

design by concept

Daniel Jackson · CSAIL, MIT

9th Summer School on Formal Techniques · Menlo College, CA · May 19-24, 2019

lecture

one

how this
project started

a simple task: sign and return

a simple task: sign and return

Field Trip Permission Form

Dear Parents:

Ms. Frizzle will again be taking her second grade class on an exciting field trip. Please sign and return the permission slip below.

Thank you!

Yes, I give permission for my child to go on the second grade "Touch and Feel" trip on Friday February 13th to the NastyCo Nuclear Dump. I understood that my child may encounter the normal risks of childhood play, including grazed knees, hurt feelings and exposure to toxic waste.

Parents signature

Date

a simple task: sign and return

Field Trip Permission Form

Dear Parents:

Ms. Frizzle will again be taking her second grade class on an exciting field trip. Please sign and return the permission slip below.

Thank you!

Yes, I give permission for my child to go on the second grade "Touch and Feel" trip on Friday February 13th to the NastyCo Nuclear Dump. I understood that my child may encounter the normal risks of childhood play, including grazed knees, hurt feelings and exposure to toxic waste.

Count Olaf

Parents signature

Date

a simple task: sign and return

Field Trip Permission Form

Dear Parents:

Ms. Frizzle will again be taking her second grade class on an exciting field trip. Please sign and return the permission slip below.

Thank you!

Yes, I give permission for my child to go on the second grade "Touch and Feel" trip on Friday February 13th to the NastyCo Nuclear Dump. I understood that my child may encounter the normal risks of childhood play, including grazed knees, hurt feelings and exposure to toxic waste.

Count Olaf

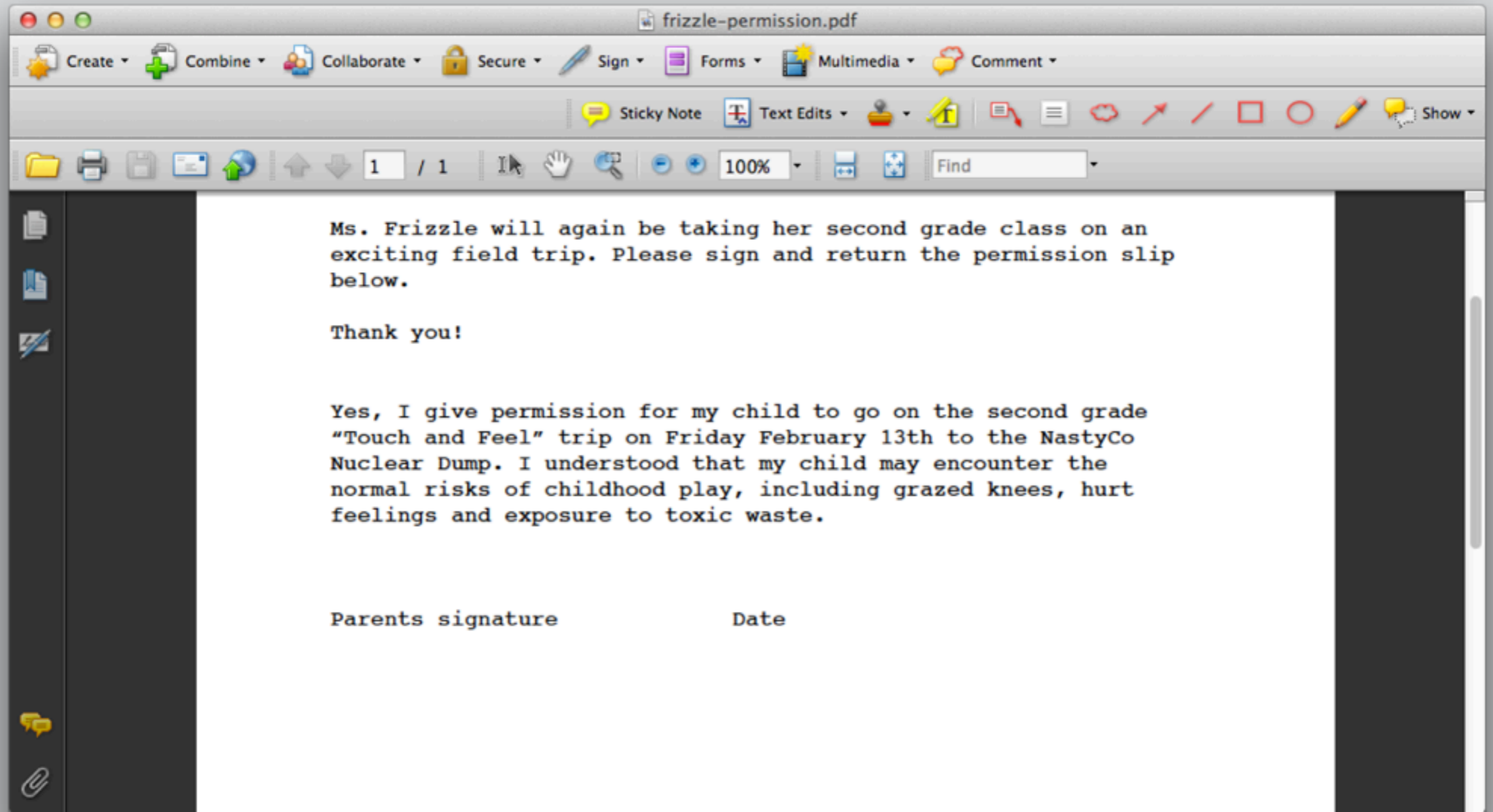
November 7, 2008

Parents signature

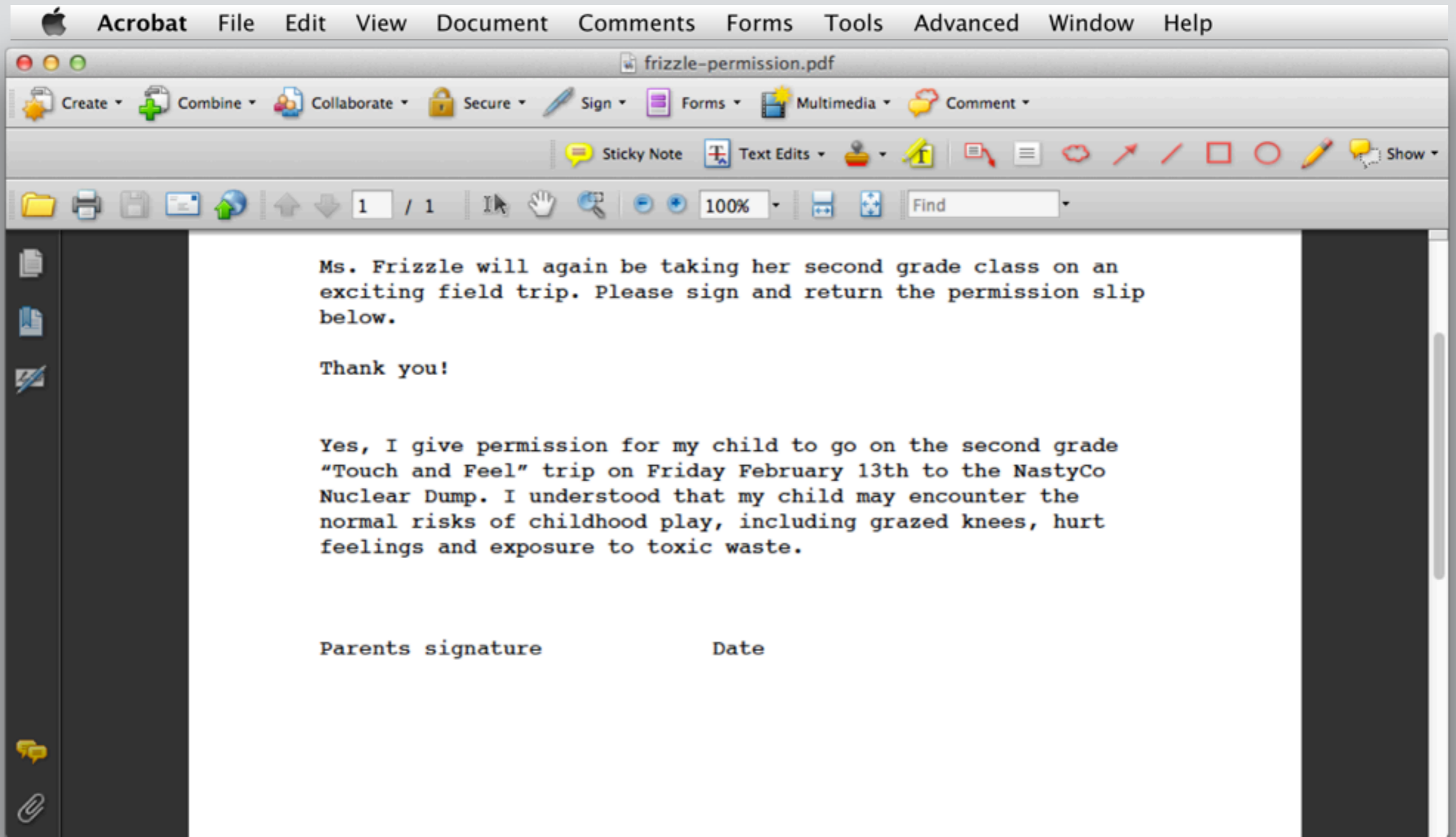
Date

acrobat to the rescue?

acrobat to the rescue?



acrobat to the rescue?



acrobat to the rescue?

The image shows a screenshot of the Adobe Acrobat application interface. The main window displays a PDF document titled "frizzle-permission.pdf". The document content reads:

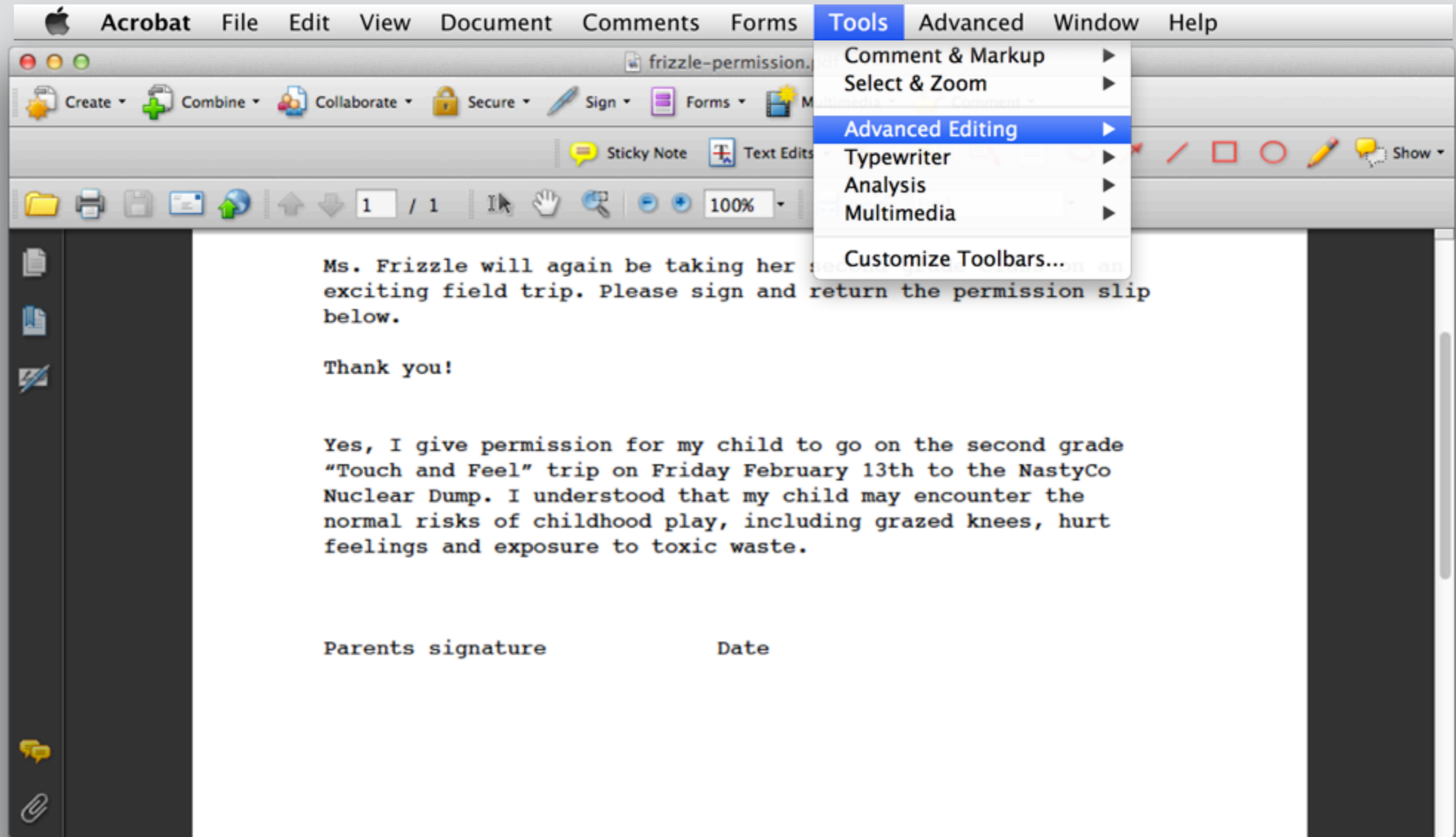
Ms. Frizzle will again be taking her second grade class on an exciting field trip. Please sign and return the permission slip below.

Thank you!

An inset window titled "acrobat-sig-paste.txt" is open, showing a list of instructions:

- 1 how to add a signature in acrobat
- 2 -- open document in acrobat
- 3 -- Tools—>Advanced Editing—>Touchup Object Tool
- 4 -- right click at desired point | Place Image...
- 5 then select jpg
- 6
- 7 how to add date
- 8 -- Tools—>Typewriter
- 9

acrobat to the rescue?



acrobat to the rescue?

The screenshot shows the Adobe Acrobat interface. The menu bar includes 'Acrobat', 'File', 'Edit', 'View', 'Document', 'Comments', 'Forms', 'Tools', 'Advanced', 'Window', and 'Help'. The 'Tools' menu is open, displaying options: 'Comment & Markup', 'Select & Zoom', 'Advanced Editing', 'Typewriter', 'Analysis', 'Multimedia', and 'Customize Toolbars...'. The 'Advanced Editing' sub-menu is also open, listing: 'Select Object Tool', 'Button', 'Article Tool', 'Crop Tool', 'Link Tool', 'TouchUp Text Tool', 'TouchUp Reading Order Tool', and 'TouchUp Object Tool', which is currently selected. The document content is a permission slip for a field trip, with fields for 'Parents signature' and 'Date'.

frizzle-permission...

Create Combine Collaborate Secure Sign Forms M

Sticky Note Text Edits

1 / 1 100%

Ms. Frizzle will again be taking her exciting field trip. Please sign and return the permission slip below.

Thank you!

Yes, I give permission for my child to go on the second grade "Touch and Feel" trip on Friday February 13th to the NastyCo Nuclear Dump. I understood that my child may encounter the normal risks of childhood play, including grazed knees, hurt feelings and exposure to toxic waste.

Parents signature Date

acrobat to the rescue?

The screenshot displays the Adobe Acrobat application window. The menu bar at the top includes 'Acrobat', 'File', 'Edit', 'View', 'Document', 'Comments', 'Forms', 'Tools', 'Advanced', 'Window', and 'Help'. The 'Tools' menu is open, showing options: 'Comment & Markup', 'Select & Zoom', 'Advanced Editing' (highlighted), 'Typewriter', 'Analysis', 'Multimedia', and 'Customize Toolbars...'. A secondary menu is open from 'Advanced Editing', listing tools: 'Select Object Tool', 'Button', 'Article Tool', 'Crop Tool', 'Link Tool', 'TouchUp Text Tool', 'TouchUp Reading Order Tool', and 'TouchUp Object Tool' (checked). Below the menu, a document page is visible with the following text:

Ms. Frizzle will again be taking her exciting field trip. Please sign and return the permission slip below.

Thank you!

Yes, I give permission for my child to go on the second grade "Touch and Feel" trip on Friday February 13th to the NastyCo Nuclear Dump. I understood that my child may encounter the normal risks of childhood play, including grazed knees, hurt feelings and exposure to toxic waste.

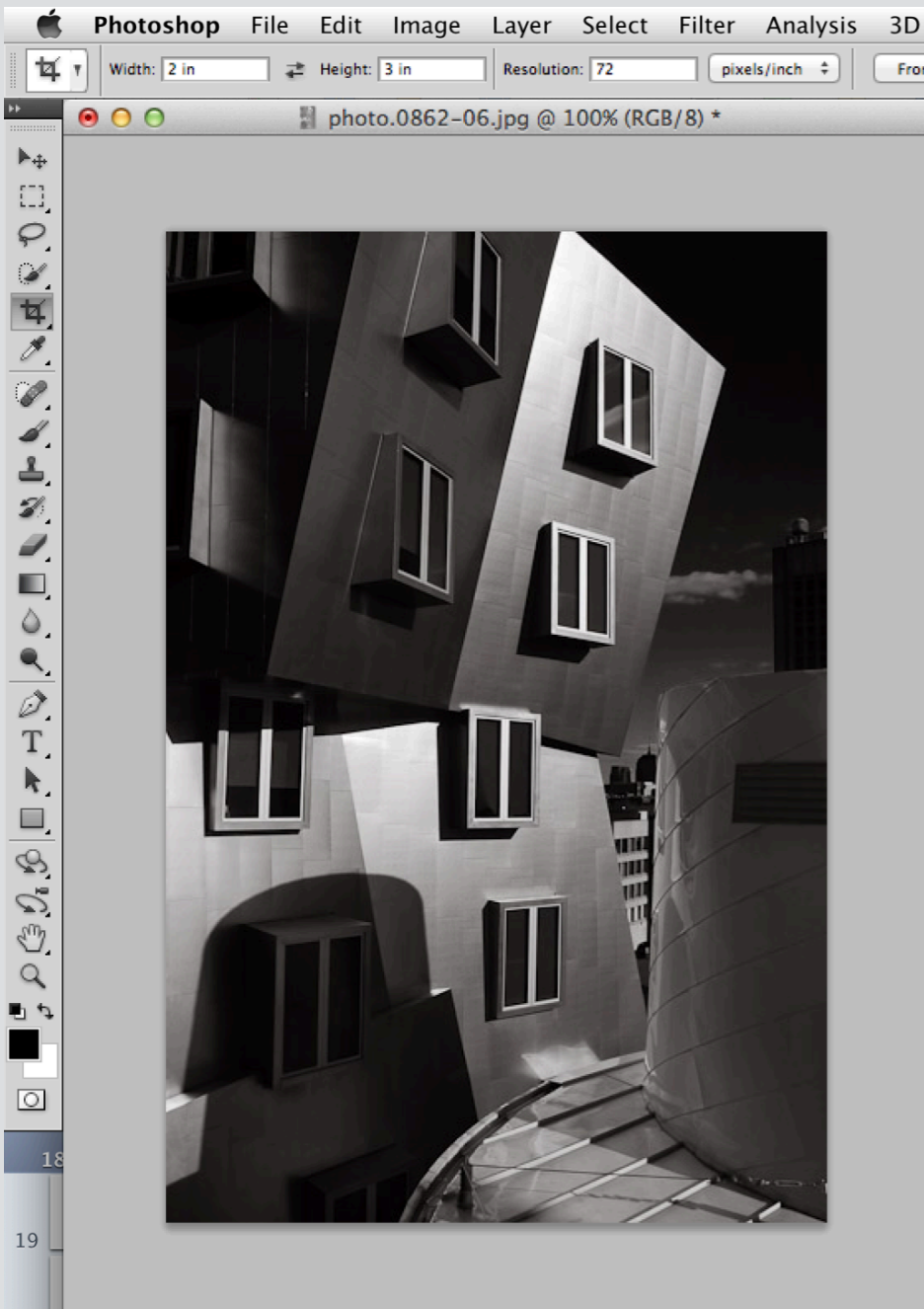
Parents _____ Date _____

A context menu is open over the 'Parents' field, containing the options: 'Select All', 'Place Image...' (highlighted), 'Find...', and 'Edit Page...'.

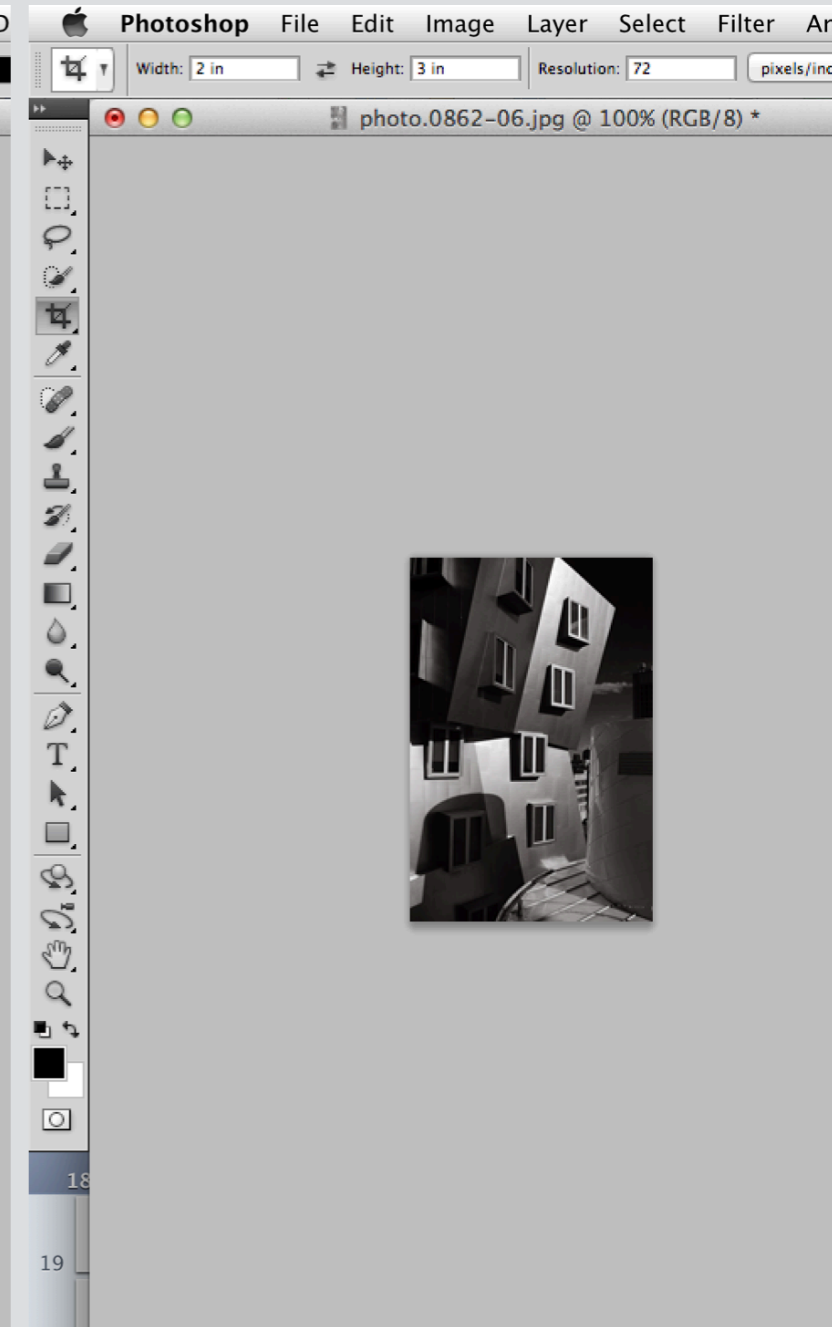
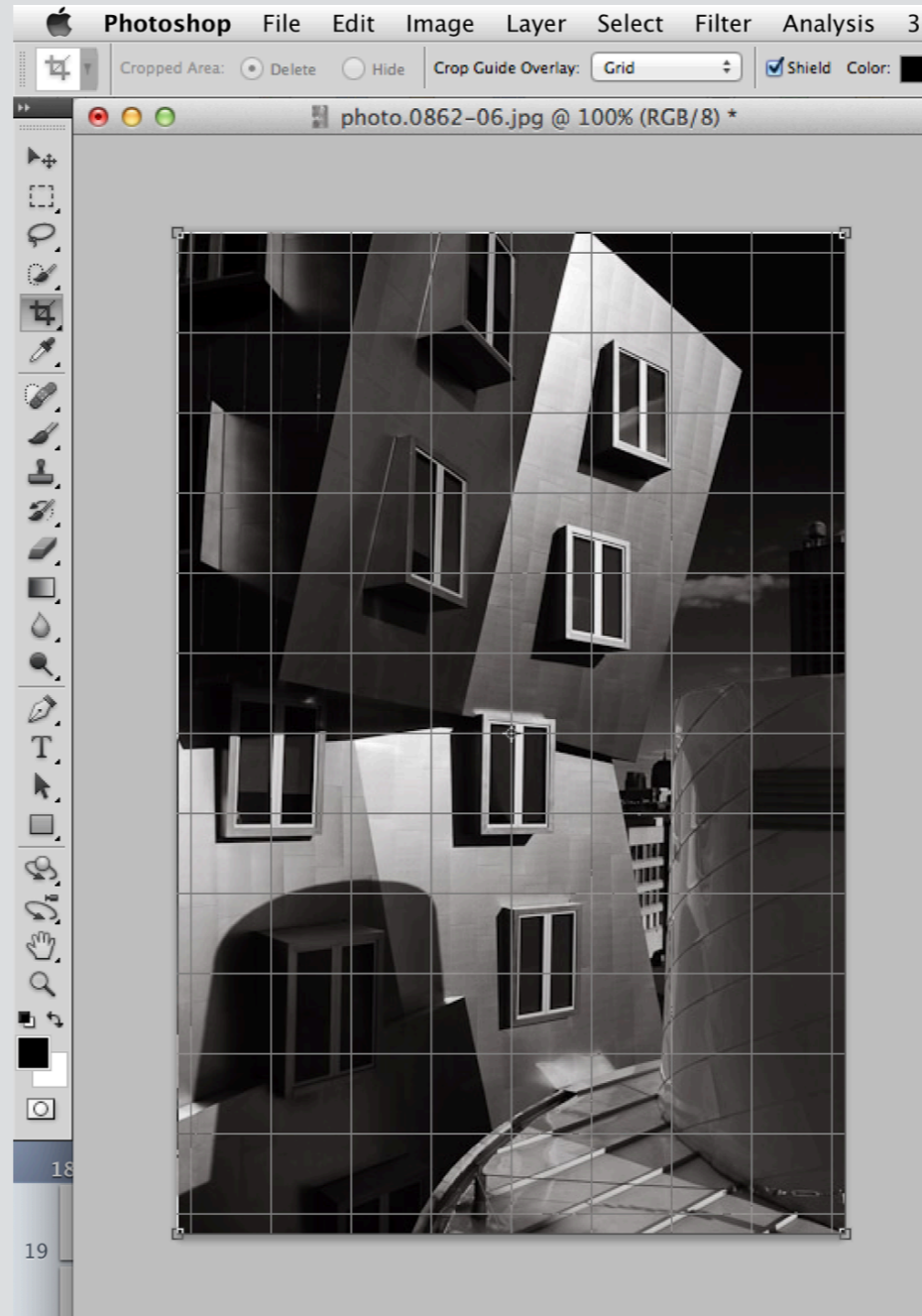
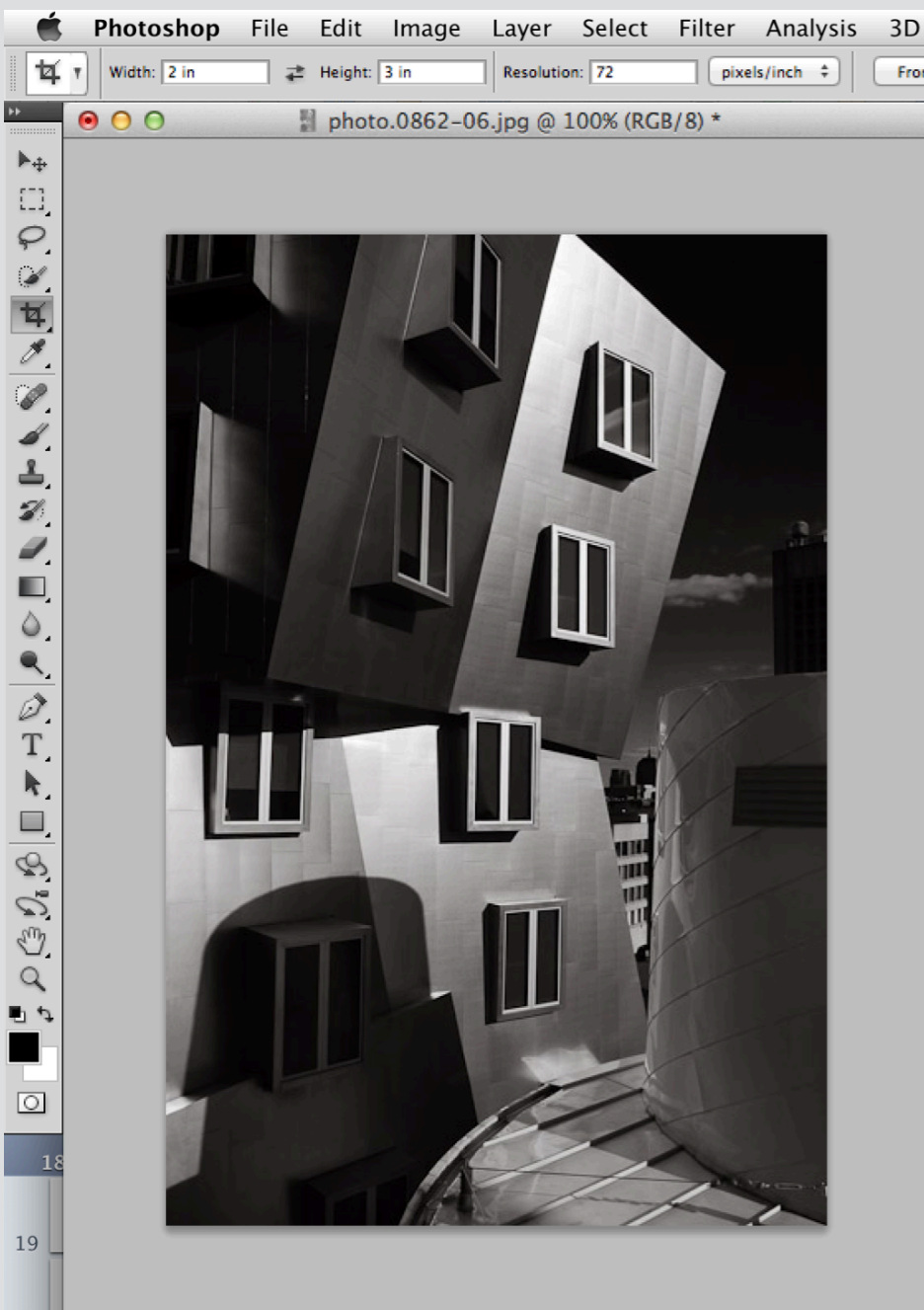
adobe photoshop: cropping surprises



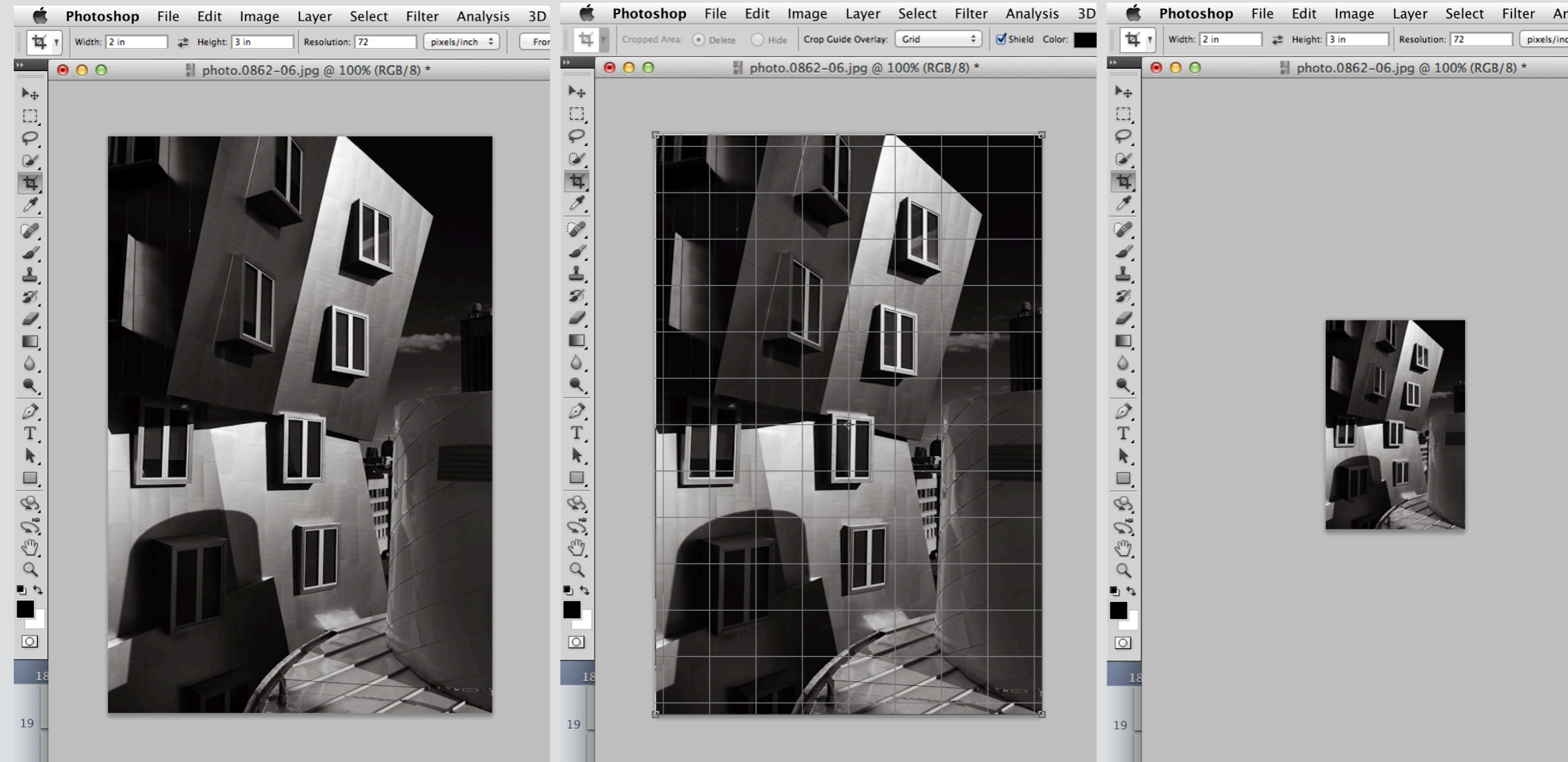
adobe photoshop: cropping surprises



adobe photoshop: cropping surprises



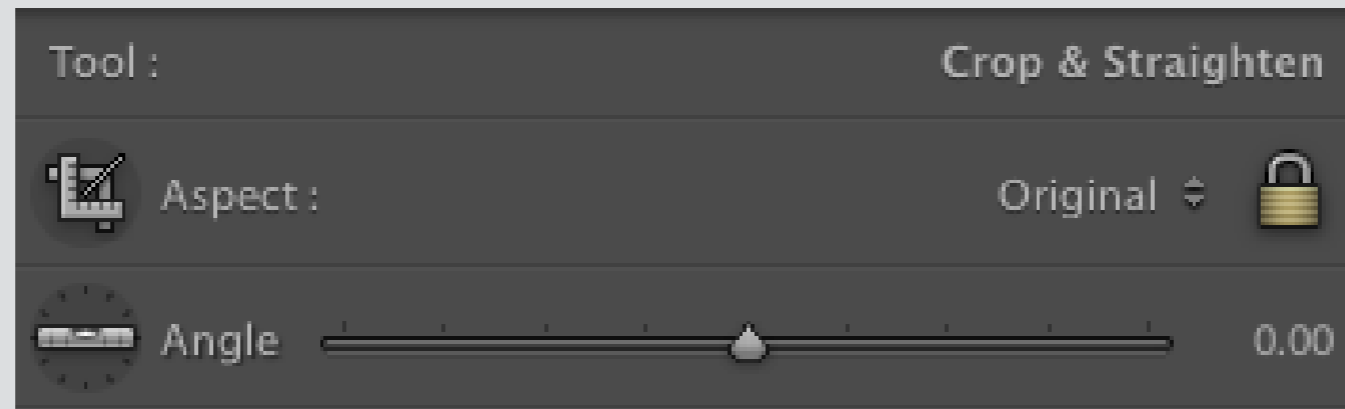
adobe photoshop: cropping surprises



any idea what's going on here?

adobe lightroom: easy cropping

adobe lightroom: easy cropping



i'm not alone

from <http://amplicate.com>

i'm not alone



Adobe Acrobat is a family of computer programs developed by Adobe Systems, designed to view, create, manipulate and manage files ... »

47% Love Acrobat



Tweet 1

+1

207 Positive Opinions out of 444



Adobe Acrobat is a family of computer programs developed by Adobe Systems, designed to view, create, manipulate and manage files ... »

53% Hate Acrobat



Tweet 15

+1

237 Negative Opinions out of 444

i'm not alone



Adobe Acrobat is a family of computer programs developed by Adobe Systems, designed to view, create, manipulate and manage files ... »

47% Love Acrobat



Tweet 1



207 Positive Opinions out of 444



Adobe Acrobat is a family of computer programs developed by Adobe Systems, designed to view, create, manipulate and manage files ... »

53% Hate Acrobat



Tweet 15



237 Negative Opinions out of 444



Adobe Photoshop Lightroom is a photography software program developed by Adobe Systems for Mac OS X and Microsoft Windows, designed ... »

89% Love Lightroom



Tweet 3

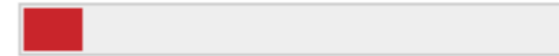


2,335 Positive Opinions out of 2,632



Adobe Photoshop Lightroom is a photography software program developed by Adobe Systems for Mac OS X and Microsoft Windows, designed ... »

11% Hate Lightroom



Tweet 5



297 Negative Opinions out of 2,632

i'm not alone



Adobe Acrobat is a family of computer programs developed by Adobe Systems, designed to view, create, manipulate and manage files ... »

47% Love Acrobat



Tweet 1



207 Positive Opinions out of 444



Adobe Photoshop is a graphics editing program developed and published by Adobe Systems Incorporated.

70% Love Photoshop



Tweet 87



30,305 Positive Opinions out of 43,283



Adobe Photoshop Lightroom is a photography software program developed by Adobe Systems for Mac OS X and Microsoft Windows, designed ... »

89% Love Lightroom



Tweet 3



2,335 Positive Opinions out of 2,632



Adobe Acrobat is a family of computer programs developed by Adobe Systems, designed to view, create, manipulate and manage files ... »

53% Hate Acrobat



Tweet 15

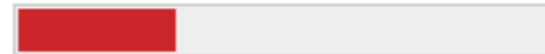


237 Negative Opinions out of 444



Adobe Photoshop is a graphics editing program developed and published by Adobe Systems Incorporated.

30% Hate Photoshop



Tweet 105

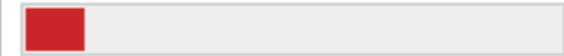


12,978 Negative Opinions out of 43,283



Adobe Photoshop Lightroom is a photography software program developed by Adobe Systems for Mac OS X and Microsoft Windows, designed ... »

11% Hate Lightroom



Tweet 5



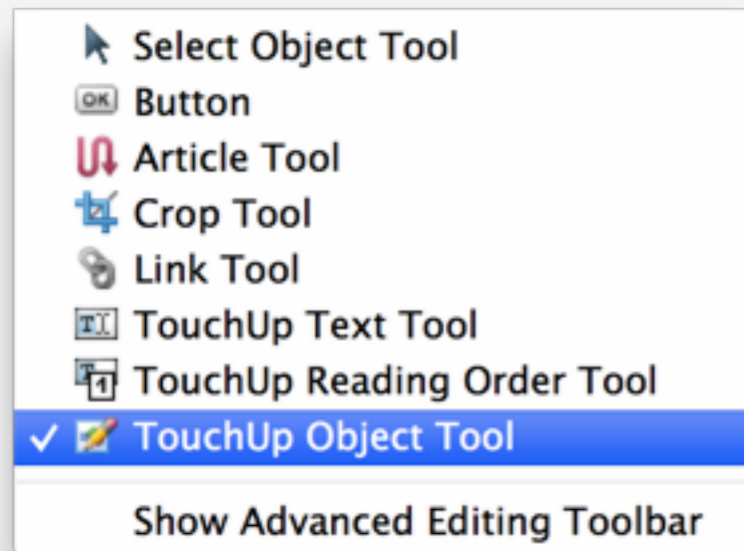
297 Negative Opinions out of 2,632

what's the essence of the problem?

not the user interface!
polished and organized

what's the essence of the problem?

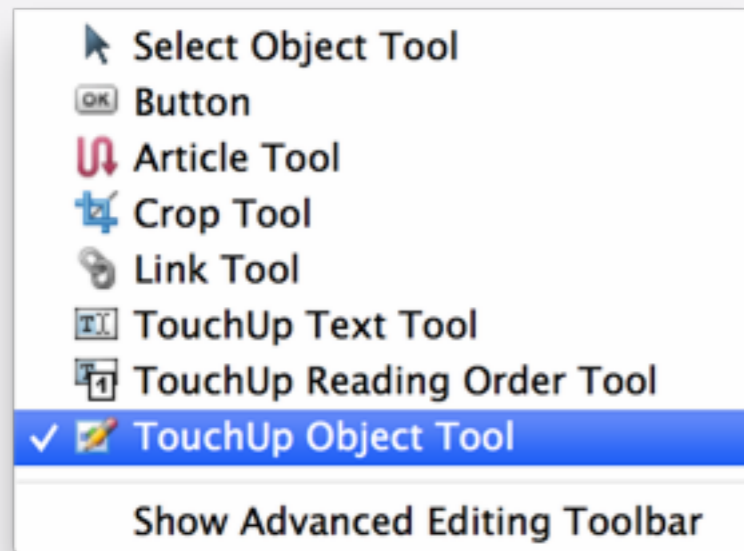
not the user interface!
polished and organized



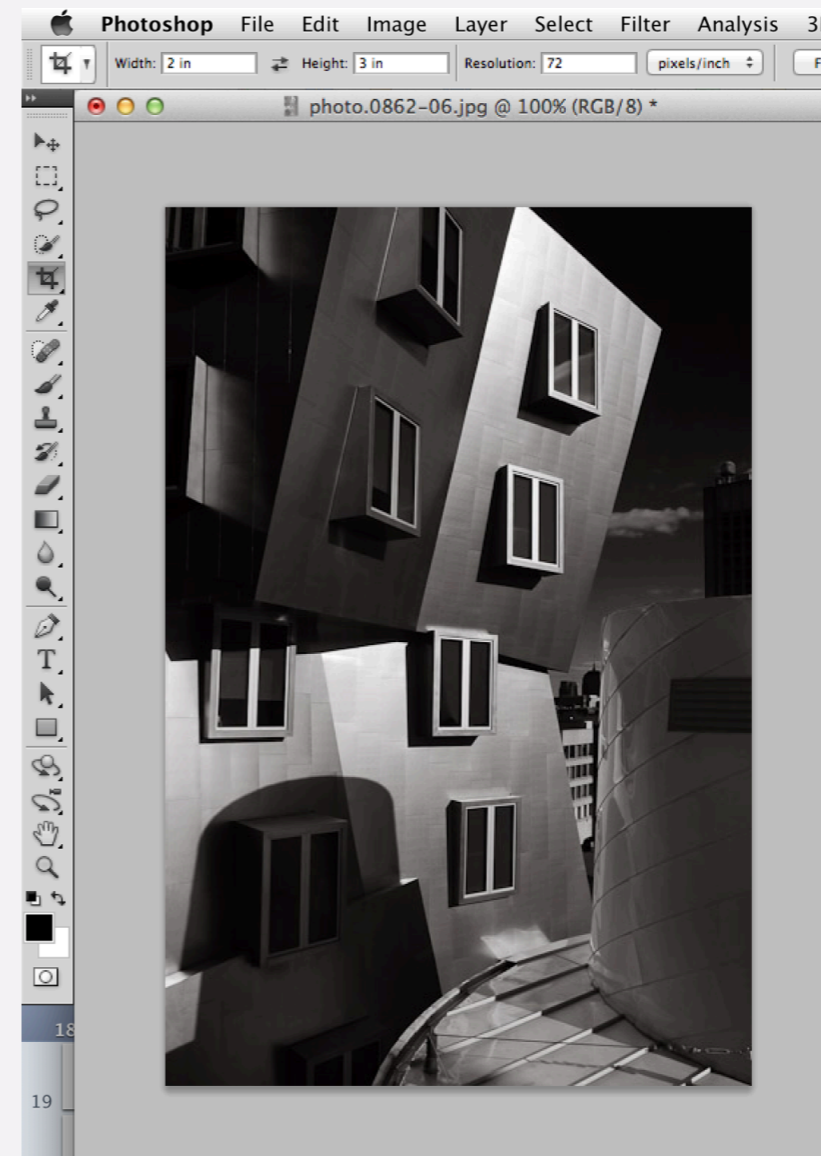
task-oriented design
no unifying concepts

what's the essence of the problem?

not the user interface!
polished and organized



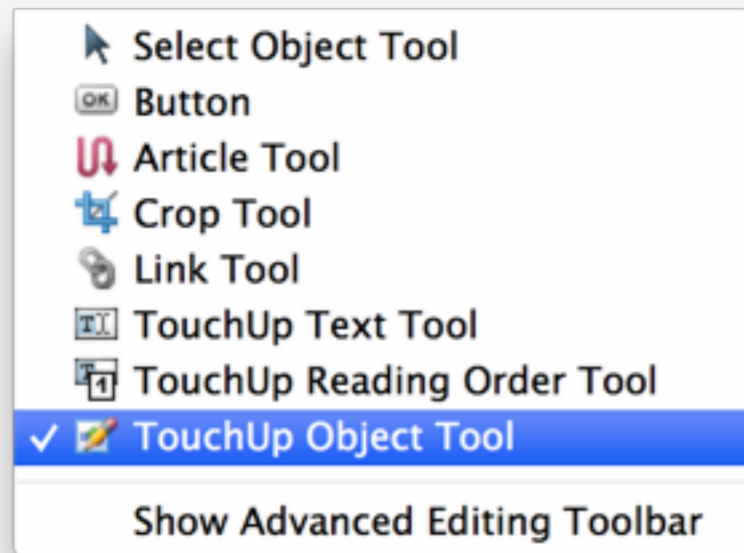
task-oriented design
no unifying concepts



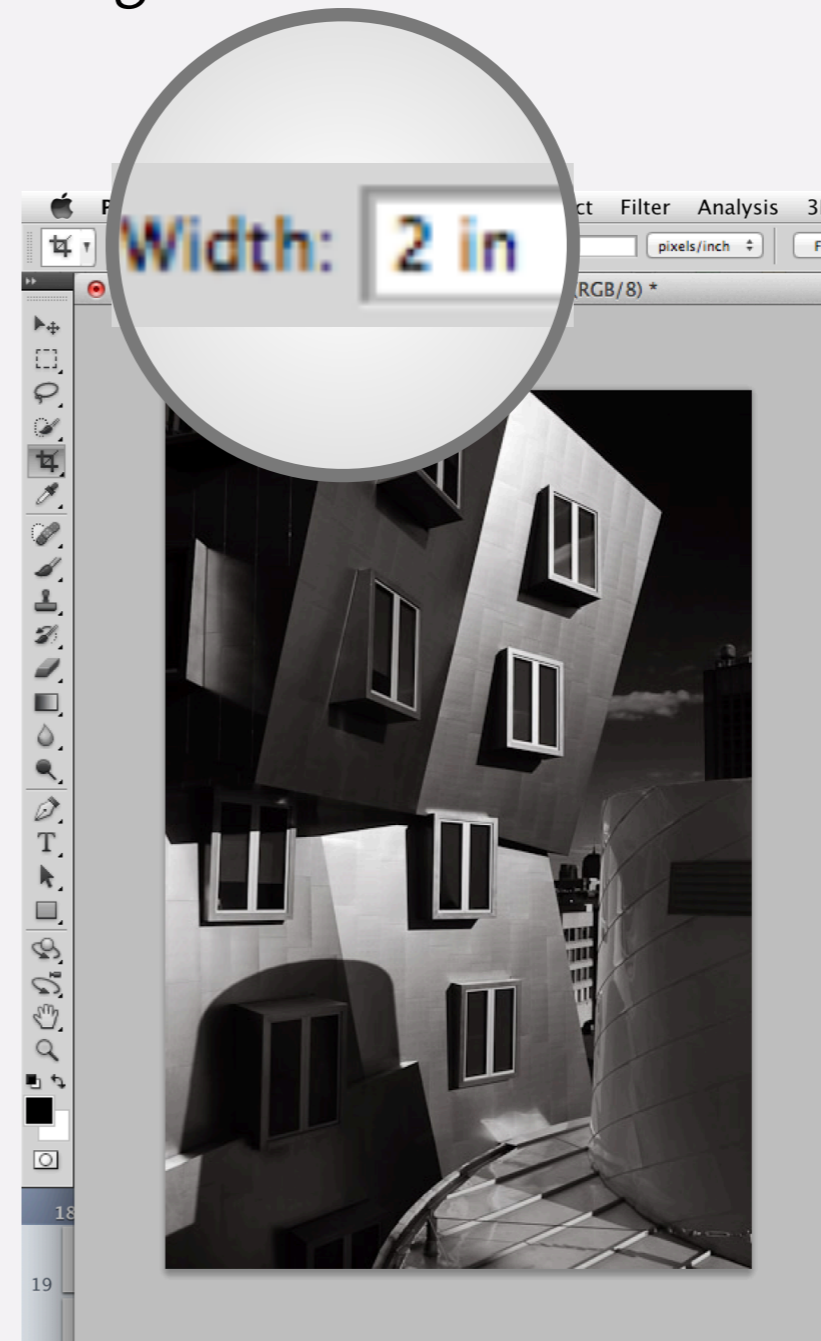
complex concepts
cropping vs. resizing

what's the essence of the problem?

