

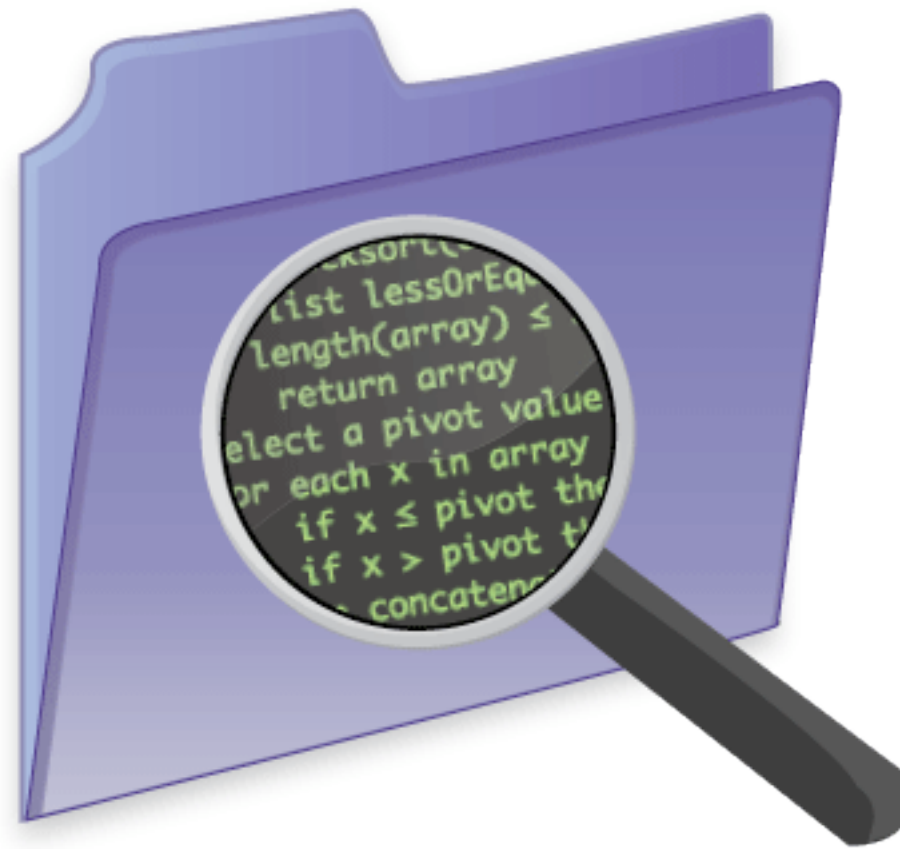
What Counts as Code to Criticize?

Interpreting Visual and Natural Language Programming

Jeremy Douglass

Postdoctoral Researcher
Software Studies Initiative
University of California San Diego

Digital Humanities 2009 - U. Maryland College Park - 2009-06-24



Software Studies

// CRITICAL CODE STUDIES

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head profile="http://gmpg.org/xfn/1.1">
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>Critical Code Studies</title>
<link rel="stylesheet" href="http://s.themes.wordpress.net/testrun/wp-content/themes/2433/css/all.css"
media="screen"/>
<link rel="shortcut icon" type="image/ico"
href="http://s.themes.wordpress.net/testrun/wp-content/themes/2433/favicon.ico" />
```

[MENU](#)[CATEGORIES](#)[ARCHIVES](#)[RECENT](#)[METHODOLOGY](#)

JOIN CCS

[Join the Critical Code
Studies Blog]

email: mark + c +
marino
[at] g + mail [dot]
com

Login

RECENT COMMENTS

Mark Marino
: Wonderful articles,

CCS: COMING TO DIGITAL HUMANITIES 2009

≡ June 23rd, 2009 | → 0 Comments | ▼ CCS, conferences |

This is just a head up about the next big even in Critical Code Studies and
Software Studies: Wed. panel at Digital Humanities 2009:

Wednesday, June 24th
University of Maryland

9:00-10:30am
Charles Carroll Room
Critical Code and Software Studies
Mark Christopher Marino, Noah Wardrip-Fruin, Jeremy Douglass,
Elizabeth Losh, Stephanie August

GO

ABOUT

Critical Code Studies

Critical Code Studies
is a forum for
resources, discussion,
and demonstrations of
the interpretation of
computer code.

CATEGORIES

Digital Humanities

Critical Code / Software Studies

expanding our concept of
what code we critique & how



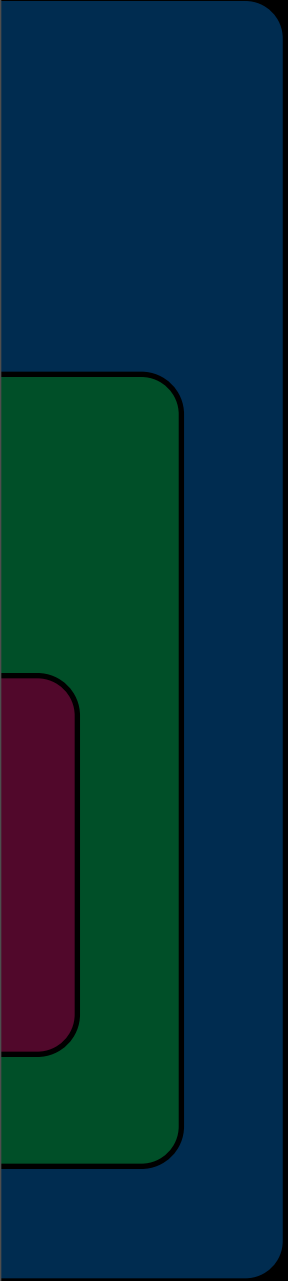
the question:

what does it mean to
study software and
criticize code?



further,

what is possible when
performing critiques of
procedure &
specification?



let's consider a variety of
**programmer-level
representations**

many starting points, but let's begin with a
focus on three paradigms



SOURCE

IMPERATIVE
PROGRAMMING

NATURAL

NATURAL
LANGUAGE
PROGRAMMING

VISUAL

PATCH AND
FLOW CONTROL
PROGRAMMING

SOURCE

IMPERATIVE PROGRAMMING

```
#include <conio.h>
#include <string.h>

#define FULLSCREEN 512
char buf[FULLSCREEN+1];

#include "cold.h"
#include "joy.h"

char * patc="\n\rPress any key to continue

char move(char now){
    char k;
    if(Up[now]==0) Up[now]=now; //if key UP
0 page
    while(k=joyinput()){
        // if (now==0 && k==JOY_UP) return now
        if(k==JOY_A &&A[now]) return A[no
        if(k==JOY_B &&B[now]) return B[no
        if(k==JOY_SEL &&Select[now]) return
        if(k==JOY_STA &&Start[now]) return
        if(k==JOY_UP &&Up[now]) return Up[no
        if(k==JOY_DN &&Down[now]) return Dow
        if(k==JOY_LF &&Left[now]) return Lef
        if(k==JOY_RT &&Right[now]) return Ri
    }
}

void cls(){
    waitvblank();
    clrscr();
```

NATURAL

NATURAL LANGUAGE PROGRAMMING

The castle exterior is scenery in the drawbridge. The printed name of the castle exterior is "castle". Understand "tower" or "tower" or "drawbridge" or "bridge" as the castle exterior. The description is "The drawbridge looks longer than it actually is; the towers are so high that the tops are lost in cloud, and looking east or west, you cannot see the furthest extent of the walls. An optical illusion: it is smaller inside.

Probably."

The iron-barred gate is a door. "An iron-barred gate leads [gate direction]." It is north of the Drawbridge and south of the Entrance Hall. It is closed and openable. Before entering the castle, try entering the gate instead. Before going inside in the Drawbridge, try going north instead. Understand "door" as the gate.

After opening the gate:
say "You shouldn't be able to

VISUAL

PATCH AND FLOW CONTROL PROGRAMMING



SOURCE

IMPERATIVE PROGRAMMING

```
#include <conio.h>
#include <string.h>

#define FULLSCREEN 512
char buf[FULLSCREEN+1];

#include "cold.h"
#include "joy.h"

char * patc="\n\nPress any key to continue

char move(char now){
    char k;
    if(Up[now]==0) Up[now]=now; //if key UP
0 page
    while(k=joyinput()){
        // if (now==0 && k==JOY_UP) return now;
        if(k==JOY_A &&A[now]) return A[now];
        if(k==JOY_B &&B[now]) return B[now];
        if(k==JOY_SEL &&Select[now]) return Select[now];
        if(k==JOY_STA &&Start[now]) return Start[now];
        if(k==JOY_UP &&Up[now]) return Up[now];
        if(k==JOY_DN &&Down[now]) return Down[now];
        if(k==JOY_LF &&Left[now]) return Left[now];
        if(k==JOY_RT &&Right[now]) return Right[now];
    }
}

void cls(){
    waitvblank();
    clrscr();
}
```

NATURAL

NATURAL LANGUAGE PROGRAMMING

VISUAL

PATCH AND FLOW CONTROL PROGRAMMING

SOURCE

IMPERATIVE PROGRAMMING

```
#include <conio.h>
#include <string.h>

#define FULLSCREEN 512
char buf[FULLSCREEN+1];

#include "cold.h"
#include "joy.h"

char * patc="\n\rPress any key to continue

char move(char now){
    char k;
    if(Up[now]==0) Up[now]=now; //if key UP
0 page
    while(k=joyinput()){
        // if (now==0 && k==JOY_UP) return now;
        if(k==JOY_A &&A[now]) return A[now];
        if(k==JOY_B &&B[now]) return B[now];
        if(k==JOY_SEL &&Select[now]) return Select[now];
        if(k==JOY_STA &&Start[now]) return Start[now];
        if(k==JOY_UP &&Up[now]) return Up[now];
        if(k==JOY_DN &&Down[now]) return Down[now];
        if(k==JOY_LF &&Left[now]) return Left[now];
        if(k==JOY_RT &&Right[now]) return Right[now];
    }
}

void cls(){
    waitvblank();
    clrscr();
}
```

prototypical
or stereotypical
code

```

#include <conio.h>
#include <string.h>

#define FULLSCREEN 512
char buf[FULLSCREEN+1];

#include "cold.h"
#include "joy.h"

char * patc="\n\rPress any key to continue ...";

char move(char now){
    char k;
    if(Up[now]==0) Up[now]=now; //if key UP not used ,use it to restart
    0 page
    while(k=joyinput()){
        // if (now==0 && k==JOY_UP) return now; //patch for the 0 page
        if(k==JOY_A &&A[now]) return A[now];
        if(k==JOY_B &&B[now]) return B[now];
        if(k==JOY_SEL &&Select[now]) return Select[now];
        if(k==JOY_STA &&Start[now]) return Start[now];
        if(k==JOY_UP &&Up[now]) return Up[now];
        if(k==JOY_DN &&Down[now]) return Down[now];
        if(k==JOY_LF &&Left[now]) return Left[now];
        if(k==JOY_RT &&Right[now]) return Right[now];
    }
}

void cls(){
    waitvblank();

```

```
#include <conio.h>
#include <string.h>

#define FULLSCREEN 512
char buf[FULLSCREEN+1];
```

```
#include "cold.h"
#include "joy.h"
```

```
char * patc="\n\rPress any key to continue ...";
```

```
char move(char now){
    char k;
```

```
    if(Up[now]==0) Up[now]=now; //if key UP not used ,
0 page
```

```
    while(k=joyinput()){
        // if (now==0 && k==JOY_UP) return now; //patch for the 0 page
        if(k==JOY_A &&A[now]) return A[now];
        if(k==JOY_B &&B[now]) return B[now];
        if(k==JOY_SEL &&Select[now]) return Select[now];
        if(k==JOY_STA &&Start[now]) return Start[now];
        if(k==JOY_UP &&Up[now]) return Up[now];
        if(k==JOY_DN &&Down[now]) return Down[now];
        if(k==JOY_LF &&Left[now]) return Left[now];
        if(k==JOY_RT &&Right[now]) return Right[now];
    }
}
```

```
void cls(){
    waitvblank();
```

SOURCE

OPERATION / DATA

CONCISION

REPETITION

EMPHASIS ON SYNTAX

SPECIALIZED VOCABULARY

COMMENTS

```
#include <conio.h>
#include <string.h>

#define FULLSCREEN 512
char buf[FULLSCREEN+1];
```

```
#include "cold.h"
#include "joy.h"
```

```
char * patc="\n\rPress any key to continue ...";
```

```
char move(char now){
    char k;
    if(Up[now]==0) Up[now]=now; //if key UP not used use it to restart
0 page
    while(k=joyinput()){
        // if (now==0 && k==JOY_UP) return now; //page 4 and 5 page
        if(k==JOY_A && A[now]) return A[now];
        if(k==JOY_B && B[now]) return B[now];
        if(k==JOY_SEL && Select[now]) return Select[now];
        if(k==JOY_STA && Start[now]) return Start[now];
        if(k==JOY_UP && Up[now]) return Up[now];
        if(k==JOY_DN && Down[now]) return Down[now];
        if(k==JOY_LF && Left[now]) return Left[now];
        if(k==JOY_RT && Right[now]) return Right[now];
    }
}
```

