Shlaer-Mellor OOA/RD with UML

Stephen J. Mellor
Ian Wilkie

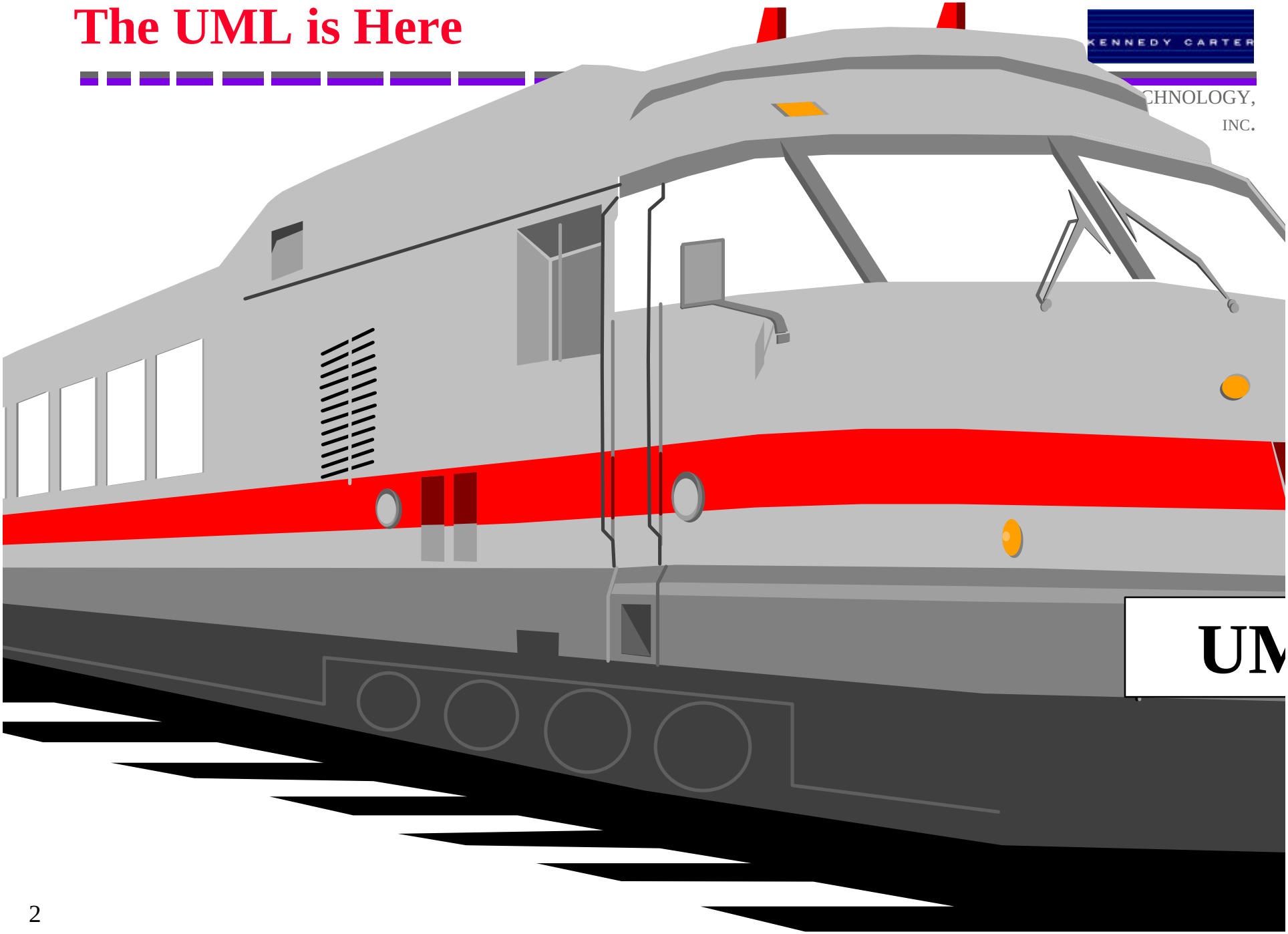




The UML is Here

KENNEDY CARTER

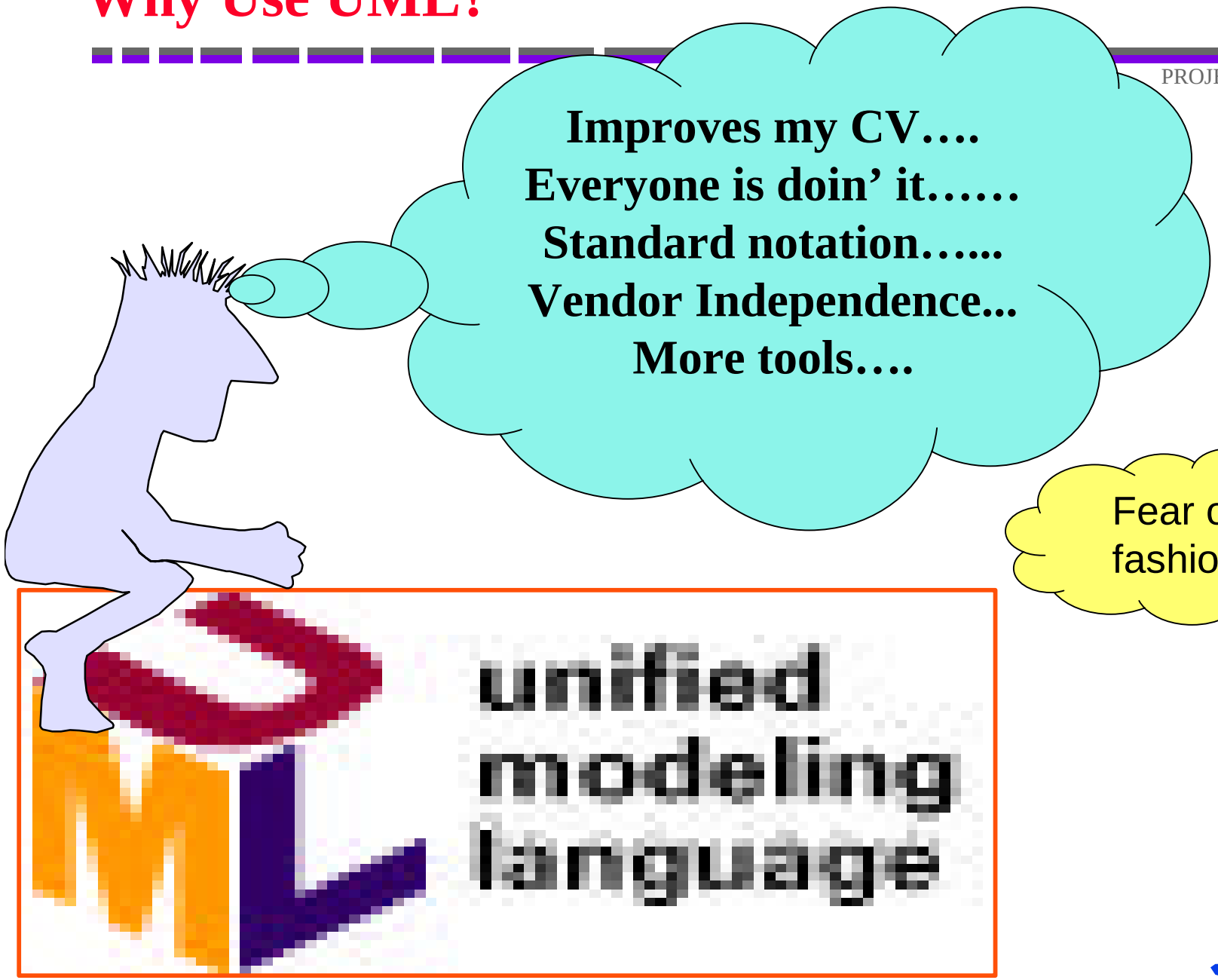
CHNOLOGY,
INC.



