

# Embassies: Radically refactoring the web

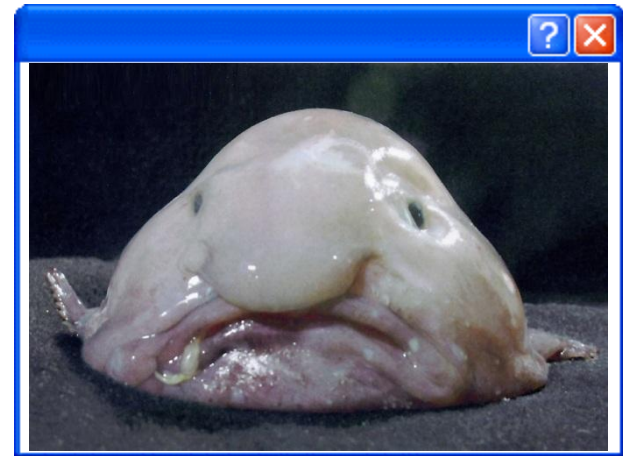
John R. Douceur

Jon Howell

Bryan Parno

*Microsoft Research*

# promise of the web model



# the web is quite vulnerable



Buffer overflows  
JavaScript API vulnerabilities  
XSS  
CSRF  
Session fixation  
clickjacking



# safe web-surfing hygiene?



“don't click dangerous links!”

# the problem

Security weaknesses in the web API

- complex execution semantics
- subtle communication & sharing semantics
- communication implicit in execution

cannot be fixed with a better browser for the same API

# this talk

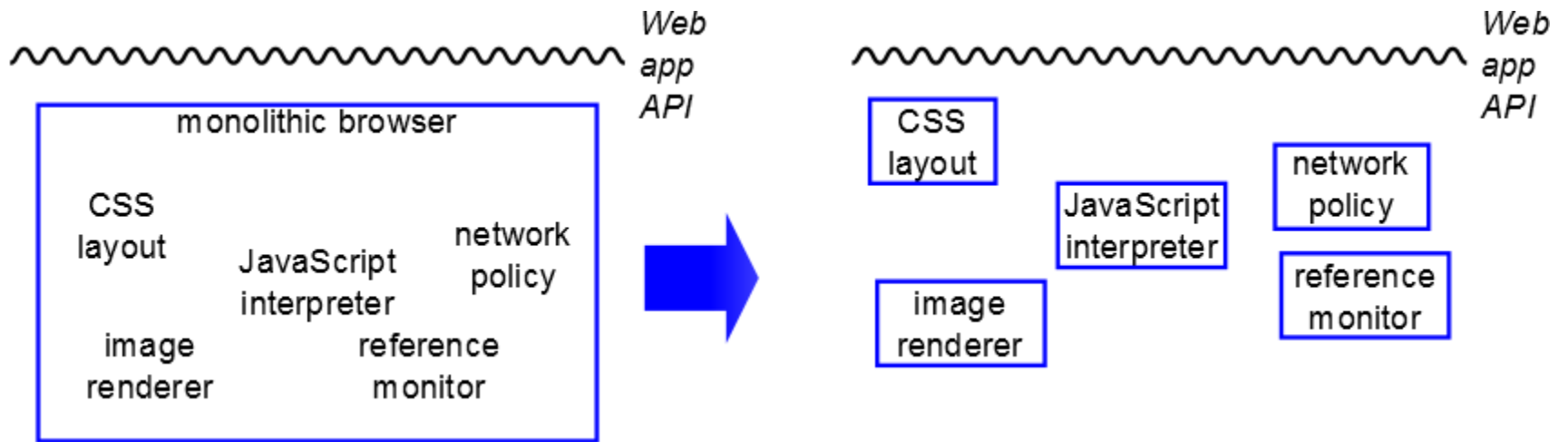
The current API is broken due to conflicting goals

Propose a new API for the web

- simple execution semantics: binary code
- explicit communication semantics: IP
- supports existing web apps and beyond

