



Concepts: A New Modularity for Software

Daniel Jackson · MIT CSAIL · SPLASH 2018

with

Santiago Perez De Rosso

& Maryam Archie, Czarina Lao, Barry McNamara

funded by

MIT-SUTD International Design Center

ten years ago...

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

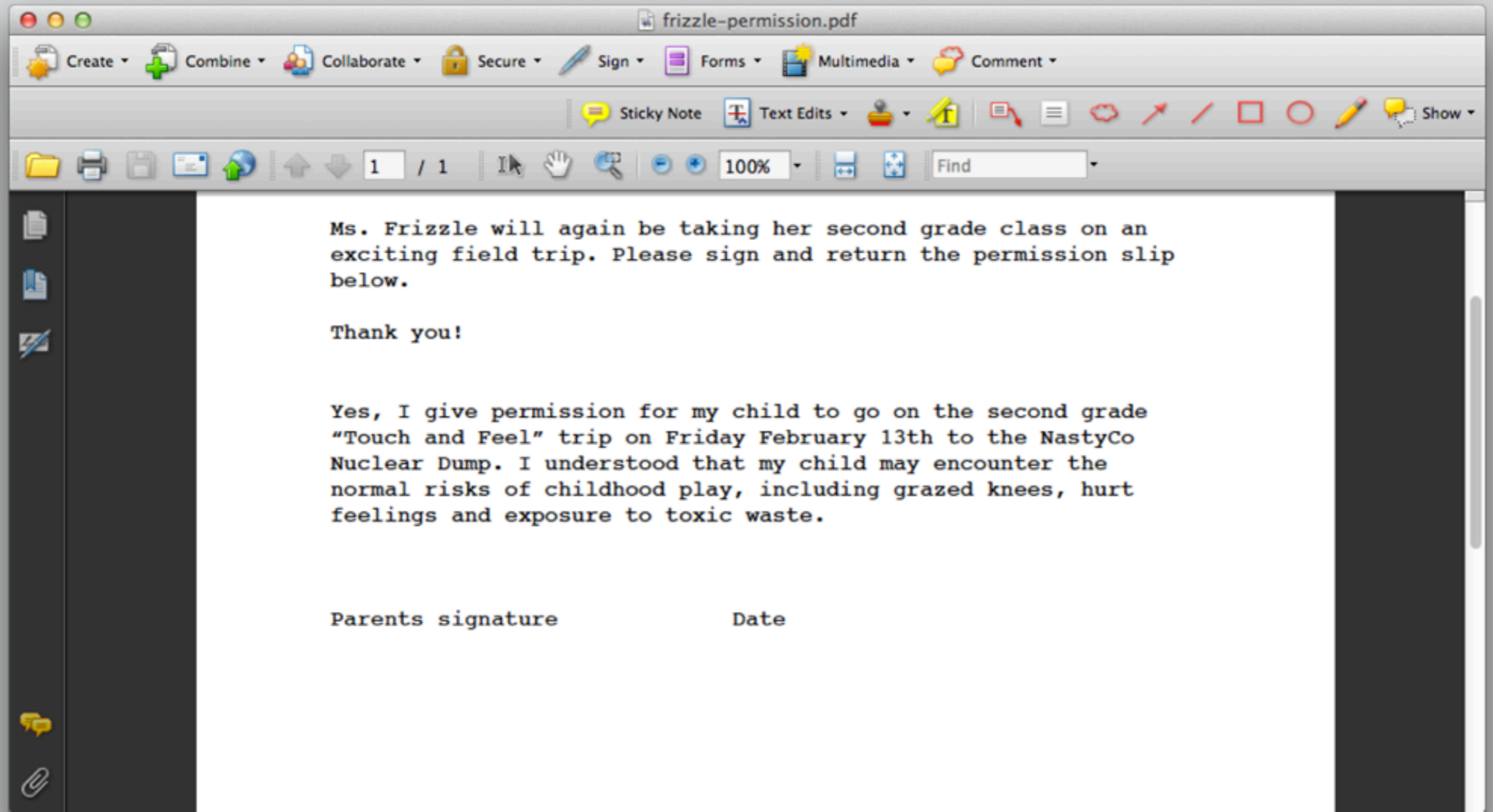
February 11, 2013

Parents signature

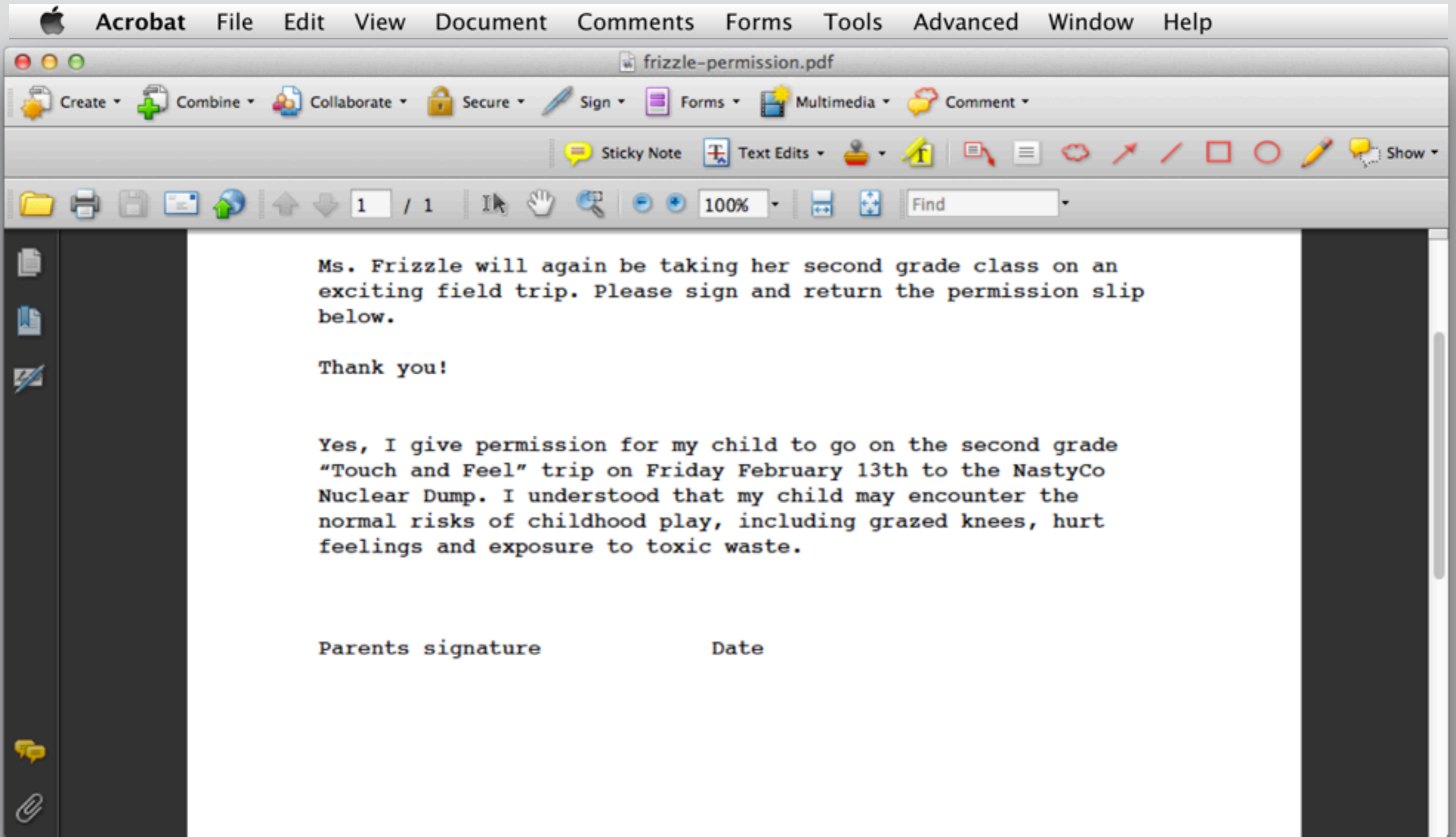
Date

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" with the following text:

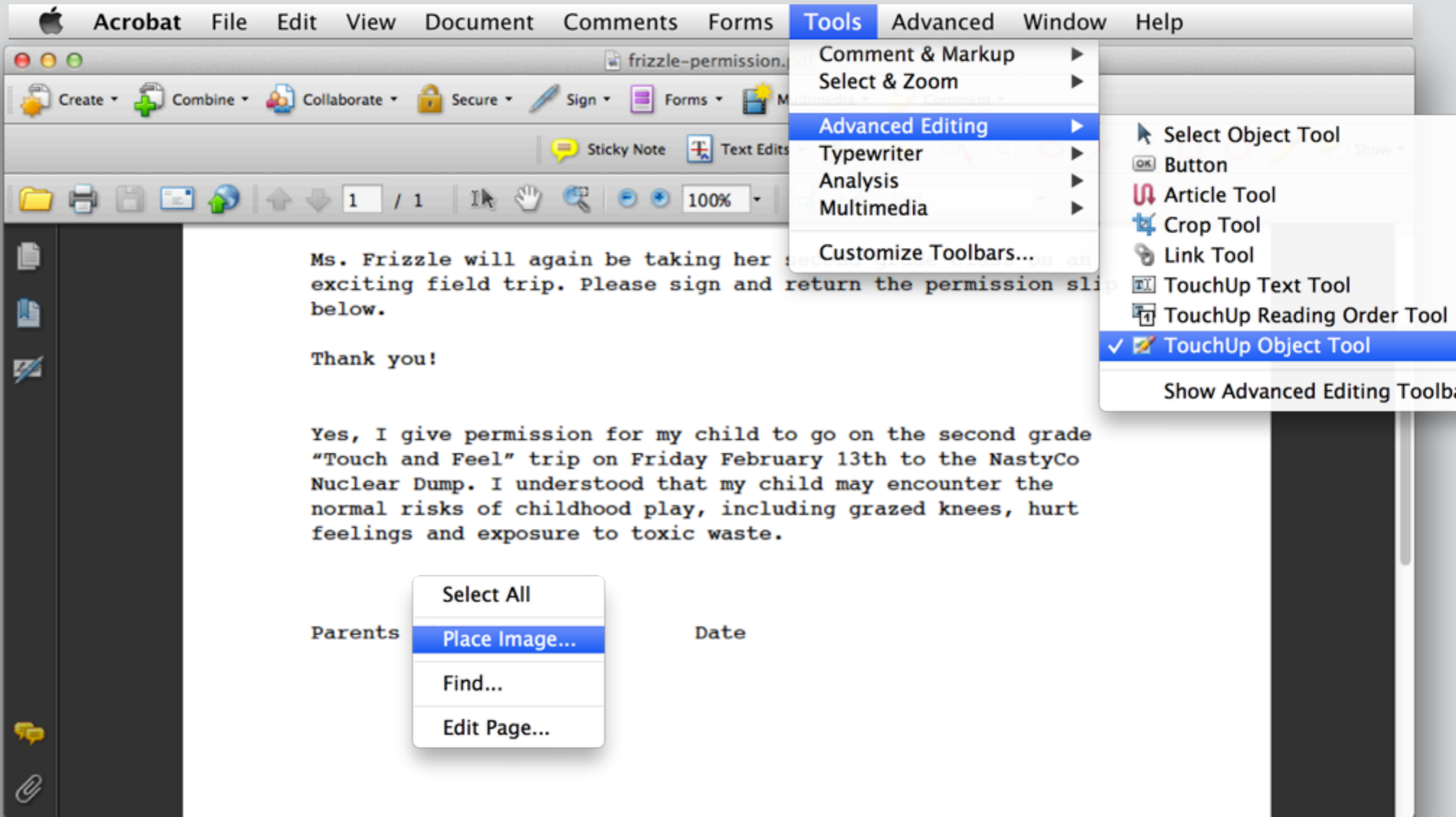
Ms. Frizzle will again be taking her second grade class on an exciting field trip. Please sign and return the permission slip below.

Thank you!

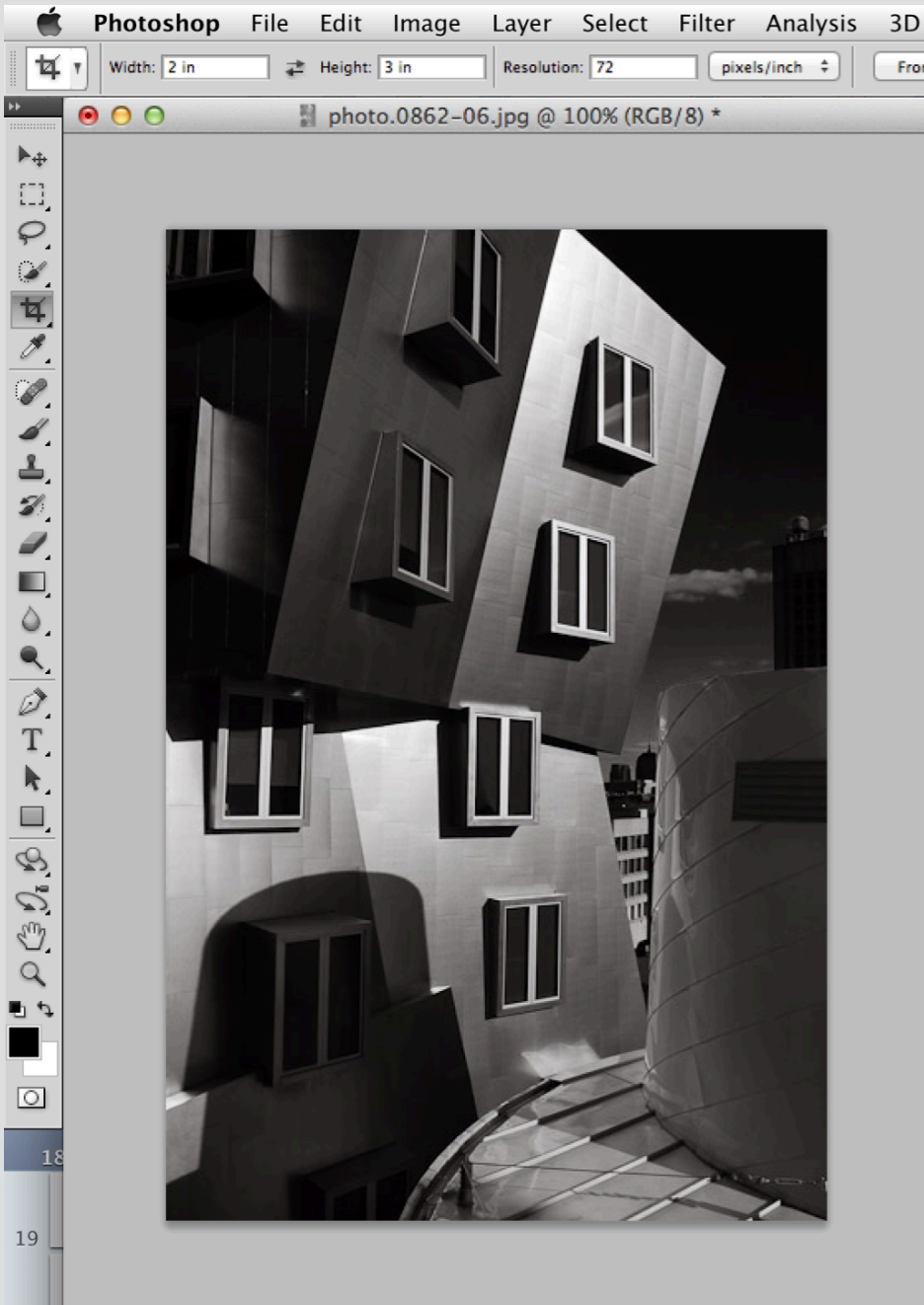
An inset window titled "acrobat-sig-paste.txt" is open, showing a list of instructions:

- 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

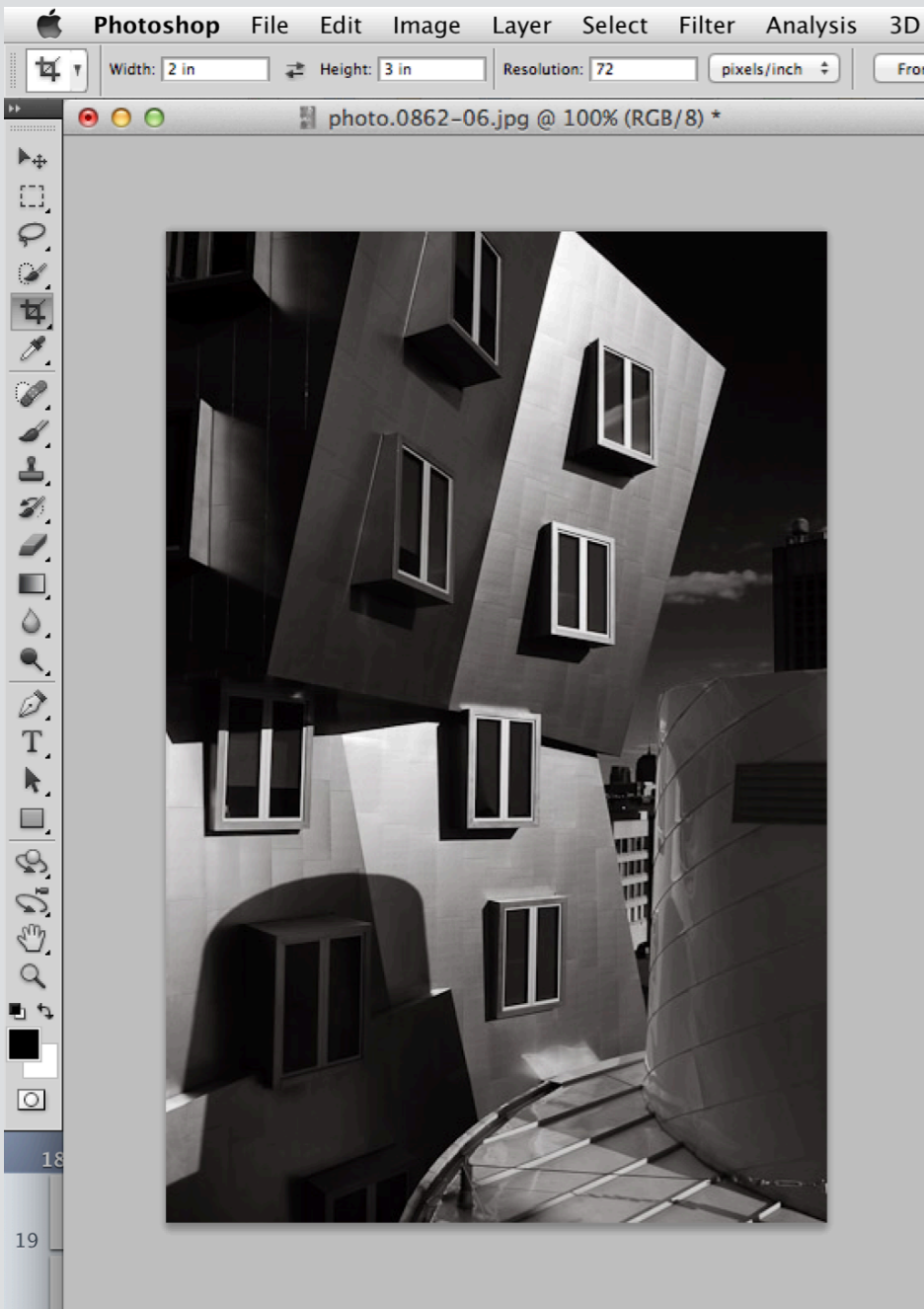
acrobat to the rescue?



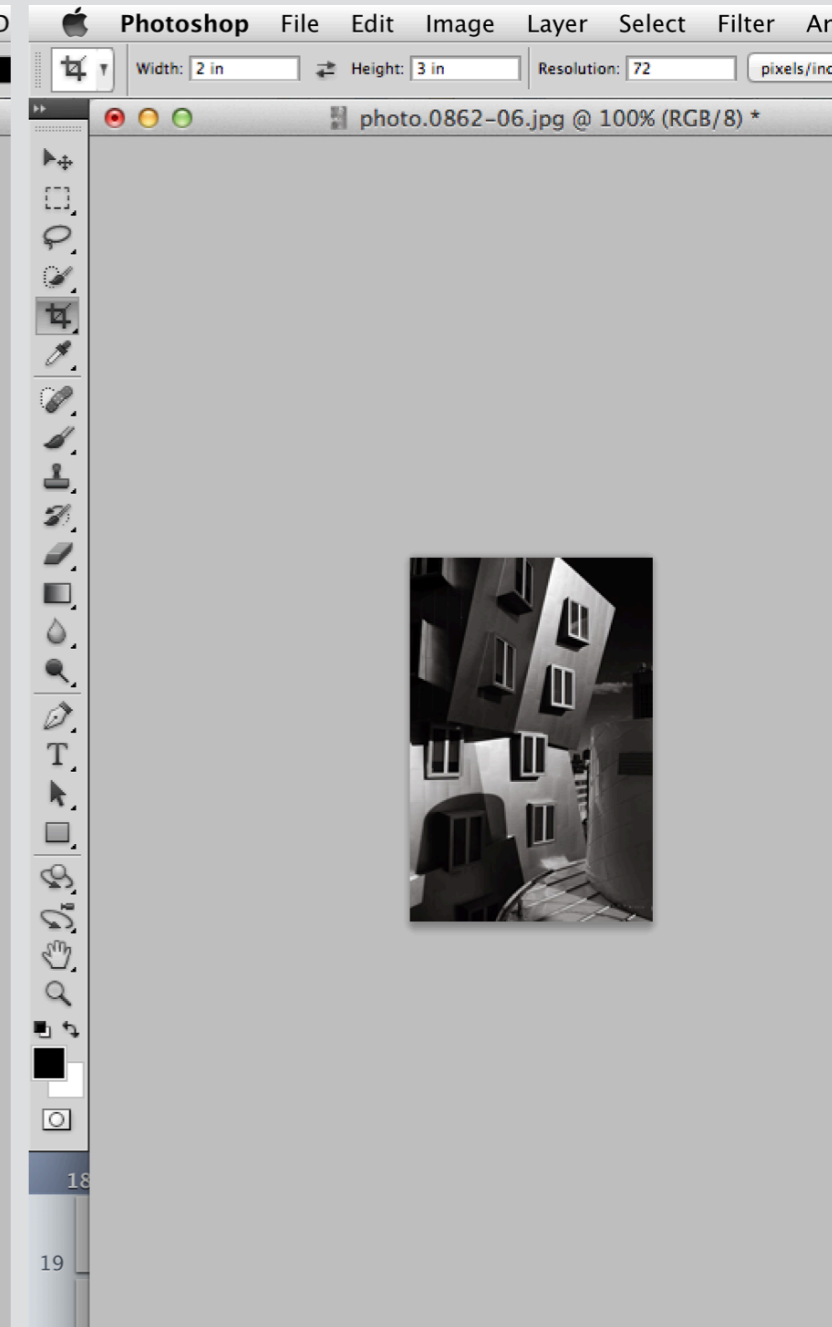
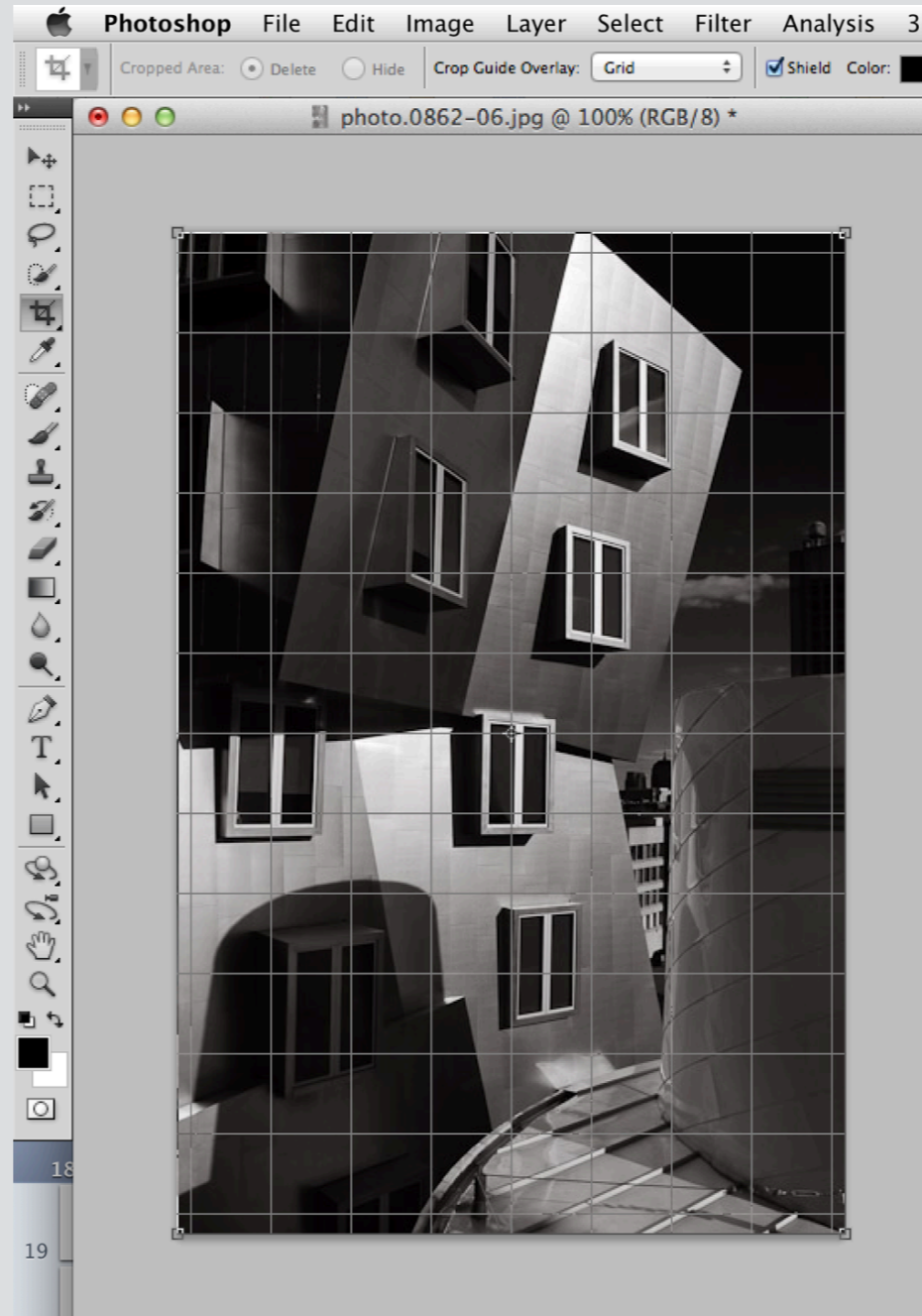
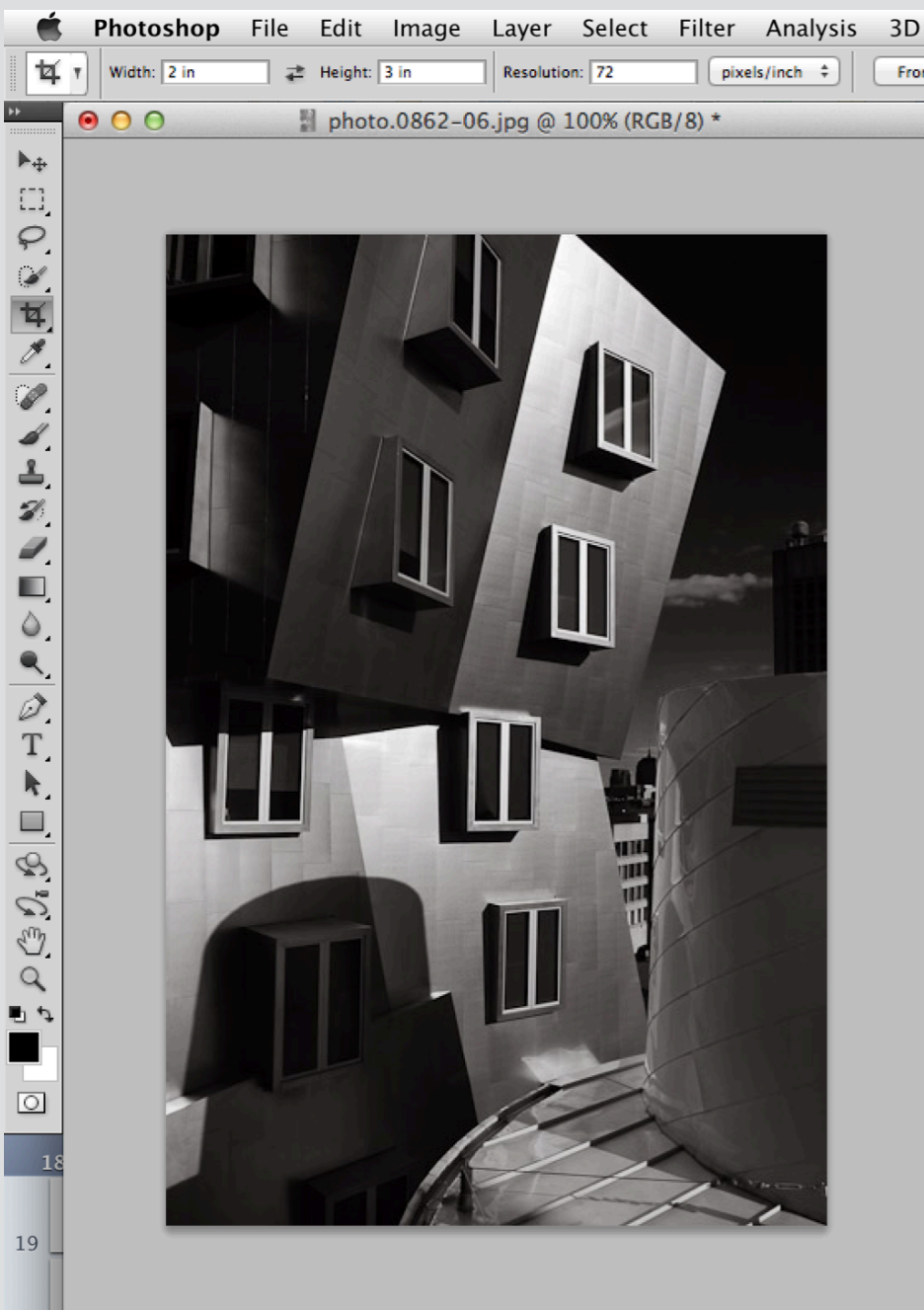
adobe photoshop: cropping surprises



adobe photoshop: cropping surprises

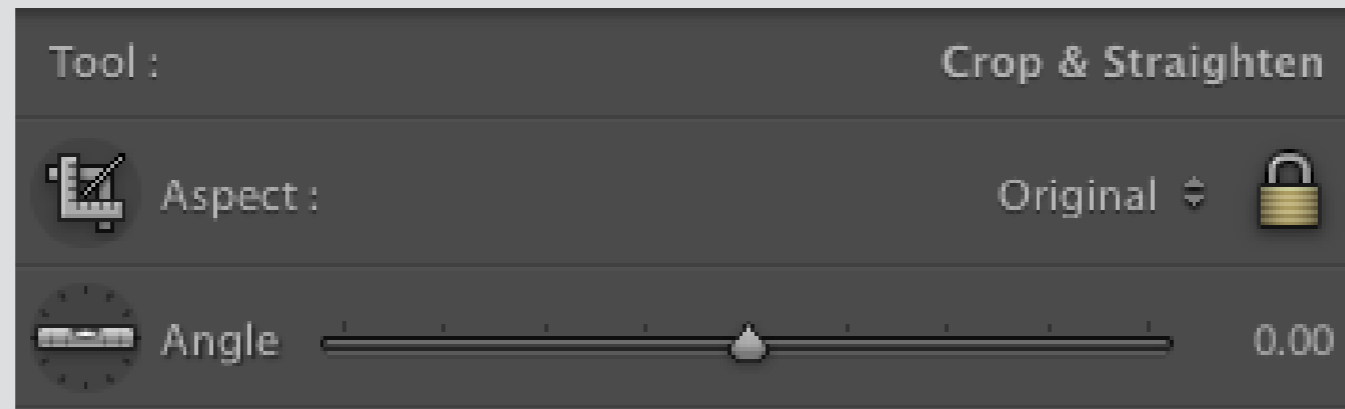


adobe photoshop: cropping surprises



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

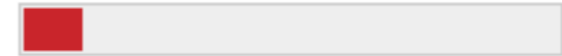


2,335 Positive Opinions out of 2,632



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

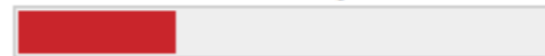


237 Negative Opinions out of 444



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

30% Hate Photoshop



Tweet 105

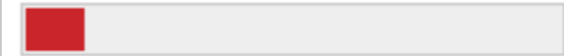


12,978 Negative Opinions out of 43,283



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



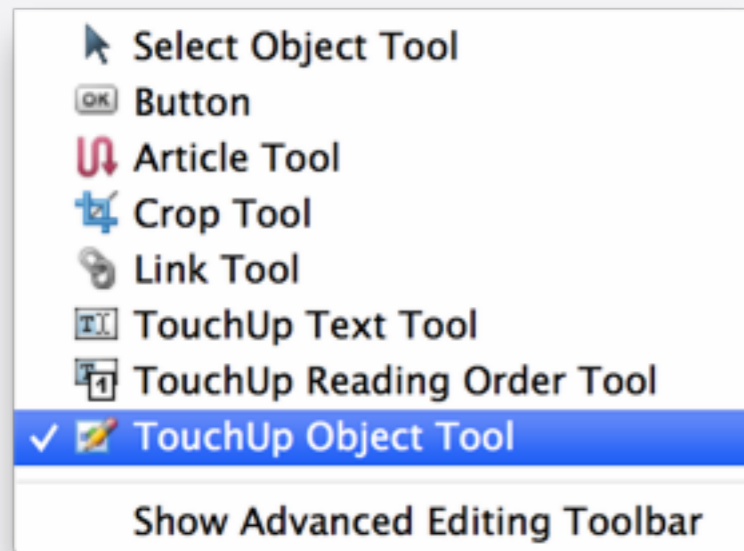
297 Negative Opinions out of 2,632

what's the essence of the problem?

not the user interface!
polished and organized

what's the essence of the problem?