```
}
void cls(){
    waitvblank();
```

“source”
a point of
departure

SOURCE

IMPERATIVE PROGRAMMING

```
#include <conio.h>
#include <string.h>

#define FULLSCREEN 512
char buf[FULLSCREEN+1];

#include "cold.h"
#include "joy.h"

char * patc="\n\nPress any key to continue

char move(char now){
    char k;
    if(Up[now]==0) Up[now]=now; //if key UP
0 page
    while(k=joyinput()){
        // if (now==0 && k==JOY_UP) return now;
        if(k==JOY_A && A[now]) return A[now];
        if(k==JOY_B && B[now]) return B[now];
        if(k==JOY_SEL && Select[now]) return Select[now];
        if(k==JOY_STA && Start[now]) return Start[now];
        if(k==JOY_UP && Up[now]) return Up[now];
        if(k==JOY_DN && Down[now]) return Down[now];
        if(k==JOY_LF && Left[now]) return Left[now];
        if(k==JOY_RT && Right[now]) return Right[now];
    }
}

void cls(){
    waitvblank();
    clrscr();
}
```

NATURAL

NATURAL LANGUAGE PROGRAMMING

VISUAL

PATCH AND FLOW CONTROL PROGRAMMING

SOURCE

IMPERATIVE PROGRAMMING

NATURAL

NATURAL LANGUAGE PROGRAMMING

```
The castle exterior is scenery in
the drawbridge. The printed name of
the castle exterior is "castle".
Understand "tower" or "tower" or
"drawbridge" or "bridge" as the
castle exterior. The description is
"The drawbridge looks longer than
it actually is; the towers are so
high that the tops are lost in
cloud, and looking east or west,
you cannot see the furthest extent
of the walls. An optical illusion:
it is smaller inside.
```

```
Probably."
```

```
The iron-barred gate is a door. "An
iron-barred gate leads [gate
direction]." It is north of the
Drawbridge and south of the
Entrance Hall. It is closed and
openable. Before entering the
castle, try entering the gate
instead. Before going inside in the
Drawbridge, try going north
instead. Understand "door" as the
gate.
```

```
After opening the gate:
say "You shouldn't be able to
```

VISUAL

PATCH AND FLOW CONTROL PROGRAMMING

The castle exterior is scenery in the drawbridge. The printed name of the castle exterior is "castle". Understand "tower" or "tower" or "drawbridge" or "bridge" as the castle exterior. The description is "The drawbridge looks longer than it actually is; the towers are so high that the tops are lost in cloud, and looking east or west, you cannot see the furthest extent of the walls. An optical illusion: it is smaller inside.

Probably."

The iron-barred gate is a door. "An iron-barred gate leads [gate direction]." It is north of the Drawbridge and south of the Entrance Hall. It is closed and openable. Before entering the castle, try entering the gate instead. Before going inside in the Drawbridge, try going north instead. Understand "door" as the gate.

After opening the gate:

say "You shouldn't be able to open it, heavy as it is, but it swings aside lightly at your

The castle exterior is scenery in the drawbridge. The printed name of the castle exterior is "castle". Understand "tower" or "tower" or "drawbridge" or "bridge" as the castle exterior. The description is "The drawbridge looks longer than it actually is; the towers are so high that the tops are lost in cloud, and looking east or west you cannot see the furthest of the walls. An optical illusion: it is smaller inside.

Probably."

The iron-barred gate is a door. "An iron-barred gate leads [gate direction]." It is north of the Drawbridge and south of the Entrance Hall. It is openable. Before entering the castle, try entering the gate instead. Before going inside in the Drawbridge, try going north instead. Understand "door" as the gate.

After opening the gate:
say "You shouldn't be able to open it, heavy as it is, but it swings aside lightly at your

NLP

EXPRESSIVE POWER

ACCESSIBILITY

IMITATION OF COMPLEXITY

OVERDETERMINATION

The castle exterior is scenery in the drawbridge. The printed name of the castle exterior is "castle". Understand "tower" or "tower" or "drawbridge" or "bridge" as the castle exterior. The description is "The drawbridge looks longer than it actually is; the towers are so high that the tops are lost in cloud, and looking eastward, you cannot see the full extent of the walls. An optical illusion: it is smaller inside.

Probably."

The iron-barred gate is a door. "An iron-barred gate leads [gate direction]." It is north of the Drawbridge and south of the Entrance Hall. It is closed and openable. Before entering the castle, try entering the gate instead. Before going inside in the Drawbridge, try going north instead. Understand "door" as the gate.

After opening the gate:
say "You shouldn't be able to open it, heavy as it is, but it swings aside lightly at your

the problem: abstraction and depth

The castle exterior is so named for the drawbridge. The proper name of the castle exterior is "castle". Understand "tower" or "tower" or "drawbridge" or "bridge" as the castle exterior. The description is "The drawbridge looks longer than it actually is; the towers are so high that the tops are lost in cloud, and looking east or west, you cannot see the furthest extent of the walls. An optical illusion: it is smaller inside.

Probably."

The iron-barred gate is a door. "An iron-barred gate leads [gate direction]." It is north of the Drawbridge and south of the Entrance Hall. It is closed and openable. Before entering the castle, try entering the gate instead. Before going inside in the Drawbridge, try going north instead. Understand "door" as the gate.

After opening the gate:
say "You shouldn't be able to open it, heavy as it is, but it swings aside lightly at your

which do we
interpret?

the code?

or the
compiler?

The castle exterior is scenery in the drawbridge. The printed name of the castle exterior is "castle". Understand "tower" or "tower" or "drawbridge" or "bridge" as the castle exterior. The description is "The drawbridge looks longer than it actually is; the towers are so high that the tops are lost in cloud, and looking east or west, you cannot see the furthest extent of the walls. An optical illusion: it is smaller inside.

Probably."

The iron-barred gate is a door. "An iron-barred gate leads [gate direction]." It is north of the Drawbridge and south of the Entrance Hall. It is closed and openable. Before entering the castle, try entering the gate instead. Before going inside in the Drawbridge, try going north instead. Understand "door" as the gate.

After opening the gate:
say "You shouldn't be able to open it, heavy as it is, but it swings aside lightly at your

implied reader
vs.
explicit reader

SOURCE

IMPERATIVE PROGRAMMING

NATURAL

NATURAL LANGUAGE PROGRAMMING

```
The castle exterior is scenery in
the drawbridge. The printed name of
the castle exterior is "castle".
Understand "tower" or "tower" or
"drawbridge" or "bridge" as the
castle exterior. The description is
"The drawbridge looks longer than
it actually is; the towers are so
high that the tops are lost in
cloud, and looking east or west,
you cannot see the furthest extent
of the walls. An optical illusion:
it is smaller inside.
```

```
Probably."
```

```
The iron-barred gate is a door. "An
iron-barred gate leads [gate
direction]." It is north of the
Drawbridge and south of the
Entrance Hall. It is closed and
openable. Before entering the
castle, try entering the gate
instead. Before going inside in the
Drawbridge, try going north
instead. Understand "door" as the
gate.
```

```
After opening the gate:
say "You shouldn't be able to
```

VISUAL

PATCH AND FLOW CONTROL PROGRAMMING

SOURCE

IMPERATIVE
PROGRAMMING

NATURAL

NATURAL
LANGUAGE
PROGRAMMING

VISUAL

PATCH AND
FLOW CONTROL
PROGRAMMING



Max/MSP



Max/MSP



VISUAL

SPATIAL RELATIONSHIPS

SELF-REPRESENTING

TRACED RATHER THAN READ

A CLOUD OF BLACK BOXES

Max/MSP



how do we
“close read”
a graphic
specification?
visual
literacy?

```
#include <conio.h>
#include <string.h>

#define FULLSCREEN 512
char buf[FULLSCREEN+1];

#include "col
#include "joy

char * patc="

char move(char
char k;
if(Up[now
0 page
while(k=j
// if (n
if(k=
if(k=
if(k=
if(k=
if(k=J
if(k=J
if(k=J
if(k=J
}