Argue that the new API evolves safely

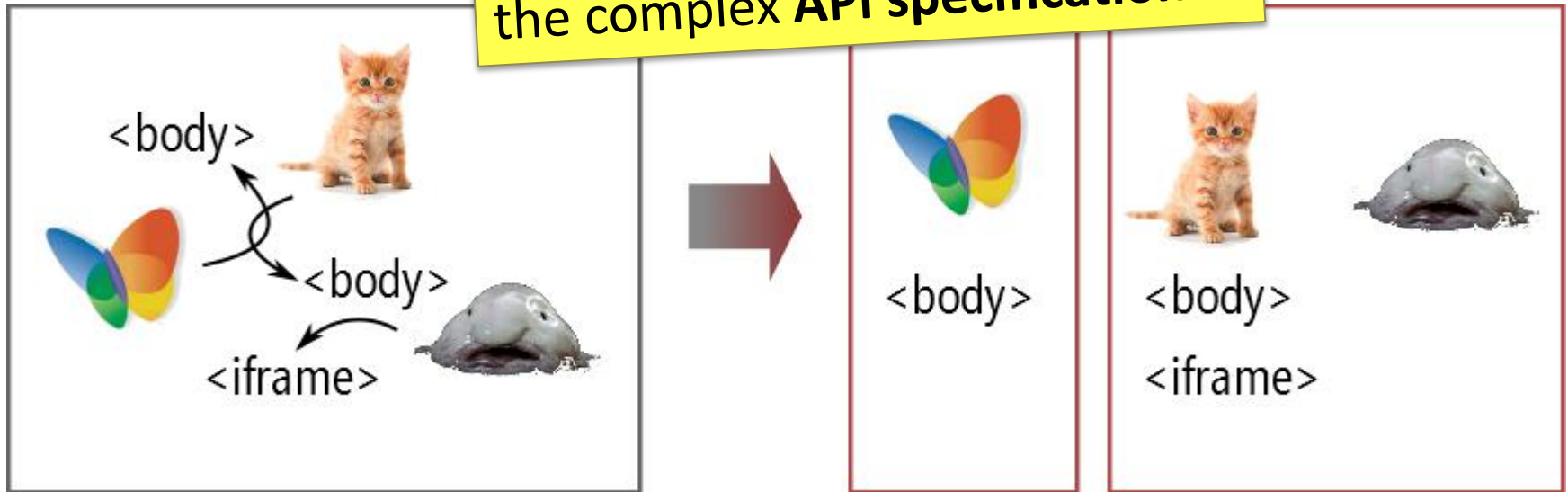
# refactoring the browser isn't enough



[OP, IBOS]

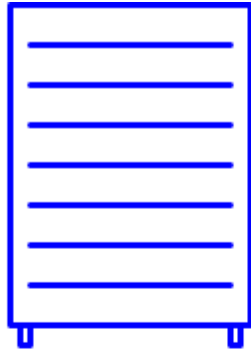
# refactoring the browser isn't enough

the **browser implementation** doesn't matter until we fix the complex **API specification**



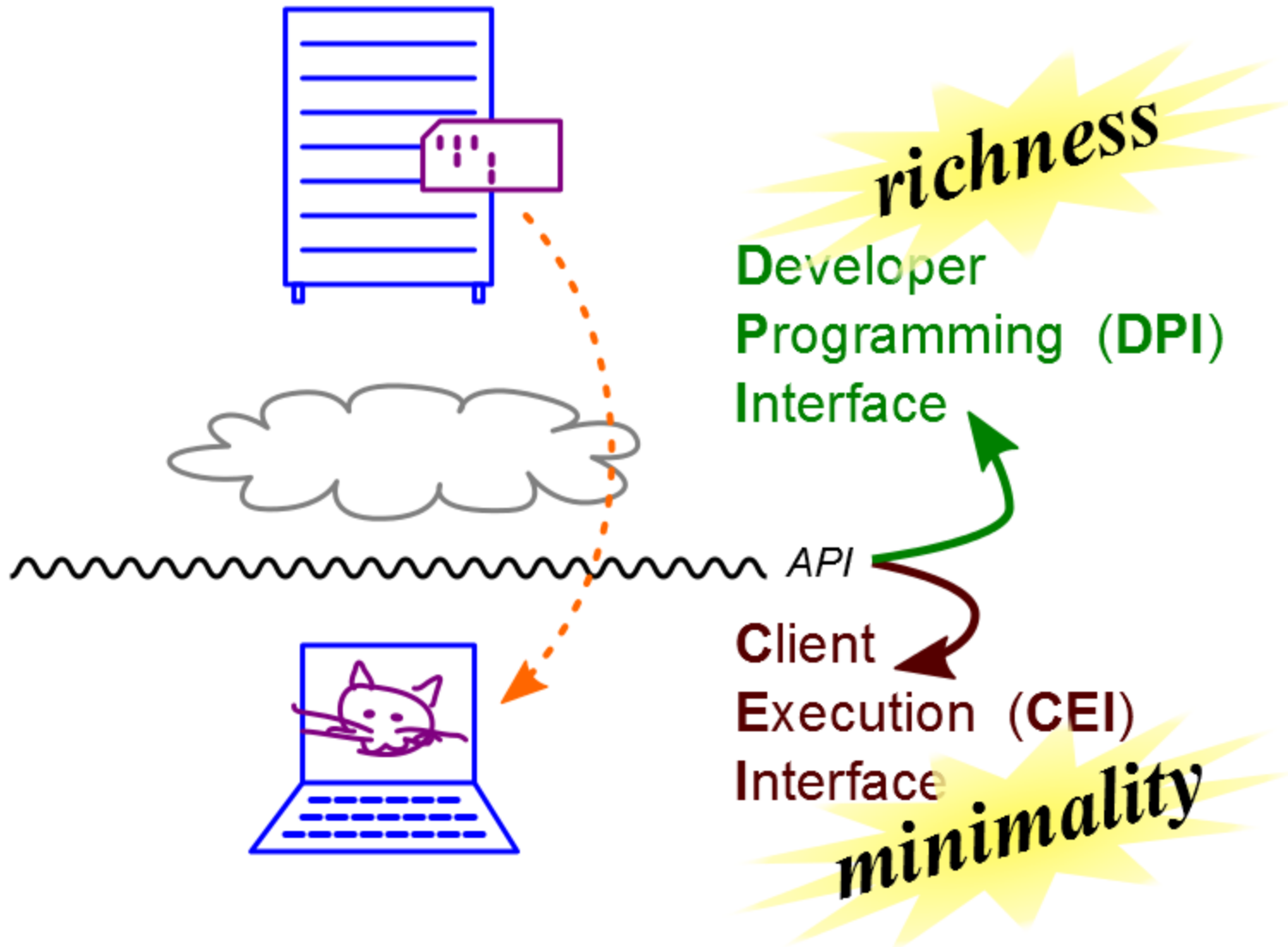
[Gazelle, Chrome]