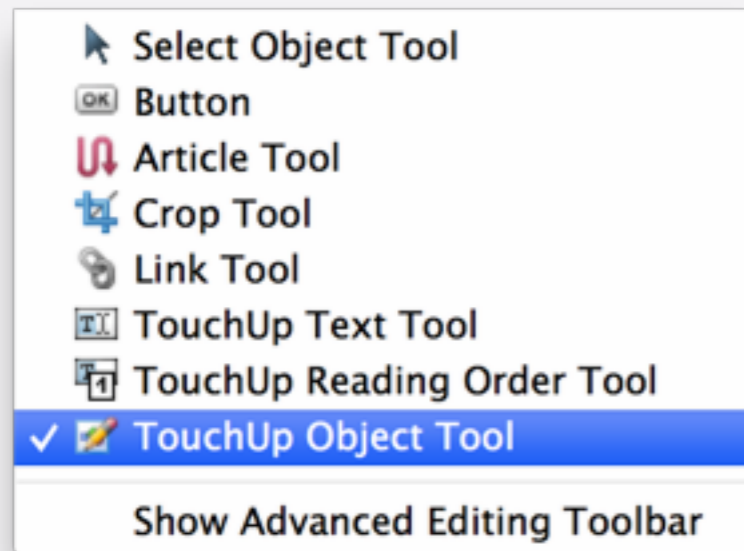
not the user interface!
polished and organized



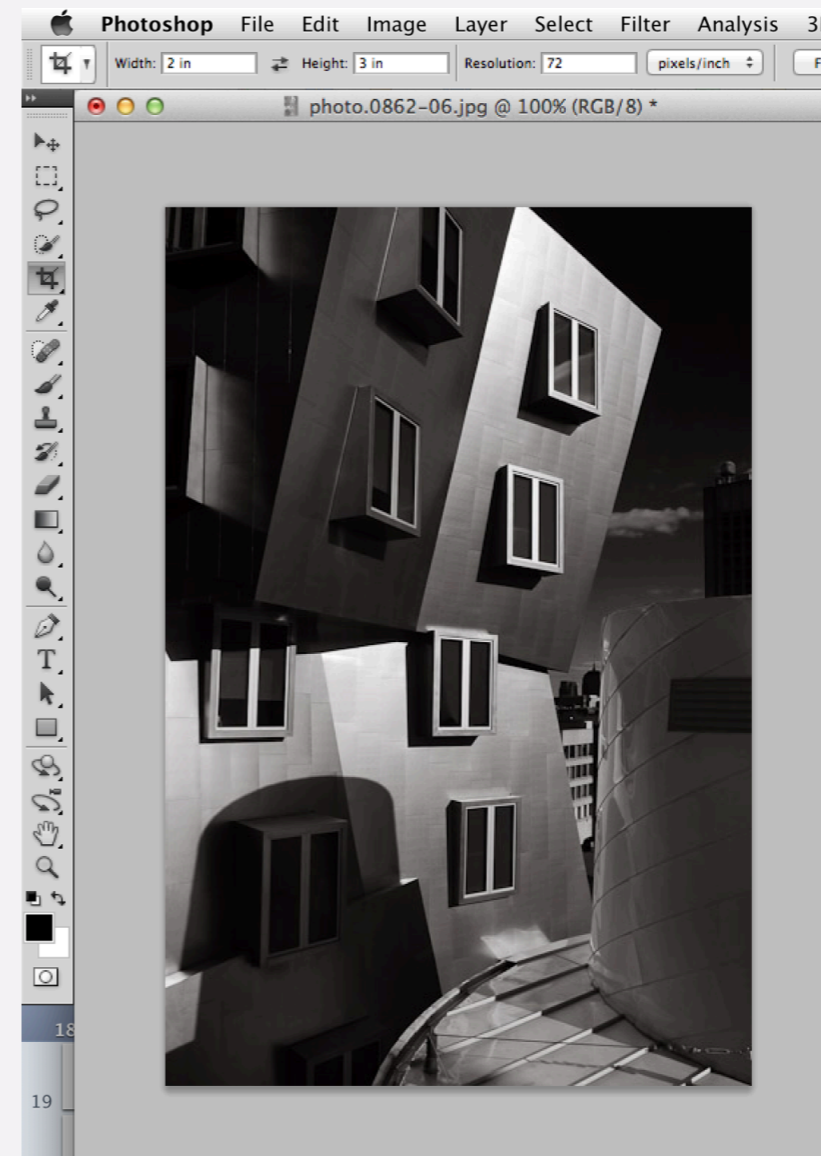
task-oriented design
no unifying concepts

what's the essence of the problem?

not the user interface!
polished and organized



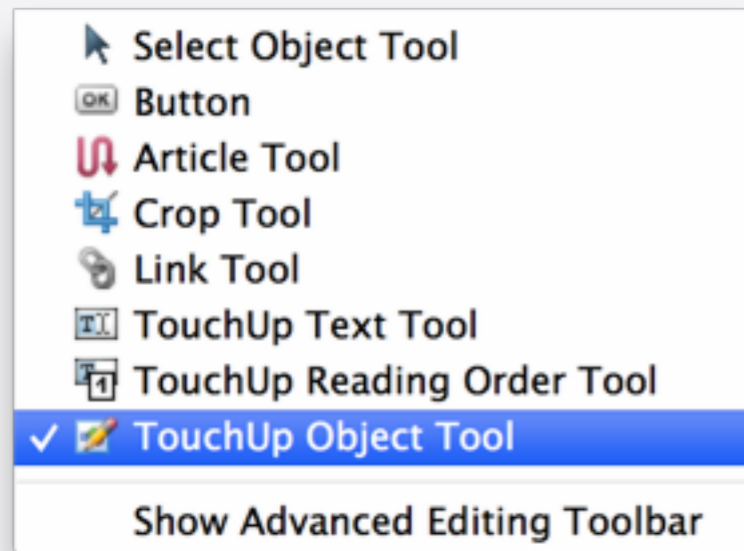
task-oriented design
no unifying concepts



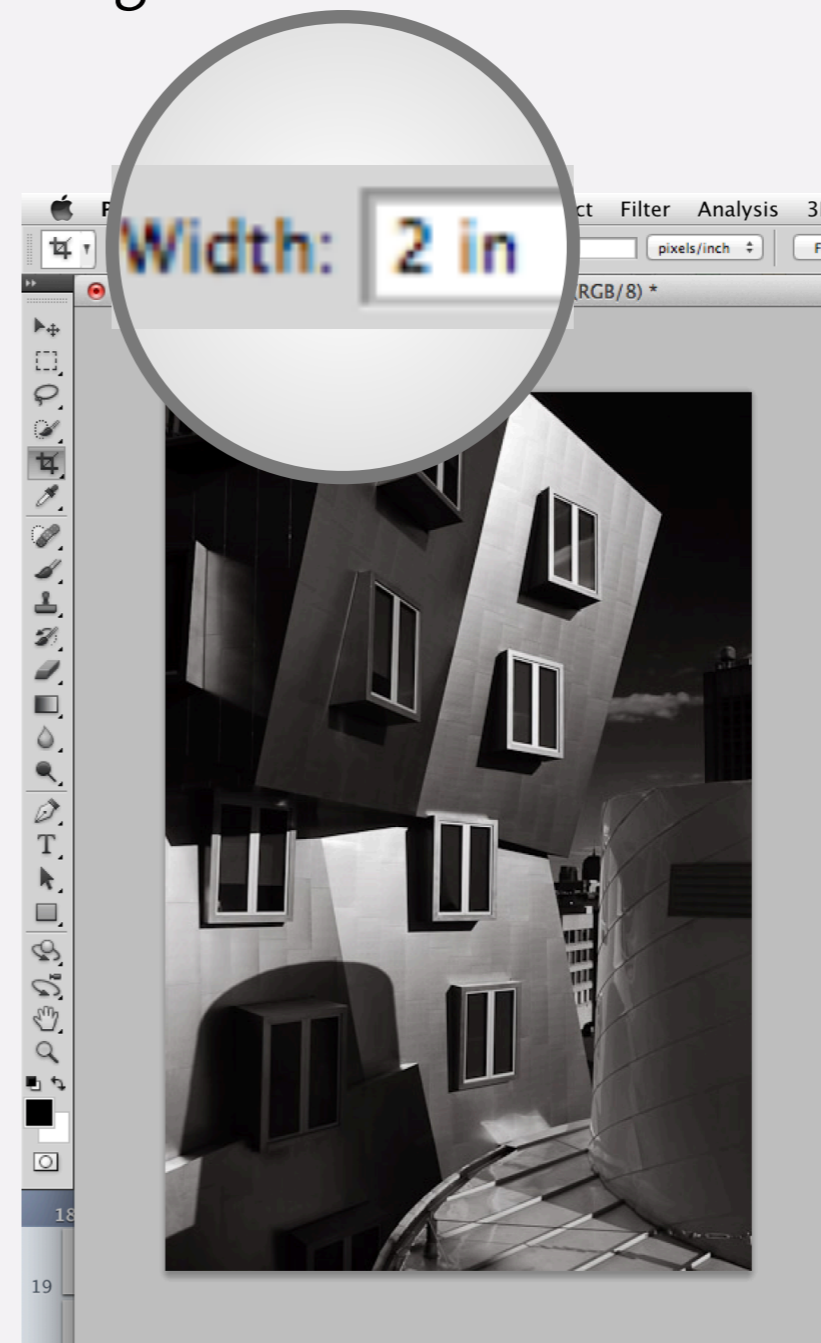
complex concepts
cropping vs. resizing

what's the essence of the problem?

not the user interface!
polished and organized



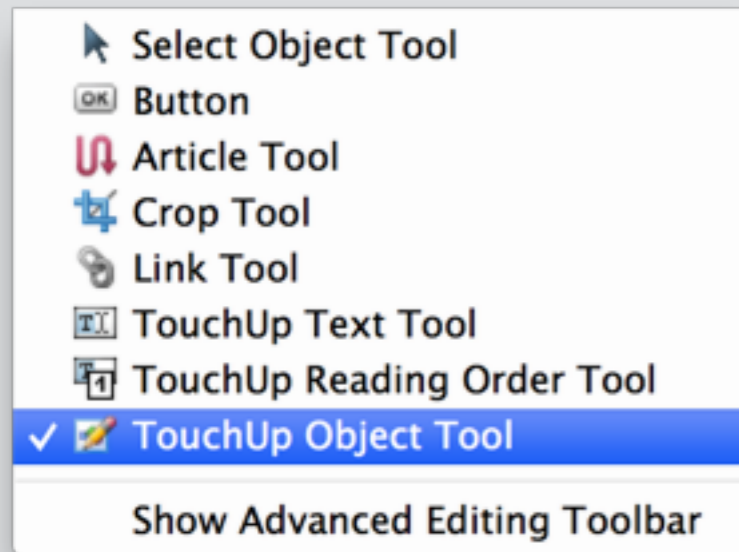
task-oriented design
no unifying concepts



complex concepts
cropping vs. resizing

adobe fixes acrobat

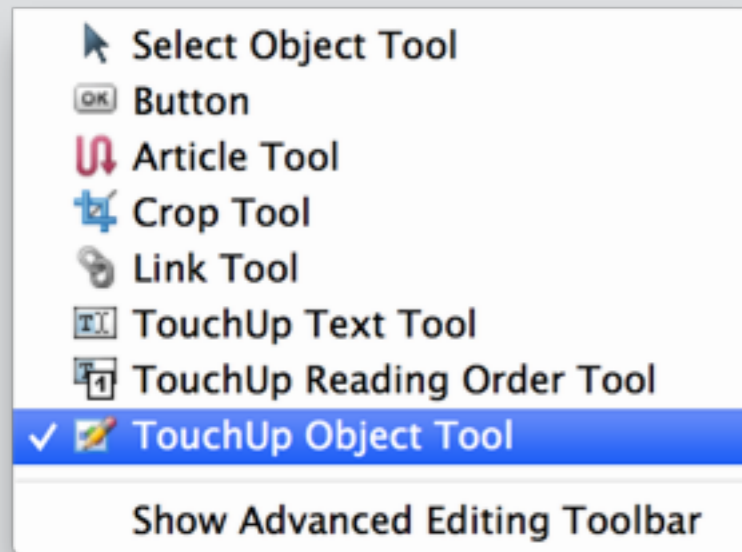
Version 9 (2008)



task-oriented design
no unifying concepts

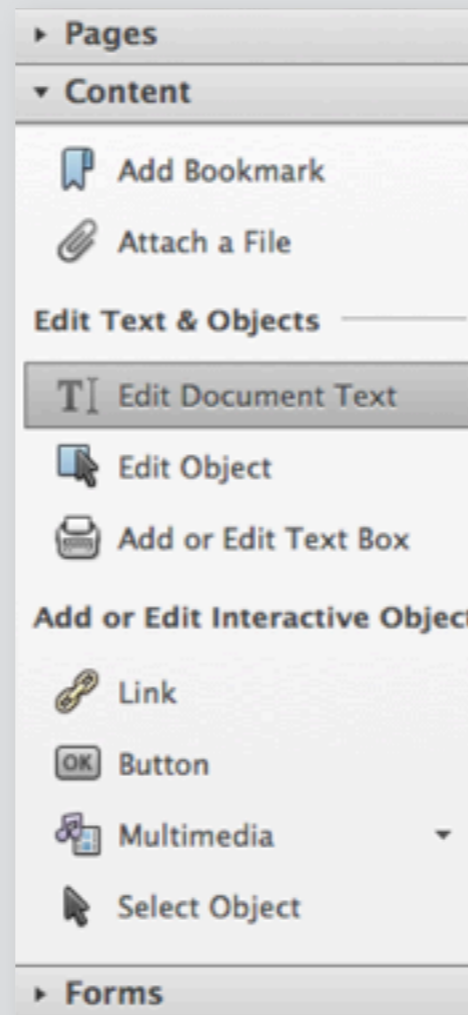
adobe fixes acrobat

Version 9 (2008)



task-oriented design
no unifying concepts

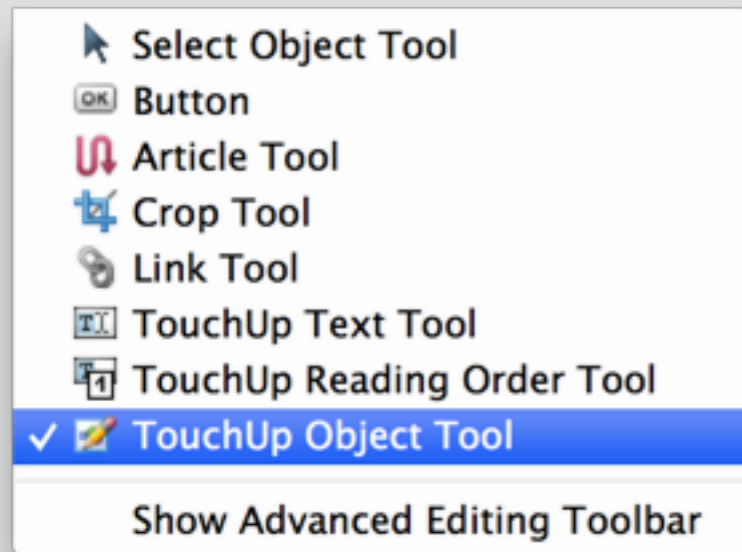
Version 10 (2010)



improved interface
but still no concepts

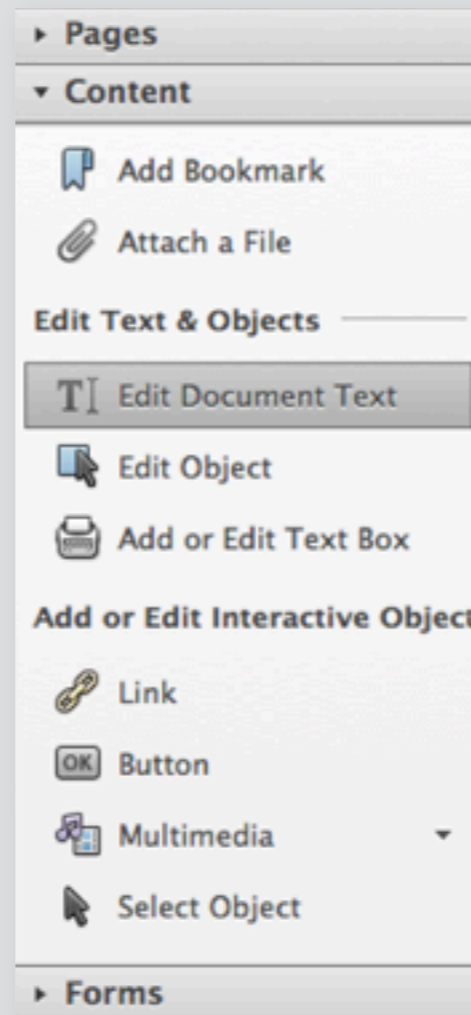
adobe fixes acrobat

Version 9 (2008)



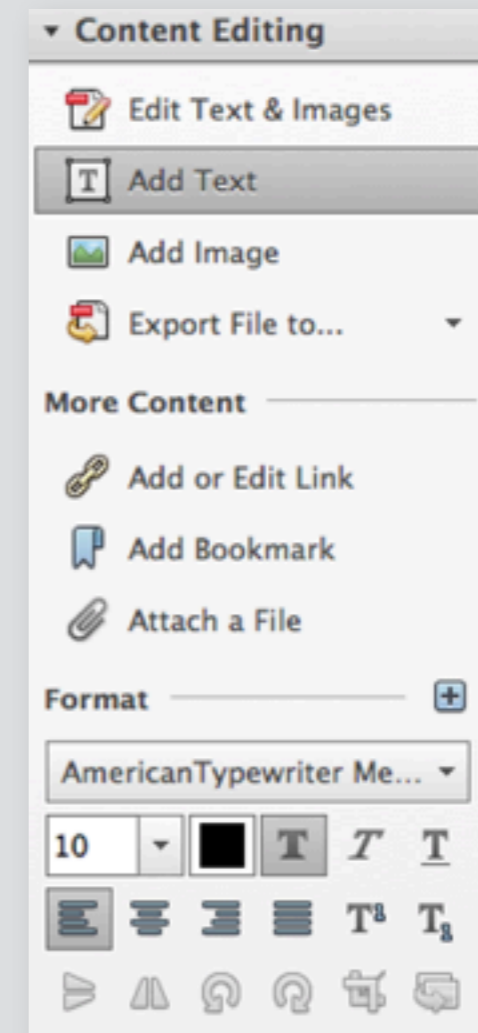
task-oriented design
no unifying concepts

Version 10 (2010)



improved interface
but still no concepts

Version 11 (2012)

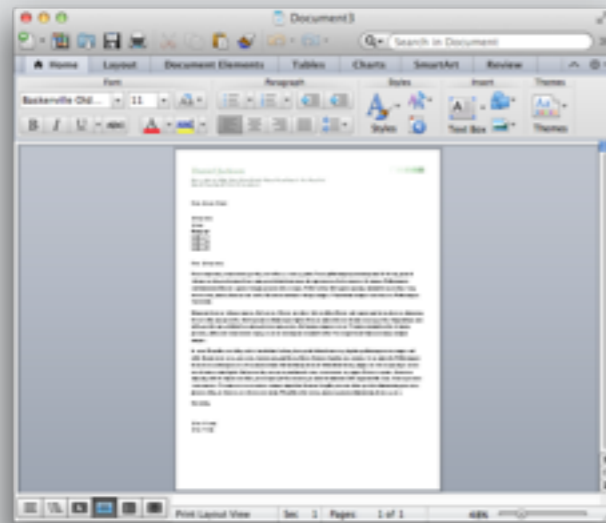


unifying concepts
text/image object

thinking
about
concepts

apps characterized by their concepts

Microsoft Word



Twitter



Photoshop



apps characterized by their concepts

Microsoft Word



Paragraph
Format
Style

Twitter

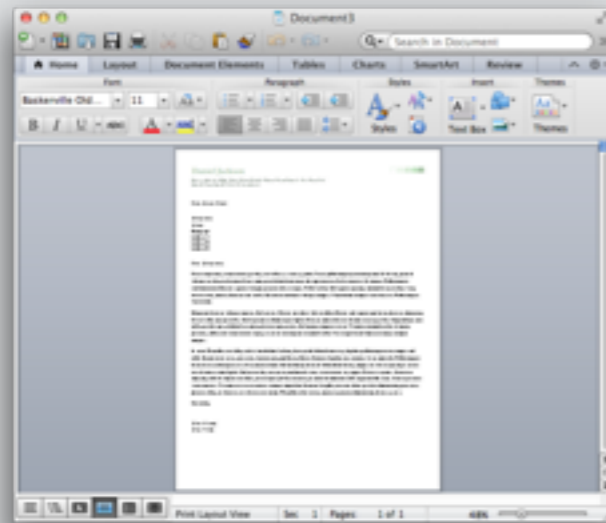


Photoshop



apps characterized by their concepts

Microsoft Word



Paragraph
Format
Style

Twitter



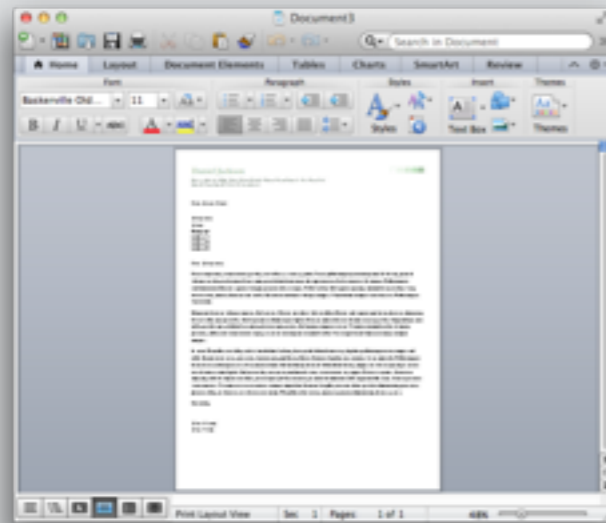
Tweet
Hashtag
Following

Photoshop



apps characterized by their concepts

Microsoft Word



Paragraph
Format
Style

Twitter



Tweet
Hashtag
Following

Photoshop



PixelMap
Layer/Mask
Adjustment

app classes characterized by concepts too

app classes characterized by concepts too

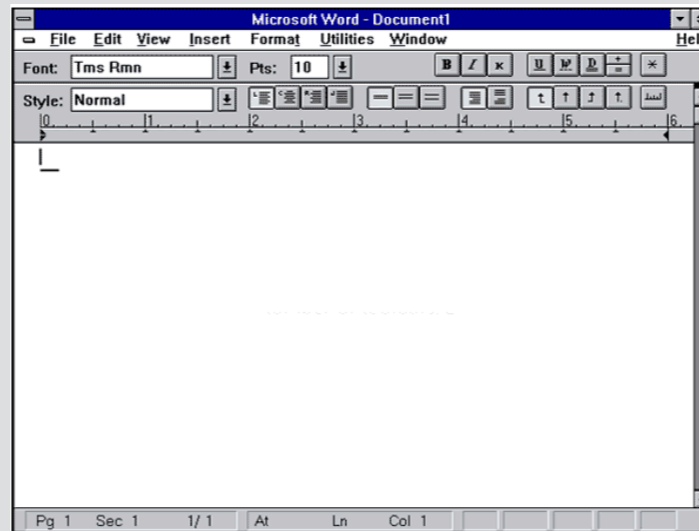
```
44 -- approaches, not satisfying: surely something essential-
45
46 Take a step back: concepts-
47 -- example: Microsoft Word: font and Co-
48 -- where concepts come from-
49 -- implications of concepts-
50 -- how concepts distinguish apps (Word from LaTeX) and app classes (Word from [old])-
51 -- 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 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 directive-
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



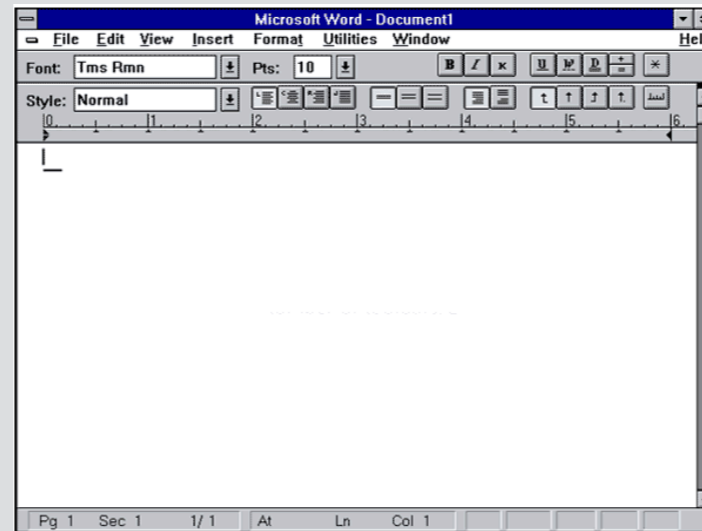
word processor

app classes characterized by concepts too

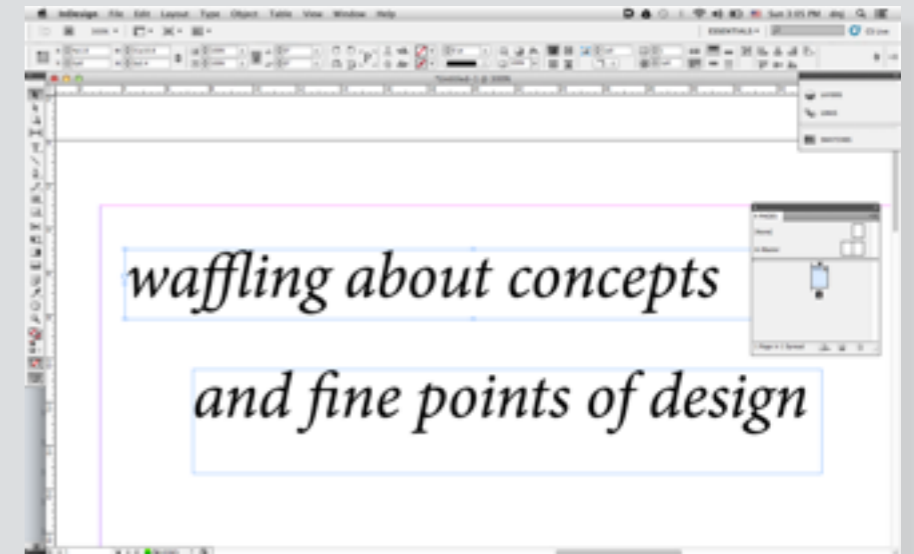


```
44 -- approaches, not satisfying: surely something essential-
45
46 Take a step back: concepts-
47 -- example: Microsoft Word: font and Co-
48 -- where concepts come from-
49 -- implications of concepts-
50 -- how concepts distinguish apps (Word from LaTeX) and app classes (Word from [old])
51 -- 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



