finding structure in software
we use structure to understand artifacts
villa designs (andrea palladio, c. 1570)
American Federal Government (1864)
The periodic table (Mendeleev, c. 1870)

<table>
<thead>
<tr>
<th>Row</th>
<th>Elements</th>
<th>Periodic Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group I</td>
<td>Li, Be, Na, K, Rb, Cs</td>
<td></td>
</tr>
<tr>
<td>Group II</td>
<td>Sc, Ti, V, Cr, Mn, Fe</td>
<td></td>
</tr>
<tr>
<td>Group III</td>
<td>Co, Ni, Cu, Zn</td>
<td></td>
</tr>
<tr>
<td>Group IV</td>
<td>Zr, Nb, Mo, Ru, Rh, Pd</td>
<td></td>
</tr>
<tr>
<td>Group V</td>
<td>Hf, Ta, W, Re, Os, Ir, Pt</td>
<td></td>
</tr>
<tr>
<td>Group VI</td>
<td>Tb, Dy, Ho, Er, Tm, Yb, Lu</td>
<td></td>
</tr>
<tr>
<td>Group VII</td>
<td>Th, Pa, U, Np, Pu, Am, Cm</td>
<td></td>
</tr>
<tr>
<td>Group VIII</td>
<td>Fr, Ra, Ac, Cm, Bk, Cf, Es</td>
<td></td>
</tr>
</tbody>
</table>

* Periodic Table of the Elements*
London Underground (Harry Beck, 1933)
experiential
structure helps you understand how it behaves not how it’s built

modular
components of the structure can be understood independently

abstract
internal workings and structure are not shown
what are the elements of software?
Jackson structured programming (wikipedia.org)

You might find helpful an annotated version [0] of Hoare's explanation of JSP that I edited for a Michael Jackson festschrift... 

I'd point to these ideas as worth knowing:

- The archetypical problems that, however you code, can't be pushed under the rug—most notably structure clashes—and just recognizing them helps.

- Coroutines (or code transformation) let you structure code more cleanly when you need to read or write more than one structure. It's why real iterators (with yield), which offer a limited form of this, are (in my view) better than Java-style iterators with a next method.

- The idea of viewing a system as a collection of asynchronous processes (Ch. 11 in the JSP book, which later became JSD) with a long-running process for each real-world entity. This was a notable contrast to OOP, and led to a strategy (seeing a resurgence with event storming for DDD) that began with events rather than objects.

[0] https://groups.csail.mit.edu/sdg/pubs/2009/hoare-jsp-3-29-09...

... this brings back memories! In the late eighties I, as a teenager, found a Jackson Struct. Pr. book at the town library. I remember I was amazed at the text and wondered why I hadn't heard about the method before.

If I remember correctly did the book clearly point out backtrack as the standard method, while mentioning that most languages lacked that, so it had to be implemented manually.

This is referenced(1) as a core inspiration in the preface to “How to Design Programs” but i never researched it further because i've found the “design recipes” approach in htdp to be pretty solid in real life problems.
but what's a concept? three things it isn't

abstract type, class/object

<table>
<thead>
<tr>
<th>Matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>multiply</td>
</tr>
<tr>
<td>add</td>
</tr>
<tr>
<td>invert</td>
</tr>
<tr>
<td>equals</td>
</tr>
</tbody>
</table>

not limited to built-in types
encapsulate representation
defined by operations alone

concept lattice

what operations can you do on an upvote?

upvotes and downvotes
are votes and then what?

entity in data model

but concept is in the relationships, not the entities!
a concept has a name

the same concept in HackerNews, NYTimes comment section, StackOverflow, etc

Reader Picks  All

John
Boston | Oct. 27
To protect children? Seems far more likely it's yet one more way to extract personal information to feed the insatiable advertising machines.

1 Reply  143 Recommend  Share
a concept has a purpose

**concept** Upvote

**purpose** rank items by popularity

**concept** Reaction

**purpose** send reactions to author

**concept** Recommendation

**purpose** use prior likes to recommend

---

This is homework and I'm having a hard time with it.

are the definitions of the objects:

```latex
sig Library {
  patrons : set Person,
  on_shelves : set Book,
}
```

---

Daniel: I think we should organize a software concepts forum.

👍 1 😊
A concept has a state like bounded context in DDD, but even more localized. Include in state only what’s needed for the concept’s own computations.

**Concept**: Upvote

**Purpose**: Rank items by popularity

**State**
- Votes: User -> set Vote
- For: Vote -> one Item
- Upvote, Downvote: set Vote
- Rank: Item -> one Int

Track users to prevent duplicate voting.

