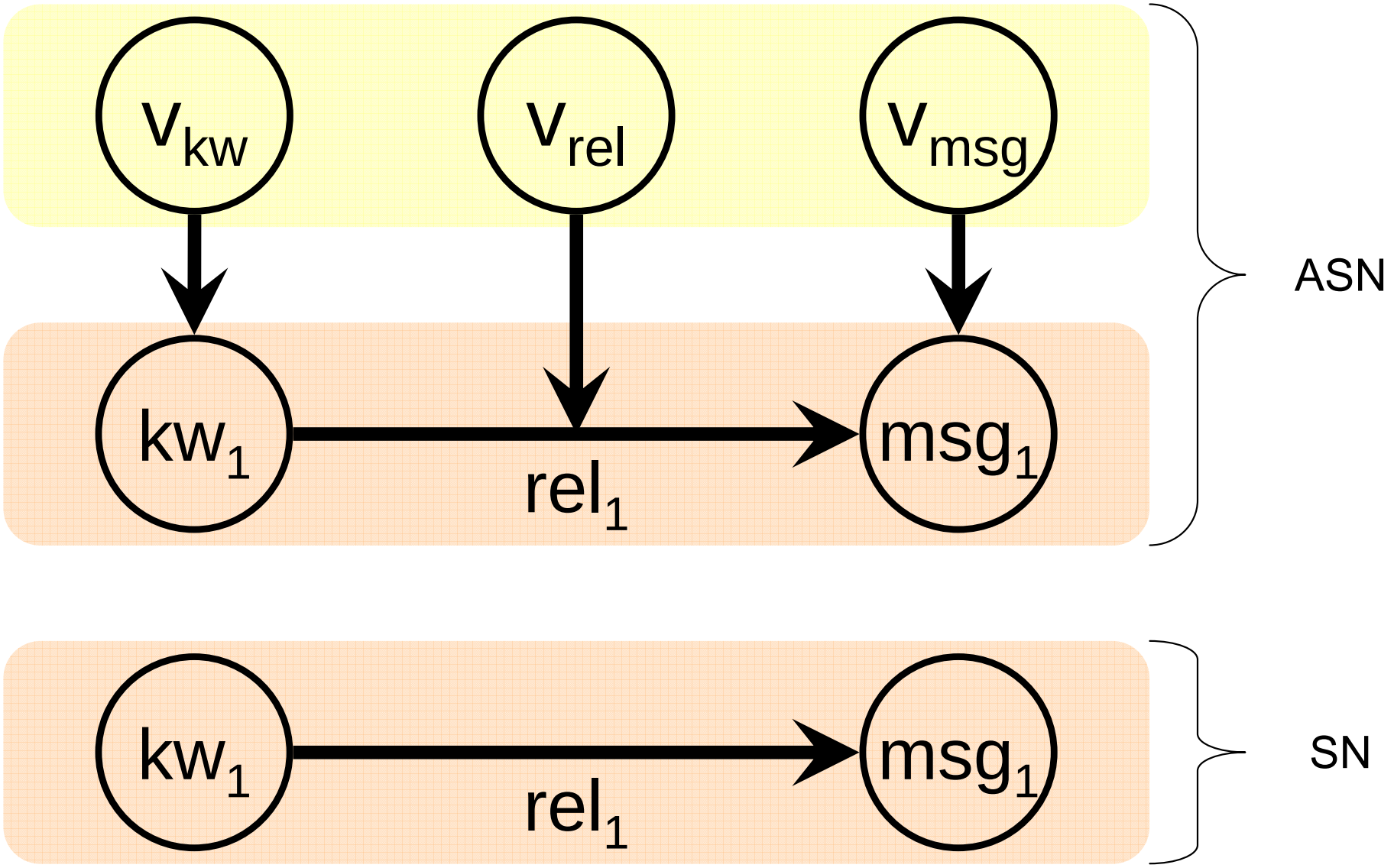
Vertex v_{rel} means possible script direction class, and vertex v_{sm} means reply message class for selection branches.

IV. Data

Every data element has at least one edge that connects from a semantically defined class vertex to the element.



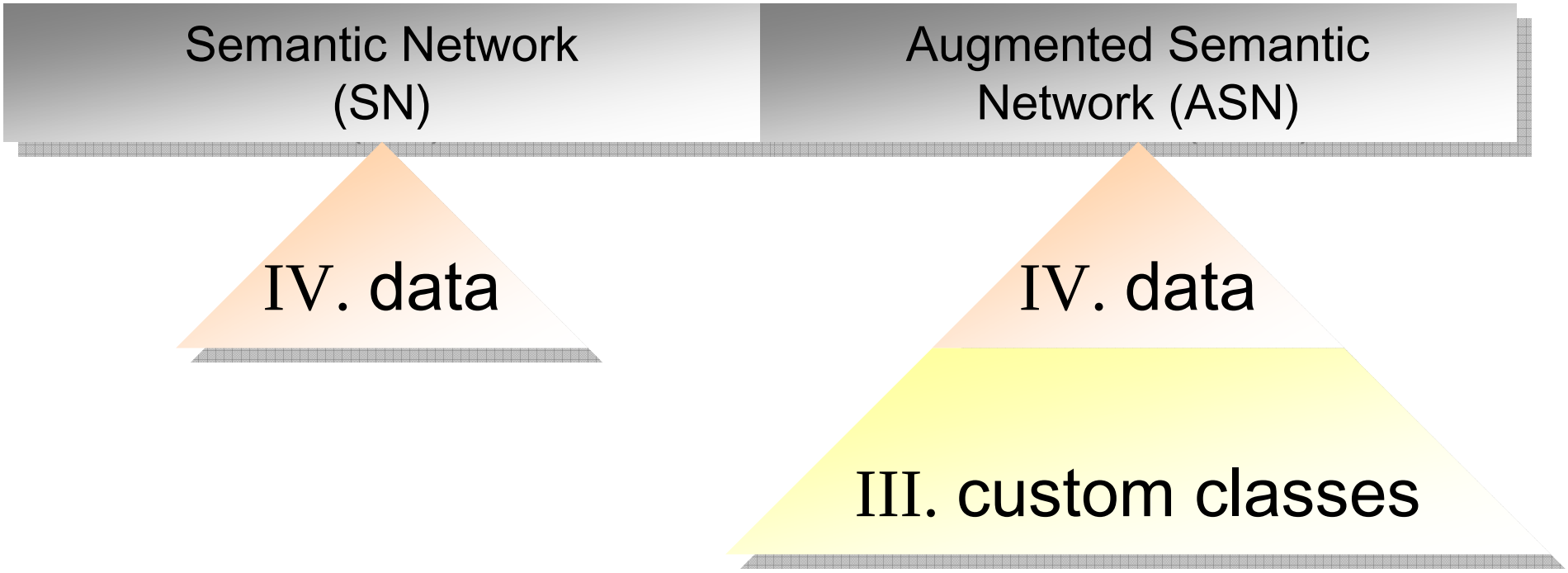
Difference between SN and ASN



Autopoietic Behavior

Ontology Modification

Self-reproducing processes of autopoietic systems require ontology modifications.



The ontology is fixed.

The ontology can be modified.

Achievement and Future Tasks

- current progress:
  - Implemented base classes for dialogue scripts that modify the *data*. (programming language reflection)
  - Trying to build enhanced classes for dialogue scripts that also modify the *custom classes*.
- open questions:
  - Can the class set finite?
  - Even if it is finite, how should we evaluate whether the system is autopoietic or not?

Thank you!