software studio

object models: relations

Daniel Jackson

relationship

kinds of relation

› property
› containment
› association
› naming

Shape →(color)→ Color

Directory →(contains)→ FSObject

Course →(enrolled)→ Student

Machine →(ip)→ IPAddress

User (friends) — recursive
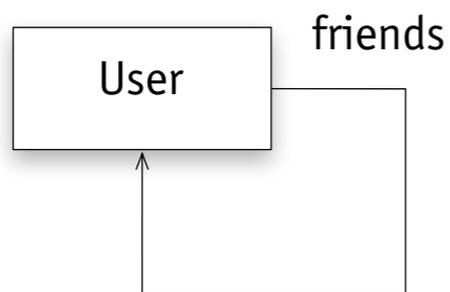
a 'homogeneous' or
'recursive' relation
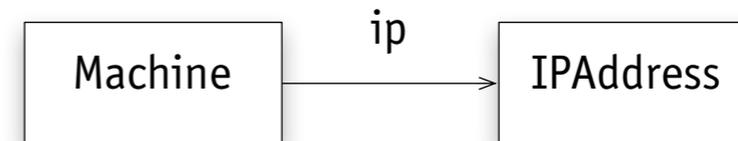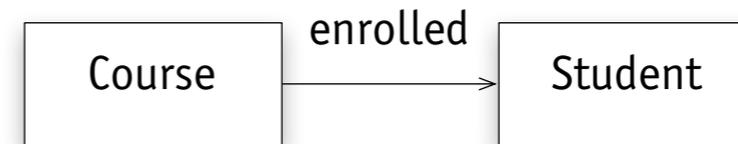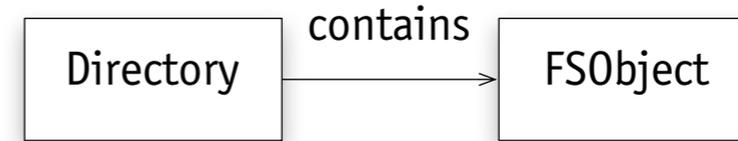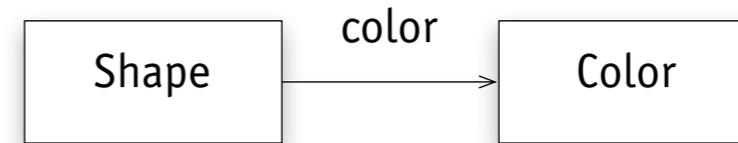
does arrow direction matter?

some relations are symmetric
› **a->b in friend** iff **b->a in friend**

but for non-symmetric relation
› **a->b in r** not same as **b->a in r**

must define & implement direction consistently
› **a->b in invites** : "a send an invitation to b"

and graphical notation may express constraint
› that depends on relation direction

