



Adobe

Scrambled Eggs

Roy T. Fielding, Ph.D. | Senior Principal Scientist, Adobe





IT'S A MYSTERY

ε | VOLVE '13



IT'S A MYSTERY

Open Source

Apache



IT'S A MYSTERY
Open Source
Apache
Standards
HTTPbis + HTTP/2
Do Not Track



IT'S A MYSTERY

Open Source
Apache

Standards

HTTPbis + HTTP/2
Do Not Track

Software
Architecture

REST



IT'S A MYSTERY

Open Source
Apache

Standards

HTTPbis + HTTP/2

Do Not Track

Software
Architecture

REST





IT'S A MYSTERY

Open Source
Apache

Standards

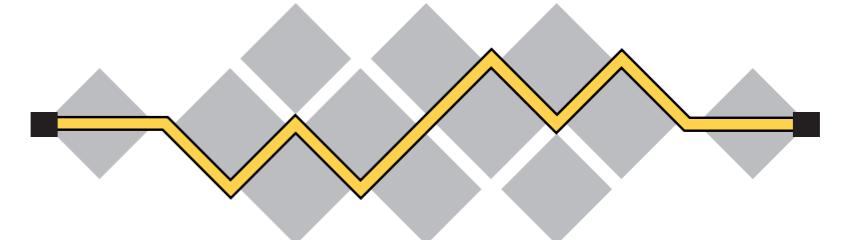
HTTPbis + HTTP/2
Do Not Track

Software
Architecture

REST

Wouldn't it be nice if
CQ supported ...

STANDARDS: HTTP



HTTP/1.1

Almost done, really

IESG last call next week or so

HTTP/2.0

Standardization of Google's SPDY

Session-layer Tunnel for HTTP with Compression (?)

- Should have been called TCP++, or TLS++

Improvement for session-heavy, authenticated sites

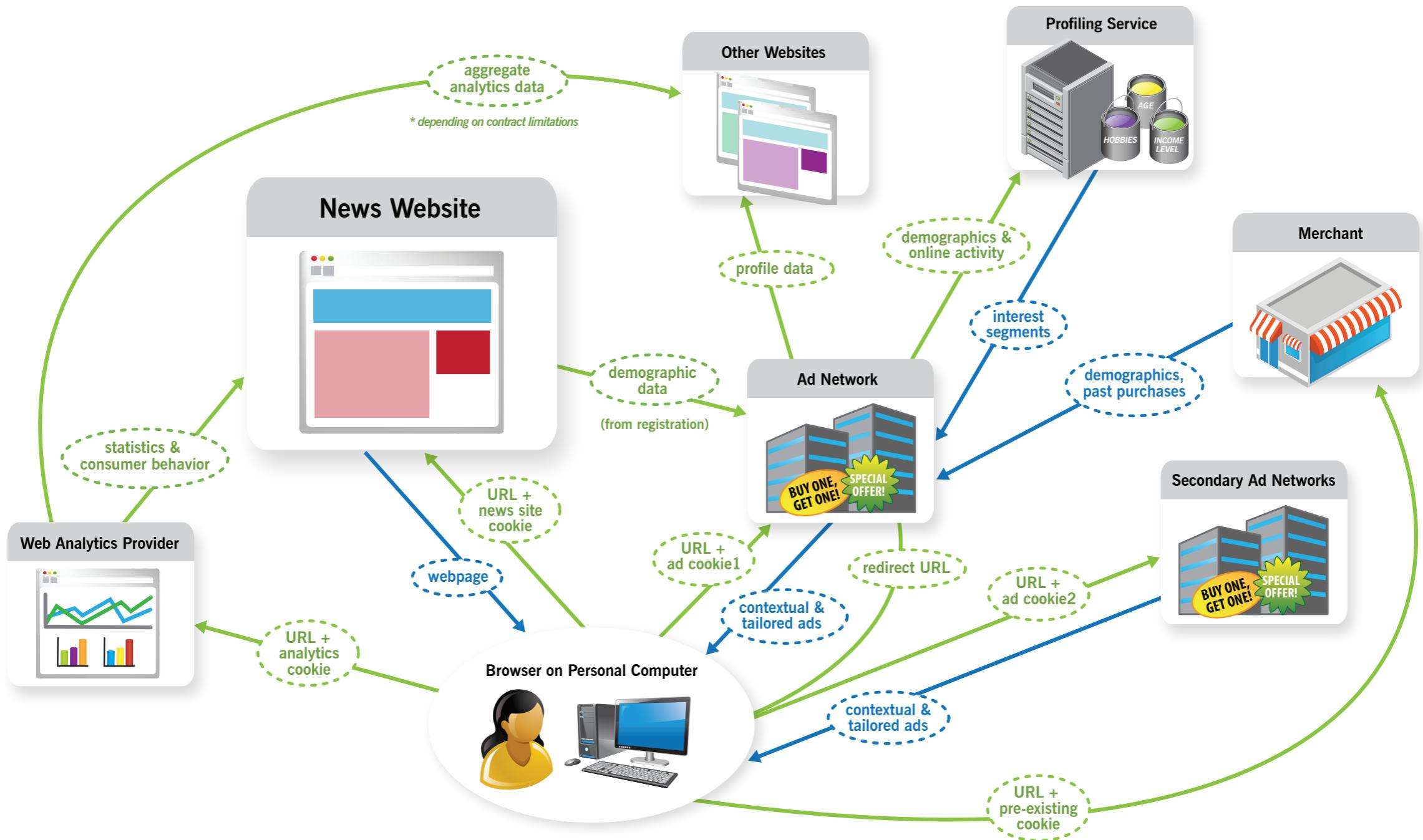
- <http://trac.tools.ietf.org/wg/httpbis/trac/wiki>

STANDARDS: DO NOT TRACK

**Tracking is pervasive
because it feeds personalization and UX**



Online Behavioral Advertising



● Green arrows represent the flow of **consumer data**.