not the user interface!
polished and organized



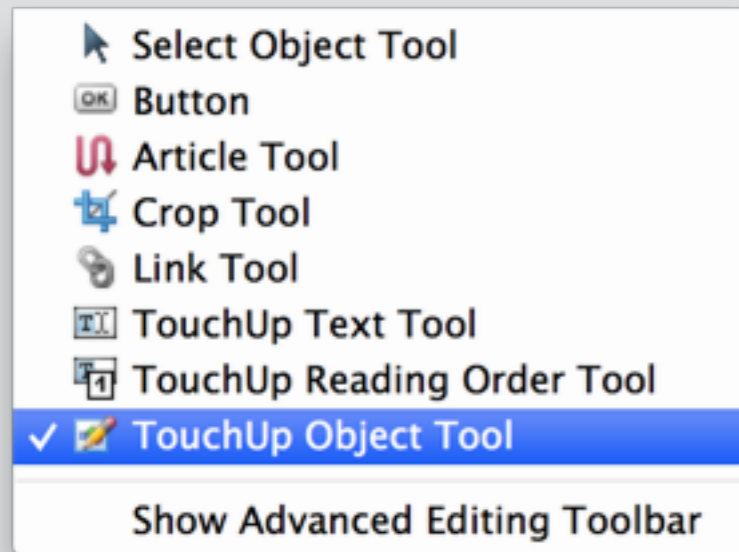
task-oriented design
no unifying concepts



complex concepts
cropping vs. resizing

adobe fixes acrobat

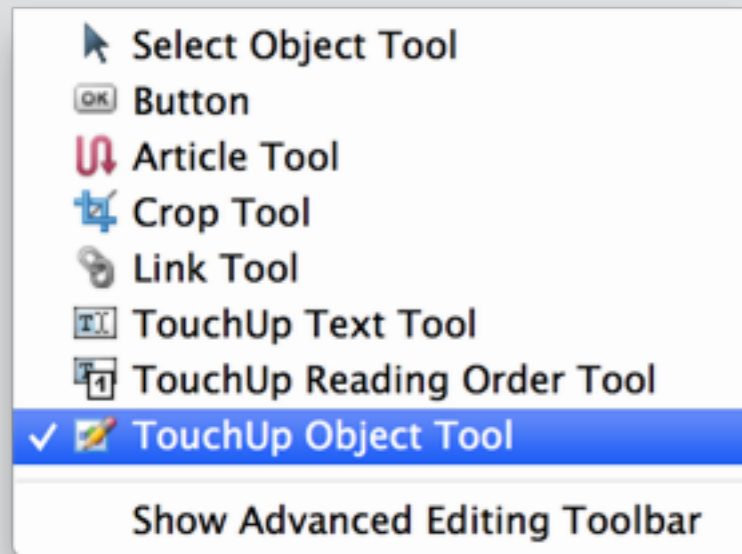
Version 9 (2008)



task-oriented design
no unifying concepts

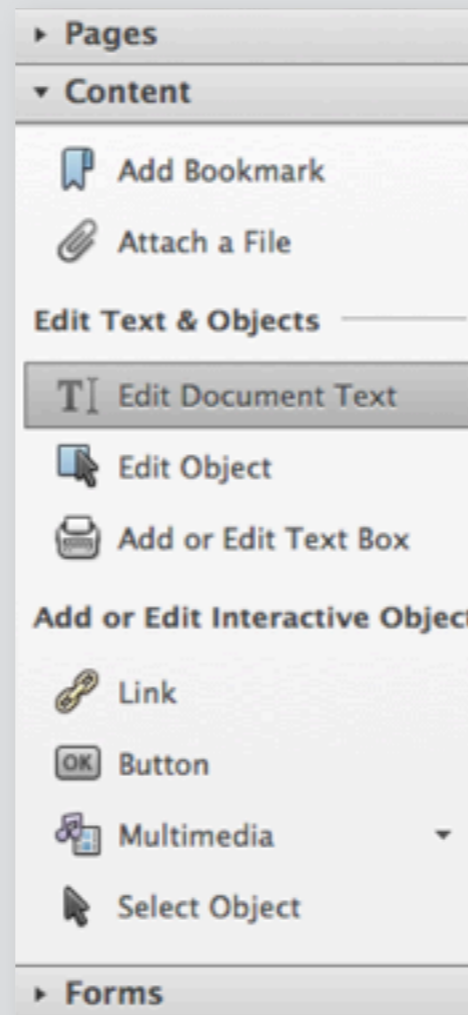
adobe fixes acrobat

Version 9 (2008)



task-oriented design
no unifying concepts

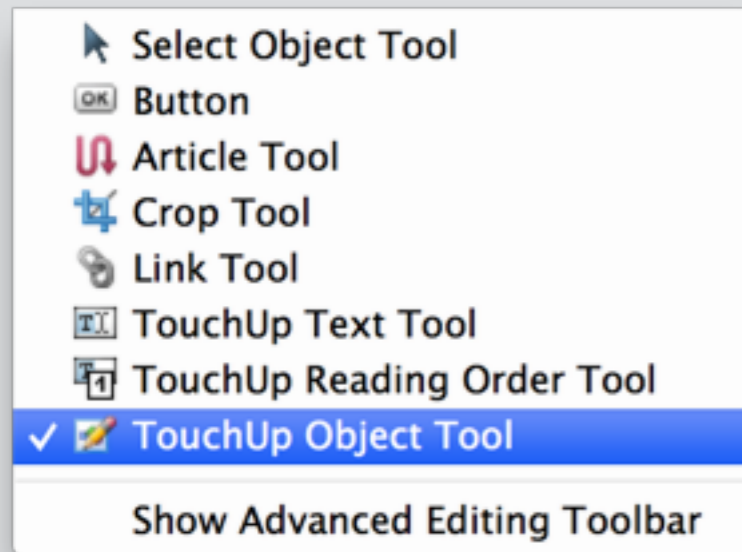
Version 10 (2010)



improved interface
but still no concepts

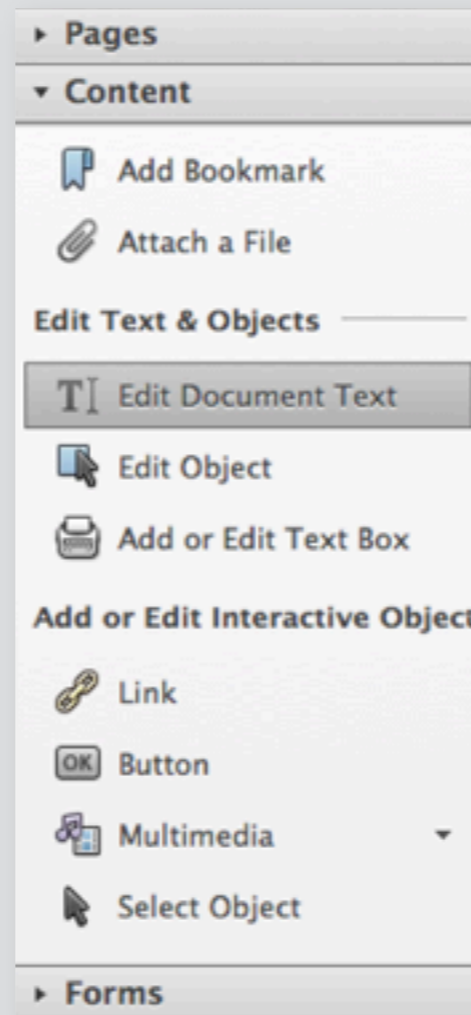
adobe fixes acrobat

Version 9 (2008)



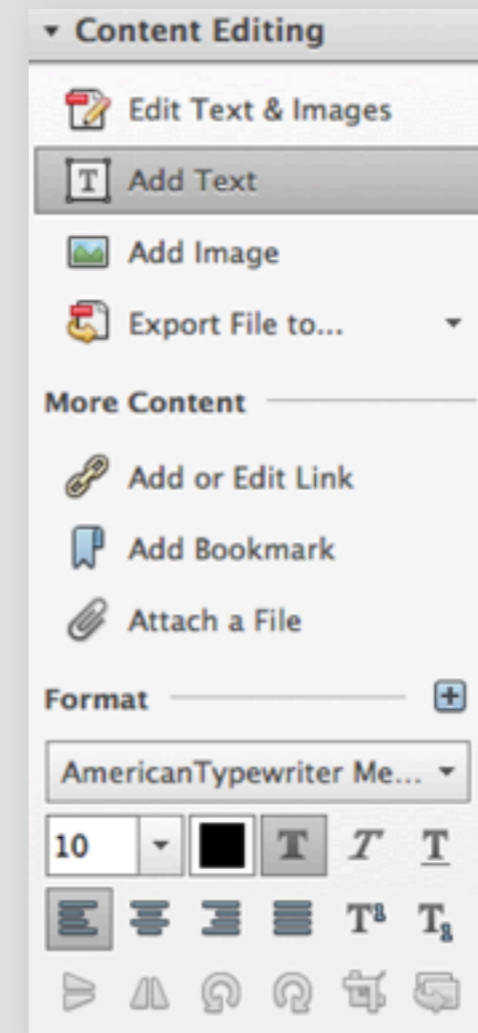
task-oriented design
no unifying concepts

Version 10 (2010)



improved interface
but still no concepts

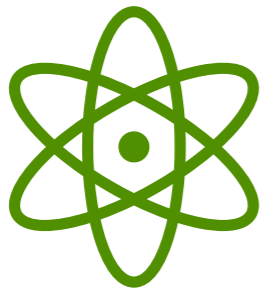
Version 11 (2012)



unifying concepts
text/image object

a research & teaching program

designing software with concepts



a design theory

*concept structure
& design rules
[Onward 15]*



design case studies

*Gitless
[Perez De Rosso,
Onward 13,
OOPSLA 16]*



design patterns

about 30 so far



code platform

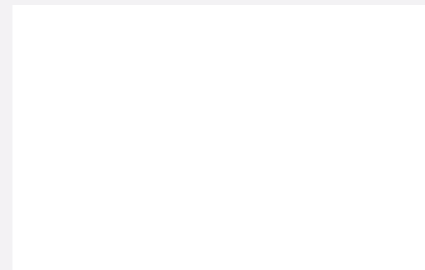
*Deja Vu
[Perez De Rosso]*

how bugs
led us astray

the software problem



need
the motivation for
building the system



implementation
the mechanisms
of the system

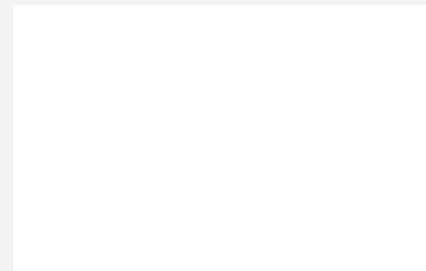
separating concerns



need
the motivation for
building the system

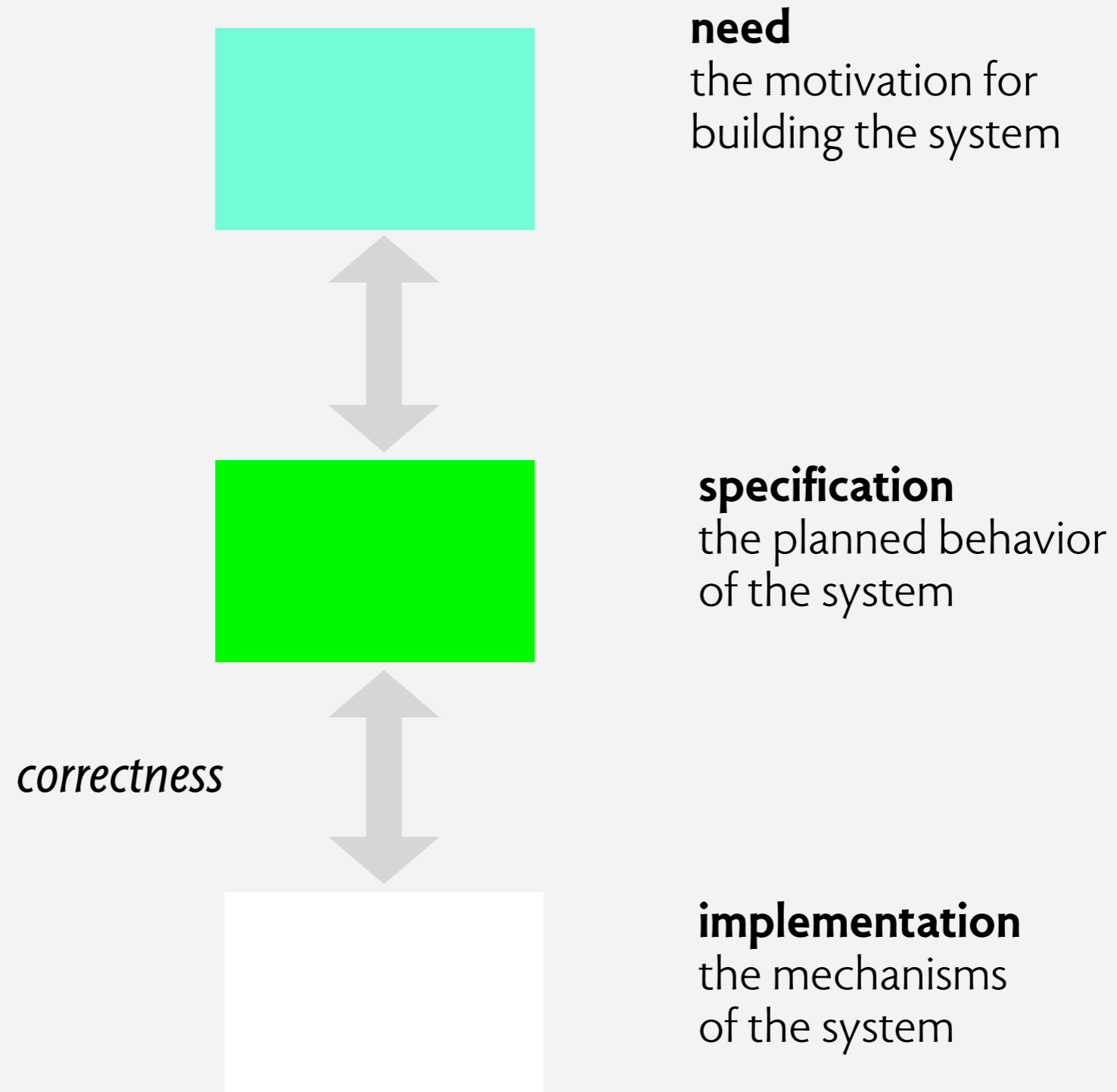


specification
the planned behavior
of the system

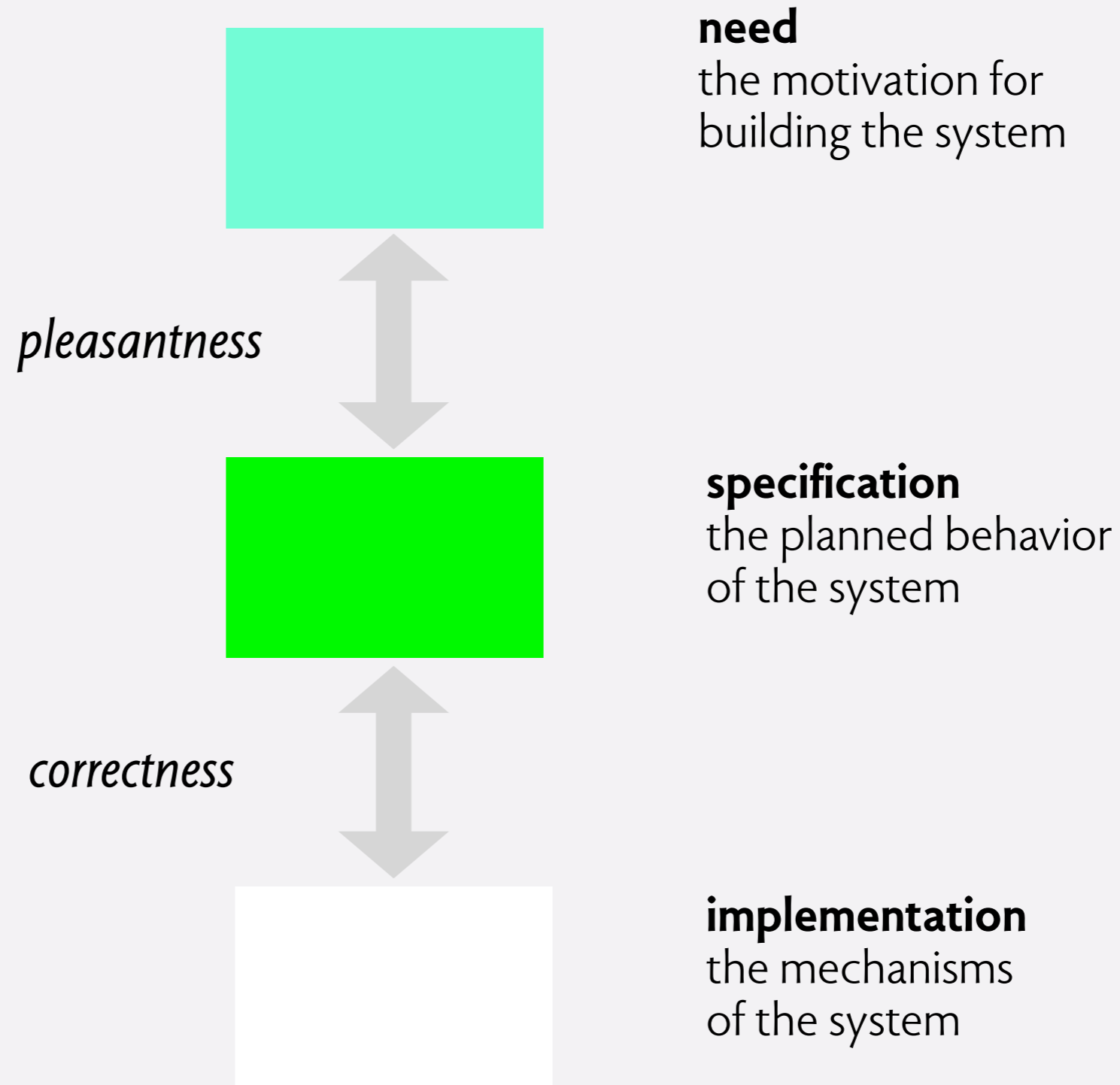


implementation
the mechanisms
of the system

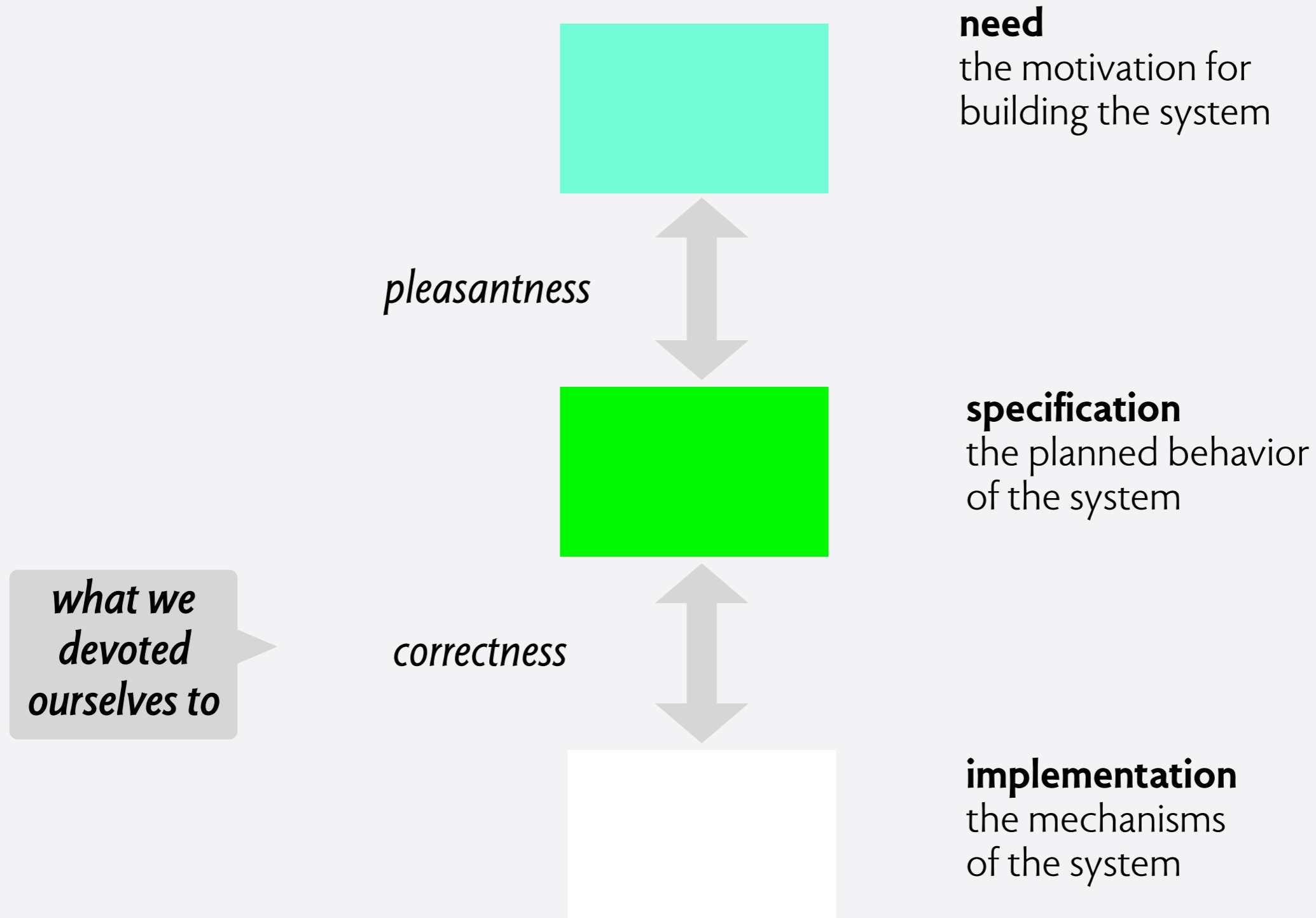
separating concerns



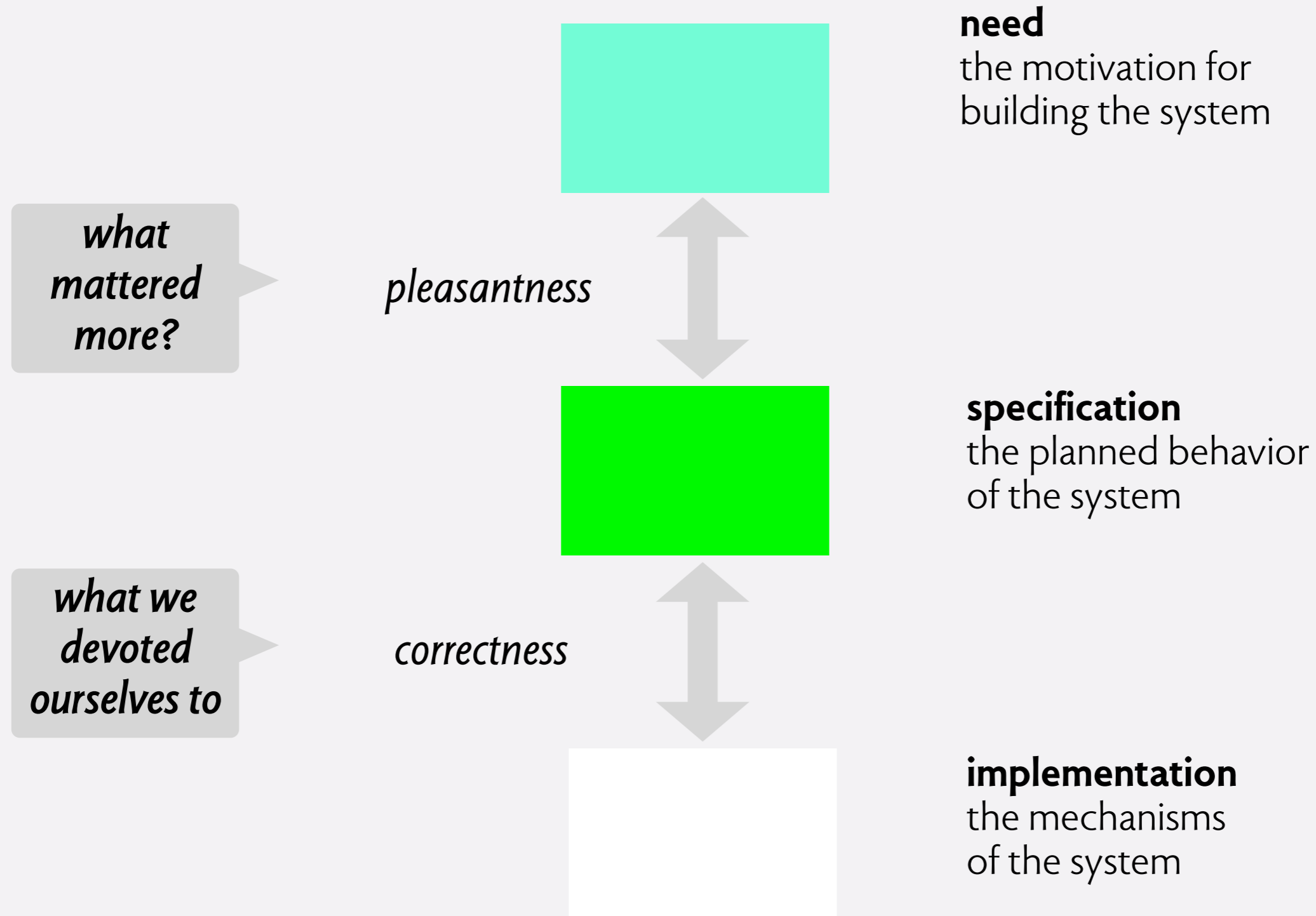
separating concerns



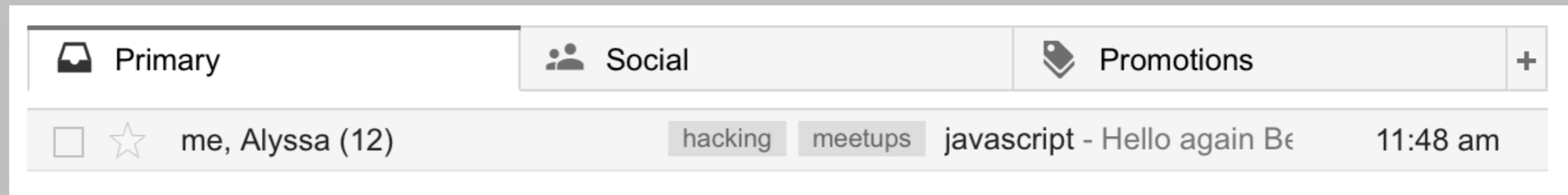
separating concerns



separating concerns



correct \Rightarrow useful ?



correct \Rightarrow useful ?

Primary Social Promotions +

☆ me, Alyssa (12) hacking meetups javascript - Hello again Be 11:48 am

label:hacking

1-1 of 1

☆ me, Alyssa (12) Inbox meetups javascript - Hello again Ben 9:43 am

correct \Rightarrow useful ?

Primary Social Promotions +

☆ me, Alyssa (12) hacking meetups javascript - Hello again Be 11:48 am

label:hacking

1-1 of 1

☆ me, Alyssa (12) Inbox meetups javascript - Hello again Ben 9:43 am

label:meetups

1-1 of 1

☆ me, Alyssa (12) Inbox hacking javascript - Hello again Ben. 9:58 am

correct \Rightarrow useful ?

Primary Social Promotions +

☆ me, Alyssa (12) hacking meetups javascript - Hello again Be 11:48 am

label:hacking

☆ me, Alyssa (12) Inbox meetups javascript - Hello again Ben 9:43 am

label:meetups

☆ me, Alyssa (12) Inbox hacking javascript - Hello again Ben. 9:58 am

label:hacking label:meetups

☆ me, Alyssa (12) Inbox hacking javascript - Hello again Ben. 9:58 am

🔍 No messages matched your search. Try using [search options](#) such as sender, date, size and more.

correct \Rightarrow useful ?

Primary Social Promotions +

☆ me, Alyssa (12) hacking meetups javascript - Hello again Be 11:48 am

label:hacking

☆ me, Alyssa (12) Inbox meetups javascript - Hello again Ben 9:43 am

label:meetups

☆ me, Alyssa (12) Inbox hacking javascript - Hello again Ben. 9:58 am

label:hacking label:meetups

☆ me, Alyssa (12) No messages match your search criteria. Try changing your search criteria, order, date, size and more.

any gmail users who can explain?

correct \Rightarrow safe ?



correct \Rightarrow safe ?

airborne \Leftrightarrow disabled

requirement



correct \Rightarrow safe ?

\neg WheelPulse \Leftrightarrow disabled

specification

airborne \Leftrightarrow disabled

requirement



correct \Rightarrow safe ?

airborne $\Leftrightarrow \neg$ WheelPulse

environment

\neg WheelPulse \Leftrightarrow disabled

specification

airborne \Leftrightarrow disabled

requirement

\wedge

$\Rightarrow?$



correct \Rightarrow secure ?

From: "TIG" <help@MIT.EDU>
Date: October 13, 2008 11:04:08 AM EDT
To: "Daniel Jackson" <dnj@csail.mit.edu>
Subject: your password

We recently ran a password checker to evaluate passwords of all CSAIL users, and your password was readily broken. Please choose a new password ASAP...

correct \Rightarrow secure ?

From: "TIG" <help@MIT.EDU>
Date: October 13, 2008 11:04:08 AM EDT
To: "Daniel Jackson" <dnj@csail.mit.edu>
Subject: your password

We recently ran a password checker to evaluate passwords of all CSAIL users, and your password was readily broken. Please choose a new password ASAP...

my password:

sergeantpepper1967

correct \Rightarrow secure ?

From: "TIG" <help@MIT.EDU>
Date: October 13, 2008 11:04:08 AM EDT
To: "Daniel Jackson" <dnj@csail.mit.edu>
Subject: your password

We recently ran a password checker to evaluate passwords of all CSAIL users, and your password was readily broken. Please choose a new password ASAP...

my password:

sergeantpepper1967

8 char limit: passwd utility silently truncated rest

correct \Rightarrow secure ?

From: "TIG" <help@MIT.EDU>
Date: October 13, 2008 11:04:08 AM EDT
To: "'Daniel Jackson'" <dnj@csail.mit.edu>
Subject: your password

We recently ran a password checker to evaluate passwords of all CSAIL users, and your password was readily broken. Please choose a new password ASAP...

my password:

sergeantpepper1967

8 char limit: passwd utility silently truncated rest

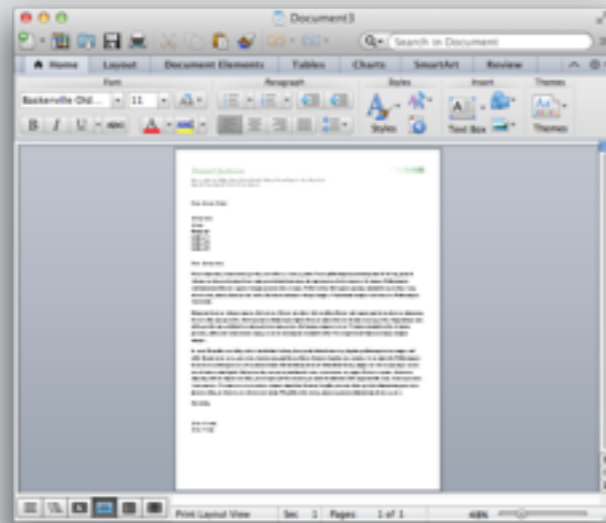
Aydal [2009]

*Analyzed Tokeneer for security
Found 9 anomalous scenarios
eg, new configuration file silently
ignored if one exists on disk*

what's a
concept?

apps characterized by their concepts

Microsoft Word



Twitter

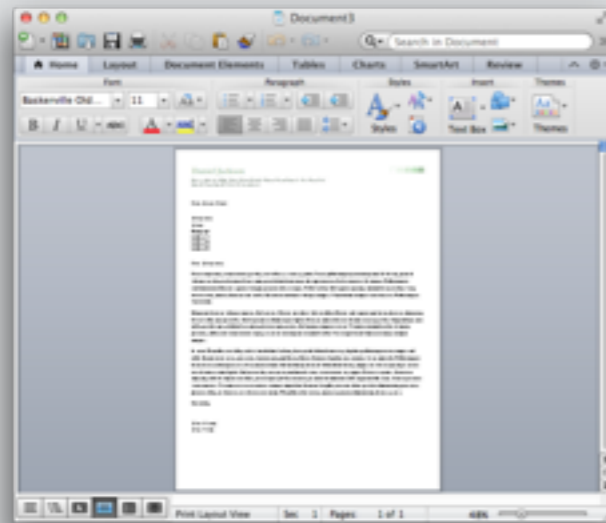


Photoshop



apps characterized by their concepts

Microsoft Word



Paragraph
Format
Style

Twitter

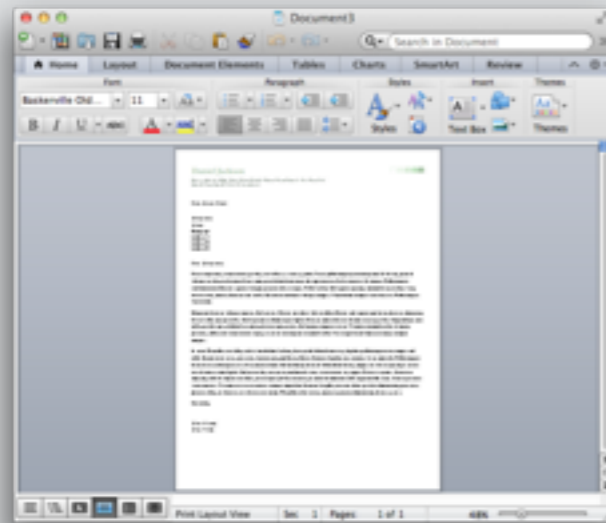


Photoshop



apps characterized by their concepts

Microsoft Word



Paragraph
Format
Style

Twitter



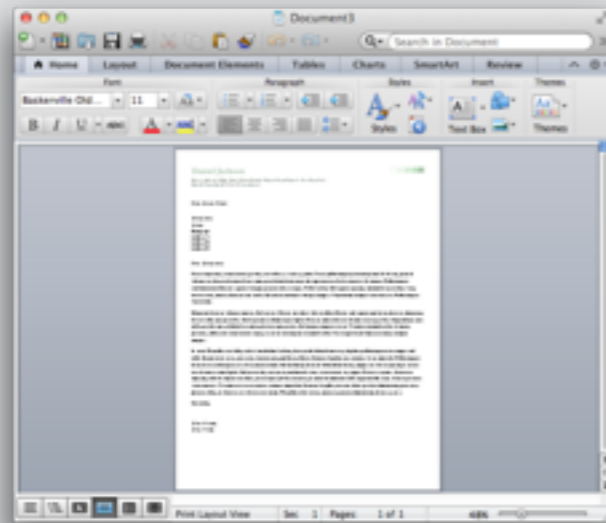
Tweet
Hashtag
Following

Photoshop



apps characterized by their concepts

Microsoft Word



Paragraph
Format
Style

Twitter



Tweet
Hashtag
Following

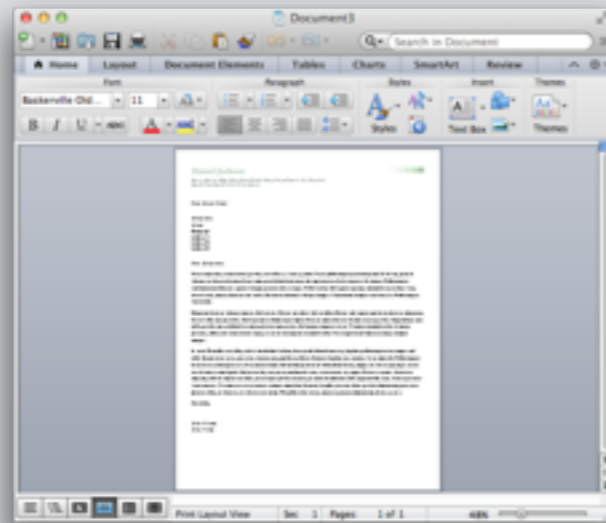
Photoshop



PixelMap
Layer/Mask
Adjustment

apps characterized by their concepts

Microsoft Word



Paragraph
Format
Style

Twitter



Tweet
Hashtag
Following

Photoshop



PixelMap
Layer/Mask
Adjustment

what's the difference between a text editor and a word processor?

app classes characterized by concepts too

app classes characterized by concepts too

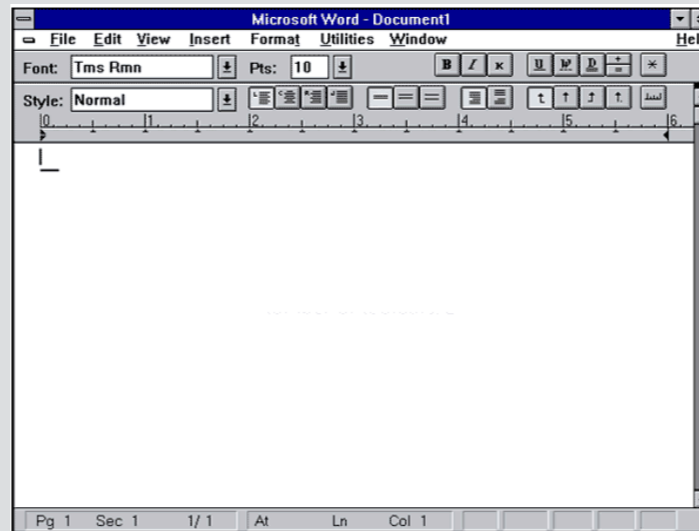
```
44 -- approaches, not satisfying: surely something essential-
45
46 Take a step back: concepts-
47 -- example: Microsoft Word: font and Co-
48 -- where concepts come from-
49 -- implications of concepts-
50 -- how concepts distinguish apps (Word from LaTeX) and app classes (Word from [old])-
51 -- what's software made of? concepts-
52
53 Towards a theory of concepts-
54 -- what's a concept?-
55 state machine of user example-
56 abstract state, exactly structural, explains behavior-
57 can explain with an operational principle without referring to purpose-
58 more complex examples: style, alias, trash, tagging, selection in Ps
59 in all cases the concept does not fulfill a function directly-
60
61 -- granularity-
62 smallest concept that's a functionality increment-
63 eg. can have layer without mask
64 but may choose to group into larger concept-
65
66 -- dependence-
67 concepts have dependencies based on functional subsets-
68 what's the point of this?
69
70 -- criteria: compelling, coherent, controllable (de-coupled), general (complete, consistent),
71 familiar (conventional)-
72
73 -- one slide and a gro and con example for each?
74
75 -- a UE notion about it being represented well?-
76 -- compelling: user facing, one motivating purpose, and an operational principle: it's about the
77 idea, not the realization-
78 -- fit for purpose: means they actually work (cf. scikit)-
79 -- familiar: idiom-
80 -- independent: orthogonal-
81 -- coherent: unlike compelling, it's about the realization; means you can predict once you
82 understand a bit-
83 -- maybe means you expect them to fit together?-
84 deletion in Mail doesn't have meta data-
85 typeface variants in Indesign don't correspond to italic and bold-
86 pixel width in CSS doesn't correspond to pixels-
87 hiding in Google forms doesn't prevent publication of data-
88 conversation in Apple Mail: can't predict what will happen with delete or move or many other
89 operations; too complex and brittle-
90 -- complete: with respect to other concepts, e.g. USB settings-
91 -- consistent: e.g., filters vs rules in Mail: maybe subsume both UE criteria under this-
92 -- analyze Acrobat, in S, concepts aren't visible: in UI, they become visible, but they're not
93 general, so they're not understandable or familiar-
94
```

text editor

app classes characterized by concepts too

```
44 -- approaches, not satisfying: surely something essential-
45
46 Take a step back: concepts-
47 -- example: Microsoft Word: font and Co-
48 -- where concepts come from-
49 -- implications of concepts-
50 -- how concepts distinguish apps (Word from LaTeX) and app classes (Word from [old])-
51 -- what's software made of? concepts-
52
53 Towards a theory of concepts-
54 -- what's a concept?
55 state machine of user example-
56 abstract state, usually structural, explains behavior-
57 can explain with an operational principle without referring to purpose-
58 more complex examples: style, alias, drag, tagging, selection in Ps
59 in all cases the concept does not fulfill a function directly-
60
61 -- granularity-
62 smallest concept that's a functionality increment-
63 eg. can have layer without mask
64 but may choose to group into larger concept-
65
66 -- dependence-
67 concepts have dependencies based on functional subsets-
68 what's the point of this?
69
70 -- criteria: compelling, coherent, controllable (de-coupled), general (complete, consistent),
71 familiar (conventional)-
72
73 -- one slide and a pro and con example for each?
74
75 -- a UE notion about it being represented well?
76 -- compelling: user facing, one motivating purpose, and an operational principle: it's about the
77 idea, not the realization-
78 -- fit for purpose: means they actually work (cf. scifits)-
79 -- familiar: obvious
80 -- independent: orthogonal-
81 -- coherent: unlike compelling, it's about the realization; means you can predict once you
82 understand a bit-
83 -- maybe means you expect them to fit together?
84 deletion in Mail doesn't have meta data-
85 typeface variants in Indesign don't correspond to italic and bold-
86 sized width in CSS doesn't correspond to pixels-
87 hiding in Google forms doesn't prevent publication of data-
88 conversation in Apple Mail: can't predict what will happen with delete or move or many other
89 operations; too complex and brittle-
90 -- compares with respect to other concepts, e.g. USB settings-
91 -- consistent: e.g., filters vs rules in Mail: maybe subsume both UE criteria under this-
92 -- analyze Acrobat, in S, concepts aren't visible in UI, they become visible, but they're not
93 general, so they're not understandable or familiar-
94
```

text editor



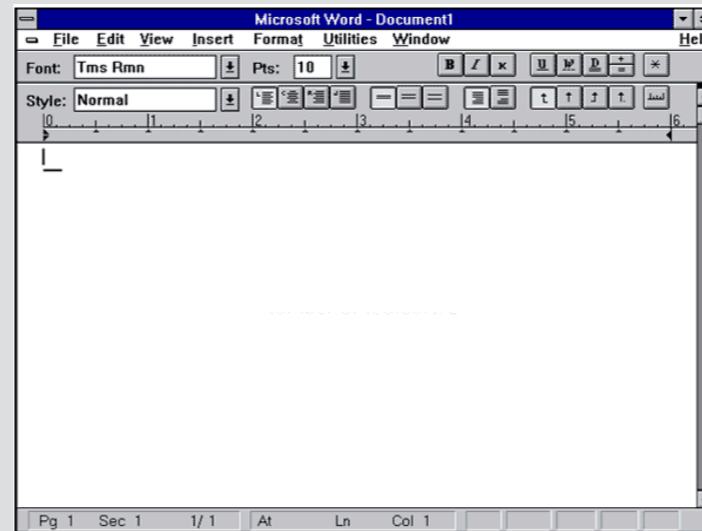
word processor

app classes characterized by concepts too

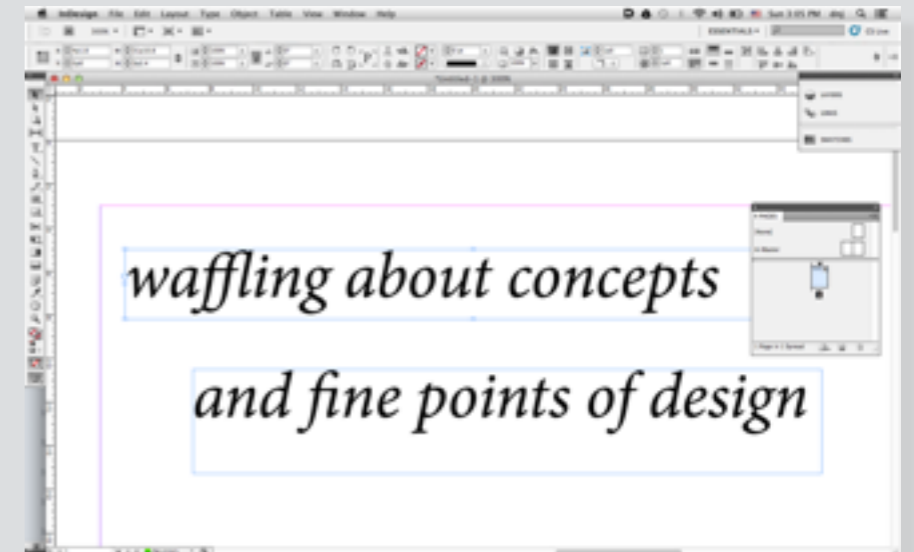


```
44 -- approaches, not satisfying: surely something essential-
45
46 Take a step back: concepts-
47 -- example: Microsoft Word: font and Co-
48 -- where concepts come from-
49 -- implications of concepts-
50 -- how concepts distinguish apps (Word from LaTeX) and app classes (Word from [old])
51 -- what's software made of? concepts-
52
53 Towards a theory of concepts-
54 -- what's a concept?
55 state machine of lines? example-
56 abstract state, usually structural, explains behavior-
57 can explain with an operational principle without referring to purpose-
58 more complex examples: style, alias, drag, tagging, selection in Ps
59 in all cases the concept does not fulfill a function directly-
60
61 -- granularity-
62 smallest concept that's a functionality increment-
63 eg. can have layer without mask
64 but may choose to group into larger concept-
65
66 -- dependence-
67 concepts have dependencies based on functional subsets-
68 what's the point of this?
69
70 -- criteria: compelling, coherent, controllable (de-coupled), general (complete, consistent),
71 familiar (conventional)-
72
73 -- one slide and a pro and con example for each?
74
75 -- a UE notion about it being represented well?
76 -- compelling: user facing, one motivating purpose, and an operational principle: it's about the
77 idea, not the realization-
78 -- fit for purpose: means they actually work (cf. profits)-
79 -- familiar: obvious
80 -- independent: orthogonal-
81 -- coherent: unlike compelling, it's about the realization; means you can predict once you
82 understand a bit-
83 -- maybe means you expect them to fit together?
84 deletion in Mail doesn't have meta data-
85 typeface variants in Indesign don't correspond to italic and bold-
86 pixel width in CSS doesn't correspond to pixels-
87 hiding in Google forms doesn't prevent publication of data-
88 conversation in Apple Mail: can't predict what will happen with delete or move or many other
89 operations; too complex and brittle)-
90 -- complete: with respect to other concepts, e.g. color settings-
91 -- consistent: e.g., filters vs rules in Mail: maybe subsume both UE criteria under this-
92 -- analyze Acrobat, in S, concepts aren't visible in UI, they become visible, but they're not
93 general, so they're not understandable or familiar-
94
```

text editor



word processor



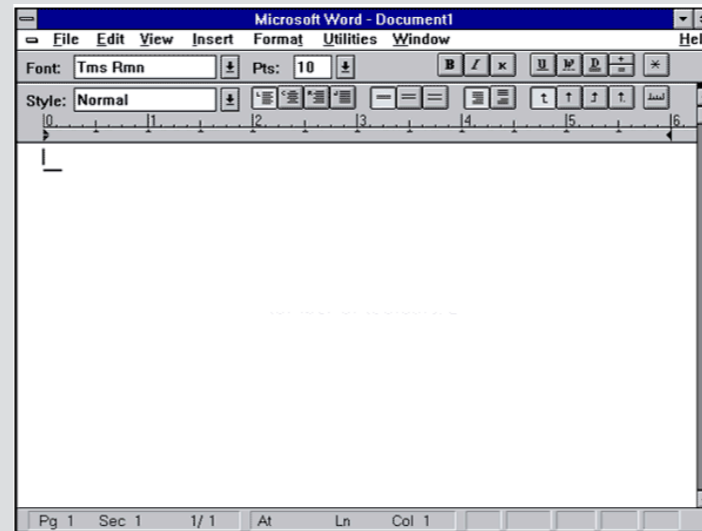
desktop publishing app

app classes characterized by concepts too

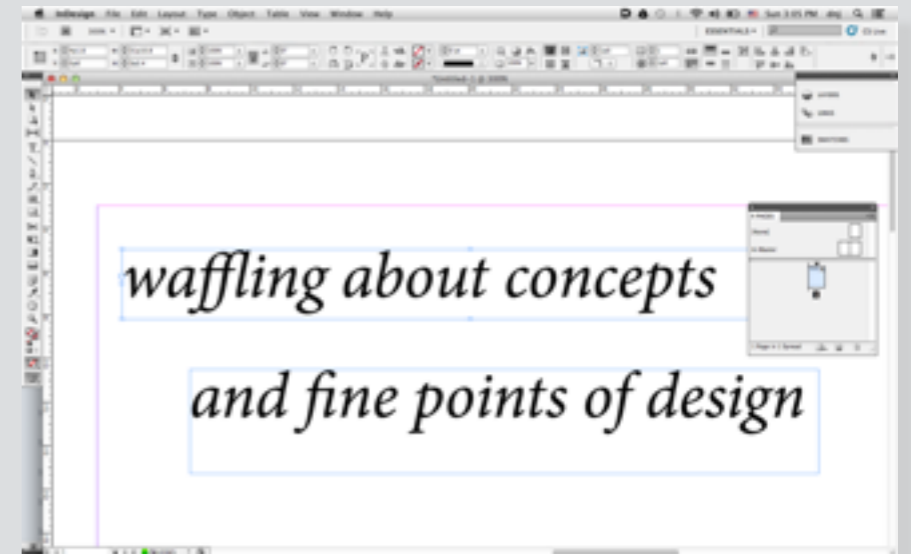


44 -- approaches, not satisfying: surely something essential-
45
46 Take a step back: concepts-
47 -- example: Microsoft Word: font and Co-
48 -- where concepts come from-
49 -- implications of concepts-
50 -- how concepts distinguish apps (Word from LaTeX) and app classes (Word from [old])
51 -- what's software made of? concepts-
52
53 Towards a theory of concepts-
54 -- what's a concept? -
55 state machine of lines? example -
56 abstract state, exactly structural, explains behavior-
57 can explain with an operational principle without referring to purpose-
58 more complex examples: style, alias, drag, tagging, selection in Ps
59 in all cases the concept does not fulfill a function redirective-
60
61 -- granularity-
62 smallest concept that's a functionality increment-
63 eg. can have layer without mask
64 but may choose to group into larger concept-
65
66 -- dependence-
67 concepts have dependencies based on functional subsets-
68 what's the point of this? -
69 -- criteria: compelling, coherent, controllable (de-coupled), general (complete, consistent),
70 familiar (conventional) -
71
72 -- one slide and a gro and an example for each? -
73
74 -- a UE notion about it being represented well? -
75 -- compelling: user facing, one motivating purpose, and an operational principle: it's about the
76 idea, not the realization -
77 -- fit for purpose: means they actually work (cf. profits) -
78 -- familiar: obvious
79 -- independent: orthogonal -
80 -- coherent: unlike compelling, it's about the realization; means you can predict once you
81 understand a bit -
82 -- maybe means you expect them to fit together? -
83 deletion in Mail doesn't have meta data -
84 typeface variants in Indesign don't correspond to italic and bold -
85 pixel width in CSS doesn't correspond to pixels -
86 hiding in Google forms doesn't prevent publication of data -
87 conversation in Apple Mail can't predict what will happen with delete or move or many other
88 operations; too complex and brittle -
89 -- compares with respect to other concepts, e.g. USB settings -
90 -- consistent: e.g., filters vs rules in Mail; maybe subsume both UE criteria under this -
91 -- analyze Acrobat, in S, concepts aren't visible in UI, they become visible, but they're not
92 general, so they're not understandable or familiar -
93
94