| Director | appoints → | CEO |

relations on subsets

when you place a relation
› pick the smallest set
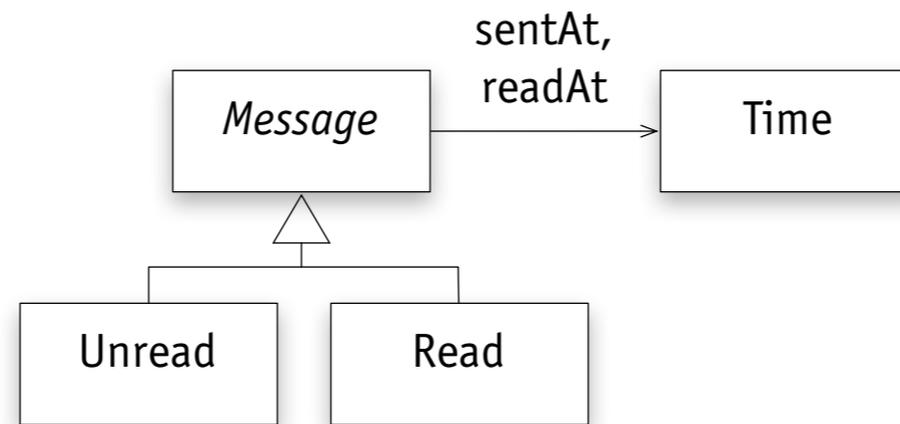
OK
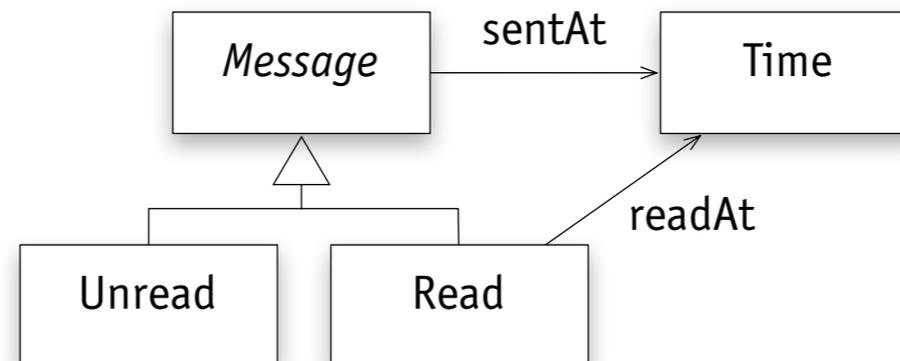
better

multiplicity

how many?

› colors per shape?
› machines per IP?

```
┌─────────┐   m        n   ┌─────────┐
│    A    │─────── R ──────▶│    B    │
└─────────┘                 └─────────┘
```
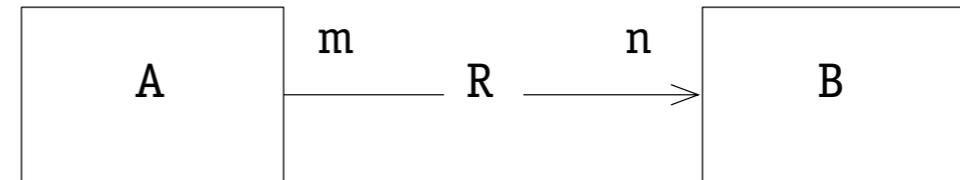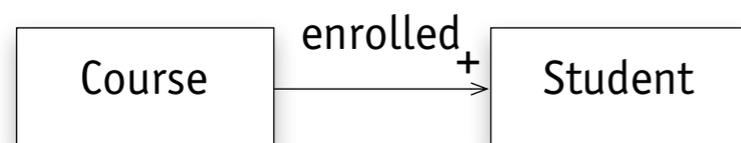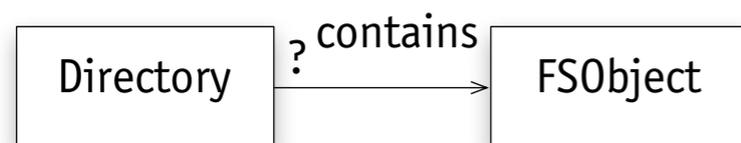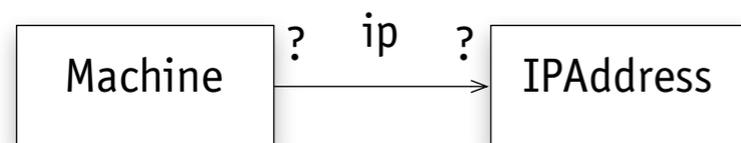
‣ R maps **m** A's to each B