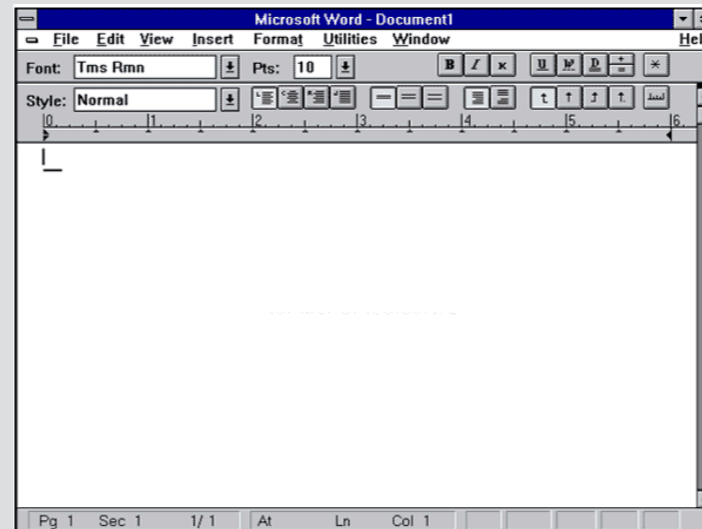
desktop publishing app

app classes characterized by concepts too

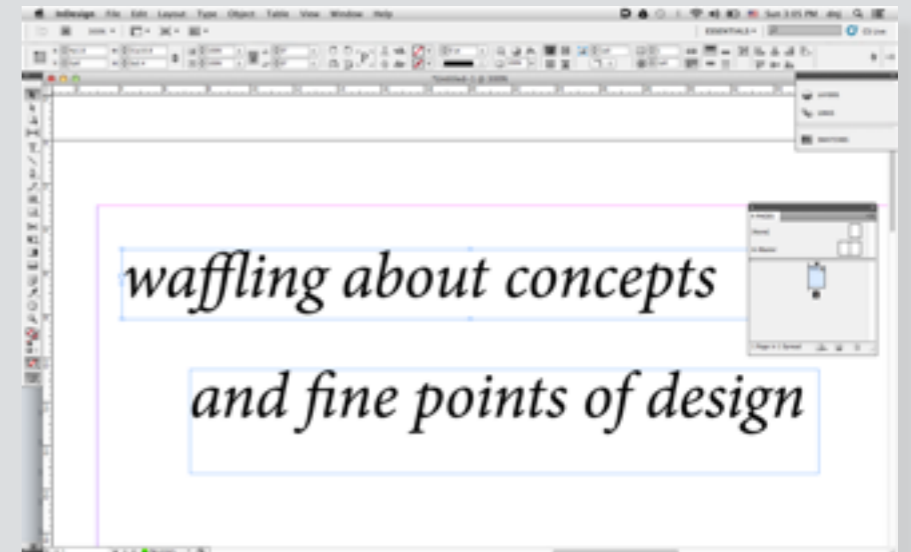


44 -- approaches, not satisfying: surely something essential-
45
46 Take a step back: concepts-
47 -- example: Microsoft Word: font and Co-
48 -- where concepts come from-
49 -- implications of concepts-
50 -- how concepts distinguish apps (Word from LaTeX) and app classes (Word from [code])
51 -- what's 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 scifits) -
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 -- 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

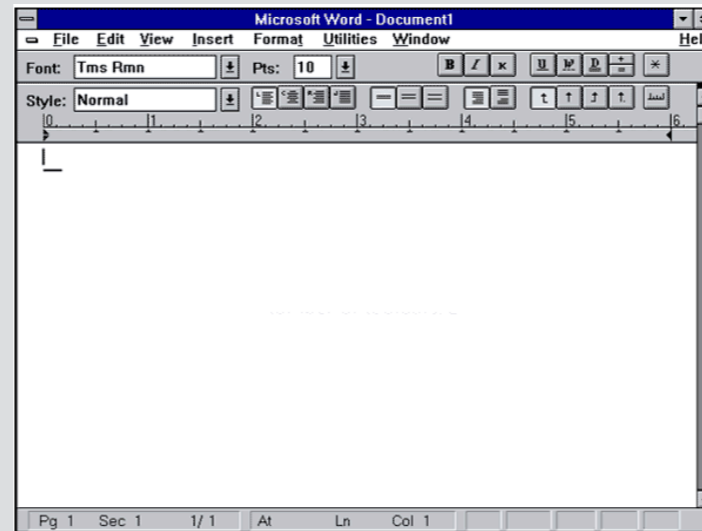


desktop publishing app

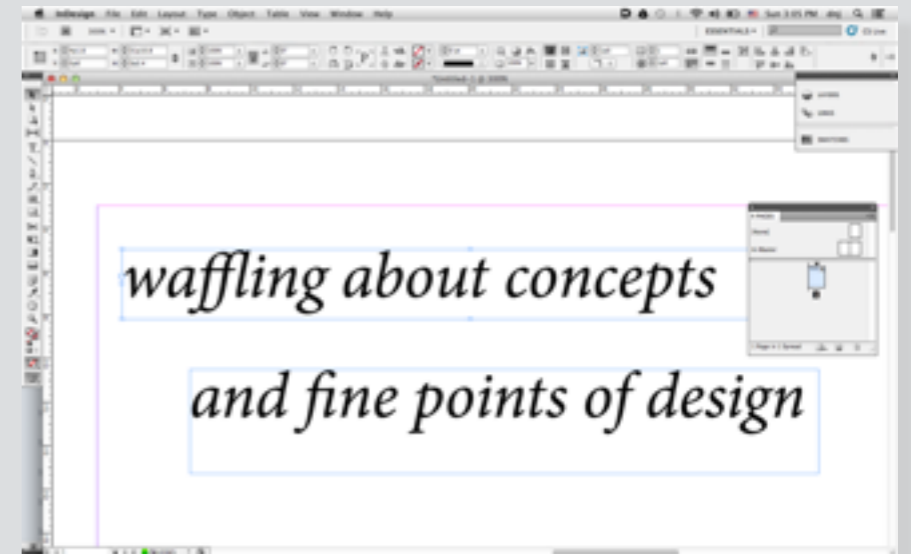
app classes characterized by concepts too



text editor
line
buffer



word processor
paragraph
format
style

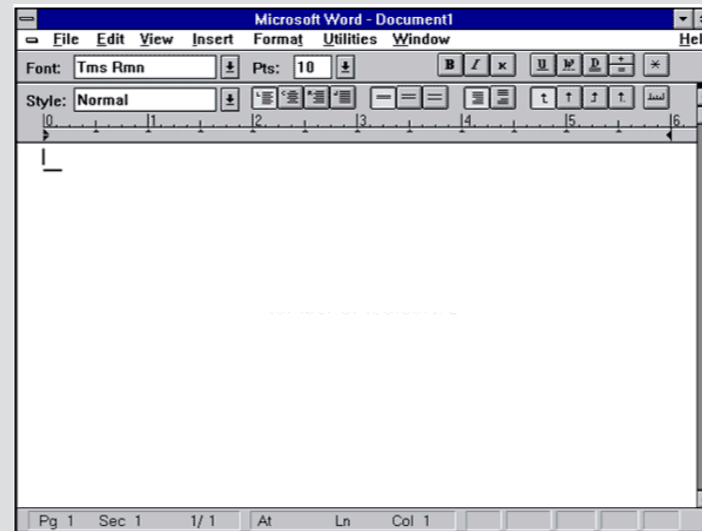


desktop publishing app

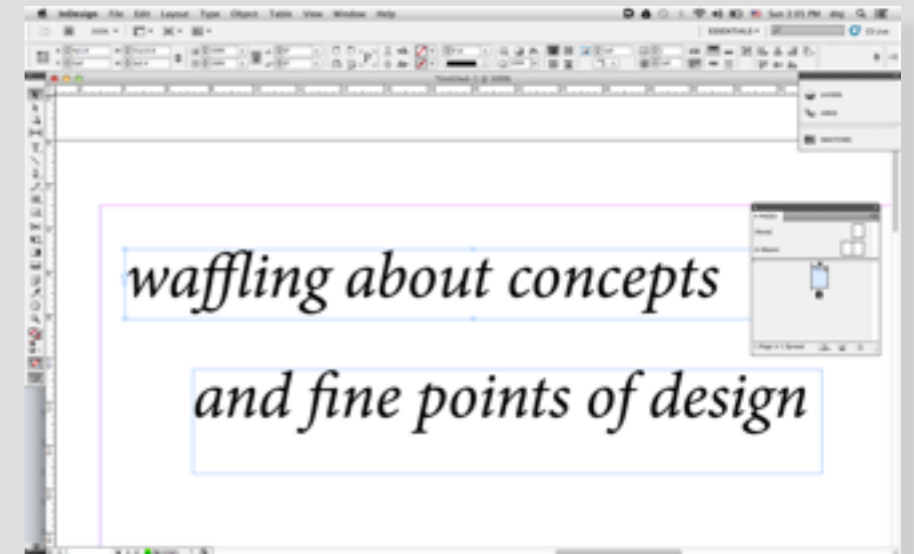
app classes characterized by concepts too



text editor
line
buffer



word processor
paragraph
format
style

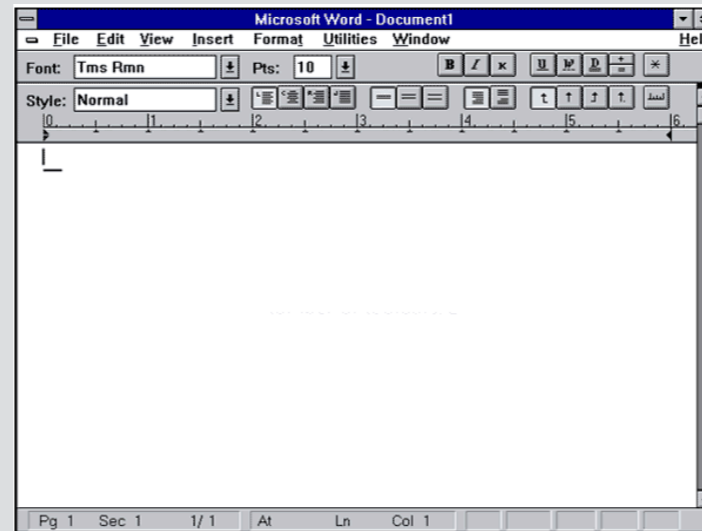


desktop publishing app
stylesheet
text flow
page template

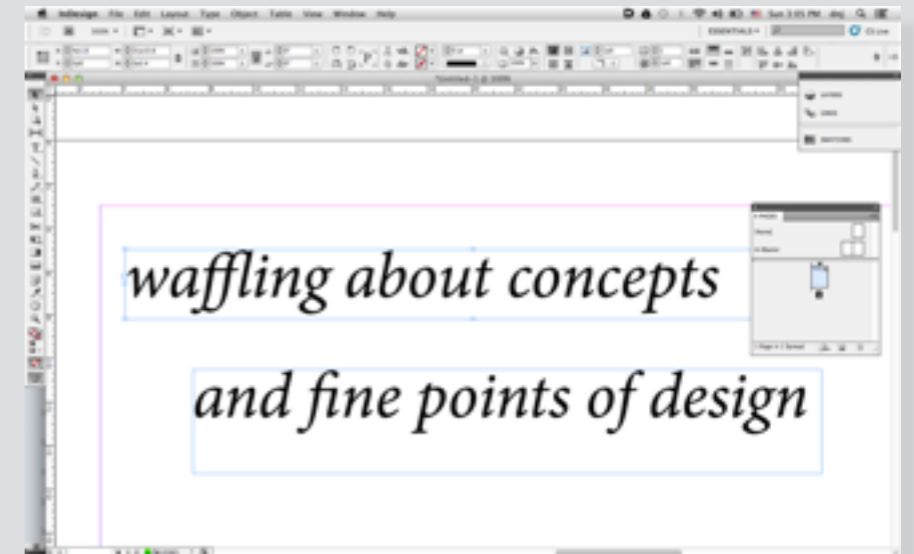
app classes characterized by concepts too



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

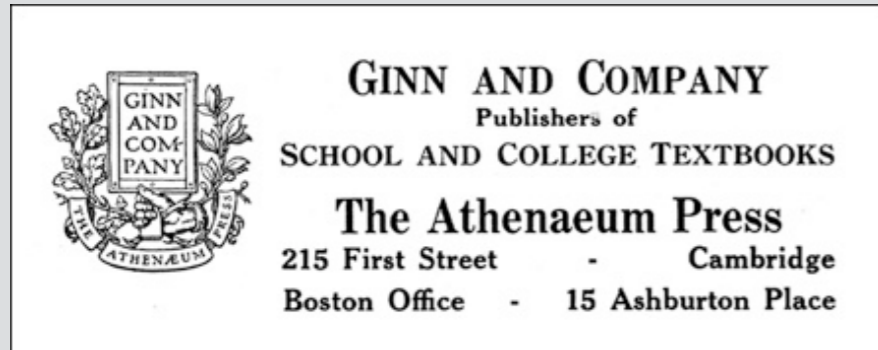
Conceptual model (computer science)

From Wikipedia, the free encyclopedia

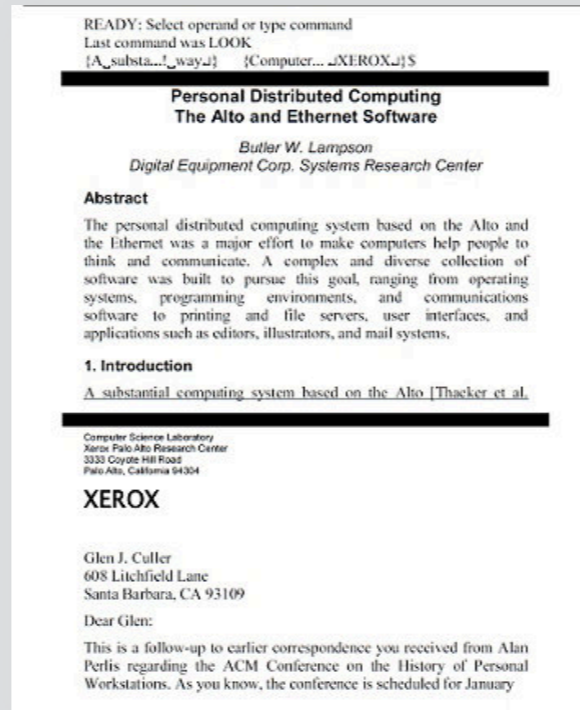
A [mental model](#) captures ideas in a [problem domain](#), while a **conceptual model** represents 'concepts' (entities) and relationships between them.

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

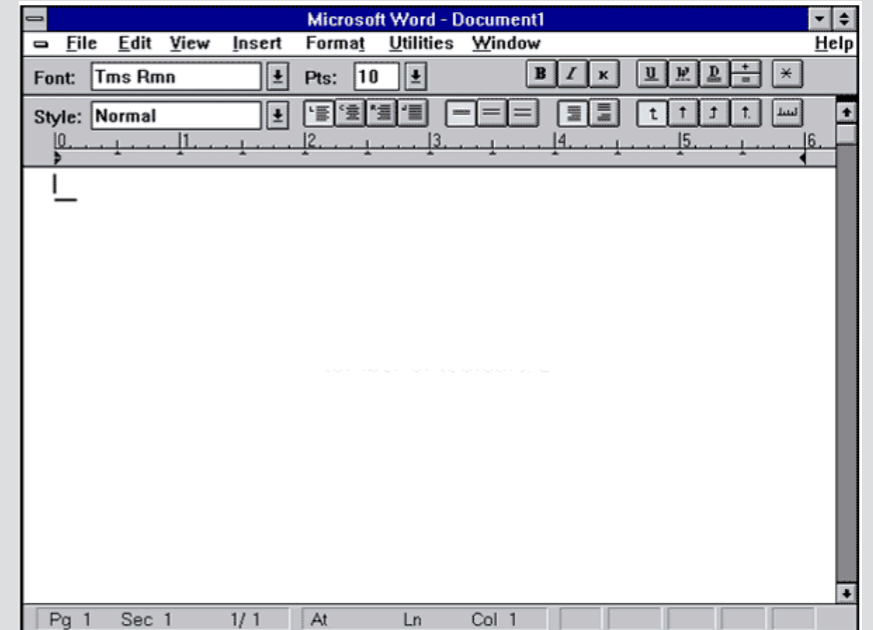
concepts are invented, not just out there



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



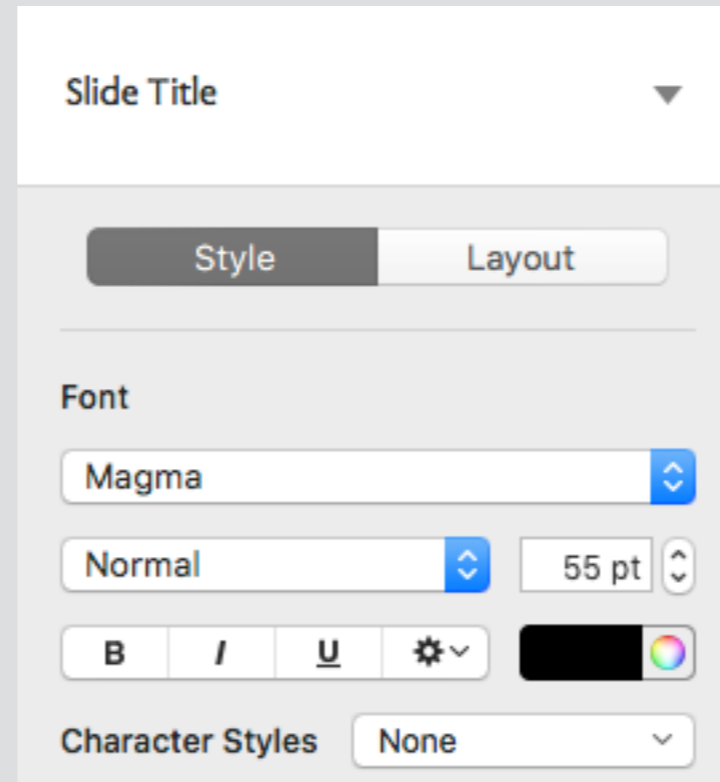
Charles Simonyi's team
implements style in
Bravo text editor



Simonyi brings style
to Microsoft in 1983

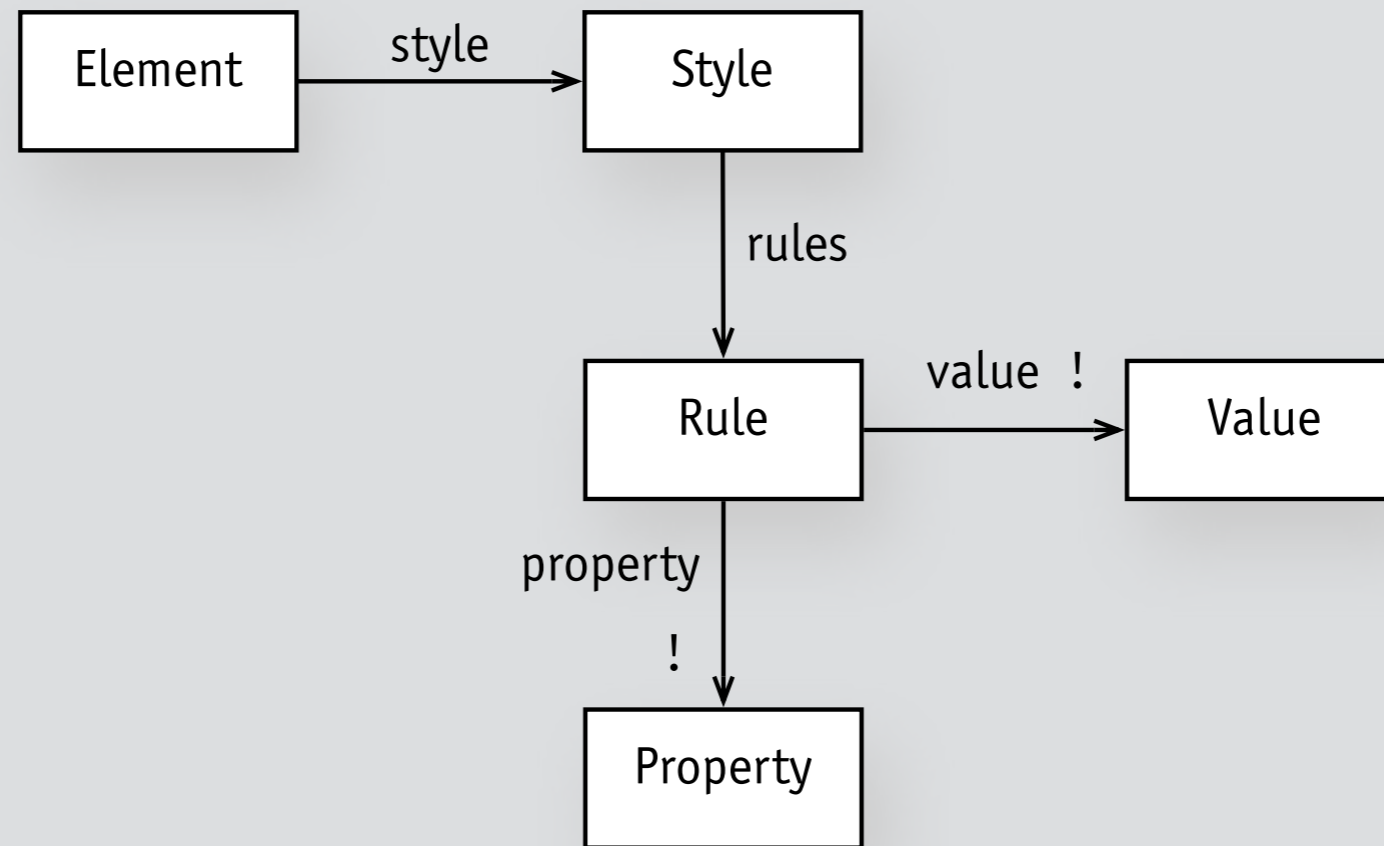
concepts have purpose

purpose of **style**: enable consistent formatting

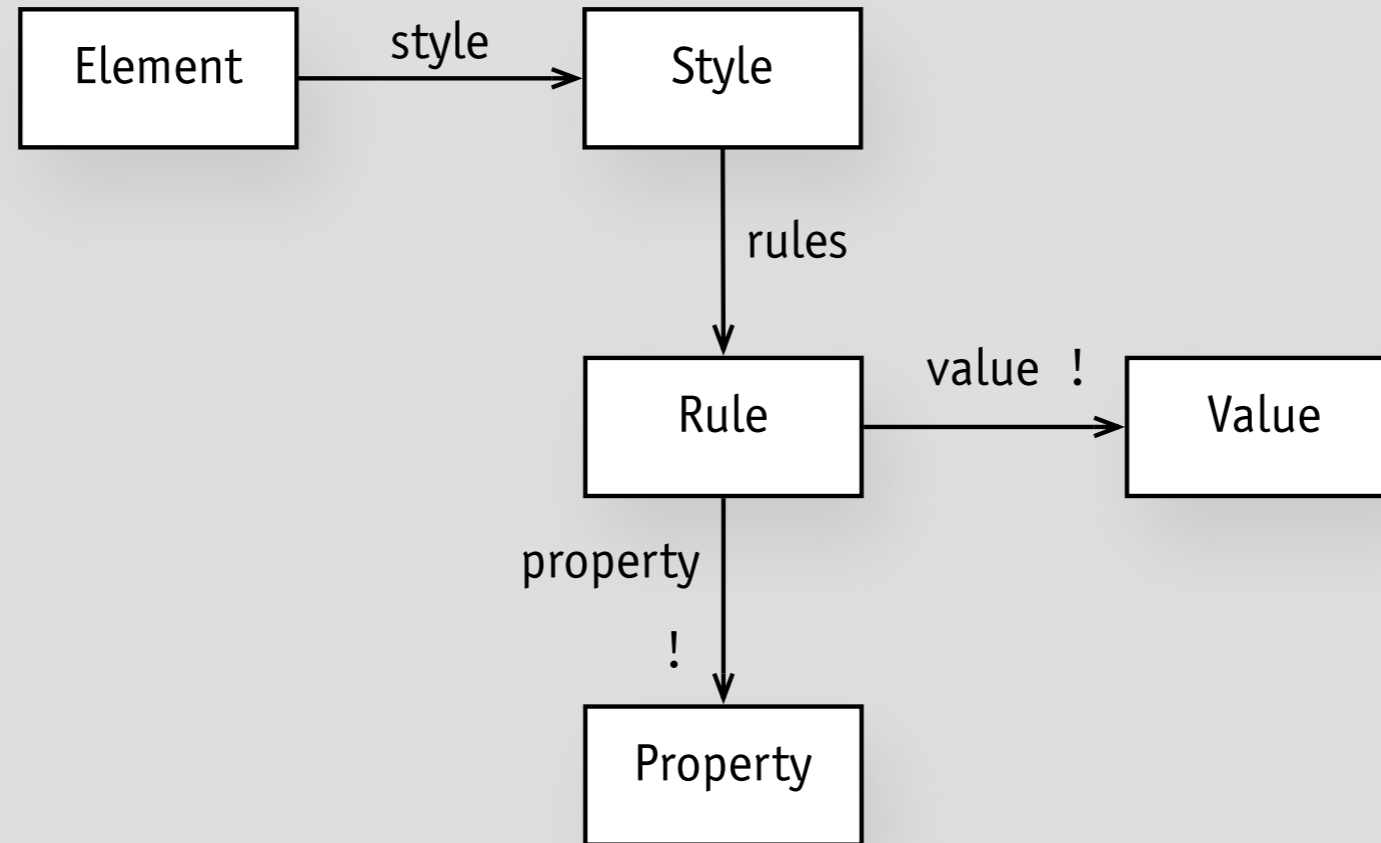


Apple Keynote
adds style concept
(2017?)

concept structure is designed not discovered

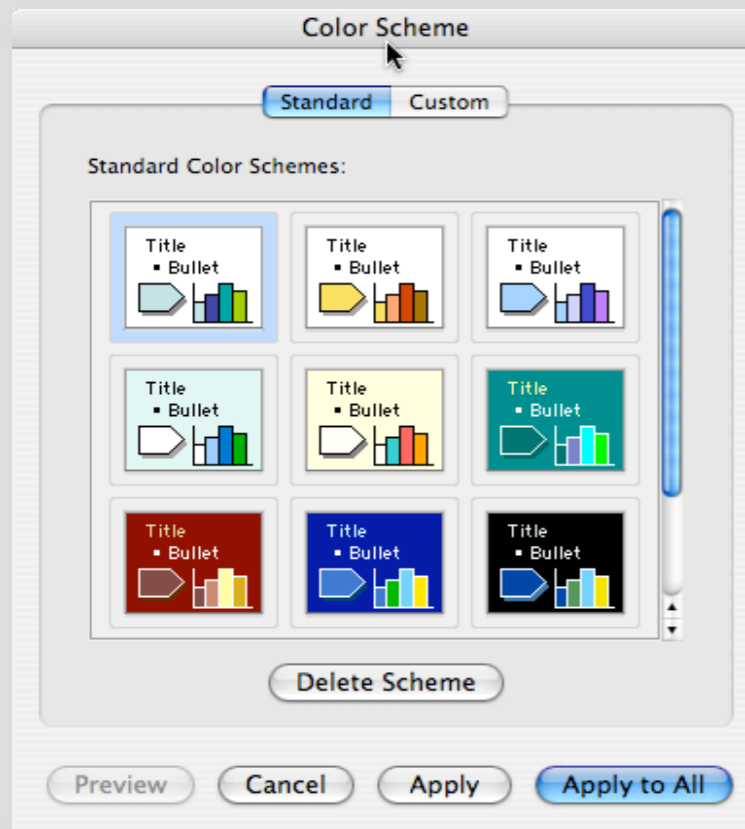


concept structure is designed not discovered

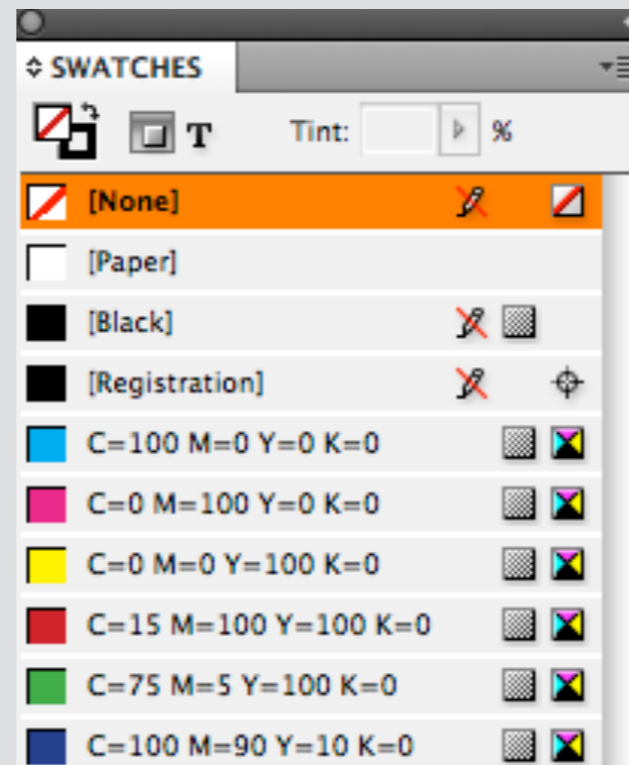


There is no problem
in computer science
that cannot be solved
by introducing another
level of indirection.
David Wheeler

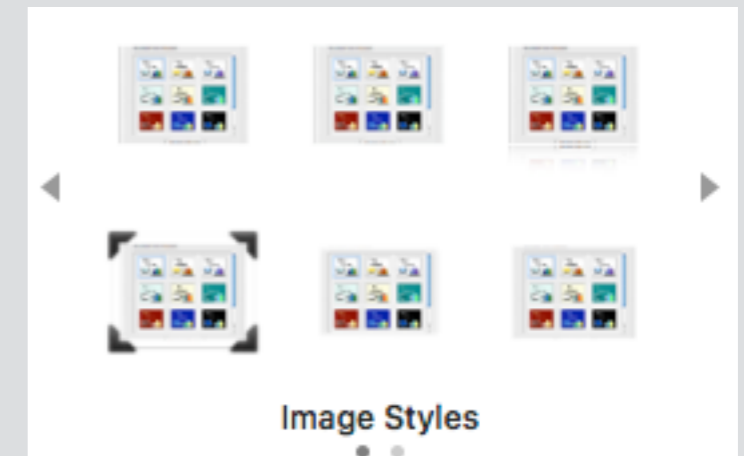
concepts are reusable



Powerpoint color schemes



Indesign swatches



Keynote image styles

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..."

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



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..."