● Blue arrows represent the flow of **3rd party information & offerings**.

* This diagram represents a typical flow of information related to some online behavioral advertising.
Not all online behavioral advertising operates exactly like this diagram.

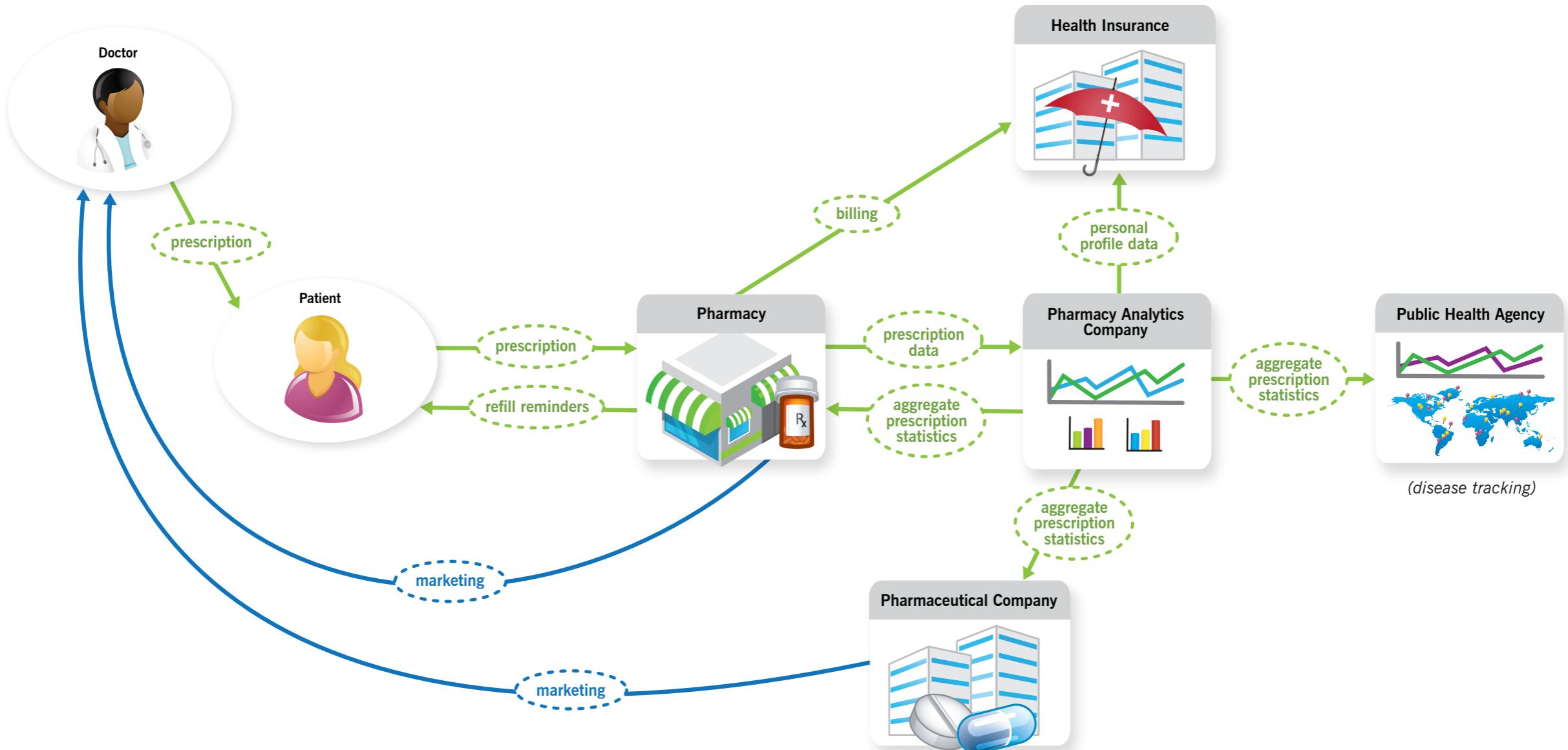
STANDARDS: DO NOT TRACK

Tracking is pervasive
because it feeds personalization and UX

**Data collection across unrelated contexts
is a privacy concern**



Medical Prescriptions



Green arrows represent the flow of **consumer data**.

Blue arrows represent the flow of **3rd party information & offerings**.