Why Use UML?

KENNEDY CARTER

PROJECT TECHNOLOGY, INC.

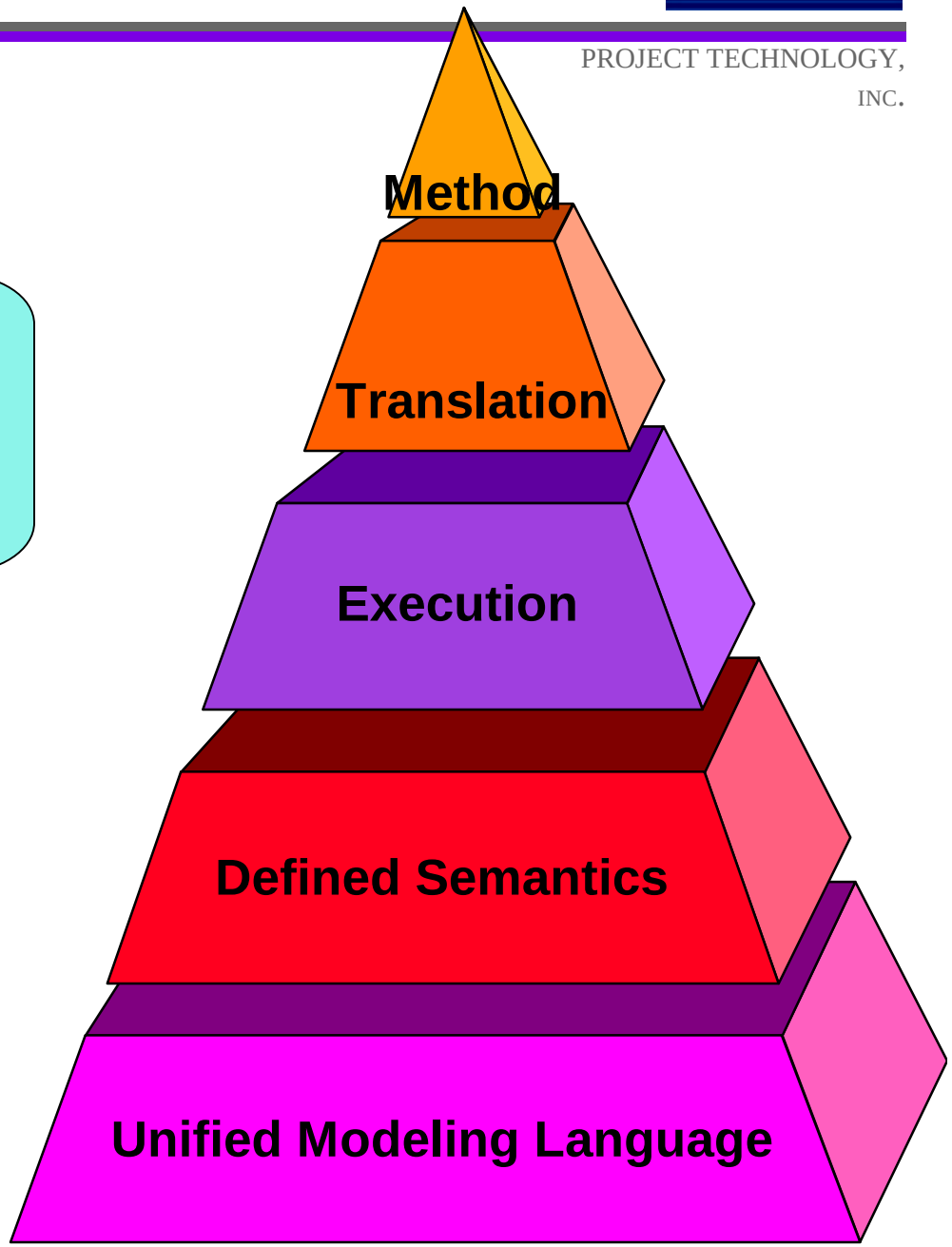


What Else is Required?

KENNEDY CARTER

PROJECT TECHNOLOGY, INC.