text editor
line buffer



word processor



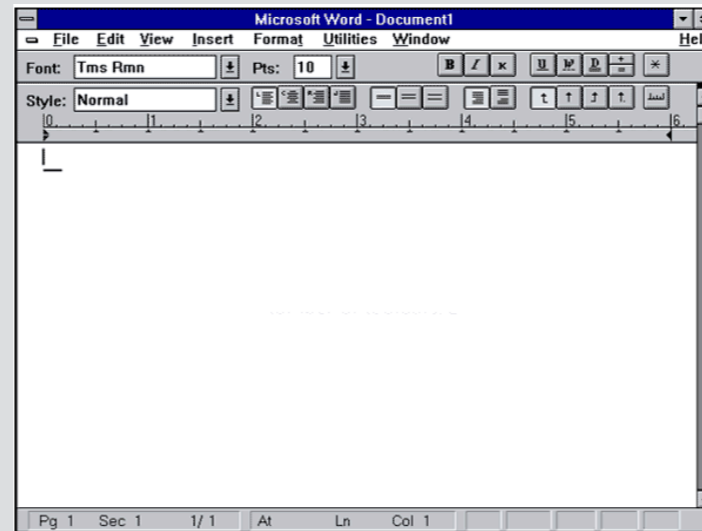
desktop publishing app

app classes characterized by concepts too

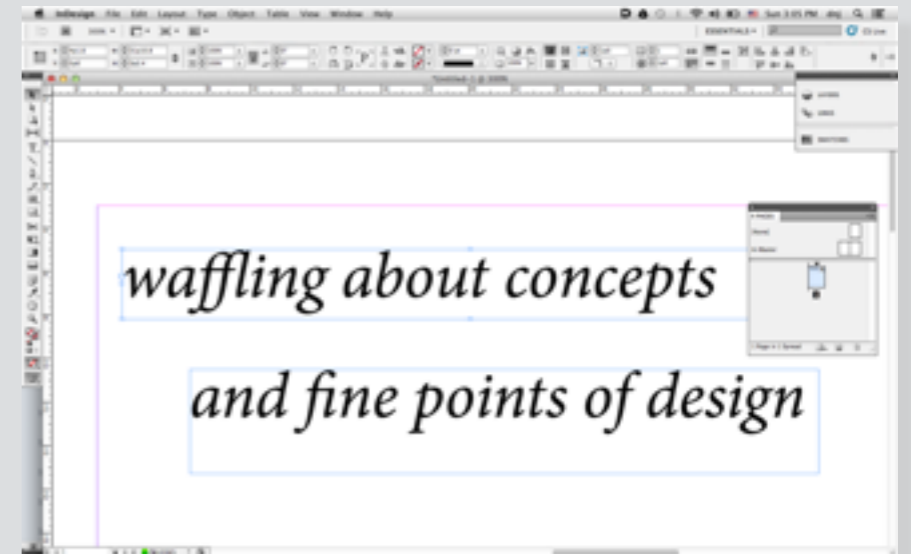


44 -- approaches, not satisfying: surely something essential-
45
46 Take a step back: concepts-
47 -- example: Microsoft Word: font and Co-
48 -- where concepts come from-
49 -- implications of concepts-
50 -- how concepts distinguish apps (Word from LaTeX) and app classes (Word from [old])
51 -- what's software made of? concepts-
52
53 Towards a theory of concepts-
54 -- what's a concept? -
55 state machine of lines? example -
56 abstract ideas, usually structural, explains behavior-
57 can explain with an operational principle without referring to purpose-
58 more complex examples: style, alias, drag, tagging, selection in Ps
59 in all cases the concept does not fulfill a function directly-
60
61 -- granularity-
62 smallest concept that's a functionality increment-
63 eg. can have layer without mask
64 but may choose to group into larger concept-
65
66 -- dependence-
67 concepts have dependencies based on functional subsets-
68 what's the point of this? -
69 -- criteria: compelling, coherent, controllable (de-coupled), general (complete, consistent),
70 familiar (conventional) -
71
72 -- one slide and a gro and an example for each? -
73
74 -- a UE notion about it being represented well? -
75 -- compelling: user facing, one motivating purpose, and an operational principle: it's about the
76 idea, not the realization -
77 -- fit for purpose: means they actually work (cf. profits) -
78 -- familiar: obvious
79 -- independent: orthogonal
80 -- coherent: unlike compelling, it's about the realization; means you can predict once you
81 understand a bit -
82 -- maybe means you expect them to fit together? -
83 definition in Mac doesn't have meta data -
84 typeface variants in Indesign don't correspond to italic and bold -
85 pixel width in CSS doesn't correspond to pixels
86 hiding in Google forms doesn't prevent publication of data -
87 conversation in Apple Mail can't predict what will happen with delete or move or many other
88 operations; too complex and brittle -
89 -- complete: with respect to other concepts, e.g. color settings -
90 -- consistent: e.g., filters vs rules in Mail; maybe subsume both UE criteria under this -
91 -- analyze Acrobat, in S, concepts aren't visible in UI, they become visible, but they're not
92 general, so they're not understandable or familiar -
93
94

text editor
line buffer



word processor
paragraph
format
style

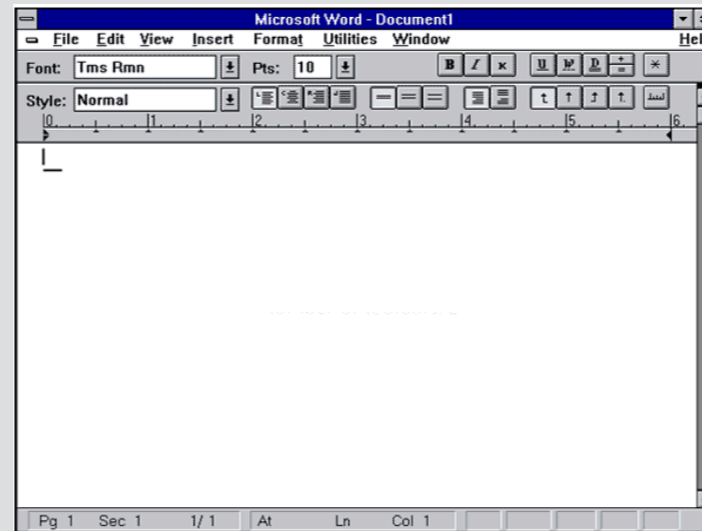


desktop publishing app

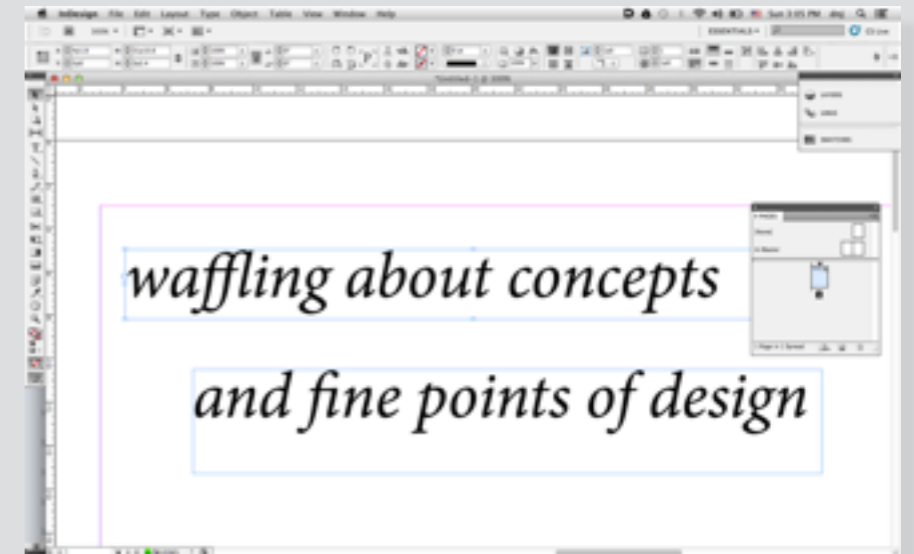
app classes characterized by concepts too



text editor
line buffer



word processor
paragraph
format
style

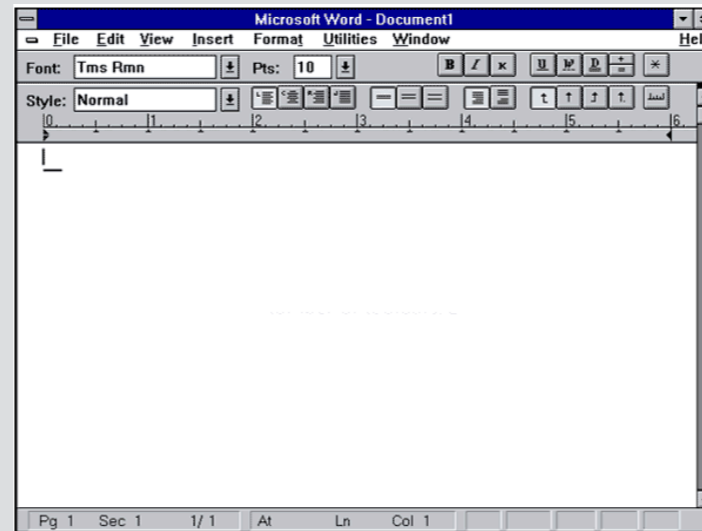


desktop publishing app
stylesheet
text flow
page template

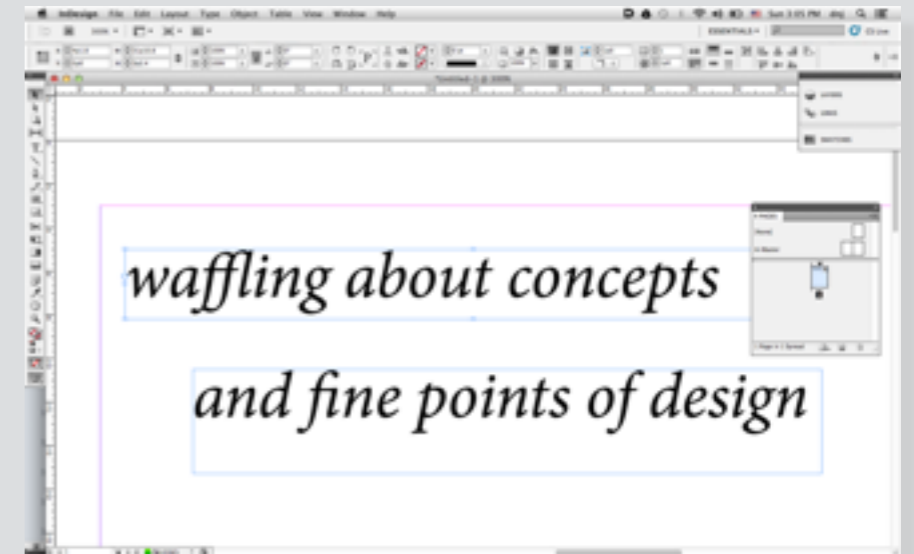
app classes characterized by concepts too



text editor
line buffer



word processor
paragraph
format
style



desktop publishing app
stylesheet
text flow
page template

jamonh
Oct 22, 2013 7:19 PM

Just upgraded to the new Pages and can't find a way to link text boxes anymore like <http://www.macobserver.com/tmo/article/pages-linking-text-boxes>

Am I missing something, or is it really not possible anymore?

the conventional view

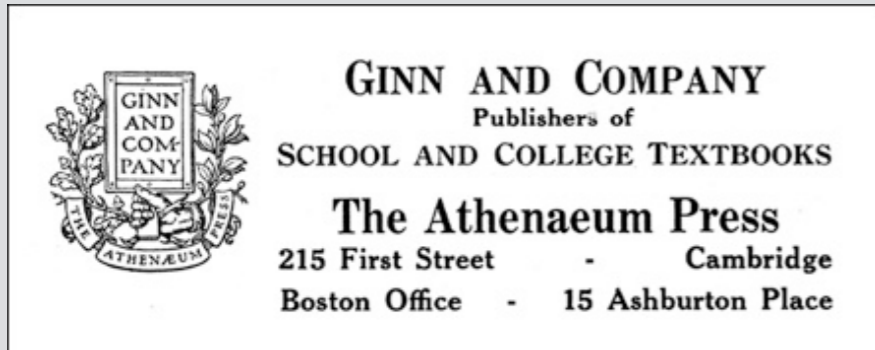
Conceptual model (computer science)

From Wikipedia, the free encyclopedia

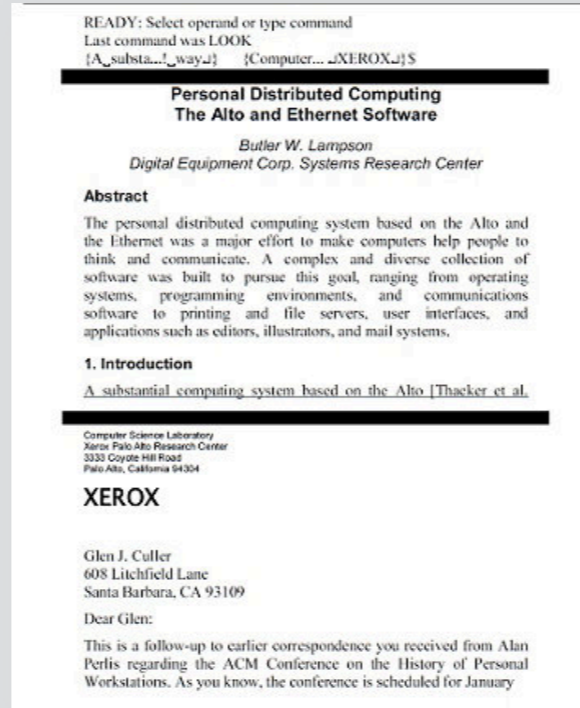
A [mental model](#) captures ideas in a [problem domain](#), while a **conceptual model** represents 'concepts' (entities) and relationships between them.

A conceptual model in the field of [computer science](#) is a special case of a general [conceptual model](#). To distinguish from other types of models, it is also known as a [domain model](#). Conceptual modeling should not be confused with other modeling disciplines such as [data modelling](#), [logical modelling](#) and

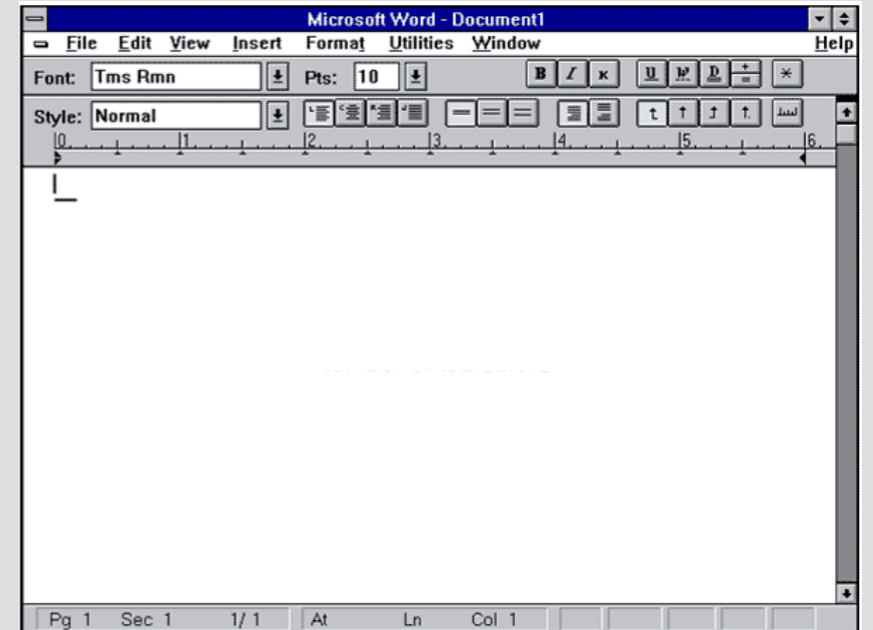
concepts are invented, not just out there



Tim Mott visits Ginn in 1974
brings idea of styles to PARC



Charles Simonyi's team
implements style in
Bravo text editor



Simonyi brings style
to Microsoft in 1983

the rewards of inventing a good concept

the rewards of inventing a good concept



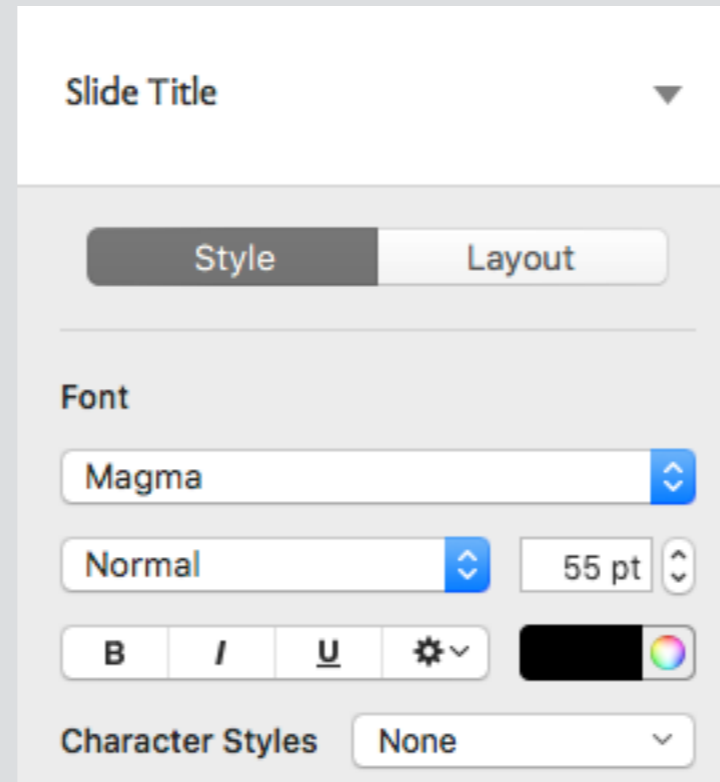
the rewards of inventing a good concept



who is this and what is he doing?

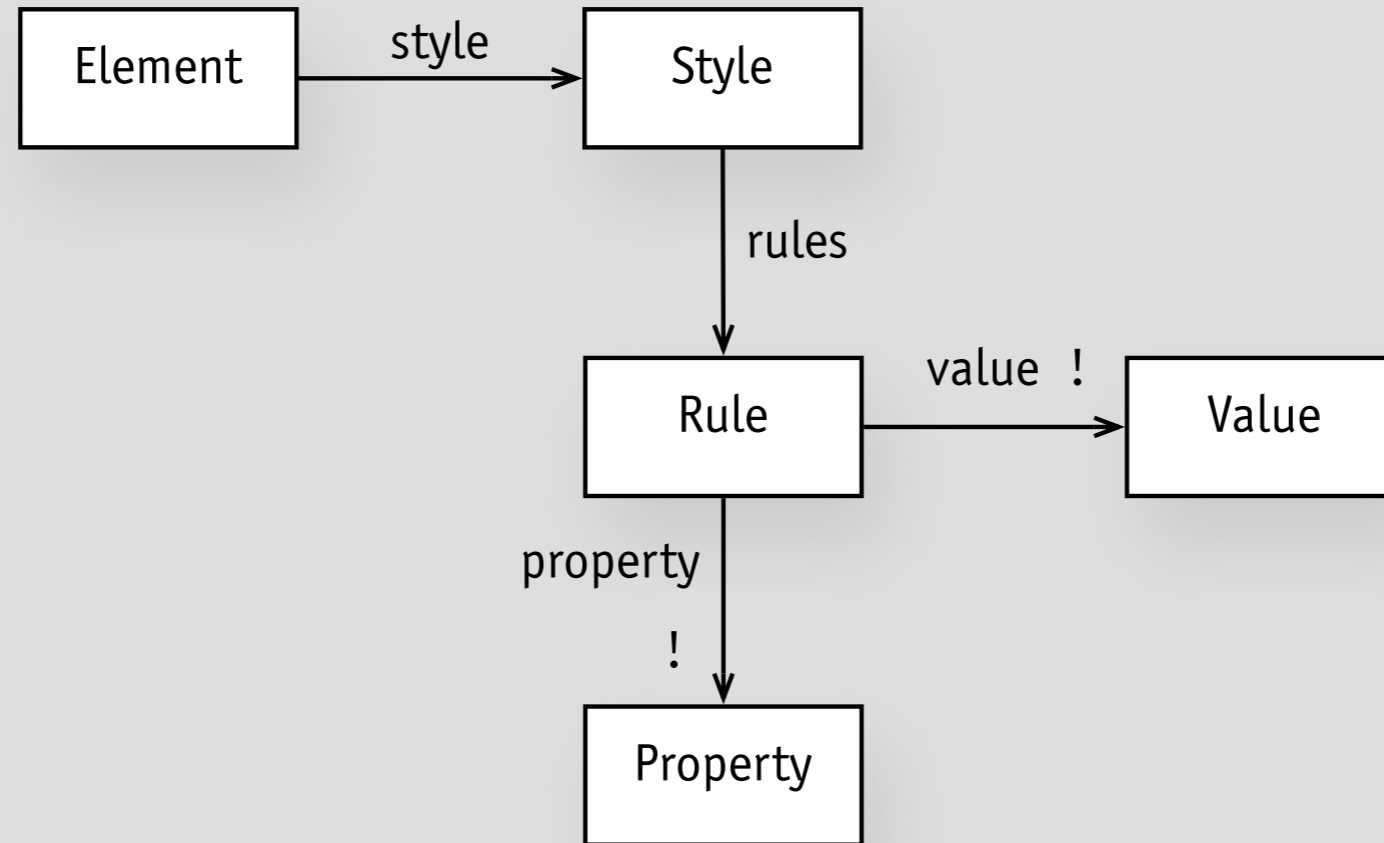
concepts have purpose

purpose of **style**: enable consistent formatting

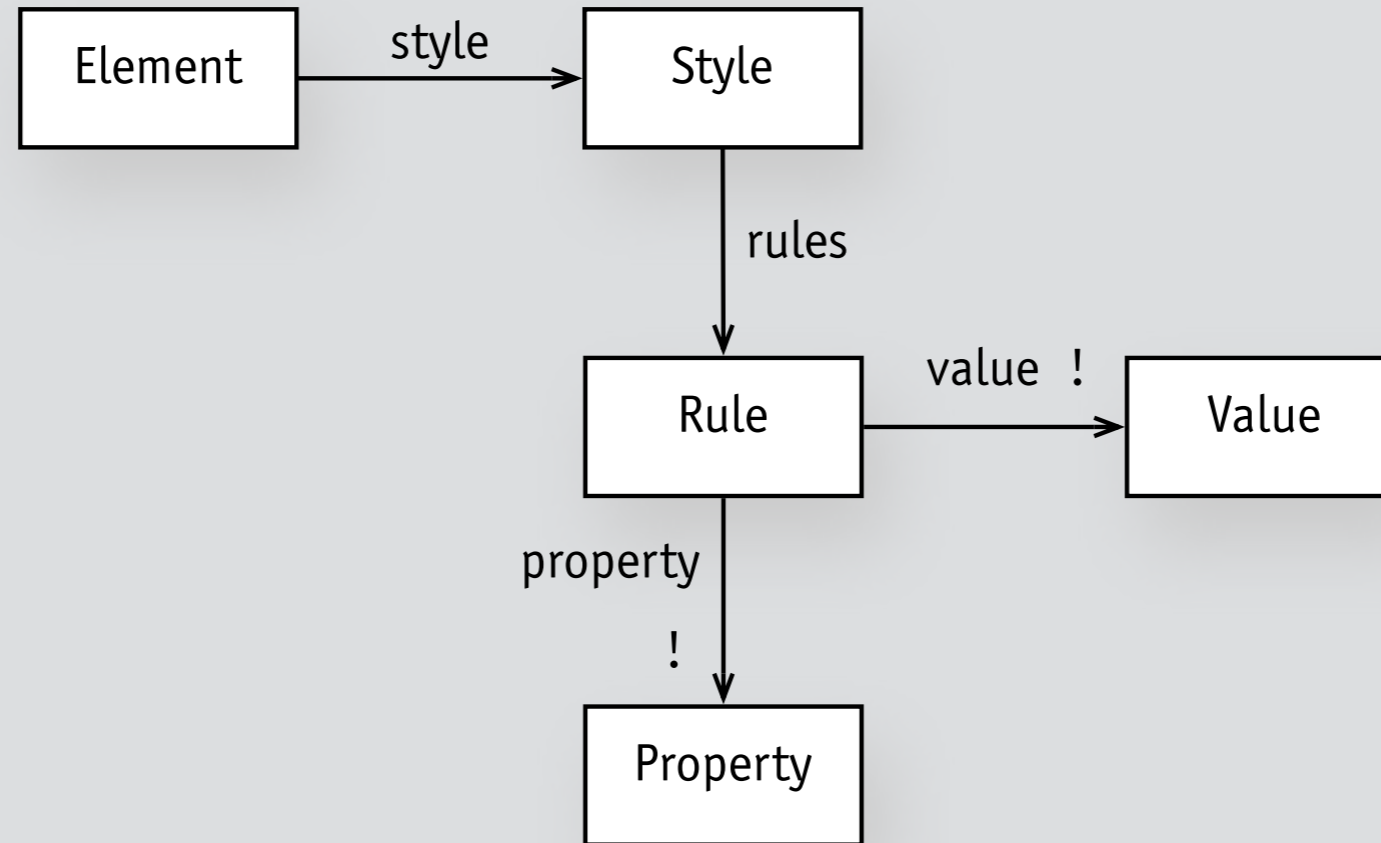


Apple Keynote
adds style concept
(2017?)

concept structure is designed not discovered

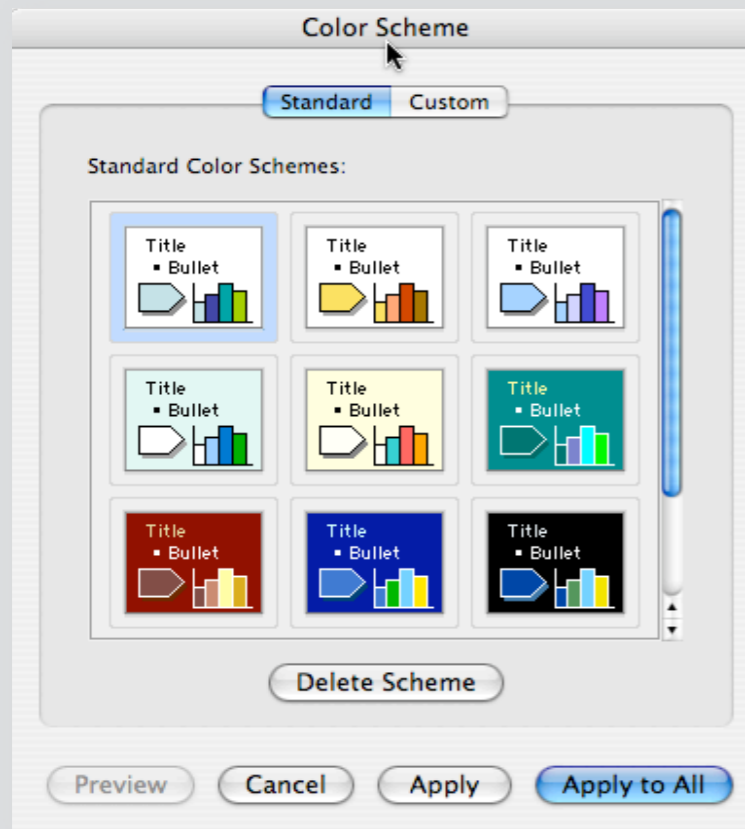


concept structure is designed not discovered

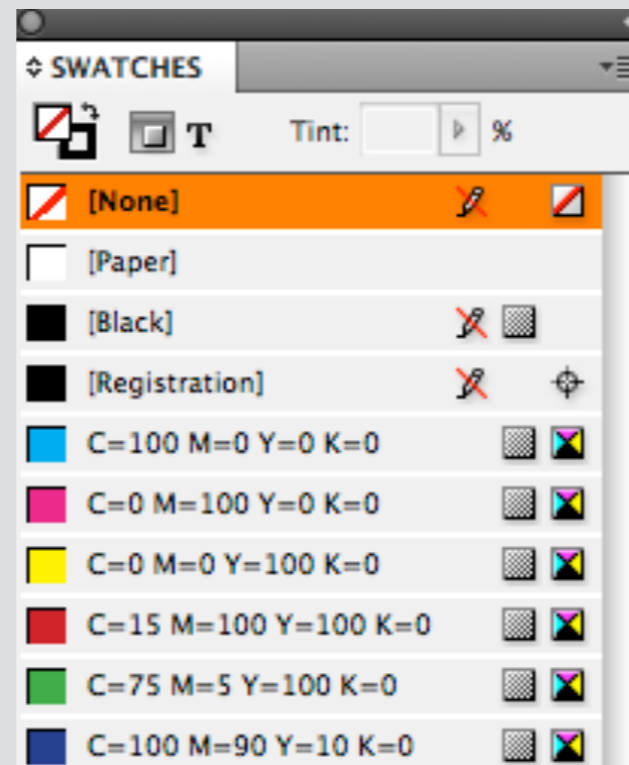


There is no problem
in computer science
that cannot be solved
by introducing another
level of indirection.
David Wheeler

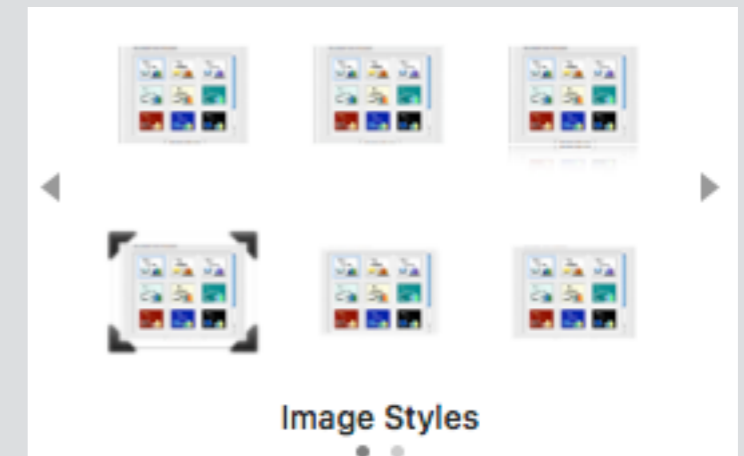
concepts are reusable



Powerpoint color schemes

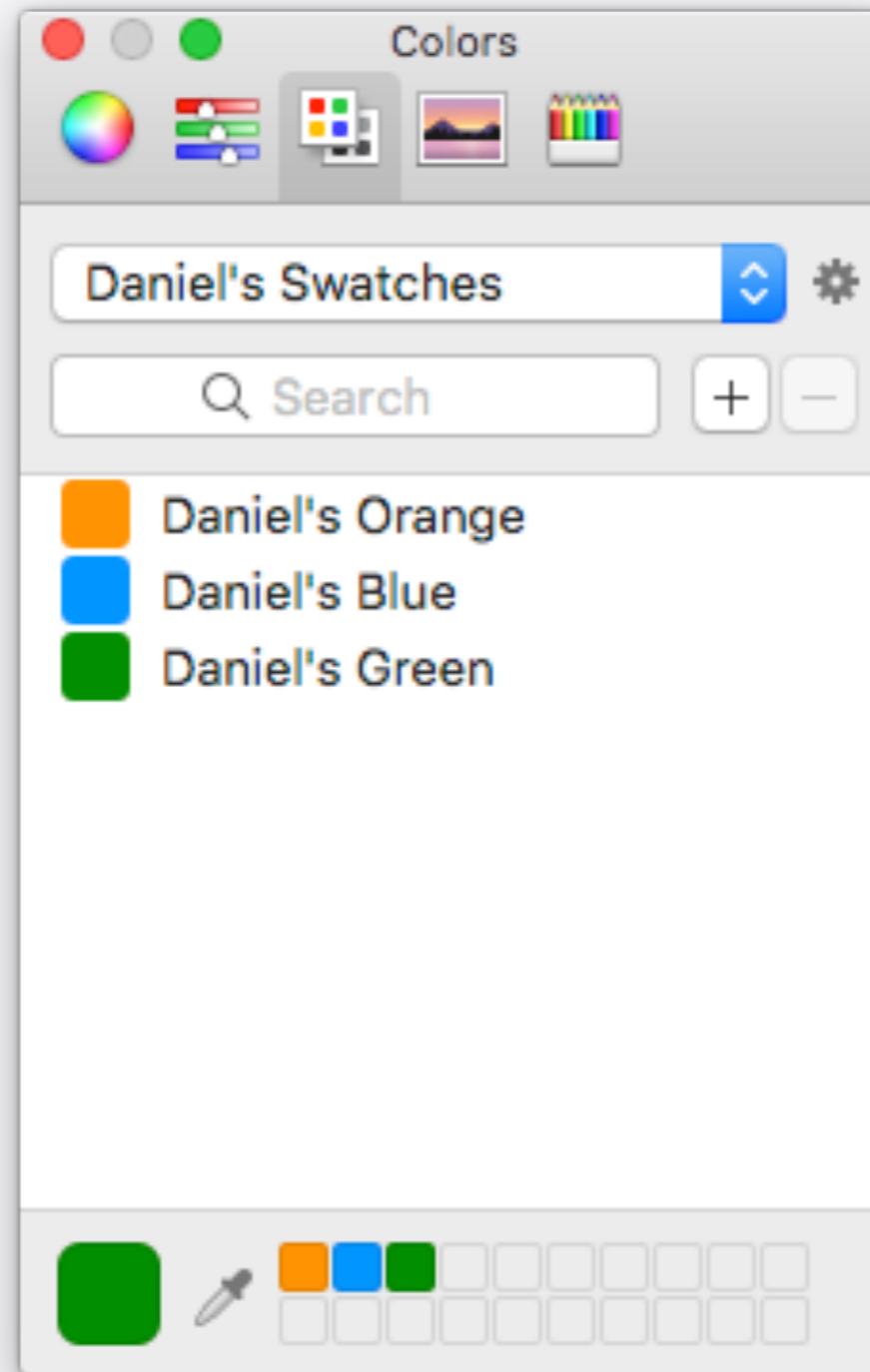


Indesign swatches

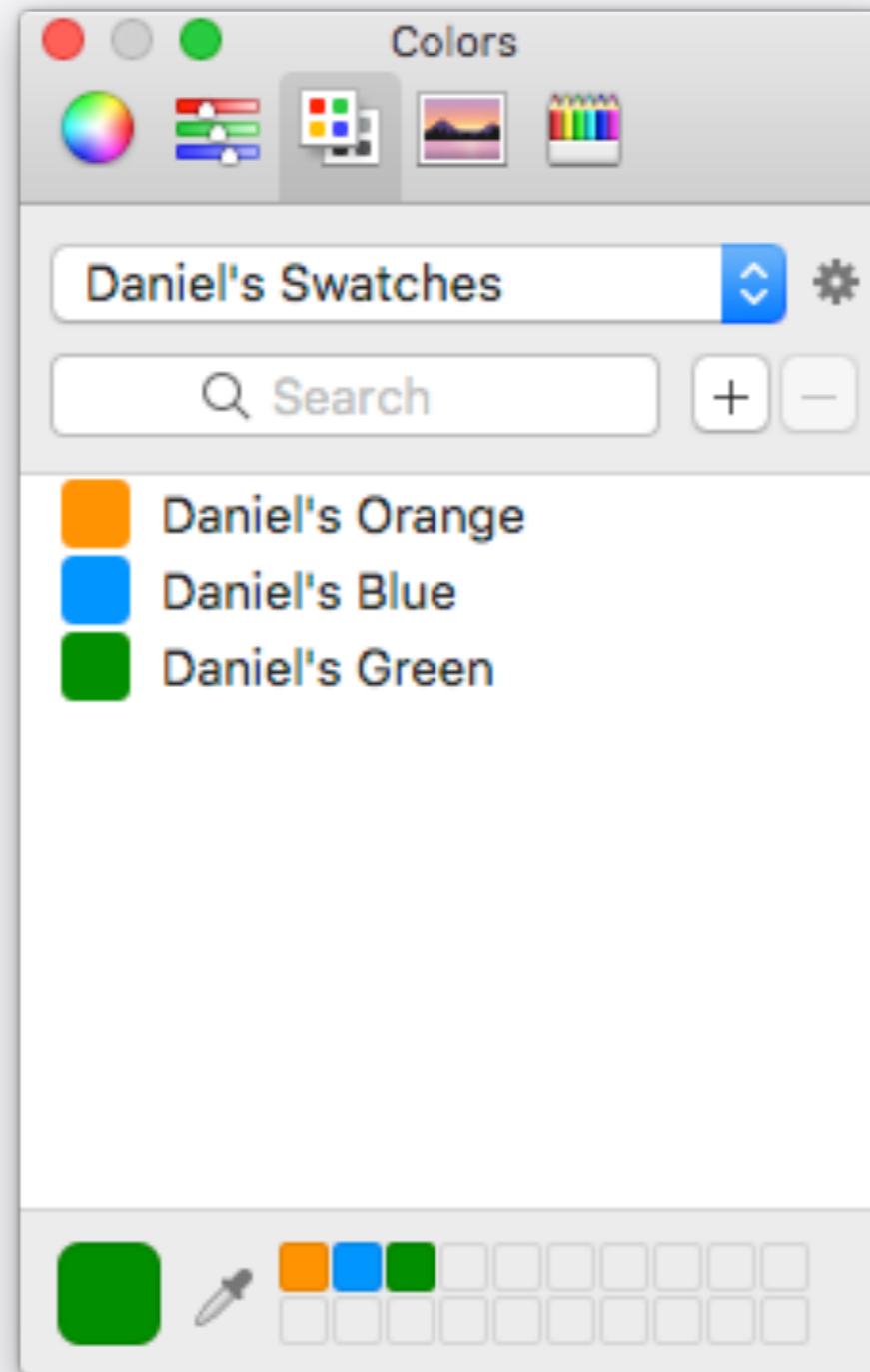


Keynote image styles

not an instance of style



not an instance of style



what crucial action is missing?

explaining concepts

how to explain the style concept?

If you assign Heading to two paragraphs
and then you change the style from bold to italic,
both paragraphs will be changed in concert

explaining concepts

how to explain the style concept?

If you assign Heading to two paragraphs and then you change the style from bold to italic, both paragraphs will be changed in concert

If you create a style and assign to two elements, then when you modify the style, both elements will change...

explaining concepts

how to explain the style concept?

If you assign Heading to two paragraphs and then you change the style from bold to italic, both paragraphs will be changed in concert

If you create a style and assign to two elements, then when you modify the style, both elements will change...

not ontological: "a style is a mapping..."

explaining concepts

how to explain the style concept?

If you assign Heading to two paragraphs and then you change the style from bold to italic, both paragraphs will be changed in concert

If you create a style and assign to two elements, then when you modify the style, both elements will change...

not ontological: "a style is a mapping..."



Johnson-Laird:
constructive semantics

explaining concepts

how to explain the style concept?

If you assign Heading to two paragraphs and then you change the style from bold to italic, both paragraphs will be changed in concert

If you create a style and assign to two elements, then when you modify the style, both elements will change...

not ontological: "a style is a mapping..."



Johnson-Laird:
constructive semantics



Michael Polanyi
operational principle

explaining concepts

how to explain the style concept?

If you assign Heading to two paragraphs and then you change the style from bold to italic, both paragraphs will be changed in concert

If you create a style and assign to two elements, then when you modify the style, both elements will change...

not ontological: "a style is a mapping..."



Johnson-Laird:
constructive semantics



Michael Polanyi
operational principle

if you have a full spec of the behavior of a concept, is the tactic redundant?

explaining concepts

how to explain the style concept?

If you assign Heading to two paragraphs and then you change the style from bold to italic, both paragraphs will be changed in concert

If you create a style and assign to two elements, then when you modify the style, both elements will change...

not ontological: "a style is a mapping..."

not redundant: unlike full spec, shows how concept meets purpose



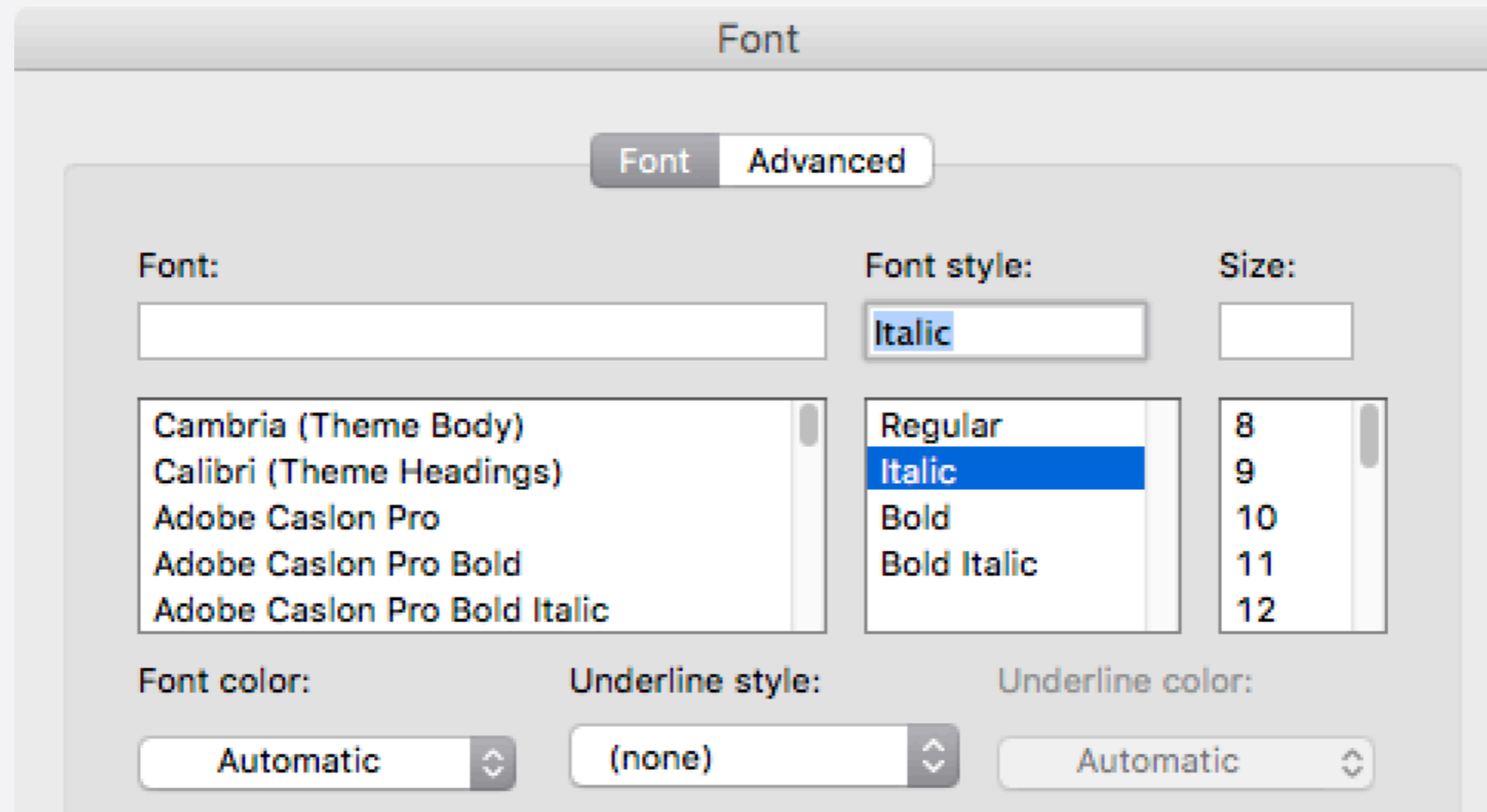
Johnson-Laird:
constructive semantics



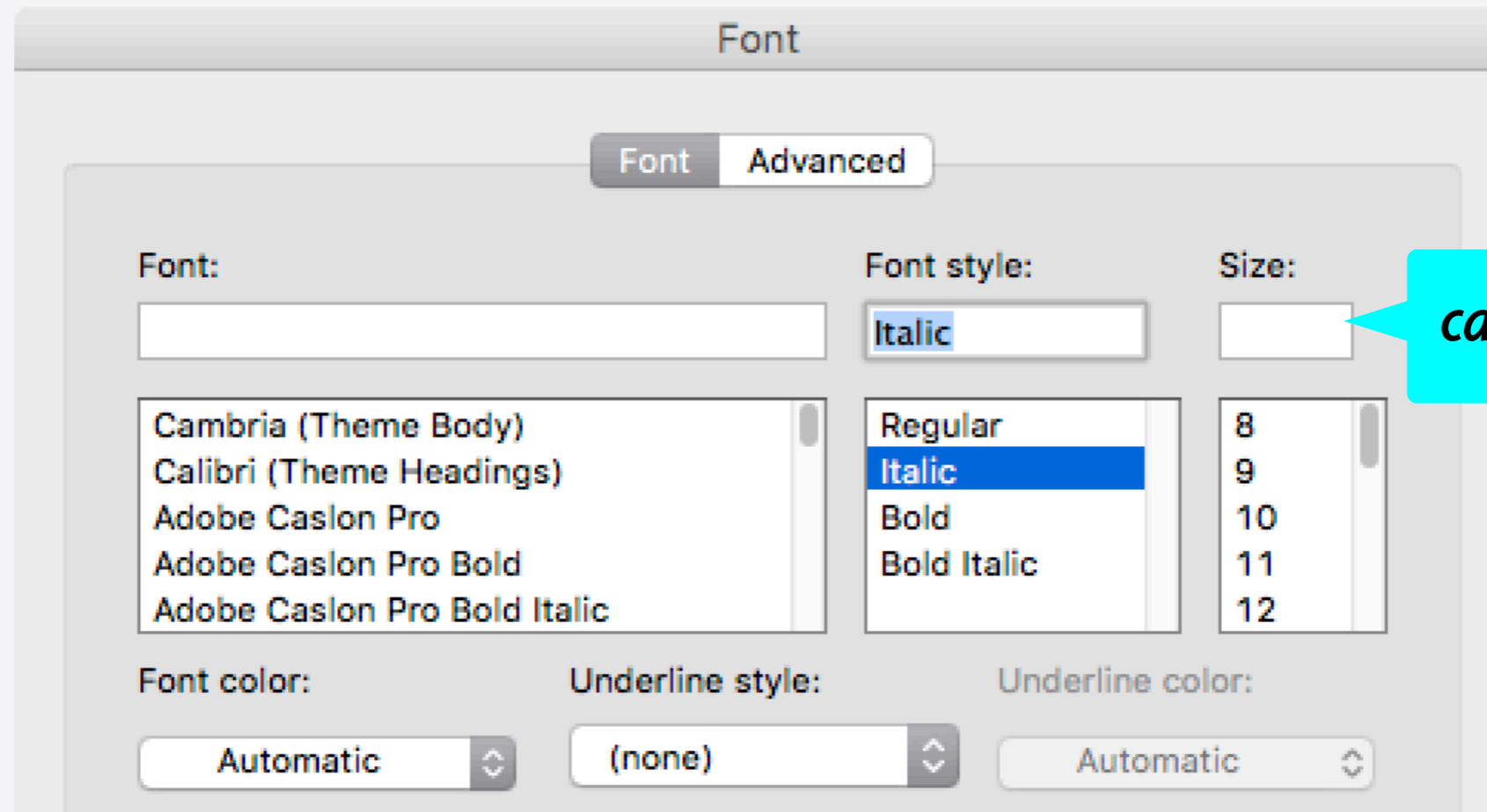
Michael Polanyi
operational principle

if you have a full spec of the behavior of a concept, is the tactic redundant?