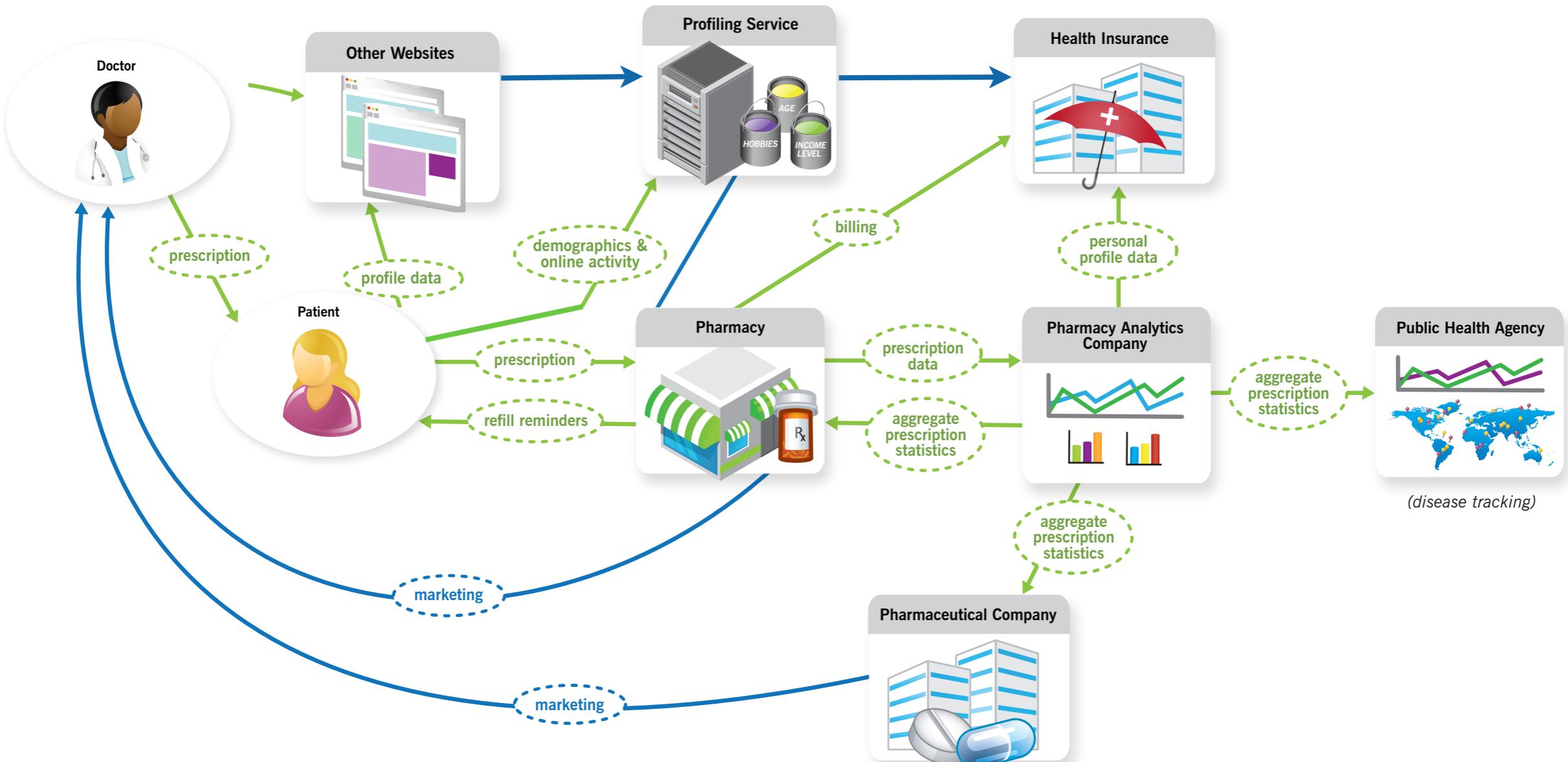
* This diagram represents a typical flow of information related to some medical and pharmaceutical companies. Not all medical and pharmaceutical companies operate exactly like this diagram.



[FTC]

ε | VOLVE '13

Medical Prescriptions



● Green arrows represent the flow of **consumer data**.

● Blue arrows represent the flow of **3rd party information & offerings**.

* This diagram represents a typical flow of information related to some medical and pharmaceutical companies. Not all medical and pharmaceutical companies operate exactly like this diagram.



[FTC]

ε | VOLVE '13

STANDARDS: DO NOT TRACK

Tracking is pervasive
because it feeds personalization and UX

Data collection across unrelated contexts
is a privacy concern

Governments want to stop it,
but they don't know how

Privacy advocates incite fear and doubt

Poor business data practices justify them!

This should be
an easy problem to fix



BUT ONLY IF WE WORK TOGETHER



... and now state governments are getting involved too ...

SOFTWARE ARCHITECTURE

What is
the best practice for
versioning
a REST API?



SERIOUSLY, WHICH IS BETTER?

Should I include a version number in the URL hierarchy?

- `http://example.com/v1/users`

Should I include a version number on the resource name?

- `http://example.com/users.v1`

Should I include a version number as a query parameter?

- `http://example.com/users?api=v1`

Should I include a version number in the media type?