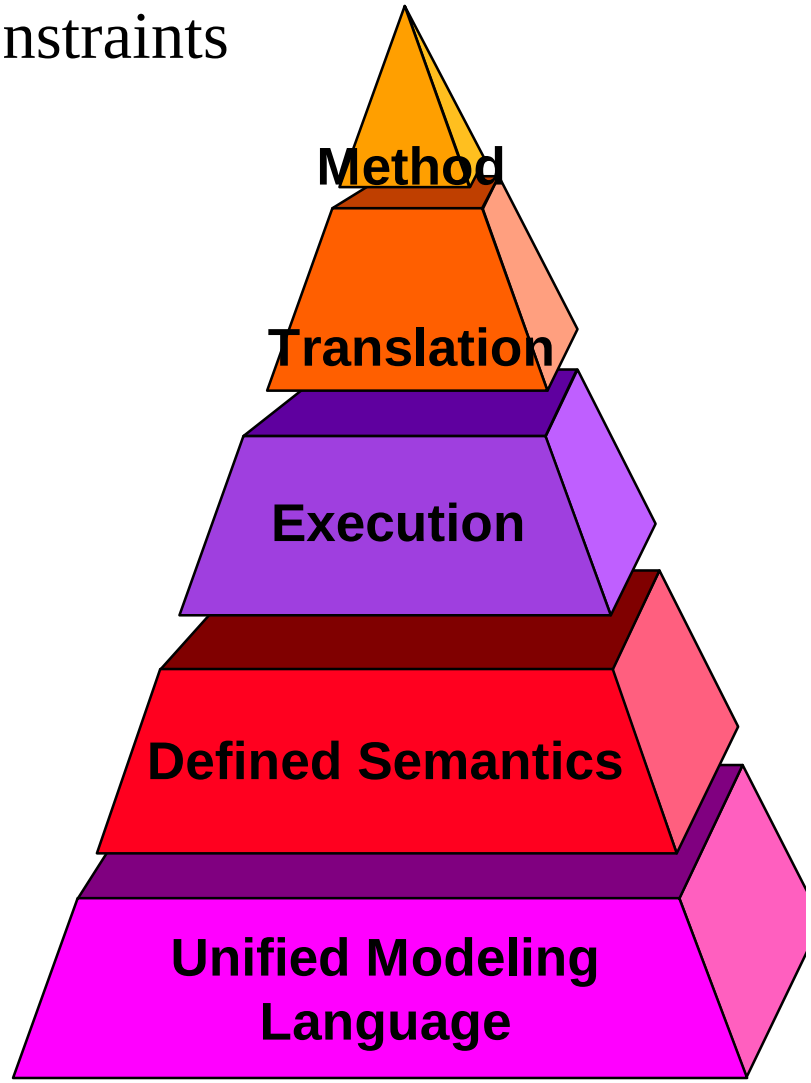


Maintain Key Features

What are the key features required to support the method?

- Defined semantics, including constraints
- Execution Model
- Translation

Do we really need:
Identifiers?
Referential attributes?
No operations?
No hierachy in state models?
Limited process types?

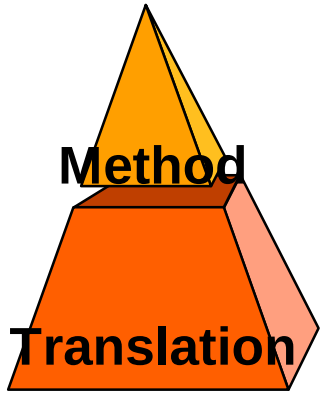


Capture Translation Information

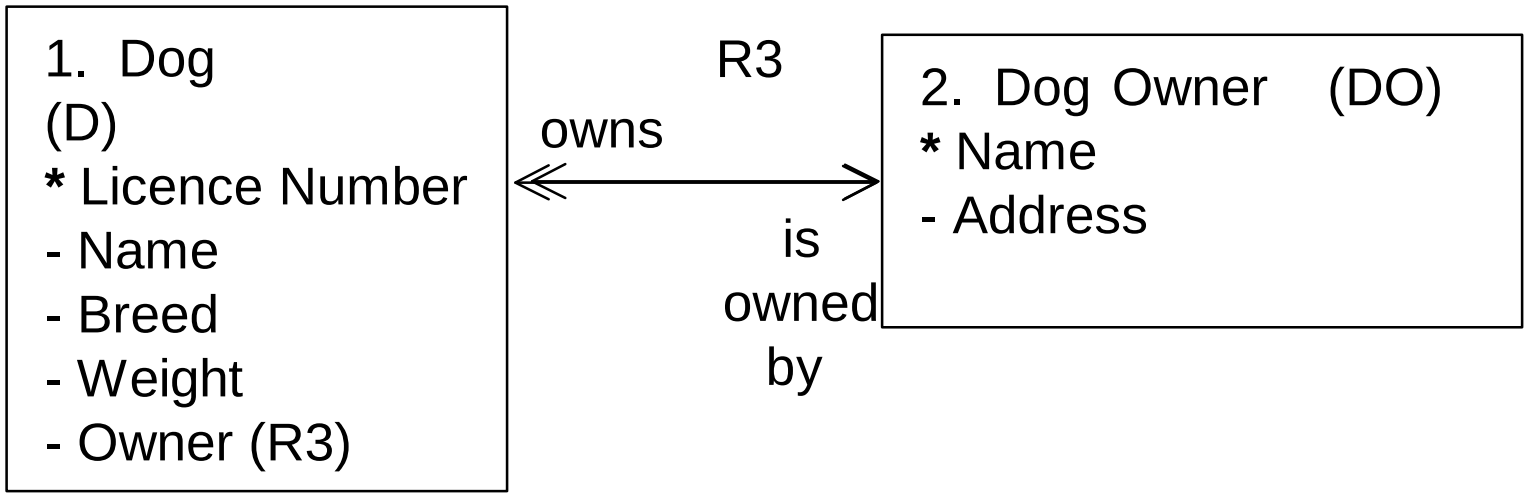
But the repository needs to have all the information required for translation.

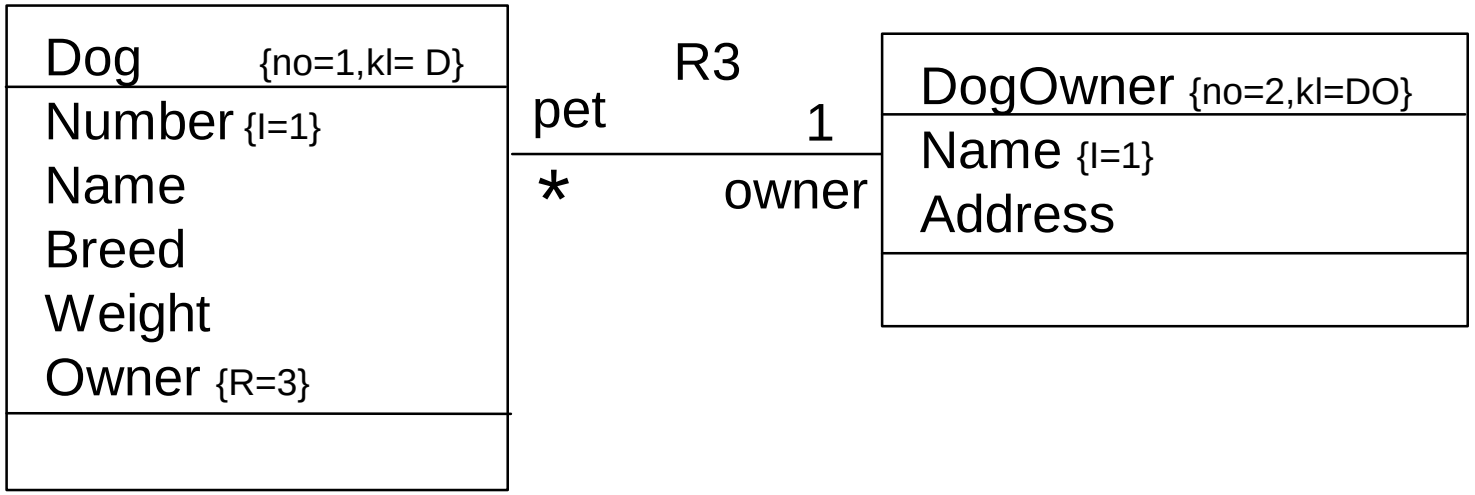
Do we really need:

Identifiers?	**What *could* be used as an identifier?**
Referential attributes?	**Which instance? (i.e. a link)**
No operations?	**Derived or fundamental?**
No hierachy in state models?	**Ease of construction, but….**
Limited process types?	**Reusable processes**

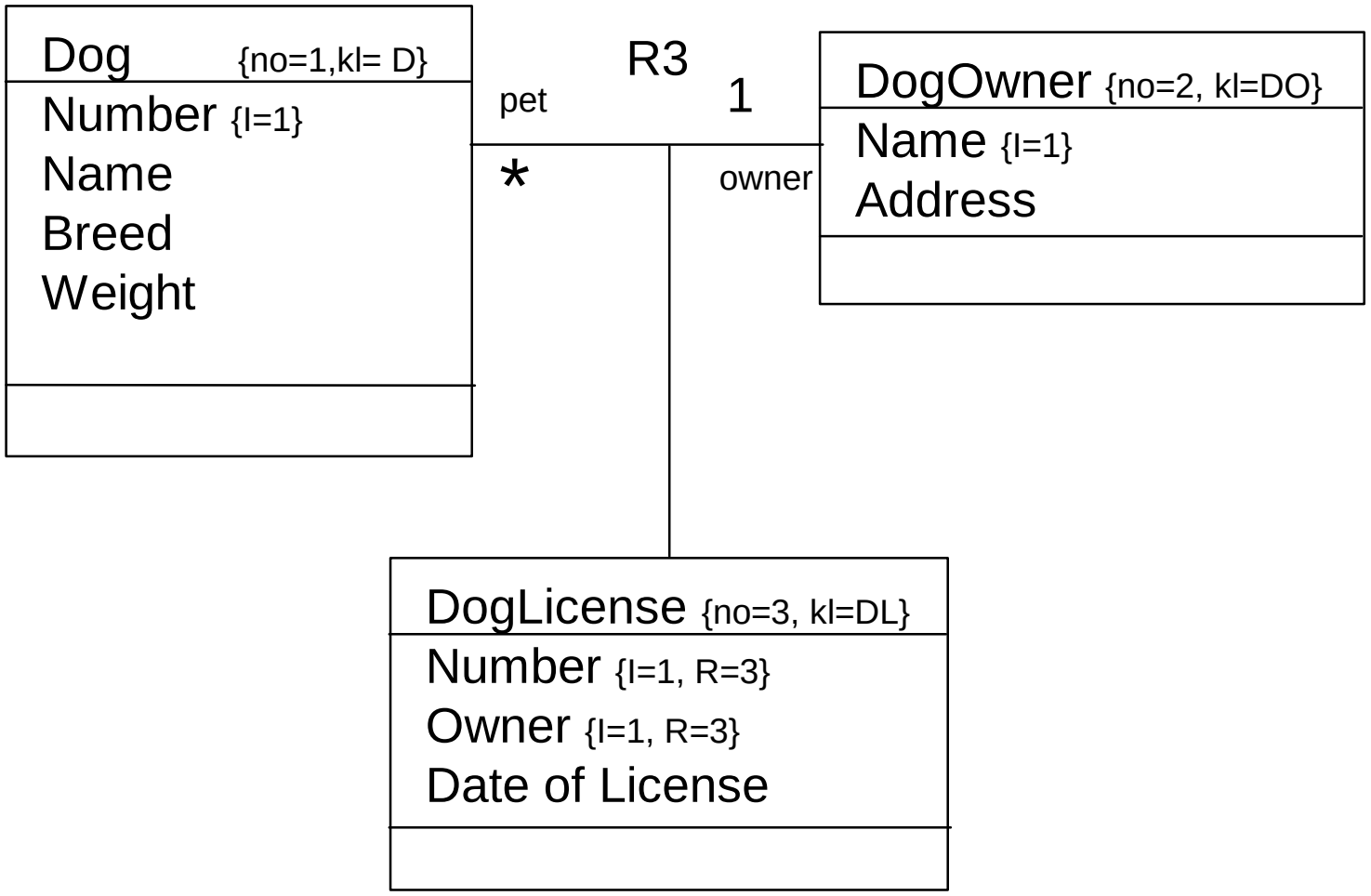


Classes


1. Dog
(D)
* Licence Number
- Name
- Breed
- Weight
- Owner (R3)
R3
owns
is
owned
by
2. Dog Owner (DO)
* Name
- Address


Dog {no=1,kl= D}
Number {I=1}
Name
Breed
Weight
Owner {R=3}
R3
pet
*
1
owner
DogOwner {no=2,kl=DO}
Name {I=1}
Address