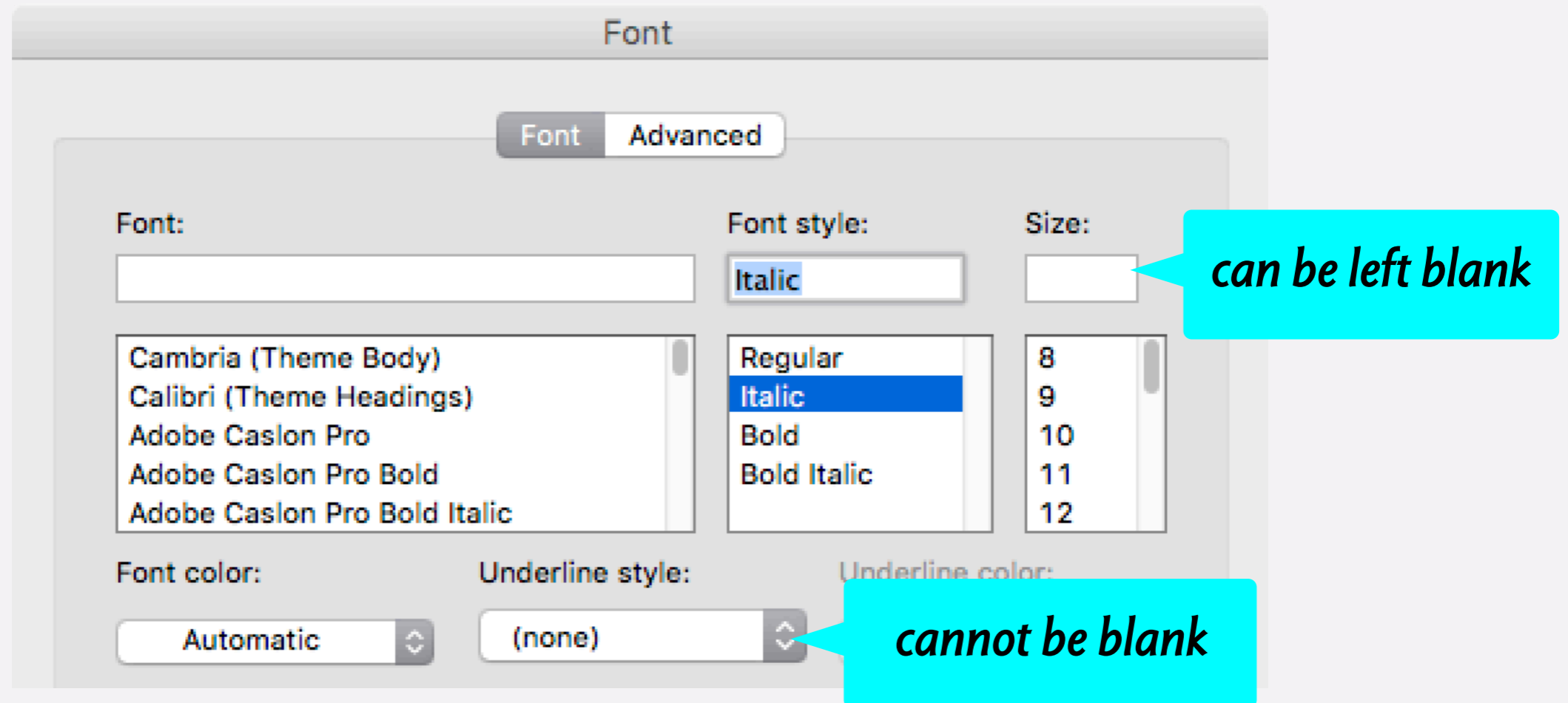
when concepts don't fulfill purpose



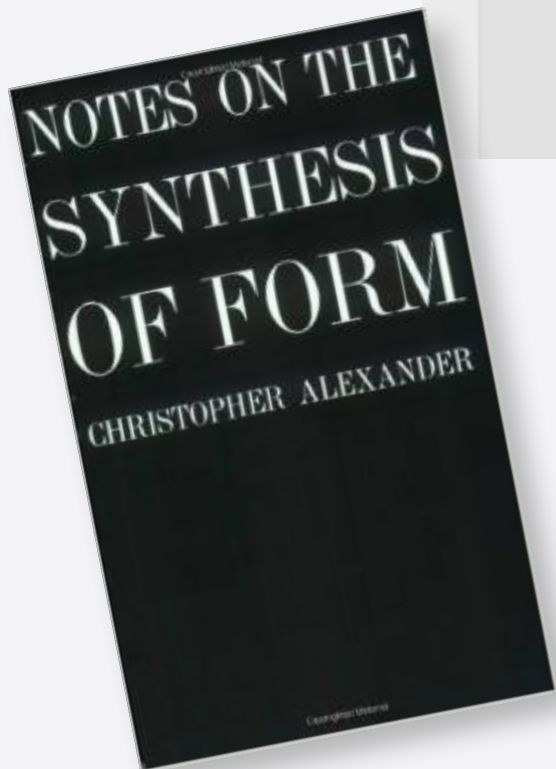
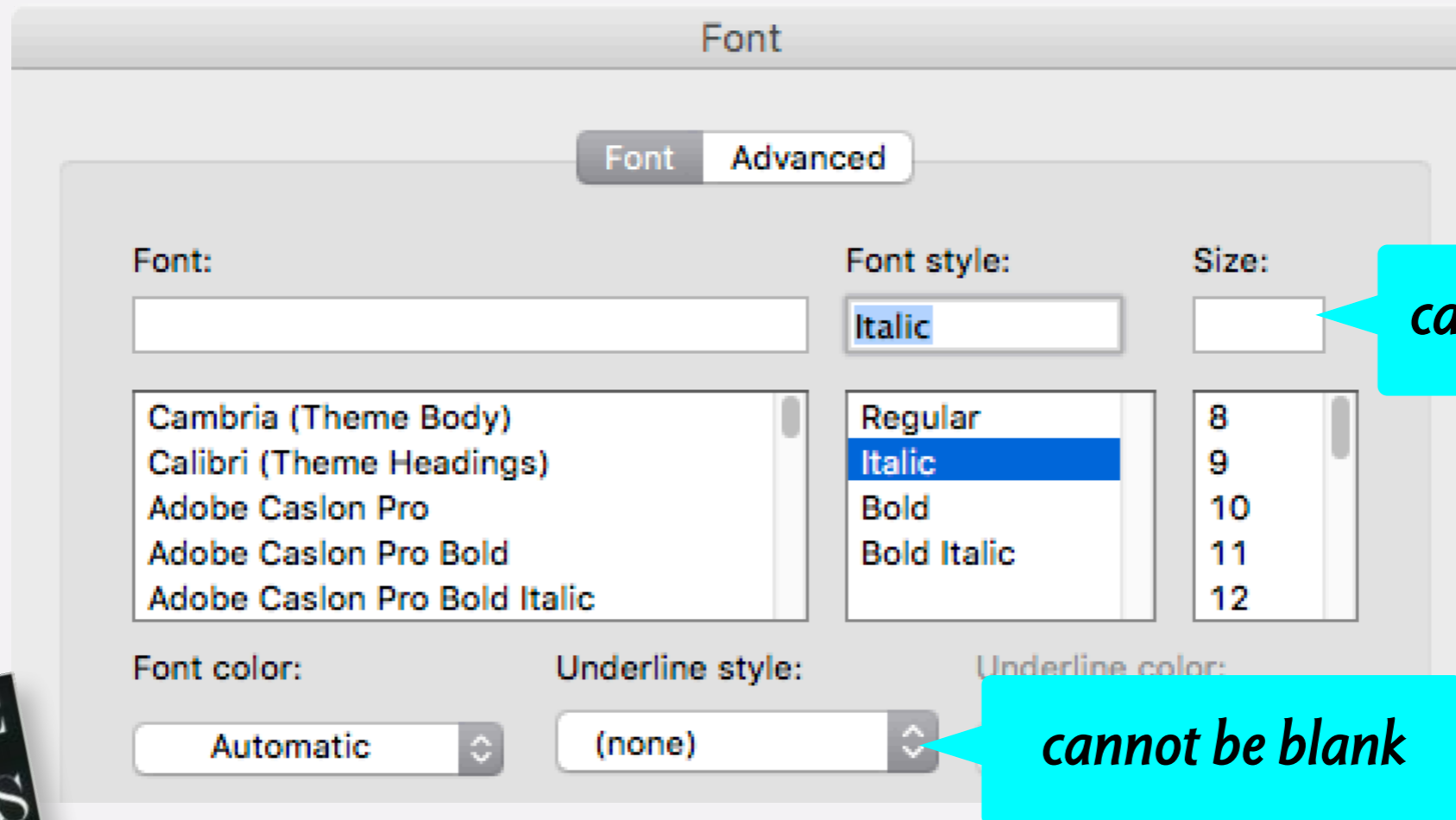
when concepts don't fulfill purpose



when concepts don't fulfill purpose



when concepts don't fulfill purpose



Alexander's **misfits**:
not bugs but bad specs

concepts: modules of behavior

concepts: modules of behavior

concepts: modules of behavior



inventive

concepts: modules of behavior



inventive



purposeful

concepts: modules of behavior



inventive



purposeful



behavioral

concepts: modules of behavior



inventive



purposeful



behavioral



self-contained

concepts: modules of behavior



inventive



purposeful



behavioral



self-contained



reusable

concept
models

a reservation concept

a reservation concept

name

reservation

a reservation concept

name

reservation

purpose

make access to shared resource reliable

a reservation concept

name

reservation

purpose

make access to shared resource reliable

structure

slots: Owner -> Slot
holds: User -> Slot

a reservation concept

name

reservation

purpose

make access to shared resource reliable

structure

slots: Owner -> Slot
holds: User -> Slot

behavior

```
create (o: Owner, s: Slot)
  no slots.s => slots += o -> s

reserve (u: User, o: Owner, s: Slot)
  no holds.s and o -> s in slots => holds += u -> s

cancel (u: User, s: Slot)
  u -> s in holds => holds -= u -> s

use (u: User, o: Owner, s: Slot)
  u -> s in holds and o -> s in slots =>
```

a reservation concept

name

reservation

purpose

make access to shared resource reliable

structure

slots: Owner -> Slot
holds: User -> Slot

behavior

```
create (o: Owner, s: Slot)
  no slots.s => slots += o -> s

reserve (u: User, o: Owner, s: Slot)
  no holds.s and o -> s in slots => holds += u -> s

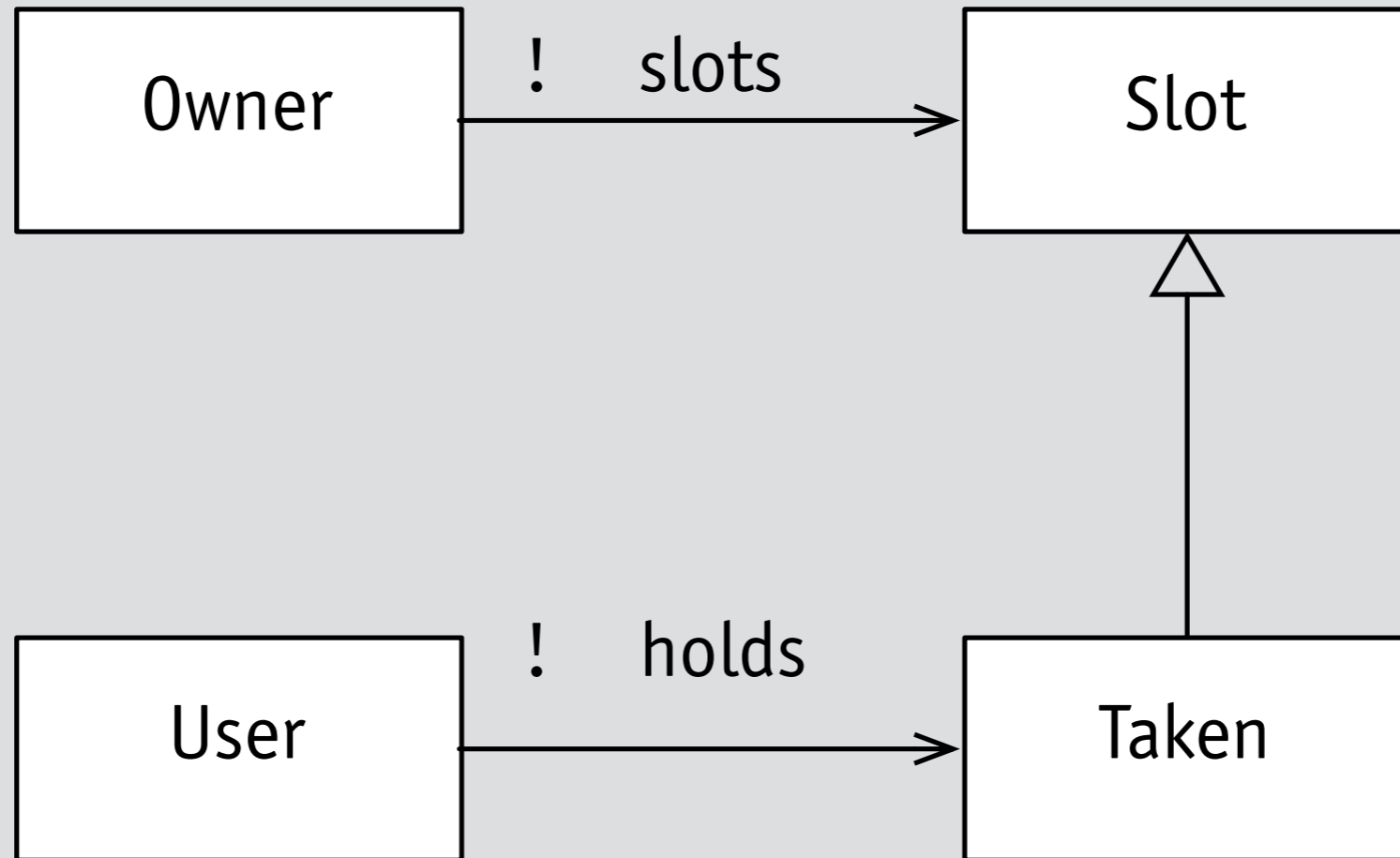
cancel (u: User, s: Slot)
  u -> s in holds => holds -= u -> s

use (u: User, o: Owner, s: Slot)
  u -> s in holds and o -> s in slots =>
```

tactic

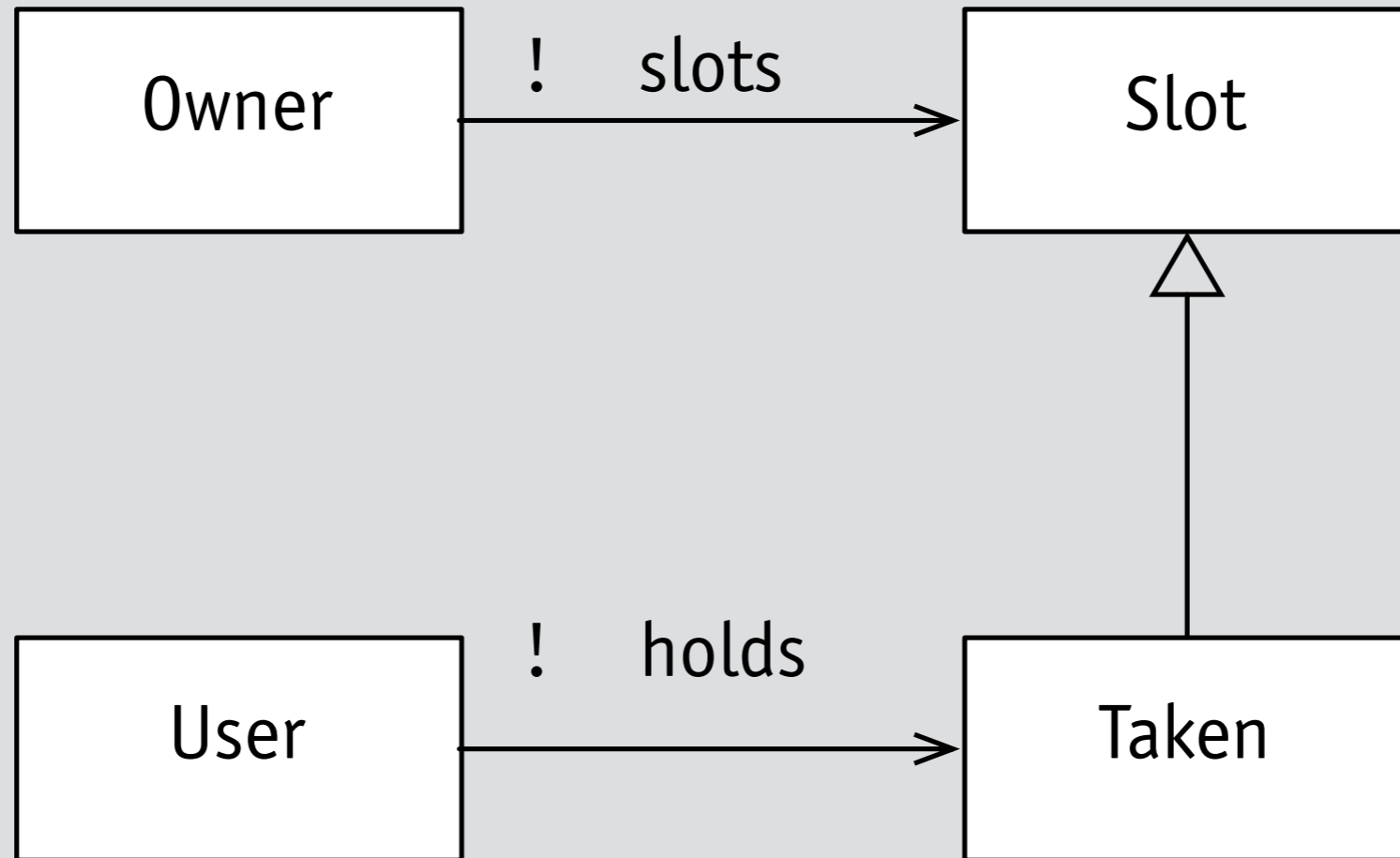
if create(o,s); reserve(u,o,s); ... **no** cancel(u,s) ... **then can** use(u,o,s)

a relational diagram

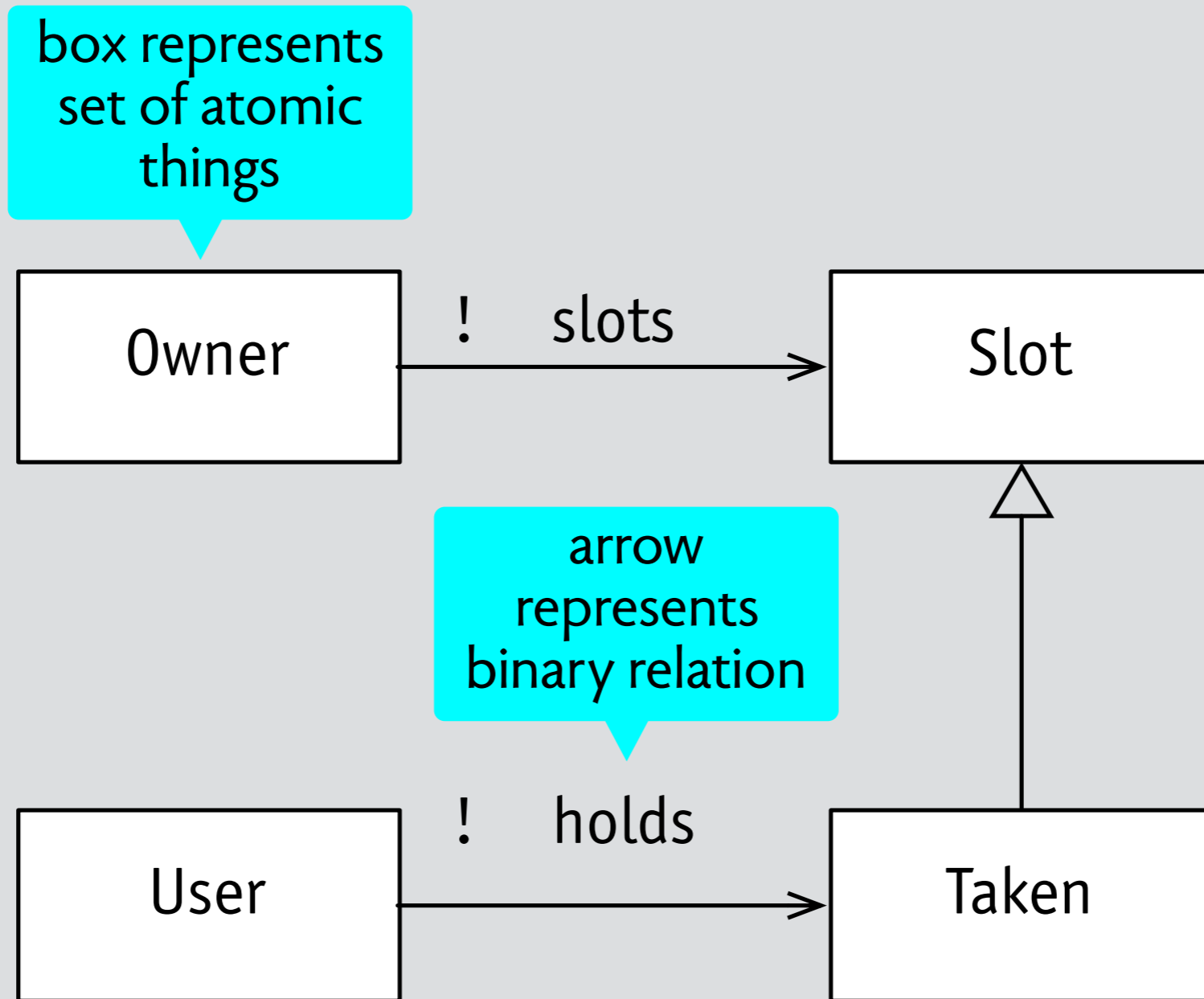


a relational diagram

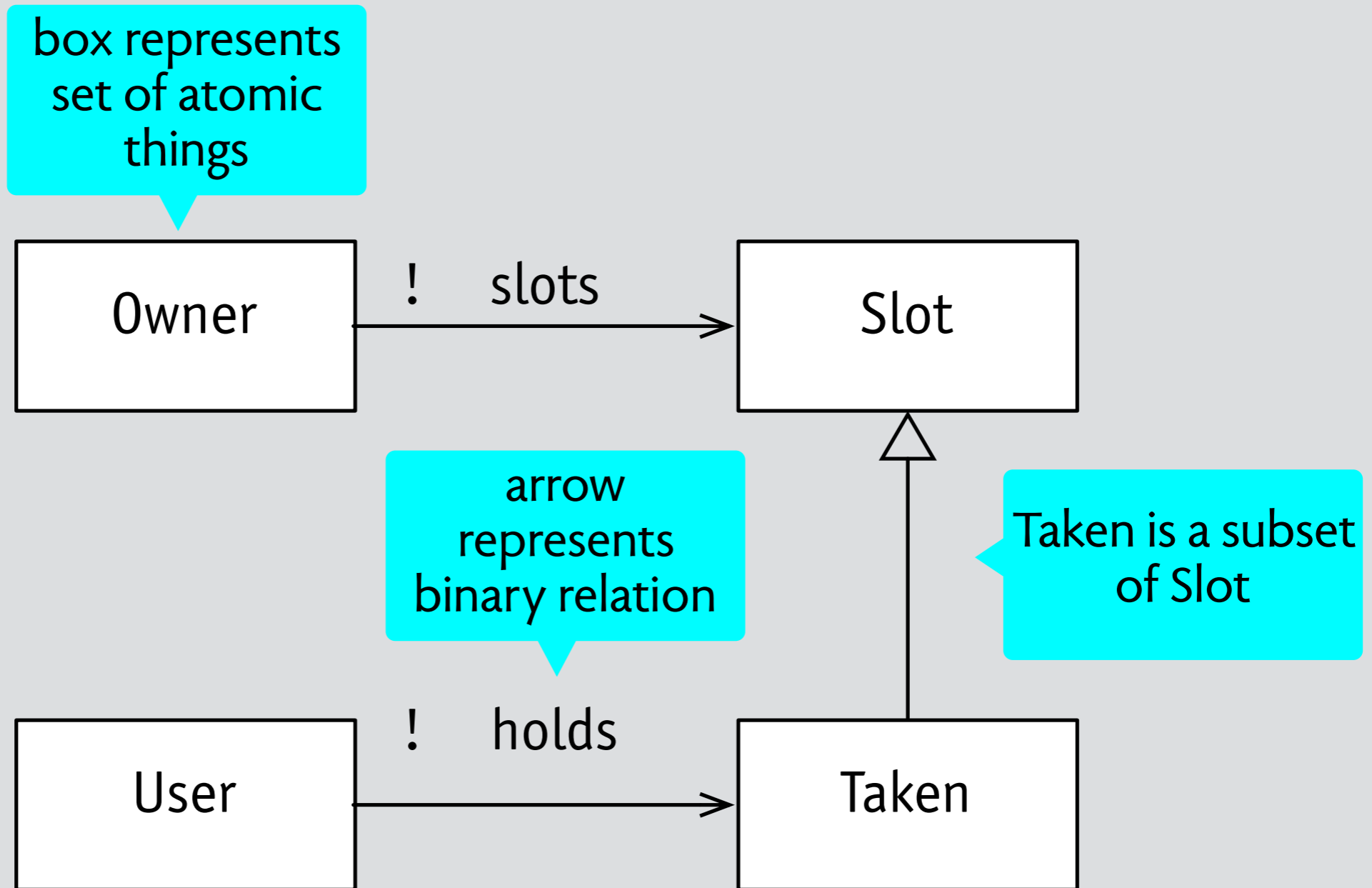
box represents
set of atomic
things



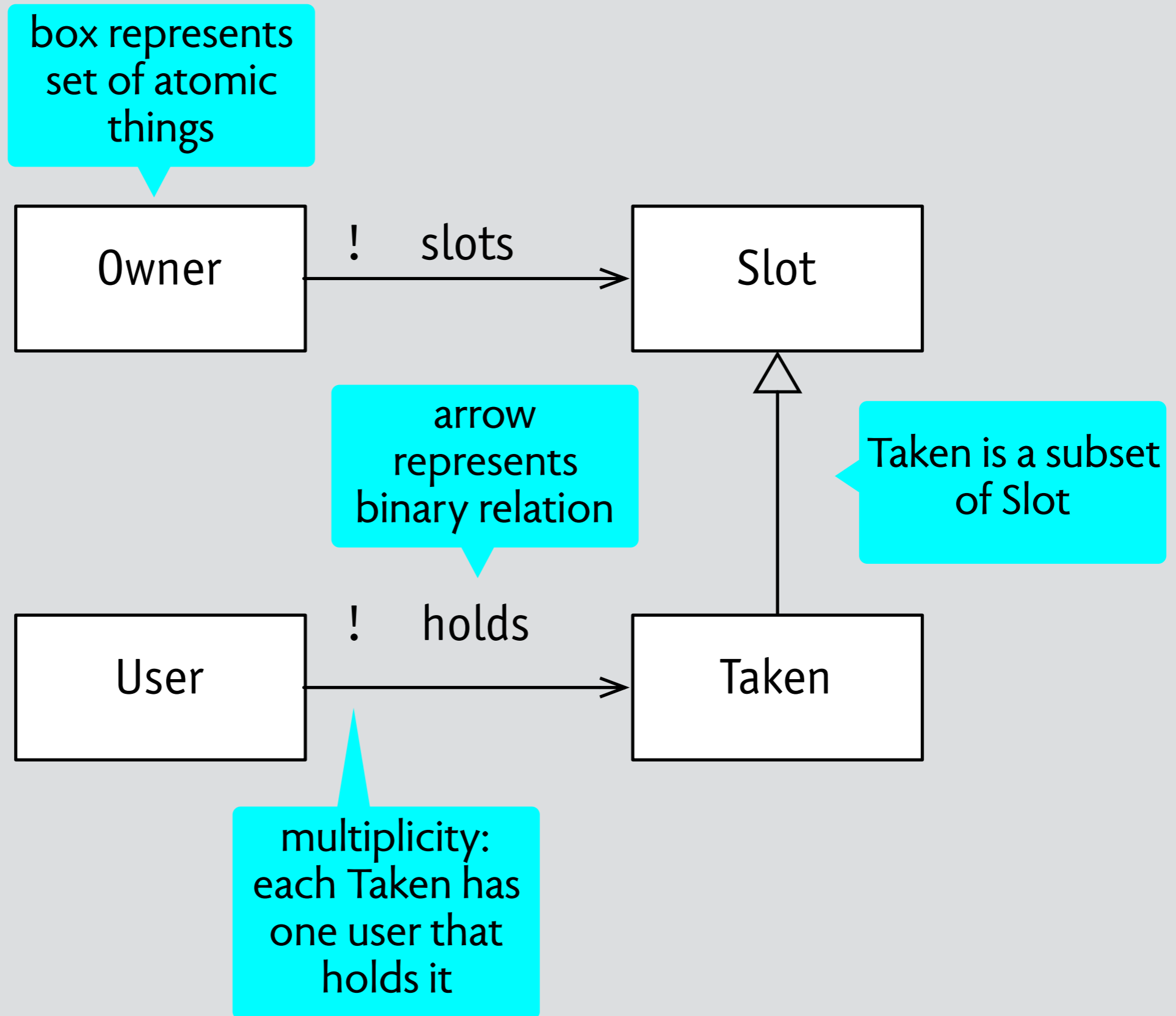
a relational diagram



a relational diagram



a relational diagram



alloy expressions in one slide

alloy expressions in one slide

alloy expressions in one slide

u: User

s: Slot

holds: User -> Slot

Taken: set Slot

alloy expressions in one slide

u: User

s: Slot

holds: User -> Slot

Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}

holds' = {(u1,s1), (u1,s2), (u2,s3)}

alloy expressions in one slide

u: User

s: Slot

holds: User -> Slot

Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}

holds' = {(u1,s1), (u1,s2), (u2,s3)}

a set is a relation with one column

Slot = {(s1), (s2), (s3), (s4)}

Taken = {(s1), (s2), (s3)}

alloy expressions in one slide

u: User

s: Slot

holds: User -> Slot

Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}

holds' = {(u1,s1), (u1,s2), (u2,s3)}

a set is a relation with one column

Slot = {(s1), (s2), (s3), (s4)}

Taken = {(s1), (s2), (s3)}

a scalar is a set with one row

u = {(u2)}

s = {(s3)}

alloy expressions in one slide

u: User

s: Slot

holds: User -> Slot

Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}

holds' = {(u1,s1), (u1,s2), (u2,s3)}

a set is a relation with one column

Slot = {(s1), (s2), (s3), (s4)}

Taken = {(s1), (s2), (s3)}

a scalar is a set with one row

u = {(u2)}

s = {(s3)}

set operators

+ union, - difference, & intersection, in subset

alloy expressions in one slide

u: User
s: Slot
holds: User -> Slot
Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}

holds' = {(u1,s1), (u1,s2), (u2,s3)}

a set is a relation with one column

Slot = {(s1), (s2), (s3), (s4)}

Taken = {(s1), (s2), (s3)}

a scalar is a set with one row

u = {(u2)}

s = {(s3)}

set operators

+ union, - difference, & intersection, in subset

Slot - Taken = {(s4)}

holds' - holds = {(u2,s3)}

alloy expressions in one slide

u: User
s: Slot
holds: User -> Slot
Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}
holds' = {(u1,s1), (u1,s2), (u2,s3)}

a set is a relation with one column

Slot = {(s1), (s2), (s3), (s4)}
Taken = {(s1), (s2), (s3)}

a scalar is a set with one row

u = {(u2)}
s = {(s3)}

set operators

+ union, - difference, & intersection, in subset

Slot - Taken = {(s4)}
holds' - holds = {(u2,s3)}

relation operators

-> product
• join

alloy expressions in one slide

u: User
s: Slot
holds: User -> Slot
Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}
holds' = {(u1,s1), (u1,s2), (u2,s3)}

a set is a relation with one column

Slot = {(s1), (s2), (s3), (s4)}
Taken = {(s1), (s2), (s3)}

a scalar is a set with one row

u = {(u2)}
s = {(s3)}

set operators

+ union, - difference, & intersection, in subset

Slot - Taken = {(s4)}
holds' - holds = {(u2,s3)}

relation operators

-> product
.join

$a \rightarrow b = \{ (a_0, \dots, a_n, b_0, \dots, b_m) \mid (a_0, \dots, a_n) \in a \wedge (b_0, \dots, b_m) \in b \}$

alloy expressions in one slide

u: User
s: Slot
holds: User -> Slot
Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}
holds' = {(u1,s1), (u1,s2), (u2,s3)}

a set is a relation with one column

Slot = {(s1), (s2), (s3), (s4)}
Taken = {(s1), (s2), (s3)}

a scalar is a set with one row

u = {(u2)}
s = {(s3)}

set operators

+ union, - difference, & intersection, in subset

Slot - Taken = {(s4)}
holds' - holds = {(u2,s3)}

relation operators

-> product
.join

product examples

u -> s = {(u2,s3)}
u -> Taken = {(u2,s1), (u2,s2), (u2,s3)}

$a \rightarrow b = \{ (a_0, \dots, a_n, b_0, \dots, b_m) \mid (a_0, \dots, a_n) \in a \wedge (b_0, \dots, b_m) \in b \}$

alloy expressions in one slide

u: User
s: Slot
holds: User -> Slot
Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}
holds' = {(u1,s1), (u1,s2), (u2,s3)}

a set is a relation with one column

Slot = {(s1), (s2), (s3), (s4)}
Taken = {(s1), (s2), (s3)}

a scalar is a set with one row

u = {(u2)}
s = {(s3)}

set operators

+ union, - difference, & intersection, in subset

Slot - Taken = {(s4)}
holds' - holds = {(u2,s3)}

relation operators

-> product
.join

product examples

u -> s = {(u2,s3)}
u -> Taken = {(u2,s1), (u2,s2), (u2,s3)}

$a \rightarrow b = \{ (a_0, \dots, a_n, b_0, \dots, b_m) \mid (a_0, \dots, a_n) \in a \wedge (b_0, \dots, b_m) \in b \}$

$a.b = \{ (a_0, \dots, a_{n-1}, b_1, \dots, b_m) \mid (a_0, \dots, a_n) \in a \wedge (a_n, b_1, \dots, b_m) \in b \}$

alloy expressions in one slide

u: User
s: Slot
holds: User -> Slot
Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}
holds' = {(u1,s1), (u1,s2), (u2,s3)}

a set is a relation with one column

Slot = {(s1), (s2), (s3), (s4)}
Taken = {(s1), (s2), (s3)}

a scalar is a set with one row

u = {(u2)}
s = {(s3)}

set operators

+ union, - difference, & intersection, in subset

Slot - Taken = {(s4)}
holds' - holds = {(u2,s3)}

relation operators

-> product
.join

product examples

u -> s = {(u2,s3)}
u -> Taken = {(u2,s1), (u2,s2), (u2,s3)}

join examples

u.holds' = {(s3)}
holds'.s = {(u2)}
holds.Slot = {(u1)}

$a \rightarrow b = \{ (a_0, \dots, a_n, b_0, \dots, b_m) \mid (a_0, \dots, a_n) \in a \wedge (b_0, \dots, b_m) \in b \}$

$a.b = \{ (a_0, \dots, a_{n-1}, b_1, \dots, b_m) \mid (a_0, \dots, a_n) \in a \wedge (a_n, b_1, \dots, b_m) \in b \}$

alloy expressions in one slide

u: User
s: Slot
holds: User -> Slot
Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}
holds' = {(u1,s1), (u1,s2), (u2,s3)}

a set is a relation with one column

Slot = {(s1), (s2), (s3), (s4)}
Taken = {(s1), (s2), (s3)}

a scalar is a set with one row

u = {(u2)}
s = {(s3)}

set operators

+ union, - difference, & intersection, in subset

Slot - Taken = {(s4)}
holds' - holds = {(u2,s3)}

relation operators

-> product
.join

product examples

u -> s = {(u2,s3)}
u -> Taken = {(u2,s1), (u2,s2), (u2,s3)}

join examples

u.holds' = {(s3)}
holds'.s = {(u2)}
holds.Slot = {(u1)}

formula examples

holds' = holds + u -> s
(also written holds += u -> s
User.holds = Taken
holds in User -> Slot

$a \rightarrow b = \{ (a_0, \dots, a_n, b_0, \dots, b_m) \mid (a_0, \dots, a_n) \in a \wedge (b_0, \dots, b_m) \in b \}$

$a.b = \{ (a_0, \dots, a_{n-1}, b_1, \dots, b_m) \mid (a_0, \dots, a_n) \in a \wedge (a_n, b_1, \dots, b_m) \in b \}$

a reservation concept

name

reservation

purpose

make access to shared resource reliable

structure

slots: Owner -> Slot
holds: User -> Slot

behavior

```
create (o: Owner, s: Slot)
  no slots.s => slots += o -> s

reserve (u: User, o: Owner, s: Slot)
  no holds.s and o -> s in slots => holds += u -> s

cancel (u: User, s: Slot)
  u -> s in holds => holds -= u -> s

use (u: User, o: Owner, s: Slot)
  u -> s in holds and o -> s in slots =>
```

tactic

if create(o,s); reserve(u,o,s); ... **no** cancel(u,s) ... **then can** use(u,o,s)

checking a tactic with electrom

```
New Open Reload Save Execute Show
sig Slot {}
sig Owner {var slots: set Slot}
sig User {var holds: set Slot}

pred create [o: Owner, s: Slot] {
  no slots.s
  slots' = slots + o -> s
  holds' = holds
}

pred reserve [u: User, o: Owner, s: Slot] {
  no holds.s
  o -> s in slots
  holds' = holds + u -> s
  slots' = slots
}

pred cancel [u: User, s: Slot] {
  u -> s in holds
  holds' = holds - u -> s
  slots' = slots
}

pred can_use [u: User, o: Owner, s: Slot] {
  u -> s in holds and o -> s in slots
}

pred skip {slots' = slots and holds' = holds}
```

see: <https://github.com/haslab/Electrum>

checking a tactic with electrom

```
New Open Reload Save Execute Show
sig Slot {}
sig Owner {var slots: set Slot}
sig User {var holds: set Slot}

pred create [o: Owner, s: Slot] {
  no slots.s
  slots' = slots + o -> s
  holds' = holds
}

pred reserve [u: User, o: Owner, s: Slot] {
  no holds.s
  o -> s in slots
  holds' = holds + u -> s
  slots' = slots
}

pred cancel [u: User, s: Slot] {
  u -> s in holds
  holds' = holds - u -> s
  slots' = slots
}

pred can_use [u: User, o: Owner, s: Slot] {
  u -> s in holds and o -> s in slots
}

pred skip {slots' = slots and holds' = holds}
```

```
fact {
  no holds and no slots -- initially
  always (skip or some u: User, s: Slot, o: Owner |
    create[o,s] or reserve [u,o,s] or cancel[u,s])
}

check {
  -- can always use after reserve: not true
  all u: User, s: Slot, o: Owner |
    always (create[o,s] and after reserve[u,o,s]
      implies after after always can_use[u,o,s])
}
```

see: <https://github.com/haslab/Electrum>

checking a tactic with electrom



```
sig Slot {}
sig Owner {var slots: set Slot}
sig User {var holds: set Slot}

pred create [o: Owner, s: Slot] {
  no slots.s
  slots' = slots + o -> s
  holds' = holds
}

pred reserve [u: User, o: Owner, s: Slot] {
  no holds.s
  o -> s in slots
  holds' = holds + u -> s
  slots' = slots
}

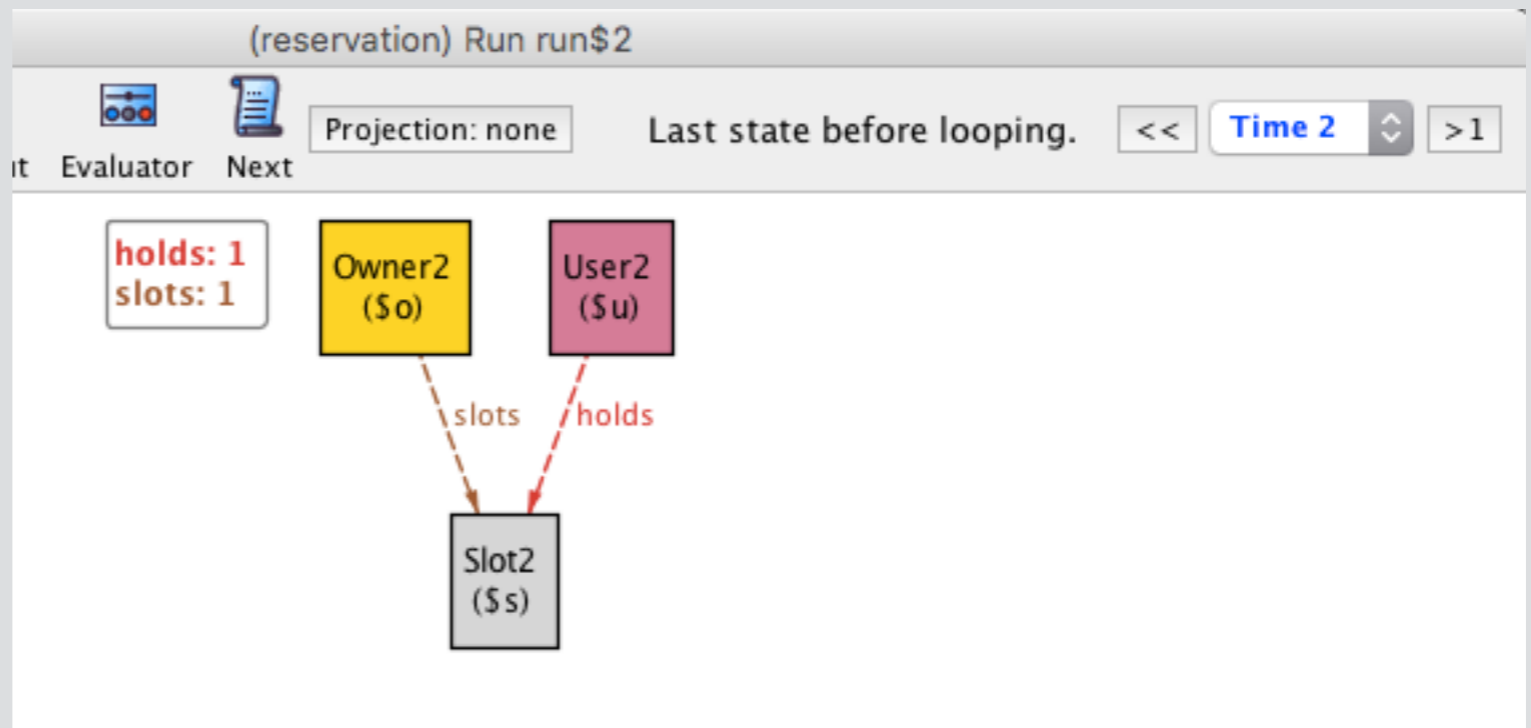
pred cancel [u: User, s: Slot] {
  u -> s in holds
  holds' = holds - u -> s
  slots' = slots
}

pred can_use [u: User, o: Owner, s: Slot] {
  u -> s in holds and o -> s in slots
}

pred skip {slots' = slots and holds' = holds}
```

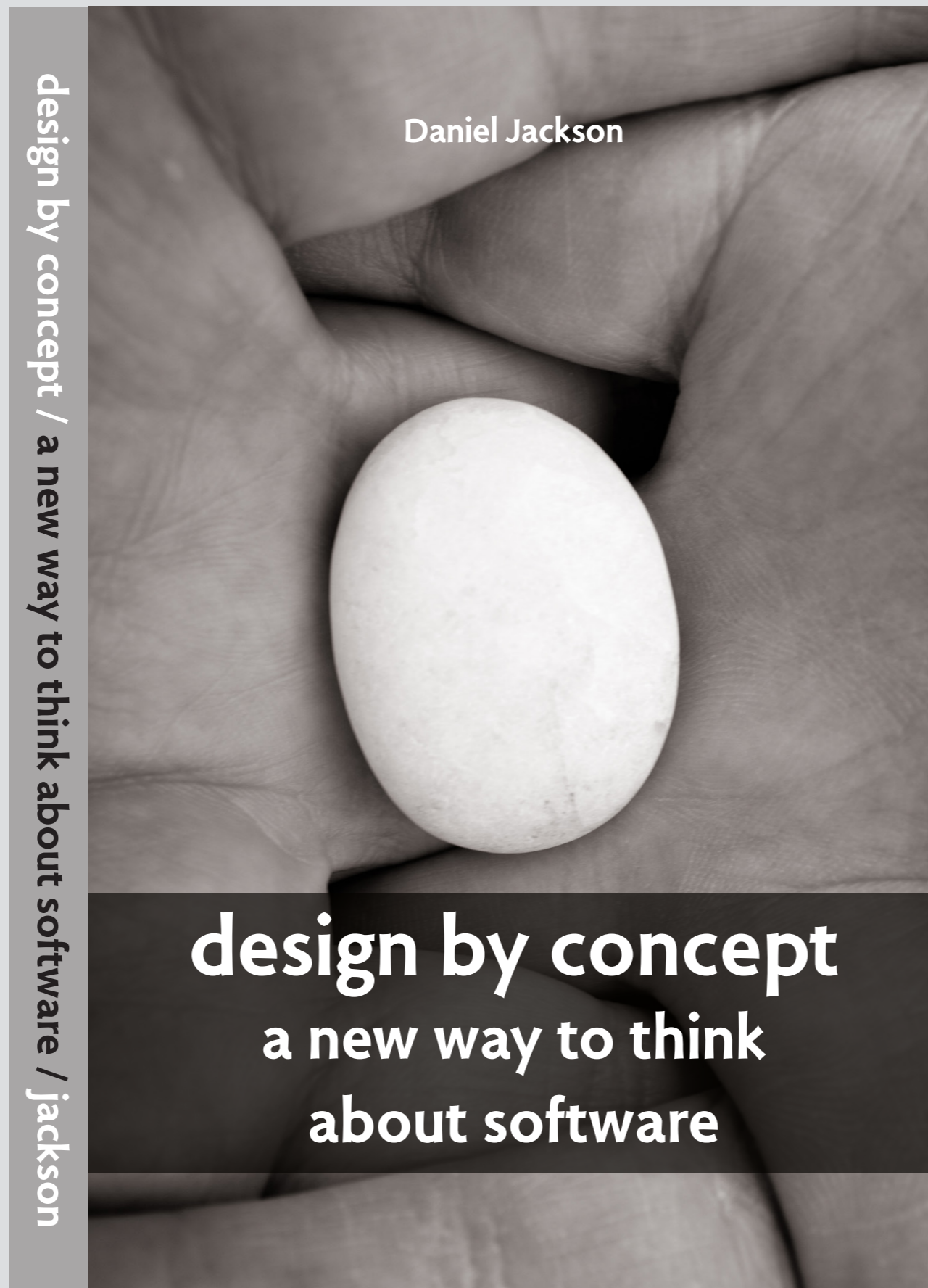
```
fact {
  no holds and no slots -- initially
  always (skip or some u: User, s: Slot, o: Owner |
    create[o,s] or reserve [u,o,s] or cancel[u,s])
}

check {
  -- can always use after reserve: not true
  all u: User, s: Slot, o: Owner |
    always (create[o,s] and after reserve[u,o,s]
      implies after after always can_use[u,o,s])
}
```



see: <https://github.com/haslab/Electrum>

a book draft

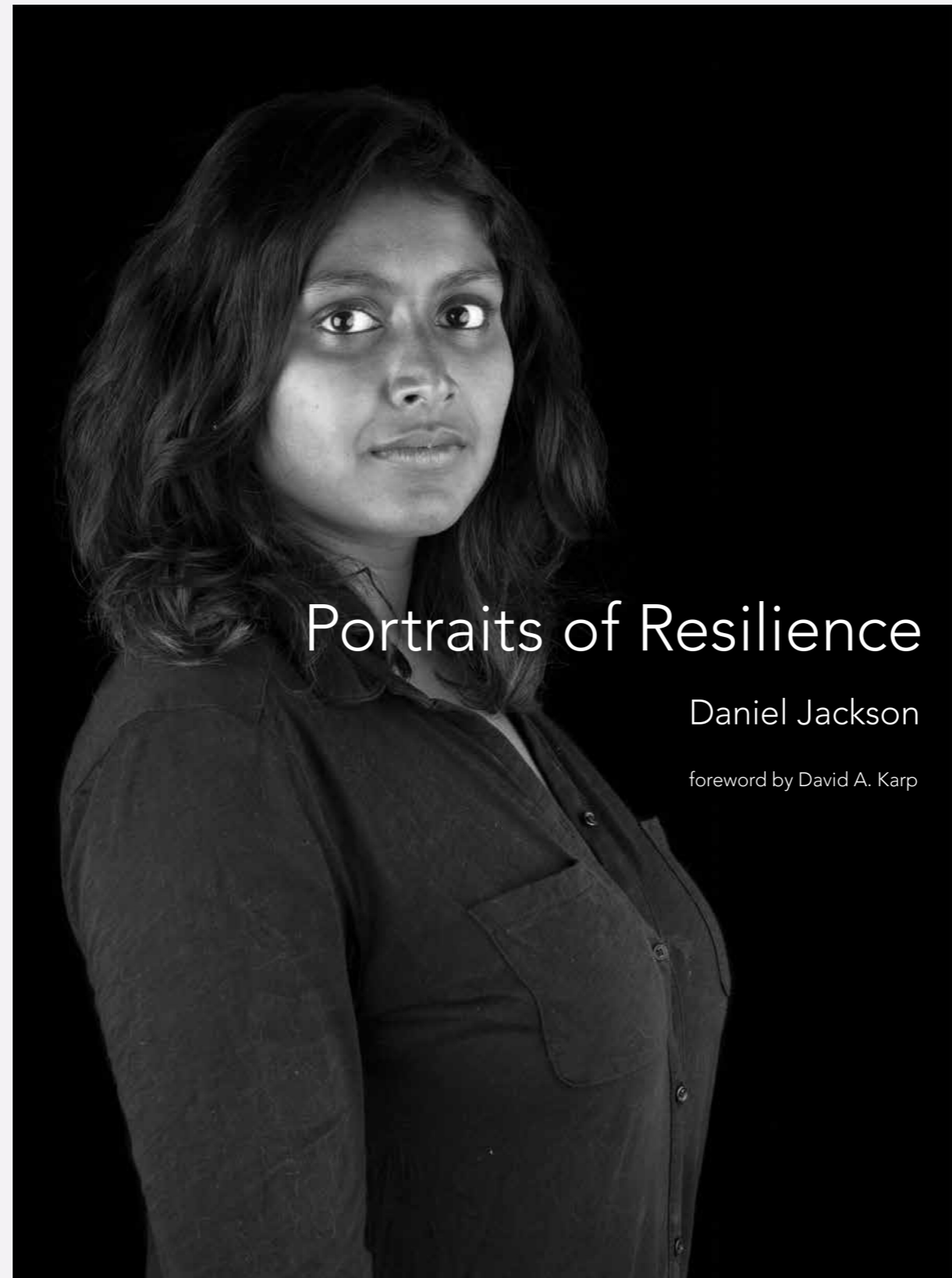


Daniel Jackson

design by concept
a new way to think
about software

design by concept / a new way to think about software / jackson

a rather different book






<http://portraitsofresilience.com>

studio 1



identifying concepts: resy

here's a typical reservation app. what concepts can you identify?


CAFE SUSHI

★★★★★ — RESY  BOOK TONIGHT  TOP RATED  NEW ON RESY


Sushi • Harvard Square • About \$50 per person


 **Tomorrow** ▾  **2 Guests** ▾ Available Unavailable Sold Out Closed

LUNCH

12:00PM DINING ROOM	12:30PM DINING ROOM	1:00PM DINING ROOM
1:30PM DINING ROOM	2:00PM DINING ROOM	 Notify Me

DINNER

8:30PM DINING ROOM	9:00PM DINING ROOM	 Notify Me
------------------------------	------------------------------	---

 BACK

RESY

Cafe Sushi uses Resy's hospitality platform to securely manage your account information and reservations.

Please enter your mobile phone number to verify or create an account:

+1 ▾

Continue

Cafe Sushi and Resy will only use this number to confirm and manage your reservation.

RESY

Check your mobile phone!

We sent a 6-digit confirmation code to **+1 (617) 396-1160**. It expires shortly, so enter it soon.

-

identifying concepts: stack exchange

another example: a typical Q&A app

Philosophy

Home
Questions
Tags
Users
Unanswered

All Questions

1,695 questions

Newest Frequent Votes Active Unanswered

Ask Question

74
votes

19
answers


32k views

Could 'cogito ergo sum' possibly be false?

I've heard it postulated by some people that "we can't truly know anything". While that does seem to apply to the vast majority of things, I can't see how 'cogito ergo sum' can possibly be false. ...

epistemology knowledge existence descartes skepticism

asked Jun 7 '11 at 22:00

 Jez
997 ● 2 ● 11 ● 22

186
votes

29
answers


26k views

Was mathematics invented or discovered?

What would it mean to say that mathematics was invented and how would this be different from saying mathematics was discovered? Is this even a serious philosophical question or just a meaningless/...

epistemology ontology philosophy-of-mathematics

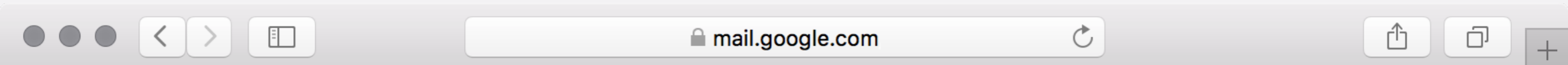
asked Jun 7 '11 at 20:04

 Ami
1,484 ● 3 ● 12 ● 12

gmail
surprises

organizing messages

organizing messages

A search input field with a magnifying glass icon on the right side.

Gmail ▾

Navigation and status bar containing a refresh button, a "More" dropdown, "1-1 of 1" status, left and right arrow buttons, a keyboard icon, and a settings gear icon.

COMPOSE

Category tabs for "Primary" (selected), "Social", and "Promotions", with a plus sign for more categories.

Inbox

☆ Alyssa P. Hacker javascript - Reminds you of the old days, eh? 9:14 pm

- Starred
- Sent Mail
- Drafts
- Trash

0 GB (0%) of 15 GB used
[Manage](#)

[Terms](#) - [Privacy](#)

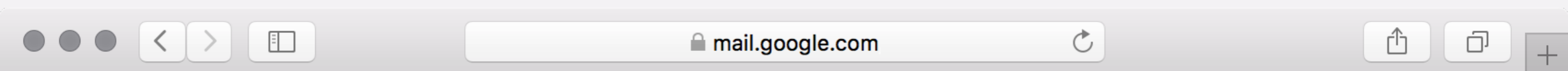
Last account activity: 14 hours ago
[Details](#)

- Categories
- Social
- Promotions
- Updates
- Forums

- hacking
- meetups
- todo
- More ▾



organizing messages

A search bar with a magnifying glass icon on the right side.

Gmail ▾



COMPOSE

- Inbox
- Starred
- Sent Mail
- Drafts
- Trash

- Categories
- Social
- Promotions
- Updates
- Forums

- hacking
- meetups
- todo
- More ▾

Primary

Alyssa P. Hacker

0 GB (0%) of 15 GB used
[Manage](#)

Label as:

hacking
 meetups
 todo
 Social
 Promotions
 Updates
 Forums

Apply

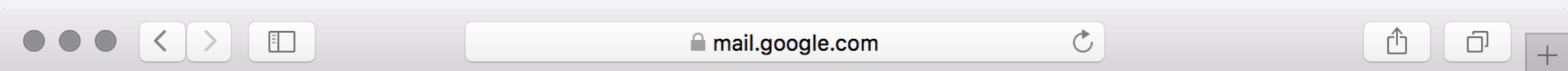
Promotions +

s you of the old days, eh? 9:14 pm

Last account activity: 14 hours ago
[Details](#)



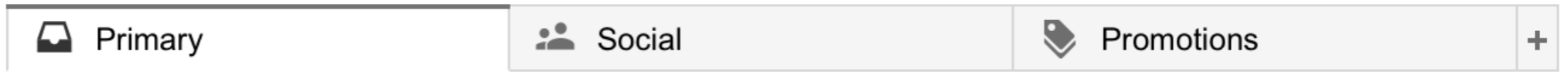
organizing messages



Gmail ▾



COMPOSE



Inbox



Starred
Sent Mail
Drafts
Trash

0 GB (0%) of 15 GB used
[Manage](#)

[Terms](#) - [Privacy](#)

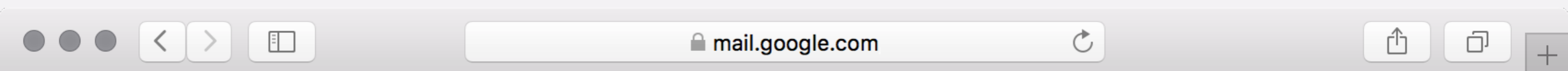
Last account activity: 14 hours ago
[Details](#)

Categories
Social
Promotions
Updates
Forums

hacking
meetups
todo
More ▾



organizing messages



label:hacking



Gmail



More

1-1 of 1



COMPOSE

Inbox

Starred

Sent Mail

Drafts

Trash

Categories

Social

Promotions

Updates

Forums

hacking

meetups

todo

More

Alyssa P. Hacker

Inbox javascript - Reminds you of the old da

9:14 pm

0 GB (0%) of 15 GB used
[Manage](#)

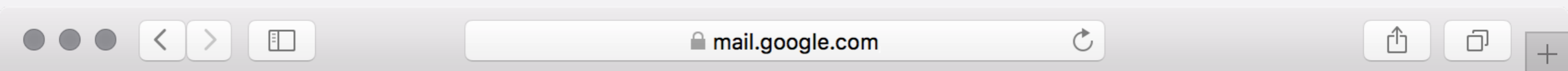
[Terms](#) - [Privacy](#)

Last account activity: 14 hours ago
[Details](#)



automating filtering

automating filtering



Gmail ▾

Navigation and utility buttons including a dropdown menu, a refresh button, a "More" button, a "1-2 of 2" indicator, left and right arrow buttons, a keyboard icon, and a settings gear icon.