- `Content-Type: application/vnd.myname.v1+json`

REST

REST

is software engineering on the scale of

DECADES

REST

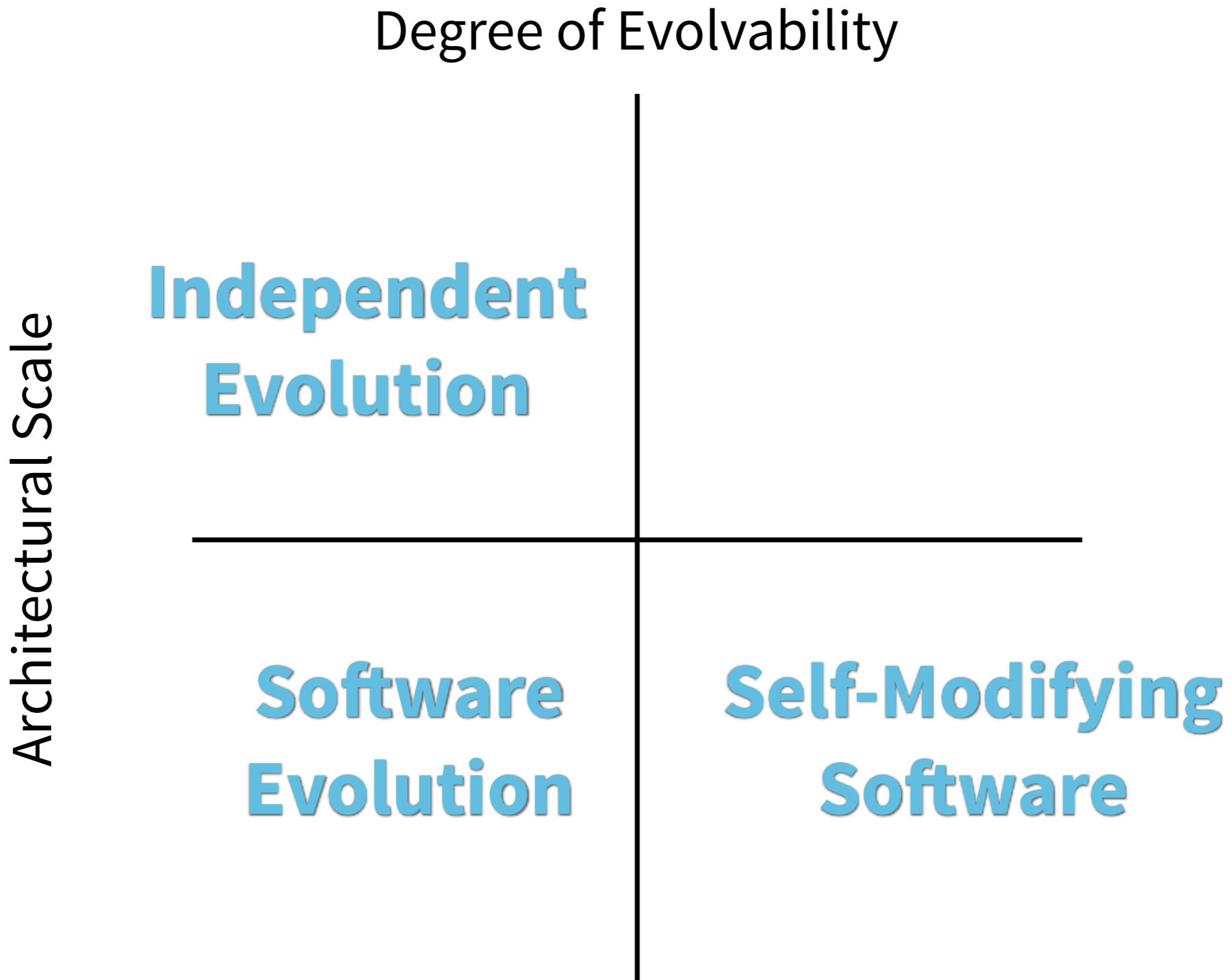
REST

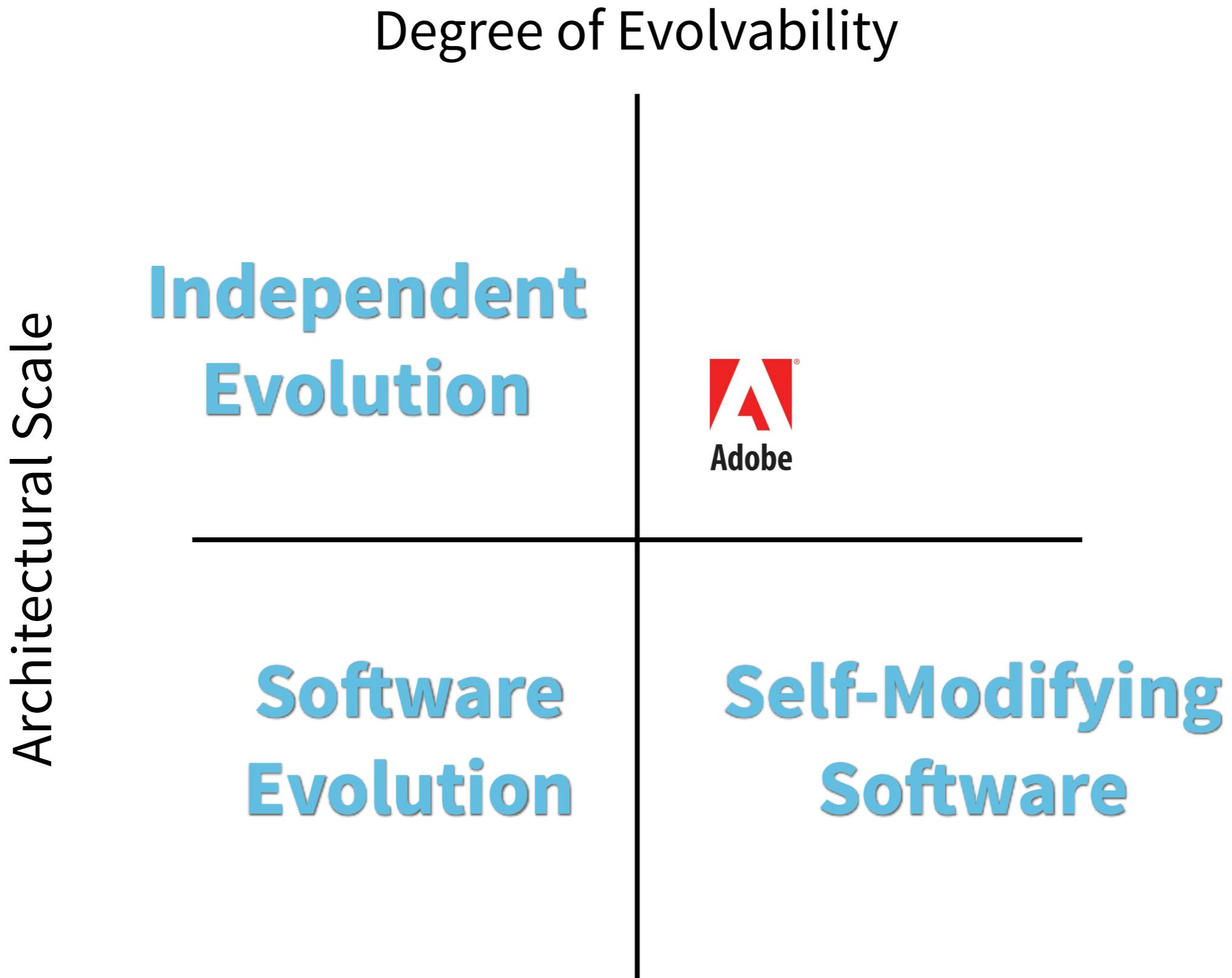
is designed primarily to improve