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

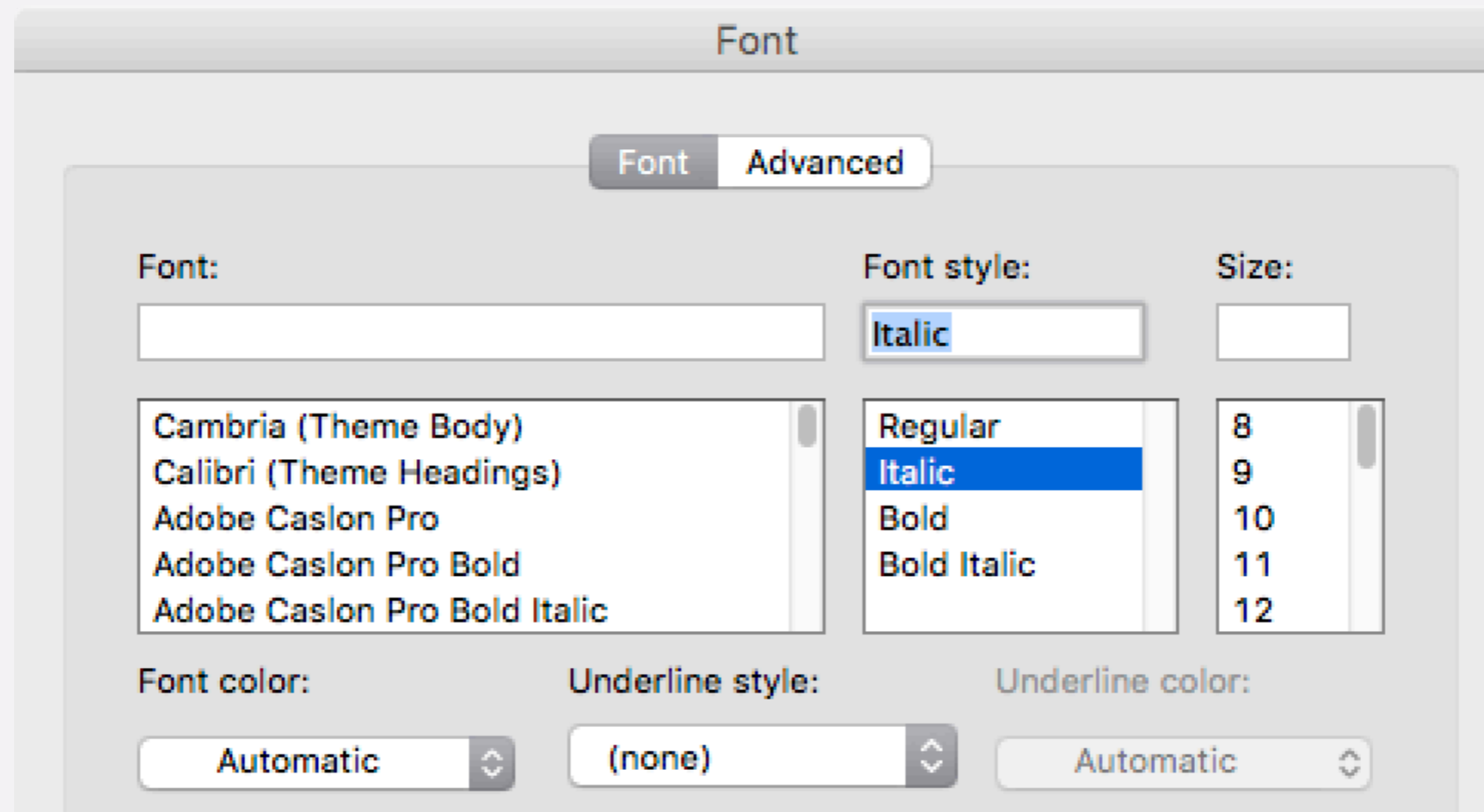


Johnson-Laird:
constructive semantics

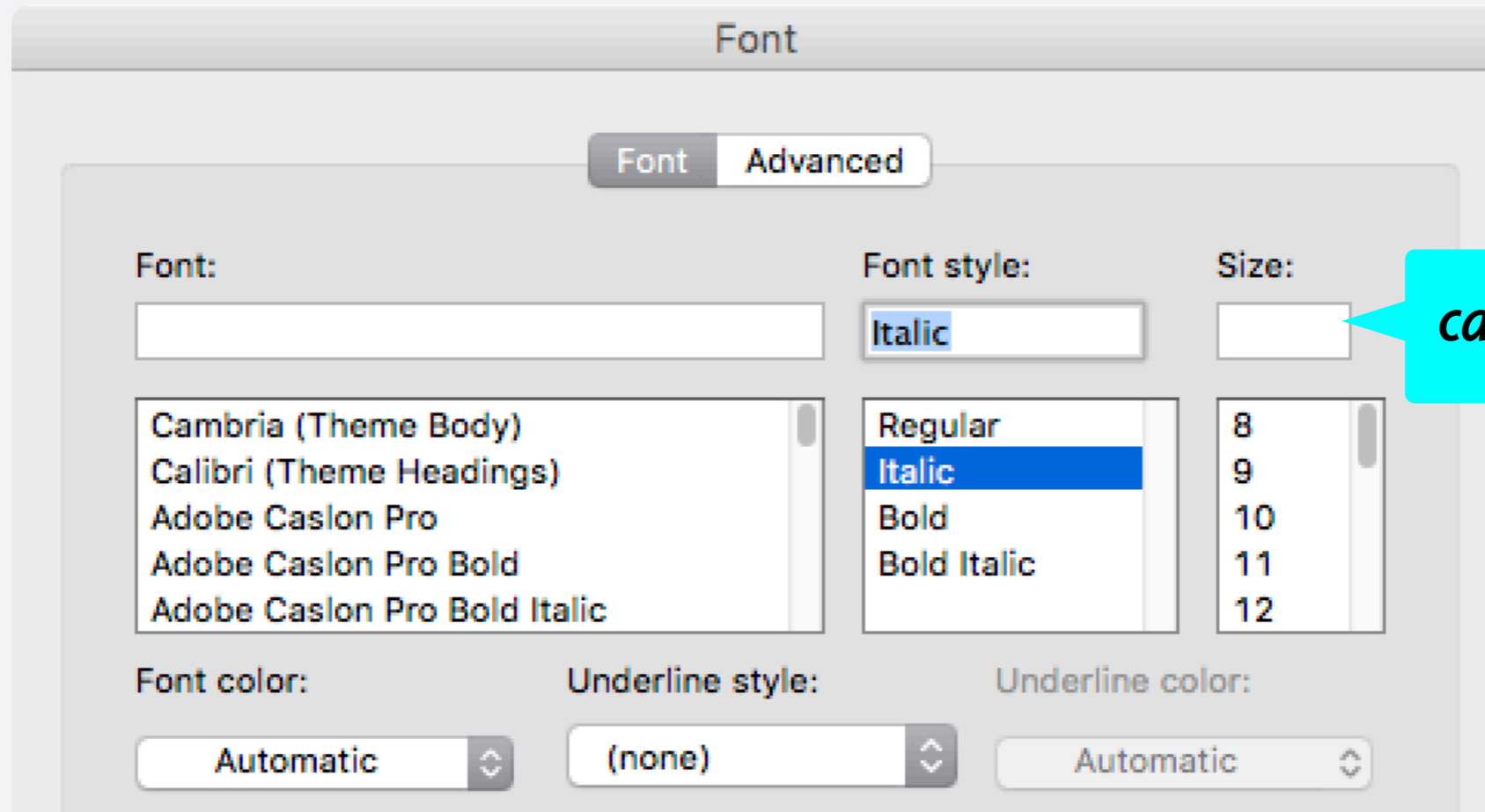


Michael Polanyi
operational principle

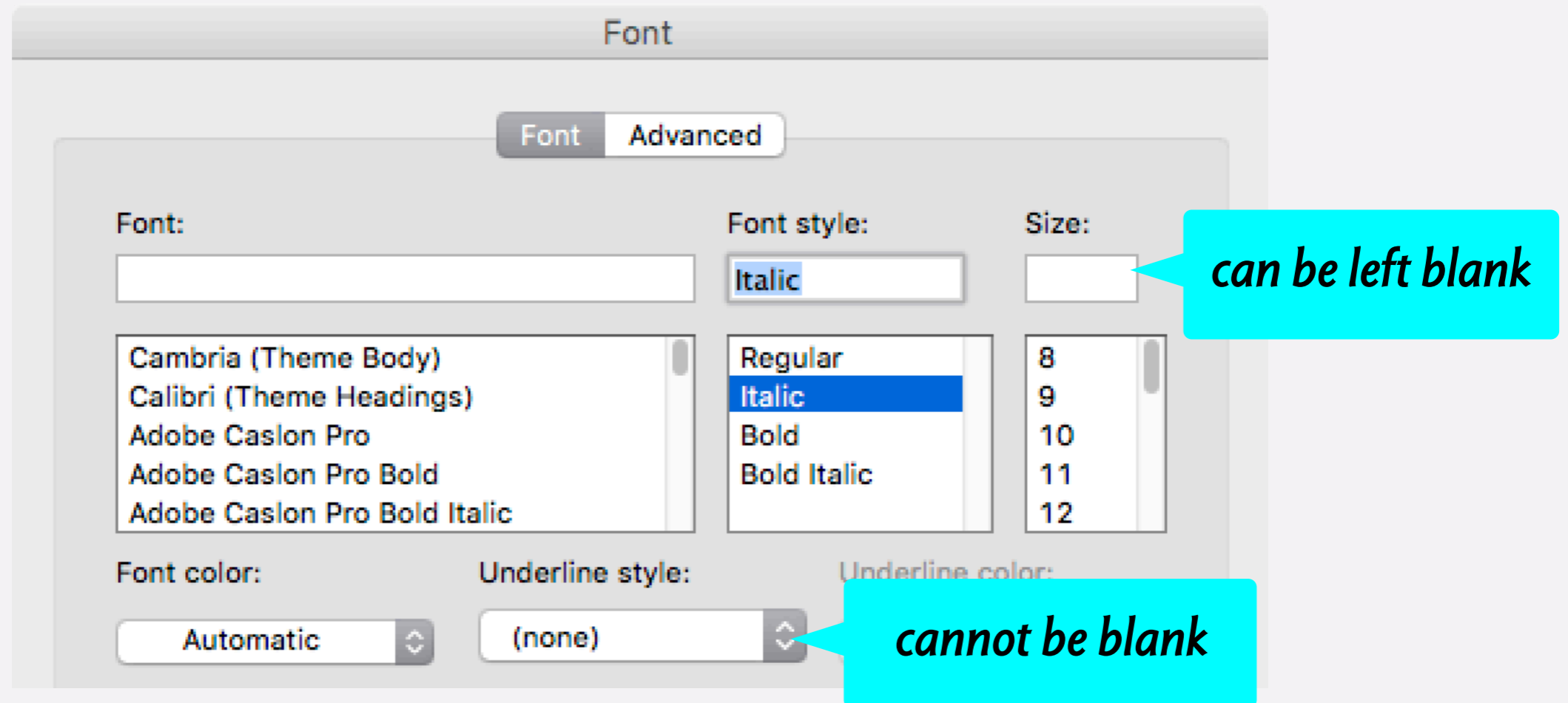
when concepts don't fulfill purpose



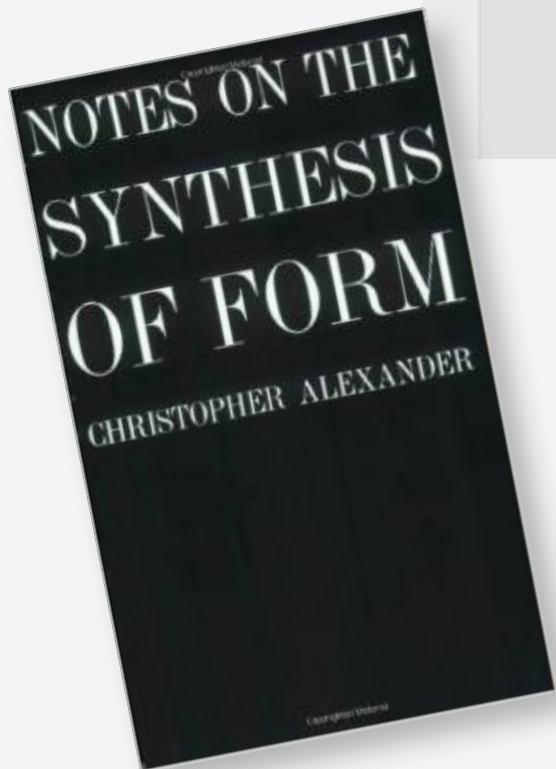
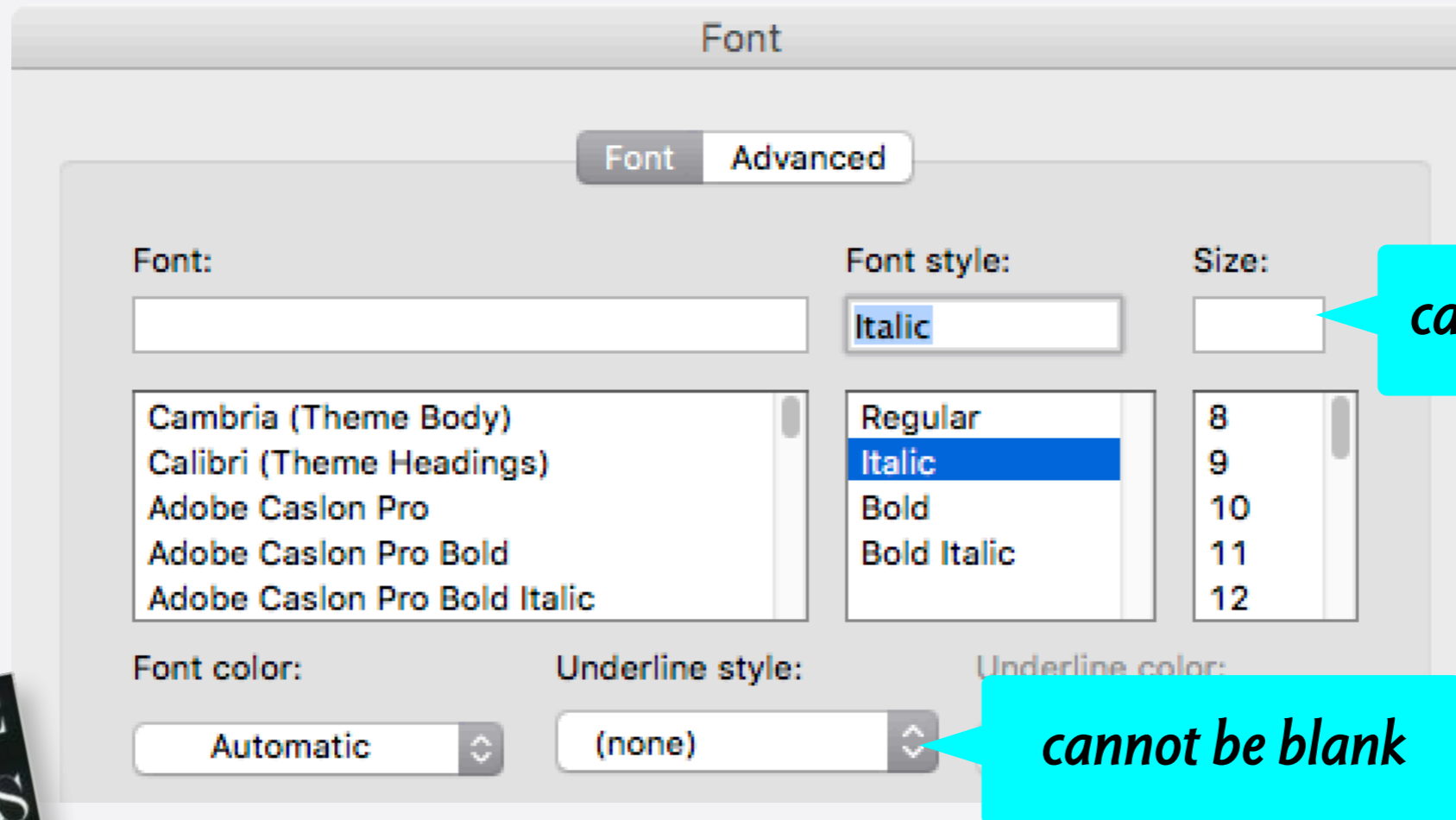
when concepts don't fulfill purpose



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

the style concept

the style concept

name

the style concept

name

purpose

the style concept

name

purpose

structure

the style concept

name

purpose

structure

actions

the style concept

name

purpose

structure

actions

operational principle

the style concept

name

style

purpose

structure

actions

operational principle

the style concept

name

style

purpose

enable consistent formatting

structure

actions

operational principle

the style concept

name

style

purpose

enable consistent formatting

structure

style: Element \rightarrow Style
value: Style \rightarrow Property \rightarrow Value

actions

operational principle

the style concept

name

style

purpose

enable consistent formatting

structure

style: Element \rightarrow Style
value: Style \rightarrow Property \rightarrow Value

actions

new (pv: Property \rightarrow Value) $\hat{=}$ new s | value := value + s \rightarrow pv
delete (s: Style) $\hat{=}$ style := style - Element \rightarrow s
assign (e: Element, s: Style) $\hat{=}$ style := style ++ e \rightarrow s
modify (s: Style, p: Property, v: Value) $\hat{=}$ value(s) := value(s) ++ p \rightarrow v

operational principle

the style concept

name

style

purpose

enable consistent formatting

structure

style: Element \rightarrow Style
value: Style \rightarrow Property \rightarrow Value

actions

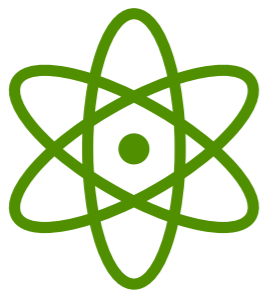
new (pv: Property \rightarrow Value) $\hat{=}$ new s | value := value + s \rightarrow pv
delete (s: Style) $\hat{=}$ style := style - Element \rightarrow s
assign (e: Element, s: Style) $\hat{=}$ style := style ++ e \rightarrow s
modify (s: Style, p: Property, v: Value) $\hat{=}$ value(s) := value(s) ++ p \rightarrow v

operational principle

if you create a new style, and assign it to two elements, and then modify the style, both elements will have the same property/value mapping

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]*

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 Ch		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

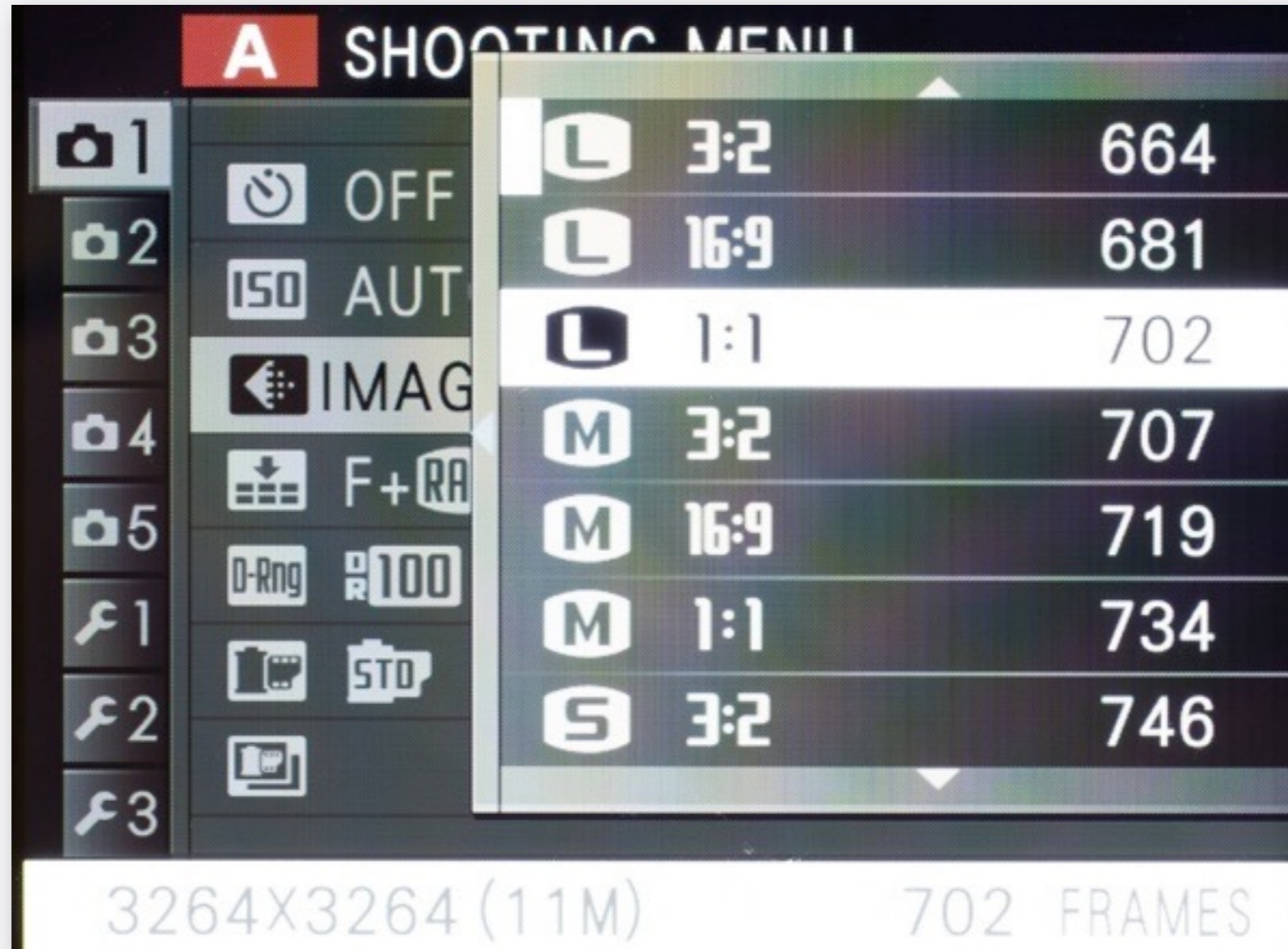


image size setting



non-standard ratio + raw?



what you can't do

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

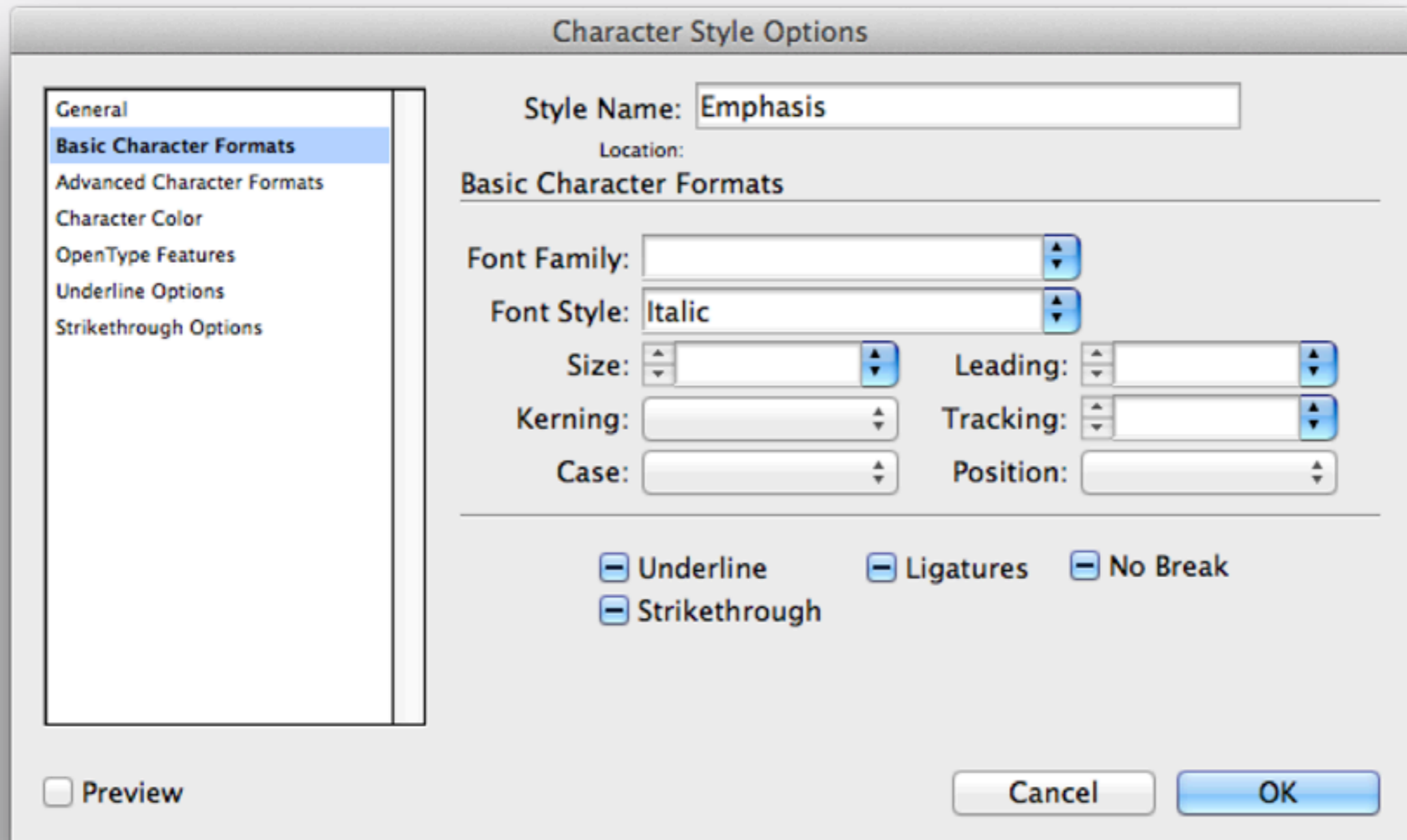
what you can't do

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

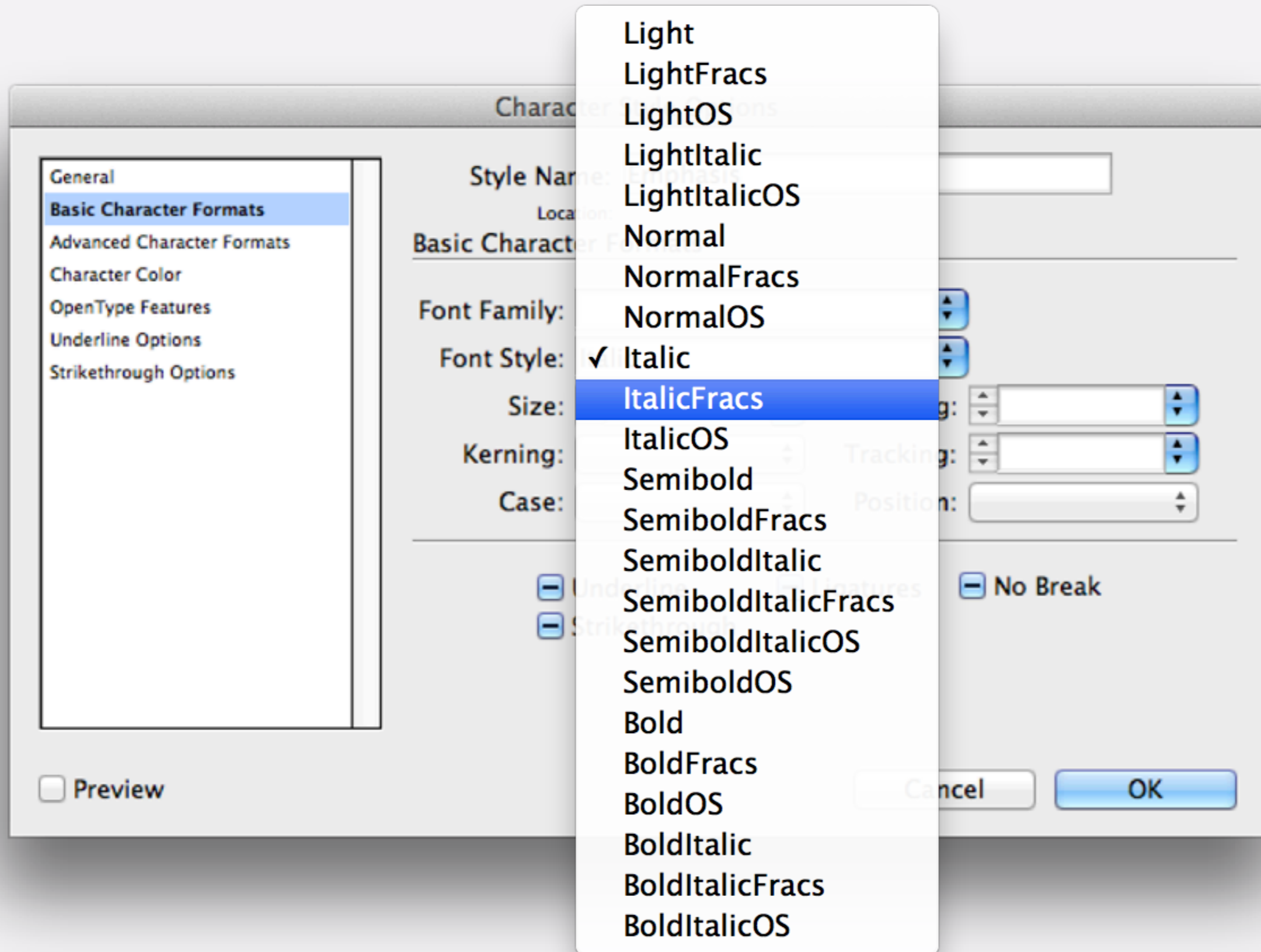


indesign
styles

what's a font?



what's a font?