void cls(){
waitvblank();
clrscr();
gotoxy(0,0);
}
```

SOURCE

sequence

literacy

write

FLOW

space

visual literacy

map / graph

VISUAL



putting visual programming in context:

procedural flow in visual art

VISUAL



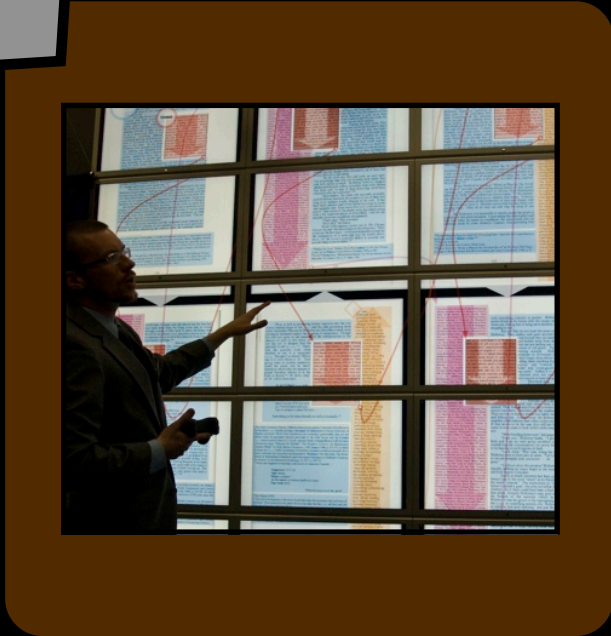
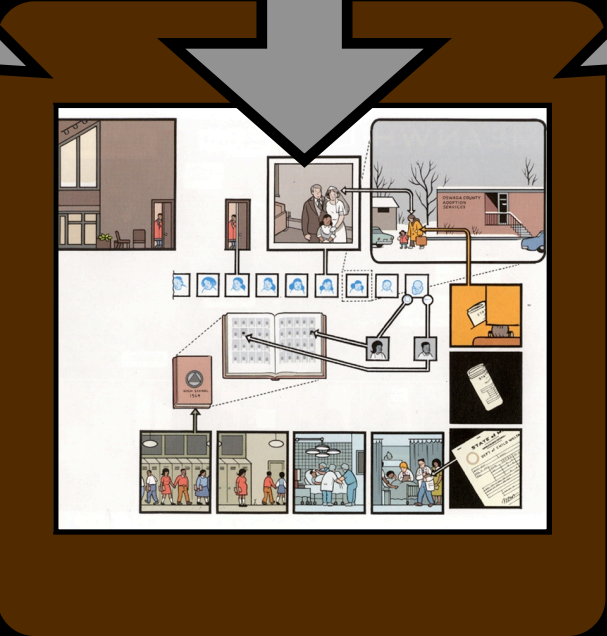
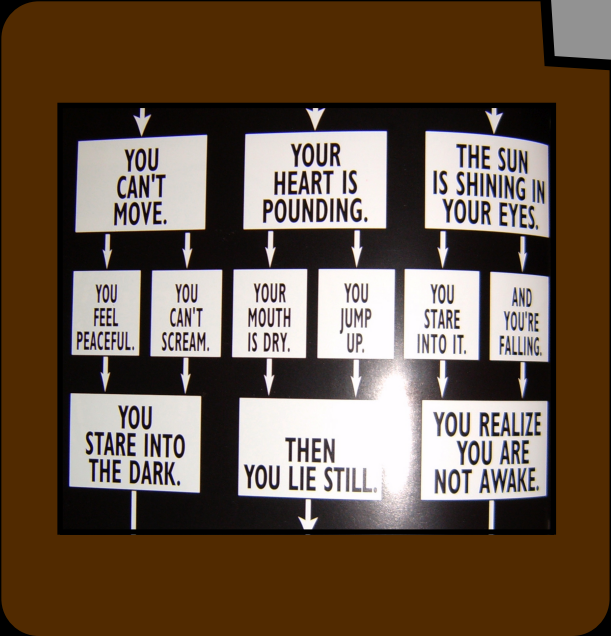
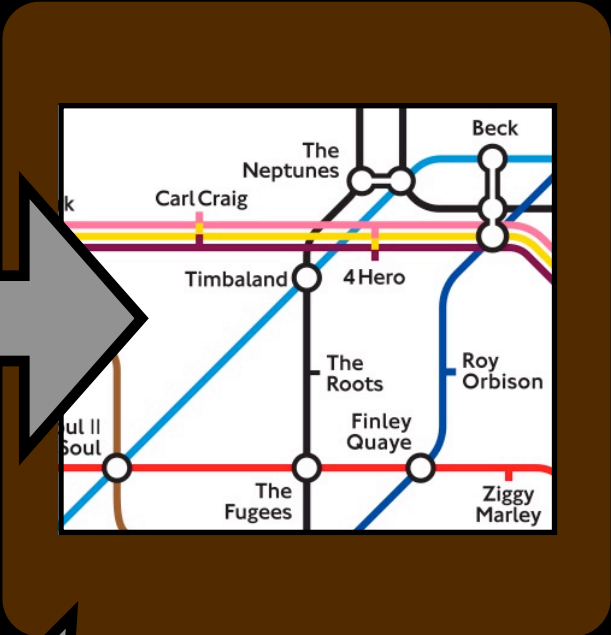
INFOVIZ

MAPS

SCRIPTS

COMICS

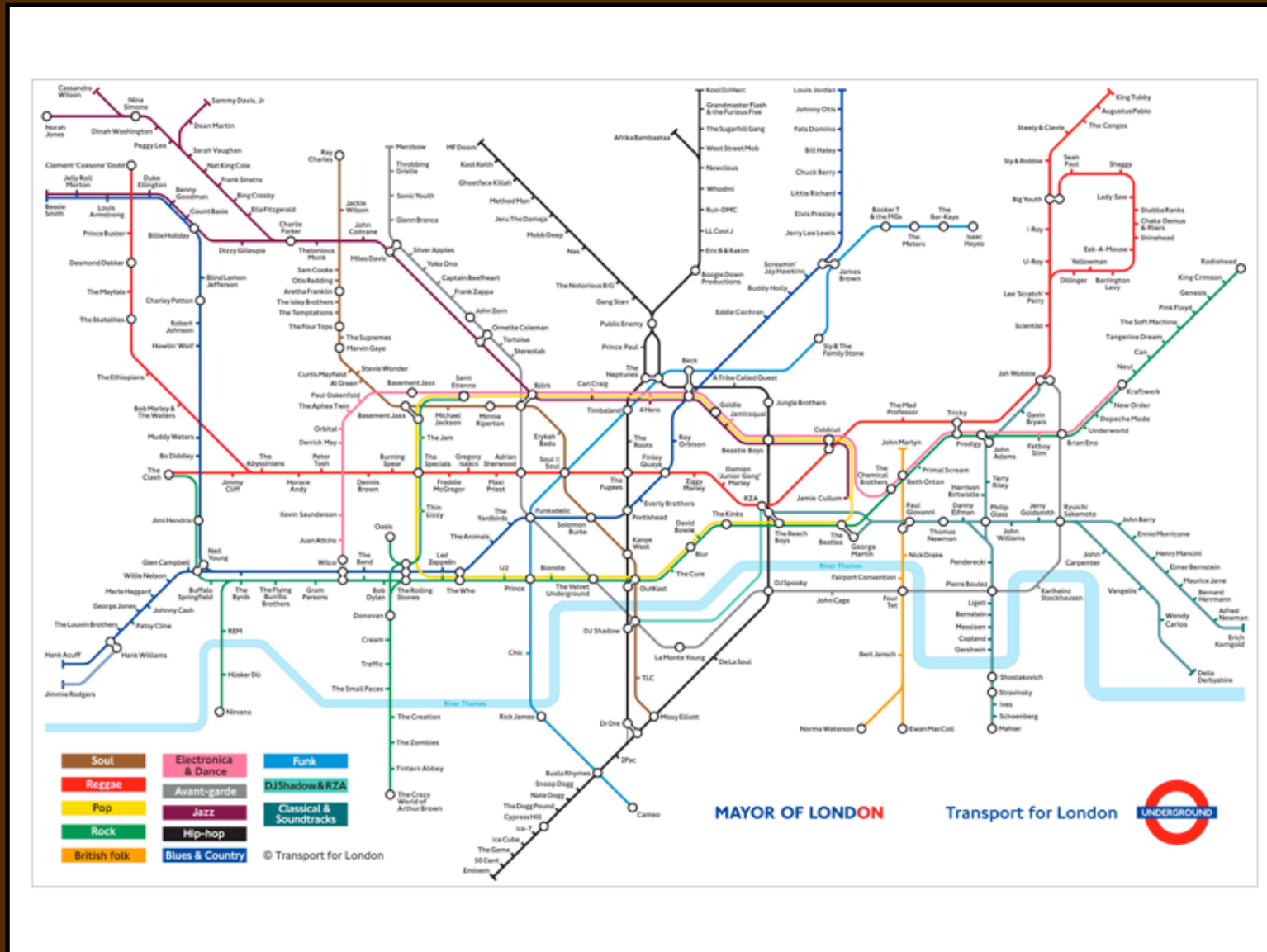
GUIs

[illegible]

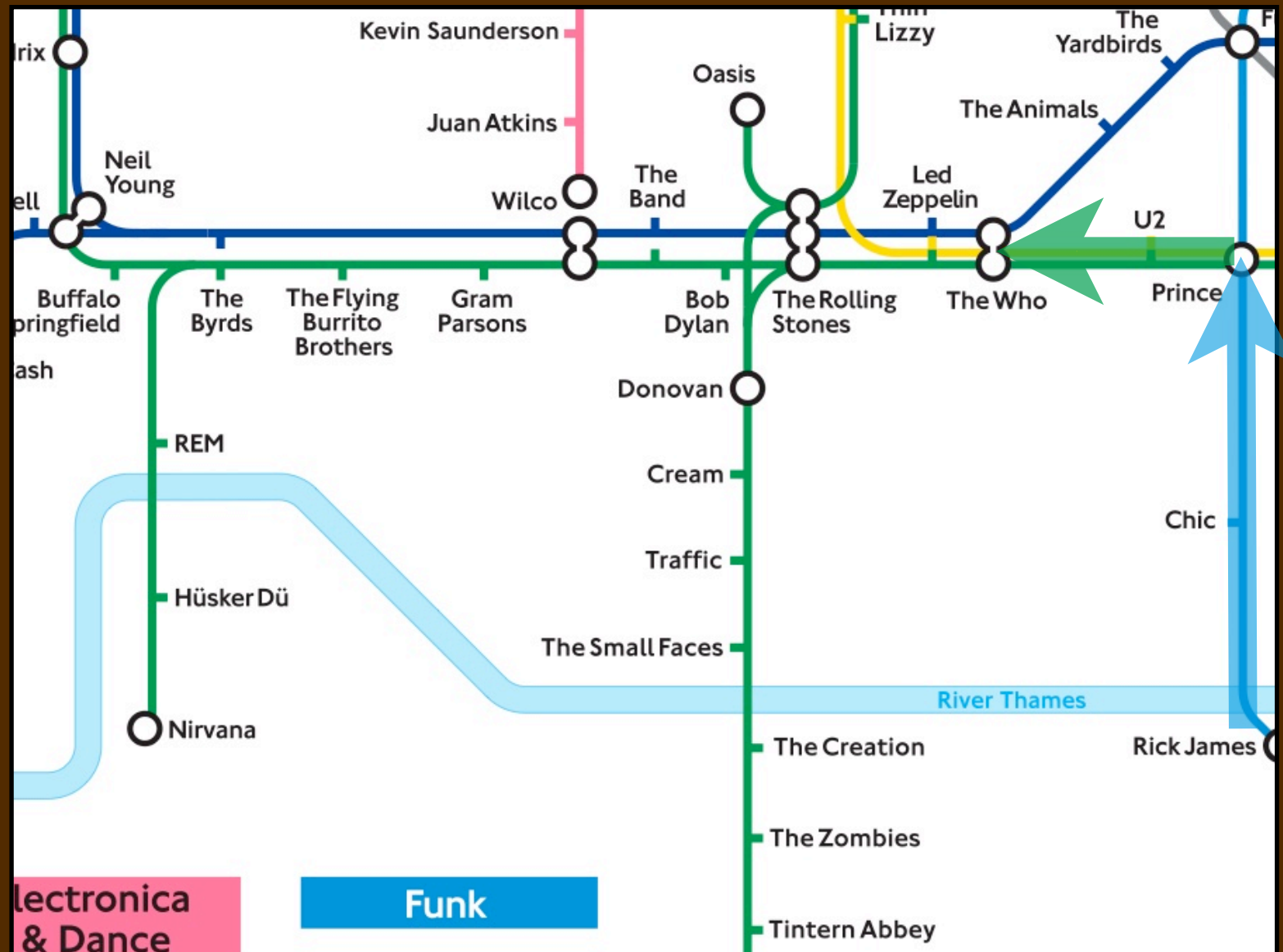
[illegible]

VisualComplexity.com (2008)

subway map remix art: specifying transitions

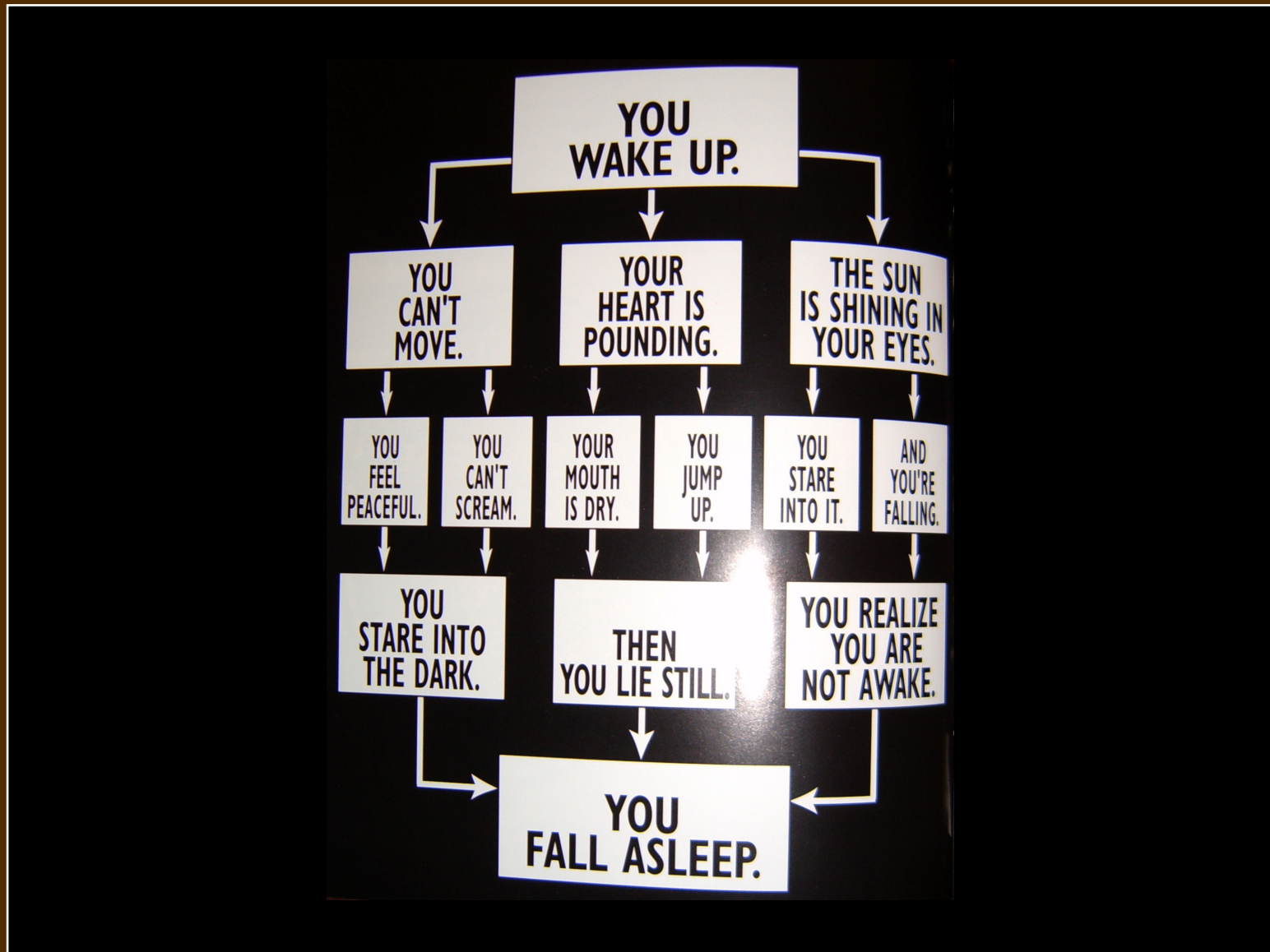


Dorian Lynskey's "Going Underground" (2006)



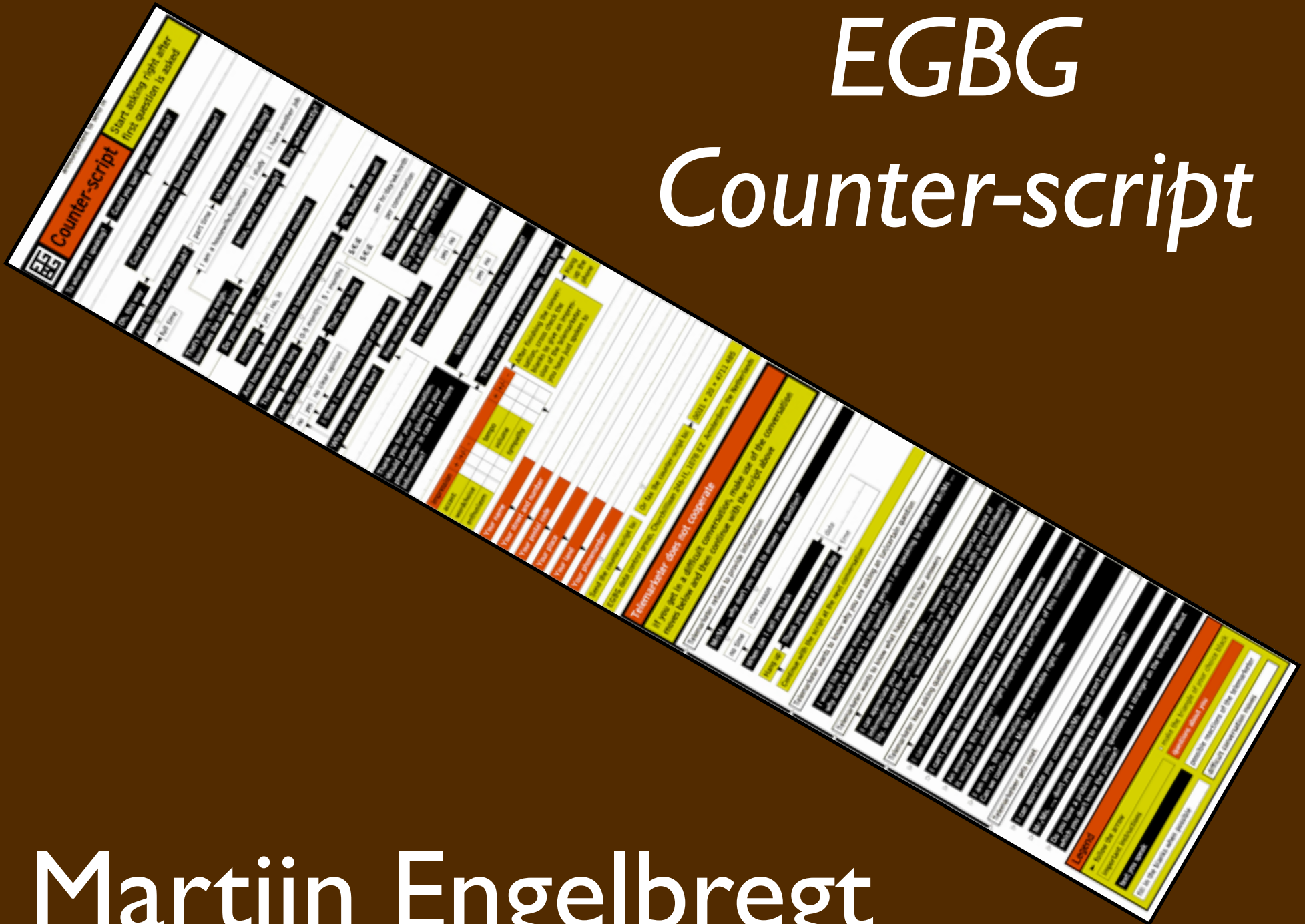
Dorian Lynskey's "Going Underground" (2006)

flowchart / directed graph as CYOA narrative



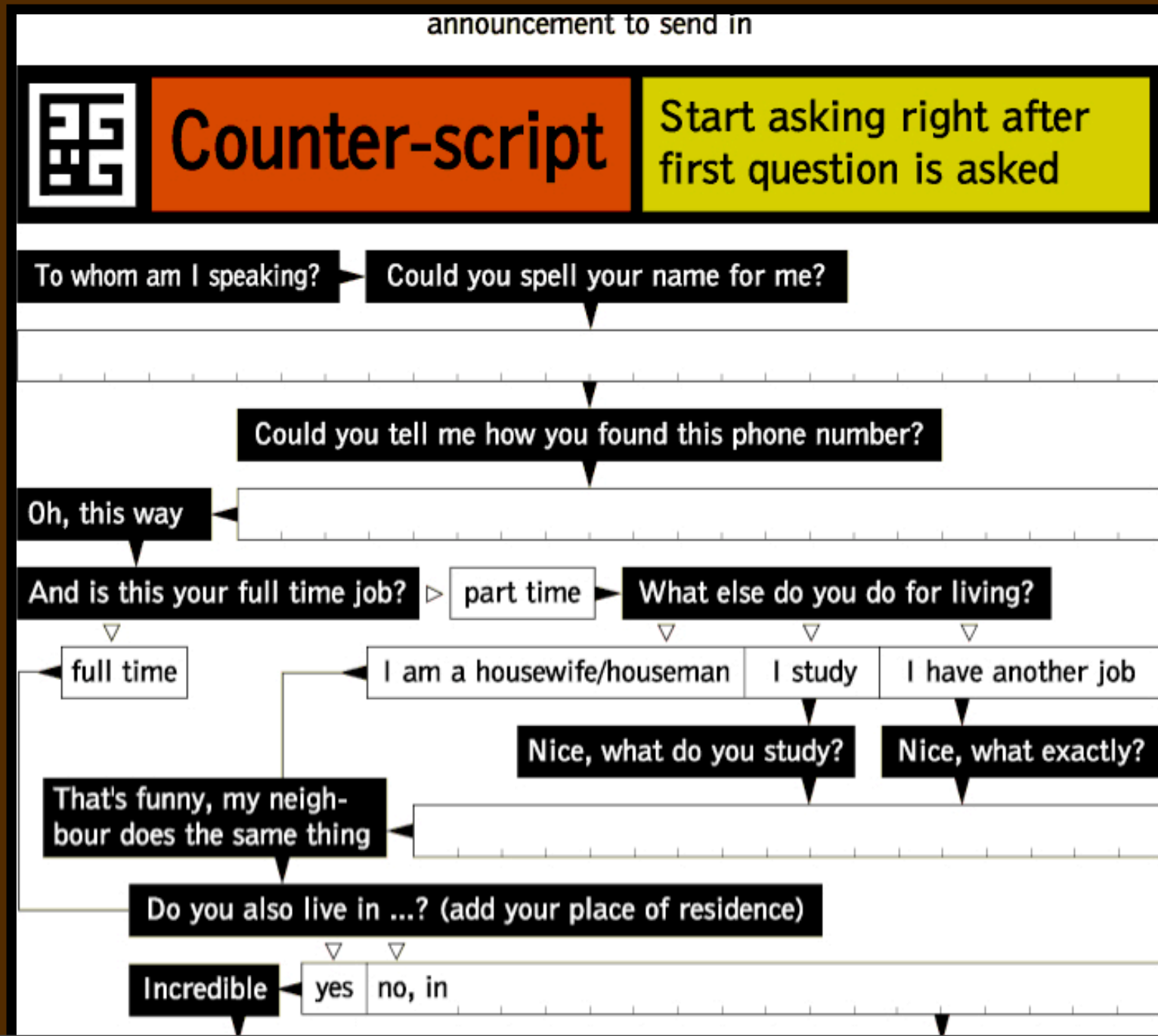
from Bill Barker's *SCHWA*

EGBG Counter-script



Martijn Engelbregt