Like bounded context in DDD, but even more localized.
A concept has actions

**concept** Upvote

**purpose** rank items by popularity

**state**
votes: User -> set Vote
for: Vote -> one Item
Upvote, Downvote: set Vote
rank: Item -> one Int

**actions**
upvote (u: User, i: Item)
downvote (u: User, i: Item)
unvote (u: User, i: Item)

**actions capture the concept behavior in full**

downvote (i: Item, u: User)
// no existing Downvote for i in u.votes
// remove any Upvote for i from u.votes
// add a Downvote for i in u.votes
// update i.rank ...

succinct specification
as actions on states
VDM (1986)
Z (1992)
Larch (1993)
Alloy (2006)
a concept catalog entry

**concept Upvote**

**design variants**
donvote as unvote
use age in ranking
weigh downvotes more

**related concepts**
Recommendation, Reaction, ...

**typical uses**
social media posts
comments on articles
Q&A responses

**often used with**
Karma, Session, ...

**known issues**
preventing double votes
(require login, use IP address, save cookie)
saving storage space
(freeze old posts and from user info)
how to compose concepts?
how to extend behavior?

**concept** Upvote

**actions**
upvote (u: User, i: Item)
downvote (u: User, i: Item)
unvote (u: User, i: Item)

**suppose I want this behavior:**
you can’t downvote an item until you’ve received N upvotes on your own items

**define a new concept!**
a hint: not just used by Upvote

**concept** Karma

**purpose** privilege good users

**state**
karma: User -> one Int
contribs: User -> set Item

**actions**
contribute (u: User, i: Item)
reward (u: User, r: Int)
permit (u: User, r: Int)
**Concept** Upvote

**Actions**
- upvote (u: User, i: Item)
- downvote (u: User, i: Item)
- unvote (u: User, i: Item)

**When** upvote (u, i) and i in u'.contribs **Also** reward (u', 10)

**Concept** Karma

**Actions**
- contribute (u: User, i: Item)
- reward (i: Item, r: Int)
- permit (u: User, r: Int)

**When** downvote (u, i) **Also** permit (u, 20)
**Concept: Upvote**

- upvote (Bob, post1)
- upvote (Carol, post1)
- downvote (Alice, post2)

**When** upvote (u, i)
**Also** reward (u, 10)

**Concept: Karma**

- contrib (Alice, post1)
- contrib (Bob, post2)
- reward (Alice, 10)
- reward (Alice, 10)
- permit (Alice, 20)

**Composition uses event sync from Hoare’s CSP**

**No concept coupling concepts preserve properties**
so what can you do with concepts?
characterize apps and families
characterize apps & families

text editor
- line
- character set
- markup

word processor
- paragraph
- format
- style

desktop publishing app
- paragraph
- format
- style
- page
- textflow

waffling about concepts and fine points of design
explore & evaluate individual concepts

how many users believe the folder concept works

how folders actually work (in Dropbox, Unix, Multics)
analyze how concepts fit together

concept Upvote
purpose rank items by popularity
actions
   upvote (u: User, i: Item)
   ...

concept Reaction
purpose convey emotion to author
actions
   reactAngry (u: User, i: Item)
   ...

unwanted sync?
design moves mechanical analogs
three pairs of design moves

merge - split

unify - specialize

loosen - tighten
split-merge: tradeoff simplicity/flexibility

photocopier

printer + scanner

emergency flashlight

flashlight + battery + charger
**unify-specialize**: tradeoff simplicity/specificity

- **set of wrenches** → **unify** → **adjustable wrench**
- **macro lens** → **specialize** → **general-purpose lens**
tighten-loosen: tradeoff automation/flexibility

- light pull / door lock
- airplane toilet lock
- dimmers with separate controls
- rotary dimmer switch
successful
design moves
in software
split: emergence of a concept in Keynote

- Full screen toggle emerges as partial concept (c. 2010?)
- Play-in-window option now an independent concept (2021)
unify: subsuming access control in MIT’s Moira

can toggle mailing list attribute

can create admin list with no login users!
tighten: label and trash concepts in Gmail

- Label
- Trash

Gmail interface showing inbox with options:
- Starred
- Snoozed
- Sent
- Drafts
- Trash

Show messages with label hacking

Also implemented as a label
<table>
<thead>
<tr>
<th>purpose</th>
<th>undo deletion</th>
</tr>
</thead>
<tbody>
<tr>
<td>concept</td>
<td>trash</td>
</tr>
<tr>
<td>structure</td>
<td>trash: set Item</td>
</tr>
<tr>
<td>actions</td>
<td>delete (i: Item)</td>
</tr>
<tr>
<td></td>
<td>restore (i: Item)</td>
</tr>
<tr>
<td></td>
<td>empty ()</td>
</tr>
</tbody>
</table>

**when delete (i)**
**also add (i, ‘trash’)**

<table>
<thead>
<tr>
<th>purpose</th>
<th>organize with overlapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>concept</td>
<td>label</td>
</tr>
<tr>
<td>structure</td>
<td>labels: Item -&gt; set Label</td>
</tr>
<tr>
<td>actions</td>
<td>add (i: Item, l: Label)</td>
</tr>
<tr>
<td></td>
<td>remove (i: Item, l: Label)</td>
</tr>
<tr>
<td></td>
<td>find (ls: set Label, out is: set Item)</td>
</tr>
</tbody>
</table>
integrating these concepts is tricky

- click on trash
- filter on todo label
- filter on todo and trash
- filter on something else
a beautiful (but tricky) synergy

Folder

New in Lion (2011)

Sortable by volume!