COMPOSE

Category tabs for "Primary", "Social", and "Promotions", with a plus sign for additional categories.

Inbox (2)

<input type="checkbox"/>	<input type="checkbox"/>	Alyssa P. Hacker	hacking	javascript - Reminds you of the old	9:14 pm
<input type="checkbox"/>	<input type="checkbox"/>	Andy from Google	Updates	Ben, welcome to your new Googl	May 6

- Starred
- Sent Mail
- Drafts
- Trash

0 GB (0%) of 15 GB used
[Manage](#)

[Terms](#) - [Privacy](#)

Last account activity: 14 hours ago
[Details](#)

- Categories
- Social
- Promotions
- Updates (1)**
- Forums
- hacking (1)**
- meetups
- todo
- More ▾



automating filtering



Gmail

Refresh, More, 1-1 of 1, Keyboard shortcuts, Settings

COMPOSE

Primary, Social, Promotions

Inbox (2)

Alyssa P. Hacker buy this! - My new JS book is out! May 6

- Starred
- Sent Mail
- Drafts
- Trash

0 GB (0%) of 15 GB used Manage Terms - Privacy Last account activity: 14 hours ago Details

Categories

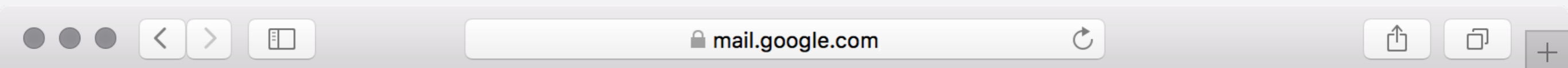
- Social
- Promotions
- Updates (1)
- Forums

hacking (1)

- meetups
- todo
- More

slightly surprising behavior #1

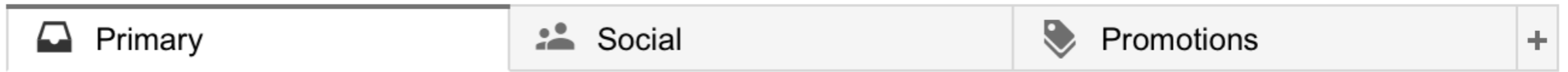
slightly surprising behavior #1

A search input field with a blue search button on the right.

Gmail ▾



COMPOSE



Inbox

☆ Alyssa, me (2) hacking javascript - Yes, it does. On Tue, Ma 9:14 pm

Starred
Sent Mail
Drafts
Trash

0 GB (0%) of 15 GB used
[Manage](#)

[Terms](#) - [Privacy](#)

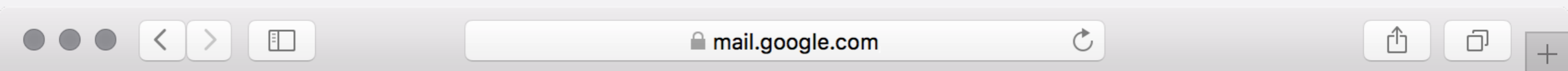
Last account activity: 26 minutes ago
[Details](#)

Categories
Social
Promotions
Updates
Forums

hacking
meetups
todo
More ▾



slightly surprising behavior #1



in:sent



Gmail



More

1-1 of 1



COMPOSE



To: Alyssa P. Hacker (2)

Inbox

hacking

javascript - Yes, it does. On

9:40 pm

0 GB (0%) of 15 GB used
[Manage](#)

[Terms](#) - [Privacy](#)

Last account activity: 26 minutes ago
[Details](#)

Inbox

Starred

Sent Mail

Drafts

Trash

Categories

Social

Promotions

Updates

Forums

hacking

meetups

todo

More



slightly surprising behavior #1



in:sent



Gmail



Move to Inbox



More



COMPOSE

Inbox

Starred

Sent Mail

Drafts

Trash

Categories

Social

Promotions

Updates

Forums

hacking

meetups

todo

More

javascript

Inbox x hacking x



Alyssa P. Hacker Reminds you of the old days, eh?

9:14 PM (33 minutes ago) ☆

Ben Bitdiddle <benito.bitdiddle@gmail.com>

9:40 PM (7 minutes ago) ☆

to Alyssa

Yes, it does.



Click here to [Reply](#) or [Forward](#)

0 GB (0%) of 15 GB used
[Manage](#)

[Terms](#) - [Privacy](#)

Last account activity: 26 minutes ago
[Details](#)



slightly surprising behavior #2

Primary Social Promotions +

☆ me, Alyssa (12) hacking meetups javascript - Hello again Be 11:48 am

slightly surprising behavior #2

The image shows a screenshot of an email client interface. At the top, there are three tabs: "Primary", "Social", and "Promotions". Below the tabs, a search bar contains the text "label:hacking". To the right of the search bar are icons for a grid, a notification bell, and a profile circle with the letter "B". Below the search bar, there is a row of controls: a checkbox with a dropdown arrow, a refresh button, a "More" button with a dropdown arrow, the text "1-1 of 1", left and right navigation arrows, a keyboard icon with a dropdown arrow, and a settings gear icon with a dropdown arrow. Below these controls, a search result is displayed for "me, Alyssa (12)". The result has a checkbox and a star icon on the left. To the right of the sender name are two labels: "hacking" and "meetups". The subject of the email is "javascript - Hello again Be" and the time is "11:48 am".

☆ me, Alyssa (12) Inbox meetups javascript - Hello again Ben 9:43 am

slightly surprising behavior #2

Primary Social Promotions +

☆ me, Alyssa (12) hacking meetups javascript - Hello again Be 11:48 am

label:hacking

1-1 of 1

☆ me, Alyssa (12) Inbox meetups javascript - Hello again Ben 9:43 am

label:meetups

1-1 of 1

☆ me, Alyssa (12) Inbox hacking javascript - Hello again Ben. 9:58 am

slightly surprising behavior #2

Primary Social Promotions +

☆ me, Alyssa (12) hacking meetups javascript - Hello again Be 11:48 am

label:hacking

☆ me, Alyssa (12) Inbox meetups javascript - Hello again Ben 9:43 am

label:meetups

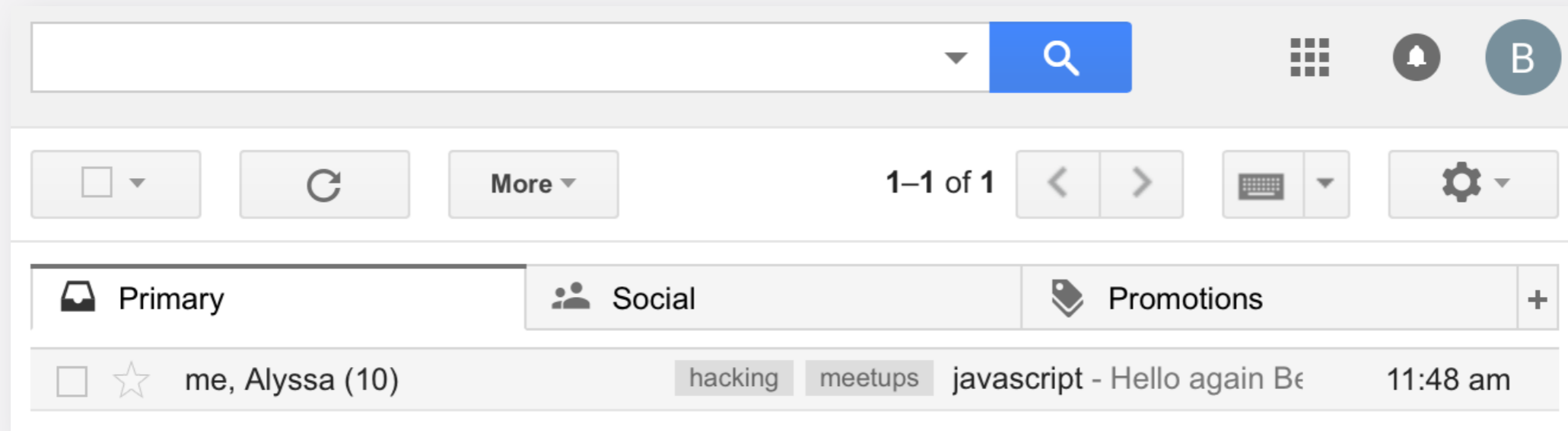
☆ me, Alyssa (12) Inbox hacking javascript - Hello again Ben. 9:58 am

label:hacking label:meetups

☆ me, Alyssa (12) Inbox hacking javascript - Hello again Ben. 9:58 am

🔍 No messages matched your search. Try using [search options](#) such as sender, date, size and more.

slightly surprising behavior #3



slightly surprising behavior #3

Search bar: [] [Q] [Grid] [Bell] [B]

Actions: [] [Refresh] [More] 1-1 of 1 [Left] [Right] [Keyboard] [Settings]

Folder tabs: Primary | Social | Promotions | +

Message: ☆ me, Alyssa (10) [hacking] [meetups] javascript - Hello again Be 11:48 am

Search bar: has:nouserlabels [Q] [Grid] [Bell] [B]

Actions: [] [Download] [Warning] [Trash] [Move to Inbox] [Tag] [Refresh] [More] [Left] [Right] [Keyboard]

Message 1: ☆ Alyssa P. Hacker [Inbox] [Promotions] buy this! - My new JS box 10:33 am

Message 2: ☆ me, Alyssa (10) [Inbox] [hacking] [meetups] javascript - Oh, Al! 9:24 am

slightly surprising behavior #4

slightly surprising behavior #4

1-2 of 2

[Empty Trash now](#) (messages that have been in Trash more than 30 days will be automatically deleted)

<input type="checkbox"/>	<input type="checkbox"/>	me, Alyssa (13)	hacking	meetups	todo	javascript - Hello a	11:48 am
<input type="checkbox"/>	<input type="checkbox"/>	Andy from Google	Updates	Ben, welcome to your new Googl			9:01 am

slightly surprising behavior #4

☐ ▾ ↻ More ▾ 1-2 of 2 < > 🗂 ▾ ⚙ ▾

[Empty Trash now](#) (messages that have been in Trash more than 30 days will be automatically deleted)

☐	🗑	me, Alyssa (13)	hacking	meetups	todo	javascript - Hello a	11:48 am
☐	🗑	Andy from Google	Updates	Ben, welcome to your new Googl			9:01 am

label:todo ▾ 🔍 🗂 🔔 B

☐ ▾ ↻ More ▾ 🗂 ▾ ⚙ ▾

There are no conversations with this label.

slightly surprising behavior #4

☐ ▾ ↻ More ▾ 1-2 of 2 < > 🗂️ ▾ ⚙️ ▾

[Empty Trash now](#) (messages that have been in Trash more than 30 days will be automatically deleted)

☐	🗑️	me, Alyssa (13)	hacking	meetups	todo	javascript - Hello a	11:48 am
☐	🗑️	Andy from Google	Updates	Ben, welcome to your new Googl			9:01 am

label:todo ▾ 🔍 🗂️ 🔔 B

☐ ▾ ↻ More ▾ 🗂️ ▾ ⚙️ ▾

There are no conversations with this label.

label:todo label:trash ▾ 🔍 🗂️ 🔔 B

☐ ▾ ↻ More ▾ 1-1 of 1 < > 🗂️ ▾ ⚙️ ▾

☐	🗑️	me, Alyssa	Trash	hacking	meetups	todo	javascript -	10:11 am
---	----	------------	-------	---------	---------	------	--------------	----------

slightly surprising behavior #4

☐ ↕ ↻ More ▾ 1-2 of 2 < > 🗃️ ▾ ⚙️ ▾

[Empty Trash now](#) (messages that have been in Trash more than 30 days will be automatically deleted)

☐ 🗑️	me, Alyssa (13)	hacking	meetups	todo	javascript - Hello a	11:48 am
☐ 🗑️	Andy from Google	Updates			Ben, welcome to your new Googl	9:01 am

label:todo 🔍 🗃️ 🔔 B

☐ ↕ ↻ More ▾ 🗃️ ▾ ⚙️ ▾

There are no conversations with this label.

label:todo label:trash 🔍 🗃️ 🔔 B

☐ ↕ ↻ More ▾ 1-1 of 1 < > 🗃️ ▾ ⚙️ ▾

☐ 🗑️	me, Alyssa	Trash	hacking	meetups	todo	javascript -	10:11 am
------	------------	-------	---------	---------	------	--------------	----------

label:todo OR label:meetup 🔍 🗃️ 🔔 B

☐ ↕ ↻ More ▾ 🗃️ ▾ ⚙️ ▾





🔍 Some messages in Trash or Spam match your search. [View messages.](#)





slightly surprising behavior #5

Primary Social Promotions +

☆ Alyssa P. Hacker Promotions buy this! - My new JS book is out 10:33 am

slightly surprising behavior #5

 Primary	 Social	 Promotions	+
<input checked="" type="checkbox"/>  Alyssa P. Hacker	Promotions	buy this! - My new JS book is out	10:33 am

 Primary	 Social	 Promotions	+
<input checked="" type="checkbox"/>  Alyssa P. Hacker	Social	buy this! - My new JS book is out!	10:33 am

slightly surprising behavior #5

The image displays three sequential screenshots of an email client interface, illustrating a specific behavior related to email categorization. The top screenshot shows a message from Alyssa P. Hacker with the subject "buy this! - My new JS book is out" and a timestamp of 10:33 am. The message is categorized under the "Promotions" label. The middle screenshot shows the same message, but the "Social" label is highlighted, indicating a change in categorization. The bottom screenshot shows the search filter "label:social" applied, and the message is now categorized under the "Social" label. The interface includes a search bar, a navigation bar with icons for Primary, Social, and Promotions, and a toolbar with various actions like "Remove label", "Inbox", "Promotions", and "Social".

Primary Social Promotions +

✓ ☆ Alyssa P. Hacker Promotions buy this! - My new JS book is out 10:33 am

Primary Social Promotions +

✓ ☆ Alyssa P. Hacker Social buy this! - My new JS book is out! 10:33 am

label:social

Remove label ! 🗑️ 📁 🏷️ More < > 📄 ⚙️

✓ ☆ Alyssa P. Hacker Inbox Promotions Social buy this! - My new JS book is out! 10:33 am

slightly surprising behavior #5

The image shows a sequence of four screenshots from a Gmail inbox, illustrating a specific behavior related to email labeling and filtering. The email in question is from Alyssa P. Hacker, titled "buy this! - My new JS book is out", and is timestamped "10:33 am".

- Top Screenshot:** The inbox is filtered by the "Social" label. The email is highlighted in yellow, and a green "Promotions" label is visible next to the subject line.
- Second Screenshot:** The inbox is filtered by the "Promotions" label. The email is highlighted in yellow, and a blue "Social" label is visible next to the subject line.
- Third Screenshot:** The search bar contains the filter "label:social". The email is highlighted in yellow, and it has three labels: "Inbox", "Promotions", and "Social".
- Bottom Screenshot:** The search bar contains the filter "label:promotions". The email is highlighted in yellow, and it has three labels: "Inbox", "Promotions", and "Social".

The interface includes standard Gmail elements: a top navigation bar with "Primary", "Social", and "Promotions" tabs; a search bar; and a toolbar with actions like "Remove label", "Inbox", "Promotions", and "Social" buttons. The email's subject line is partially visible as "buy this! - My new JS book is out".

slightly surprising behavior #5

The image shows a sequence of Gmail interface elements illustrating a search behavior. It consists of five main sections:

- Section 1:** A Gmail header with tabs for 'Primary', 'Social', and 'Promotions'. Below it, a message from Alyssa P. Hacker is shown with a 'Promotions' label.
- Section 2:** The same Gmail header, but the 'Promotions' tab is highlighted in green. The message below now has a 'Social' label.
- Section 3:** A search bar containing 'label:social'. Below it, a toolbar with various action buttons (checkbox, 'Remove label', folder, tag, 'More', navigation, keyboard, settings) is visible. The message below has labels for 'Inbox', 'Promotions', and 'Social'.
- Section 4:** A search bar containing 'label:promotions'. The toolbar and message below are identical to Section 3.
- Section 5:** A search bar containing 'label:social label:promotions'. The toolbar below it is simplified, showing a checkbox, a refresh button, and a 'More' button. At the bottom, a message states: 'No messages matched your search. Try using [search options](#) such as sender, date, size and more.'

exercise

find a partner so you can work in a pair

pick one of the Gmail surprises

all slides at <https://tinyurl.com/ssft9a>

analyze it in term of concepts

what are the key concepts involved?

which concept(s) is responsible for the surprise?

is the surprise a bug, a conceptual flaw or a user misunderstanding?

can you explain precisely what's going wrong?

can you generalize your observation?

design a fix

propose a modification that eliminates the surprise

lecture two

three
design
problems

gmail
categories

gmail's categories

Primary			Social 23 new		Promotions 100+ new		+
<input type="checkbox"/>	<input type="star"/>	<input type="arrow"/>	Google	New sign-in from Chrome on Mac - New sign-in from Cr		12:30 pm	
<input type="checkbox"/>	<input type="star"/>	<input type="arrow"/>	Keith Muhammad at DeMont.	DeMontrond Auto Group - 14101 North Freeway Housto		12:19 pm	
<input type="checkbox"/>	<input type="star"/>	<input type="arrow"/>	AT&T High Speed Internet.	AT&T High Speed Internet Service Activation - Your A1		10:37 am	
<input type="checkbox"/>	<input type="star"/>	<input type="arrow"/>	Keith Muhammad at DeMont.	DeMontrond Auto Group - 14101 North Freeway Housto		Aug 26	
<input type="checkbox"/>	<input type="star"/>	<input type="arrow"/>	betterbatonrougejobs.com	Job Update -- 2015-08-26 - Looking For An Advantage W		Aug 26	

category tab settings

Categories:

[Learn more](#)

- Primary
- Social
- Promotions
- Updates
- Forums

Starred messages

- Include starred in Primary



Choose which message categories to show as inbox tabs. Other messages will appear in the Primary tab.

some reactions

some reactions

Everything You Need to Know About Gmail's New, **Super-Confusing** Layout



Melanie Pinola

Filed to: GMAIL 6/07/13 9:00am

301,482



12



some reactions

Google gets it terribly, terribly wrong with Gmail tabs and makes me angry



By [Mark Wilson](#)

Published 2 years ago

Follow

71 Comments

Like 55

Share 11

37

Tweet 30

Like many Gmail users, I greeted the [news of the introduction of tabs to the interface](#) with a degree of anticipation -- now it was just a matter of waiting for the feature to roll out so I could try it for myself. Earlier today I was randomly signed out of my Gmail account, and after signing back in and checking the settings menus, I could see that tabs were now available to me. Excitement was short-lived, however; it quickly became apparent that this new feature is a disaster.



some reactions

[Home](#) › [Quick Tech Tip: Disabling Gmail's Category Tabs](#)

Quick Tech Tip: Disabling Gmail's Category Tabs

Mon, 07/29/2013 - 12:17 | [Chuck Gray](#)

in [LibraryPoint Blog](#) [Tech Tutorials](#) [Teen Blog](#) [Tech Answers](#) [Science and Technology](#) [Self-Help and Instructional](#)



Are you a Gmail user? Did you wake up a week or two ago to find that your new messages were now being automatically organized by Gmail into tabs of different, pre-determined categories? And, did you think, like me, that they were **really ugly, stupid, and unnecessary?** Here's a quick tip on how to rid yourself of them!

how google explains labels (!)

Gmail Help



GMAIL



FORUM

Using labels

Labels help you organize your messages into categories -- work, family, to do, read later, jokes, recipes, any category you want. Labels do all the work that folders do, but with an added bonus: you can add more than one to a message.

what you can't do

what you can't do

associate tabs with labels
feature available only for categories

what you can't do

associate tabs with labels

feature available only for categories

create new categories

only new labels

what you can't do

associate tabs with labels

feature available only for categories

create new categories

only new labels

use tabs outside inbox

tabs disappear when you filter on a label

fuji

aspect

ratio

my camera fuji x100s



image quality setting

image quality setting



image quality setting



aspect ratio



aspect ratio



image size setting

image size setting



image size setting

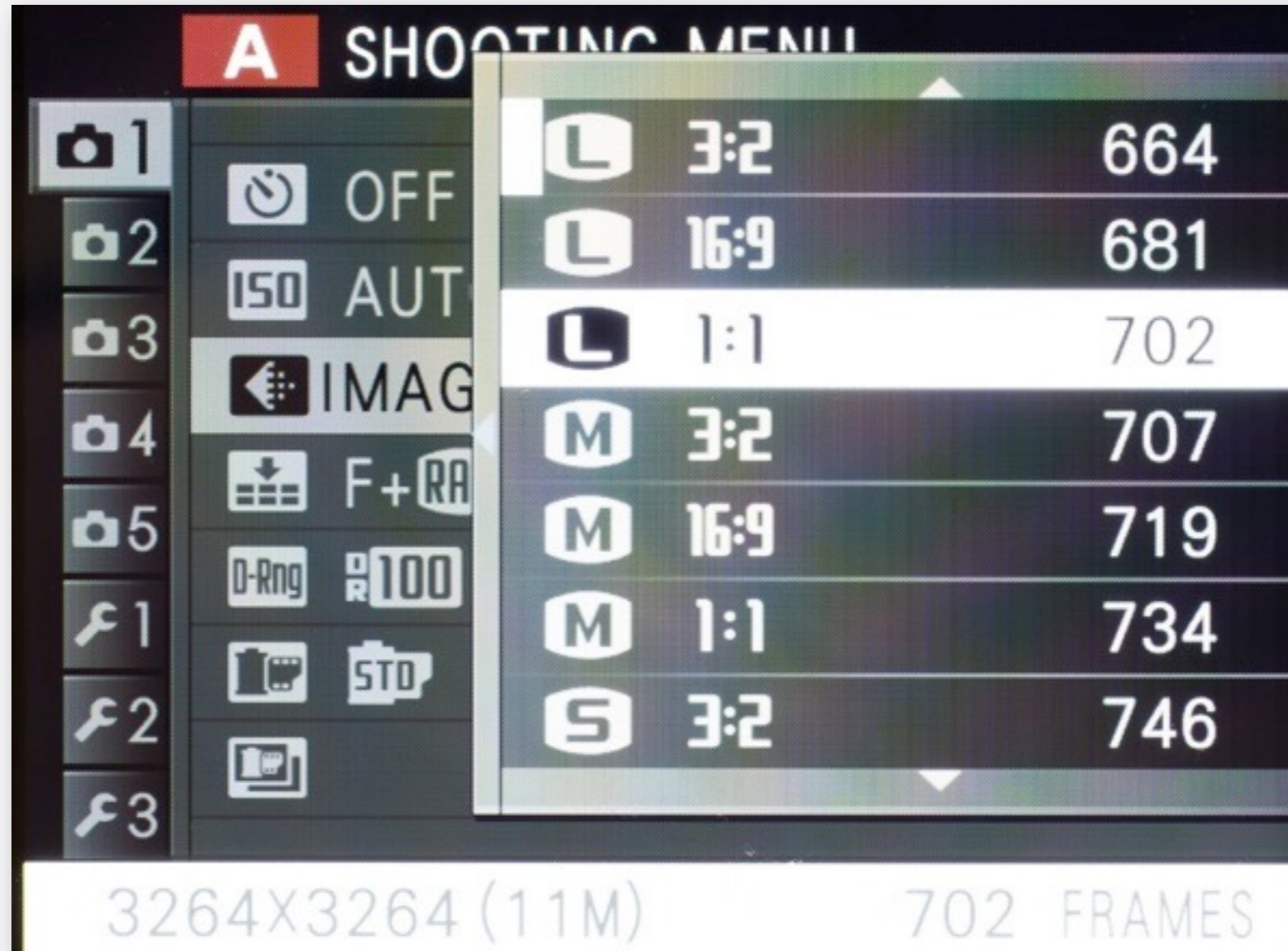


image size setting



non-standard ratio + raw?



what you can't do

non-standard aspect ratio + raw
even though raw images get nice nondestructive crop!

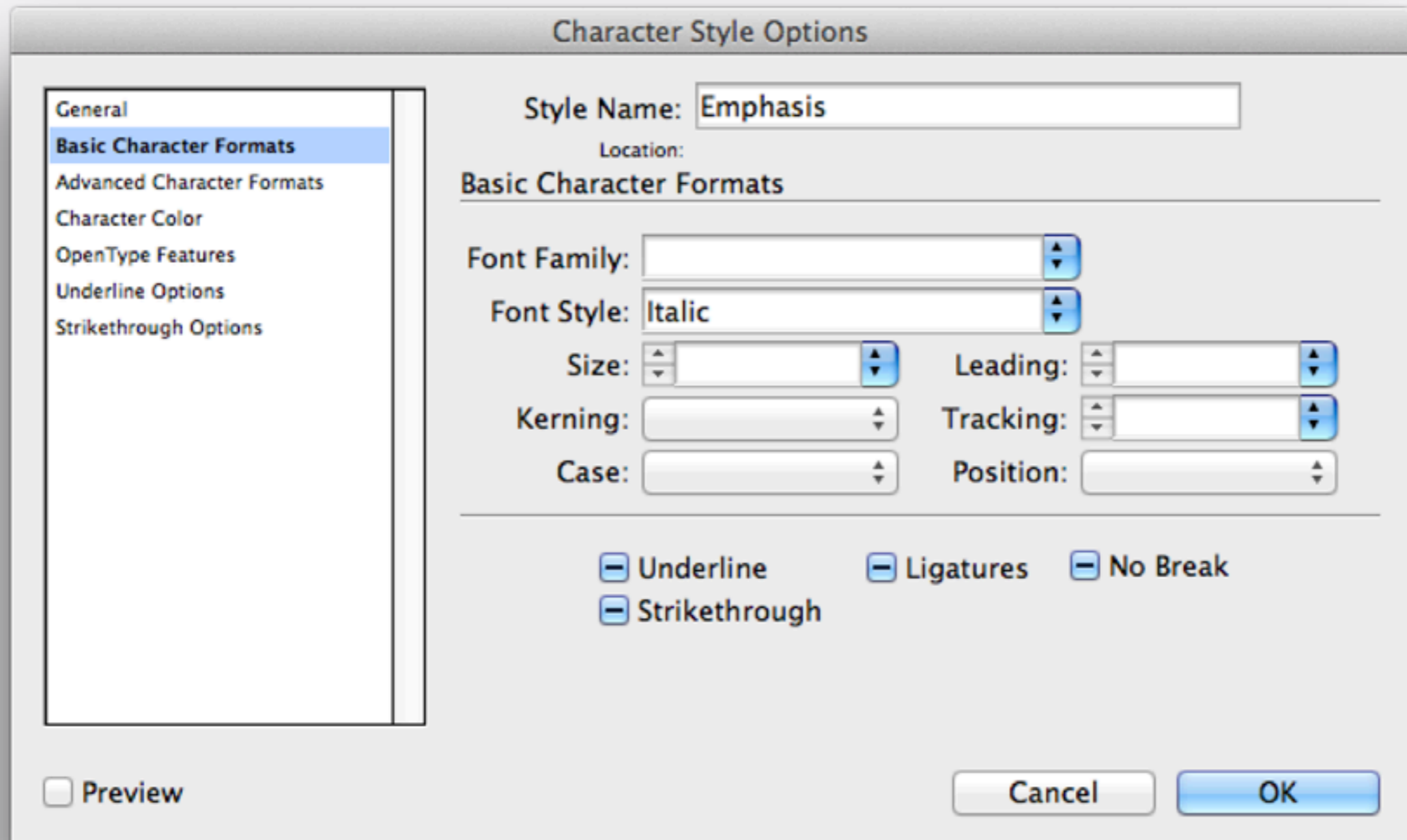
what you can't do

non-standard aspect ratio + raw
even though raw images get nice nondestructive crop!

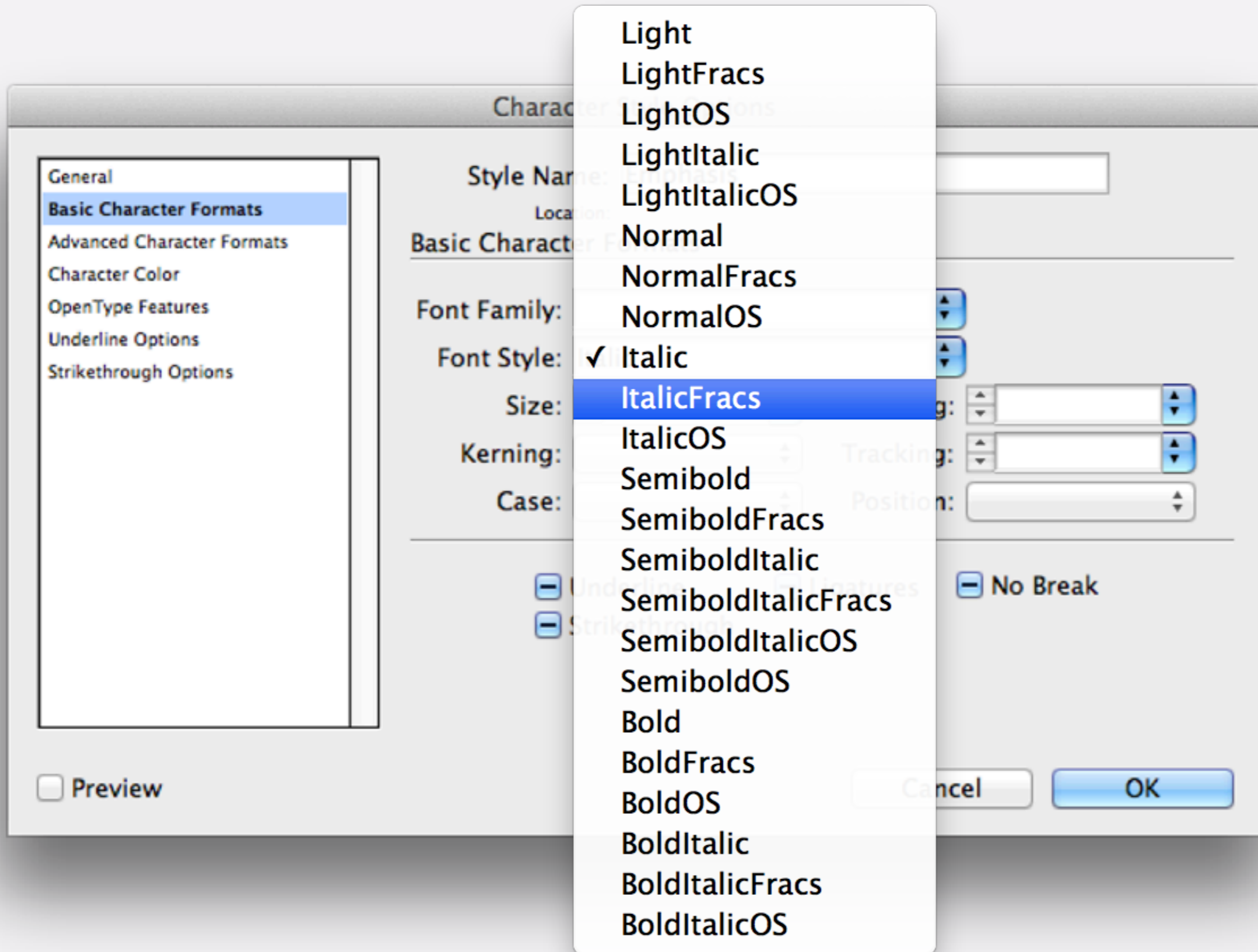


indesign
styles

what's a font?



what's a font?



what you can't do

define a style that italicizes

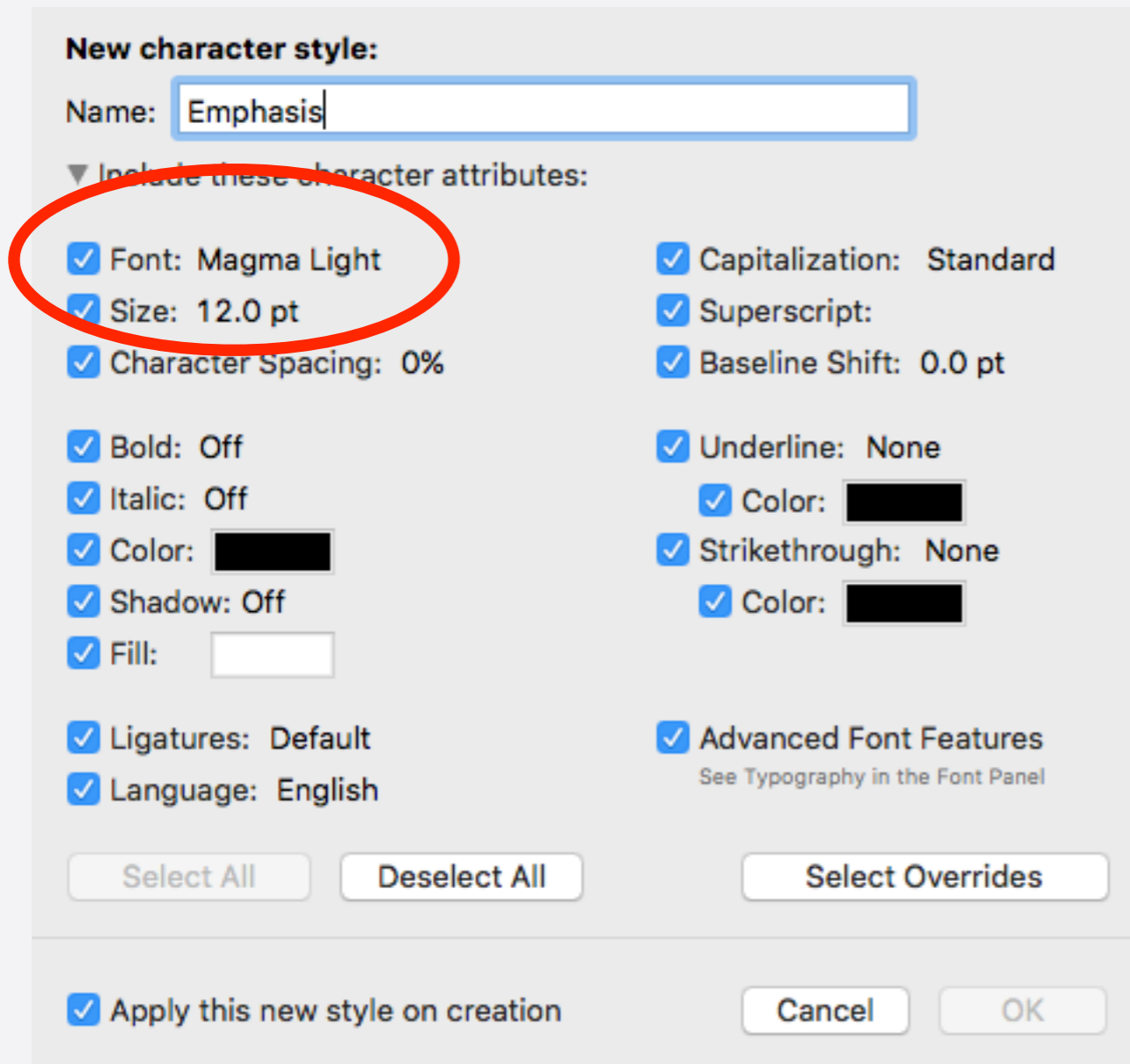
Arno Regular to Arno Italic

Futura Book to Futura Book Oblique

Magma Light to Magma Light Italic

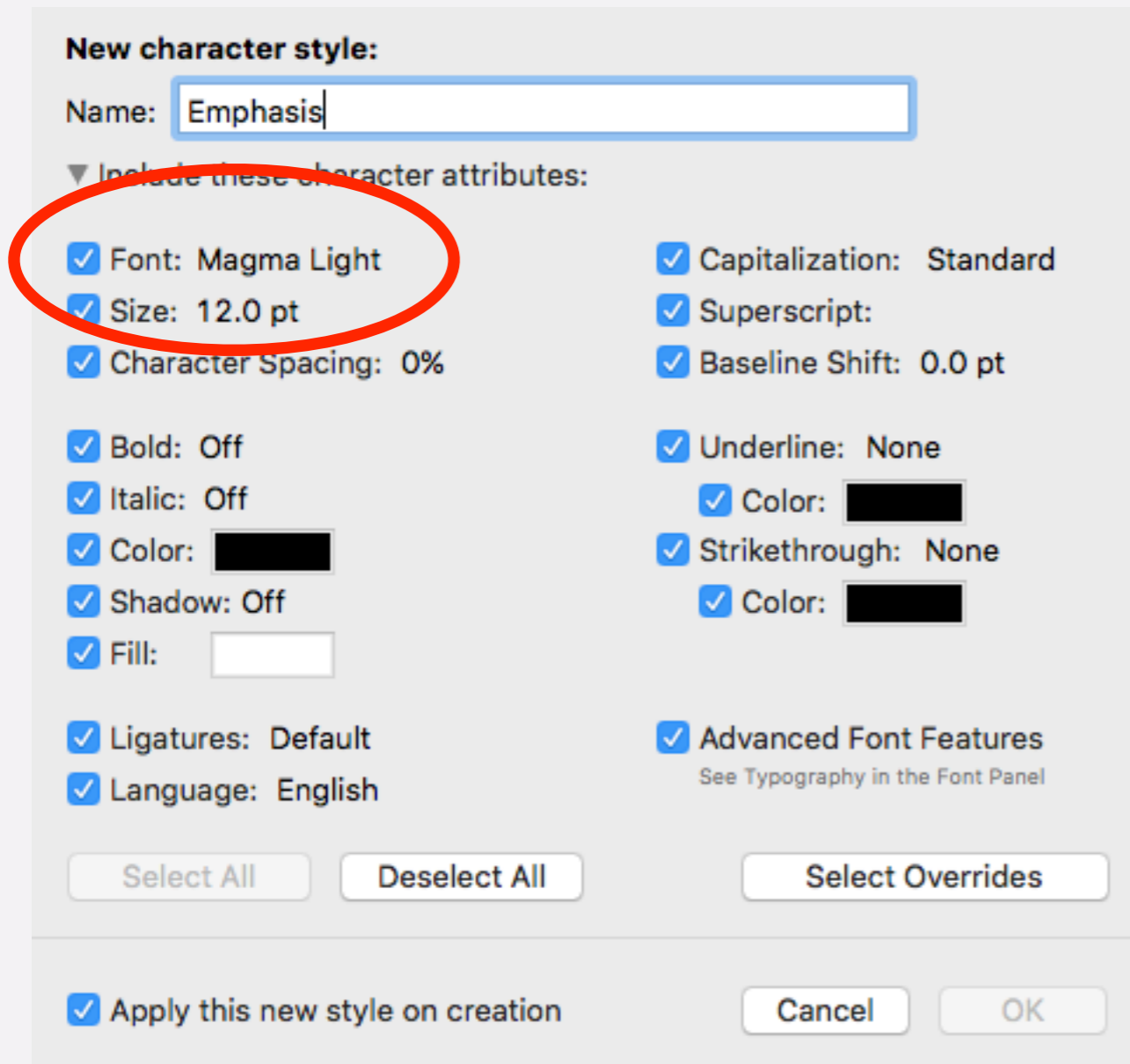
introducing a concept

introducing a concept

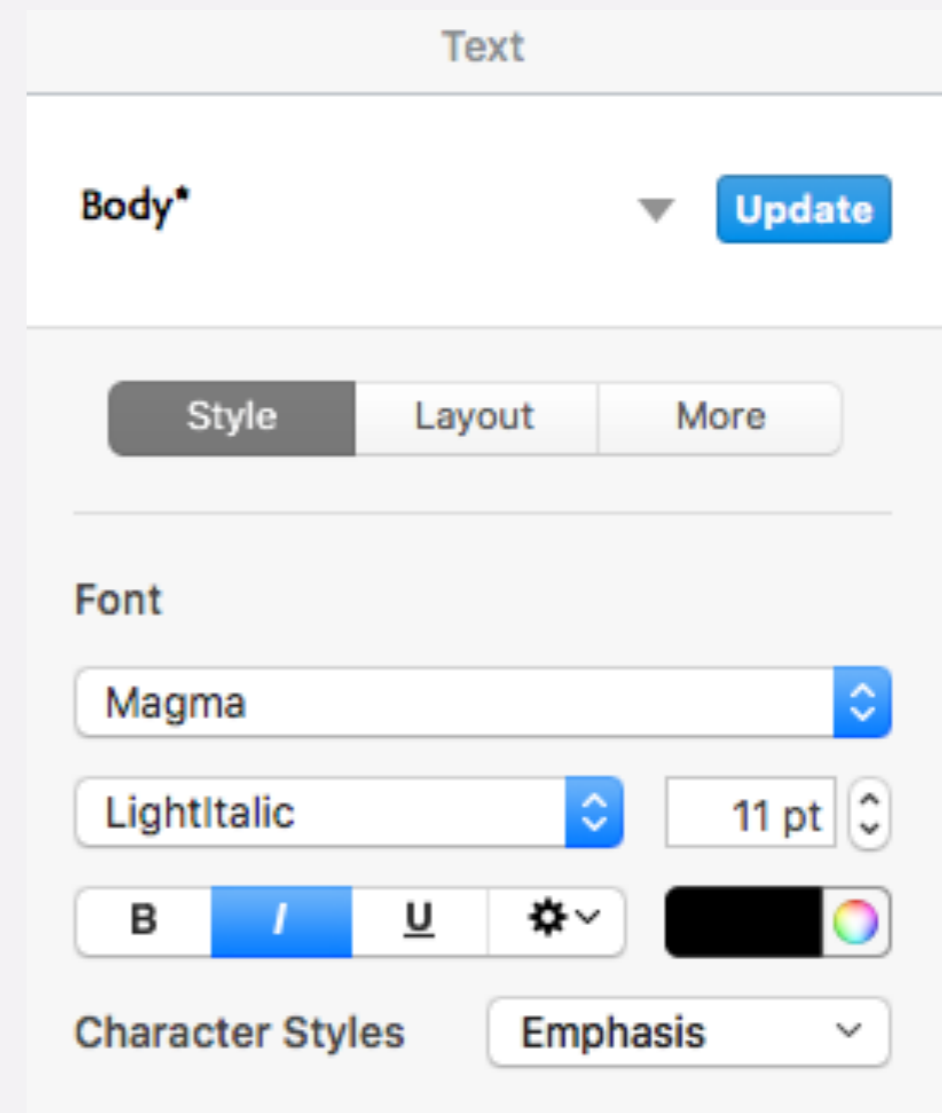


Keynote '09: has subfamilies

introducing a concept



Keynote '09: has subfamilies



Keynote 6: gone again!

what's going on?

what's going on?

gmail

one purpose :: two concepts
organizing messages :: label + category

what's going on?

gmail

one purpose :: two concepts
organizing messages :: label + category

camera

two purposes :: one concept
aspect ratio + image resolution :: image size

what's going on?

gmail

one purpose :: two concepts
organizing messages :: label + category

camera

two purposes :: one concept
aspect ratio + image resolution :: image size

style

one purpose :: no concept
specify a font-independent styling :: ?

the
singularity
rule



Mitchell and Webb: "Unity of Purpose"

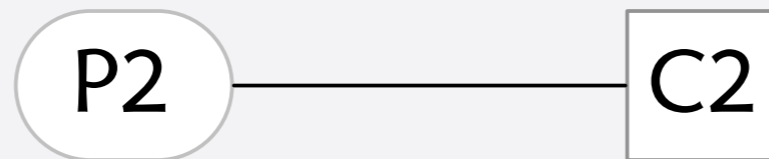
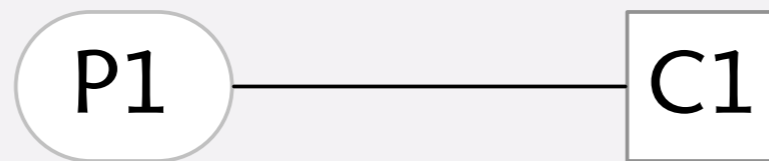


Mitchell and Webb: "Unity of Purpose"

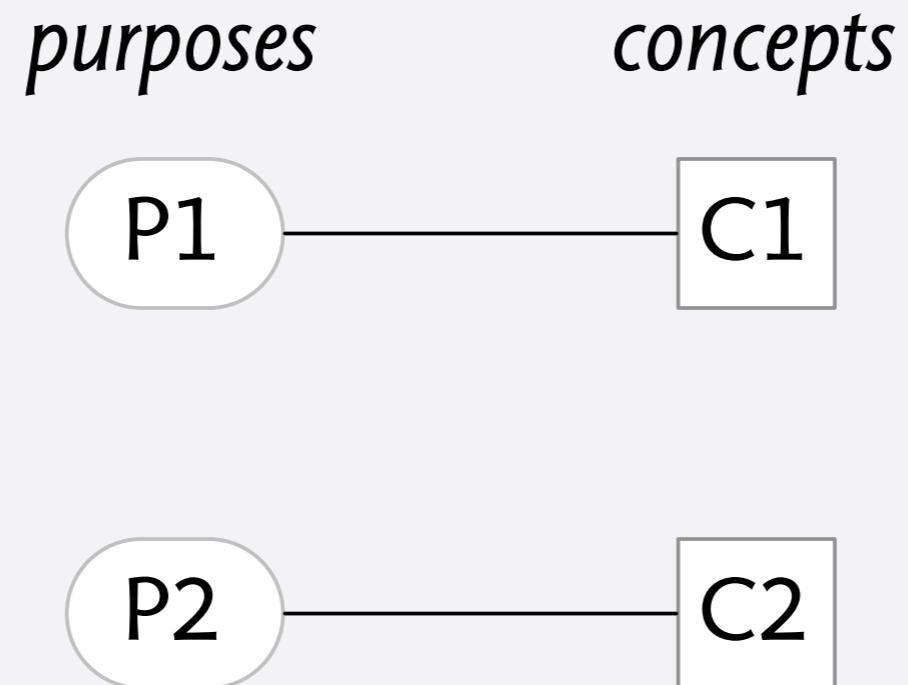
one-to-one mapping

purposes

concepts



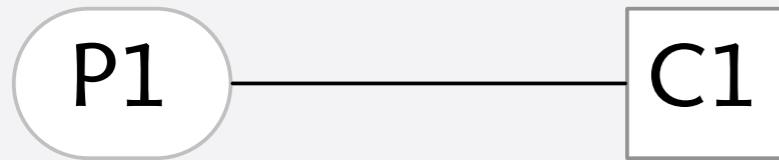
one-to-one mapping



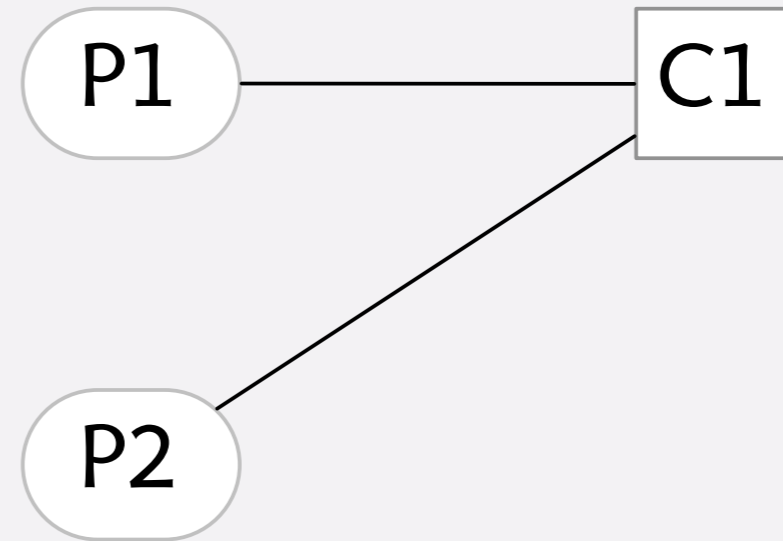
Nam Suh:
Axiomatic Design

four ways to fail

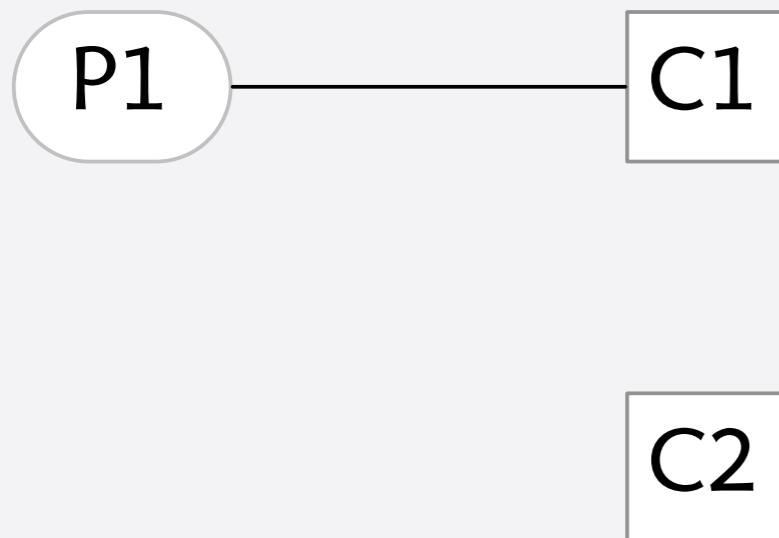
unfulfilled purpose



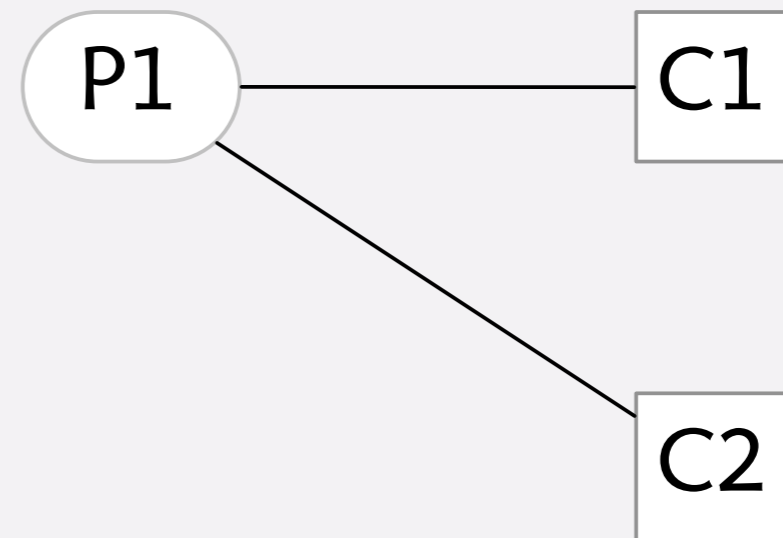
overloaded concept



unmotivated concept



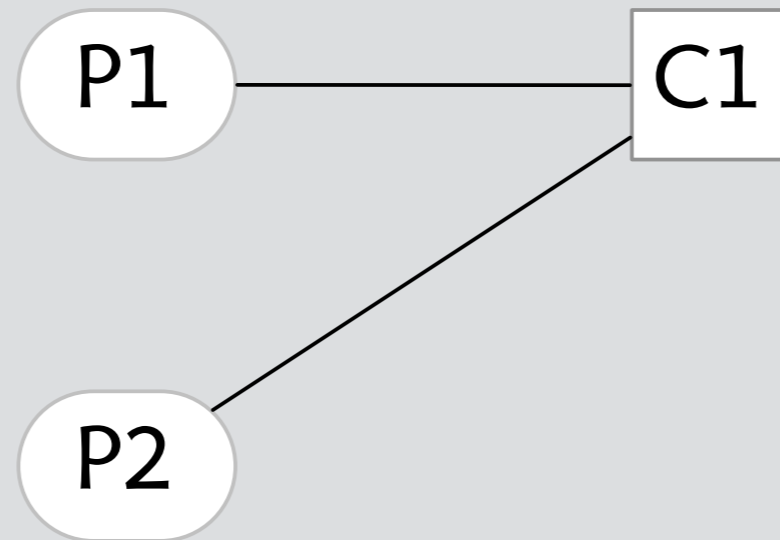
redundant concepts



kinds of
overloading

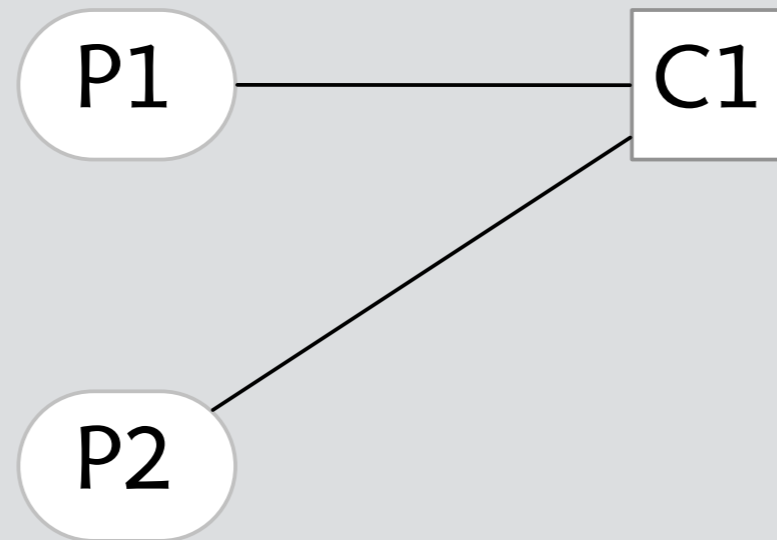
overloaded concepts

No one can serve two masters. Either you will hate the one and love the other, or you will be devoted to the one and despise the other. [Matthew 6:24]



overloaded concepts

No one can serve two masters. Either you will hate the one and love the other, or you will be devoted to the one and despise the other. [Matthew 6:24]