what you can't do

define a style that italicizes

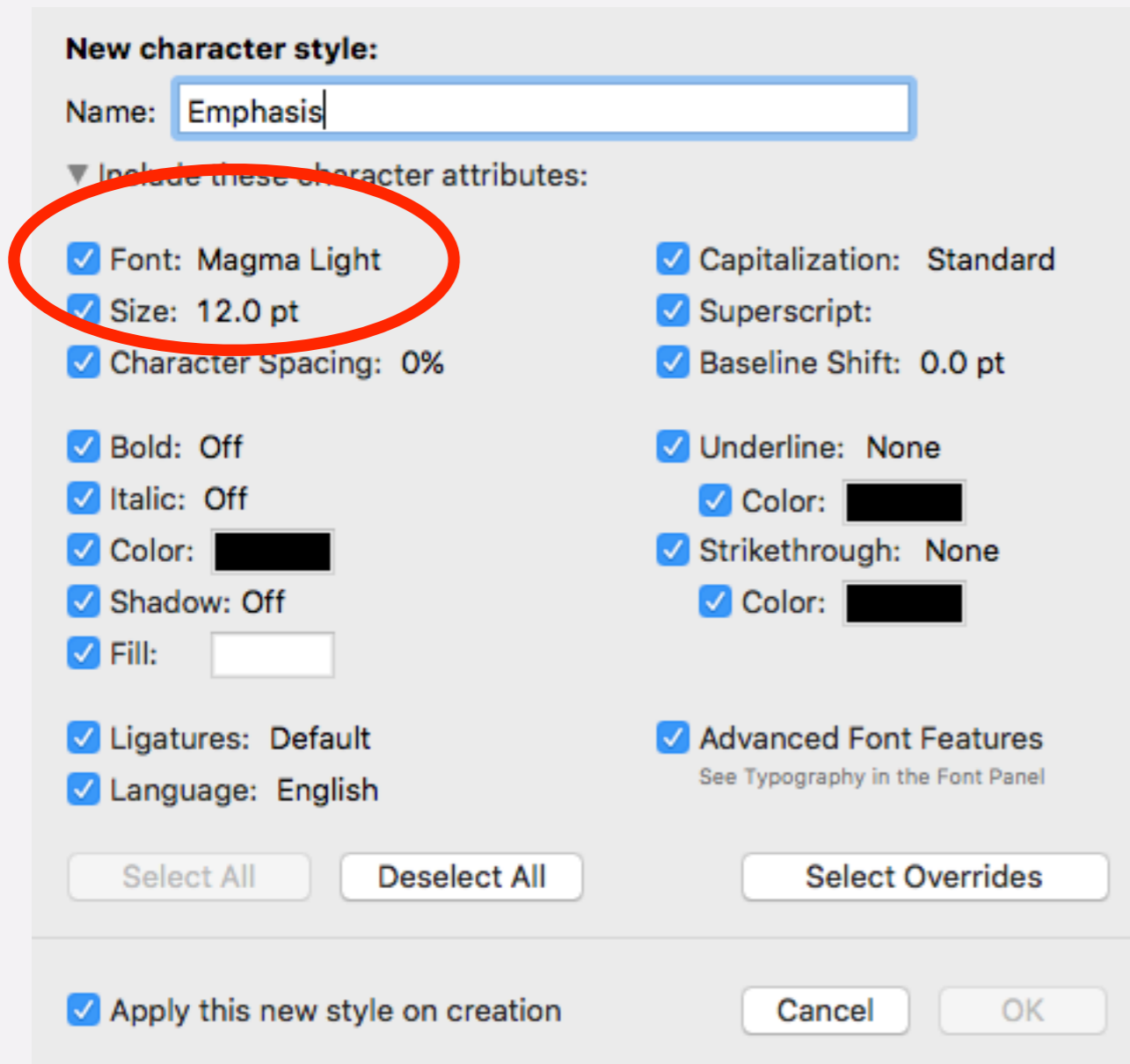
Arno Regular to Arno Italic

Futura Book to Futura Book Oblique

Magma Light to Magma Light Italic

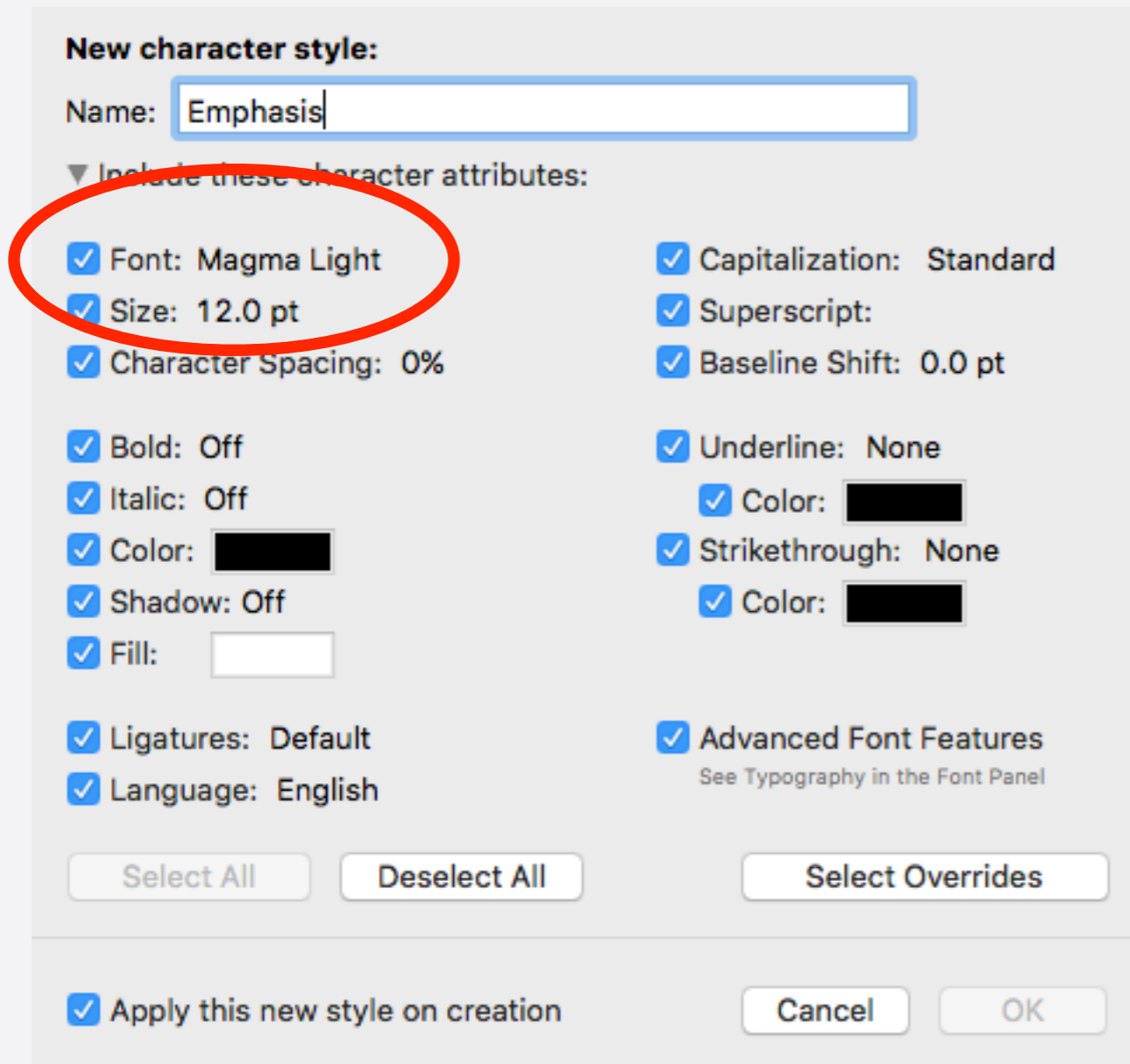
introducing a concept

introducing a concept

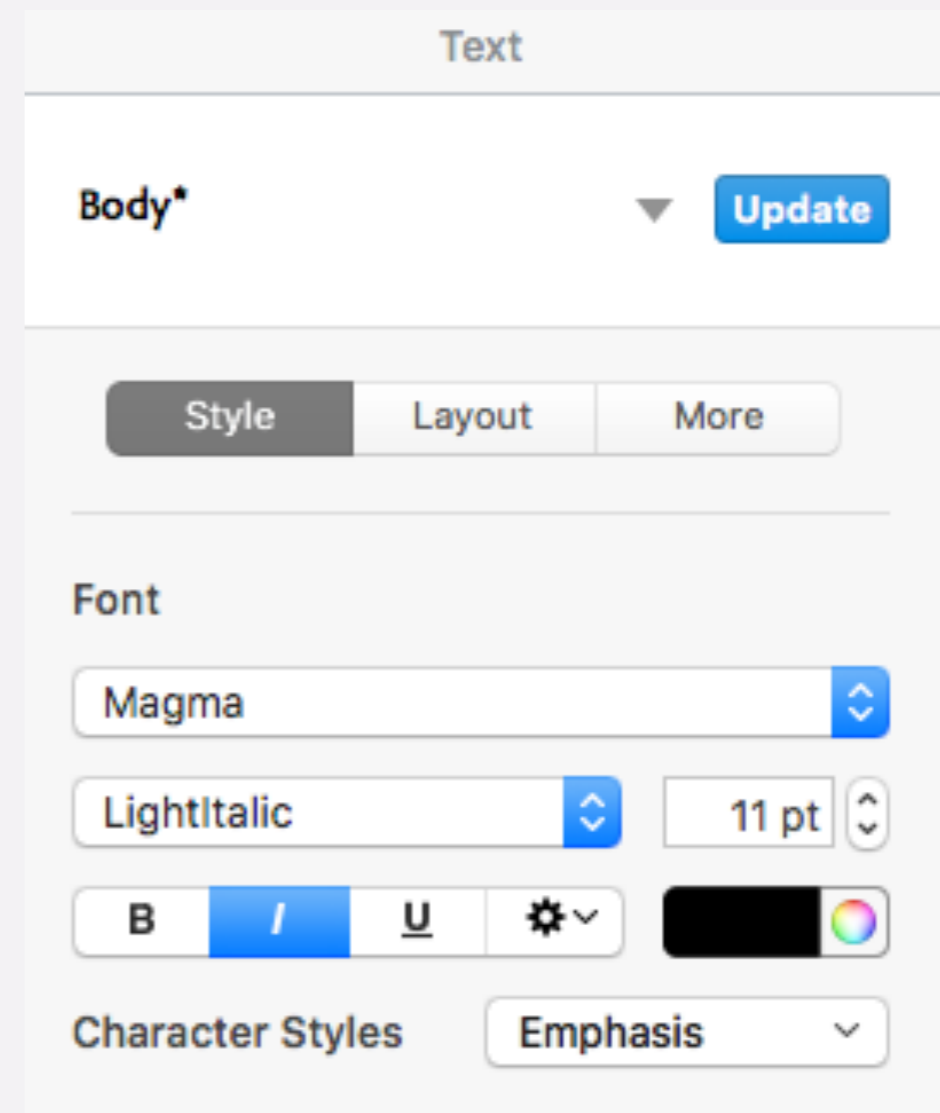


Keynote '09: has subfamilies

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

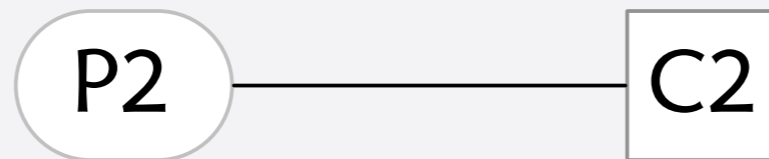
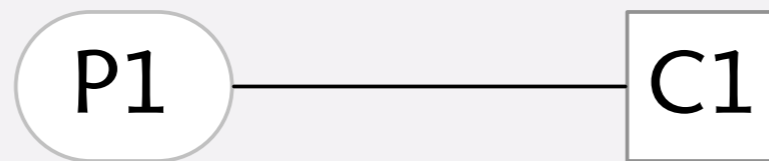
one purpose :: no concept
specify a font-independent styling :: ?

the
singularity
rule

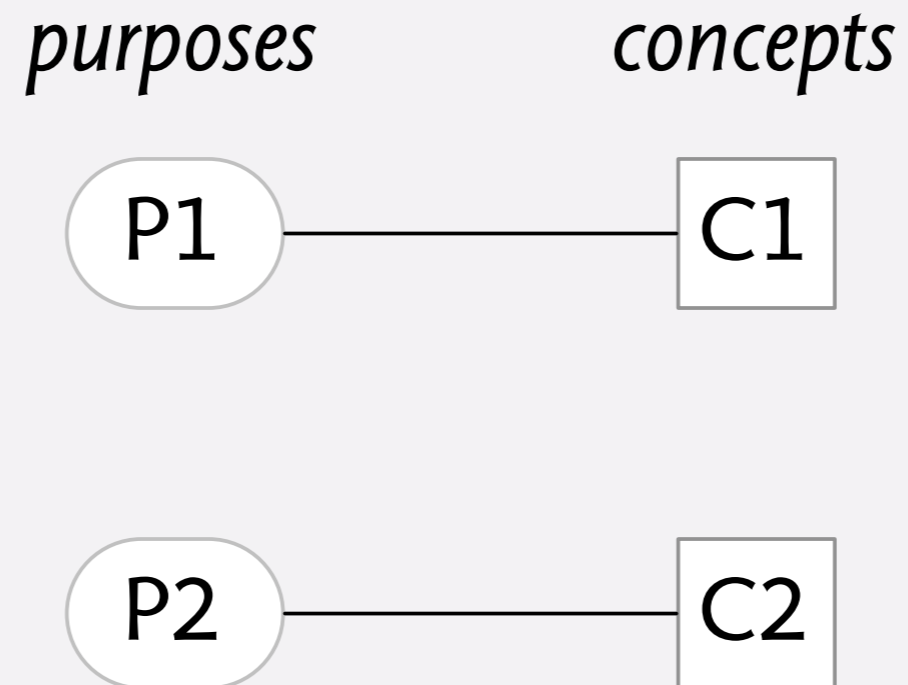
one-to-one mapping

purposes

concepts



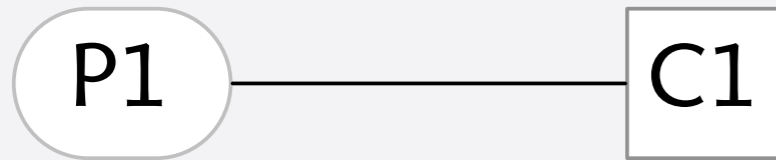
one-to-one mapping



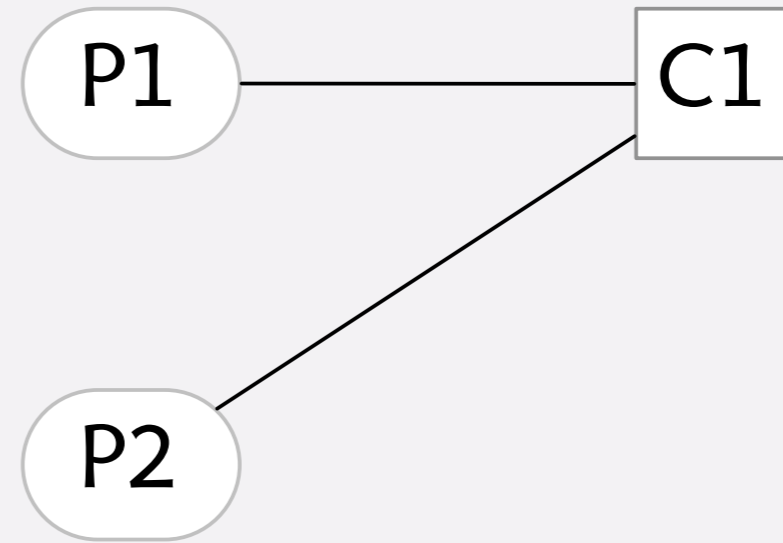
Nam Suh:
Axiomatic Design

four ways to fail

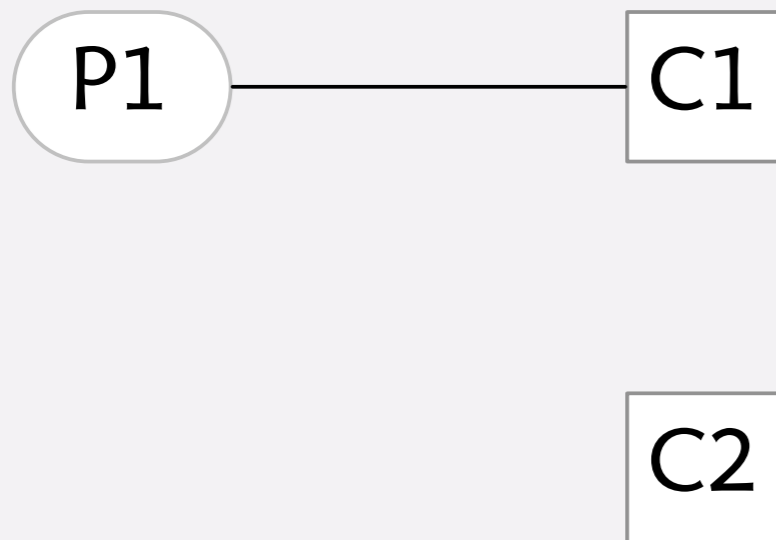
unfulfilled purpose



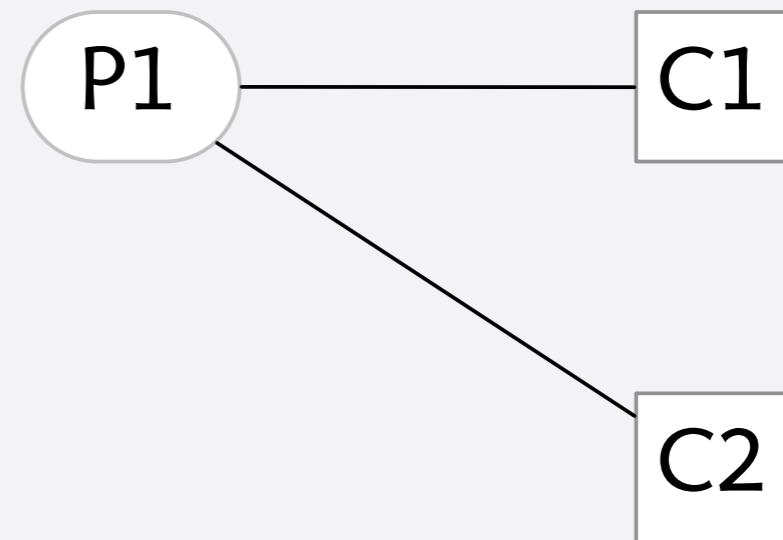
overloaded concept



unmotivated concept



redundant concepts



overloading fuji camera

new purpose piggybacked on old concept

overloading fuji camera

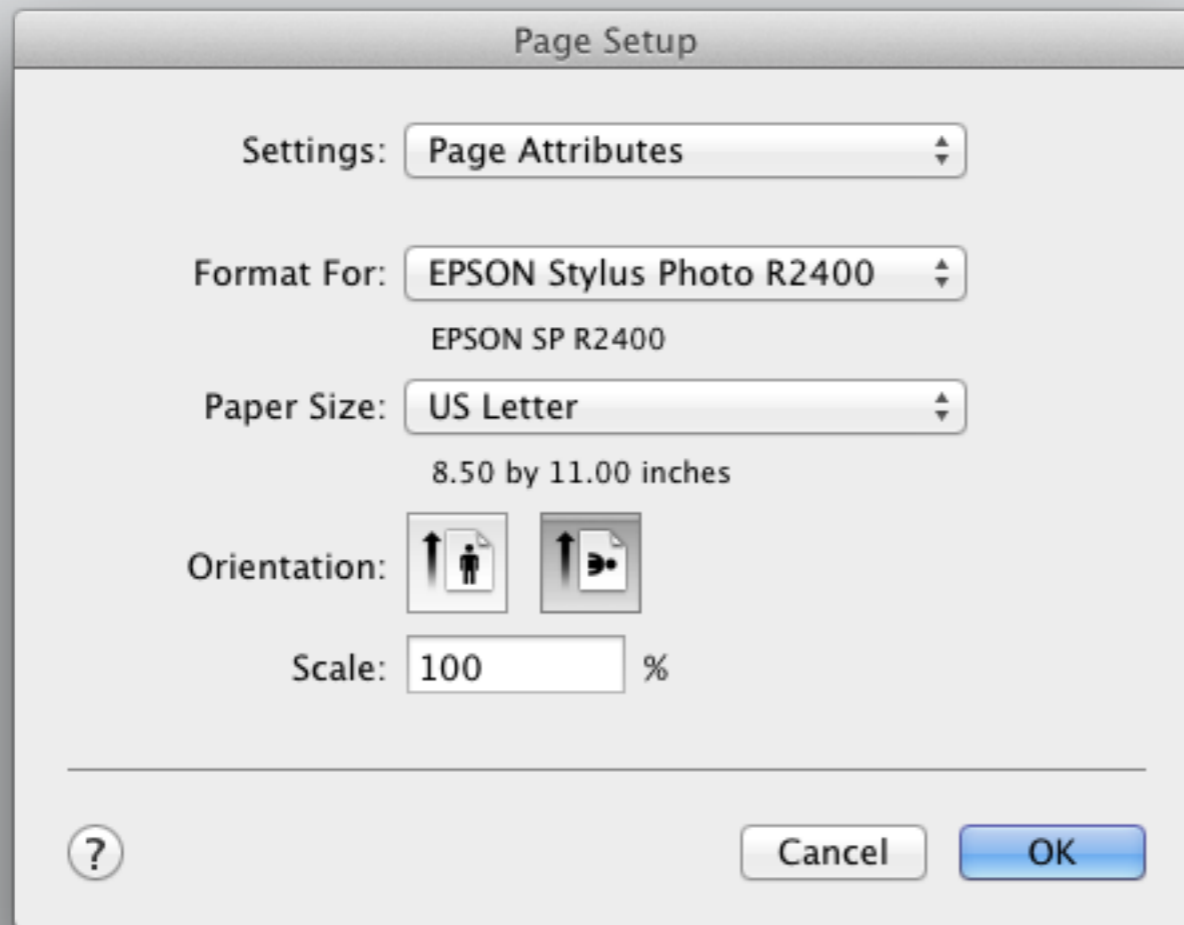
new purpose piggybacked on old concept

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

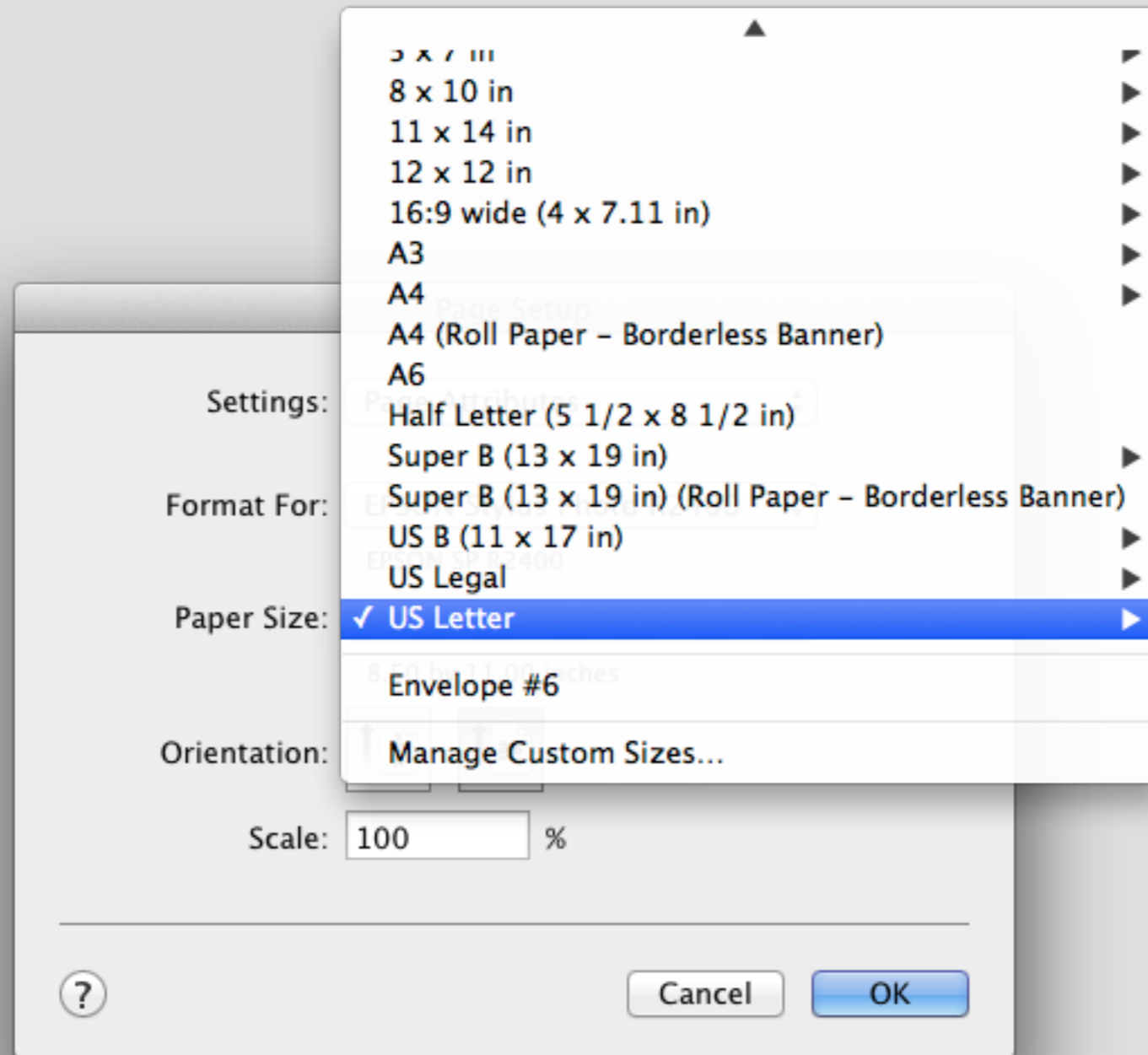
image size
aspect ratio piggybacked
on JPEG dimensions