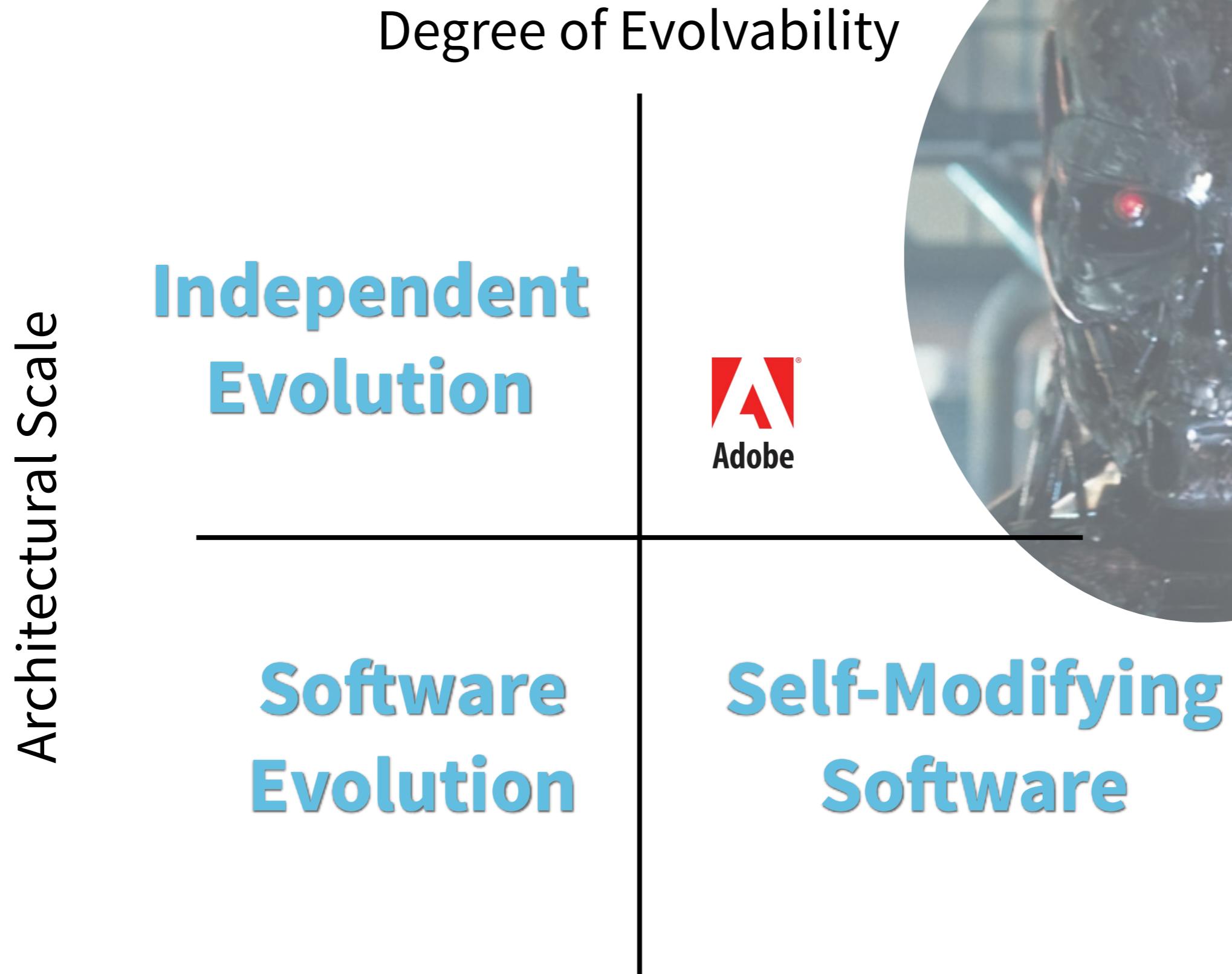
EVOLVABILITY

EVOLVABILITY

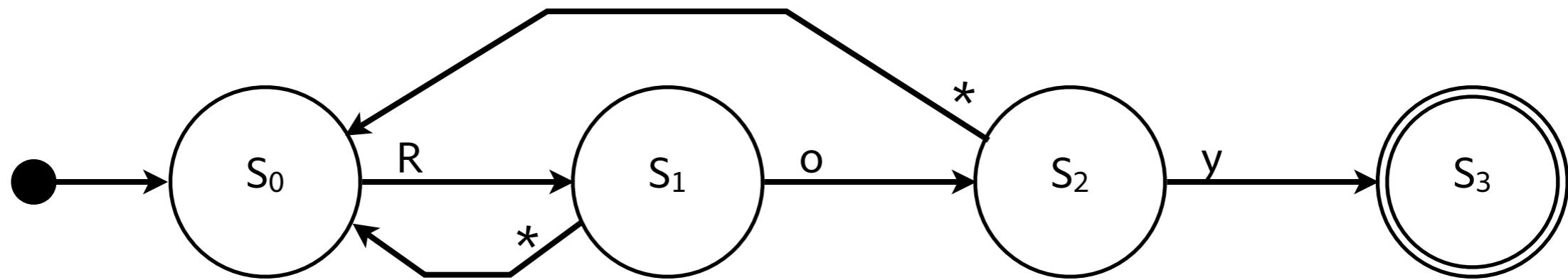
Evolvability is the **ability to change** over time,
in response to changing user needs
or a changing environment,
without starting over