visual flow specifies the protocol of performance



BAD || GOOD

PRO GUIDE | IMPROVE YOURSELF

8 FREE ISSUES FOR NEW SUBSCRIBERS

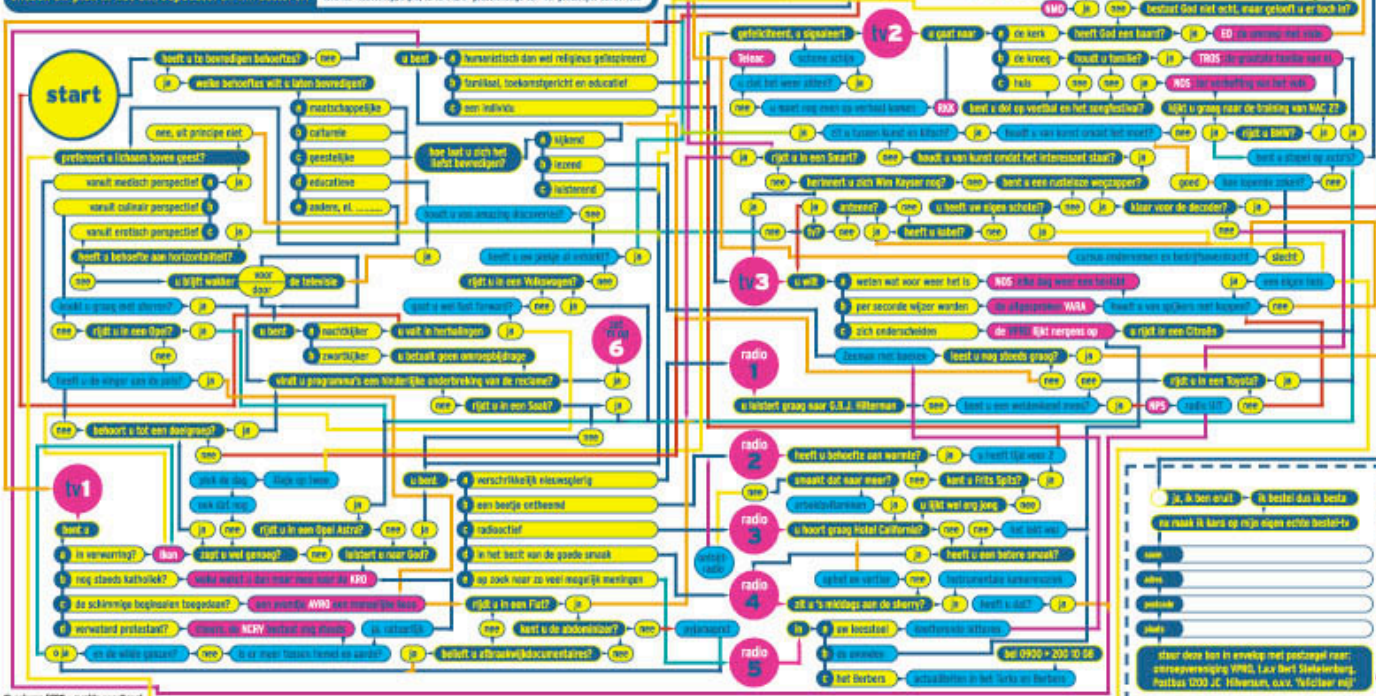
PHONE ➔ **+31-35-6712888**



ontdek uw plek in het omroepbestel en win bestel tv!

BESTELFORMULIER

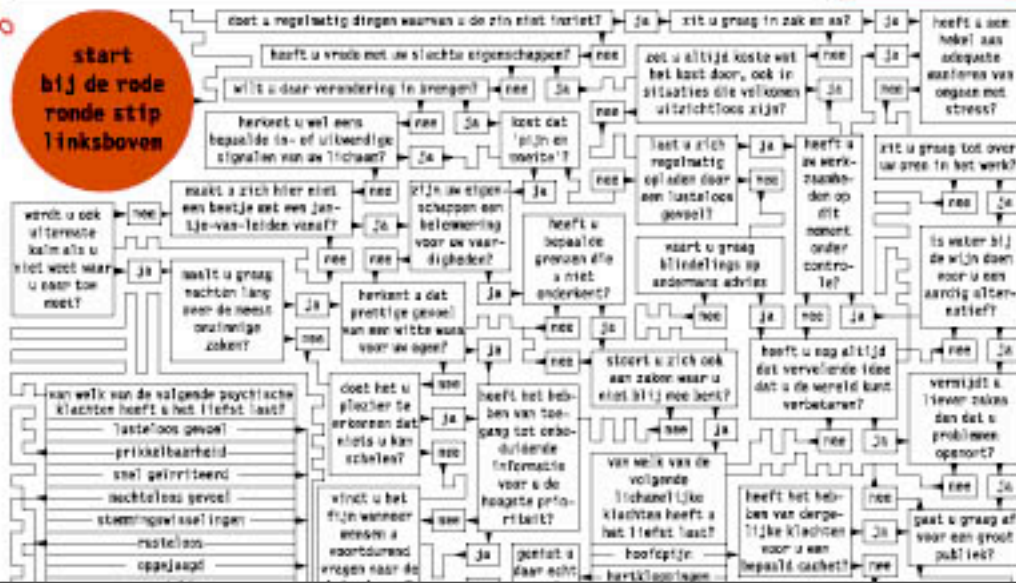
ontdek uw plek in het omroepbestel en win bestel-tv!



© software IGIS - martin esselbrecht

doe de test en maak kans op een uurtje ontspannende inspanning

Denk lang en diep na over elke vraag. Uw eerste antwoord is doorgaans niet het beste.
Volg uw intuïtie met een blauwe pen.



Metropoli

welk type

geef antwoord op de vragen in het schema: Metropolis M, antwoordschema B niet nodig, als u uw Metropolis M ni

Isabel a. José María Rodríguez * (MPC) así como?

Is uw naam, of de
aankondigende naam
van de aanbesteding, te
vermelden?

A diagram showing a red box labeled 'love' and a blue box labeled 'job' with arrows pointing to a white box labeled 'well-being'.

heeft u hi