Trash

Tighten

Folder

Trash
design moves in response to problems
aspect ratio in Fujifilm cameras
a lovely camera fuji x100
complex menu system: image quality setting
aspect ratio
image size setting
non-standard ratio + raw?
problem #1: no non-standard ratio unless also save JPG!

raw image showing non-destructive aspect ratio crop
problem #2: very few ratio options

Fuji, give us 4:3, 5:4, and 6:7 aspect ratios on X-series cameras

636 have signed. Let’s get to 1,000!

At 1,000 signatures, this petition is more likely to be featured in recommendations!
aspect ratio is not a concept merged into JPEG image size concept so cannot be controlled independently
I call this “overloading by piggybacking”

**solution: split concepts**
would allow ratio change to raws without JPEGs
would avoid combinatoric explosion of options
message filters in apple mail
How many ways to filter messages?

- Search options
  - From
  - To
  - Subject
  - Message is addressed to my full name
  - Message is not addressed to my full name
  - Date sent
  - Date received
  - Account
    - Sender is in my contacts
    - Sender is not in my contacts
    - Sender is VIP
    - Sender is member of group
    - Sender is not a member of group
  - Message content
    - Message is junk mail
    - Message is signed
    - Message is encrypted
    - Priority is high
    - Priority is low
    - Priority is normal
    - Date last viewed
    - Message has flag
    - Message was replied to
    - Message was not replied to
    - Message is in mailbox
    - Message is not in mailbox
    - Contains attachments
    - Any attachment name
    - Attachment type
    - Every Message
    - Edit header list...

- Rule options
  - From
  - To
  - Subject
  - Any recipient
  - Message is addressed to my full name
  - Message is not addressed to my full name
  - Date sent
  - Date received
  - Account
    - Sender is in my contacts
    - Sender is not in my contacts
    - Sender is VIP
    - Sender is member of group
    - Sender is not a member of group
  - Message content
    - Message is junk mail
    - Message is signed
    - Message is encrypted
    - Priority is high
    - Priority is low
    - Priority is normal
    - Date last viewed
    - Message has flag
    - Message was replied to
    - Message was not replied to
    - Message is in mailbox
    - Message is not in mailbox
    - Contains attachments
    - Any attachment name
    - Attachment type
    - Every Message
    - Edit header list...

- Search for a message
  - 

- Create a rule
  - 

- Define a smart folder
  - 

- Smart folder options
  - Include messages from Trash
  - Include messages from Sent
search, rule and smart folder
all include their own specialized concepts
incomparable features, different UIs

unify in a single message filter concept
include “create folder from search”, eg
sticky hands
in zoom
event deletion in google calendar
Despite some erroneous messages sent to this list accidentally, Kanit's talk is happening! Please join us on Monday.
Canceling and deleting events in the Google Calendar mobile app is similar to on a desktop.

1. First, open Google Calendar.
2. Tap on the event you wish to cancel.
3. Press on the three dots in the top right corner of the event window.
4. Select Delete.
5. Tap Delete event. Google Calendar will send a cancellation email to the guests.

Mar 22, 2021

https://wpamelia.com › Blog

How to Cancel an Event in Google Calendar - Amelia booking ...
a long time problem in iCal too
how to delete spam calendar events?
concept calendar
purpose record engagements
actions
  createEvent (...): Event
  deleteEvent (e: Event)
...

concept invitation
purpose coordinate events
actions
  accept (e: Event)
  decline (e: Event)
...

unwanted sync!
apple’s solution

Are you sure you want to delete this event?
Deleting this event will notify the organizer that you’re declining the event and deleting it from your calendar. You can’t undo this action.

- Cancel
- Delete and Don’t Notify
- Delete and Notify

resolution to design problem
make sync optional

loosen
takeaways

structure your software design with concepts
inventory the concepts, identify the critical ones
see if you can describe them fully independently
then formulate interactions as synchronizations

apply design moves to explore new options
never a panacea, always a tradeoff

software concepts as patterns
only hinted at this, but equally important
don’t reinvent the wheel!
express your design as sync of familiar concepts?

in formal methods
can concepts help structure & validate models?