HTTP MIME HTML PNG JPG CSS DOM JavaScript  
API





# separate DPI from CEI

*JavaScript - CSS - HTML*  
existing web **API**

```
document  
.getElementById("txt")  
.style.height="100px"
```

*JavaScript - CSS - HTML*

Developer Programming Ifc (**DPI**)

```
document  
.getElementById("txt")  
.style.height="100px"
```

*simple, low-level, well-defined*

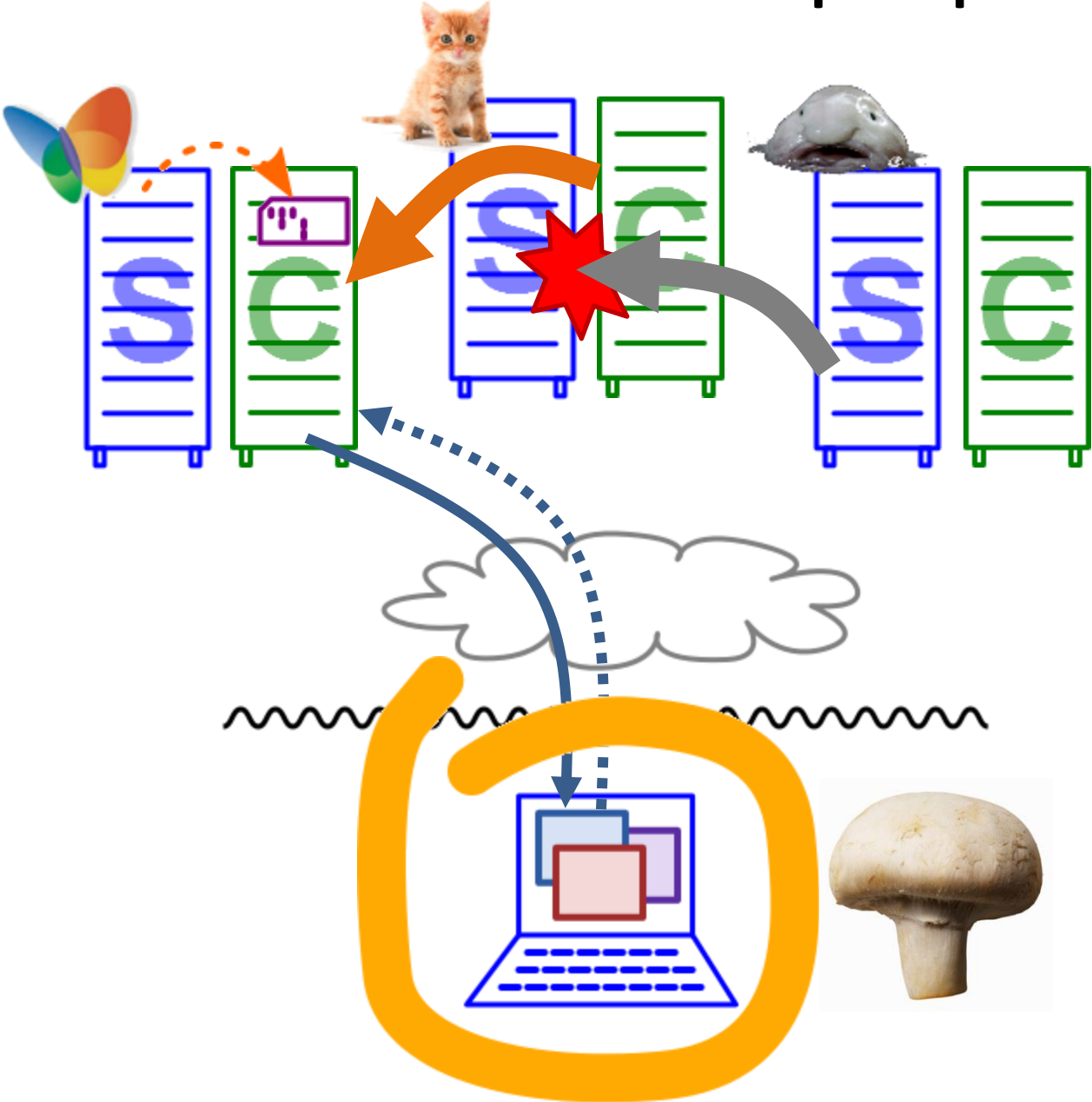
Client Execution Interface (**CEI**)

```
10001010111011001  
01010110111010100  
00000110100100011
```