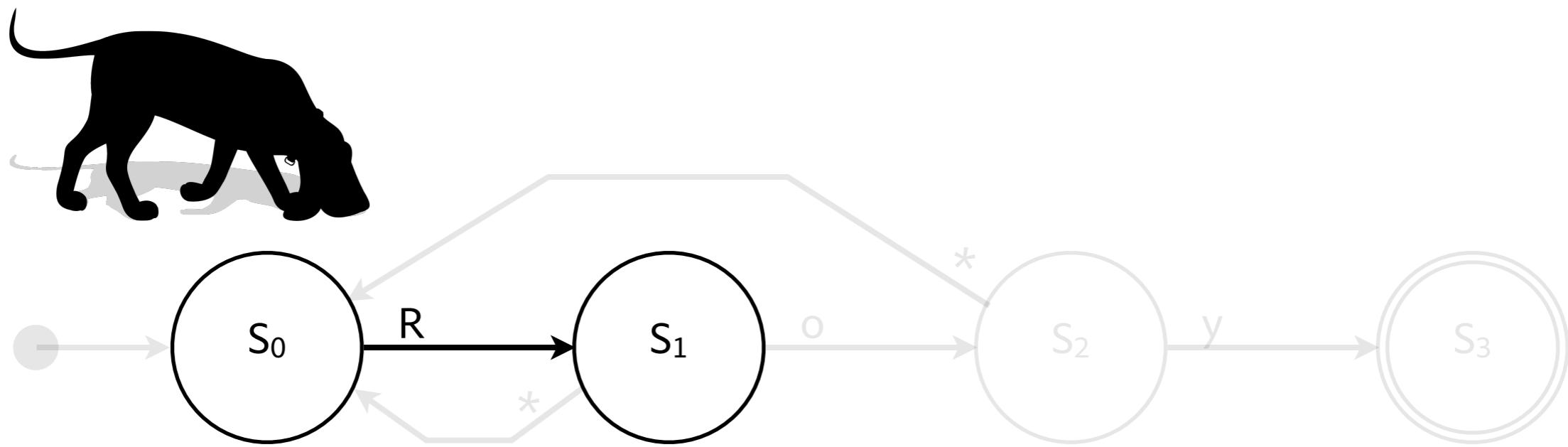




Hypertext as the Engine of Application State

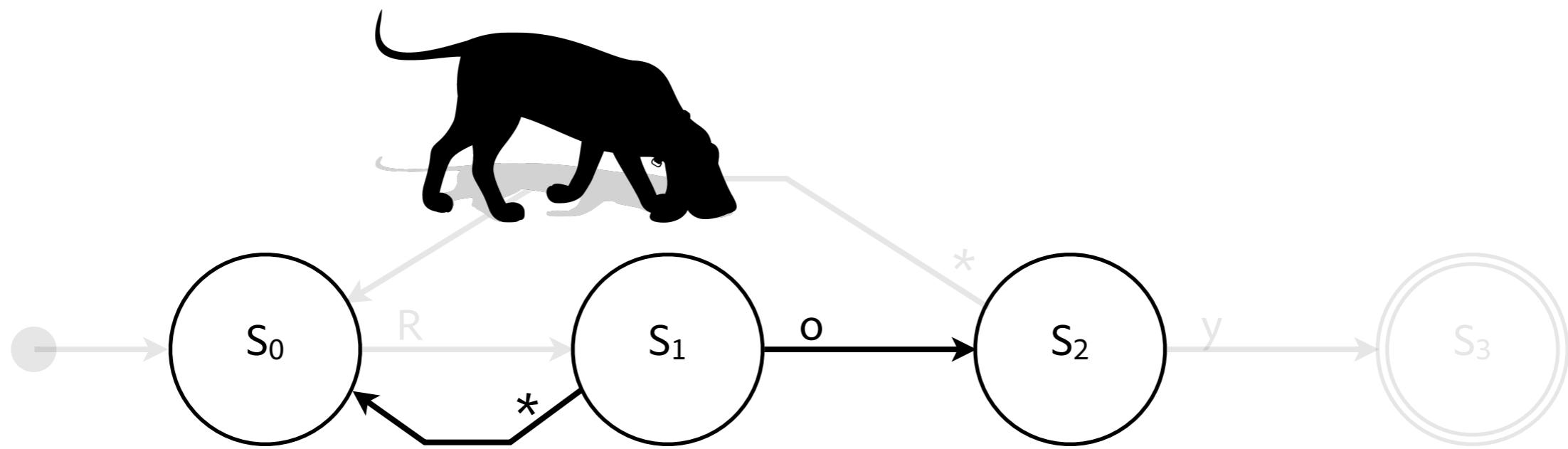


REST



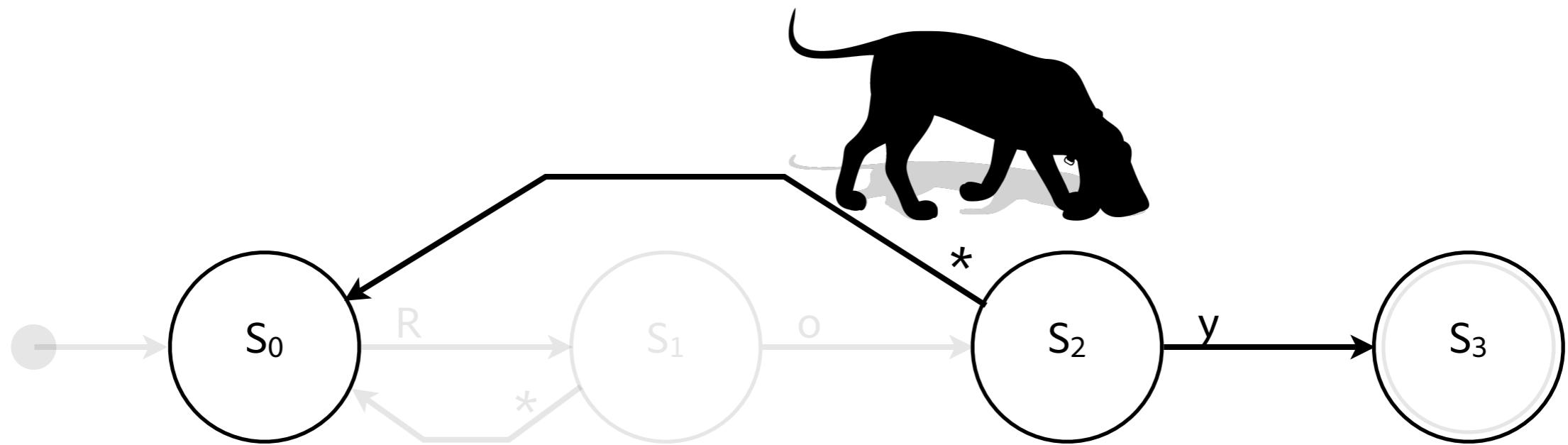
Follow Your Nose

REST



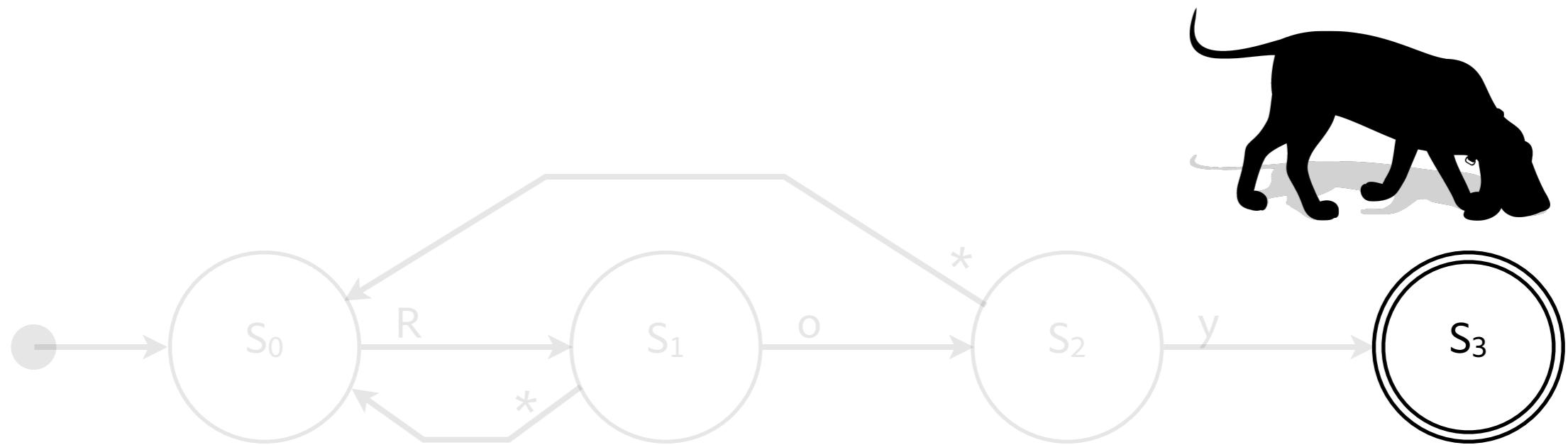
Follow Your Nose

REST



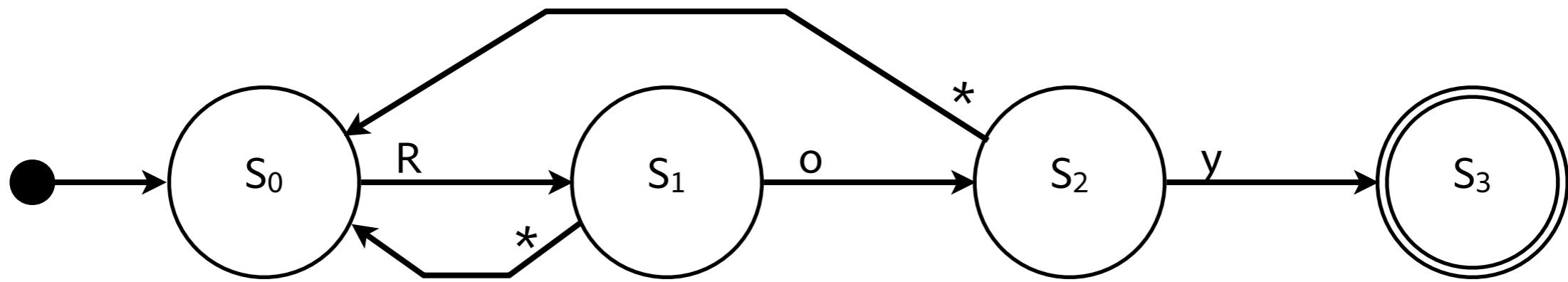
Follow Your Nose

REST



Follow Your Nose

Hypertext as the Engine of Application State



each state can be dynamic
each transition can be redirected

What is
the best practice for
versioning
a REST API?



What is
the best practice for
versioning
a REST API?

DON'T

Versioning an interface
is just a “polite” way
to kill deployed applications

SERIOUSLY?

SERIOUSLY?

When was the last time you saw
a version number on a website?

SERIOUSLY?

When was the last time you saw
a version number on a website?

a REST API is just a website
for users with a limited vocabulary
(machine to machine interaction)

BREAKING COMPATIBILITY IS BAD

Websites are supposed to retain
backwards compatibility
(avoid broken links)

If you want to break with the past,
use a different hostname,
with new branding!

ALWAYS SHIP TRUNK

why web applications are different ...

problems that none of the
revision control systems solve ...

how you can solve some of them yourself

[Paul Hammond (Typekit)] <http://www.paulhammond.org/2010/06/trunk/>
based on real deployment experience at Flickr, Etsy, Typekit, ...

ALWAYS SHIP TRUNK

ALWAYS SHIP TRUNK

What would a revision control system built for supporting deployed web applications be like?