overloading `epson driver`

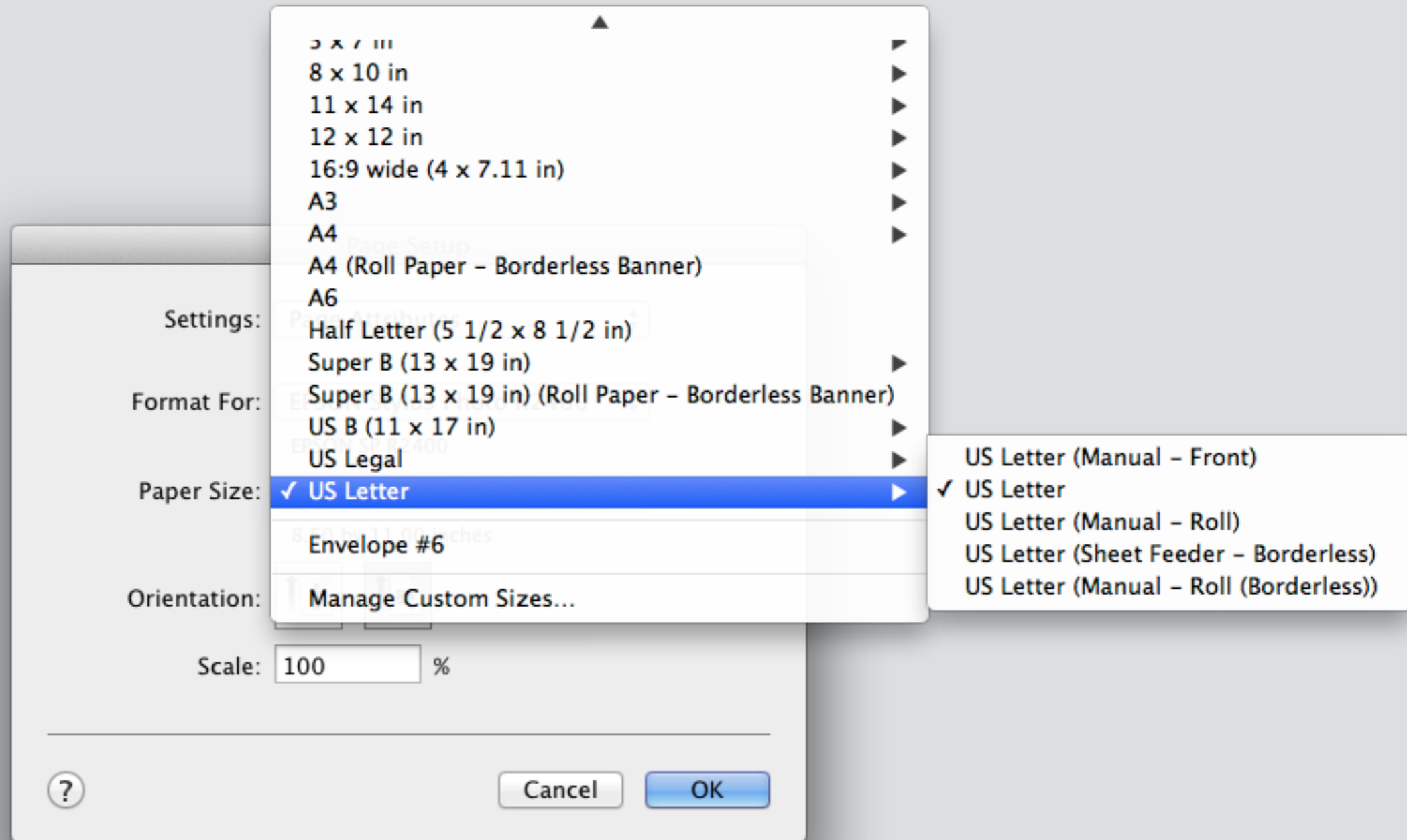
overloading epson driver



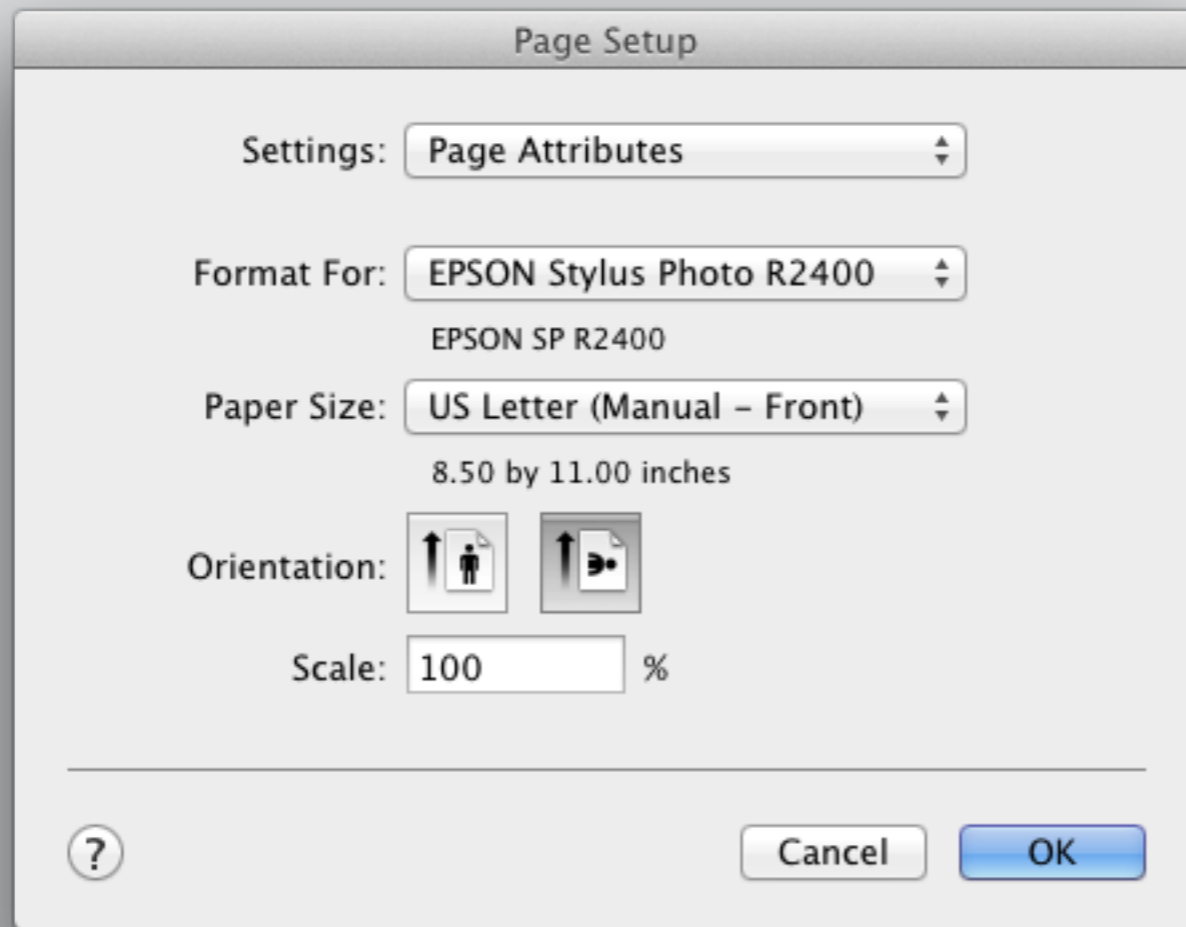
overloading epson driver



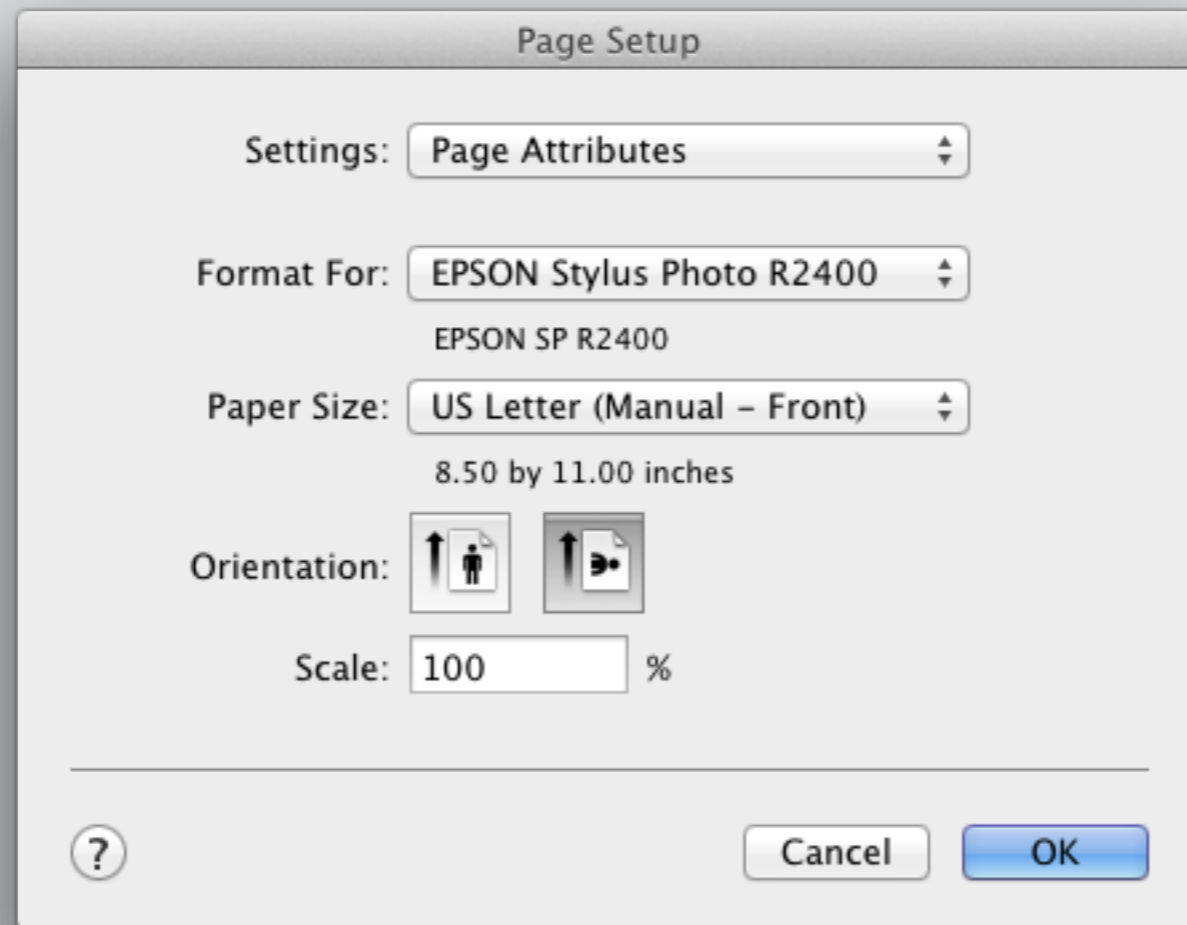
overloading Epson driver



overloading epson driver



overloading epson driver



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

overloading email subject

new purposes emerge for old concept

overloading email subject

new purposes emerge for old concept

To: Daniel Jackson <dnj@mit.edu>

Re: Catch me if you can in real life!

initial purpose: summarize content

overloading email subject

new purposes emerge for old 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

overloading email subject

new purposes emerge for old 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

the
uniformity
rule

the uniformity rule

the operational principle should apply uniformly over states, types, arguments

the uniformity rule

the operational principle should apply uniformly over states, types, arguments

The image shows a spreadsheet interface with a table and a formula bar. The table has two columns: 'item' and 'cost'. The rows are numbered 1 to 6. The 'cost' column contains values for 'apples' (\$4), 'bananas' (\$2), 'grapes' (\$6), and 'kiwis' (\$3). The formula bar shows a sum formula for the range B2:B5.

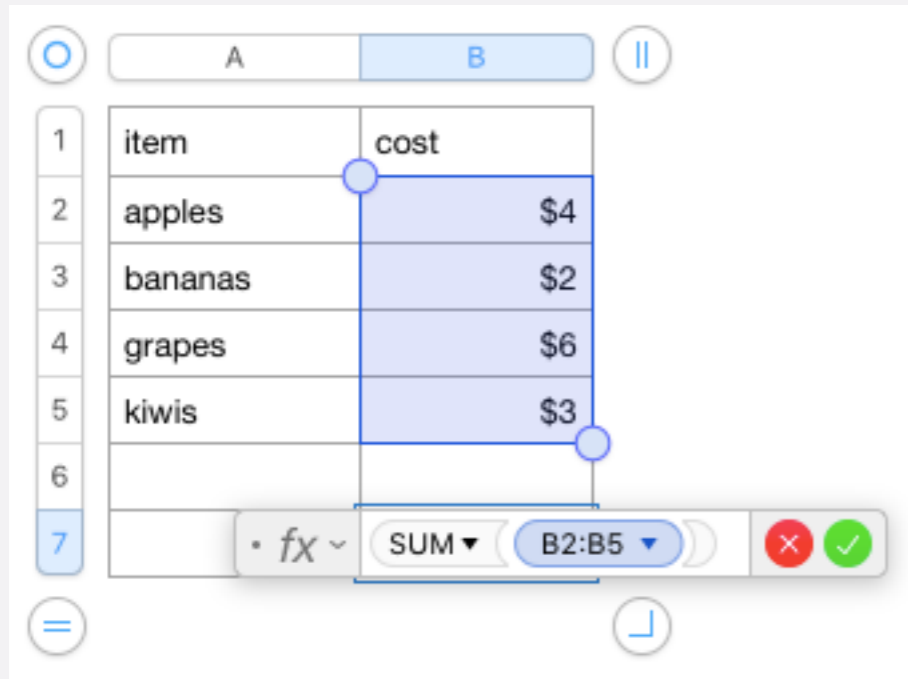
	A	B
1	item	cost
2	apples	\$4
3	bananas	\$2
4	grapes	\$6
5	kiwis	\$3
6		
7		

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

Numbers range: ... unless new cell is at end

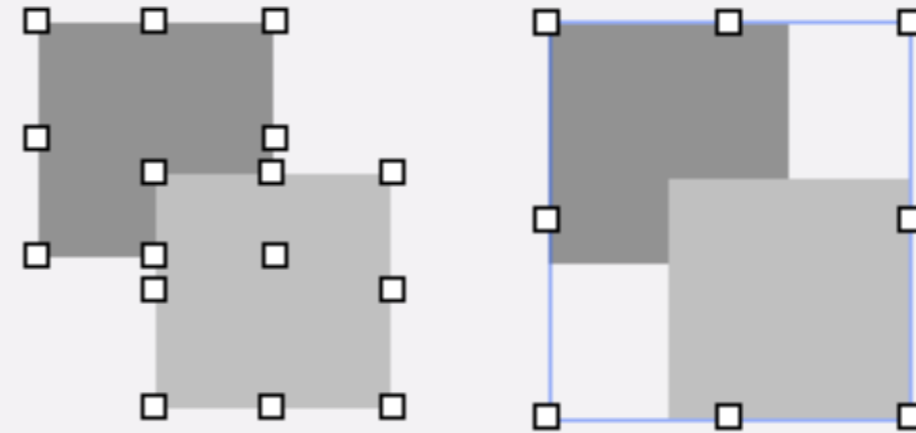
the uniformity rule

the operational principle should apply uniformly over states, types, arguments



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

fx SUM B2:B5



Keynote group: ... unless object is placeholder

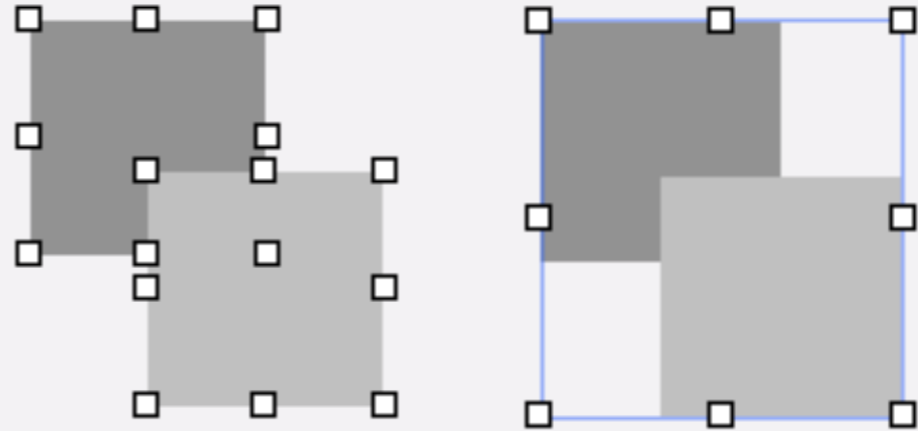
Numbers range: ... unless new cell is at end

the uniformity rule

the operational principle should apply uniformly over states, types, arguments

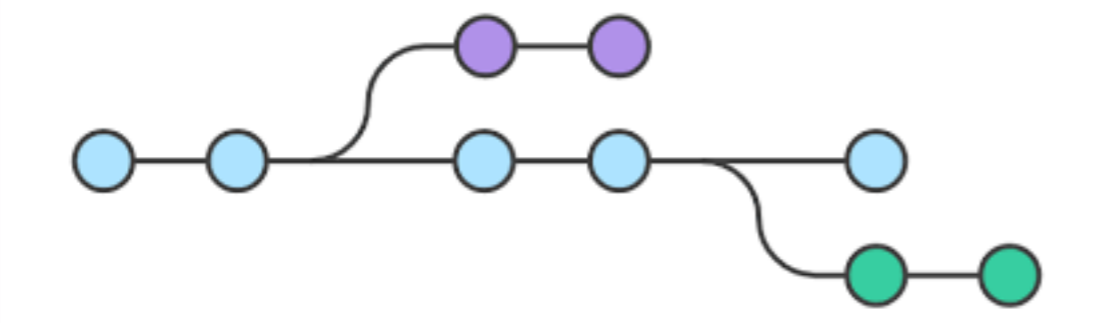
	A	B
1	item	cost
2	apples	\$4
3	bananas	\$2
4	grapes	\$6
5	kiwis	\$3
6		
7		

fx SUM B2:B5



Keynote group: ... unless object is placeholder

Numbers range: ... unless new cell is at end



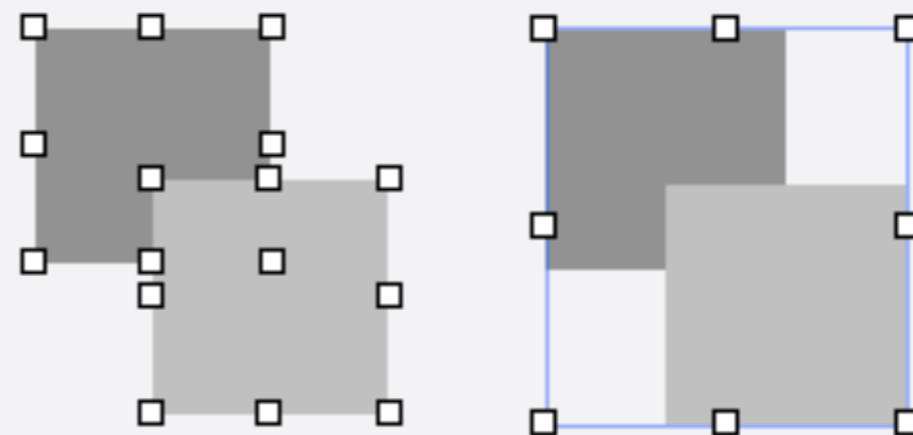
Git branch: ... unless conflicts

the uniformity rule

the operational principle should apply uniformly over states, types, arguments

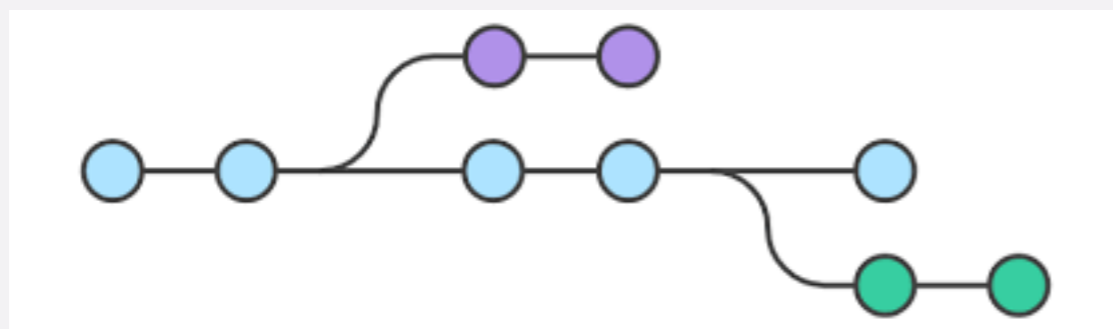
	A	B
1	item	cost
2	apples	\$4
3	bananas	\$2
4	grapes	\$6
5	kiwis	\$3
6		
7		

fx SUM B2:B5



Keynote group: ... unless object is placeholder

Numbers range: ... unless new cell is at end



Git branch: ... unless conflicts



Dropbox share: ... unless ancestor

the
genericity
rule

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 Keynote



the genericity rule

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

in Keynote

A vertical navigation pane from a Keynote presentation. It shows a list of slides with their titles and slide numbers. The slides are: slide 14 (title: "course: analyzing a design"), slide 15 (title: "kinds of overloading"), slide 22 (title: "the uniformity rule"), slide 27 (title: "generic concepts"), slide 28 (title: "how concepts get applied"), slide 29 (title: "why reuse a concept?"), and slide 30 (title: "using vs. inventing").

in Powerpoint

A vertical navigation pane from a Powerpoint presentation. It shows a list of slides with their titles and slide numbers. The slides are: slide 27 (title: "generic concepts"), slide 28 (title: "how concepts get applied"), slide 29 (title: "why reuse a concept?"), and slide 30 (title: "reuse and not").

the genericity rule

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

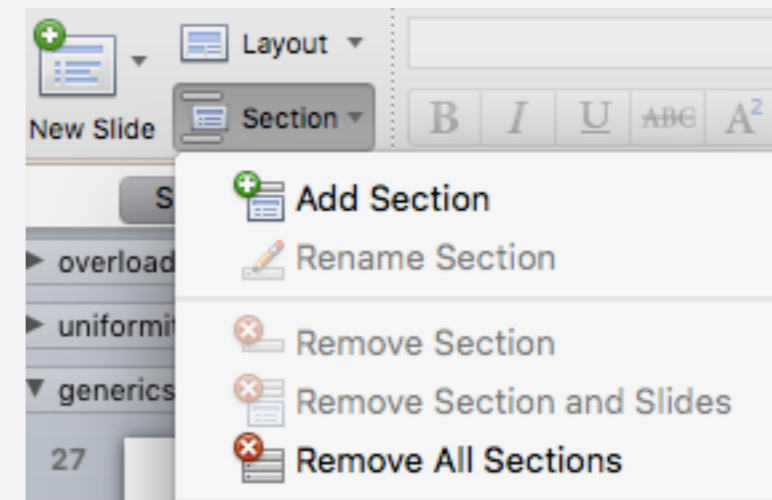
in Keynote



in Powerpoint

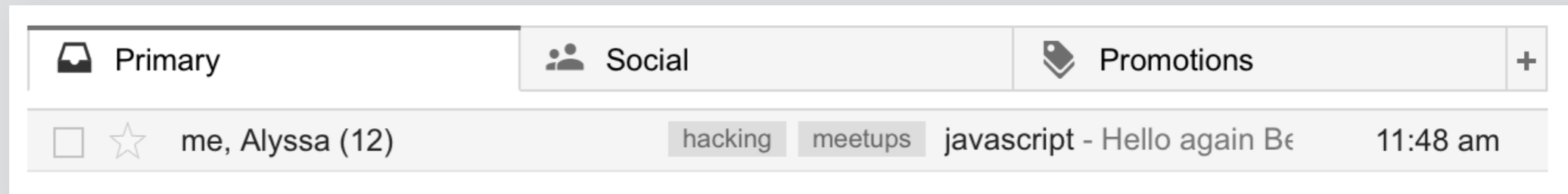


Powerpoint commands



the
integrity
rule

slightly surprising gmail behavior



slightly surprising gmail behavior

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

slightly surprising gmail behavior

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

slightly surprising gmail behavior

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.

less surprising but more problematic?

The screenshot shows the Gmail interface. At the top left, there is a 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 icons for a checkbox, refresh, and a vertical ellipsis. On the right side of this bar, it says "1-2 of 2" with navigation arrows and a keyboard icon. On the left side, there is a "Compose" button and a list of folders: "Inbox" (with a red badge showing "1"), "Starred", and "Snozed". The main area shows three email categories: "Primary" (selected), "Social", and "Promotions". Below these are two email entries:

Category	Sender	Subject	Date
Primary	Google Photos	Updates Ben, finish setting up your Google Photos account - Google Photos Welcome to Google Photos! ...	Oct 16
Primary	Alyssa, me 4	hacking meetups javascript - Is JavaScript just Scheme with prototypes and some hacky coercions? On M...	Jul 30

less surprising but more problematic?

This screenshot shows the Gmail interface with a search filter of 'in:sent'. The search results are displayed in a list view. The first result is from 'Google Photos' with the subject 'Ben, finish setting up your Google Photos account' and a date of 'Oct 16'. The second result 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'. The interface includes a search bar at the top, a left sidebar with navigation options like 'Compose', 'Inbox', 'Starred', and 'Snozed', and a top navigation bar with 'Primary', 'Social', and 'Promotions' tabs.

Search mail

Compose

Inbox 1

Starred

Snozed

Primary Social Promotions

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

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

This screenshot shows the Gmail search results for 'in:sent'. The search bar at the top contains 'in:sent'. The search results are displayed in a list view. The first result is from 'To: Alyssa 4' with the subject 'javascript - Is JavaScript just Scheme with prototypes and some hacky coercions...' and a time of '1:15 PM'. The interface includes a search bar at the top, a left sidebar with navigation options like 'Compose', 'Inbox', 'Starred', and 'Snozed', and a top navigation bar with 'Primary', 'Social', and 'Promotions' tabs.

in:sent

Compose

Inbox 1

Starred

Snozed

Primary Social Promotions

To: Alyssa 4 Inbox hacking meetups javascript - Is JavaScript just Scheme with prototypes and some hacky coercions... 1:15 PM

less surprising but more problematic?

The screenshot shows the Gmail interface with a search bar containing 'javascript'. The search results are displayed in a list view. The first result is from 'Google Photos' with the subject 'Ben, finish setting up your Google Photos account' and a date of 'Oct 16'. The second result is from 'Alyssa, me 4' with the subject 'Is JavaScript just Scheme with prototypes and some hacky coercions? On M...' and a date of 'Jul 30'. The interface includes a 'Compose' button, a 'Search mail' bar, and navigation icons for 'Primary', 'Social', and 'Promotions'.

The screenshot shows a Gmail conversation thread for the search term 'javascript'. The thread includes three messages:

- Message 1:** From Alyssa P. Hacker <alyssa.pure.hacker@gmail.com> to me, dated Tue, May 8, 9:14 PM. Subject: 'Reminds you of the old days, eh?'.
- Message 2:** From Ben Bitdiddle <benito.bitdiddle@gmail.com> to Alyssa, dated Tue, May 8, 9:40 PM. Subject: 'Yes, it does.'.
- Message 3:** From Alyssa P. Hacker, dated Mon, Jul 30, 1:24 PM. Subject: 'JavaScript makes me feel nostalgic for Scheme.'.
- Message 4:** From Ben Bitdiddle <benito.bitdiddle@gmail.com> to Alyssa, dated 1:15 PM (1 minute ago). Subject: 'Is JavaScript just Scheme with prototypes and some hacky coercions?'.

The interface includes a search bar with 'in:sent', a toolbar with navigation icons, and a '1 of 1' indicator.

less surprising but more problematic?

This screenshot shows the Gmail inbox interface. The search bar at the top contains the text 'in:sent'. Below the search bar, the inbox is filtered to show messages related to 'javascript'. The visible messages are:

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

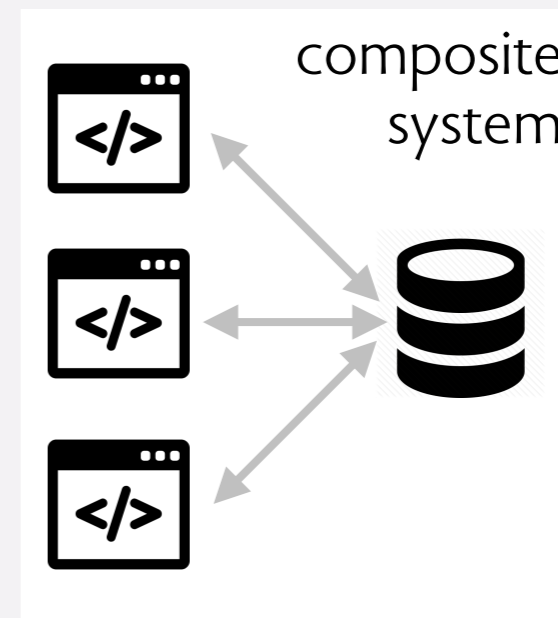
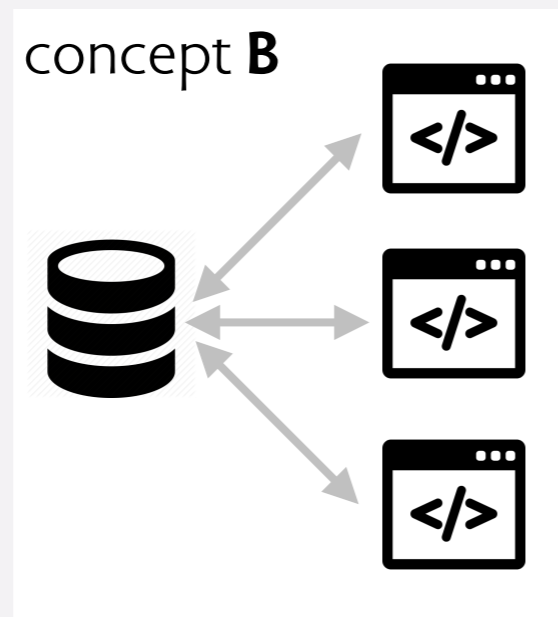
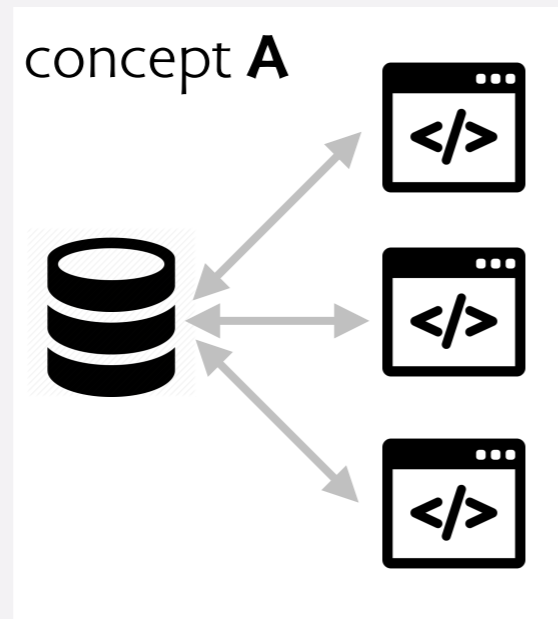
This screenshot shows a detailed view of a Gmail conversation thread for the search term 'javascript'. The thread includes the following messages:

- Alyssa P. Hacker** <alyssa.pure.hacker@gmail.com> to me (Tue, May 8, 9:14 PM): Reminds you of the old days, eh?
- Ben Bitdiddle** <benito.bitdiddle@gmail.com> to Alyssa (Tue, May 8, 9:40 PM): Yes, it does.
- Alyssa P. Hacker** JavaScript makes me feel nostalgic for Scheme.
- Ben Bitdiddle** <benito.bitdiddle@gmail.com> to Alyssa (1:15 PM (1 minute ago)): Is JavaScript just Scheme with prototypes and some hacky coercions?