van invlo
op de test

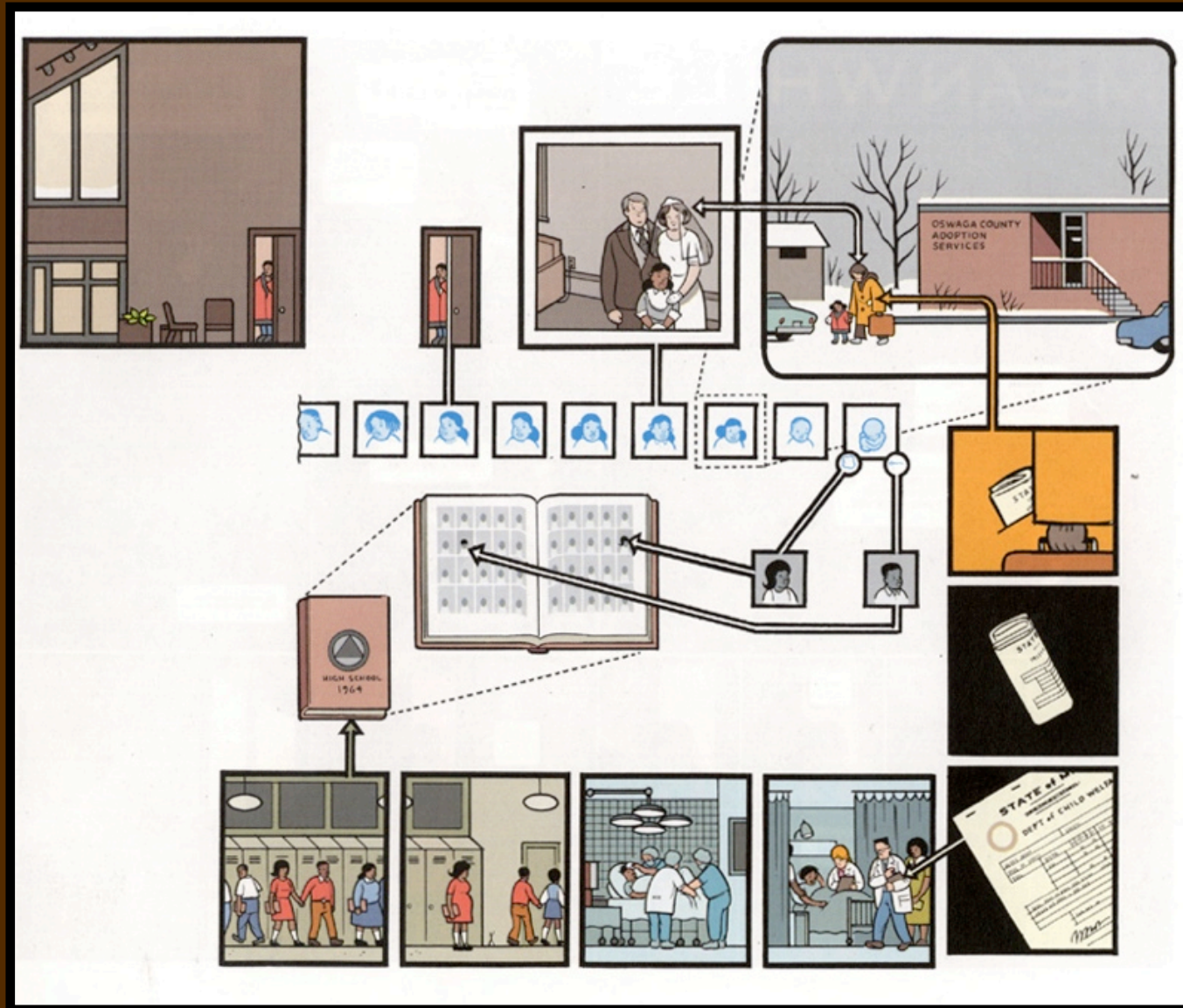
kijk u op tegen invattingen
of mensen waaraan de naam is

document mapping to trace branching paths



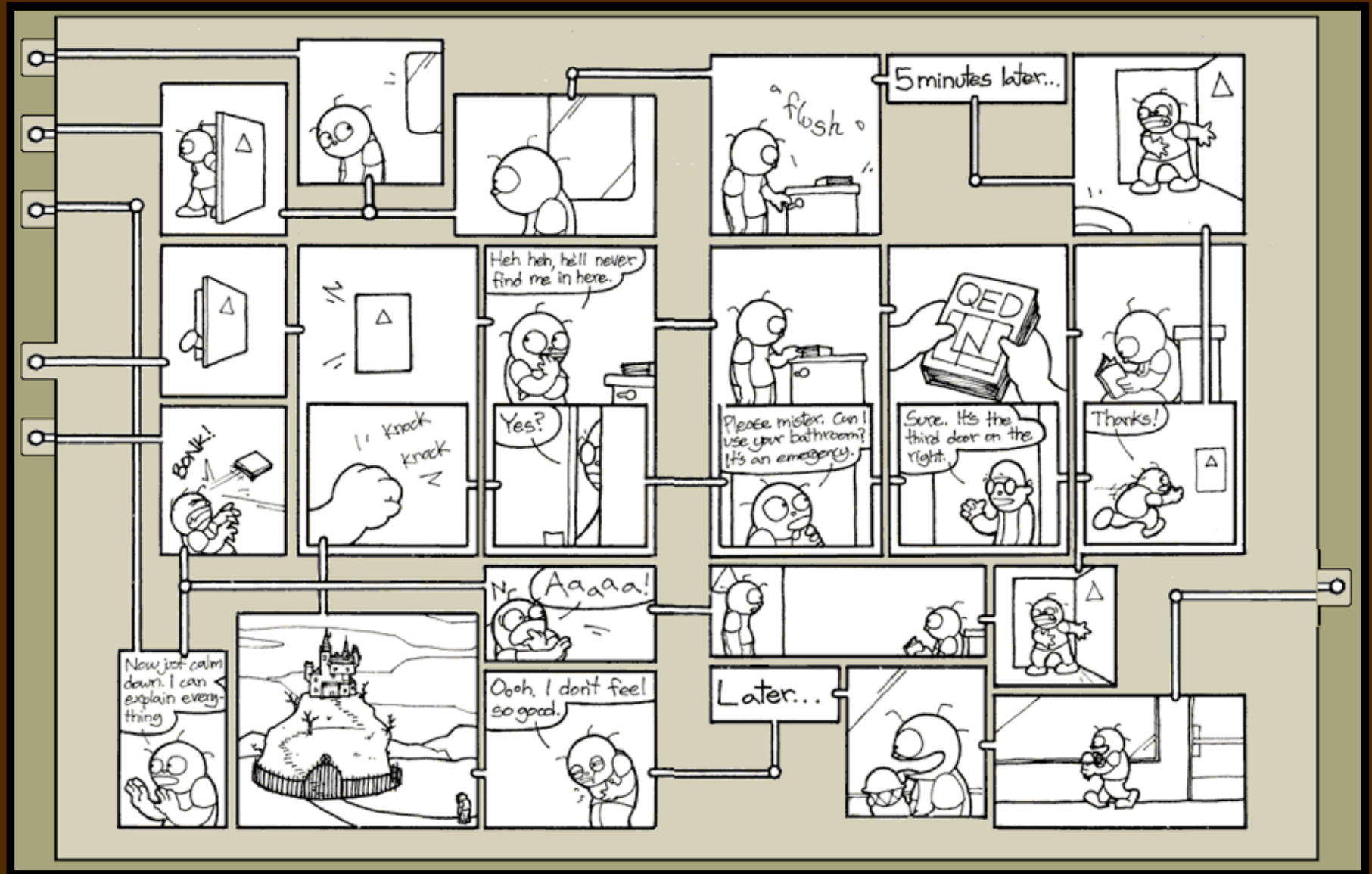
House of Leaves: mapping the labyrinth (2008)

comic links complex backstories with paths



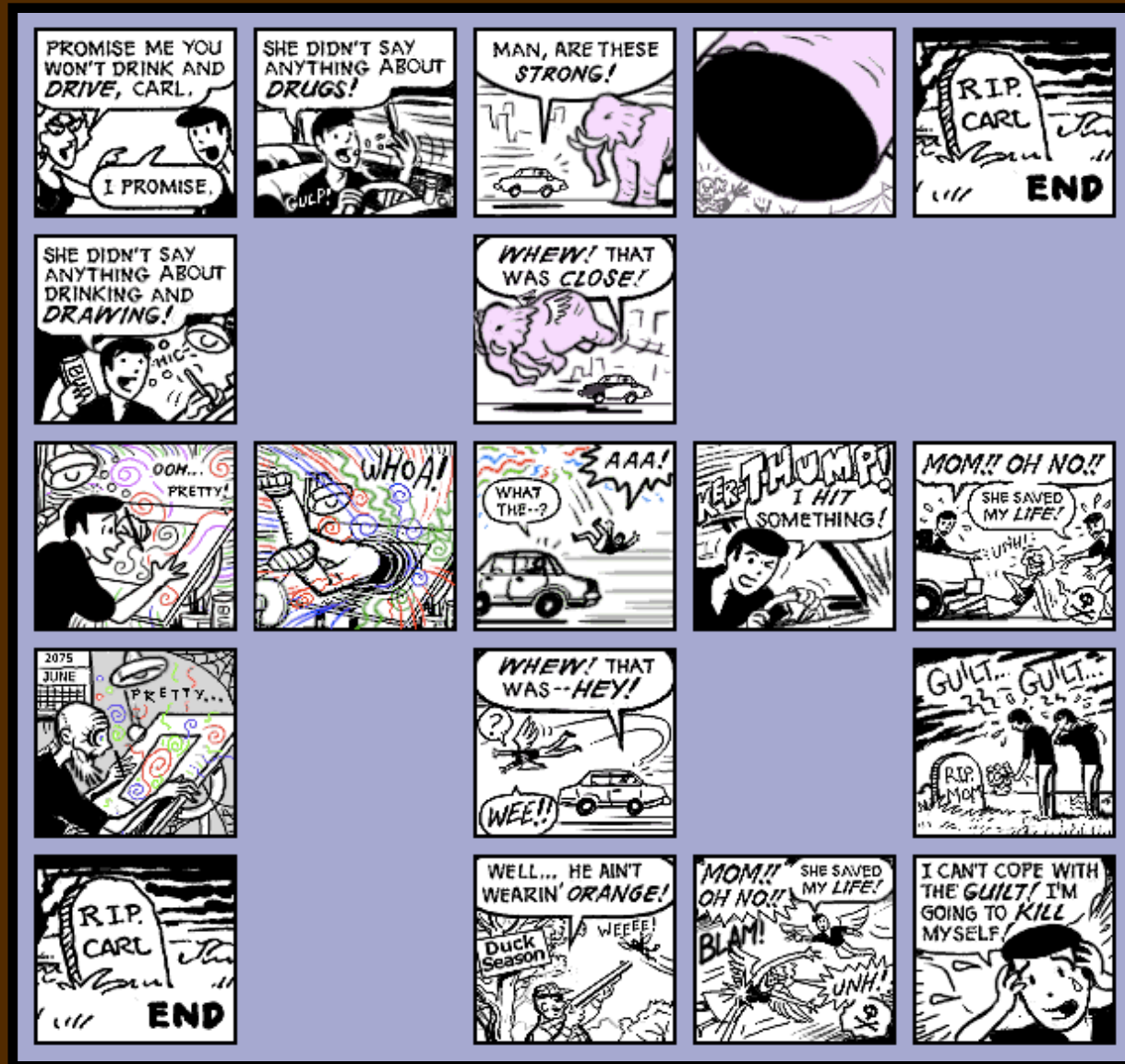
Chris Ware's *Jimmy Corrigan* [detail]

branching comic connects choices with paths



Jason Shiga's *Meanwhile*

branching comic embeds choices in a grid



Scott McCloud's *Carl* [detail]

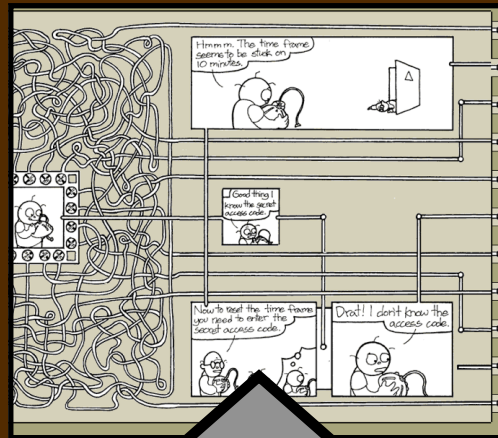
a variety of ways to signify
sequence,
relation, &
procedure

what techniques do
flow art &

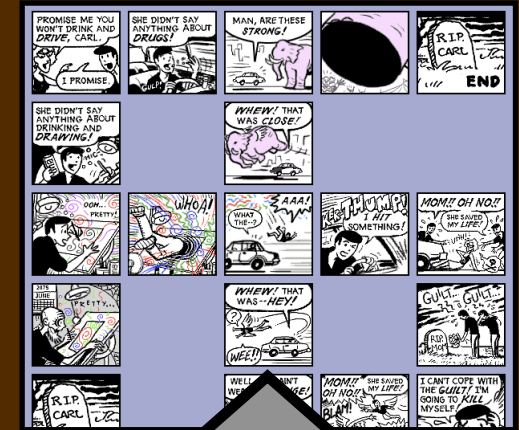
flow programming
have in common?

FLOW ART

PATCH

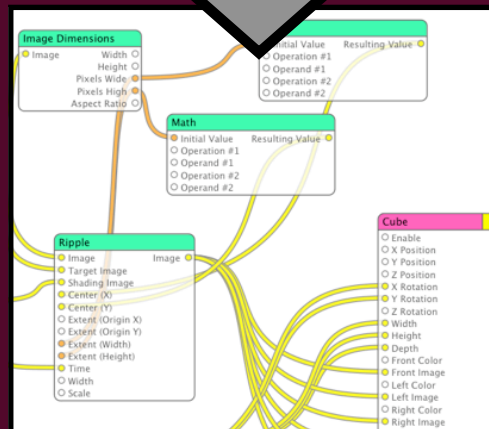


GRID



VISUAL PROGRAMMING

PATCH

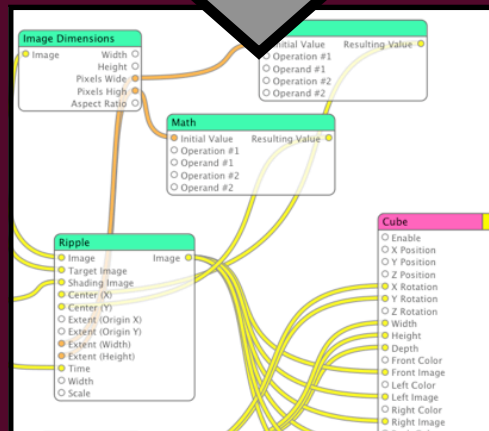
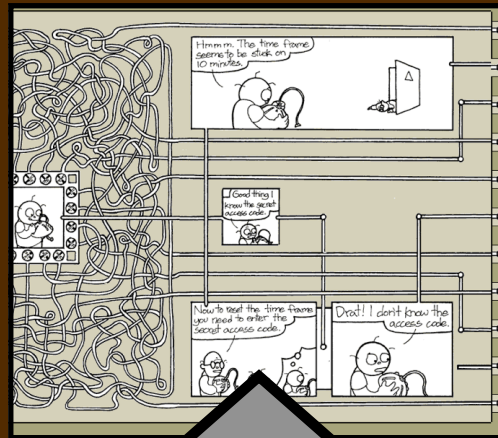


GRID



FLOW
ART

PATCH



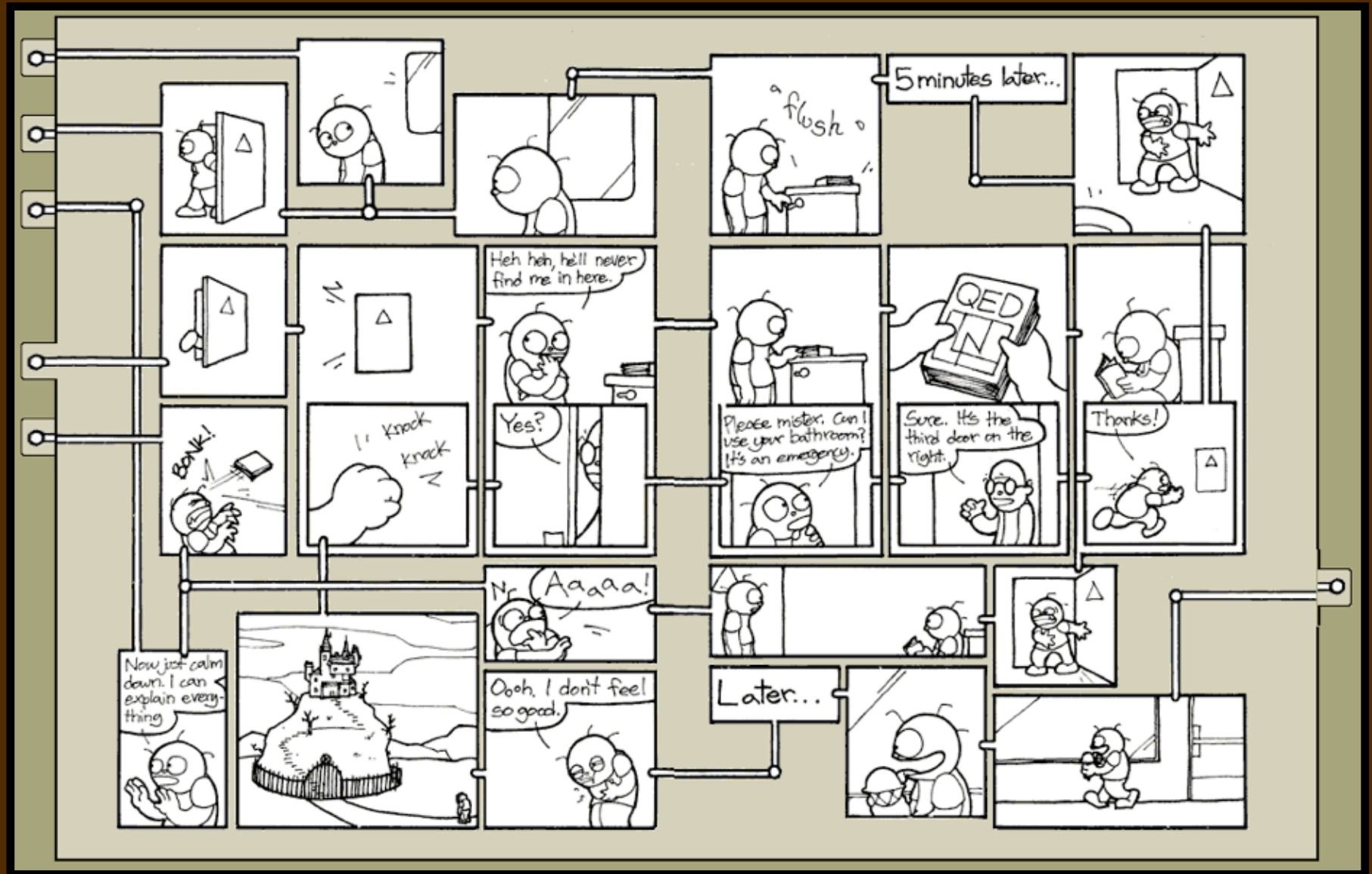