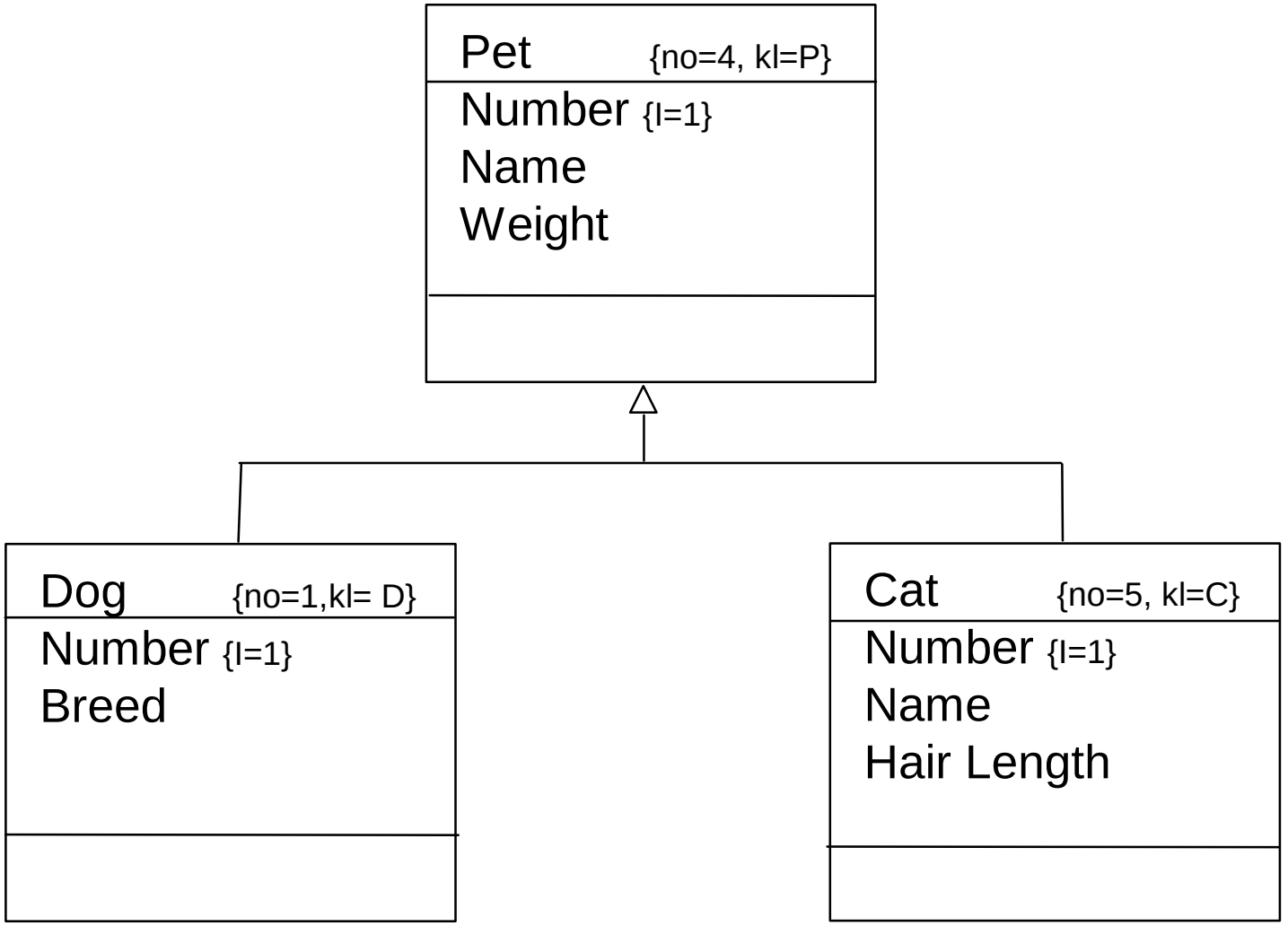
Association Classes



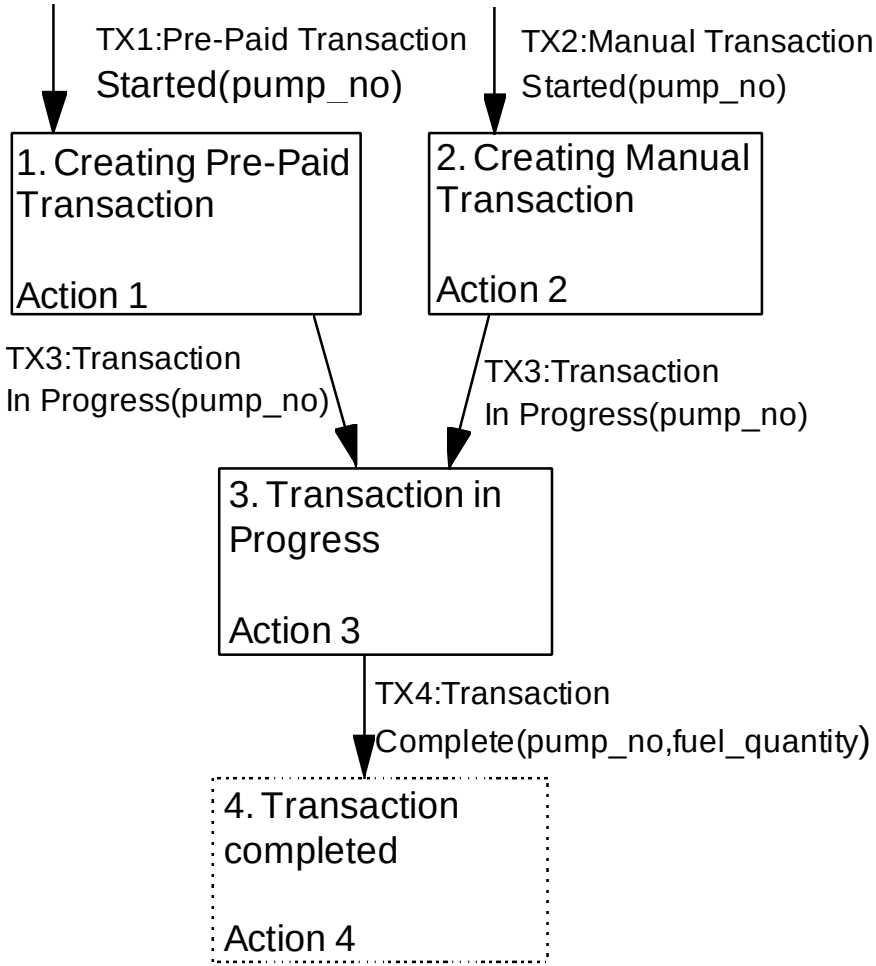
Generalization

KENNEDY CARTER

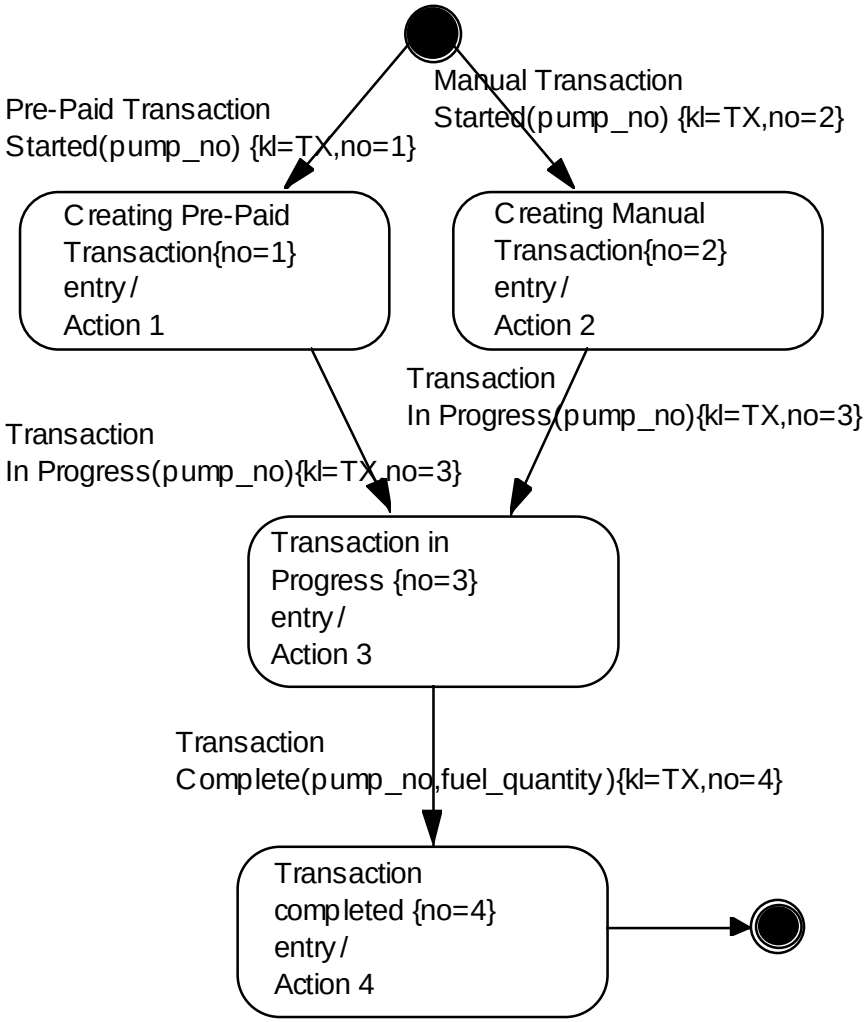
PROJECT TECHNOLOGY, INC.