why is this model different?



# a ridiculous straw-proposal



# confounded by reality



Network reliability



High bandwidth

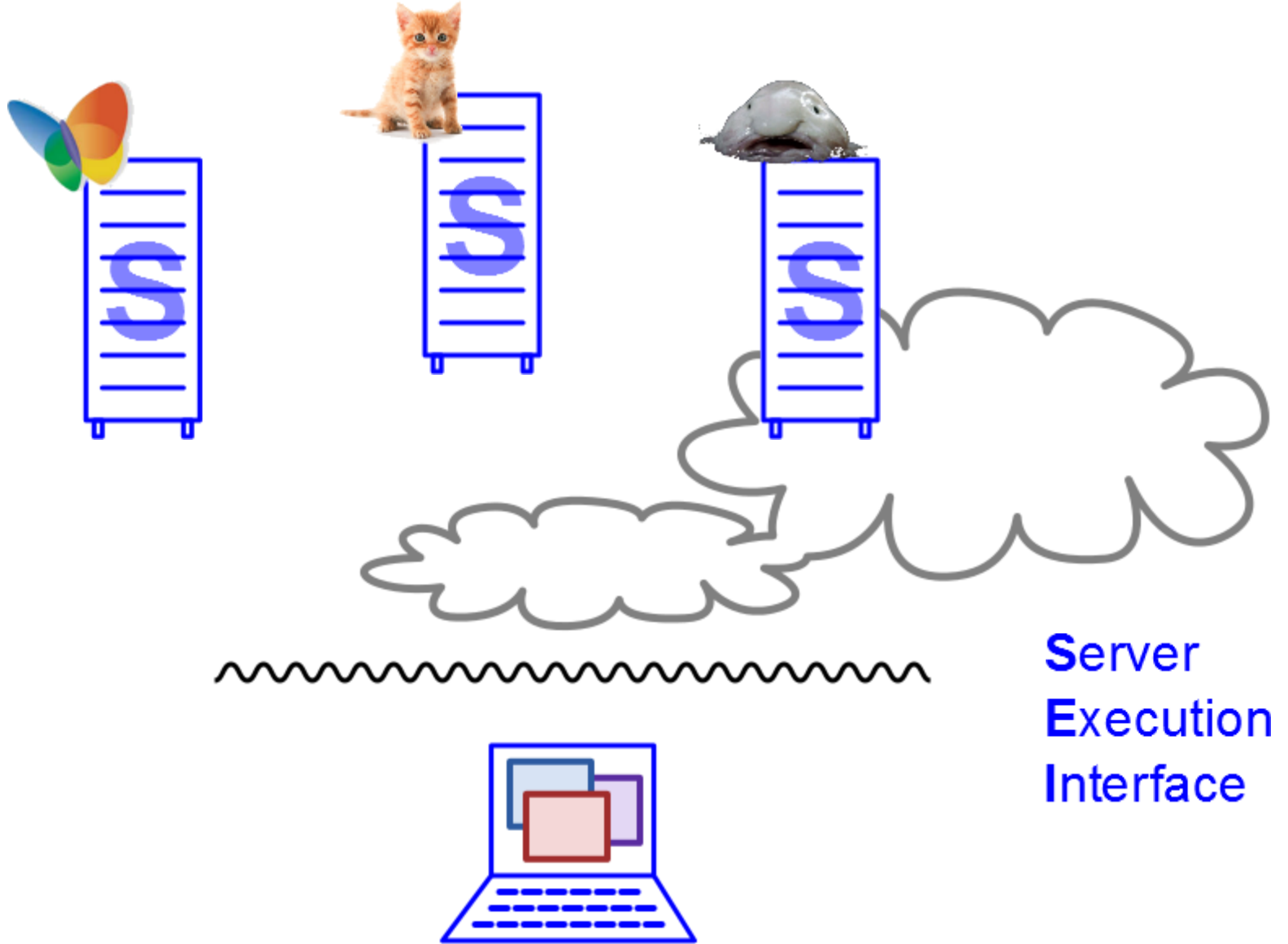


Low latency

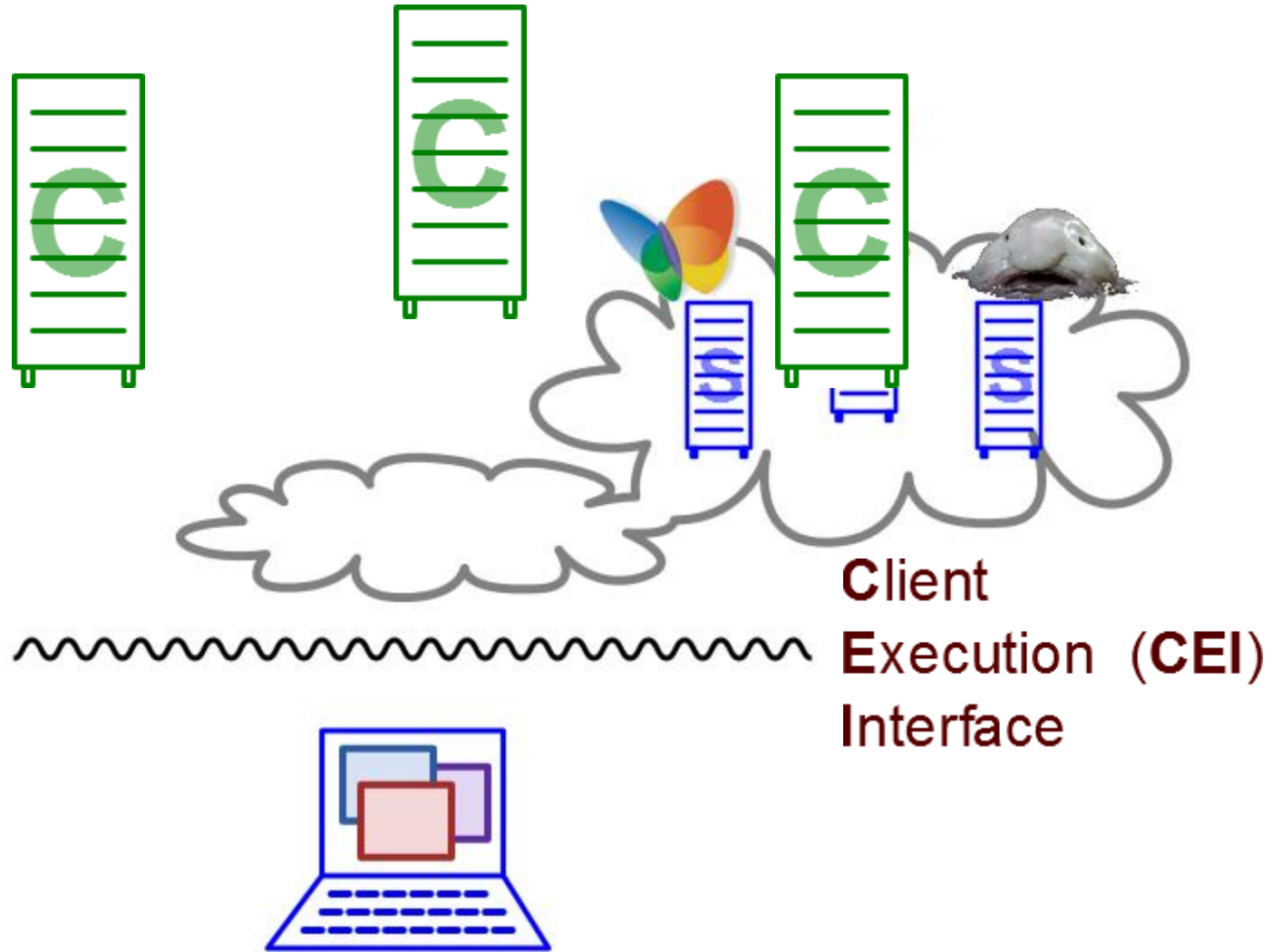


Ample server resources

# the multitenant datacenter



# the client pico-datacenter





# the entire Embassies CEI

## execution: binary code

alloc\_mem, free\_mem  
thread\_create, thread\_exit  
x86\_set\_segments, exit  
ensure\_alive  
futex\_wait, futex\_wake, get\_alarms,  
set\_clock\_alarm, get\_time