can't see which messages were sent

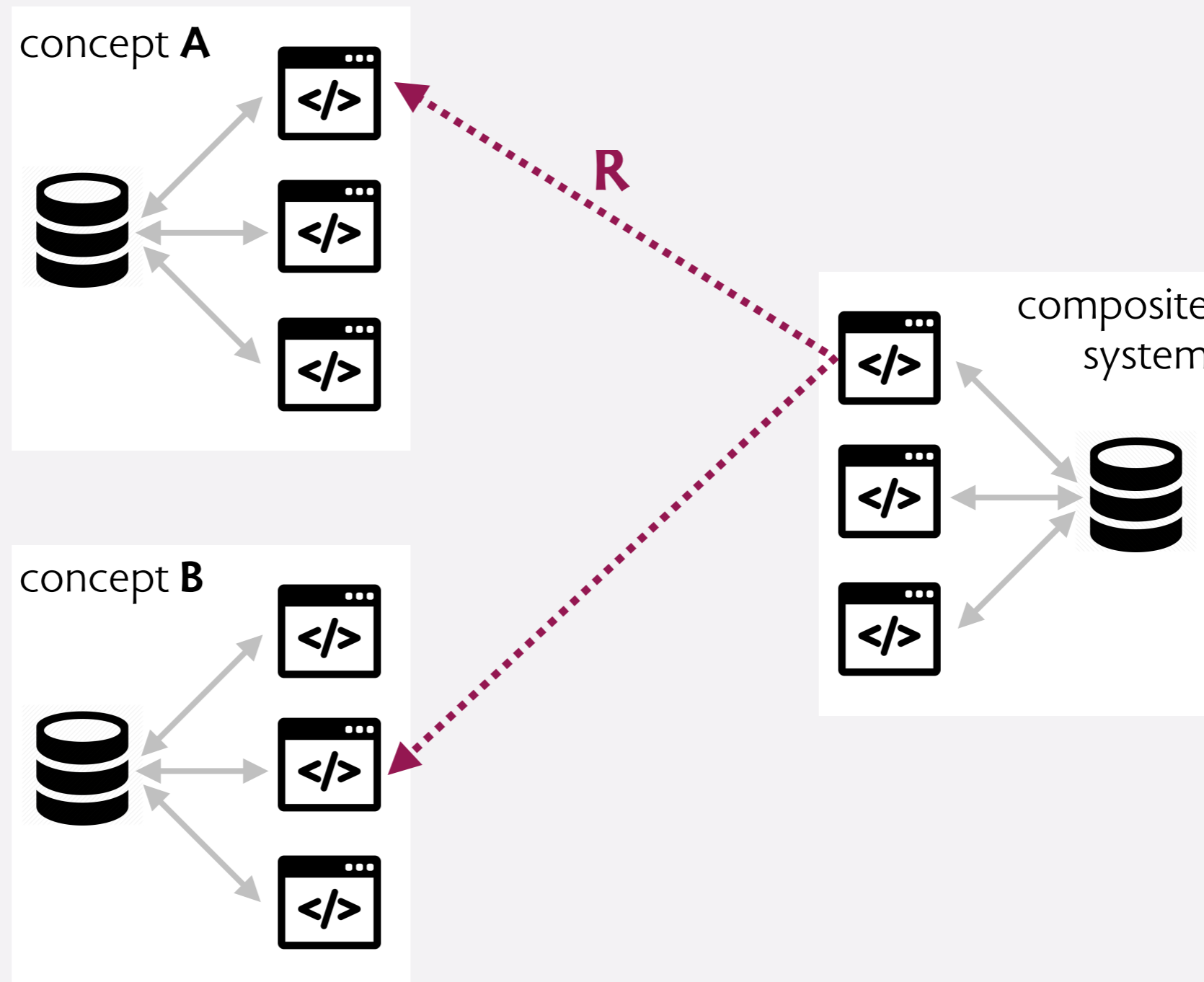
interpreting composite behavior

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



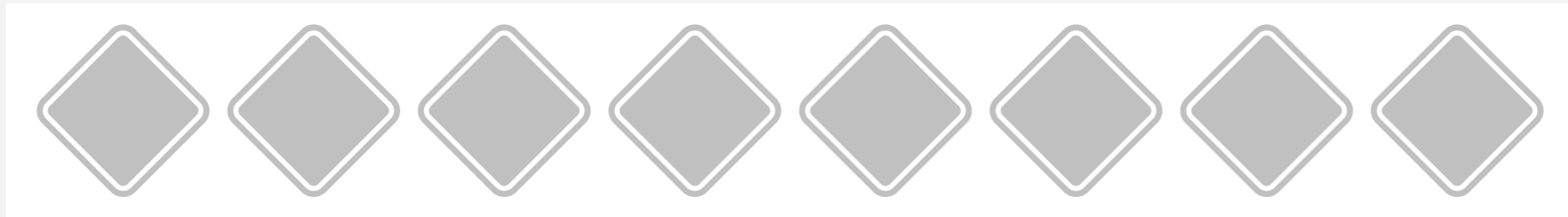
the integrity rule

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**

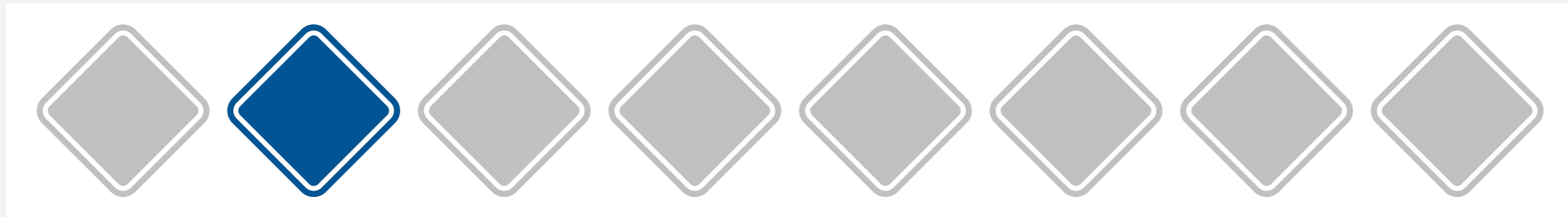
the integrity rule

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**

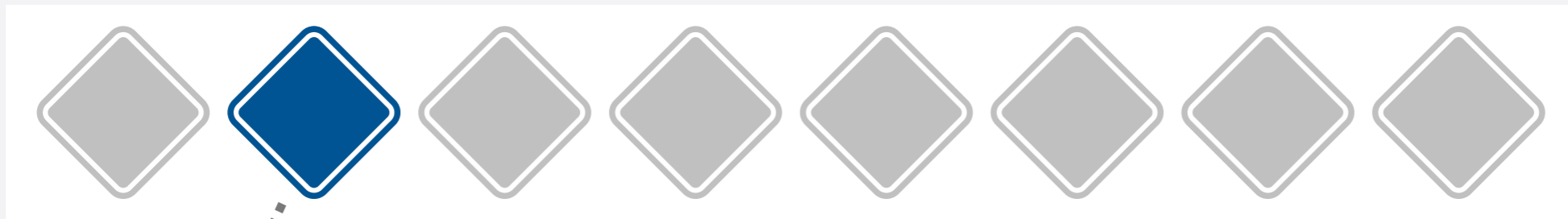
the integrity rule

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

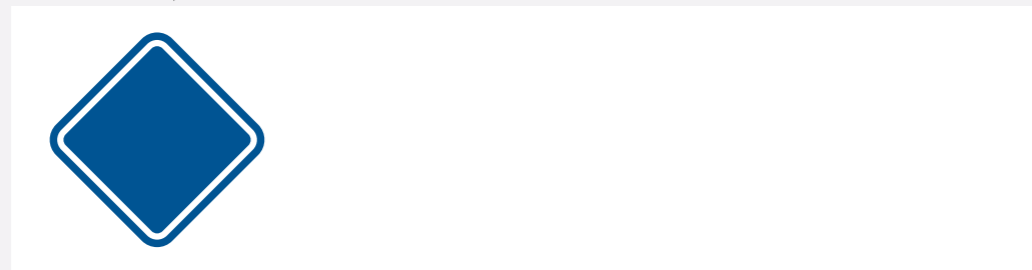
a simple criterion

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**

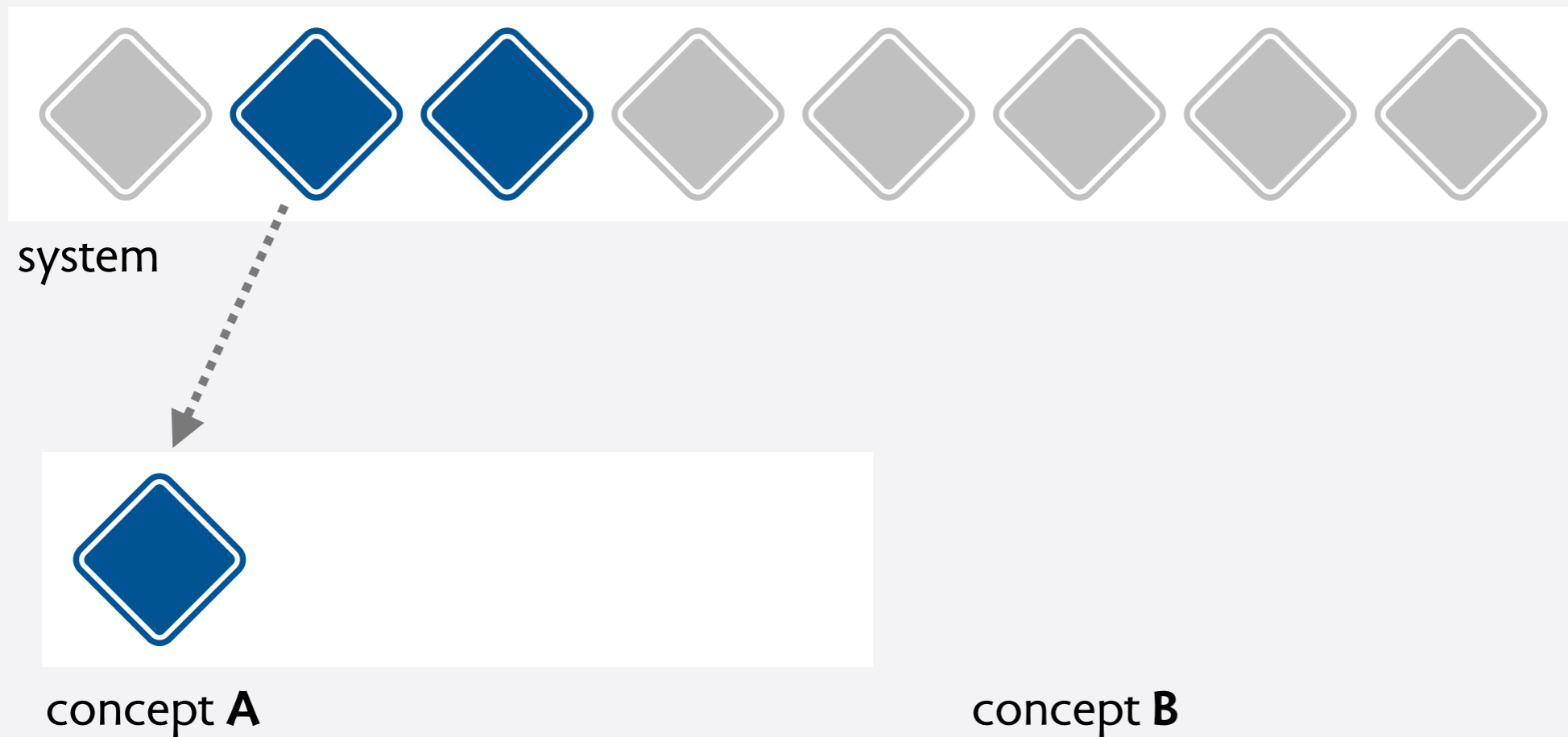
the integrity rule

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

a simple criterion

projected behavior must satisfy concept spec:

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



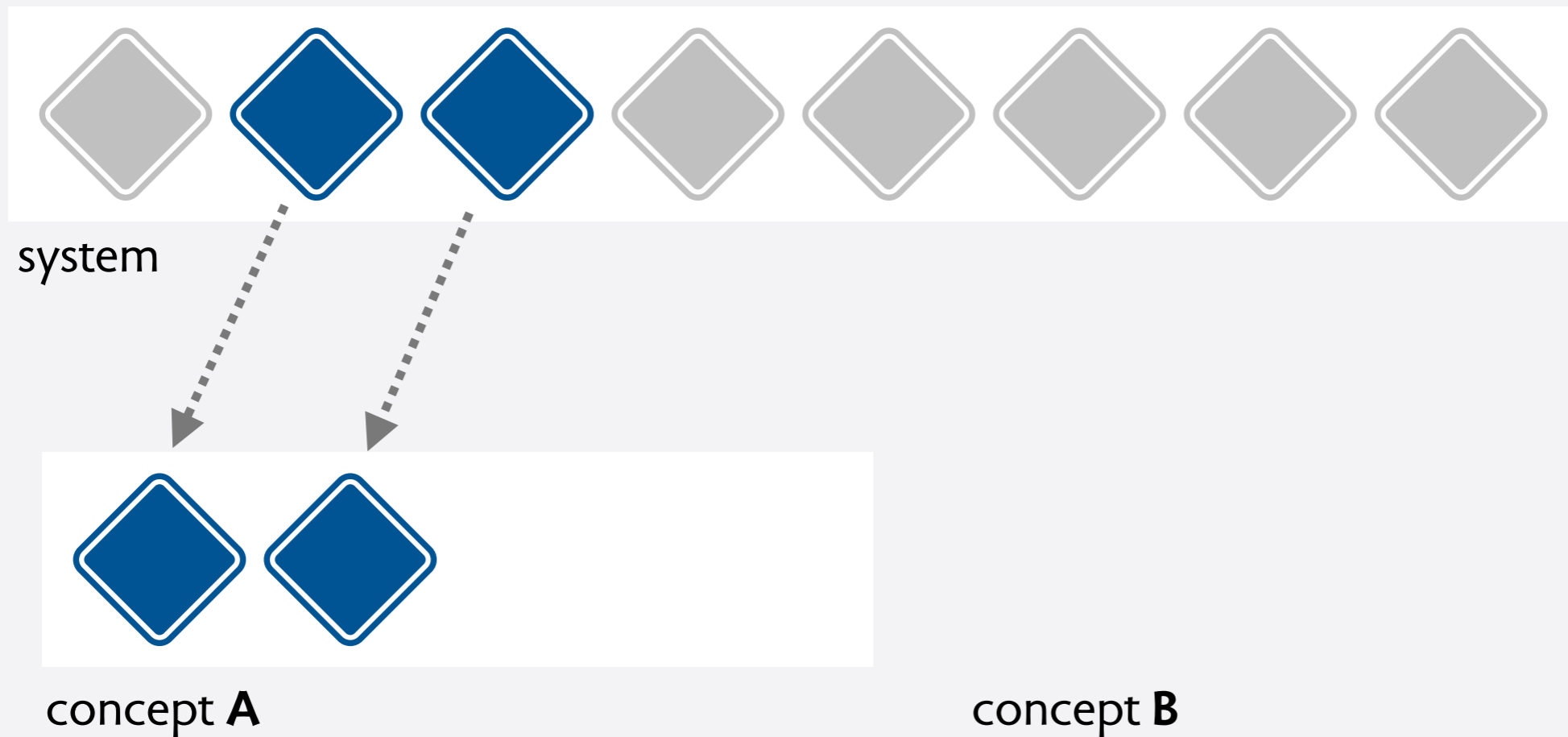
the integrity rule

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

a simple criterion

projected behavior must satisfy concept spec:

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



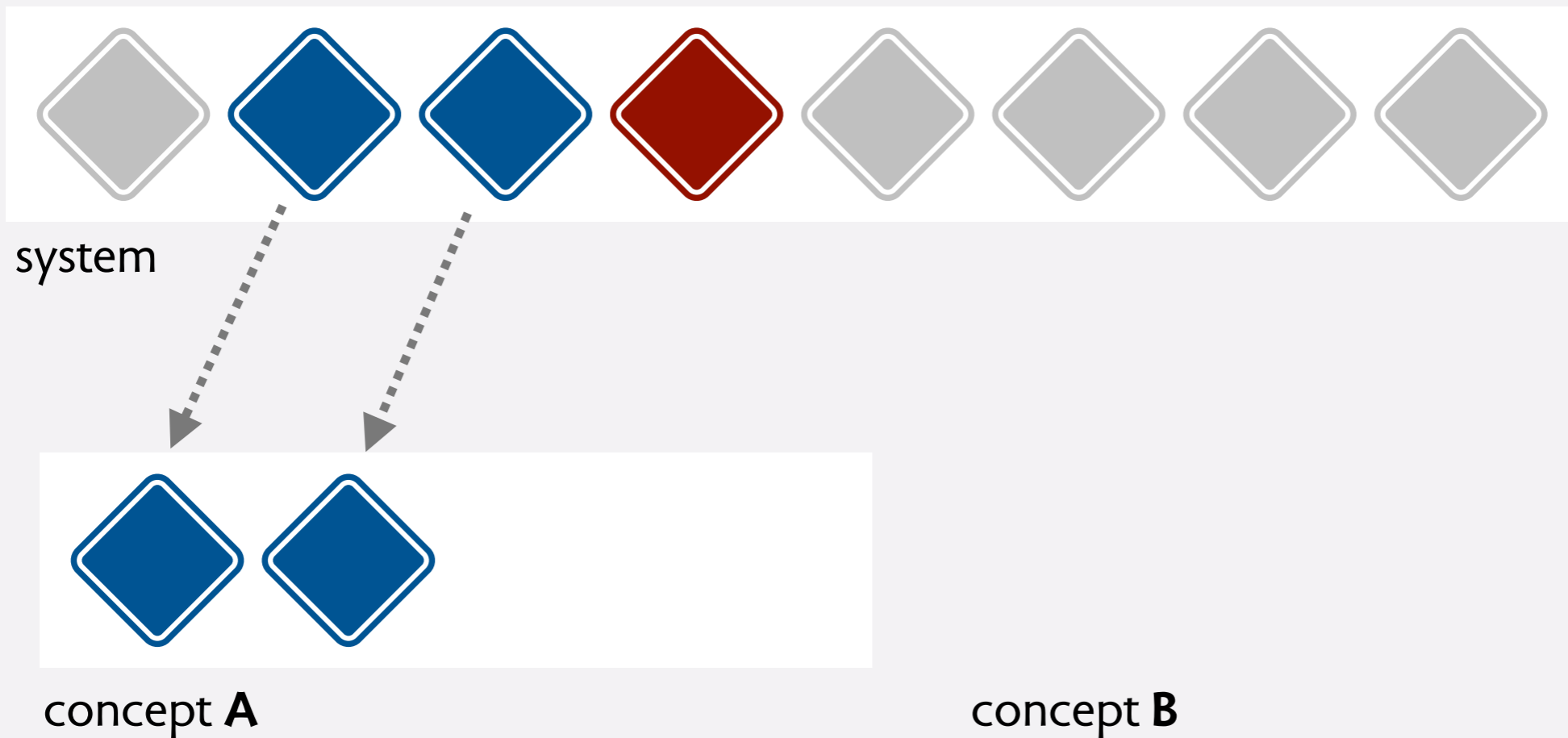
the integrity rule

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

a simple criterion

projected behavior must satisfy concept spec:

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



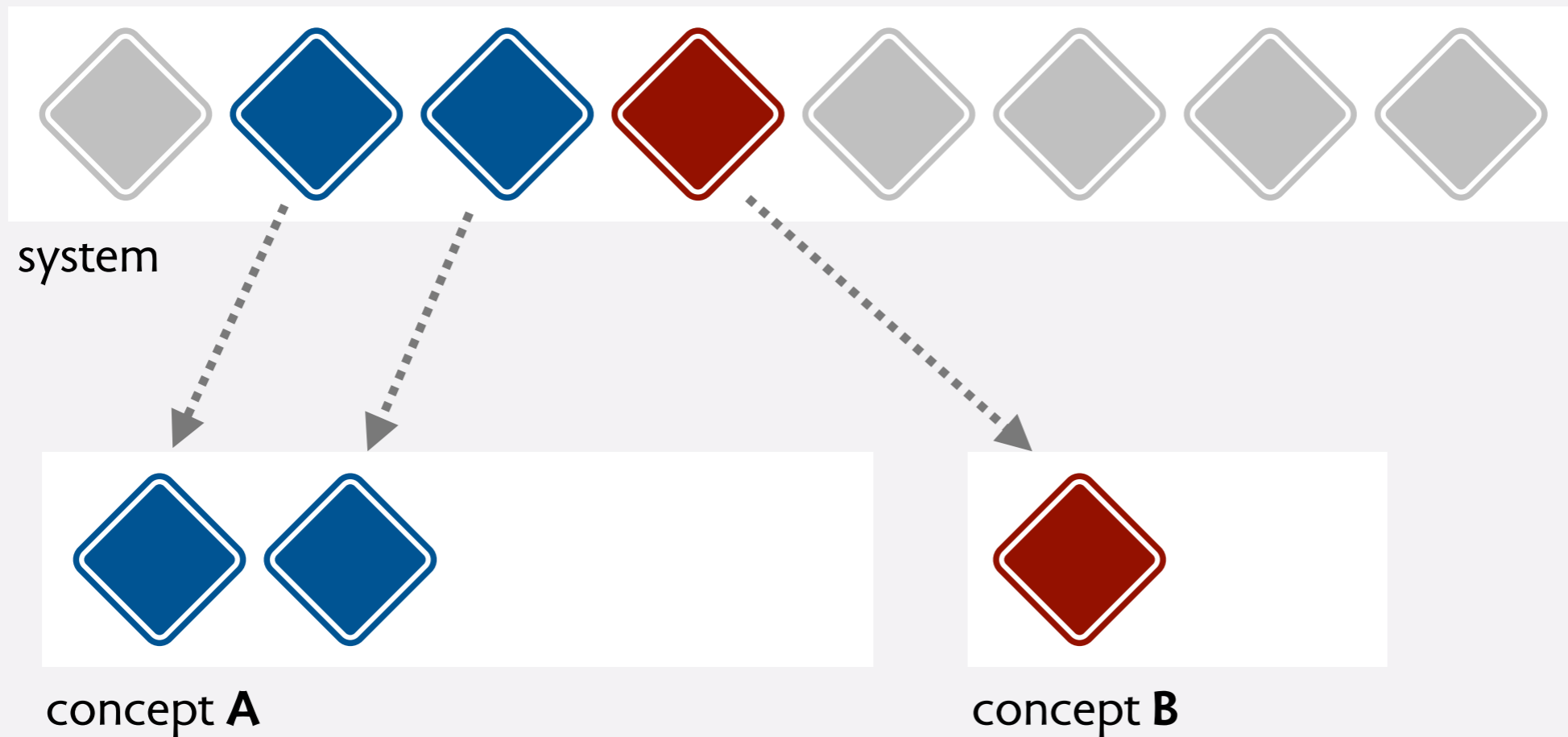
the integrity rule

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

a simple criterion

projected behavior must satisfy concept spec:

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



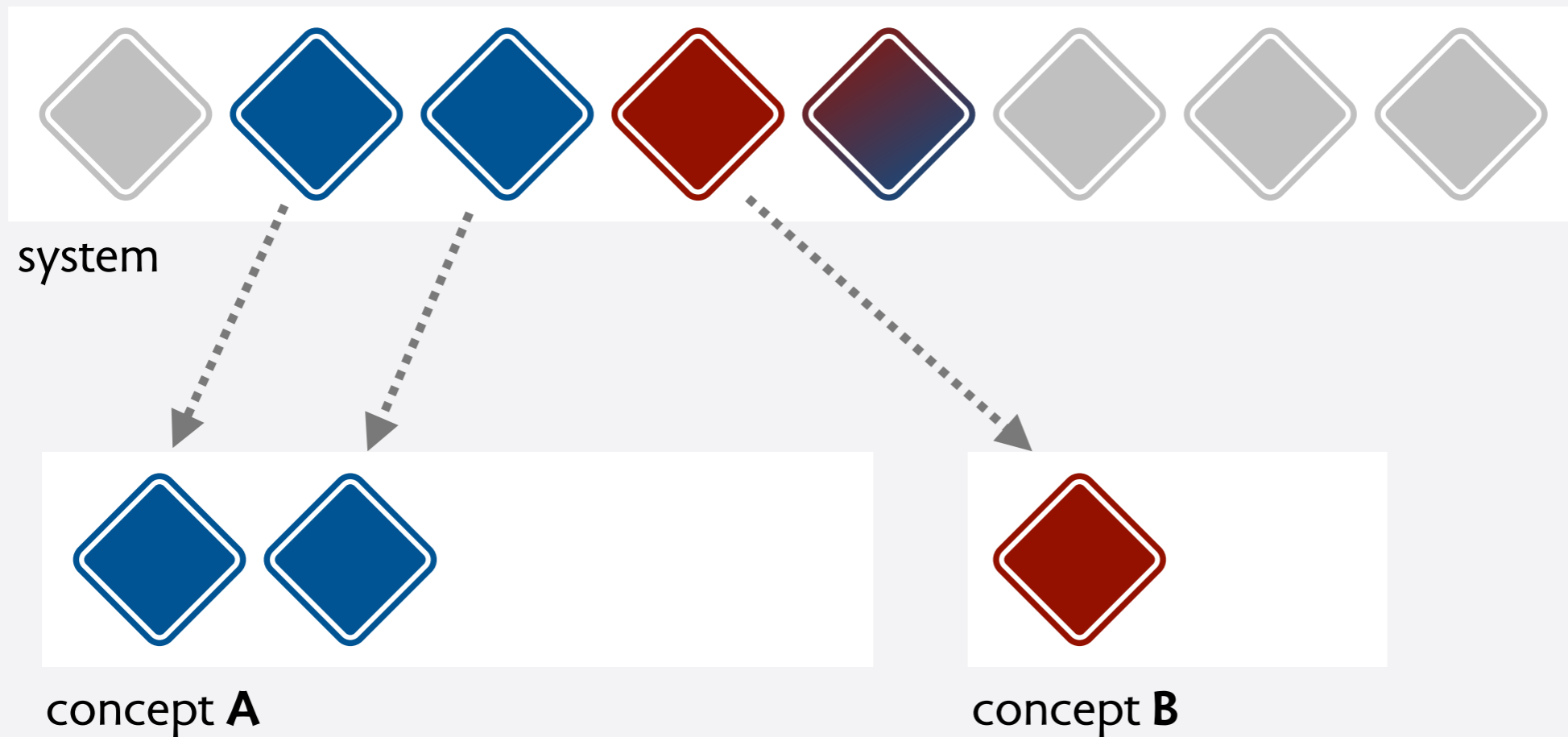
the integrity rule

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

a simple criterion

projected behavior must satisfy concept spec:

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



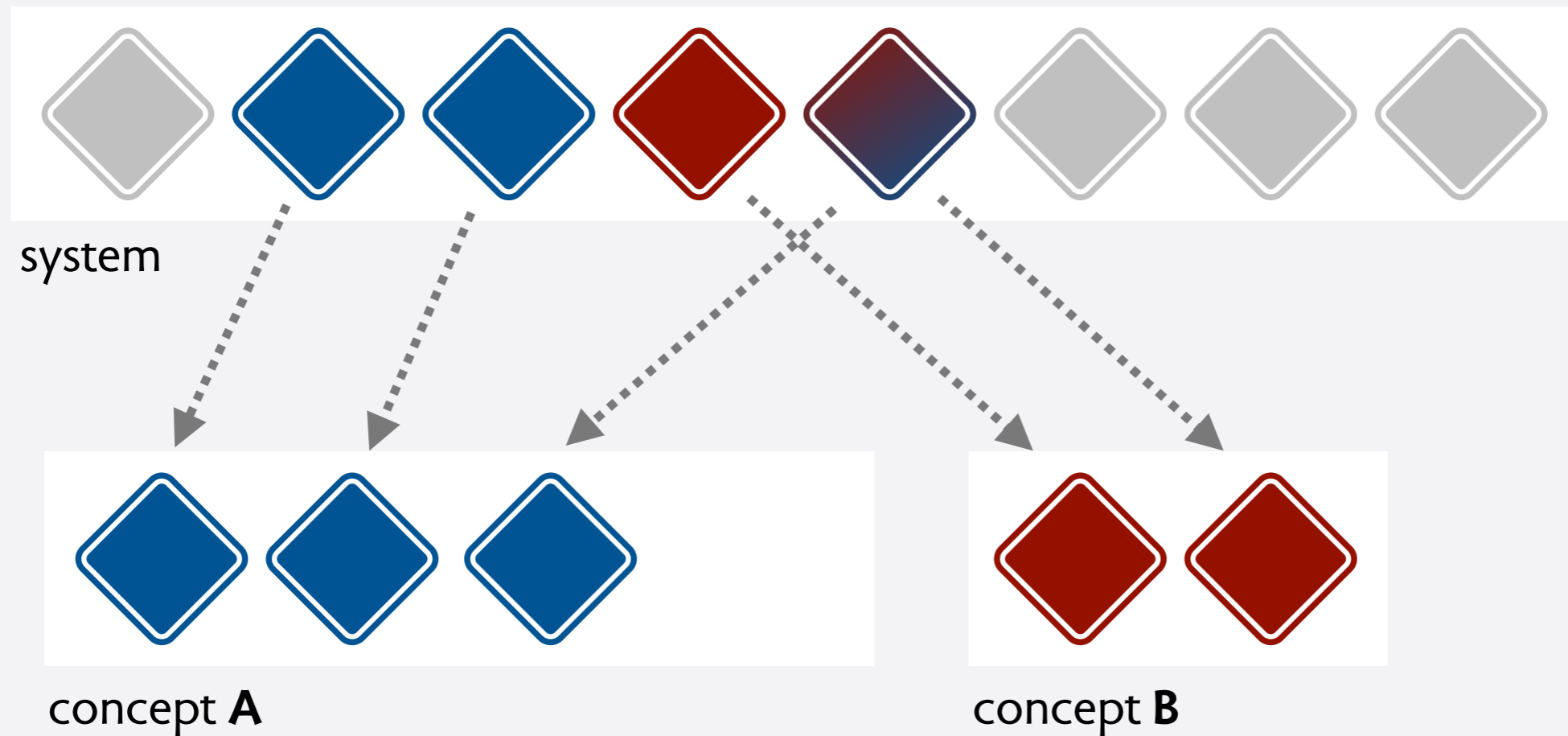
the integrity rule

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

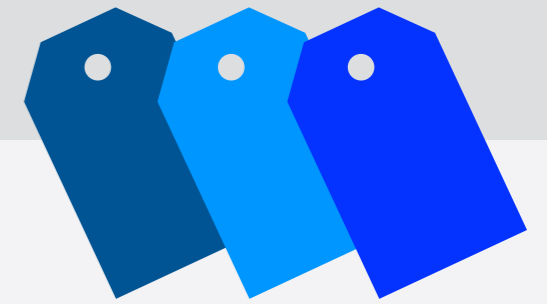
a simple criterion

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



purpose

organize items for
easy retrieval

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

actions

mark ($x: X, p: \text{Label}$) \triangleq labels' = labels + $x \rightarrow p$

unmark ($x: X, p: \text{Label}$) \triangleq labels' = labels - $x \rightarrow p$

spec

find ($ps: \mathbf{set} \text{Label}$): $\mathbf{set} X \triangleq \mathbf{result} = \{x \mid ps \mathbf{in} x.\text{labels}\}$

operational principle

mark(x,p) then find(p): $xs \Rightarrow x \mathbf{in} xs$

if no mark(x,P) and then find(p): $xs \Rightarrow x \mathbf{!in} xs$

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

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

The screenshot shows a Gmail inbox with the search filter 'in:sent' applied. The interface includes a search bar at the top, a toolbar with icons for back, forward, archive, report spam, delete, mute, and flag, and a '1 of 1' indicator. The inbox contains four messages from Alyssa P. Hacker and Ben Bitdiddle. Two yellow callout boxes are overlaid on the messages, providing information about label events:

- when message m is sent**
`Label.mark(m, 'sent')`
occurs implicitly
- when Sent link is clicked**
`Label.find('sent'):ms`
occurs

The messages in the inbox are:

- javascript** (Inbox x, hacking x, meetups x) from Alyssa P. Hacker to me: "Reminds you of the old days, eh?" (Tue, May 8, 9:14 PM)
- from Ben Bitdiddle to Alyssa: "Yes, it does." (Tue, May 8, 9:40 PM)
- from Alyssa P. Hacker: "JavaScript makes me feel nostalgic for Scheme." (Mon, Jul 30, 1:24 PM)
- from Ben Bitdiddle 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. The email content is a conversation about JavaScript. Three yellow callout boxes are overlaid on the messages, providing technical details about the 'sent' label. The first callout points to the first message from Alyssa P. Hacker. The second callout points to the second message from Ben Bitdiddle. The third callout points to the fourth message from Ben Bitdiddle.

Search: 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?