4 forms of overloading:

piggybacking new purpose hacked onto old concept

false convergence two purposes looked the same

emergent purpose second purpose emerged with use

denial designer believes second purpose unnecessary

piggybacking fuji camera

new purpose hacked onto old concept

piggybacking fuji camera

new purpose hacked onto old concept

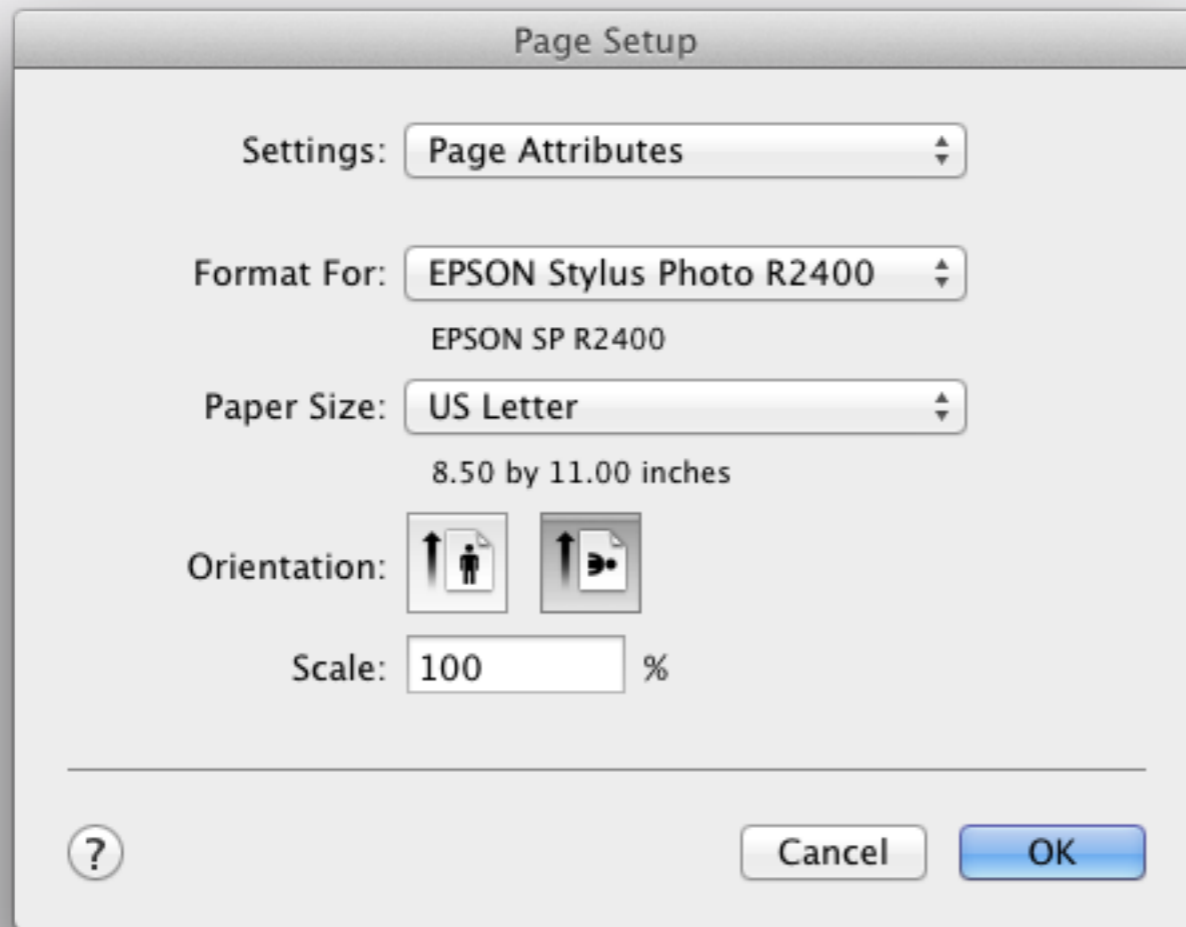


L	3:2	664
L	16:9	681
L	1:1	702
M	3:2	707
M	16:9	719
M	1:1	734
S	3:2	746

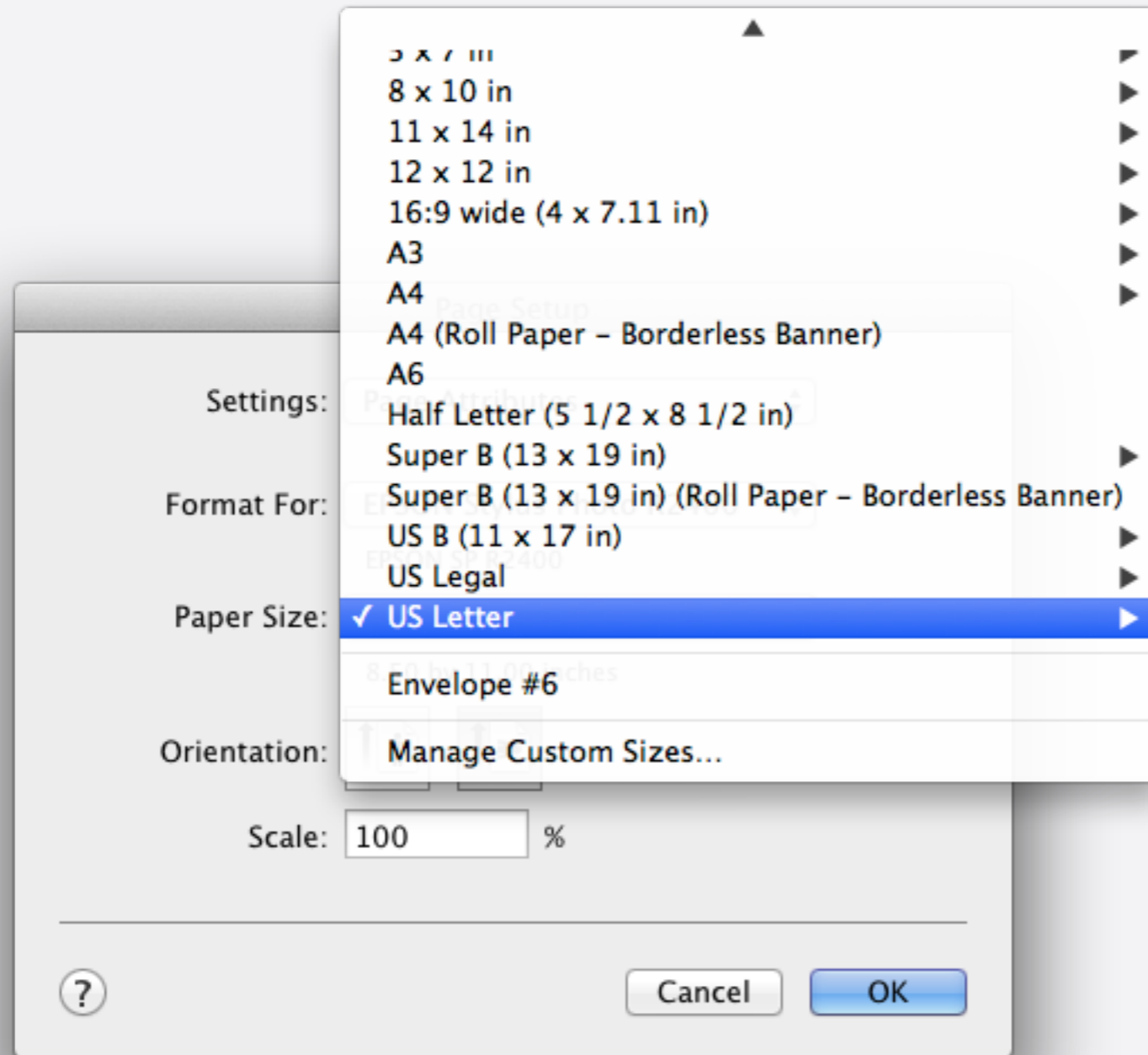
image size
aspect ratio piggybacked
on JPEG dimensions

piggybacking `epson driver`

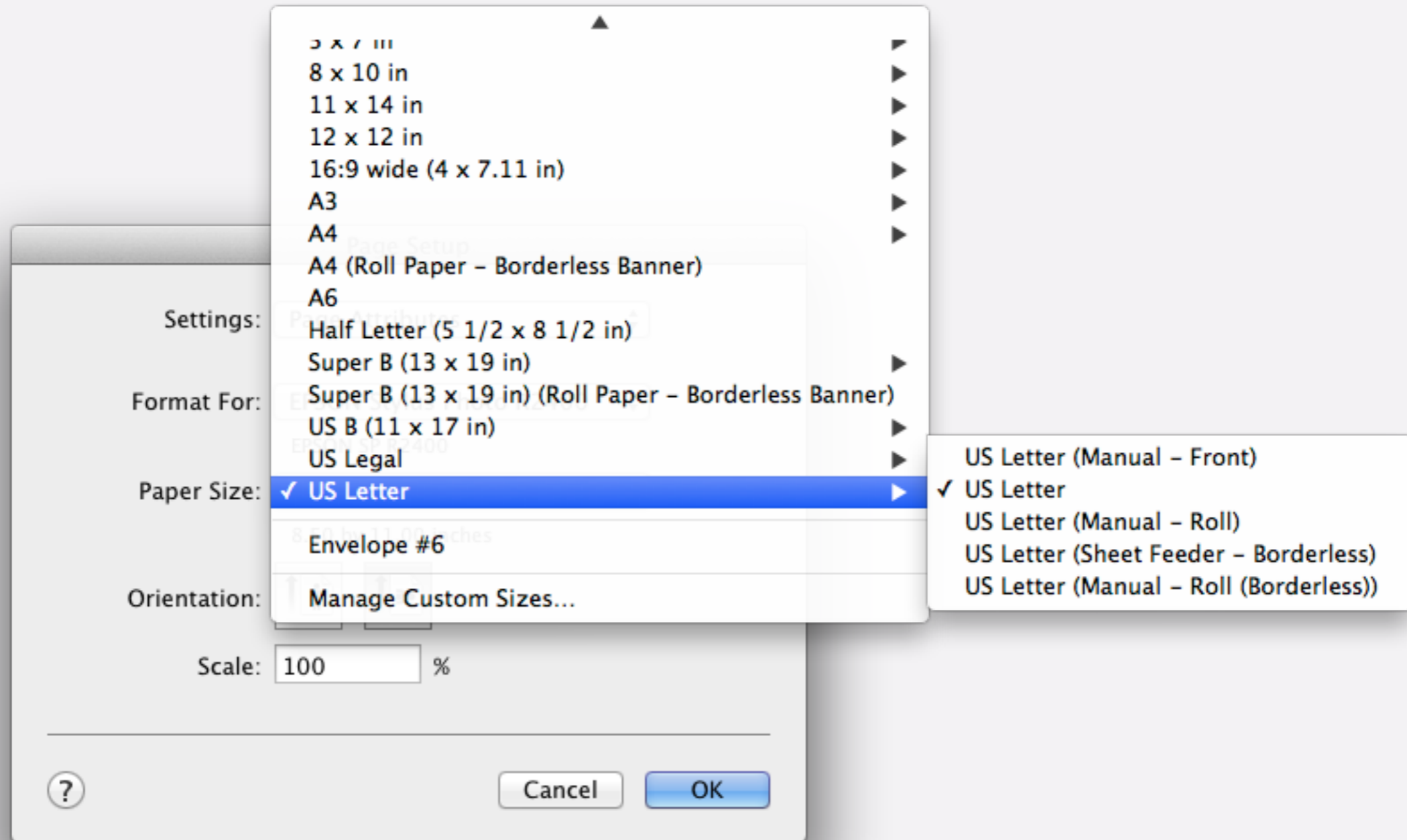
piggybacking epson driver



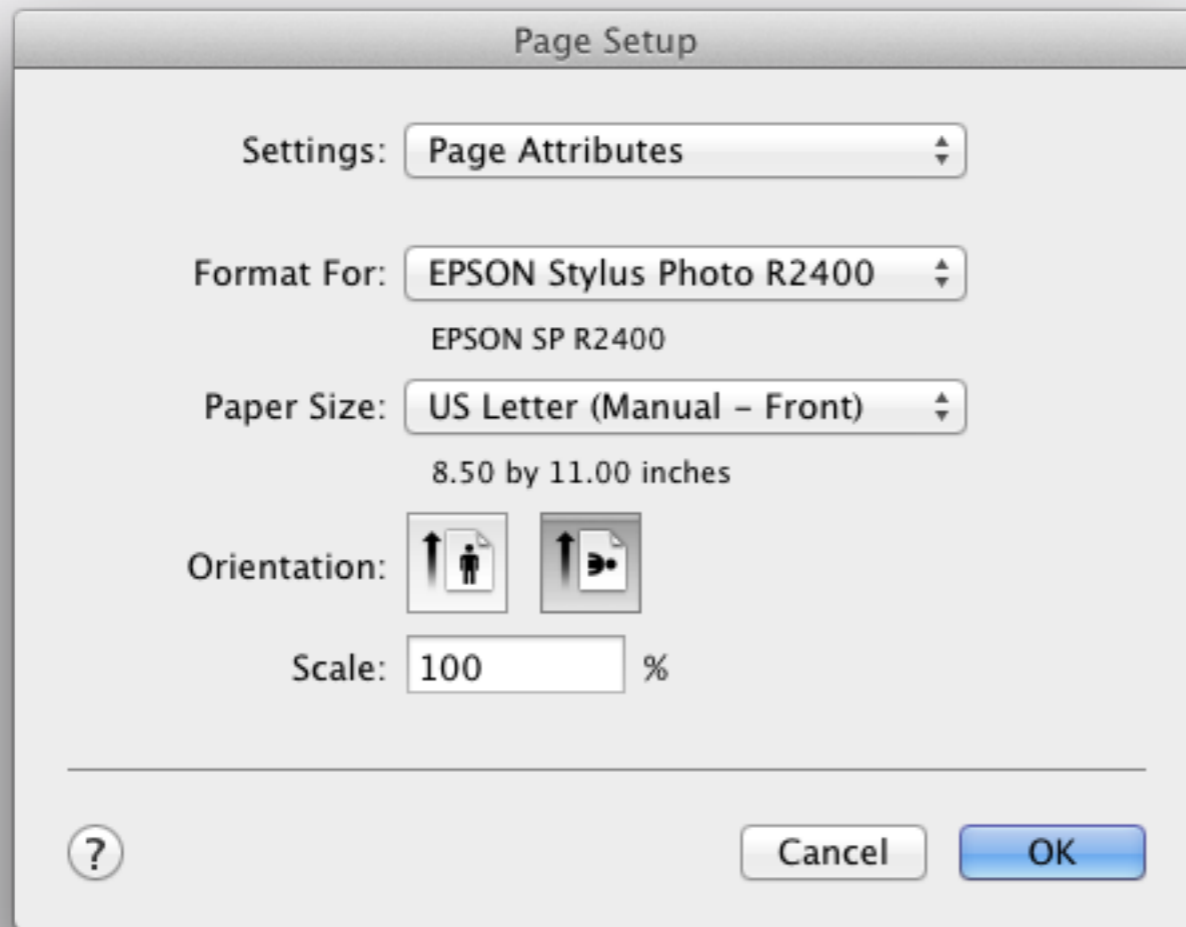
piggybacking epson driver



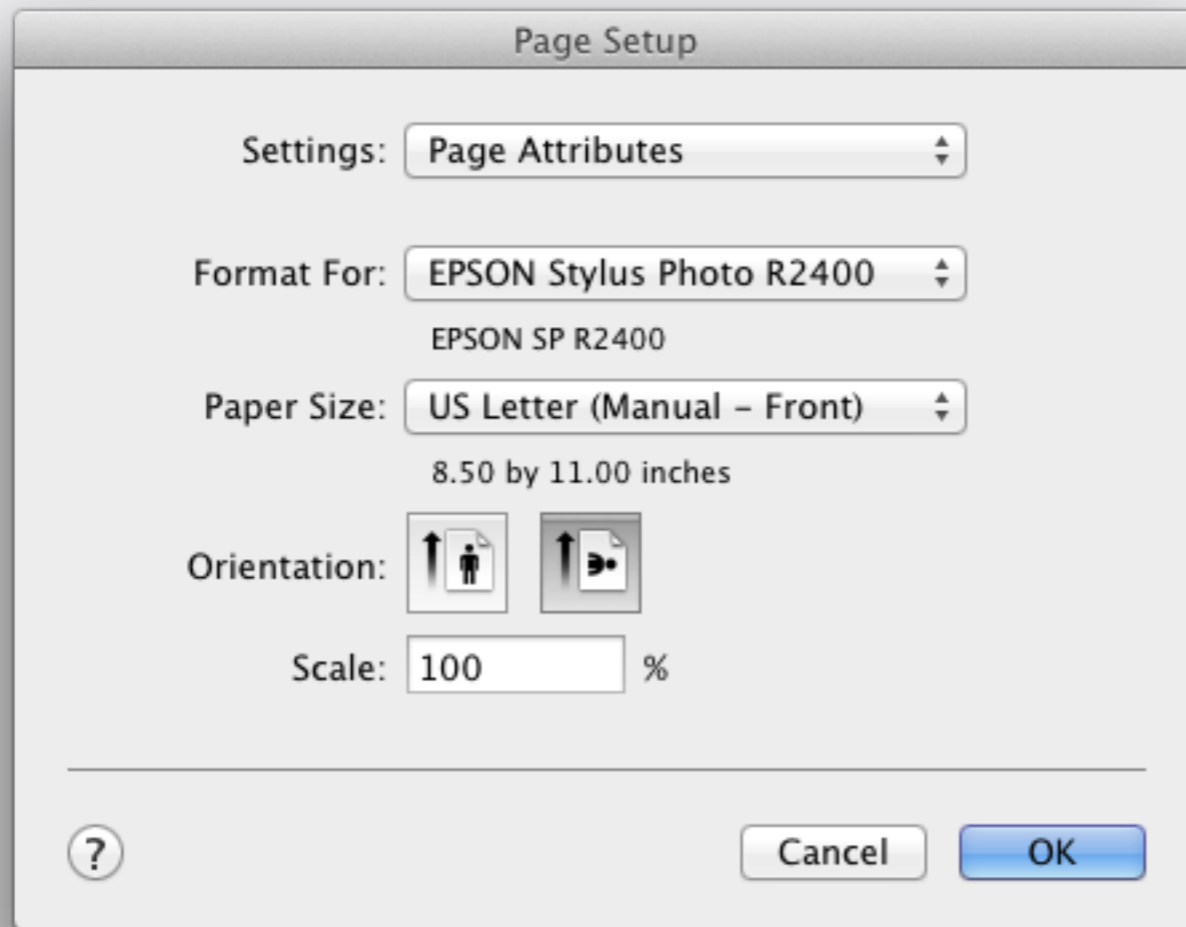
piggybacking epson driver



piggybacking epson driver



piggybacking epson driver



result: can't create custom size for front loading
also, page size presets in Lightroom hold feed setting

false convergence facebook friend

two purposes looked the same



filter incoming posts
control access to my posts
distinct purposes

2011: Facebook added
subscribe/follow

emergent purpose email subject

users find second purpose for concept

emergent purpose email subject

users find second purpose for concept

To: Daniel Jackson <dnj@mit.edu>

Re: Catch me if you can in real life!

initial purpose: summarize content

emergent purpose email subject

users find second purpose for concept

To: Daniel Jackson <dnj@mit.edu>

Re: Catch me if you can in real life!

initial purpose: summarize content

To: csail-related@lists.csail.mit.edu

Re: [csail-related] turn off the lights?

emergent purpose: show sender
if you bcc a list, subject reveals to-address

thanks to Shriram Krishnamurthi

emergent purpose email subject

users find second purpose for concept

To: Daniel Jackson <dnj@mit.edu>

Re: Catch me if you can in real life!

initial purpose: summarize content

To: csail-related@lists.csail.mit.edu

Re: [csail-related] turn off the lights?

emergent purpose: show sender
if you bcc a list, subject reveals to-address

thanks to Shriram Krishnamurthi

To: Daniel Jackson <dnj@mit.edu>

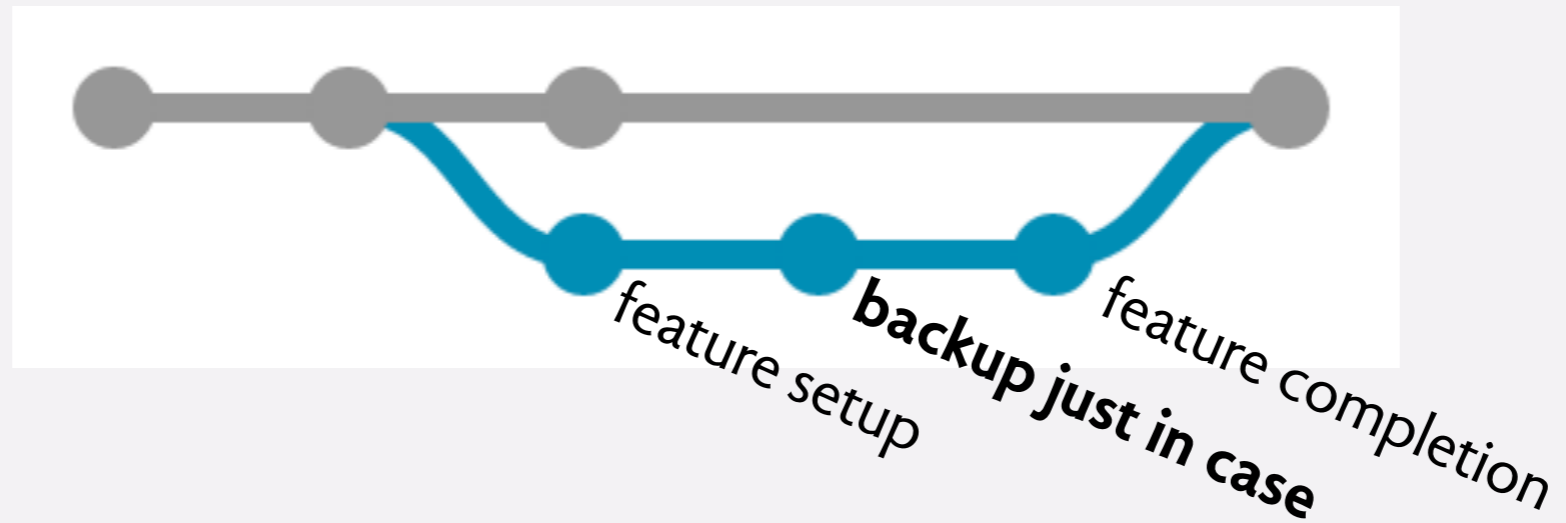
your trip reservation

emergent purpose: group by conversation
can't label reservations from Expedia by trip

thanks to Eunsuk Kang

denial commit

designer believes second purpose unnecessary



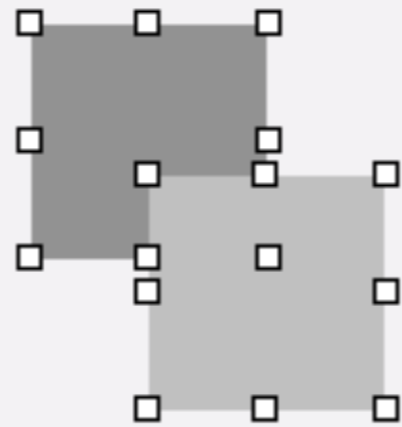
the
uniformity
rule

what makes a usable concept?

operational principle is uniform
always the same actions, irrespective of context

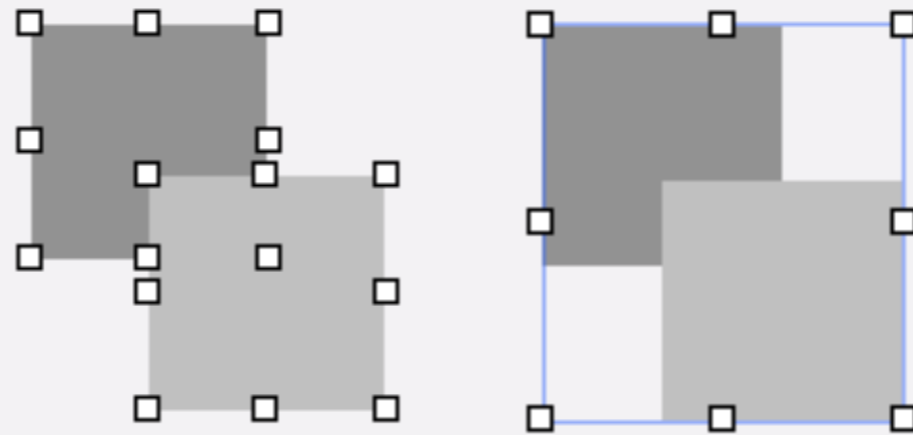
what makes a usable concept?

operational principle is uniform
always the same actions, irrespective of context



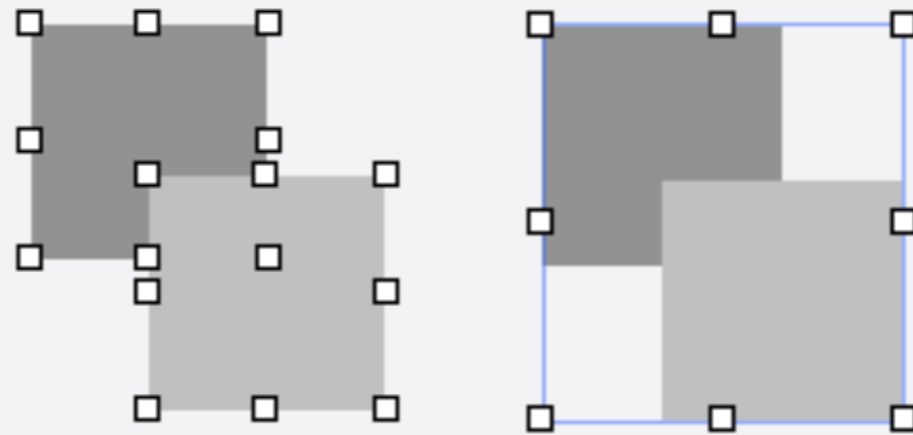
what makes a usable concept?

operational principle is uniform
always the same actions, irrespective of context



what makes a usable concept?

operational principle is uniform
always the same actions, irrespective of context



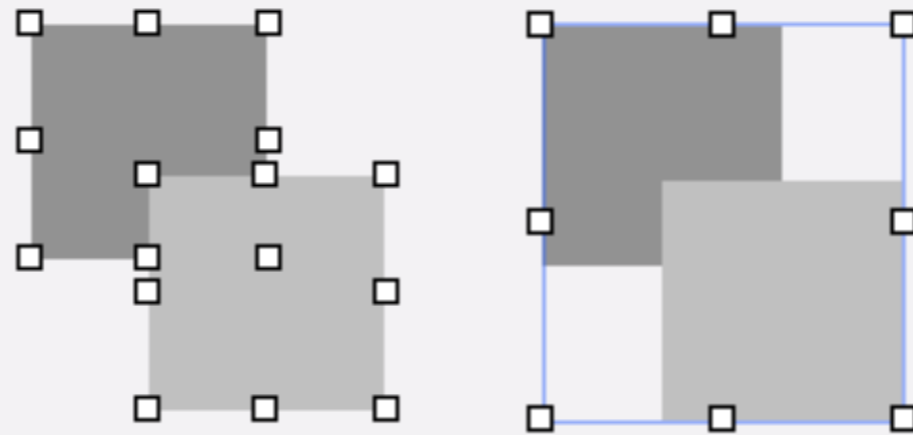
concept: **Group** (Keynote)

purpose: treat set as one

OP: ... select(objs); group(); mutate()...

what makes a usable concept?

operational principle is uniform
always the same actions, irrespective of context



concept: **Group** (Keynote)

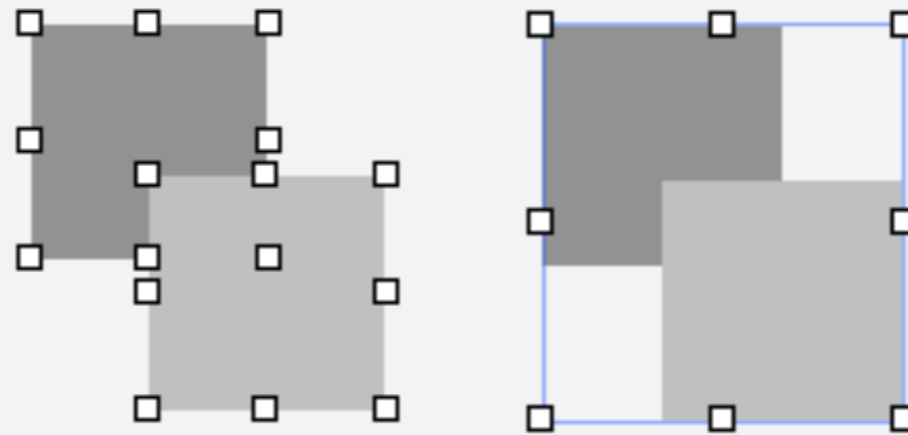
purpose: treat set as one

OP: ... select(objs); group(); mutate()...

← quantified over state & args

what makes a usable concept?

operational principle is uniform
always the same actions, irrespective of context



concept: **Group** (Keynote)

purpose: treat set as one

OP: ... select(objs); group(); mutate()...

quantified over state & args

unless objs contains a text body object

non-uniformity range

	A	B
1	item	cost
2	apples	\$4
3	bananas	\$2
4	grapes	\$6
5	kiwis	\$3
6		
7		

Formula bar: $\cdot fx$ SUM B2:B5

	A	B
1	item	cost
2	apples	\$4
3	bananas	\$2
4	cherries	
5	grapes	\$6
6	kiwis	\$3
7		
8		\$15

Formula bar: $\cdot fx$ SUM B2:B5

non-uniformity range

concept: **Range** (Numbers)

purpose: define formula over adjustable group of cells

OP: ... define formula over range... select(c) in range... add(direction)...formula updated

	A	B
1	item	cost
2	apples	\$4
3	bananas	\$2
4	grapes	\$6
5	kiwis	\$3
6		
7		

• fx SUM B2:B5

	A	B
1	item	cost
2	apples	\$4
3	bananas	\$2
4	cherries	
5	grapes	\$6
6	kiwis	\$3
7		
8		\$15

non-uniformity range

concept: **Range** (Numbers)

purpose: define formula over adjustable group of cells

OP: ... define formula over range... select(c) in range... add(direction)...formula updated

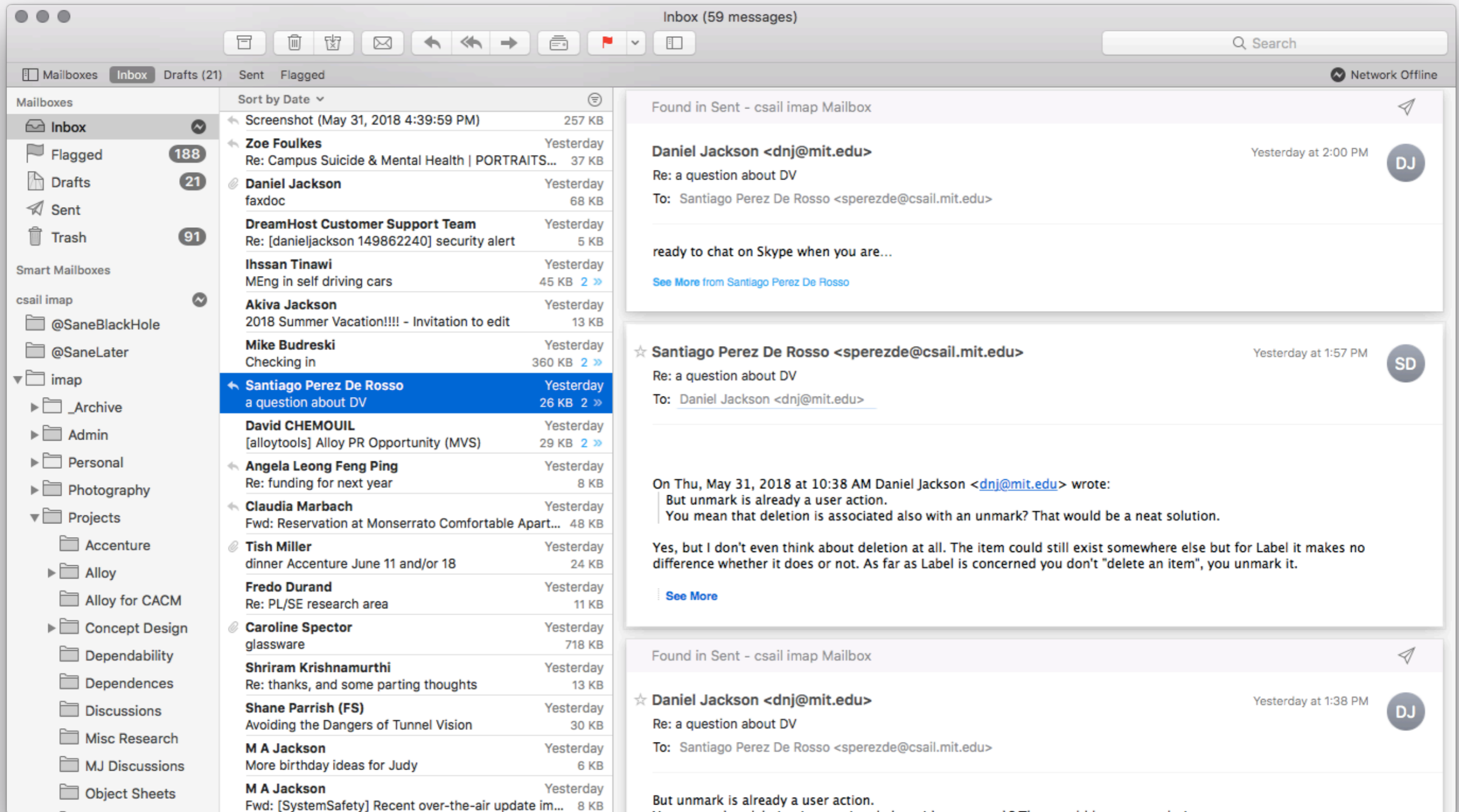
unless range cell c is at top of range and dir is above or...

	A	B
1	item	cost
2	apples	\$4
3	bananas	\$2
4	grapes	\$6
5	kiwis	\$3
6		
7		

• fx SUM B2:B5

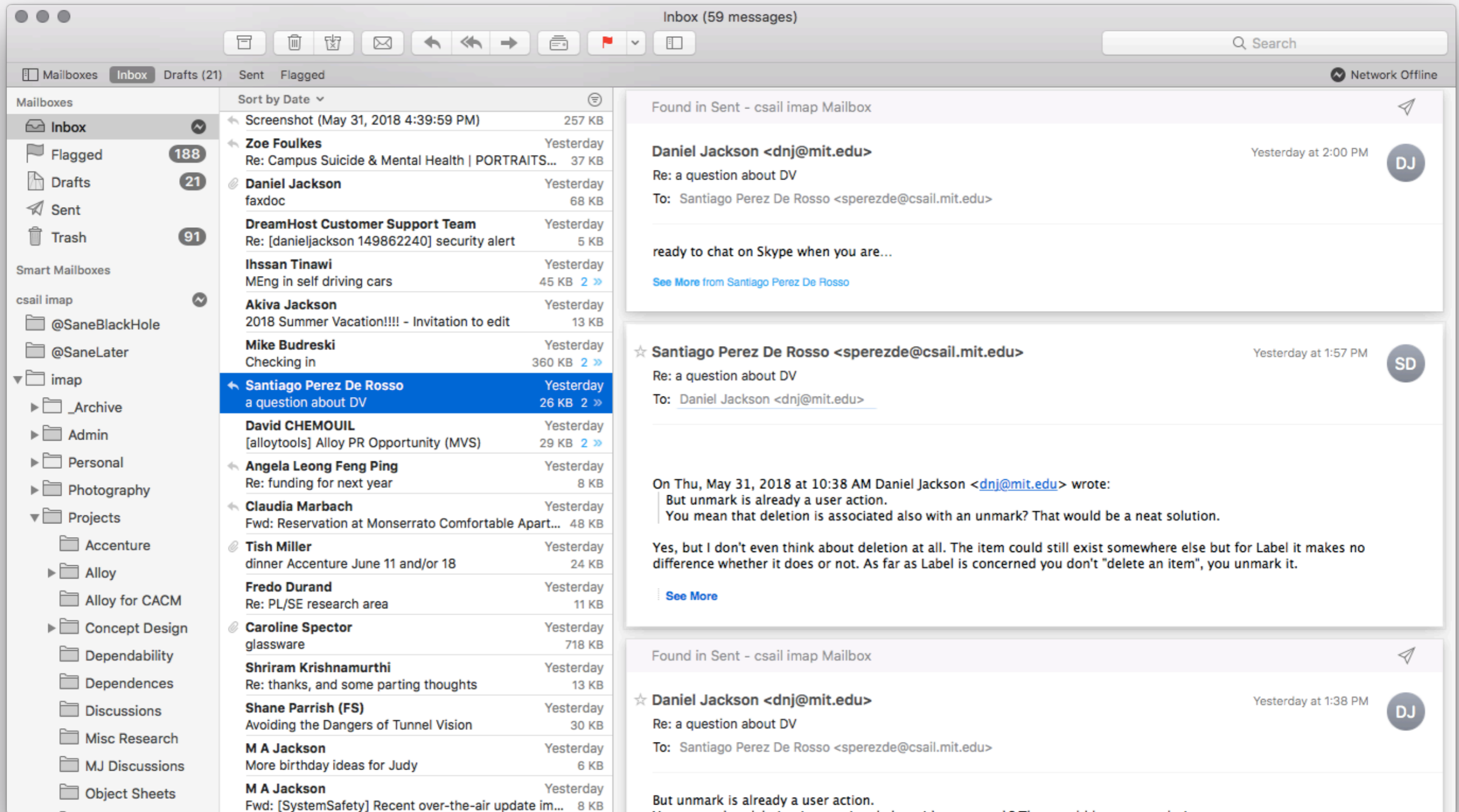
	A	B
1	item	cost
2	apples	\$4
3	bananas	\$2
4	cherries	
5	grapes	\$6
6	kiwis	\$3
7		
8		\$15

non-uniformity conversation



action applied to every message in conversation

non-uniformity conversation



action applied to every message in conversation **unless** message in other folder or action is reply ...

kinds of non-uniformity

Keynote grouping **unless** objs contains a text body object

Fuji aspect ratio setting **unless** set to raw only mode

Dropbox share folder **unless** folder is ancestor or descendant of shared folder

Git branch **unless** working directory contains uncommitted file or...

Twitter mention **unless** mention includes first character of tweet

kinds of non-uniformity

*varies
over type*

Keynote grouping **unless** objs contains a text body object

Fuji aspect ratio setting **unless** set to raw only mode

Dropbox share folder **unless** folder is ancestor or descendant of shared folder

Git branch **unless** working directory contains uncommitted file or...

Twitter mention **unless** mention includes first character of tweet

kinds of non-uniformity

*varies
over type*

Keynote grouping **unless** objs contains a text body object

*varies over
mode*

Fuji aspect ratio setting **unless** set to raw only mode

Dropbox share folder **unless** folder is ancestor or descendant of shared folder

Git branch **unless** working directory contains uncommitted file or...

Twitter mention **unless** mention includes first character of tweet

kinds of non-uniformity

*varies
over type*

Keynote grouping **unless** objs contains a text body object

*varies over
mode*

Fuji aspect ratio setting **unless** set to raw only mode

*varies
over state*

Dropbox share folder **unless** folder is ancestor or descendant of shared folder

*varies
over state*

Git branch **unless** working directory contains uncommitted file or...

Twitter mention **unless** mention includes first character of tweet

kinds of non-uniformity

*varies
over type*

Keynote grouping **unless** objs contains a text body object

*varies over
mode*

Fuji aspect ratio setting **unless** set to raw only mode

*varies
over state*

Dropbox share folder **unless** folder is ancestor or descendant of shared folder

*varies
over state*

Git branch **unless** working directory contains uncommitted file or...

*varies
over arg*

Twitter mention **unless** mention includes first character of tweet

the
genericity
rule

how concepts get applied

NY Times

StackExchange

Amazon

upvote

notification

related

how concepts get applied

NY Times

StackExchange

Amazon

upvote

comments

notification

related

how concepts get applied

NY Times

StackExchange

Amazon

upvote

comments

answers

notification

related

how concepts get applied

NY Times

StackExchange

Amazon

upvote

comments

answers

reviews

notification

related

how concepts get applied

NY Times

StackExchange

Amazon

upvote

comments

answers

reviews

notification

breaking news

related

how concepts get applied

NY Times

StackExchange

Amazon

upvote

comments

answers

reviews

notification

breaking news

replies

related

how concepts get applied

upvote

notification

related

NY Times

comments

breaking news

StackExchange

answers

replies

Amazon

reviews

when shipped

how concepts get applied

	NY Times	StackExchange	Amazon
upvote	comments	answers	reviews
notification	breaking news	replies	when shipped
related	articles		

how concepts get applied

	NY Times	StackExchange	Amazon
upvote	comments	answers	reviews
notification	breaking news	replies	when shipped
related	articles	questions	

how concepts get applied

	NY Times	StackExchange	Amazon
upvote	comments	answers	reviews
notification	breaking news	replies	when shipped
related	articles	questions	items

why reuse a concept?

familiarity

users will get it

save work

design options known

no surprises

misfits anticipated

why reuse a concept?

familiarity
users will get it

save work
design options known

no surprises
misfits anticipated

options for upvote?

why reuse a concept?

familiarity
users will get it

save work
design options known

no surprises
misfits anticipated

options for upvote?

misfit of notification?

the genericity rule

reusing a well-known generic concept is usually preferable to inventing one

the genericity rule

reusing a well-known generic concept is usually preferable to inventing one

in Powerpoint

► overloading (7)

► uniformity (5)

▼ generics (4)

27



generic
concepts

28 ☆



how concepts get applied

	on face	underlying	even
specific	specific	specific	specific
generalized	generalized	specific	generalized
abstract	abstract	generalized	abstract

29 ☆



why reuse a concept?