ALWAYS SHIP TRUNK

What would a revision control system built for supporting deployed web applications be like?

CQ5



right?

ALWAYS SHIP TRUNK

What would a revision control system built for supporting deployed web applications be like?

CQ5



right?

Wouldn't it be nice if CQ supported ...

Feature Flags

for conditional activation
of content fragments during
continuous deployment

GLOBAL CONFIGURATION SETTINGS

```
if (frags("saml_auth")) {  
    credentials = saml.authenticate(user);  
}  
else {  
    credentials = httpAuth.check(user);  
}
```

- **testable for conditional content (i.e., everything)**
- **readable via all development interfaces**
- **writable with ops authority**

FLIP SWITCHES VIA CONSOLE

Feature Flip

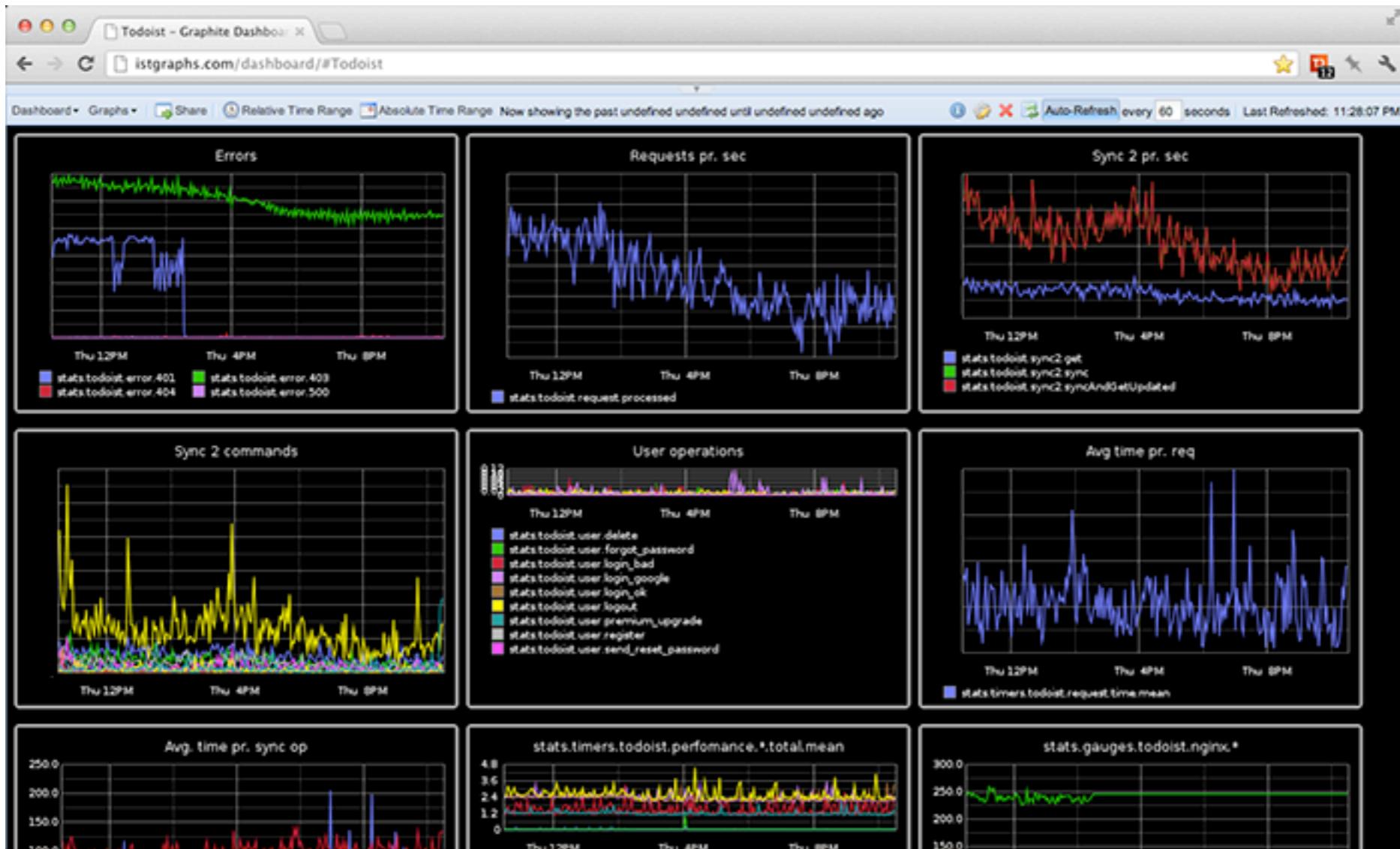
You can temporarily turn all these features off: [IGNORE ALL](#)

url param	config name	kind	status	change?
explore_search	\$cfg[disable_feature_explore_search]	feature_flag	ENABLED FOR THIS HOST	-
	Turns on the new full width search page.			
Unicorn Polo	\$cfg[enable_unicorn_polo]	feature_flag	ENABLED	turn off
	Very secret; ask Dunstan.			
Hot Tub Time Machine	\$cfg[enable_hot_tub]	feature_flag	DISABLED	turn off
	Not as fun as it sounds.			

[Ross Harmes, <http://www.flickr.com/photos/rossharmes/4153769740/>]

- **easy UI for (proportional) enabling or disabling of frags**
- **activation by frag, recorded with timestamps**
- **activation by AB testing, recorded for comparative analytics**

ACTIVATION-AWARE MONITORING



- **dashboard interfaces for time-series graphs with activation markers**
- **AB comparisons based on automated percentile activation**
- **data available as more resources, for reuse by enterprise monitoring**

FRAGS

Are you a CQ/AEM customer/prospect?

Do you want a feature like Frags?

Do you need more than what I described?

Ask for it!

(we prioritize features by customer demand)