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

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

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

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 c

*but ms includes
messages never marked*

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

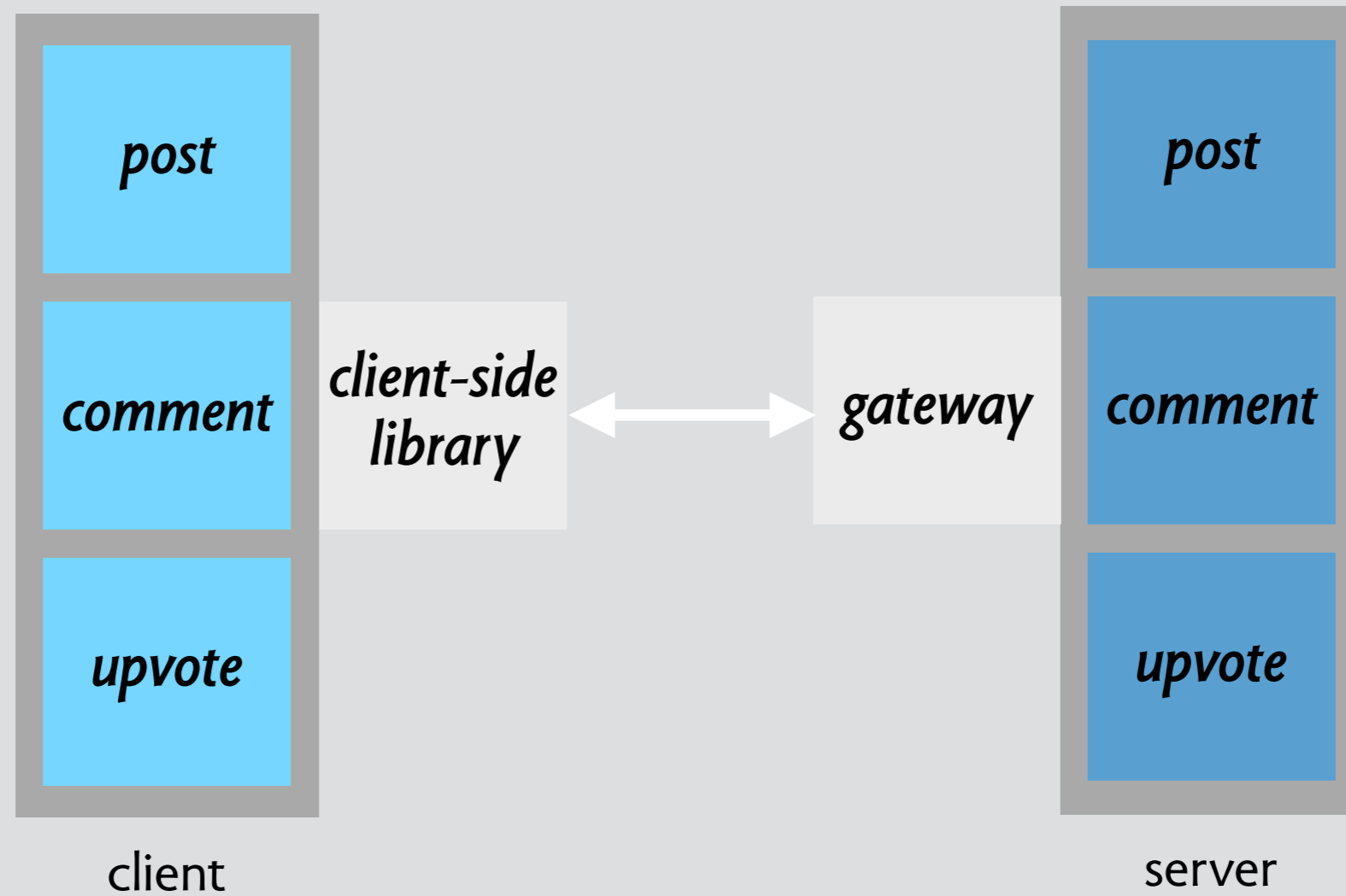


idea: concept cliches
full stack implementation

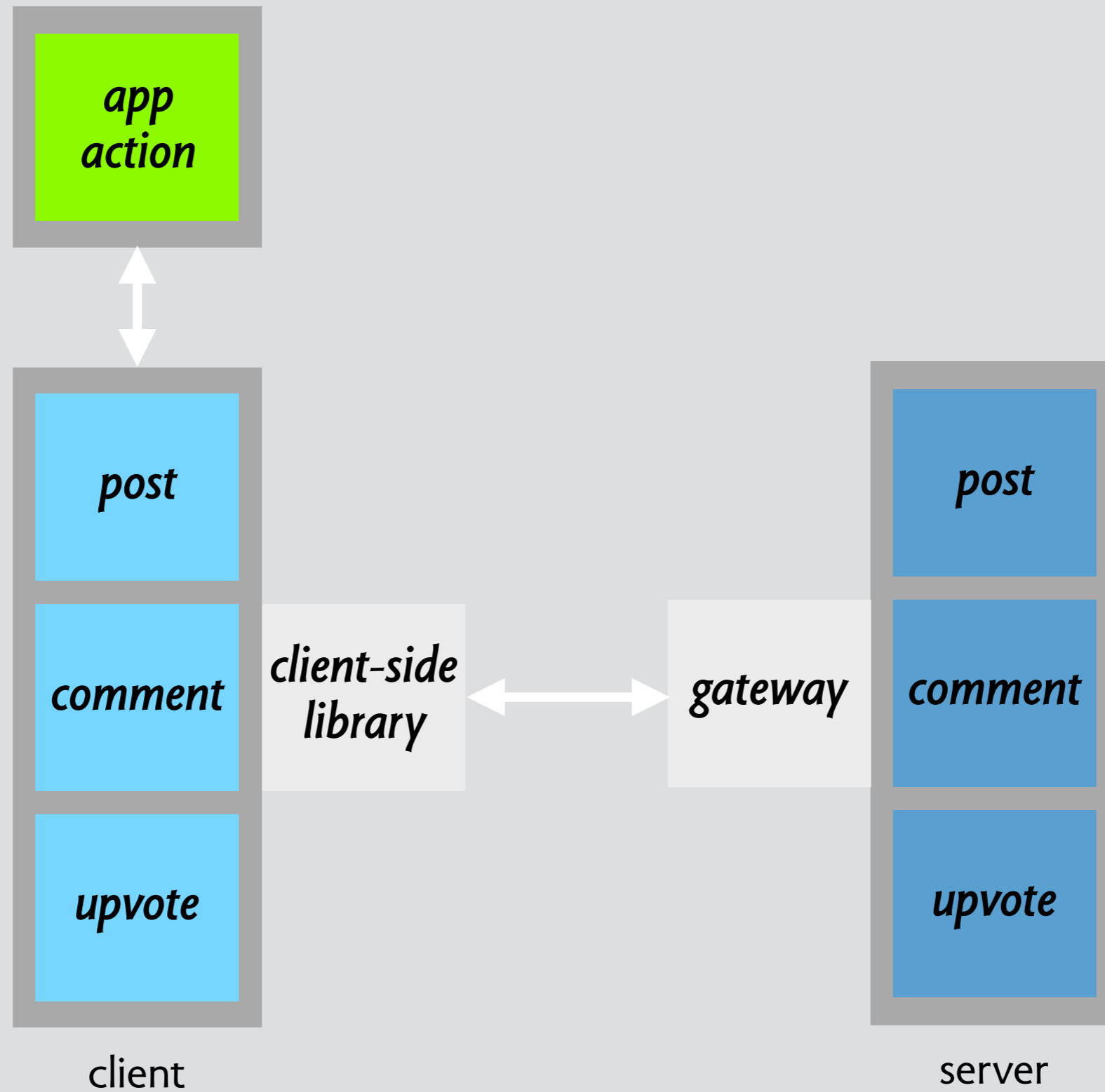
app-specific assembly
in HTML, no JS or backend code

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](#)

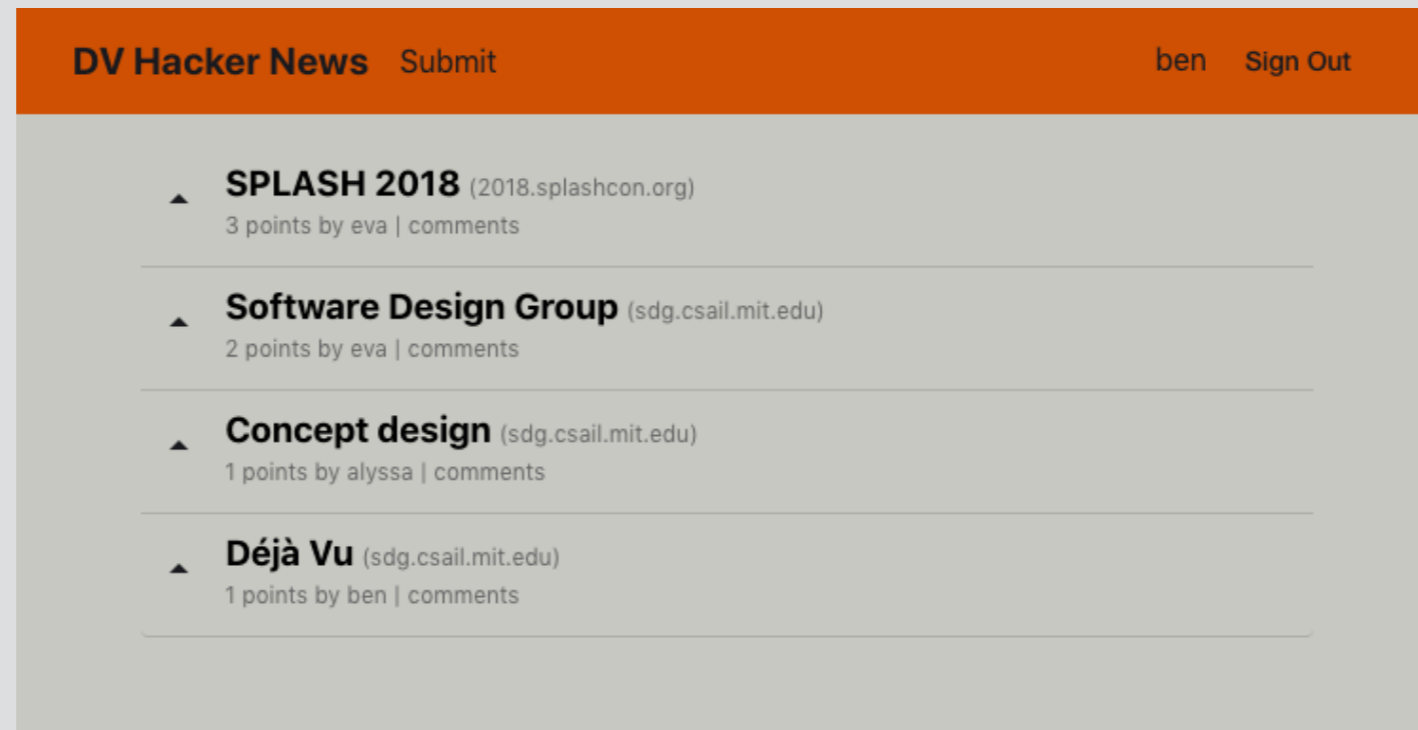
▲ **SPLASH 2018** (2018.splashcon.org)
3 points by eva | [comments](#)

▲ **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 interface. 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.

```
<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>
```

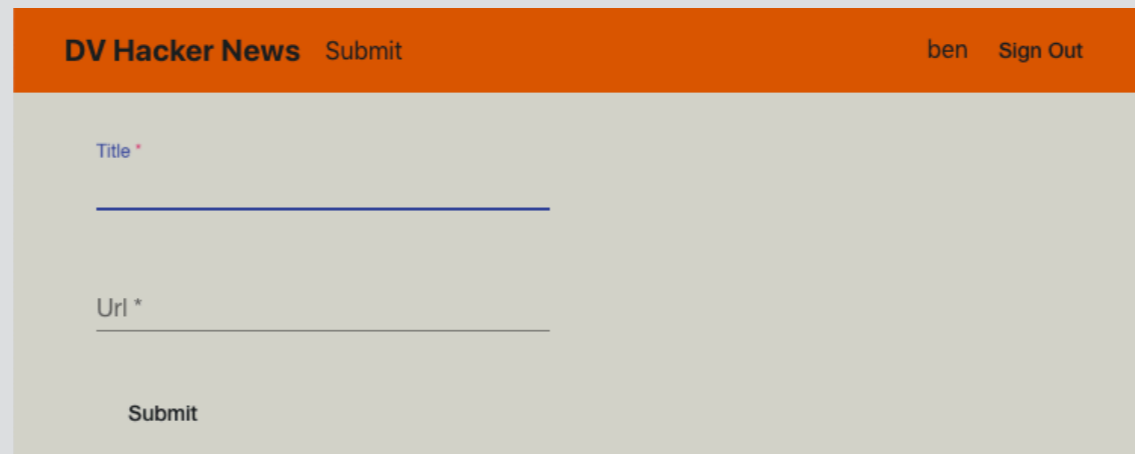

submit post action

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

Title *

Url *

submit post action



DV Hacker News Submit ben Sign Out

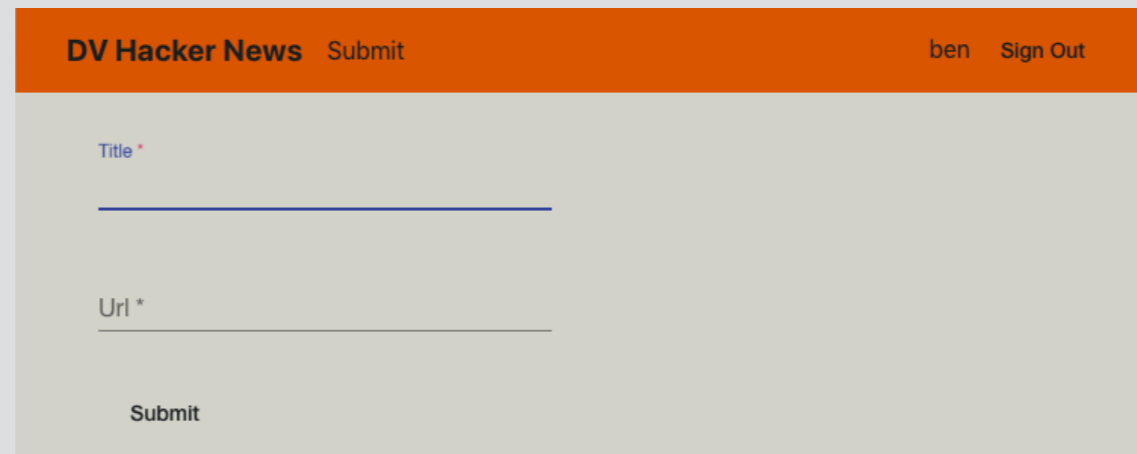
Title *

Url *

Submit

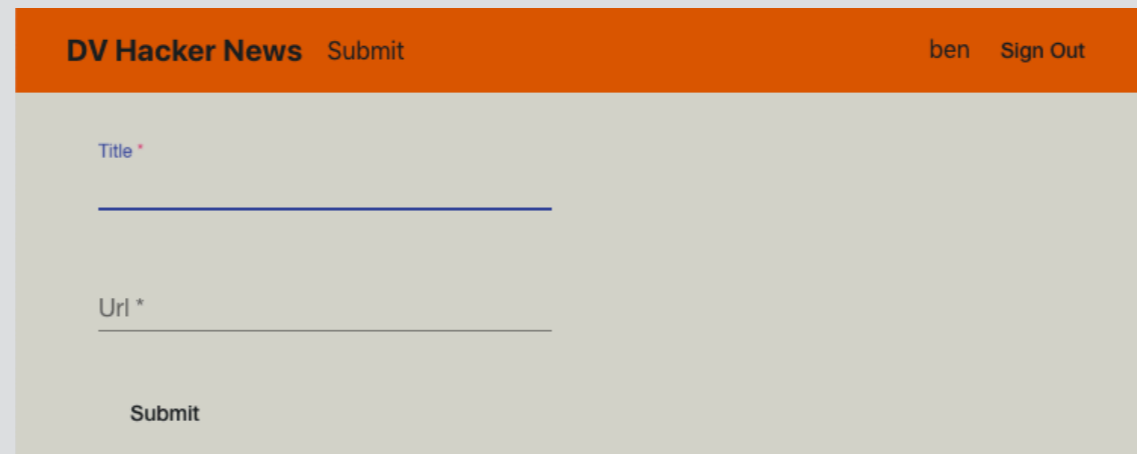
```
<dv.action name="submit-post">  
<hackernews.navbar />  
<div class="main"> <dv.tx>  
  <dv.gen-id />  
  <property.create-object  
    id=dv.gen-id.id  
    initialValue={ author: hackernews.navbar.user.username }  
    showExclude=["author"]  
    buttonLabel="submit"  
    newObjectSavedText="Post 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

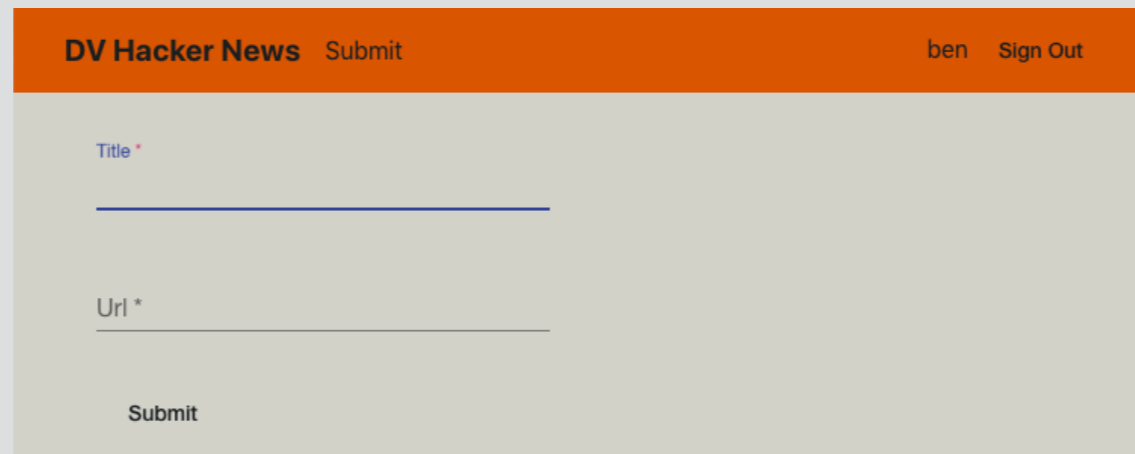


```
<dv.action name="submit-post">  
<hackernews.navbar />  
<div class="main"> <dv.tx>  
  <dv.gen-id />  
  <property.create-object  
    id=dv.gen-id.id  
    initialValue={ author: hackernews.navbar.user.username }  
    showExclude=["author"]  
    buttonLabel="submit"  
    newObjectSavedText="Post 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.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>
```

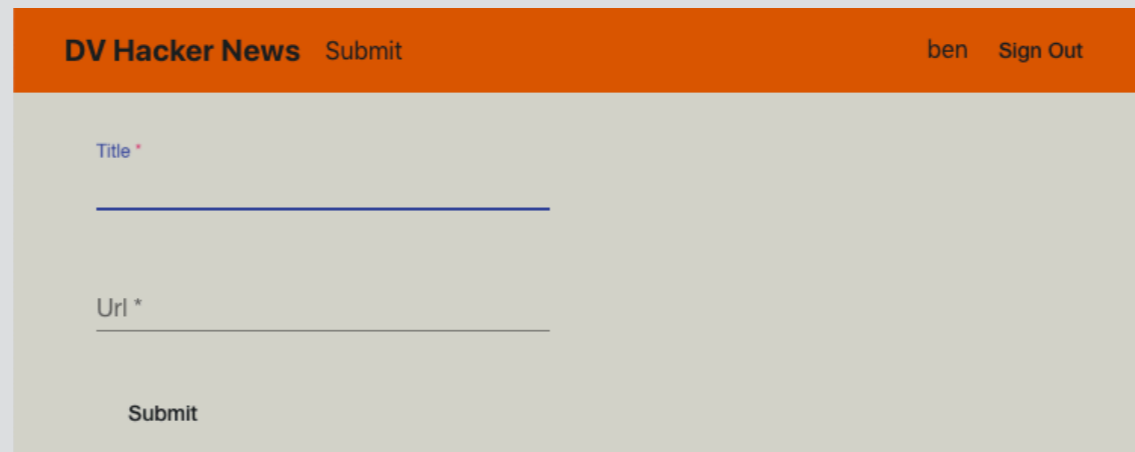
generate id

transaction

id used

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

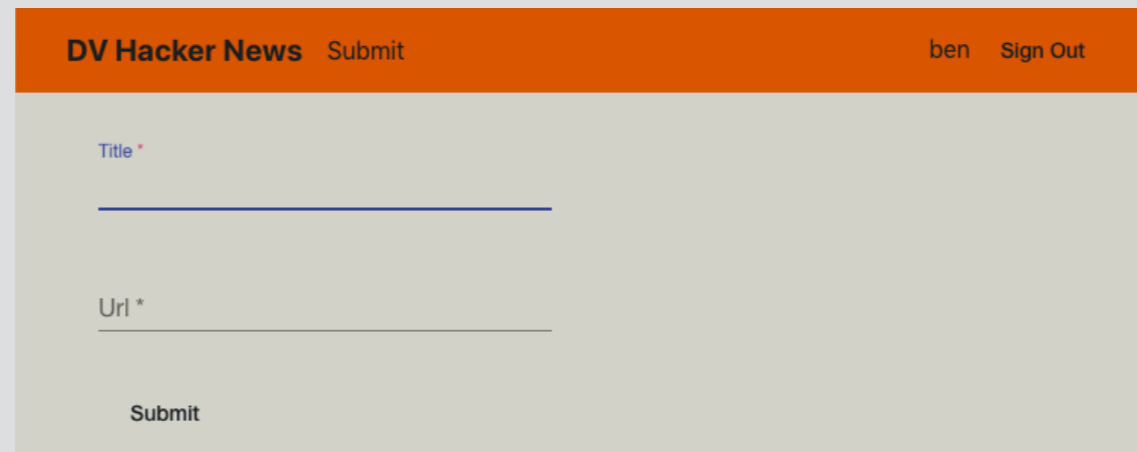
transaction

id used

set param

id used

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

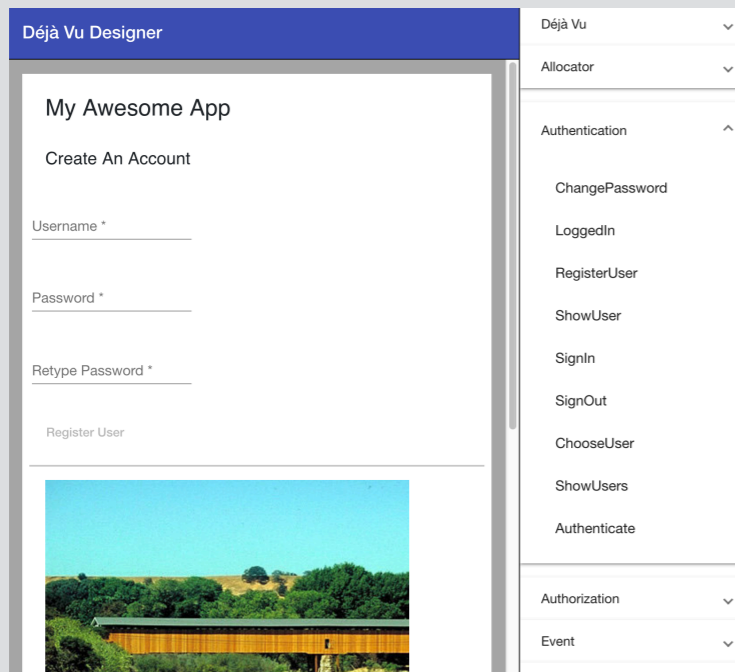
id used

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

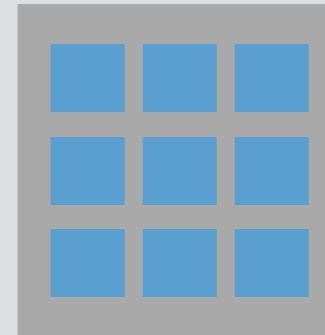
deja vu plans



WYSIWYG designer
(Barry McNamara)



security: stop
request forgeries



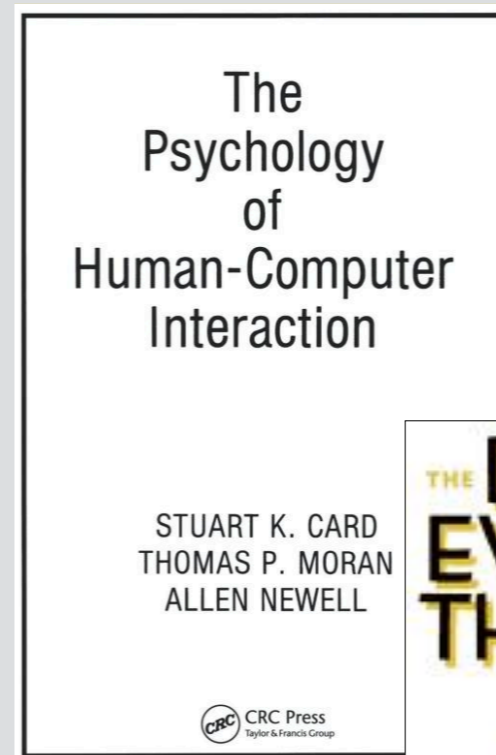
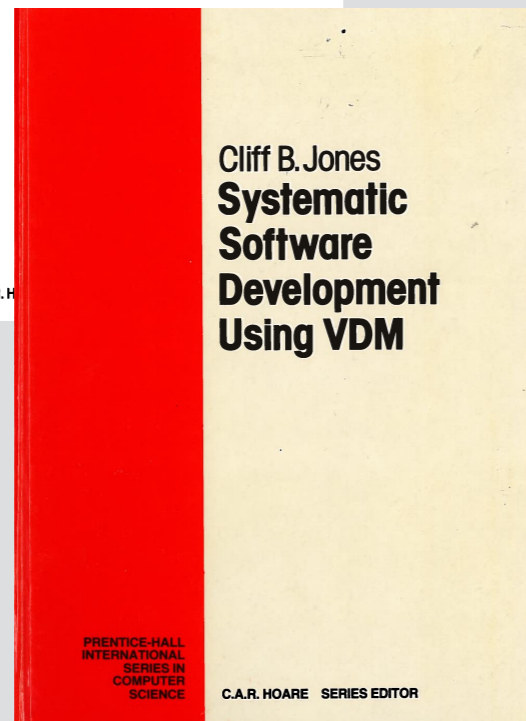
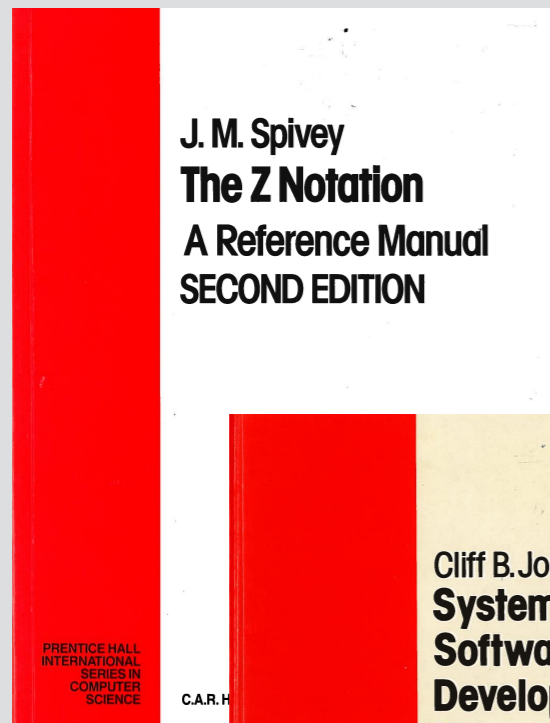
cliche library
for social apps
(Maryam Archie)



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

putting the “thinking” into “design thinking”



theory of usability: collections of heuristics
design thinking: focused on empirical evaluation
see Natasha Jen, *Design Thinking is Bullsh*t*

patterns

128 INDOOR SUNLIGHT*



614

... according to SOUTH FACING OUTDOORS (105), the building is placed in such a way as to allow the sun to shine directly into it, across its gardens. From INTIMACY GRADIENT (127), you have some idea of the overall distribution of public and private rooms within the building. This pattern marks those rooms and areas along the intimacy gradient which need the sunlight most, and helps to place them so that the indoor sunlight can be made to coincide with the rooms in the INTIMACY GRADIENT which are most used.

❖ ❖ ❖

If the right rooms are facing south, a house is bright and sunny and cheerful; if the wrong rooms are facing south, the house is dark and gloomy.

Everyone knows this. But people may forget about it, and get confused by other considerations. The fact is that very few things have so much effect on the feeling inside a room as the sun shining into it. If you want to be sure that your house, or building, and the rooms in it are wonderful, comfortable places, give this pattern its due. Treat it seriously; cling to it tenaciously; insist upon it. Think of the rooms you know which do have sunshine in them, and compare them with the many rooms you know that don't.

From the pattern SOUTH FACING OUTDOORS (105), the building gets an orientation toward the south. Now the issue is the particular arrangement of rooms along this south edge. Here are some examples: (1) a porch that gets the evening sun late in the day; (2) a breakfast nook that looks directly into a garden which is sunny in the morning; (3) a bathing room arranged to get full morning sun; (4) a workshop that gets full southern exposure during the middle of the day; (5) an edge of a living room where the sun falls on an outside wall and warms a flowering plant.

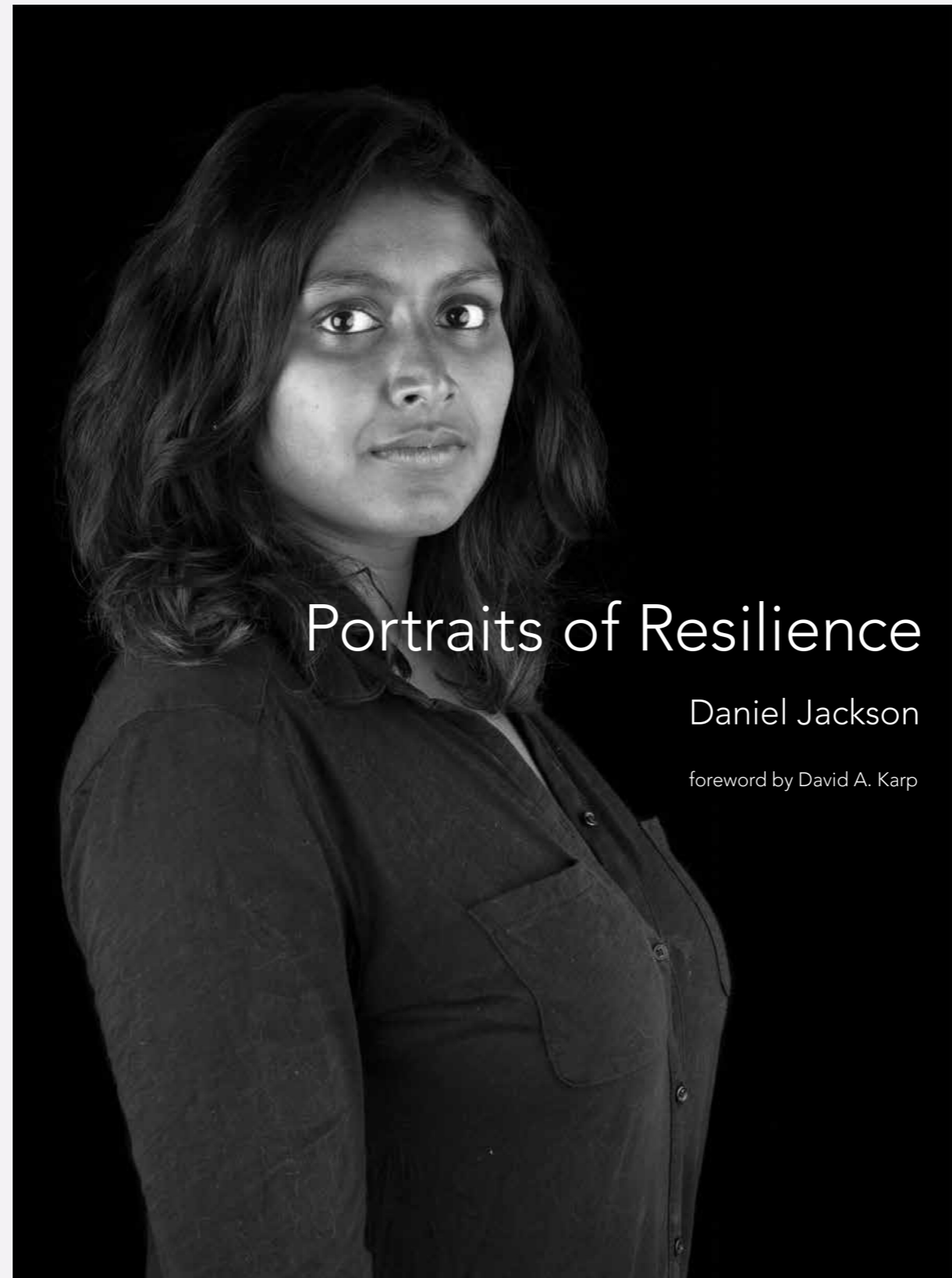
The key diagram for this pattern summarizes the relations between parts of the house and the morning, the afternoon and

615

concepts: more like Alexander's patterns?

<https://tinyurl.com/conceptproject>
to be notified about release of book & platform

a different project



<http://portraitsofresilience.com>