PATCH

GRID

GRID

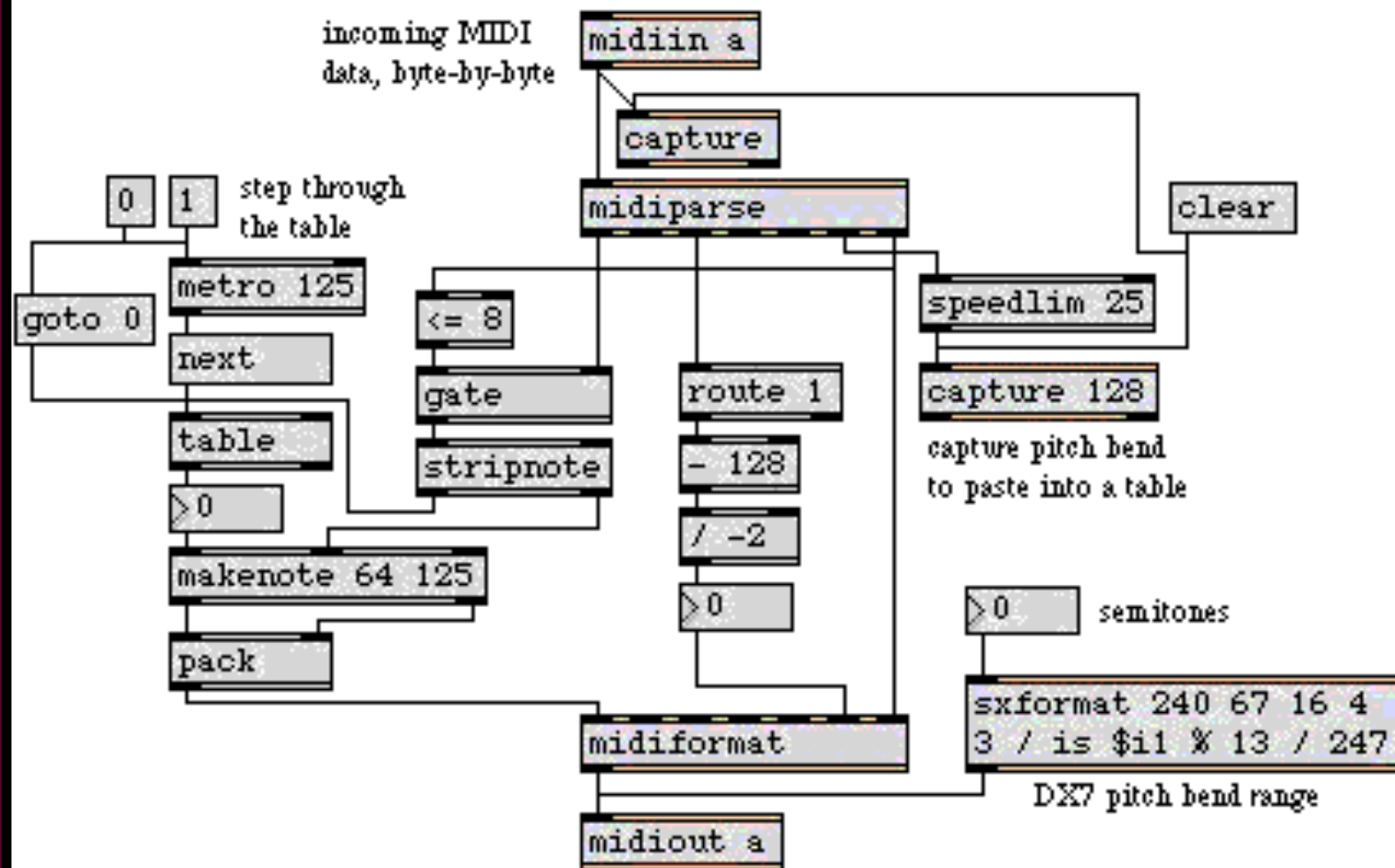
VISUAL
PROGRAMMING

branching comic connects choices with paths



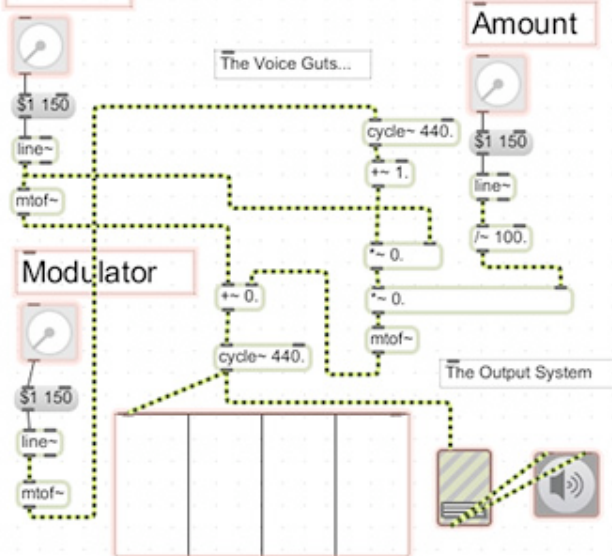
Jason Shiga's *Meanwhile*

Max/MSP

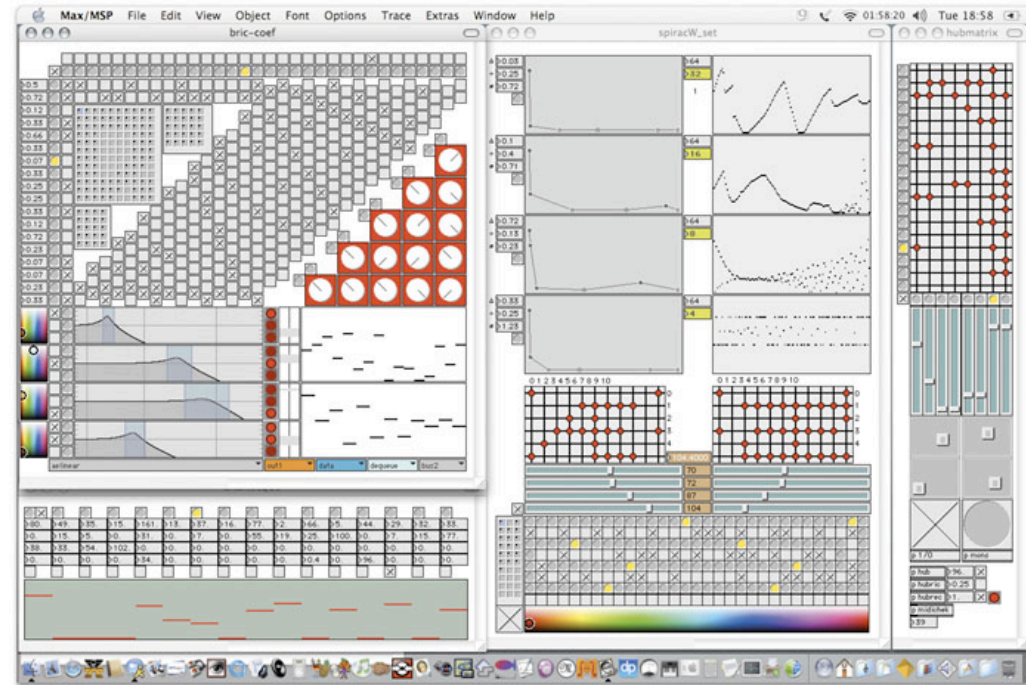
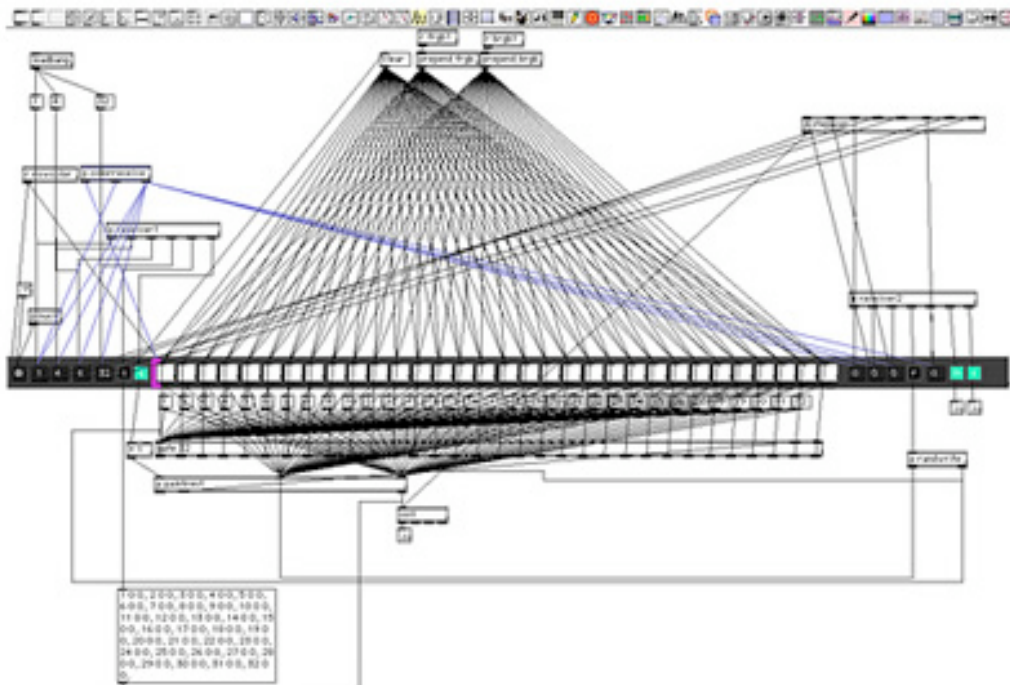
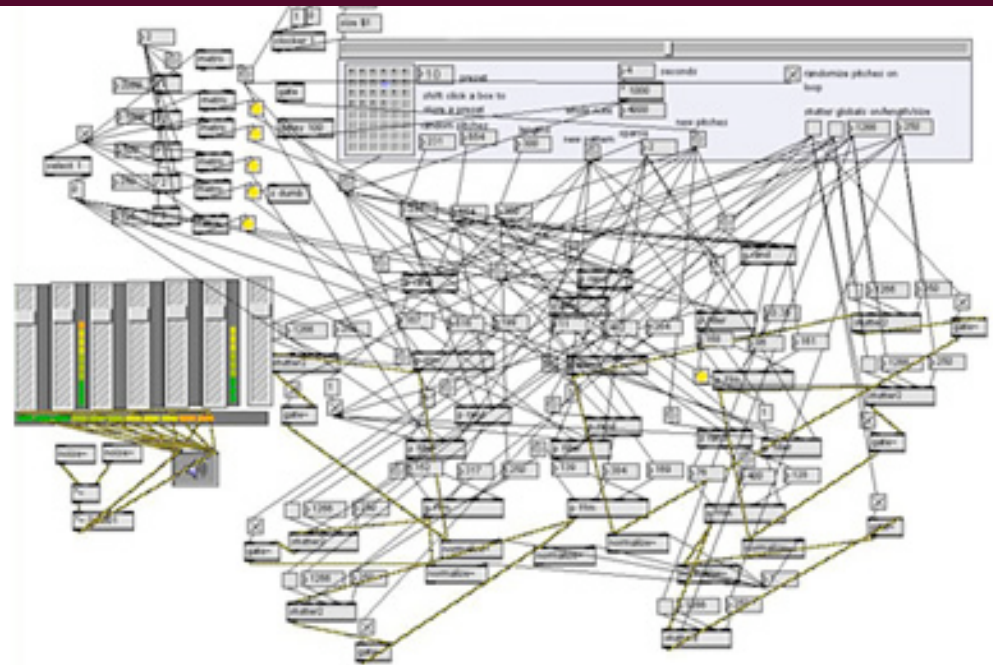
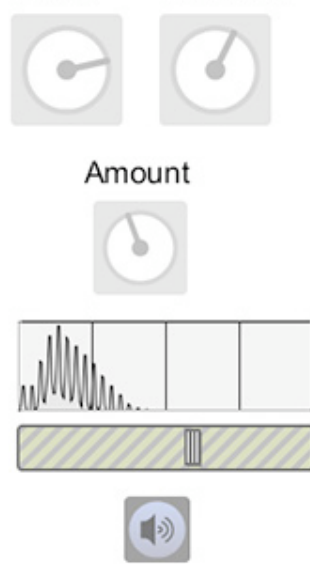


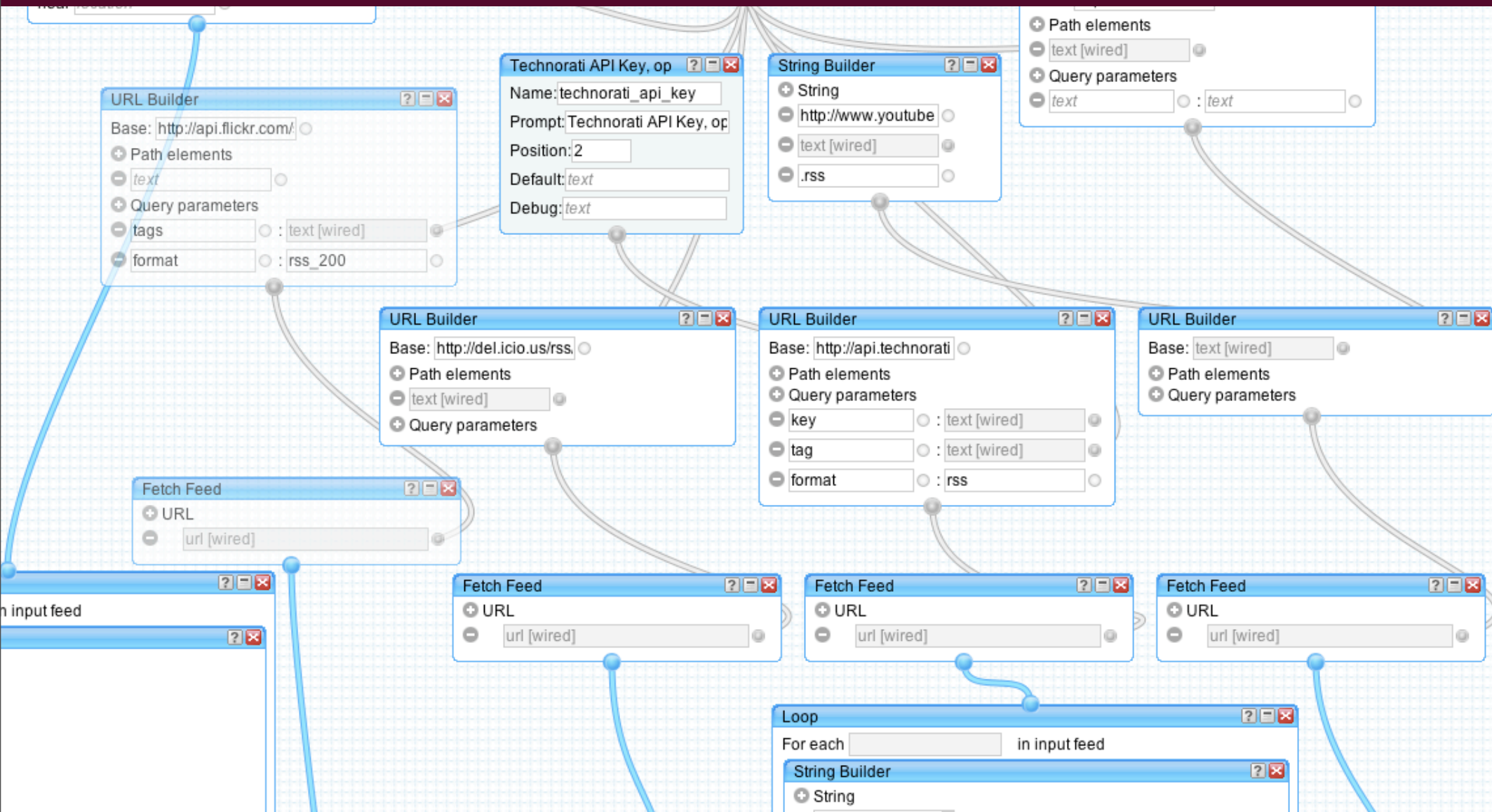
Miller Puckette / Cycling74's *Max / MSP*

Carrier

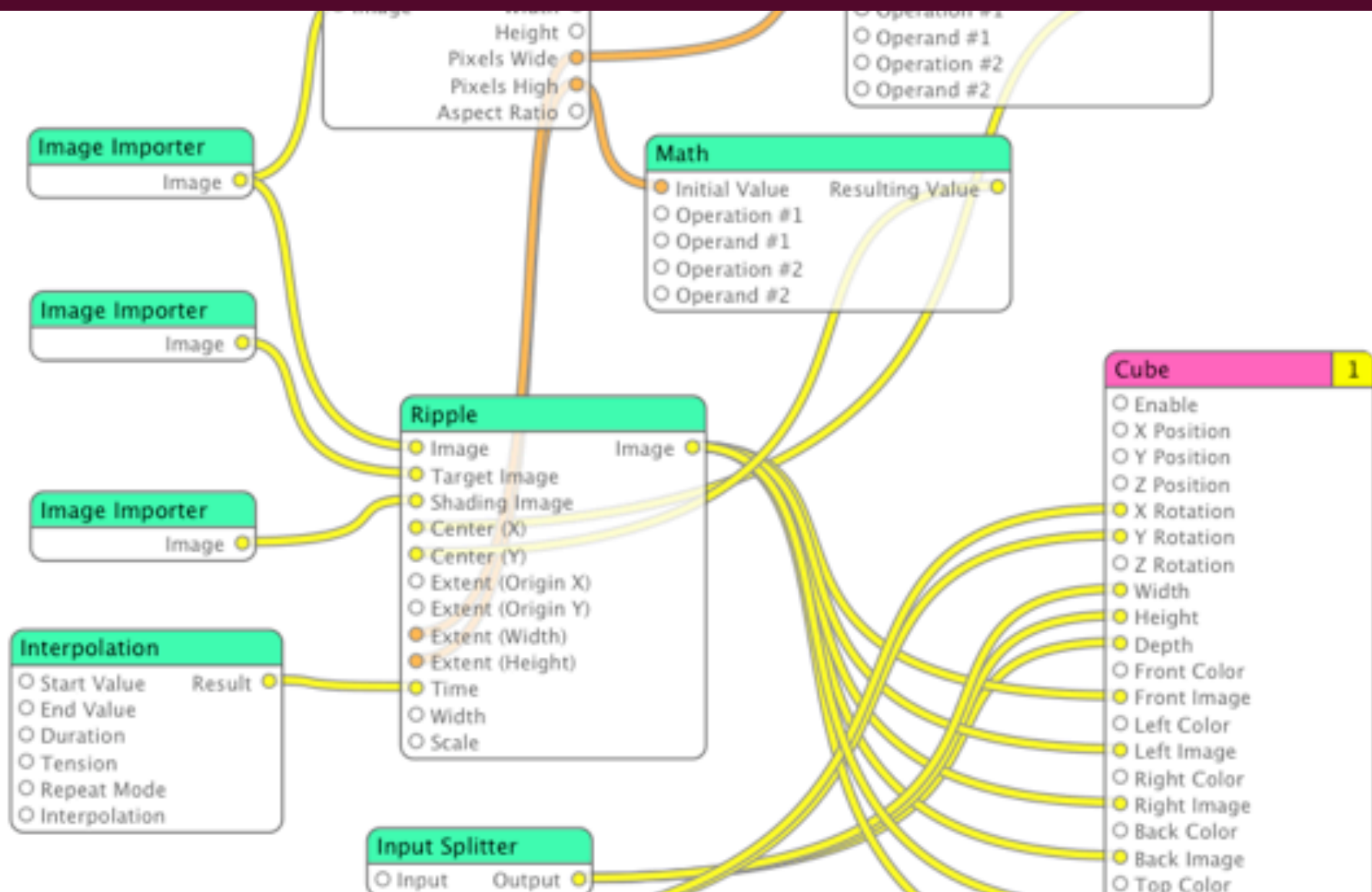


Carrier Modulator





Yahoo! Pipes



Quartz Composer (OS X)

Jason Shiga's *Meanwhile*

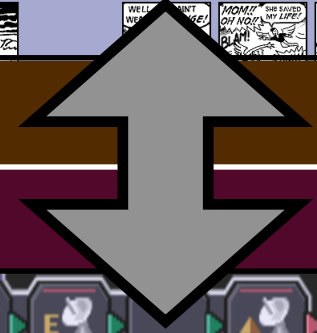
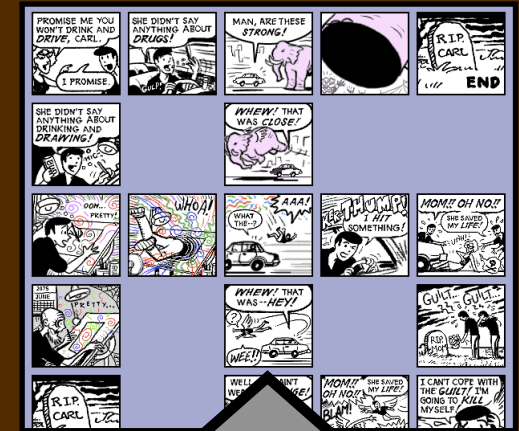


Quartz Composer (OS X)

FLOW
ART

PATCH

GRID



VISUAL
PROGRAMMING

PATCH

GRID





Carnage Heart

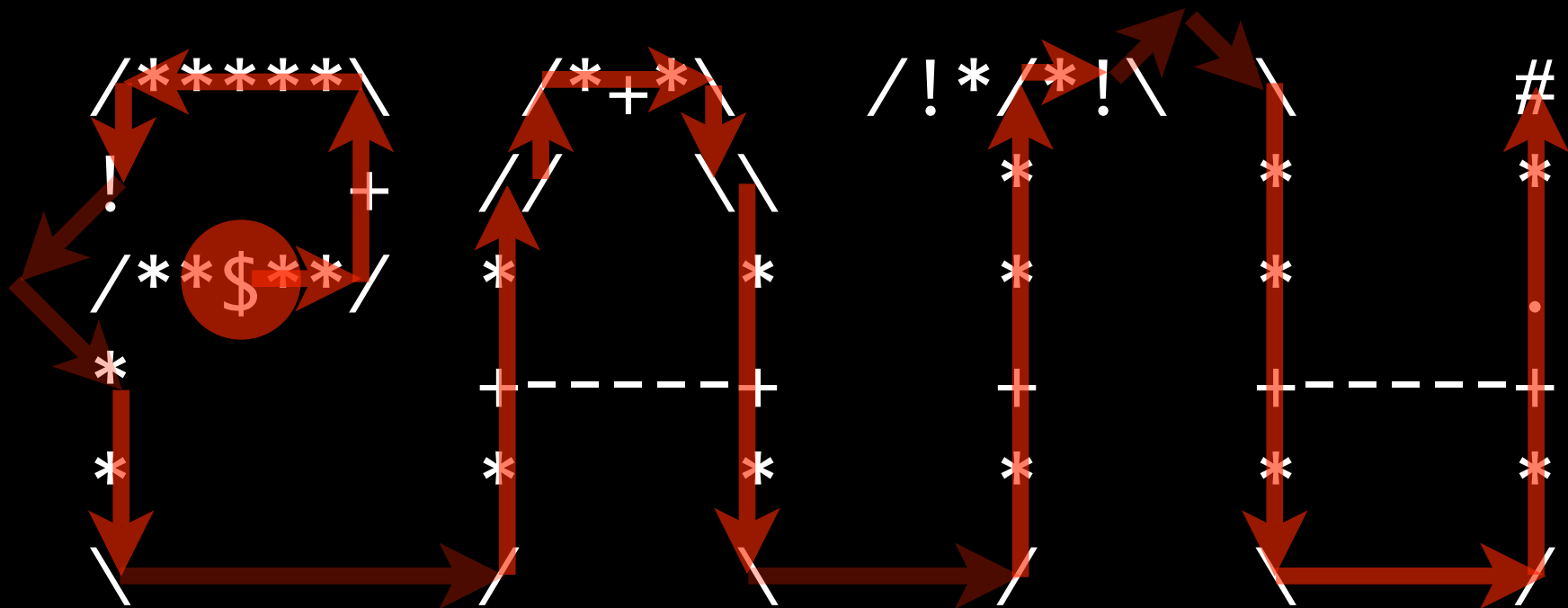


Scott McCloud's *Carl* [detail]



Carnage Heart: robot program [detail]

... “cartesian programming” ...



PATH esoteric / weird programming language

“ the PATH interpreter follows a path of instruction symbols which make up a program. The program can go up, down, left and right, and can also overlap itself. Also, the interpreter simply skips over any characters that aren't valid instruction symbols. ”

\$ Start the program here, heading right.
End the program.
/ Turn 90 degrees without crossing line.
\ Turn 90 degrees without crossing line
+ Increment the current memory cell.
- Decrement the current memory cell.
! Jump over the next symbol.
. Output a character from current memory.
, Input a character from current memory.
} Move to the next memory cell.
{ Move to the previous memory cell.
^ If current memory is not 0, turn up.
< If current memory is not 0, turn left.
> If current memory is not 0, turn right.
v If current memory is not 0, turn down.

PATCH

GRID

objects

directions

sequences

axes

relationships

spatial relations

conclusion

SOURCE

IMPERATIVE PROGRAMMING