## communication: IP packets

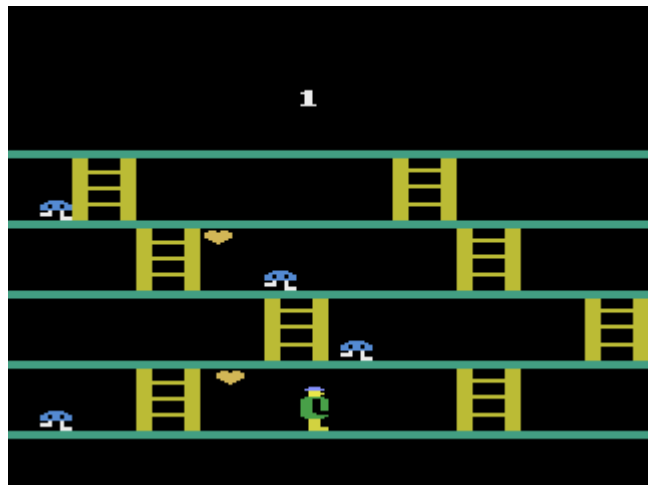
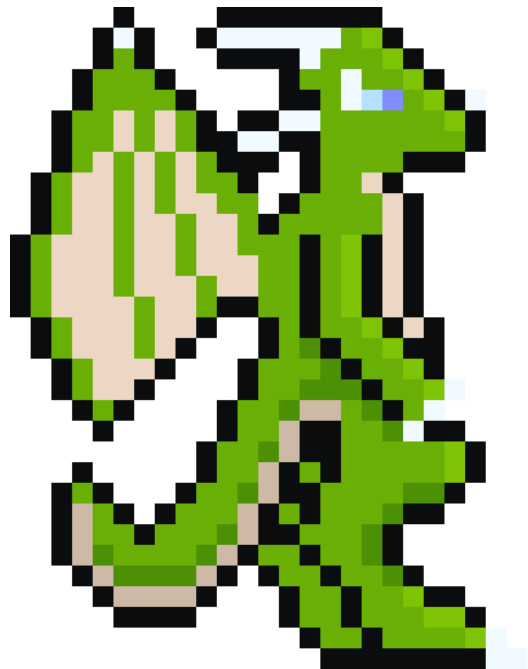
alloc\_buffer, send\_buffer  
receive\_buffer, free\_buffer

## user interface: pixels and clicks

sublet\_viewport, repossess\_viewport  
get\_deed\_key  
accept\_viewport, transfer\_viewport  
map\_canvas, unmap\_canvas, update\_canvas  
receive\_ui\_event

## privacy & integrity primitives:

get\_random, get\_app\_secret  
endorse\_me, verify\_label



# Developer Programming (DPI) Interface

The screenshot shows a web browser window with the address bar containing 'craigslist'. The page displays the Craigslist logo, navigation links like 'post to classifieds' and 'my account', a search bar, and a calendar for the month of August. A yellow callout box with the text 'binary code enables rich apps' is overlaid on the browser window.

The screenshot shows the GNU Image Manipulation Program (GIMP) interface. A 'Change Foreground Color' dialog box is open, displaying a color wheel and various color selection tools. The background shows a drawing of a colorful bird.

The screenshot shows a vector graphics application interface. It features a yellow star and a cyan triangle with the text 'Illustrations R Us' overlaid on them. The interface includes various toolbars and a zoom level of 42%.

binary code enables rich apps

The screenshot shows a globe application interface. It displays a 3D map of the world with labels for continents like North America, Europe, and South America. The interface includes a search bar, a legend, and a map view.

The screenshot shows a game interface with the word 'HYPEROID' displayed in a stylized, glowing font. The background is dark with some geometric shapes and a small character.



# challenge: cross-app interactions

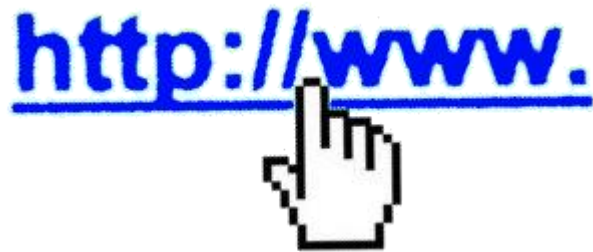
First name:

Last name:

email:

Male

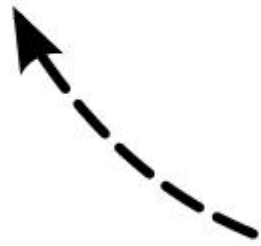
Female



# interaction: **today's** form submission



**implicit ambient authority**



First name:

Last name:

email:

Male

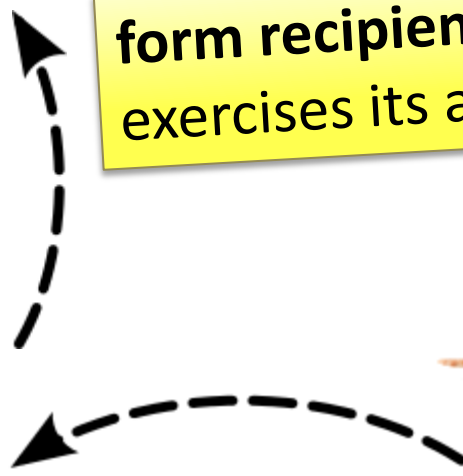
Female



# interaction: **Embassies** form submission



**form recipient** explicitly  
exercises its authority



# interaction: **today's** link coloring



[kittens.com/siamese](http://kittens.com/siamese)

[kittens.com/tabbies](http://kittens.com/tabbies)

[kittens.com/calico](http://kittens.com/calico)

[kittens.com/persian](http://kittens.com/persian)



[kittens.com/siamese](http://kittens.com/siamese)

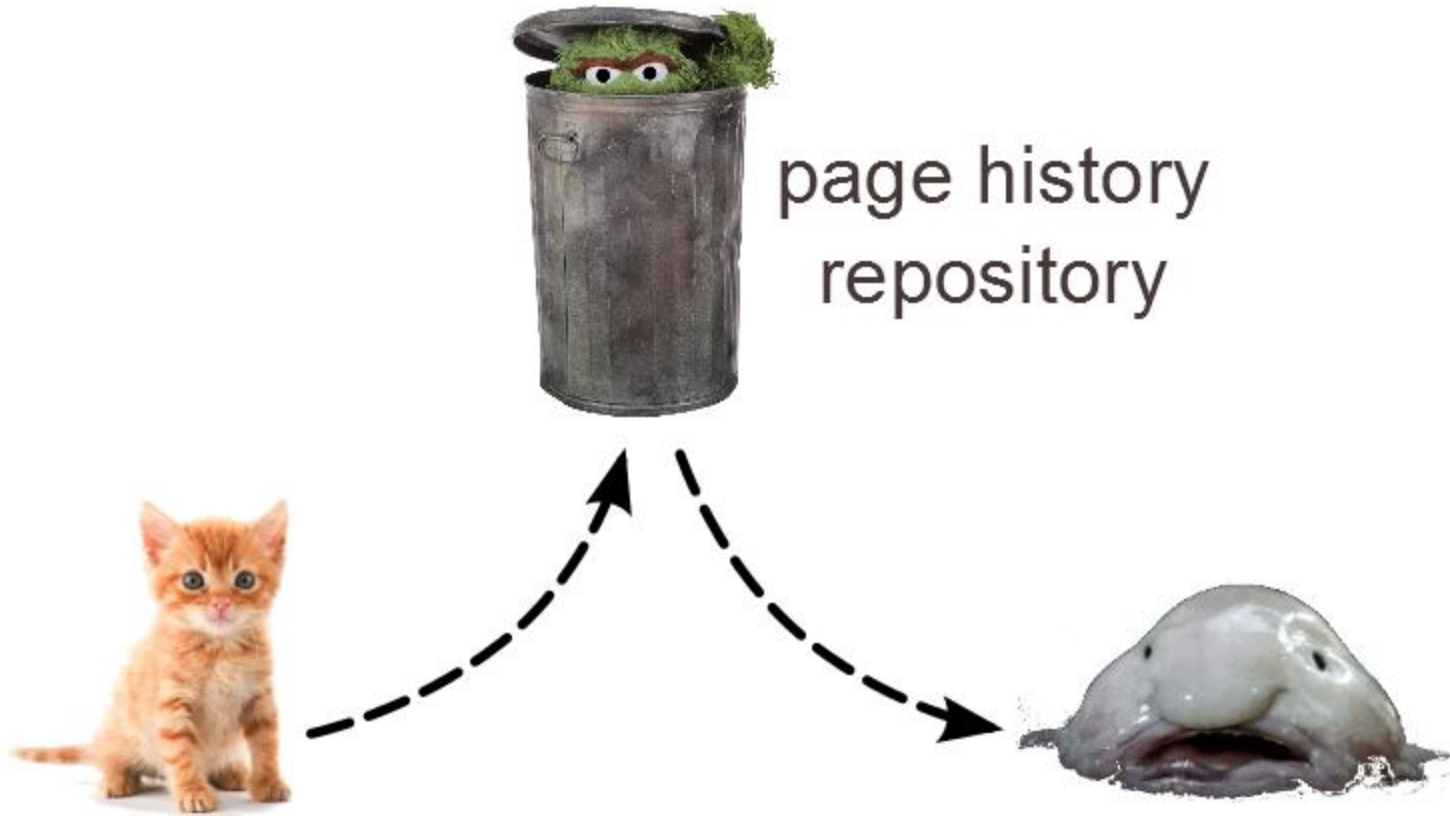
[kittens.com/tabbies](http://kittens.com/tabbies)