- functionality
- easy to learn
- easy to teach
- designs are consistent
- no cognitive load
- easy to understand

30



reuse and not

► integrity (35)

the genericity rule

reusing a well-known generic concept is usually preferable to inventing one

in Powerpoint

► overloading (7)
► uniformity (5)
▼ generics (4)
27
28 ☆
29 ☆
30
► integrity (35)

The screenshot shows a vertical navigation pane for a PowerPoint presentation. It lists several sections: 'overloading (7)', 'uniformity (5)', and 'generics (4)'. Under 'generics', there are four slides: slide 27 titled 'generic concepts', slide 28 titled 'how concepts get applied' (marked with a star), slide 29 titled 'why reuse a concept?' (marked with a star), and slide 30. At the bottom, there is a section for 'integrity (35)'.

in Keynote

14 ○○○
15
22
27
28 ○○○
29 ○○○
30 ○○○

The screenshot shows a vertical sequence of slides from a Keynote presentation. Slide 14 is titled 'generic: evaluating a design' and includes three small images. Slide 15 is titled 'kinds of overloading'. Slide 22 is titled 'the uniformity rule'. Slide 27 is titled 'generic concepts'. Slide 28 is titled 'how concepts get applied' and includes a table. Slide 29 is titled 'why reuse a concept?' and includes a diagram. Slide 30 is titled 'reuse vs. inventing' and includes a diagram. Each slide has a progress indicator (three circles) at the bottom left.

the genericity rule

reusing a well-known generic concept is usually preferable to inventing one

in Powerpoint



in Keynote



what would you call this concept?

the genericity rule

reusing a well-known generic concept is usually preferable to inventing one

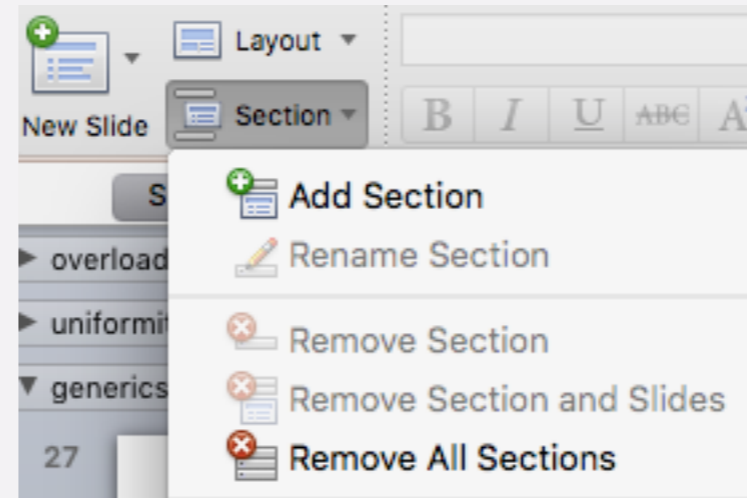
in Powerpoint



in Keynote



Powerpoint commands



what would you call this concept?

the genericity rule

reusing a well-known generic concept is usually preferable to inventing one

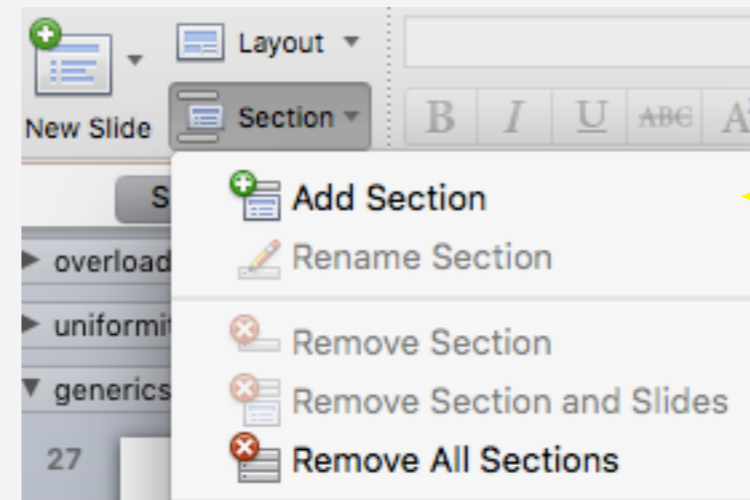
in Powerpoint



in Keynote



Powerpoint commands



what role does slide selection play in add?

what would you call this concept?

concept

composition

example: reservation

name

reservation

purpose

make access to shared resource reliable

structure

reserved: bool = false

behavior

```
reserve()  
  reserved := true  
use ()  
  reserved => reserved := false  
cancel ()  
  reserved => reserved := false
```

tactic

if reserve() and no cancel then can use()

example: reservation

name	reservation
purpose	make access to shared resource reliable
structure	reserved: bool = false
behavior	reserve() reserved := true use () reserved => reserved := false cancel () reserved => reserved := false
tactic	if reserve() and no cancel then can use()

structure
defines
state space

example: reservation

name	reservation
purpose	make access to shared resource reliable
structure	reserved: bool = false
behavior	reserve() reserved := true use () reserved => reserved := false cancel () reserved => reserved := false
tactic	if reserve() and no cancel then can use()

structure
defines
state space

actions
give a labeled transition relation
defining a trace set

example: reservation

name	reservation
purpose	make access to shared resource reliable
structure	reserved: bool = false
behavior	reserve() reserved := true use () reserved => reserved := false cancel () reserved => reserved := false
tactic	if reserve() and no cancel then can use()

structure
defines
state space

actions
give a labeled transition relation
defining a trace set

```
{<>,
<reserve>,
<reserve, cancel>,
<reserve, use>,
<reserve, use, cancel>,
...
}
```


example: reservation

name

reservation

purpose

make access to shared resource reliable

structure

reserved: bool = false

behavior

```
reserve()  
  reserved := true  
use ()  
  reserved => reserved := false  
cancel ()  
  reserved => reserved := false
```

tactic

if reserve() and no cancel then can use()

tactic defines a
property of the
trace set

structure
defines
state space

actions
give a labeled transition relation
defining a trace set

```
{<>,  
<reserve>,  
<reserve, cancel>,  
<reserve, use>,  
<reserve, use, cancel>,  
...  
}
```

example: authentication

name	authentication
purpose	identify participant in interaction
structure	ok: bool = false
behavior	login() ok := true logout () ok => ok := false auth () ok =>
tactic	if login() and no logout() then can auth()

example: authentication

name	authentication
purpose	identify participant in interaction
structure	ok: bool = false
behavior	login() ok := true logout () ok => ok := false auth () ok =>
tactic	if login() and no logout() then can auth()

what are the traces?

example: authentication

name	authentication
purpose	identify participant in interaction
structure	ok: bool = false
behavior	login() ok := true logout () ok => ok := false auth () ok =>
tactic	if login() and no logout() then can auth()

what are the traces?

```
{<>,
<login>,
<login, auth>,
<login, auth, auth>,
<login, logout>,
<login, auth, logout>,
...
}
```

composing concepts

application

MyReservationApp

includes

reservation, authentication

behavior

login

authentication.login

logout

authentication.logout

reserve:

reservation.reserve

authentication.auth

cancel:

reservation.cancel

authentication.auth

use:

reservation.use

composing concepts

application

MyReservationApp

includes

reservation, authentication

behavior

login
 authentication.login
logout
 authentication.logout
reserve:
 reservation.reserve
 authentication.auth
cancel:
 reservation.cancel
 authentication.auth
use:
 reservation.use

concepts
used

composing concepts

application MyReservationApp

includes reservation, authentication

behavior
login
 authentication.login
logout
 authentication.logout
reserve:
 reservation.reserve
 authentication.auth
cancel:
 reservation.cancel
 authentication.auth
use:
 reservation.use

concepts used

action of app is binding of concept actions

binding: Action -> Concept -> Action

```
{(login, authentication, login),  
(logout, authentication, logout),  
(reserve, reservation, reserve),  
(reserve, authentication, auth),  
(cancel, reservation, cancel),  
(cancel, authentication, auth),  
(use, reservation, use)}
```

semantics of composition

semantics of composition

traces of **reservation**

```
{<>,
<reserve>,
<reserve, cancel>,
<reserve, use>,
<reserve, use, cancel>,
...
}
```

semantics of composition

traces of **reservation**

```
{<>,
<reserve>,
<reserve, cancel>,
<reserve, use>,
<reserve, use, cancel>,
...
}
```

traces of **authentication**

```
{<>,
<login>,
<login, auth>,
<login, auth, auth>,
<login, logout>,
<login, auth, logout>,
...
}
```

semantics of composition

traces of **reservation**

```
{<>,
<reserve>,
<reserve, cancel>,
<reserve, use>,
<reserve, use, cancel>,
...
}
```

traces of **authentication**

```
{<>,
<login>,
<login, auth>,
<login, auth, auth>,
<login, logout>,
<login, auth, logout>,
...
}
```

binding: Action -> Concept -> Action

```
{(login, authentication, login),
(logout, authentication, logout),
(reserve, reservation, reserve),
(reserve, authentication, auth),
(cancel, reservation, cancel),
(cancel, authentication, auth),
(use, reservation, use)}
```

semantics of composition

traces of **reservation**

```
{<>,
<reserve>,
<reserve, cancel>,
<reserve, use>,
<reserve, use, cancel>,
...
}
```

traces of **authentication**

```
{<>,
<login>,
<login, auth>,
<login, auth, auth>,
<login, logout>,
<login, auth, logout>,
...
}
```

binding: Action -> Concept -> Action

```
{(login, authentication, login),
(logout, authentication, logout),
(reserve, reservation, reserve),
(reserve, authentication, auth),
(cancel, reservation, cancel),
(cancel, authentication, auth),
(use, reservation, use)}
```

map trace t onto concept C with binding B

```
let map(t, C, B) =
map (<>, C, B) = <>
map (append(t, a), C, B) =
  if no C.(a.B) then map(t, C, B)
  else append(map(t, C, B), C.(a.B))
```

semantics of composition

traces of **reservation**

```
{<>,
<reserve>,
<reserve, cancel>,
<reserve, use>,
<reserve, use, cancel>,
...
}
```

traces of **authentication**

```
{<>,
<login>,
<login, auth>,
<login, auth, auth>,
<login, logout>,
<login, auth, logout>,
...
}
```

binding: Action -> Concept -> Action

```
{(login, authentication, login),
(logout, authentication, logout),
(reserve, reservation, reserve),
(reserve, authentication, auth),
(cancel, reservation, cancel),
(cancel, authentication, auth),
(use, reservation, use)}
```

map trace t onto concept C with binding B

```
let map(t, C, B) =
map (<>, C, B) = <>
map (append(t, a), C, B) =
  if no C.(a.B) then map(t, C, B)
  else append(map(t, C, B), C.(a.B))
```

traces are all those consistent with concept traces

```
traces = {t in action* | all C: includes | map(t, C, B) in traces(C)}
```

semantics of composition

traces of **reservation**

```
{<>,
<reserve>,
<reserve, cancel>,
<reserve, use>,
<reserve, use, cancel>,
...
}
```

traces of **authentication**

```
{<>,
<login>,
<login, auth>,
<login, auth, auth>,
<login, logout>,
<login, auth, logout>,
...
}
```

binding: Action -> Concept -> Action

```
{(login, authentication, login),
(logout, authentication, logout),
(reserve, reservation, reserve),
(reserve, authentication, auth),
(cancel, reservation, cancel),
(cancel, authentication, auth),
(use, reservation, use)}
```

map trace t onto concept C with binding B

```
let map(t, C, B) =
map (<>, C, B) = <>
map (append(t, a), C, B) =
  if no C.(a.B) then map(t, C, B)
  else append(map(t, C, B), C.(a.B))
```

traces are all those consistent with concept traces

```
traces = {t in action* | all C: includes | map(t, C, B) in traces(C)}
```

```
{<>,
<login>,
<login, logout>,
<login, reserve>,
<login, reserve, use>,
...
}
```

reservation (again)

name

reservation

purpose

make access to shared resource reliable

structure

slots: Owner -> Slot
holds: User -> Slot

behavior

```
create (o: Owner, s: Slot)
  no slots.s => slots += o -> s

reserve (u: User, o: Owner, s: Slot)
  no holds.s and o -> s in slots => holds += u -> s

cancel (u: User, s: Slot)
  u -> s in holds => holds -= u -> s

use (u: User, o: Owner, s: Slot)
  u -> s in holds and o -> s in slots =>
```

tactic

if create(o,s); reserve(u,o,s); ... **no** cancel(u,s) ... **then can** use(u,o,s)

authentication (again)

name

authentication

purpose

identify participant in interaction

structure

password: User \rightarrow Password
sessions: set User

behavior

register (u: User, p: Password)
 no u.password \Rightarrow password += u \rightarrow p
login (u: User, p: Password)
 u.password = p \Rightarrow sessions += u
logout (u: User)
 u **in** sessions \Rightarrow sessions -= u
auth (u: User)
 u **in** sessions \Rightarrow

tactic

if register(u,p), login(u,p), no logout(u) then can auth(u)

rating

name

rating

purpose

identify participant in interaction

structure

used: User -> Item

rated: User -> Item -> Int

rating: Item -> Int = {i: Item, r: Int | avg (User, rated)}

behavior

use (u: User, i: Item)

used += u -> i

rate (u: User, i: Item, r: Int)

u -> i **in** used => u.rated += i -> r

show (i: Item): Int

result = i.rating

tactic

if user(u,i), rate(u,i,r)... for multiple u... and show(i):r then r is avg of user's ratings

reservation app (again)

application

MyReservationApp

includes

reservation, authentication, rating

behavior

```
register(u,p)
  authentication.register(u,p)
login(u,p)
  authentication.login(u,p)
logout(u)
  authentication.logout(u)
reserve(u,o,s)
  reservation.reserve(u,o,s)
  authentication.auth(u)
use(u,o,s)
  reservation.use(u,o,s)
  rating.use(u,o)
cancel(u,s)
  reservation.cancel(u,s)
  authentication.auth(u)
```

```
rate(u,o,r)
  authentication.auth(u)
  rating.rate(u,o,r)
showRating(o)
  rating.show(o)
```

reservation app (again)

application

MyReservationApp

includes

reservation, authentication, rating

behavior

```
register(u,p)
  authentication.register(u,p)
login(u,p)
  authentication.login(u,p)
logout(u)
  authentication.logout(u)
reserve(u,o,s)
  reservation.reserve(u,o,s)
  authentication.auth(u)
use(u,o,s)
  reservation.use(u,o,s)
  rating.use(u,o)
cancel(u,s)
  reservation.cancel(u,s)
  authentication.auth(u)
```

```
rate(u,o,r)
  authentication.auth(u)
  rating.rate(u,o,r)
showRating(o)
  rating.show(o)
```

ratings are
authenticated

reservation app (again)

application

MyReservationApp

includes

reservation, authentication, rating

behavior

```
register(u,p)
  authentication.register(u,p)
login(u,p)
  authentication.login(u,p)
logout(u)
  authentication.logout(u)
reserve(u,o,s)
  reservation.reserve(u,o,s)
  authentication.auth(u)
use(u,o,s)
  reservation.use(u,o,s)
  rating.use(u,o)
cancel(u,s)
  reservation.cancel(u,s)
  authentication.auth(u)
```

```
rate(u,o,r)
  authentication.auth(u)
  rating.rate(u,o,r)
showRating(o)
  rating.show(o)
```

ratings are
authenticated

can't rate
until you've used
reservation

the
integrity
rule

looking at sent messages in gmail

The screenshot displays the Gmail web interface. At the top left, there is a hamburger menu icon, the Gmail logo, and the word "Gmail". To the right is a search bar labeled "Search mail". In the top right corner, there are icons for a grid view and a notification bell. Below the search bar, there are icons for a checkbox, refresh, and a vertical ellipsis. On the right side of this bar, it shows "1-2 of 2" and navigation arrows. The left sidebar contains a "Compose" button and a list of folders: "Inbox" (highlighted in red with a "1" badge), "Starred", and "Snozed". The main content area shows three tabs: "Primary" (selected), "Social", and "Promotions". Below the tabs, there is a list of two email messages. The first message is from "Google Photos" with a subject line "Ben, finish setting up your Google Photos account" and a date of "Oct 16". The second message is from "Alyssa, me 4" with a subject line "Is JavaScript just Scheme with prototypes and some hacky coercions?" and a date of "Jul 30".

Search mail

Compose

Inbox 1

Starred

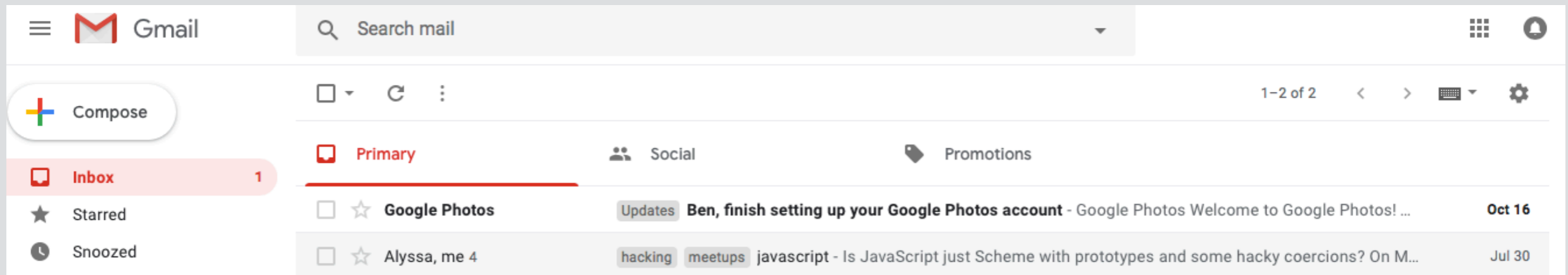
Snozed

Primary Social Promotions

Google Photos **Ben, finish setting up your Google Photos account** - Google Photos Welcome to Google Photos! ... Oct 16

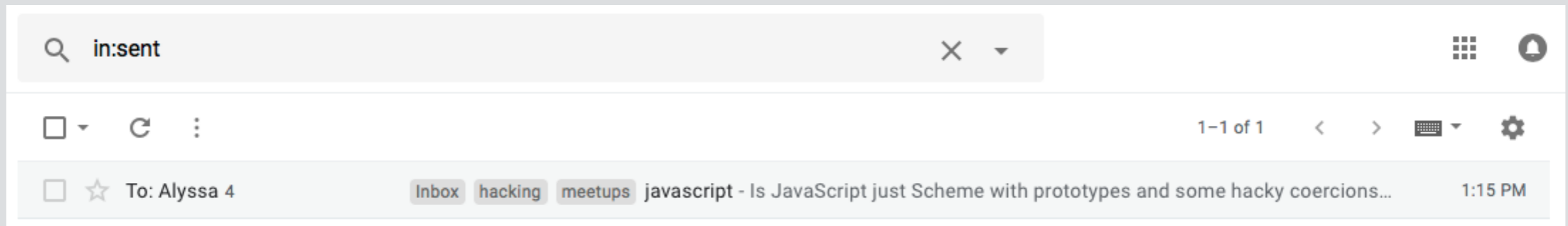
Alyssa, me 4 **Is JavaScript just Scheme with prototypes and some hacky coercions?** On M... Jul 30

looking at sent messages in gmail



This screenshot shows the Gmail inbox interface. The top bar includes the Gmail logo, a search bar with the text "Search mail", and navigation icons. On the left, a sidebar contains "Compose", "Inbox" (with a notification for 1 message), "Starred", and "Snoozed". The main area displays three email categories: "Primary", "Social", and "Promotions". Two email entries are visible in the Primary category:

- Google Photos** (starred): "Ben, finish setting up your Google Photos account - Google Photos Welcome to Google Photos! ..." dated Oct 16. Includes an "Updates" label.
- Alyssa, me 4** (starred): "javascript - Is JavaScript just Scheme with prototypes and some hacky coercions? On M..." dated Jul 30. Includes "hacking" and "meetups" labels.



This screenshot shows the search results for the query "in:sent". The search bar at the top contains "in:sent" and a close button. The results list shows one email entry:

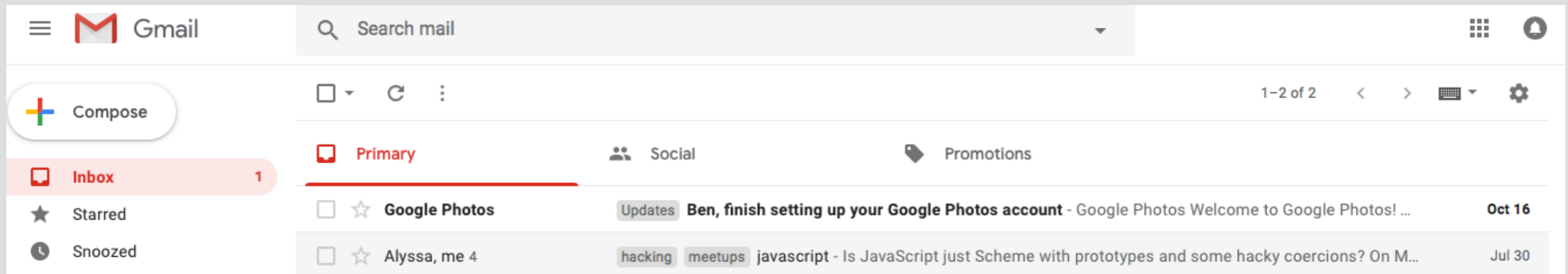
- To: Alyssa 4** (starred): "javascript - Is JavaScript just Scheme with prototypes and some hacky coercions..." dated 1:15 PM. Includes "Inbox", "hacking", and "meetups" labels.

looking at sent messages in gmail

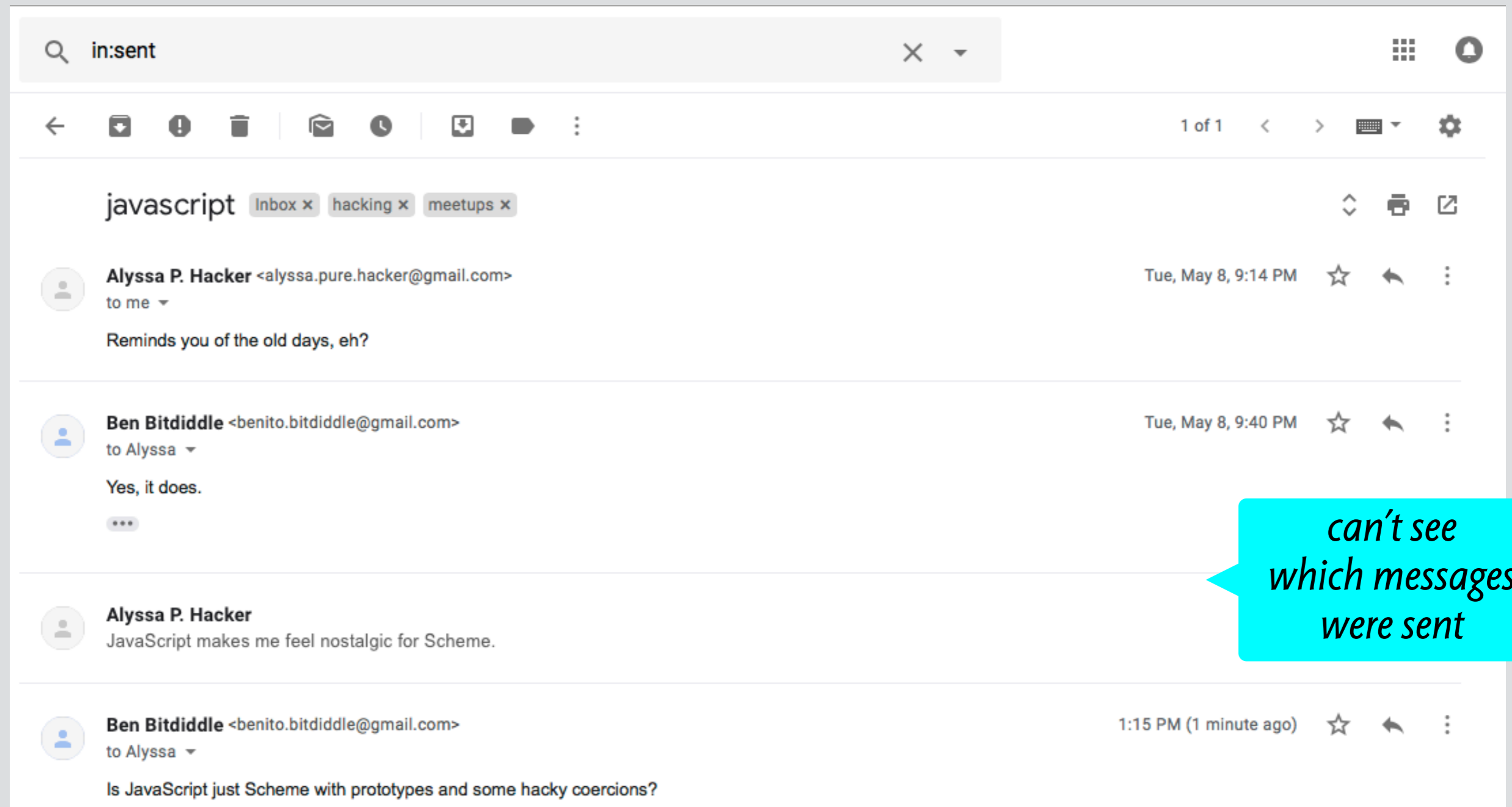
The screenshot shows the Gmail inbox interface. At the top left, there is a hamburger menu icon, the Gmail logo, and the text "Gmail". To the right is a search bar labeled "Search mail". Below the search bar, there are navigation icons for "Compose", "Inbox" (with a red notification bubble containing the number "1"), "Starred", and "Snoozed". The main area displays a list of email categories: "Primary", "Social", and "Promotions". Two email entries are visible: one from "Google Photos" with the subject "Ben, finish setting up your Google Photos account" dated "Oct 16", and another from "Alyssa, me 4" with the subject "javascript - Is JavaScript just Scheme with prototypes and some hacky coercions?" dated "Jul 30".

This screenshot shows a detailed view of a Gmail conversation thread. The search bar at the top contains the filter "in:sent". The thread title is "javascript" with tags for "Inbox", "hacking", and "meetups". The first message is from "Alyssa P. Hacker" to "me" on "Tue, May 8, 9:14 PM", with the text "Reminds you of the old days, eh?". The second message is from "Ben Bitdiddle" to "Alyssa" on "Tue, May 8, 9:40 PM", with the text "Yes, it does.". The third message is from "Alyssa P. Hacker" on "Mon, Jul 30, 1:24 PM", with the text "JavaScript makes me feel nostalgic for Scheme.". The fourth message is from "Ben Bitdiddle" to "Alyssa" on "1:15 PM (1 minute ago)", with the text "Is JavaScript just Scheme with prototypes and some hacky coercions?".

looking at sent messages in gmail



A screenshot of the Gmail inbox interface. The top bar shows the Gmail logo, a search bar with the text "Search mail", and a notification bell. Below the search bar, there are navigation icons for "Compose", "Inbox" (with a red badge showing "1"), "Starred", and "Snoozed". The main inbox area shows a list of emails. The first email is from "Google Photos" with the subject "Ben, finish setting up your Google Photos account" and a date of "Oct 16". The second email is from "Alyssa, me 4" with the subject "javascript - Is JavaScript just Scheme with prototypes and some hacky coercions?" and a date of "Jul 30".

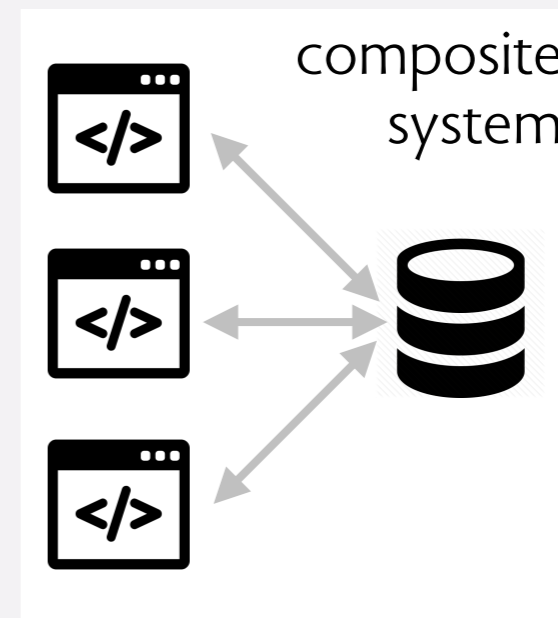
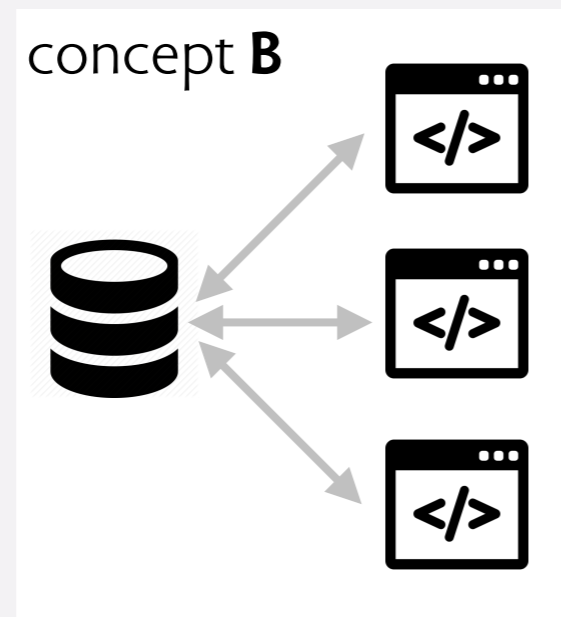
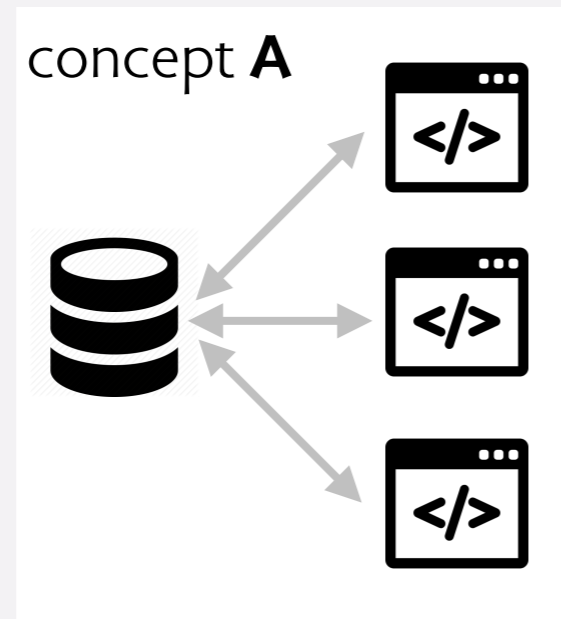


A screenshot of Gmail search results for the query "in:sent". The search bar at the top shows "in:sent" and a close button. Below the search bar, there are navigation icons and a "1 of 1" indicator. The search results are filtered by "javascript" and show three messages. The first message is from "Alyssa P. Hacker" to "me" with the subject "Reminds you of the old days, eh?" and a date of "Tue, May 8, 9:14 PM". The second message is from "Ben Bitdiddle" to "Alyssa" with the subject "Yes, it does." and a date of "Tue, May 8, 9:40 PM". The third message is from "Alyssa P. Hacker" with the subject "JavaScript makes me feel nostalgic for Scheme." and a date of "1:15 PM (1 minute ago)". The fourth message is from "Ben Bitdiddle" to "Alyssa" with the subject "Is JavaScript just Scheme with prototypes and some hacky coercions?" and a date of "1:15 PM (1 minute ago)".

*can't see
which messages
were sent*

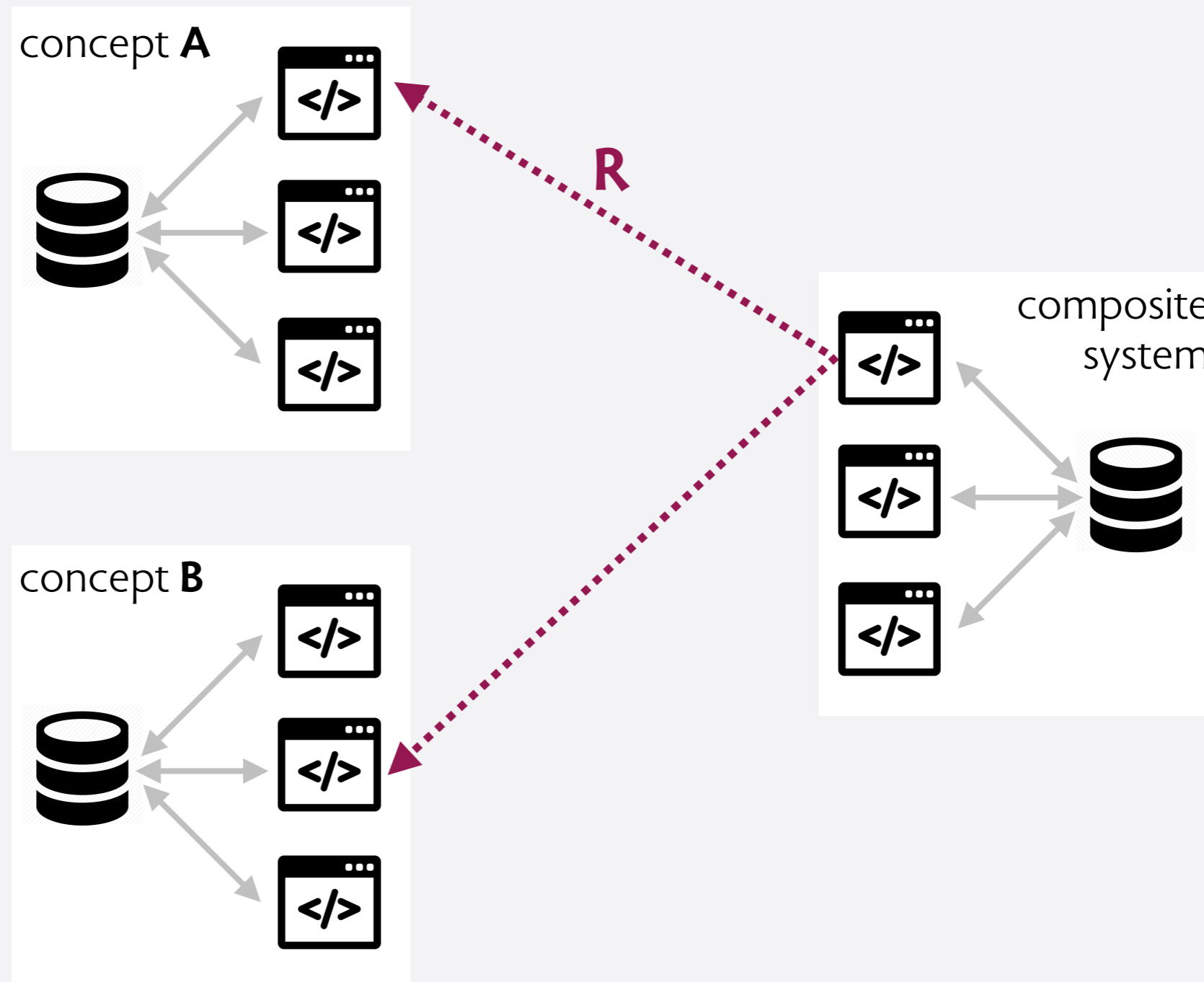
interpreting composite behavior

each action in composite system
interpreted as zero or more actions in each concept



interpreting composite behavior

each action in composite system
interpreted as zero or more actions in each concept



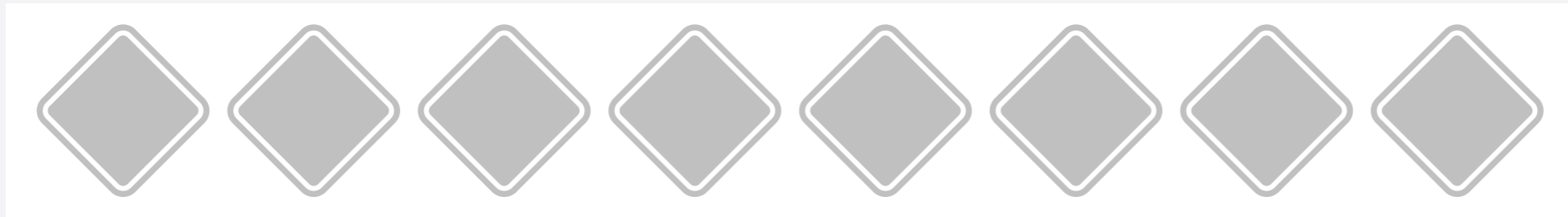
the integrity rule

when concepts are combined, each concept's behavior and OP should still apply

a simple criterion

projected behavior must satisfy concept spec:

$\forall c: \text{concept} \mid \forall t: \text{traces}(\text{sys}) \mid R_c(t) \in \text{traces}(c)$



system

concept **A**

concept **B**

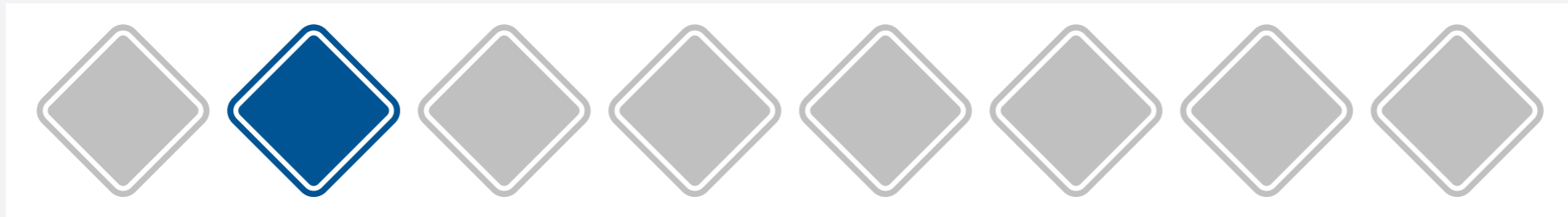
the integrity rule

when concepts are combined, each concept's behavior and OP should still apply

a simple criterion

projected behavior must satisfy concept spec:

$\forall c: \text{concept} \mid \forall t: \text{traces}(\text{sys}) \mid R_c(t) \in \text{traces}(c)$



system

concept **A**

concept **B**

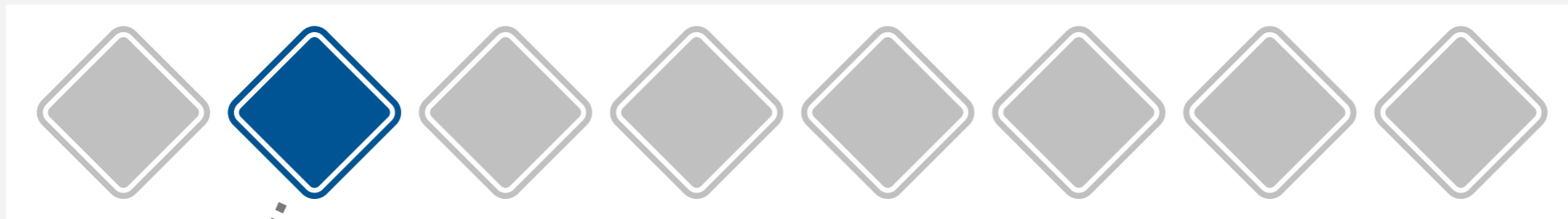
the integrity rule

when concepts are combined, each concept's behavior and OP should still apply

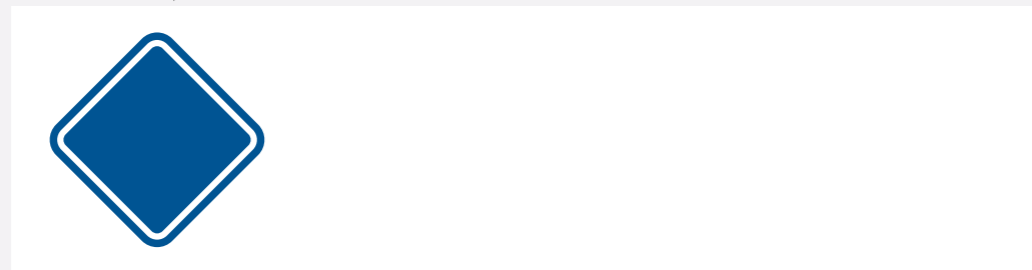
a simple criterion

projected behavior must satisfy concept spec:

$$\forall c: \text{concept} \mid \forall t: \text{traces}(\text{sys}) \mid R_c(t) \in \text{traces}(c)$$



system



concept **A**

concept **B**

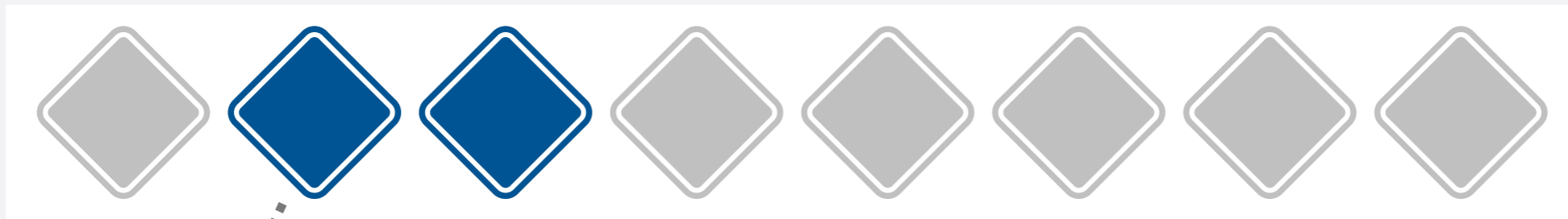
the integrity rule

when concepts are combined, each concept's behavior and OP should still apply

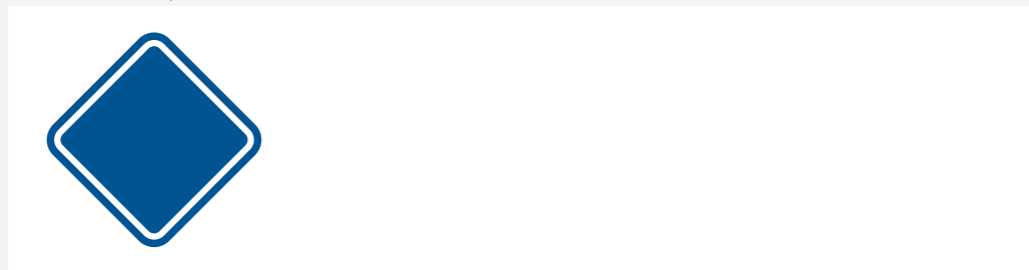
a simple criterion

projected behavior must satisfy concept spec:

$\forall c: \text{concept} \mid \forall t: \text{traces}(\text{sys}) \mid R_c(t) \in \text{traces}(c)$



system



concept **A**

concept **B**

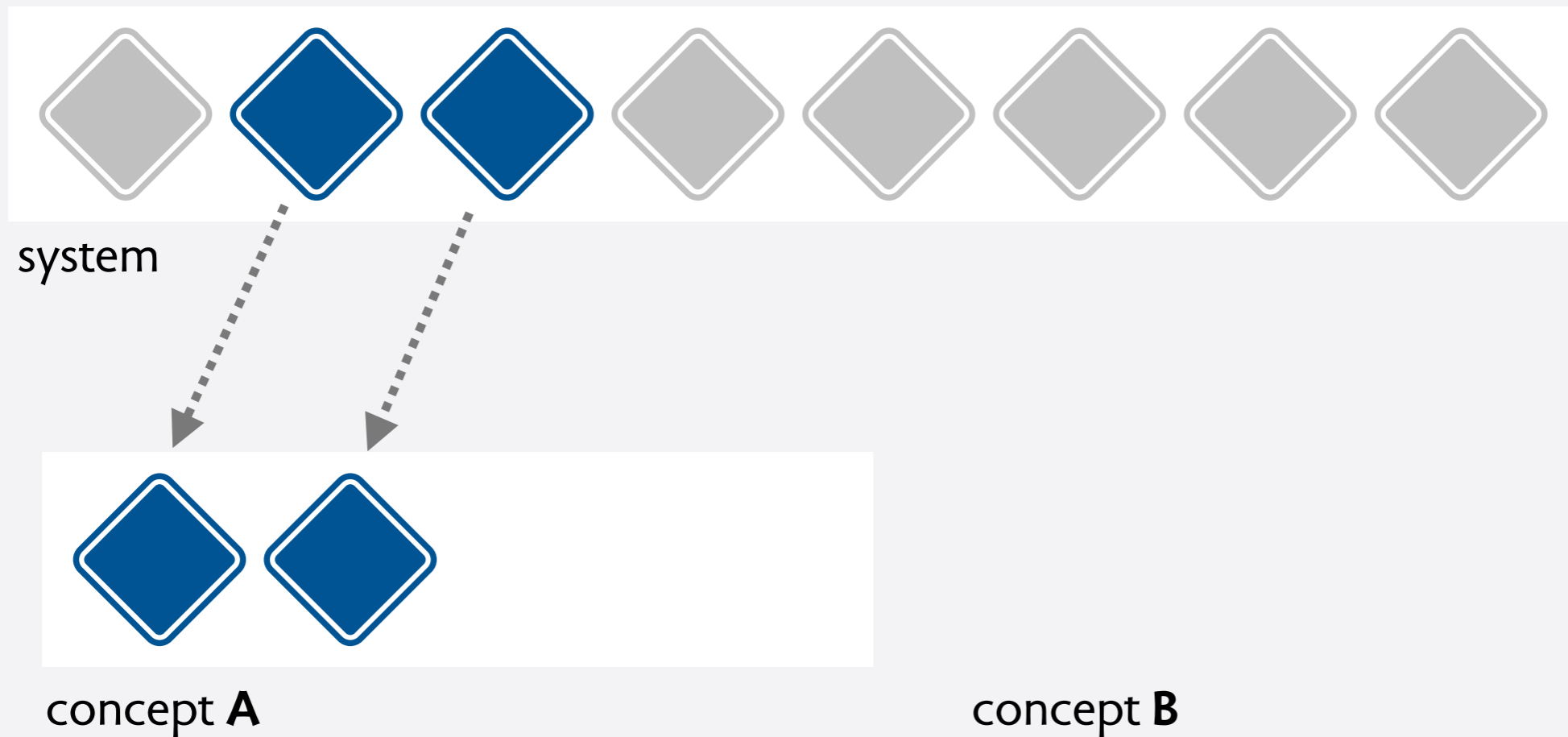
the integrity rule

when concepts are combined, each concept's behavior and OP should still apply

a simple criterion

projected behavior must satisfy concept spec:

$\forall c: \text{concept} \mid \forall t: \text{traces}(\text{sys}) \mid R_c(t) \in \text{traces}(c)$



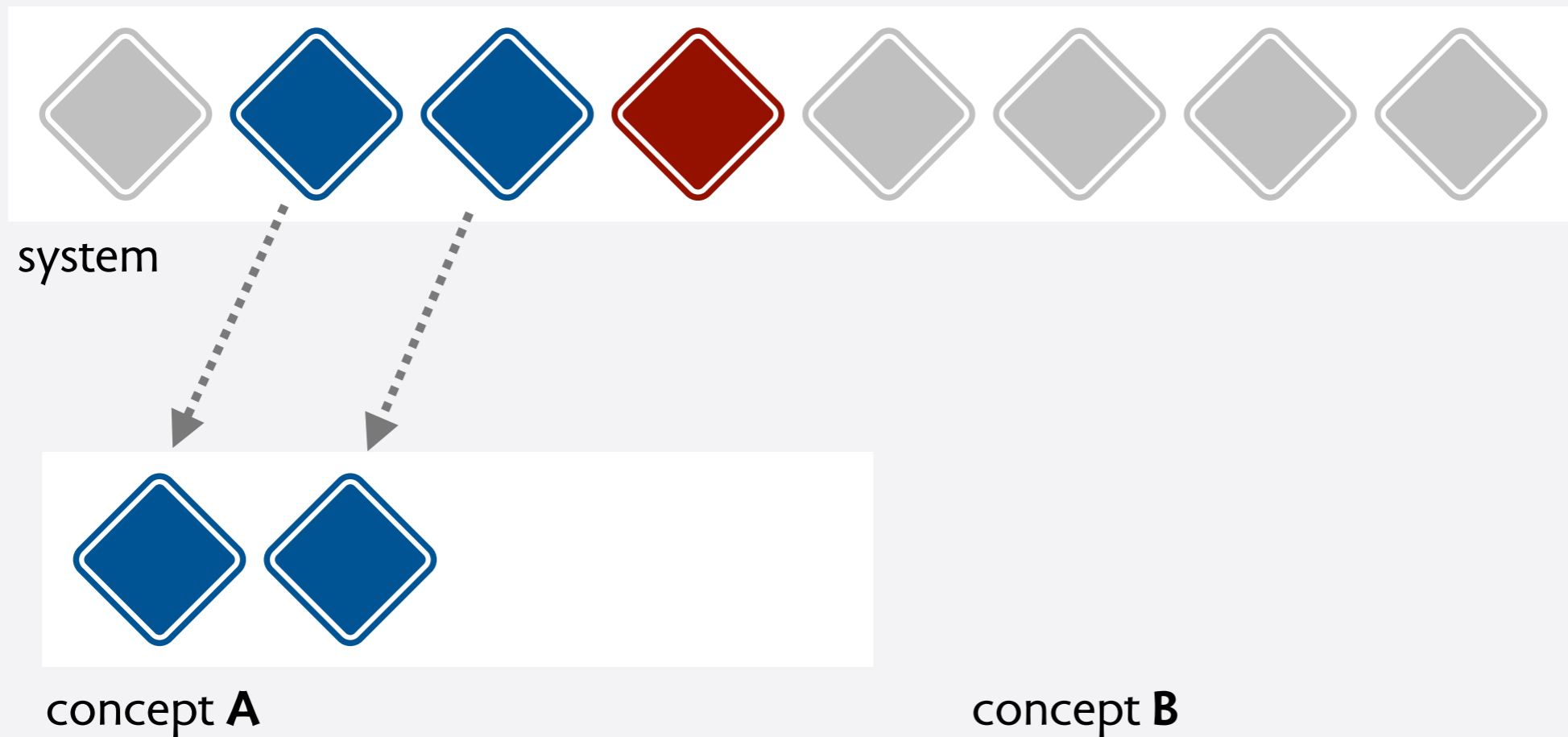
the integrity rule

when concepts are combined, each concept's behavior and OP should still apply

a simple criterion

projected behavior must satisfy concept spec:

$\forall c: \text{concept} \mid \forall t: \text{traces}(\text{sys}) \mid R_c(t) \in \text{traces}(c)$



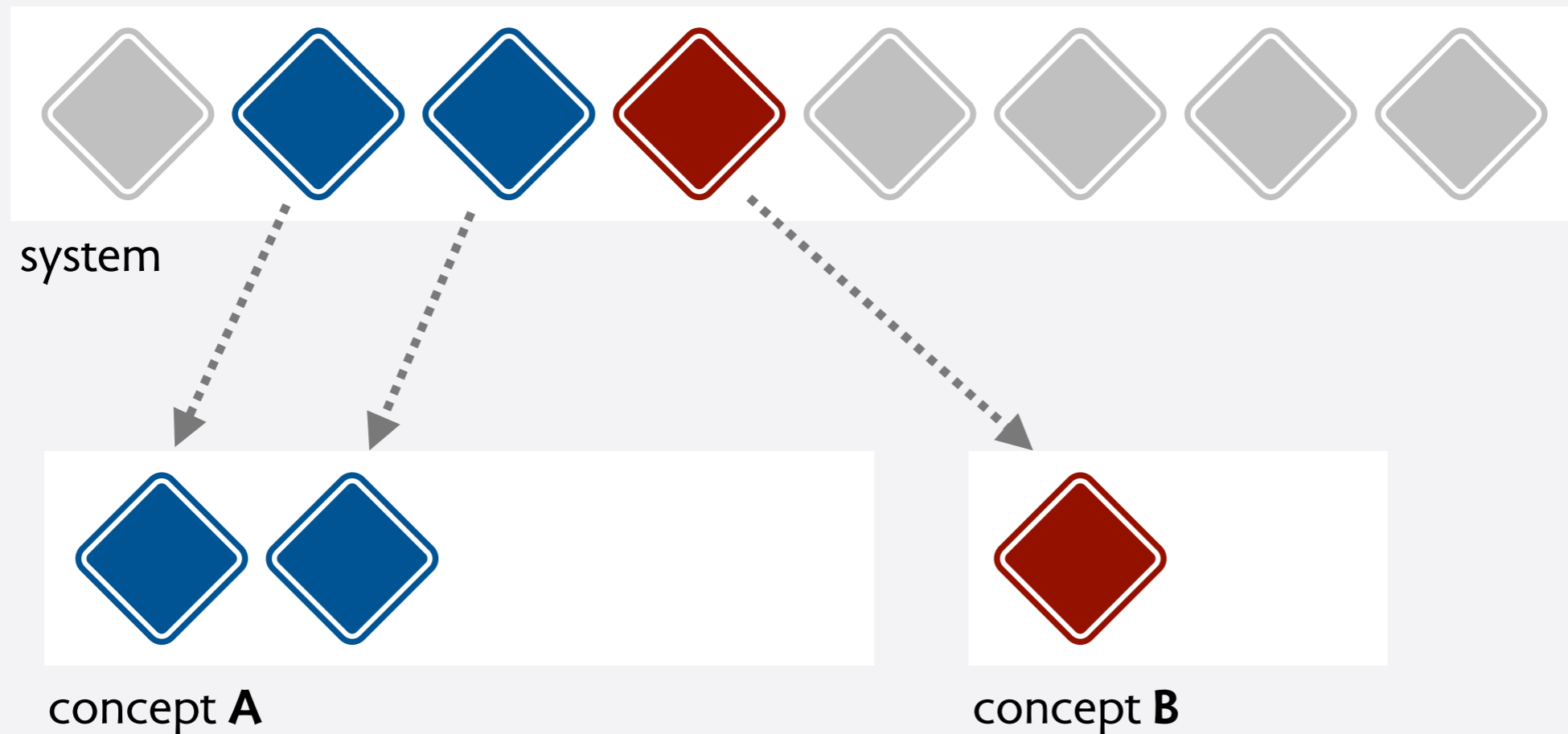
the integrity rule

when concepts are combined, each concept's behavior and OP should still apply

a simple criterion

projected behavior must satisfy concept spec:

$\forall c: \text{concept} \mid \forall t: \text{traces}(\text{sys}) \mid R_c(t) \in \text{traces}(c)$



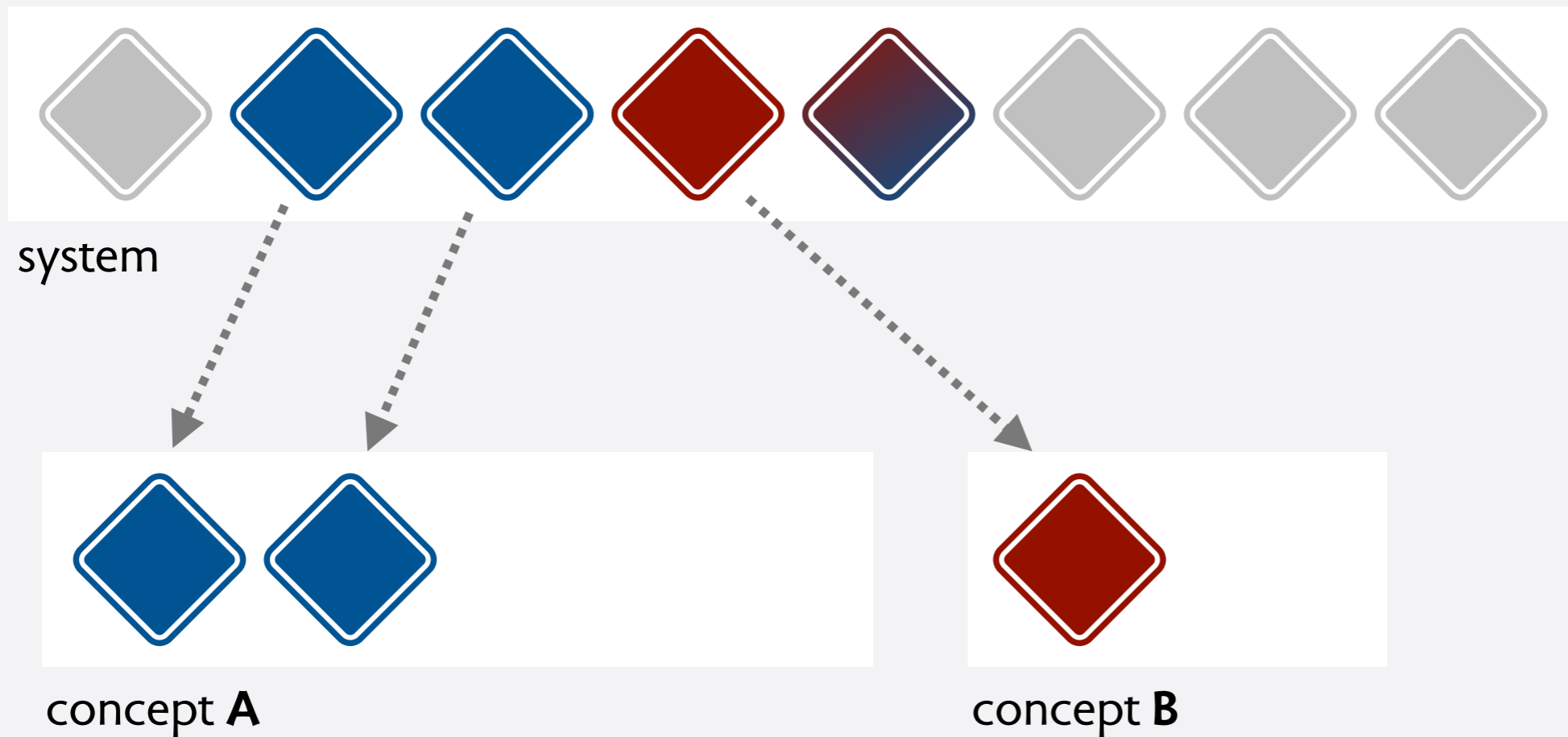
the integrity rule

when concepts are combined, each concept's behavior and OP should still apply

a simple criterion

projected behavior must satisfy concept spec:

$\forall c: \text{concept} \mid \forall t: \text{traces}(\text{sys}) \mid R_c(t) \in \text{traces}(c)$