```
#include <conio.h>
#include <string.h>

#define FULLSCREEN 512
char buf[FULLSCREEN+1];

#include "cold.h"
#include "joy.h"

char * patc="\n\rPress any key to continue

char move(char now){
    char k;
    if(Up[now]==0) Up[now]=now; //if key UP
0 page
    while(k=joyinput()){
        // if (now==0 && k==JOY_UP) return now
        if(k==JOY_A &&A[now]) return A[no
        if(k==JOY_B &&B[now]) return B[no
        if(k==JOY_SEL &&Select[now]) return
        if(k==JOY_STA &&Start[now]) return
        if(k==JOY_UP &&Up[now]) return Up[no
        if(k==JOY_DN &&Down[now]) return Dow
        if(k==JOY_LF &&Left[now]) return Lef
        if(k==JOY_RT &&Right[now]) return Ri
    }
}

void cls(){
    waitvblank();
    clrscr();
```

NATURAL

NATURAL LANGUAGE PROGRAMMING

The castle exterior is scenery in the drawbridge. The printed name of the castle exterior is "castle". Understand "tower" or "tower" or "drawbridge" or "bridge" as the castle exterior. The description is "The drawbridge looks longer than it actually is; the towers are so high that the tops are lost in cloud, and looking east or west, you cannot see the furthest extent of the walls. An optical illusion: it is smaller inside.

Probably."

The iron-barred gate is a door. "An iron-barred gate leads [gate direction]." It is north of the Drawbridge and south of the Entrance Hall. It is closed and openable. Before entering the castle, try entering the gate instead. Before going inside in the Drawbridge, try going north instead. Understand "door" as the gate.

After opening the gate:
say "You shouldn't be able to

VISUAL

PATCH AND FLOW CONTROL PROGRAMMING



```
#include <conio.h>
#include <string.h>

#define FULLSCREEN 512
char buf[FULLSCREEN+1];

#include "col
#include "joy

char * patc="

char move(char
char k;
if(Up[now
0 page
while(k=j
// if (n
if(k=
if(k=
if(k=
if(k=
if(k=J
if(k=J
if(k=J
if(k=J
}

void cls(){
waitvblank();
clrscr();
gotoxy(0,0);
}
```

SOURCE

sequence

literacy

write

FLOW

space

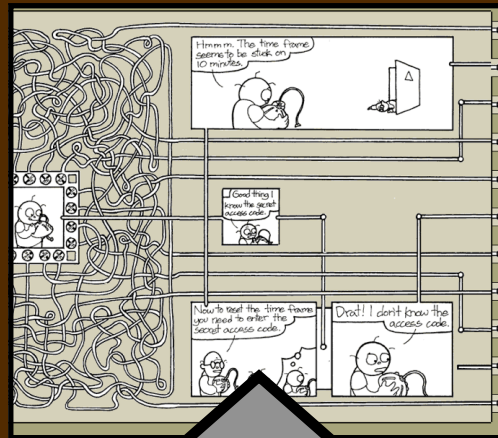
visual literacy

map / graph

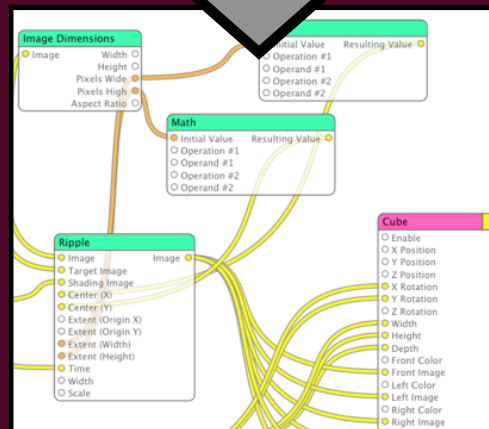
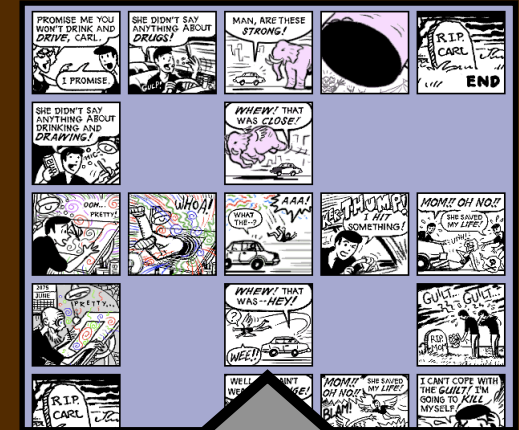
FLOW ART

VISUAL PROGRAMMING

PATCH



GRID



PATCH

GRID



PATCH

GRID

objects

directions

sequences

axes

relationships

spatial relations

there is no one type of
code to criticize

every logic of specification
is its own
paradigm for critique

Jeremy Douglass

Postdoctoral Researcher
Software Studies Initiative
University of California San Diego