State Models



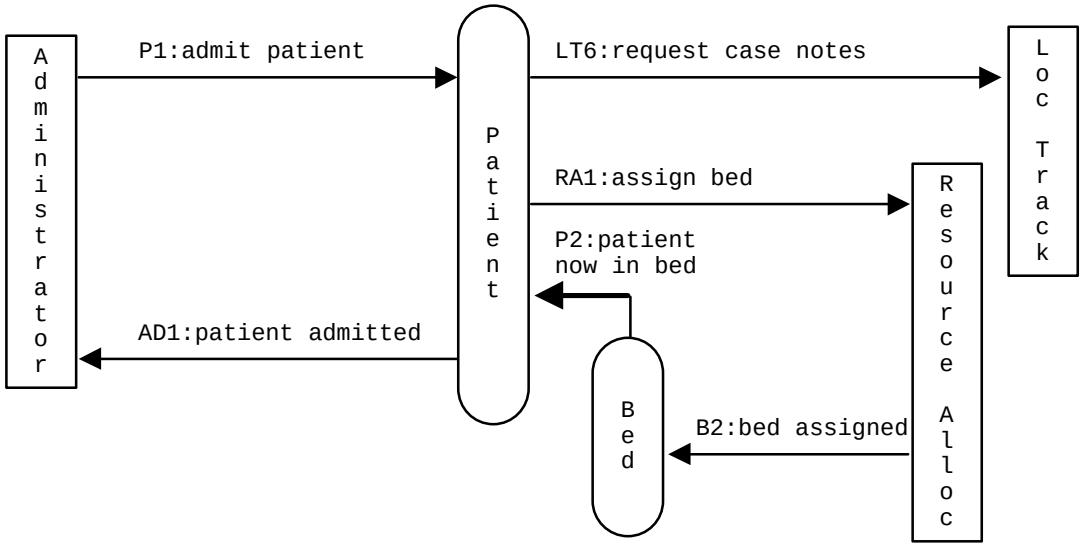
State Models



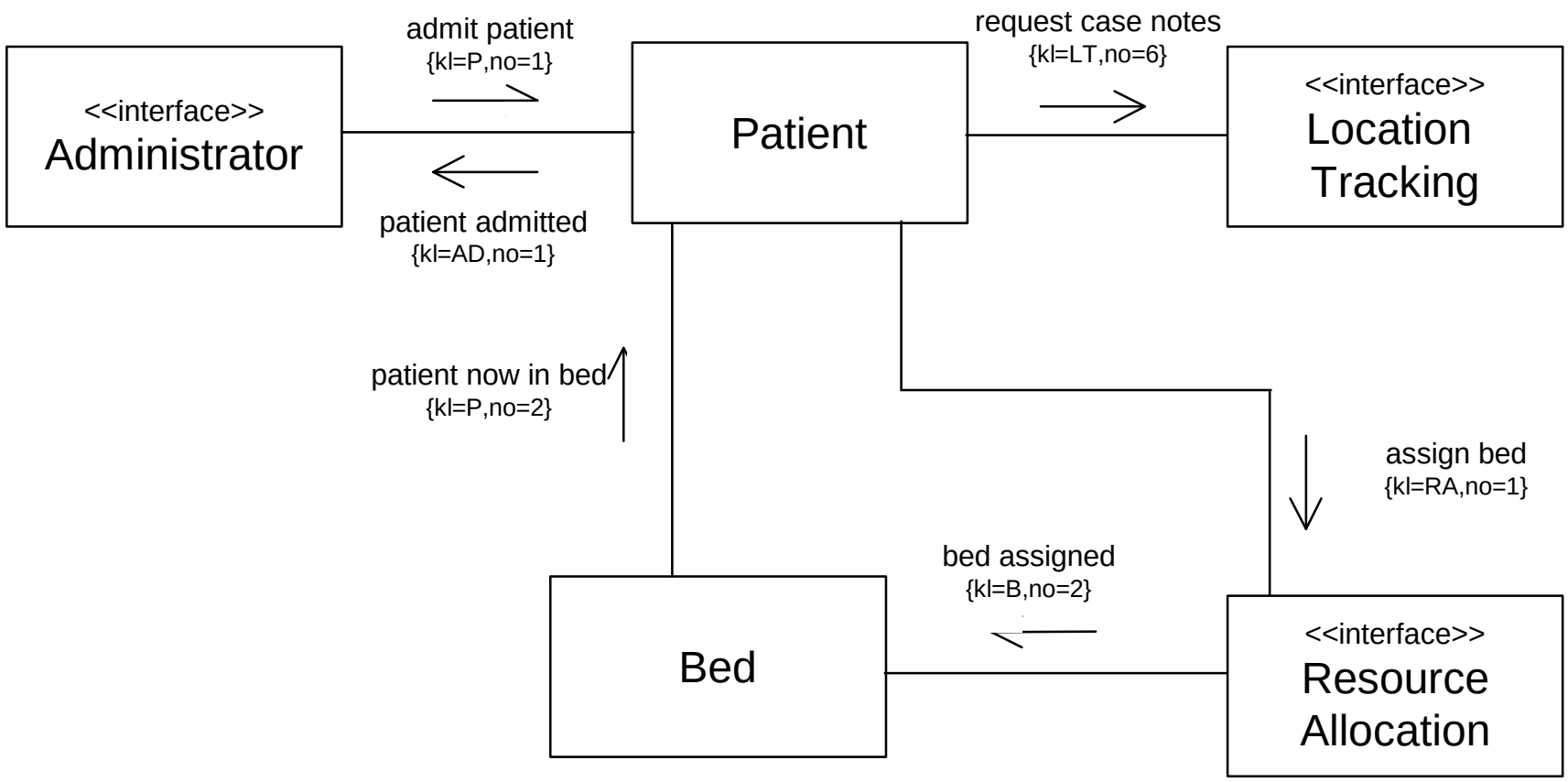
Collaboration Diagrams

KENNEDY CARTER

PROJECT TECHNOLOGY, INC.