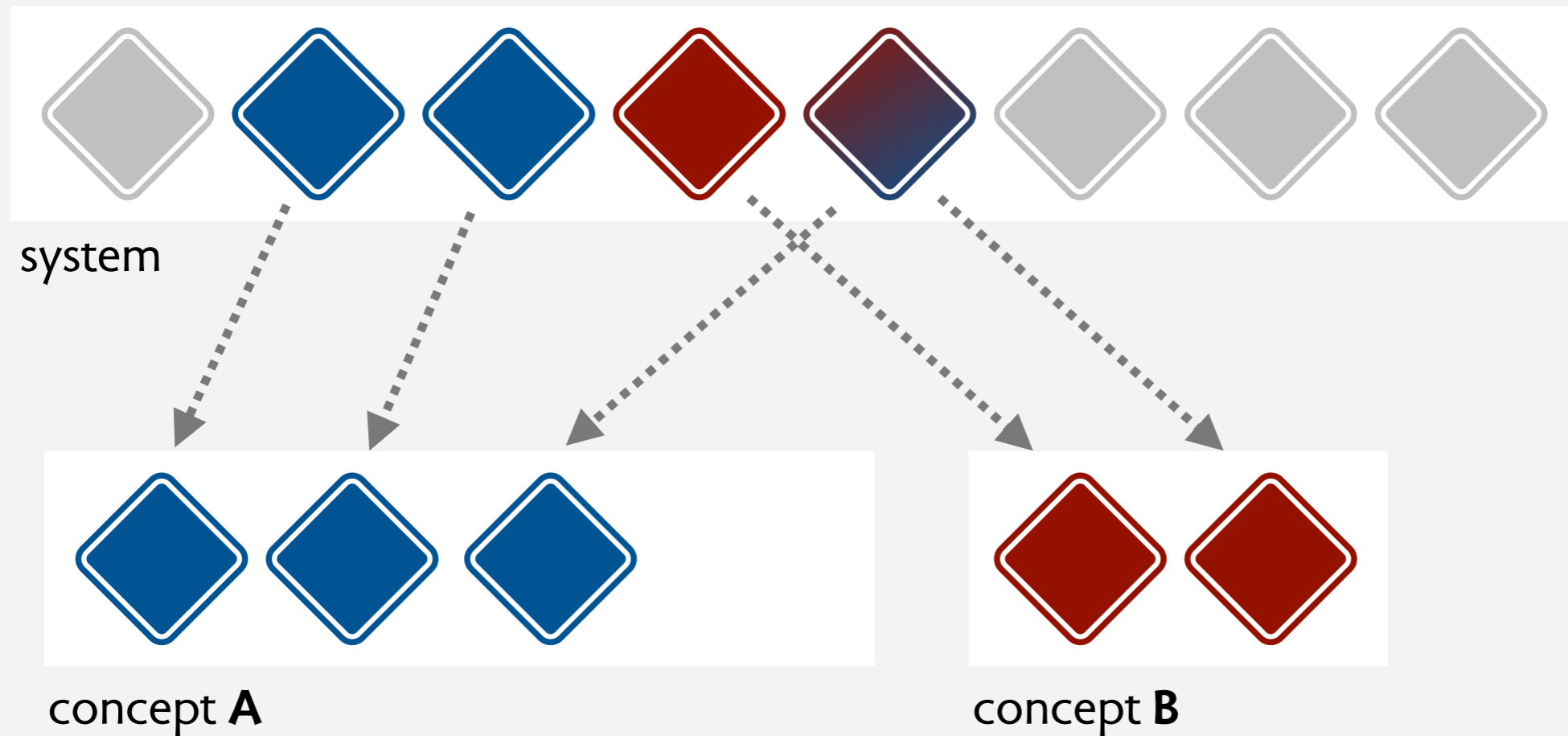
the integrity rule

when concepts are combined, each concept's behavior and OP should still apply

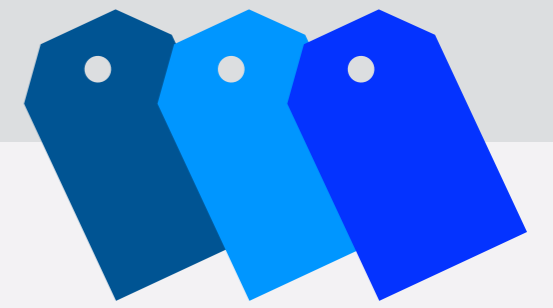
a simple criterion

projected behavior must satisfy concept spec:

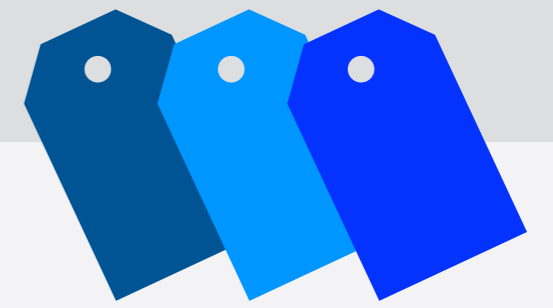
$\forall c: \text{concept} \mid \forall t: \text{traces}(\text{sys}) \mid R_c(t) \in \text{traces}(c)$



the label concept

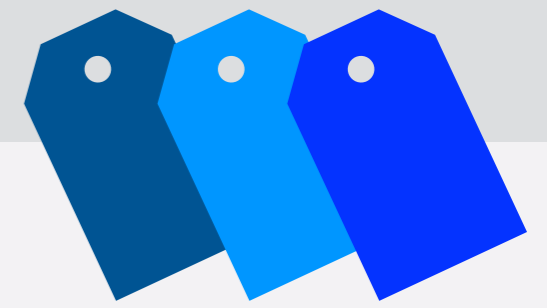


the label concept



name label

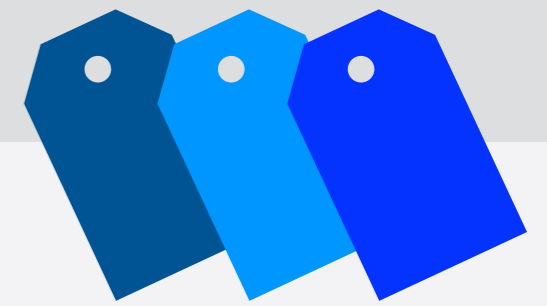
the label concept



name label

purpose organize items for easy retrieval

the label concept

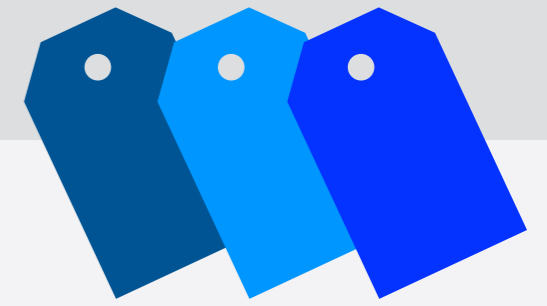


name label

purpose organize items for easy retrieval

structure labels: X -> Label

the label concept



name label

purpose organize items for easy retrieval

structure labels: $X \rightarrow \text{Label}$

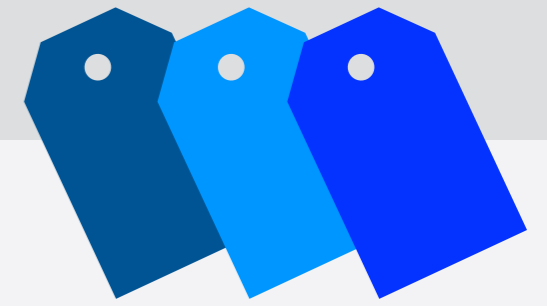
behavior

```
mark (x: X, p: Label)
  labels += x -> p

unmark (x: X, p: Label)
  p in x.labels => labels -= x -> p

find (ps: set Label): set X
  result = {x | ps in x.labels}
```

the label concept



name label

purpose organize items for easy retrieval

structure labels: X -> Label

behavior

```
mark (x: X, p: Label)
  labels += x -> p

unmark (x: X, p: Label)
  p in x.labels => labels -= x -> p

find (ps: set Label): set X
  result = {x | ps in x.labels}
```

tactic

```
if mark(x,p); find(p):xs then x in xs
if no mark(x,P); find(p):xs then x !in xs
```

conversation breaks label

conversation breaks label

in:sent

1 of 1

javascript Inbox x hacking x meetups x

Alyssa P. Hacker <alyssa.pure.hacker@gmail.com>
to me ▾
Reminds you of the old days, eh?
Tue, May 8, 9:14 PM

Ben Bitdiddle <benito.bitdiddle@gmail.com>
to Alyssa ▾
Yes, it does.
Tue, May 8, 9:40 PM

Alyssa P. Hacker
JavaScript makes me feel nostalgic for Scheme.
Mon, Jul 30, 1:24 PM

Ben Bitdiddle <benito.bitdiddle@gmail.com>
to Alyssa ▾
Is JavaScript just Scheme with prototypes and some hacky coercions?
1:15 PM (1 minute ago)

conversation breaks label

The screenshot shows an email interface with a search bar at the top containing "in:sent". Below the search bar is a toolbar with various icons for email actions. The main content area displays a conversation thread with four messages. A red text box is overlaid on the first two messages, containing the text: "when message m is sent Label.mark(m, 'sent') occurs implicitly".

Search: in:sent

1 of 1

javascript Inbox x hacking x meetups x

Alyssa P. Hacker <alyssa.pure.hacker@gmail.com>
to me ▾
Reminds you of the old days, eh?

Ben Bitdiddle <benito.bitdiddle@gmail.com>
to Alyssa ▾
Yes, it does.

Alyssa P. Hacker
JavaScript makes me feel nostalgic for Scheme.

Ben Bitdiddle <benito.bitdiddle@gmail.com>
to Alyssa ▾
Is JavaScript just Scheme with prototypes and some hacky coercions?

when message m is sent
Label.mark(m, 'sent')
occurs implicitly

Tue, May 8, 9:14 PM

Tue, May 8, 9:40 PM

Mon, Jul 30, 1:24 PM

1:15 PM (1 minute ago)

conversation breaks label

in:sent

javascript Inbox x hacking x meetups x

Alyssa P. Hacker <alyssa.pure.hacker@gmail.com>
to me ▾
Reminds you of the old days, eh?

Ben Bitdiddle <benito.bitdiddle@gmail.com>
to Alyssa ▾
Yes, it does.

Alyssa P. Hacker
JavaScript makes me feel nostalgic for Scheme.

Ben Bitdiddle <benito.bitdiddle@gmail.com>
to Alyssa ▾
Is JavaScript just Scheme with prototypes and some hacky coercions?

when message m is sent
`Label.mark(m, 'sent')`
occurs implicitly

when Sent link is clicked
`Label.find('sent'):ms`
OCCURS

1 of 1

Tue, May 8, 9:14 PM

Tue, May 8, 9:40 PM

Mon, Jul 30, 1:24 PM

1:15 PM (1 minute ago)

conversation breaks label

The screenshot shows a Gmail inbox with a search filter 'in:sent' applied. The search results are for a conversation about JavaScript. The messages are as follows:

- Message 1:** From Alyssa P. Hacker to me, dated Tue, May 8, 9:14 PM. Subject: javascript. Content: "Reminds you of the old days, eh?".
- Message 2:** From Ben Bitdiddle to Alyssa, dated Tue, May 8, 9:40 PM. Content: "Yes, it does.".
- Message 3:** From Alyssa P. Hacker, dated Mon, Jul 30, 1:24 PM. Content: "JavaScript makes me feel nostalgic for Scheme.".
- Message 4:** From Ben Bitdiddle to Alyssa, dated 1:15 PM (1 minute ago). Content: "Is JavaScript just Scheme with prototypes and some hacky c...".

Annotations in green boxes explain the 'sent' label:

- when message m is sent
Label.mark(m, 'sent')
occurs implicitly*
- when Sent link is clicked
Label.find('sent'):ms
occurs*
- but ms includes
messages never marked*

integrity violations trash



integrity violations trash

*what happens when
you unmount a drive?*



integrity violations trash

*what happens when
you unmount a drive?*



interaction of Trash and Volume (Apple Finder)
unmount of Volume removes files from Trash
not expressible in terms of Trash actions
a solution: one trash/volume?

deja vu

reversing the process

reversing the process



the same concepts, again & again
post, comment, upvote, notification, ...

hard work to build
libraries often just client- or server-side
easy in a CMS, but structure hard-wired

reversing the process



the same concepts, again & again
post, comment, upvote, notification, ...

hard work to build
libraries often just client- or server-side
easy in a CMS, but structure hard-wired

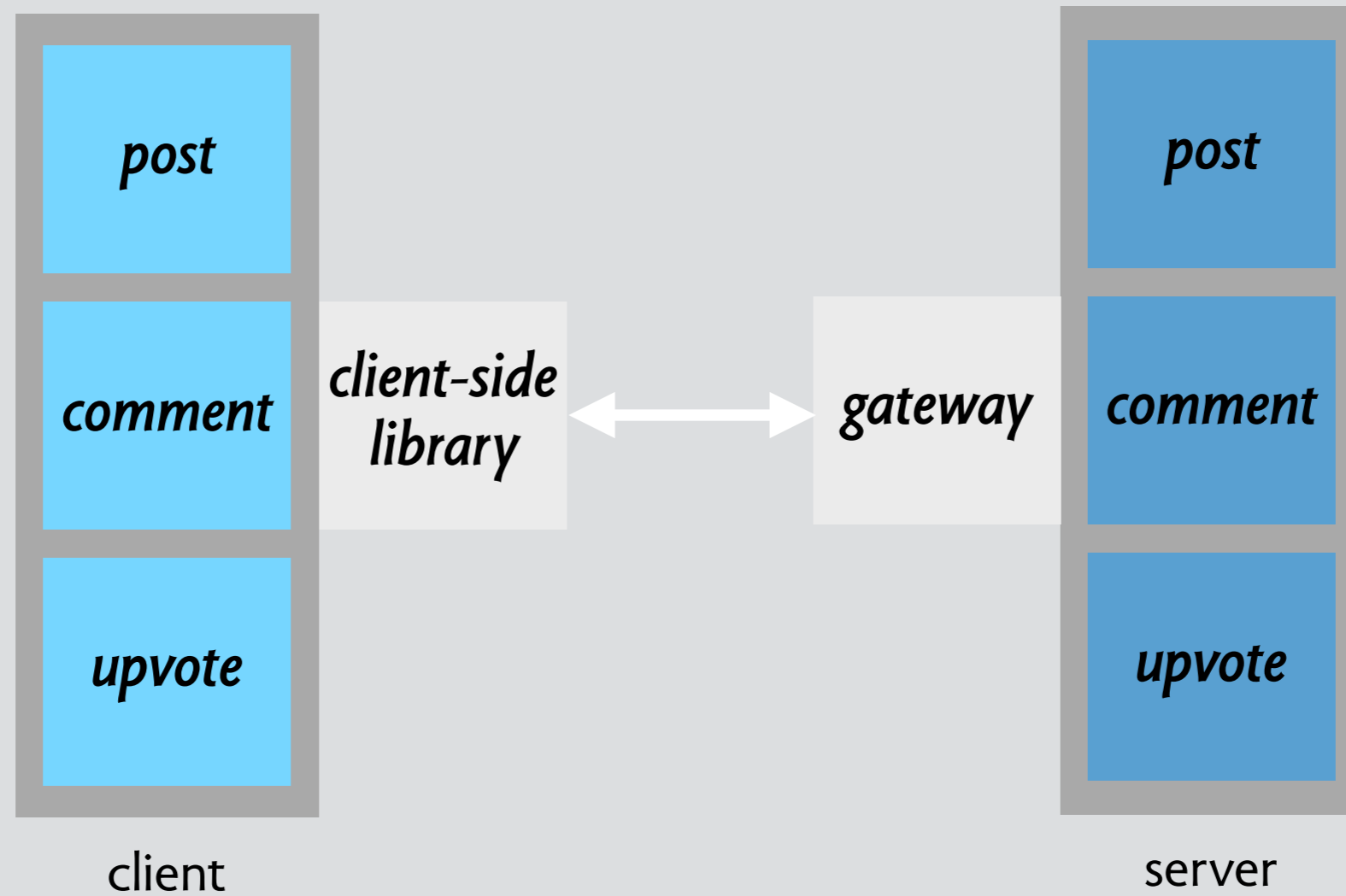


idea: concept cliches
full stack implementation

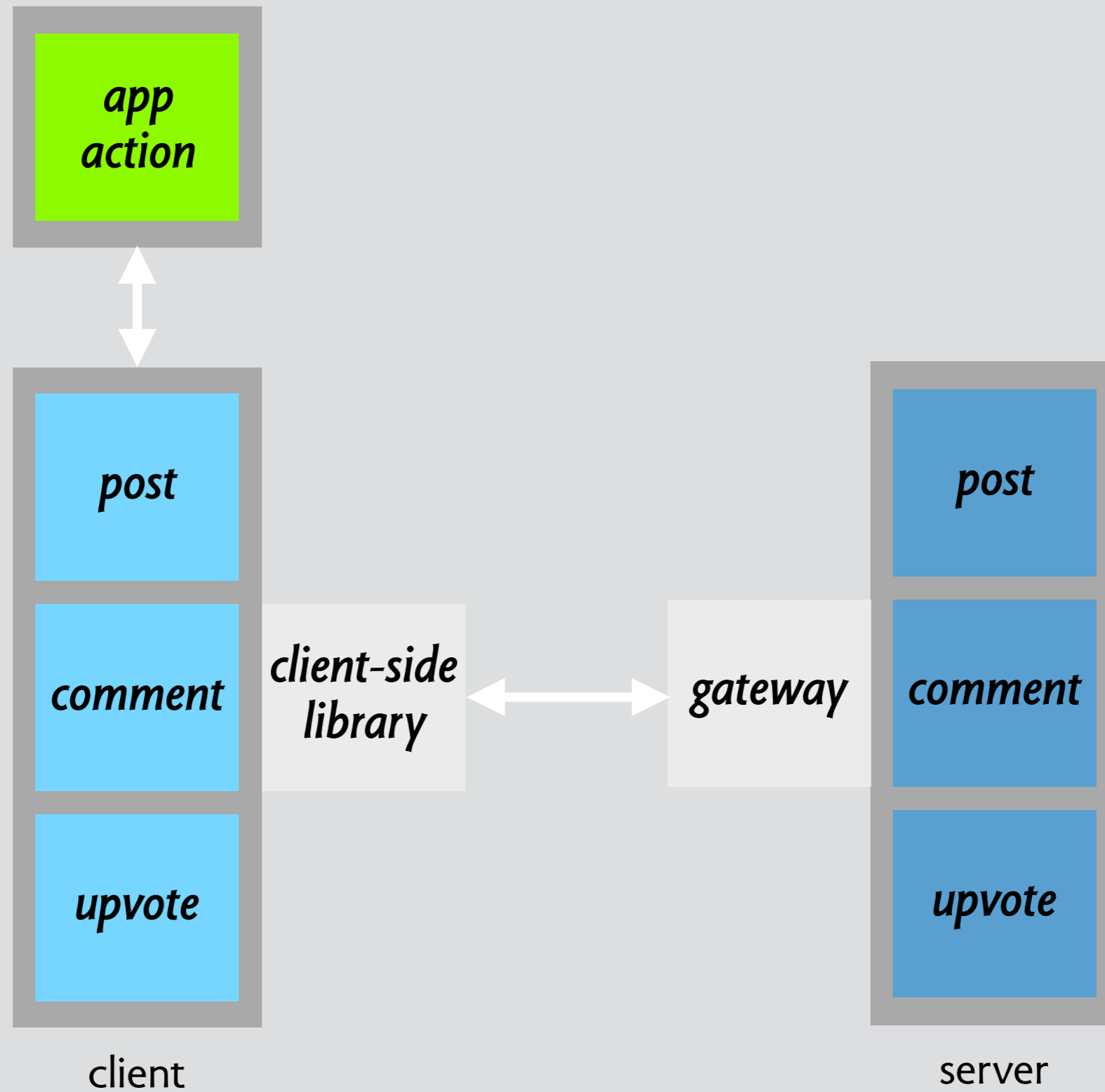
app-specific assembly
in HTML, no JS or backend code

action synchronization
build app action by joining cliché actions

architecture of deja vu



architecture of deja vu



a sample app

DV Hacker News Submit

ben Sign Out

▲ **SPLASH 2018** (2018.splashcon.org)
3 points by eva | comments

▲ **Software Design Group** (sdg.csail.mit.edu)
2 points by eva | comments

▲ **Concept design** (sdg.csail.mit.edu)
1 points by alyssa | comments

▲ **Déjà Vu** (sdg.csail.mit.edu)
1 points by ben | comments

a sample app

DV Hacker News Submit

ben Sign Out

▲ **SPLASH 2018** (2018.splashcon.org)
3 points by eva | comments

▲ **Software Design Group** (sdg.csail.mit.edu)
2 points by eva | comments

▲ **Concept design** (sdg.csail.mit.edu)
1 points by alyssa | comments

▲ **Déjà Vu** (sdg.csail.mit.edu)
1 points by ben | comments

a sample app

auth

DV Hacker News Submit

ben Sign Out

▲ **SPLASH 2018** (2018.splashcon.org)
3 points by eva | comments

▲ **Software Design Group** (sdg.csail.mit.edu)
2 points by eva | comments

▲ **Concept design** (sdg.csail.mit.edu)
1 points by alyssa | comments

▲ **Déjà Vu** (sdg.csail.mit.edu)
1 points by ben | comments

a sample app

auth

DV Hacker News Submit

ben Sign Out

▲ **SPLASH 2018** (2018.splashcon.org)
3 points by eva | comments

▲ **Software Design Group** (sdg.csail.mit.edu)
2 points by eva | comments

▲ **Concept design** (sdg.csail.mit.edu)
1 points by alyssa | comments

▲ **Déjà Vu** (sdg.csail.mit.edu)
1 points by ben | comments

comment

a sample app

auth

DV Hacker News Submit

ben Sign Out

▲ **SPLASH 2018** (2018.splashcon.org)
3 points by eva | comments

property

▲ **Software Design Group** (sdg.csail.mit.edu)
2 points by eva | comments

▲ **Concept design** (sdg.csail.mit.edu)
1 points by alyssa | comments

▲ **Déjà Vu** (sdg.csail.mit.edu)
1 points by ben | comments

comment

a sample app

auth

DV Hacker News Submit

ben Sign Out

▲ **SPLASH 2018** (2018.splashcon.org)
3 points by eva | comments

property

*scoring
(x2)*

▲ **Software Design Group** (sdg.csail.mit.edu)
2 points by eva | comments

▲ **Concept design** (sdg.csail.mit.edu)
1 points by alyssa | comments

▲ **Déjà Vu** (sdg.csail.mit.edu)
1 points by ben | comments

comment

a sample app

auth

DV Hacker News Submit

ben Sign Out

▲ **SPLASH 2018** (2018.splashcon.org)
3 points by eva | comments

property

*scoring
(x2)*

▲ **Software Design Group** (sdg.csail.mit.edu)
2 points by eva | comments

▲ **Concept design** (sdg.csail.mit.edu)
1 points by alyssa | comments

▲ **Déjà Vu** (sdg.csail.mit.edu)
1 points by ben | comments

comment

```
{
  "name": "hackernews",
  "usedCliches": {
    "authentication": {},
    "comment": {},
    "property": {...},
    "scoringposts": {"name": "scoring"}
    "scoringcomments": {"name": "scoring"}
  },
}
```

a sample app

auth

DV Hacker News Submit

ben Sign Out

▲ **SPLASH 2018** (2018.splashcon.org)

3 points by eva | comments

property

*scoring
(x2)*

▲ **Software Design Group** (sdg.csail.mit.edu)

2 points by eva | comments

▲ **Concept design** (sdg.csail.mit.edu)

1 points by alyssa | comments

▲ **Déjà Vu** (sdg.csail.mit.edu)

1 points by ben | comments

comment

```
{
  "name": "hackernews",
  "usedCliches": {
    "authentication": {},
    "comment": {},
    "property": {...},
    "scoringposts": {"name": "scoring"}
    "scoringcomments": {"name": "scoring"}
  },
```

```
"routes": [
  { "route": "", "action": "home" },
  { "route": "news", "action": "home" },
  { "route": "post", "action": "post-detail" },
  { "route": "login", "action": "login" },
  { "route": "submit", "action": "submit-post" }
]
```


home action

DV Hacker News [Submit](#)

[ben](#) [Sign Out](#)

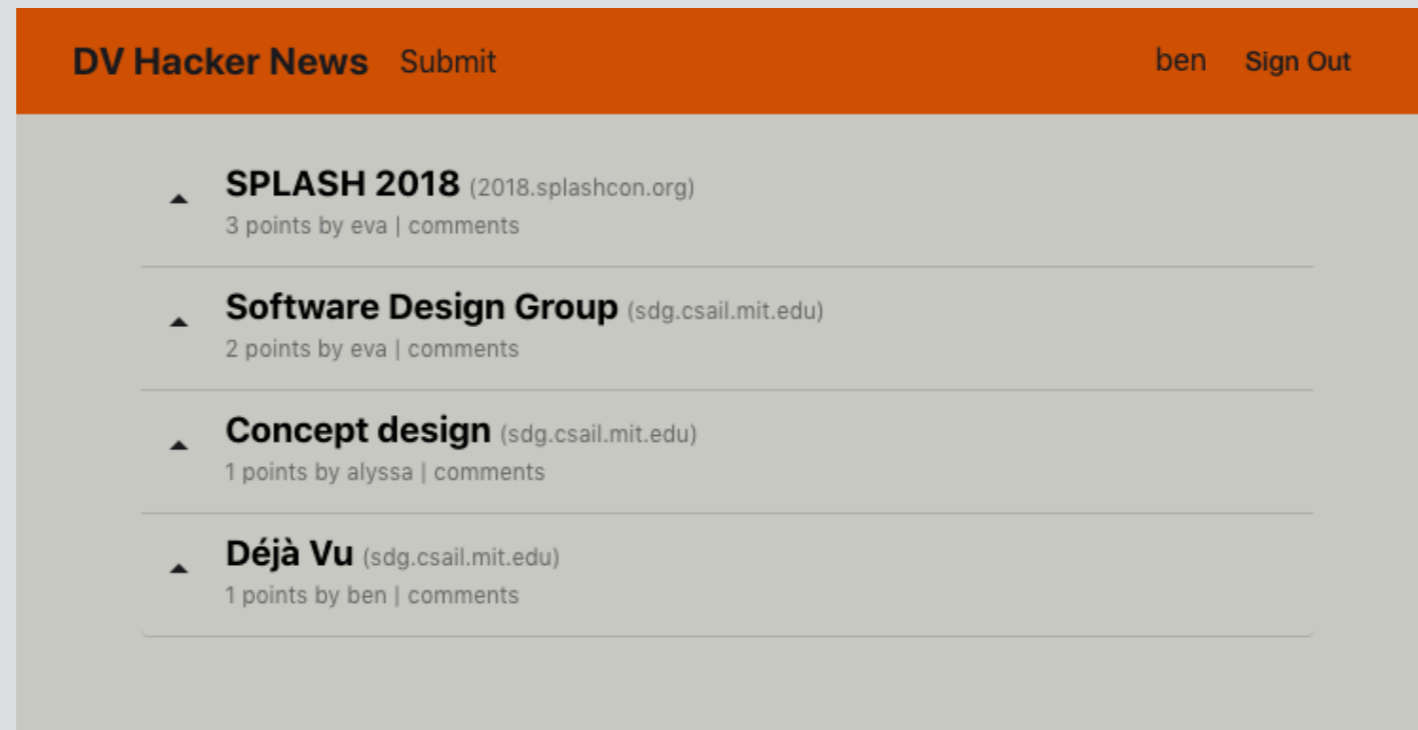
▲ **SPLASH 2018** (2018.splashcon.org)
3 points by eva | [comments](#)

▲ **Software Design Group** (sdg.csail.mit.edu)
2 points by eva | [comments](#)

▲ **Concept design** (sdg.csail.mit.edu)
1 points by alyssa | [comments](#)

▲ **Déjà Vu** (sdg.csail.mit.edu)
1 points by ben | [comments](#)

home action



The screenshot shows the DV Hacker News home page. At the top, there is an orange navigation bar with the text "DV Hacker News" and a "Submit" button on the left, and the user name "ben" and a "Sign Out" link on the right. Below the navigation bar, there is a list of four posts, each with a small upward-pointing triangle icon to its left. The posts are: "SPLASH 2018 (2018.splashcon.org)" with 3 points by eva and a link to comments; "Software Design Group (sdg.csail.mit.edu)" with 2 points by eva and a link to comments; "Concept design (sdg.csail.mit.edu)" with 1 point by alyssa and a link to comments; and "Déjà Vu (sdg.csail.mit.edu)" with 1 point by ben and a link to comments. Each post is separated from the next by a thin horizontal line.

```
<dv.action name="home">
  <hackernews.navbar />
  <div class="main">
    <scoringposts.show-targets-by-score
      noTargetsText="No posts yet"
      showAscending=false
      showScores=false
      showTarget=<hackernews.show-post post=$target id=$id />
    </scoringposts.show-targets-by-score>
  </div>
</dv.action>
```

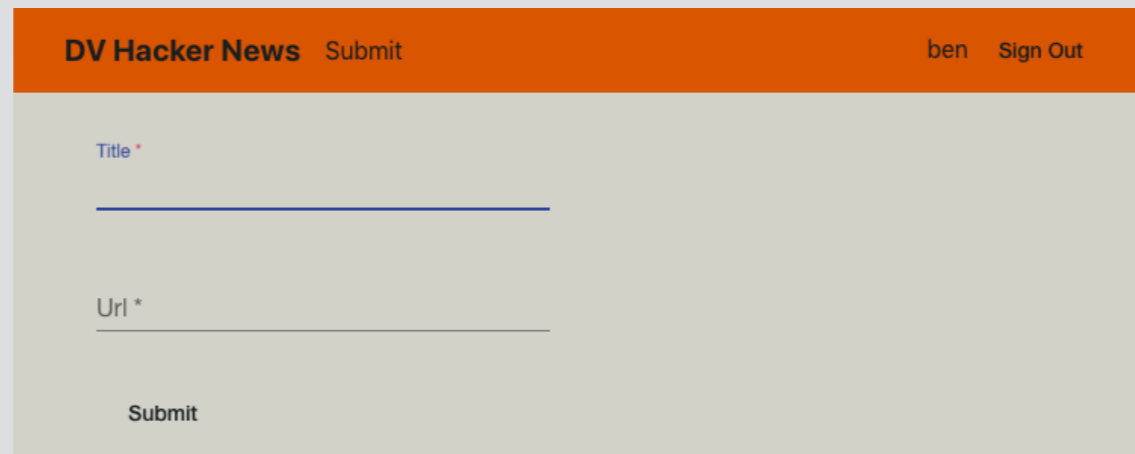
submit post action

DV Hacker News [Submit](#) ben [Sign Out](#)

Title *

Url *

submit post action



DV Hacker News Submit ben Sign Out

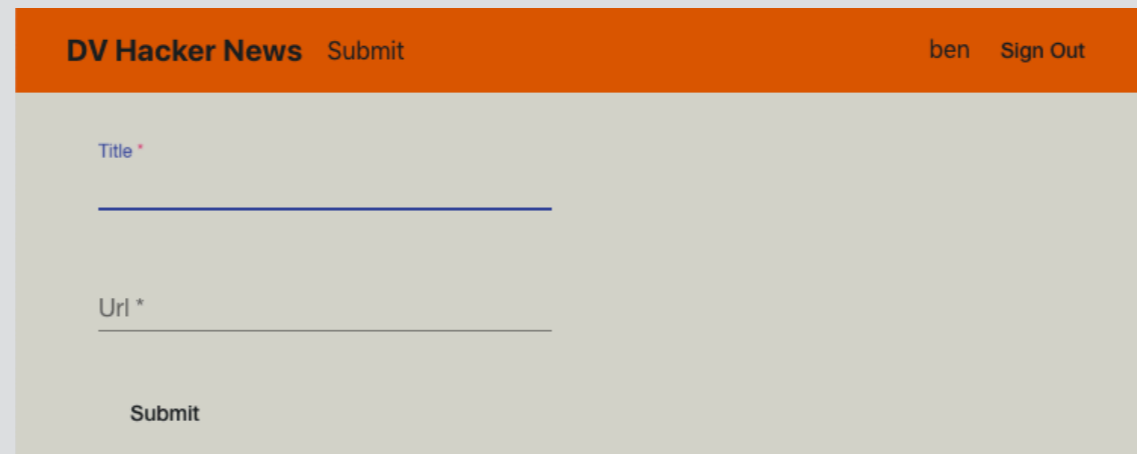
Title *

Url *

Submit

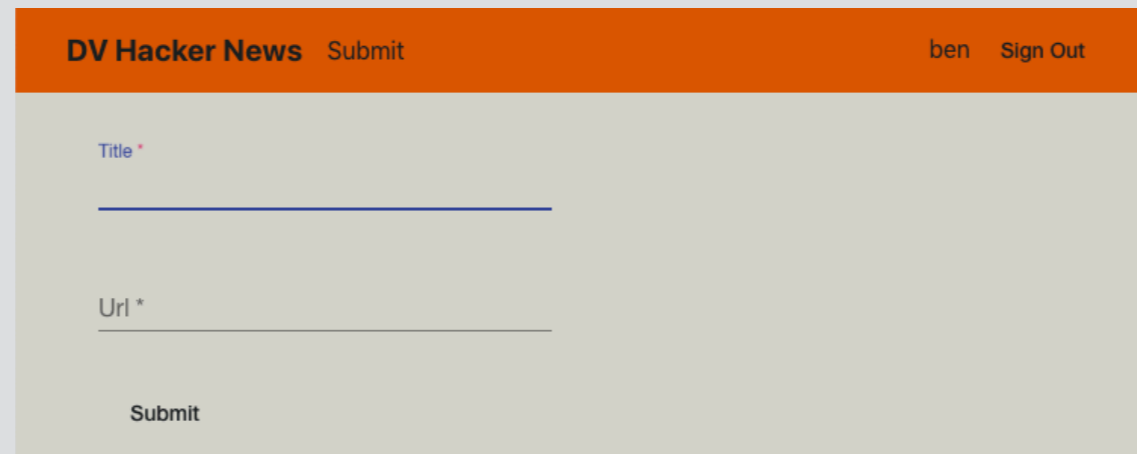
```
<dv.action name="submit-post">  
<hackernews.navbar />  
<div class="main"> <dv.tx>  
  <dv.gen-id />  
  <property.create-object  
    id=dv.gen-id.id  
    initialValue={ author: hackernews.navbar.user.username }  
    showExclude=["author"]  
    buttonLabel="submit"  
    newObjectSavedText="Post submitted" />  
  <scoringposts.create-score  
    targetId=dv.gen-id.id  
    value=0  
    hidden=true />  
  <authentication.authenticate id=hackernews.navbar.user hidden=true />  
  <dv.link href="/item" params={ id: dv.gen-id.id } />  
</dv.tx> </div> </dv.action>
```

submit post action



```
<dv.action name="submit-post">  
<hackernews.navbar />  
<div class="main"> <dv.tx> transaction  
  <dv.gen-id />  
  <property.create-object  
    id=dv.gen-id.id  
    initialValue={ author: hackernews.navbar.user.username }  
    showExclude=["author"]  
    buttonLabel="submit"  
    newObjectSavedText="Post submitted" />  
  <scoringposts.create-score  
    targetId=dv.gen-id.id  
    value=0  
    hidden=true />  
  <authentication.authenticate id=hackernews.navbar.user hidden=true />  
  <dv.link href="/item" params={ id: dv.gen-id.id } />  
</dv.tx> </div> </dv.action>
```

submit post action



```
<dv.action name="submit-post">  
<hackernews.navbar />  
<div class="main"> <dv.tx>  
  <dv.gen-id />  
  <property.create-object  
    id=dv.gen-id.id  
    initialValue={ author: hackernews.navbar.user.username }  
    showExclude=["author"]  
    buttonLabel="submit"  
    newObjectSavedText="Post submitted" />  
  <scoringposts.create-score  
    targetId=dv.gen-id.id  
    value=0  
    hidden=true />  
  <authentication.authenticate id=hackernews.navbar.user hidden=true />  
  <dv.link href="/item" params={ id: dv.gen-id.id } />  
</dv.tx> </div> </dv.action>
```

generate id

transaction

submit post action

DV Hacker News Submit ben Sign Out

Title *

Url *

Submit

```
<dv.action name="submit-post">
<hackernews.navbar />
<div class="main"> <dv.tx>
  <dv.gen-id />
  <property.create-object
    id=dv.gen-id.id
    initialValue={ author: hackernews.navbar.user.username }
    showExclude=["author"]
    buttonLabel="submit"
    newObjectSavedText="Post submitted" />
  <scoringposts.create-score
    targetId=dv.gen-id.id
    value=0
    hidden=true />
  <authentication.authenticate id=hackernews.navbar.user hidden=true />
  <dv.link href="/item" params={ id: dv.gen-id.id } />
</dv.tx> </div> </dv.action>
```

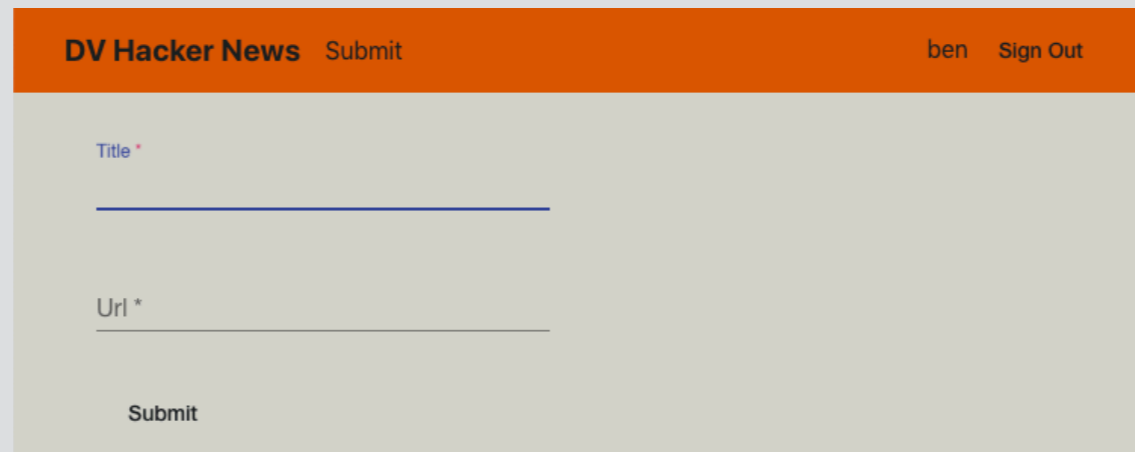
generate id

id used

id used

transaction

submit post action



```
<dv.action name="submit-post">
<hackernews.navbar />
<div class="main"> <dv.tx>
  <dv.gen-id />
  <property.create-object
    id=dv.gen-id.id
    initialValue={ author: hackernews.navbar.user.username }
    showExclude=["author"]
    buttonLabel="submit"
    newObjectSavedText="Post"
  </property.create-object>
  <scoringposts.create-score
    targetId=dv.gen-id.id
    value=0
    hidden=true />
  <authentication.authenticate id=hackernews.navbar.user hidden=true />
  <dv.link href="/item" params={ id: dv.gen-id.id } />
</dv.tx> </div> </dv.action>
```

generate id

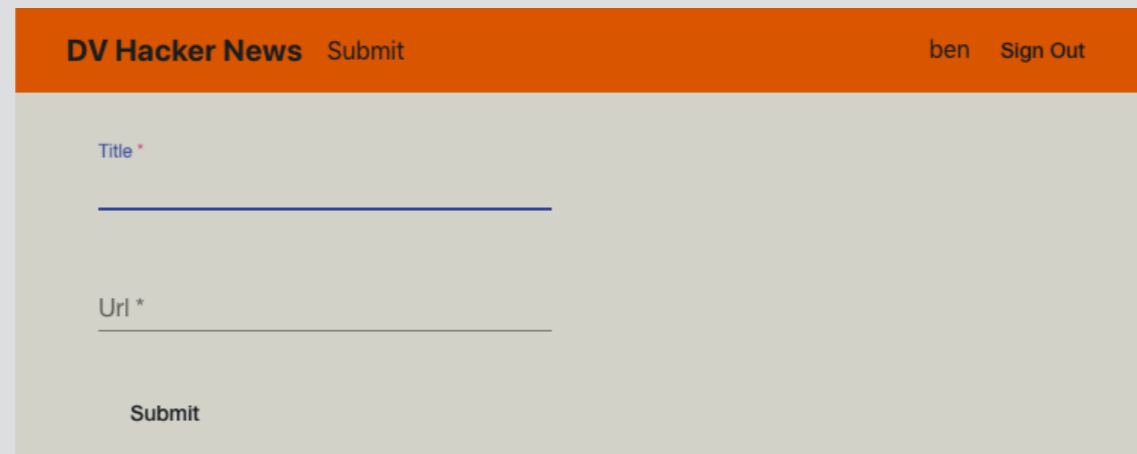
transaction

id used

set param

id used

submit post action



```
<dv.action name="submit-post">  
<hackernews.navbar />  
<div class="main"> <dv.tx>  
  <dv.gen-id />  
  <property.create-object  
    id=dv.gen-id.id  
    initialValue={ author: hackernews.navbar.user.username }  
    showExclude=["author"]  
    buttonLabel="submit"  
    newObjectSavedText="Post"  
  <scoringposts.create-score  
    targetId=dv.gen-id.id  
    value=0  
    hidden=true />  
  <authentication.authenticate id=hackernews.navbar.user hidden=true />  
  <dv.link href="/item" params={ id: dv.gen-id.id } />  
</dv.tx> </div> </dv.action>
```

generate id

transaction

id used

set param

id used

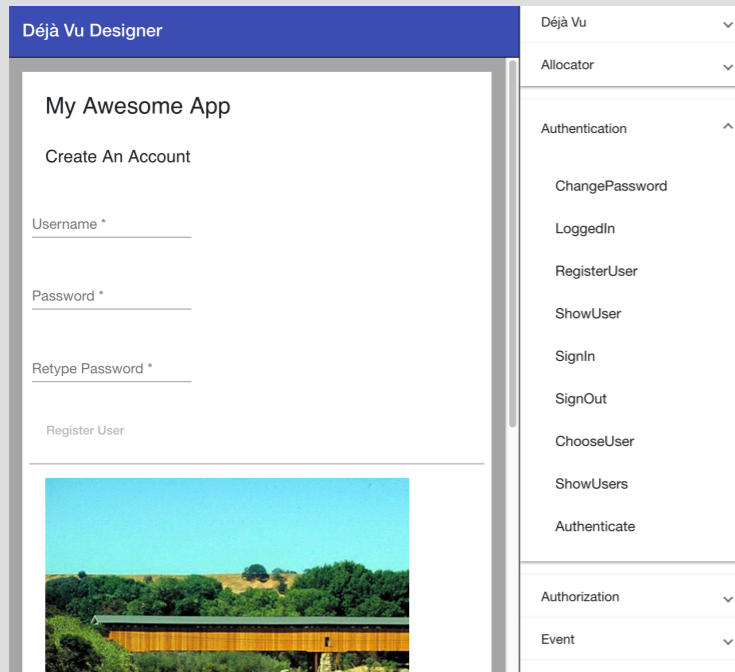
redirect

rebuilding class projects

app	Authentication	Authorization	Passkey	Transfer	Rating	Follow	Geolocation	Comment	Event	Label	Group	Task	Scoring	Property	Chat
Accord	1	1	0	0	1	0	0	0	0	0	1	0	1	2	0
ChoreStar	2	2	0	1	1	0	0	0	0	0	0	1	0	0	0
MapMIT	1	1	0	0	0	0	2	0	1	0	1	0	0	1	0
Rendezvous	1	1	0	0	0	0	2	1	1	1	1	0	0	1	0
SweetSpots	1	1	0	0	1	1	1	1	0	1	0	0	0	1	0
Potluck	1	1	0	1	0	0	0	0	1	0	1	0	0	1	0
GroceryShip	1	1	0	2	1	0	0	0	1	0	0	1	0	2	0
Lingua	1	1	0	0	0	0	0	1	0	1	0	0	1	1	1
LiveScorecard	1	1	2	0	0	0	0	0	1	1	2	1	1	1	0

cliché

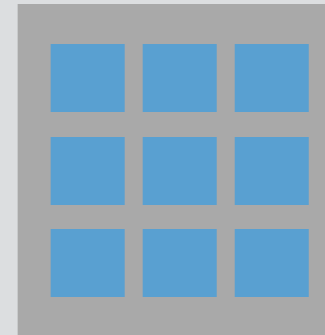
other aspects of deja vu



WYSIWYG designer
(Barry McNamara)



security: stop
request forgeries



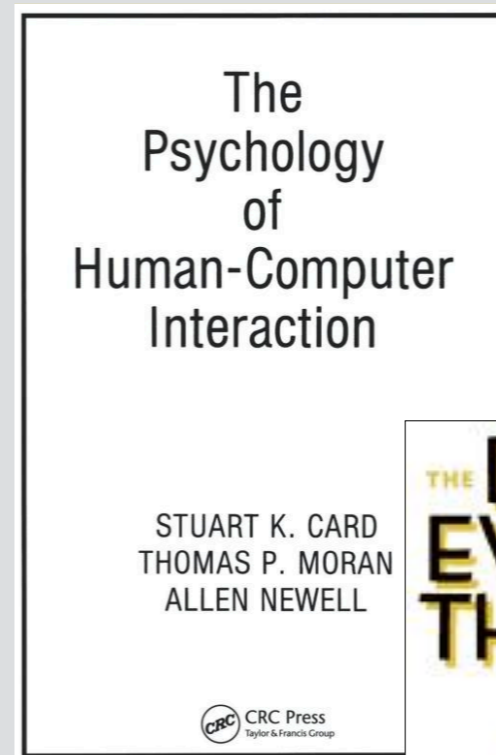
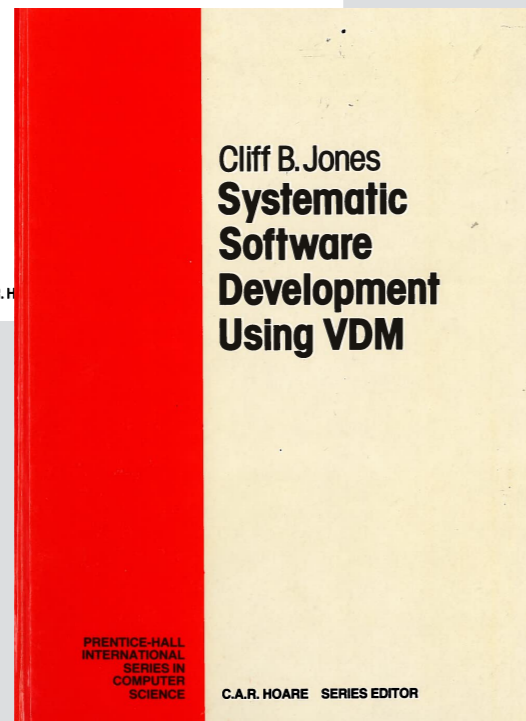
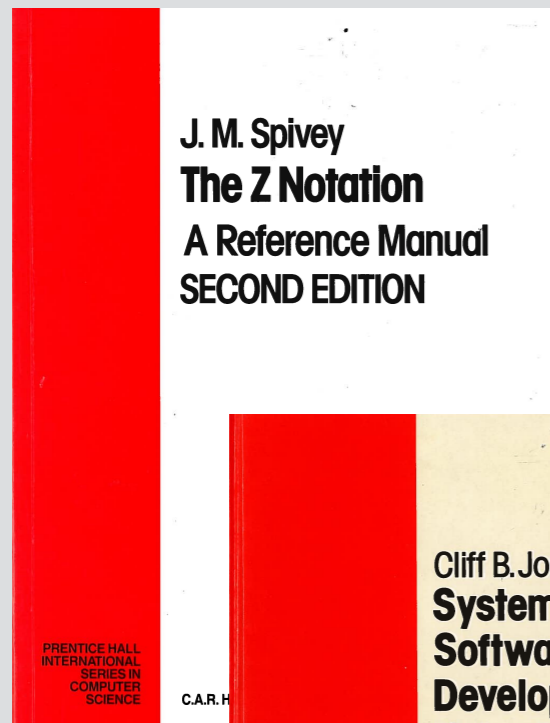
cliche library
for social apps
(Maryam Archie)



make it easier
to author cliches
(Czarina Lao)

closing
thoughts

bringing two fields together



user-centered design: conceptual model should be designed
formal methods: software defined by its behavior
both originating around 1974

<https://tinyurl.com/dbctouch>

to keep in touch and be notified about publication of book

<https://tinyurl.com/postcard-get>

to sign up for monthly resilience postcards

studio 2

facebook



Civility = optimism.

When our country and the world seem to be unraveling, it's tempting to dream of radical responses. If civic norms are undermined daily by the most powerful members of our society, why should we, the little people, not cast them off too? How will we ever save our society, our country, our planet if we continue to be fettered by conventional niceties?

Subscribe to Resilience Postcards at:
<https://www.portraitsofresilience.com/subscribe/...> See More



👍 and 14 others 1 Comment 1 Share

👍 Like 💬 Comment ➦ Share

👤 Thank you for writing this Daniel, so true!!!

Like · Reply · 18w



Write a comment...



construct concept models in this order
post, friend, comment, upvote, tag

specify application binding

for each concept, give
purpose: informally stated
structure: text or diagram
behavior: actions specified formally
tactic: informal scenario

hints: make each concept
minimal: only essential functionality
free-standing: makes sense alone
orthogonal: avoid overlap

what issues came up?

reminder: a reservation concept

name

reservation

purpose

make access to shared resource reliable

structure

slots: Owner -> Slot
holds: User -> Slot

behavior

create (o: Owner, s: Slot)
no slots.s => slots += o -> s
reserve (u: User, o: Owner, s: Slot)
no holds.s **and** o -> s **in** slots => holds += u -> s
cancel (u: User, s: Slot)
u -> s **in** holds => holds -= u -> s
use (u: User, o: Owner, s: Slot)
u -> s **in** holds **and** o -> s **in** slots =>

tactic

if create(o,s); reserve(u,o,s); ... **no** cancel(u,s) ... **then can** use(u,o,s)

reminder: alloy expressions in one slide

u: User
s: Slot
holds: User -> Slot
Taken: set Slot

a relation is a table of rows

holds = {(u1,s1), (u1,s2)}
holds' = {(u1,s1), (u1,s2), (u2,s3)}

a set is a relation with one column

Slot = {(s1), (s2), (s3), (s4)}
Taken = {(s1), (s2), (s3)}

a scalar is a set with one row

u = {(u2)}
s = {(s3)}

set operators

+ union, - difference, & intersection, in subset

Slot - Taken = {(s4)}
holds' - holds = {(u2,s3)}

relation operators

-> product
.join

product examples

u -> s = {(u2,s3)}
u -> Taken = {(u2,s1), (u2,s2), (u2,s3)}

join examples

u.holds' = {(s3)}
holds'.s = {(u2)}
holds.Slot = {(u1)}

formula examples

holds' = holds + u -> s
(also written holds += u -> s
User.holds = Taken
holds in User -> Slot

$a \rightarrow b = \{ (a_0, \dots, a_n, b_0, \dots, b_m) \mid (a_0, \dots, a_n) \in a \wedge (b_0, \dots, b_m) \in b \}$

$a.b = \{ (a_0, \dots, a_{n-1}, b_1, \dots, b_m) \mid (a_0, \dots, a_n) \in a \wedge (a_n, b_1, \dots, b_m) \in b \}$