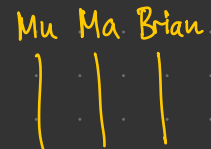


- initially, place boxes in rows/columns like a matrix

top choice = #1 = highest y

- GUI: initially, a user should be able to grab a column and move it left \rightarrow right

e.g.



Class Ranking

n = # proposals

π = permutation = ordered list

→ set/list of proposals S

reviewer_name (e.g. Murray Participant 1)

$|S| \geq n$

some reviewers don't see all proposals

class Proposal-box

* proposal →

{ graphical attributes

(x,y) screen location

· bg color, fg color

· text font, boldface, italic, ...

· frame → linetype, linewidth, line color, ...

frame2 → — " — " — — —

class Proposal

title

short_title

author, co-authors

→ review list

Class Review

reviewer

proposal

various ratings

review text

class Reviewer

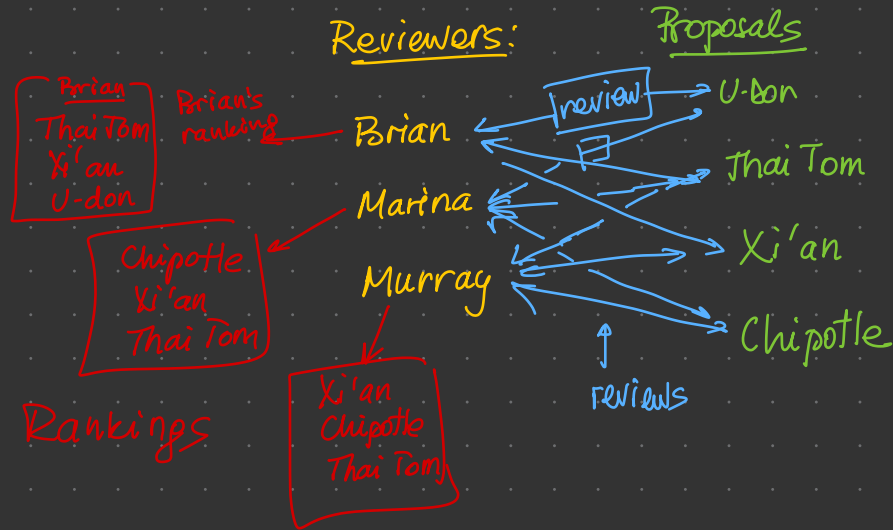
ranking

conflicts = list of Prop

reviews = list of Review

How do these come together?

- 3 -



What we want: (initially)

- to display the rankings
 - + as much information as possible, graphically
- GUI - when hovering/clicking... to pop up more info
 - e.g. Proposal info, Reviews...
- (later) allow user to move around Proposal-boxes

```
function display-listset-of-rankings
  input: list, or array of Ranking
  for r in list-of-Rankings
    x = calculate column of r ≈ x location
    display-ranking(r, x)
```

```
function display-ranking (r, x, * keywords)
```

for each proposal

```
  calculate location, display attributes
  display on screen
```

anything you like

temporarily
these y-values are
sequential like in
a table