‣ R maps each A to **n** B's

+ one or more
* zero or more
! exactly one
? at most one
omitted = *

```
┌─────────┐  color  !  ┌─────────┐
│  Shape  │──────────▶ │  Color  │
└─────────┘            └─────────┘
```

```
┌─────────┐  ? ip  ?  ┌───────────┐
│ Machine │────────▶  │ IPAddress │
└─────────┘           └───────────┘
```

```
┌───────────┐ ? contains ┌──────────┐
│ Directory │──────────▶ │ FSObject │
└───────────┘            └──────────┘
```

```
┌─────────┐ enrolled  + ┌─────────┐
│ Course  │──────────▶  │ Student │
└─────────┘             └─────────┘
```

```
┌────────┐  friends
│  User  │ *
└────────┘
   ▲    *
   └────────┘
```
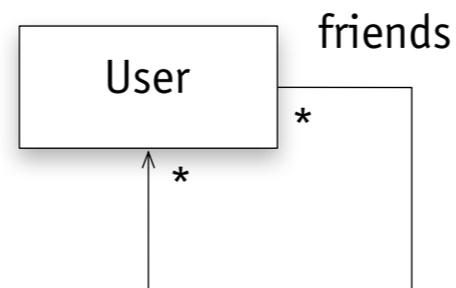
function properties

easily expressed with multiplicities
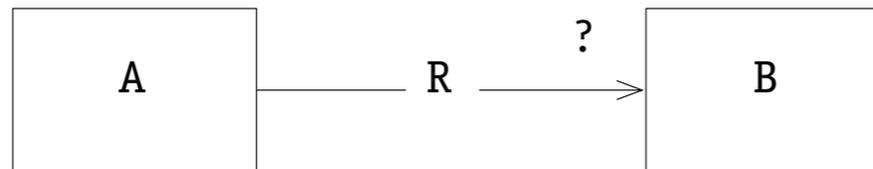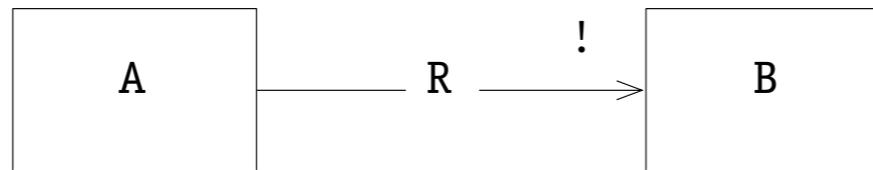
A	R —?→ B	R is a function
A	R —!→ B	R is a total function
A —?— R —!→ B		R is an injection
A —+— R —!→ B		R is a surjection
A —!— R —!→ B		R is a bijection
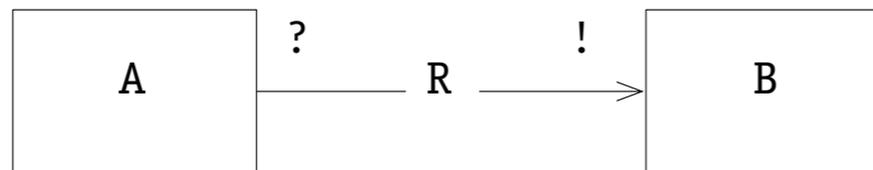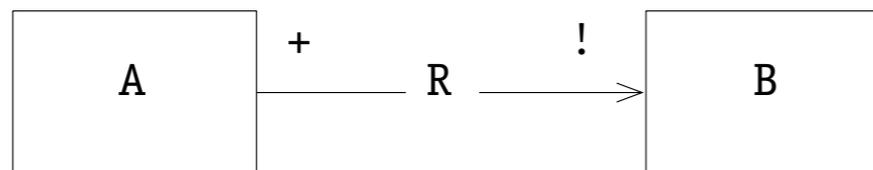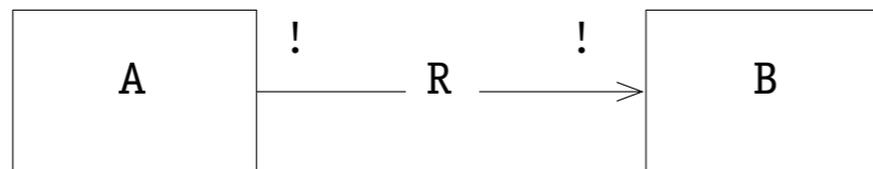
common mistakes

#1. not a stateful relation
  arrivesAt: Elevator -> Floor

#2. should be split into multiple relations
  lines: Address -> AddressLine

#3. relates >2 atoms
  lecturer: Student -> Faculty

6.170 Software Studio
Spring 2013