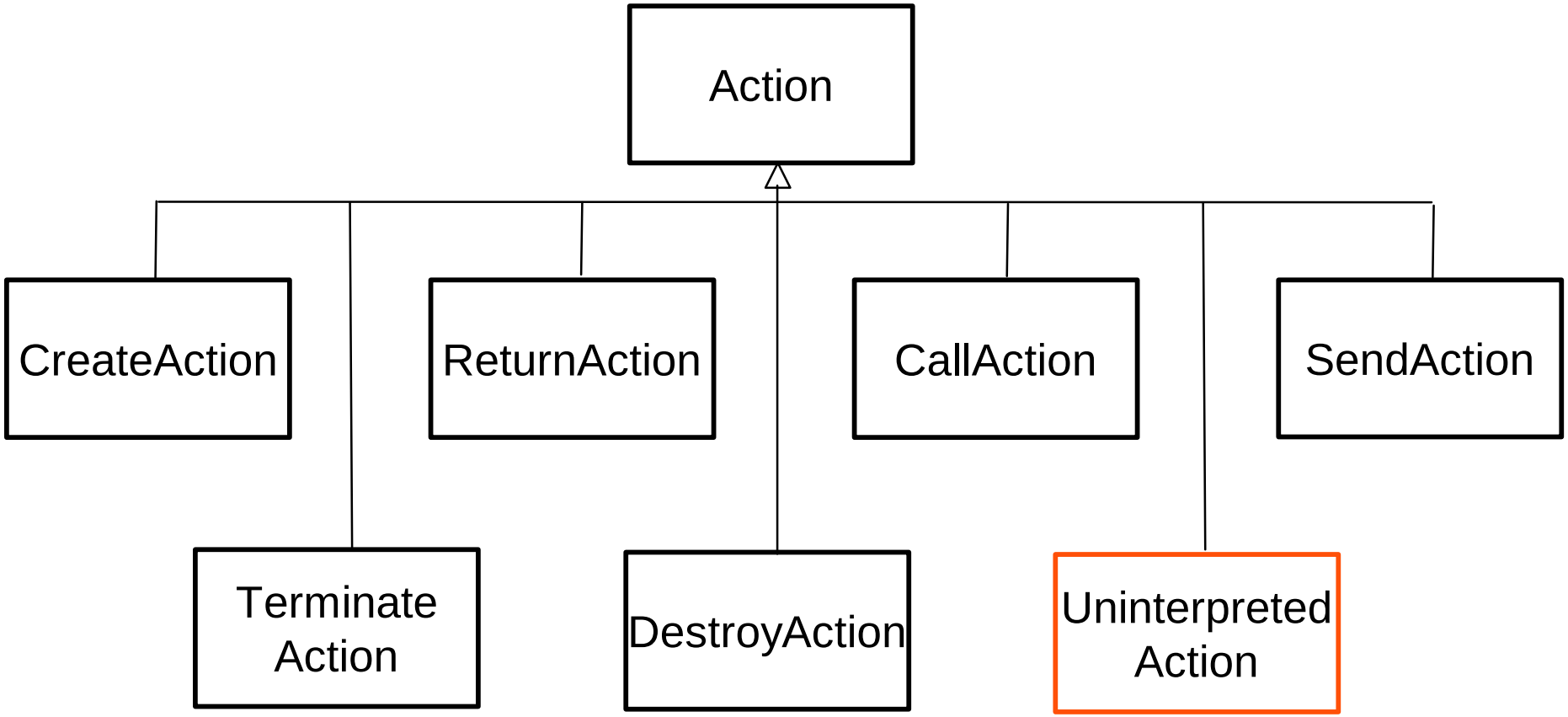


Collaboration Diagrams



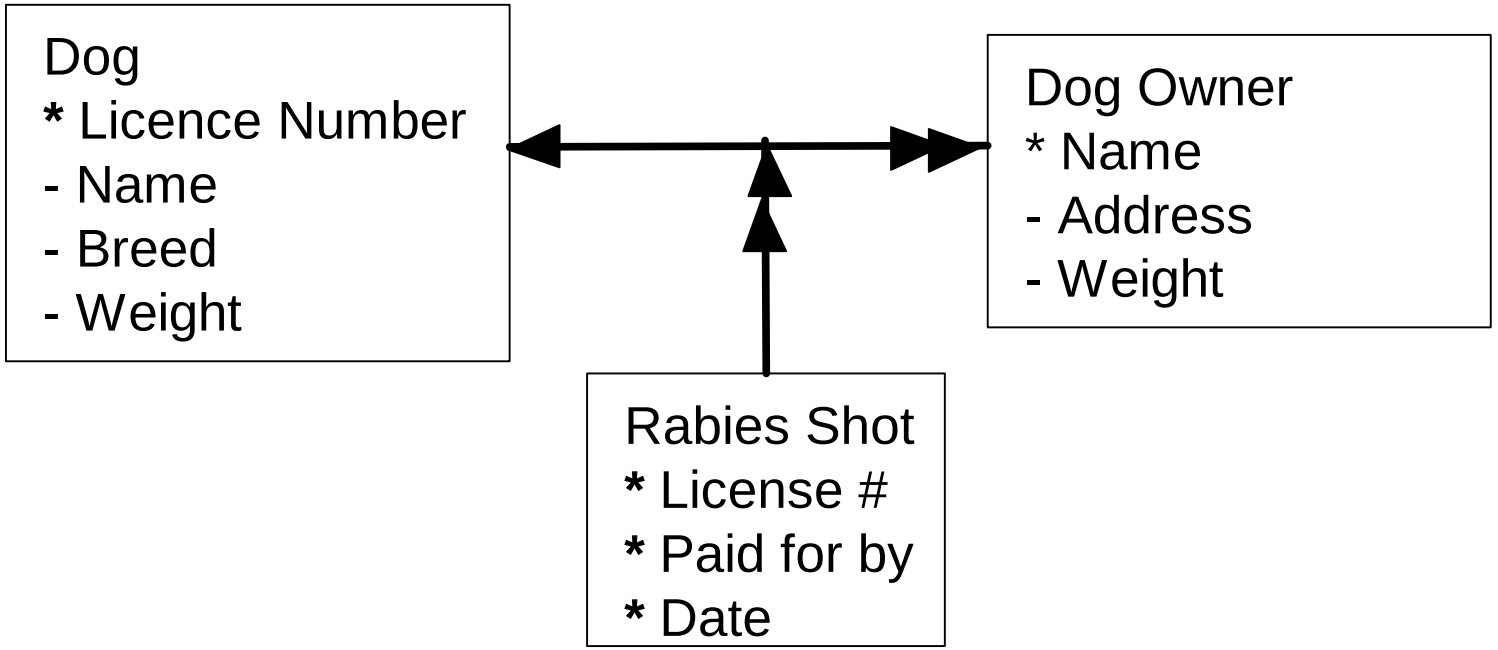
Actions

The UML has a limited view of actions.



Mismatches

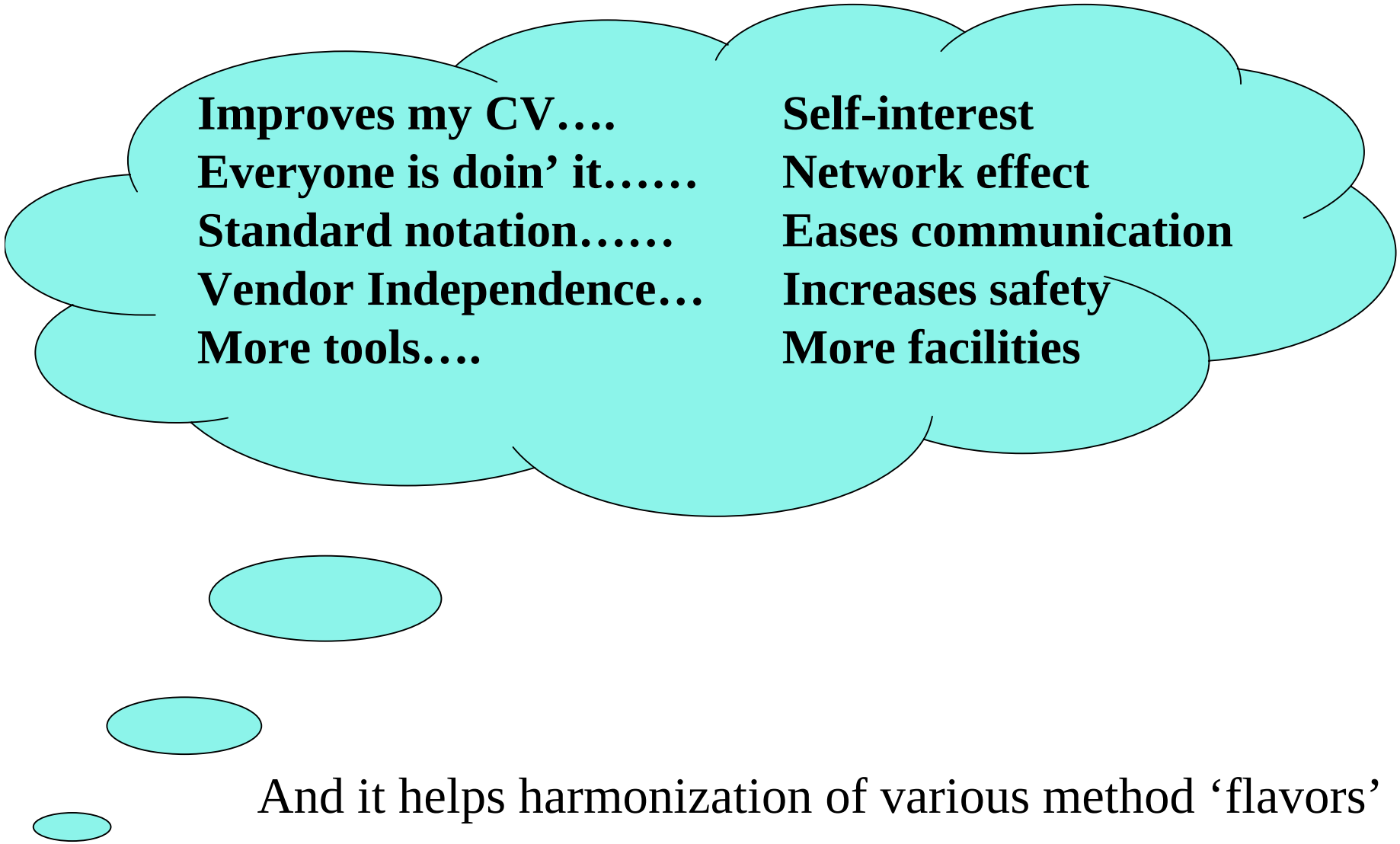
(1) M-(x:x)



(2) Generalisation is not necessarily inheritance in UML

- ... Super/subtypes can be supported

Aims of the Shlaer-Mellor in UML project



And it helps harmonization of various method 'flavors'

Deliverables of the Project

- Agreement between PT and KC on both content and representation in UML
- A “how to” guide that concentrates on notation
- A detailed mapping to the UML meta-model
  - Repository Exchange
  - Executability in UML tools
  - Proofs in UML tools
  - ... moving towards an executable UML

The Future: Action Semantics



PROJECT TECHNOLOGY, INC.

The Action Semantics Consortium now comprises:

- Project Technology, Inc.
- Kennedy Carter
- ObjectSwitch
- T-VEC
- Software Productivity Consortium
- CS VERILOG
- Rational

The Future: Execution Profiles

Two views of profiles:

- we need a separate mechanism to define
  - interpretations (semantic variations)
  - extensions

Alternatively,

- we don't need a separate mechanism
  - simply subset the UML and
  - use existing extension mechanisms