[kittens.com/calico](http://kittens.com/calico)

[kittens.com/persian](http://kittens.com/persian)

**implicit history leaks**

# interaction: **today's** link coloring



[kittens.com/tabbies](http://kittens.com/tabbies)



# interaction: **Embassies** link coloring

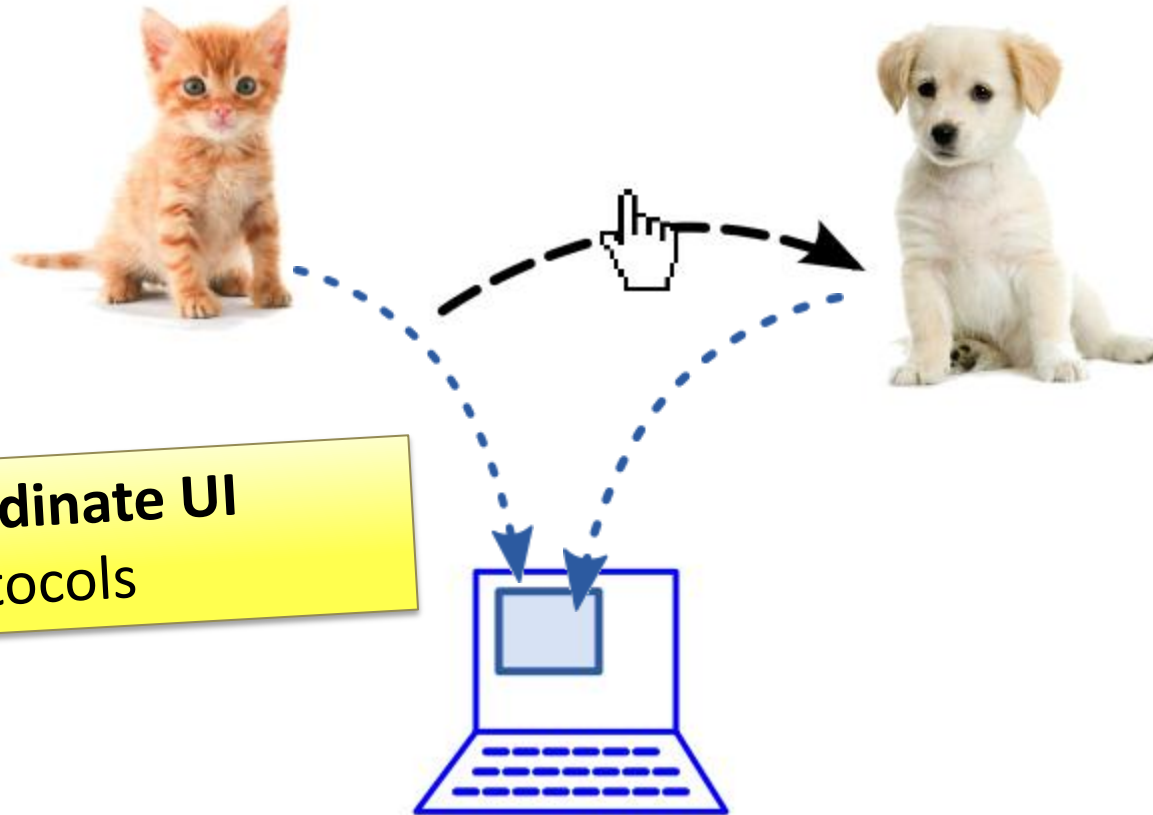


# interaction: **today's** page navigation



existing API requires  
browser to mediate  
navigation

# interaction: **Embassies** page navigation

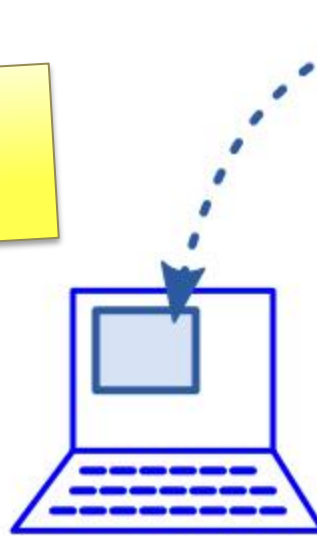


apps coordinate UI  
using protocols

# interaction: **Embassies** page navigation



**protocols** make it clear how leaky an interaction is.



# challenge: app launch performance



Linux  
Apache  
MySQL  
Perl



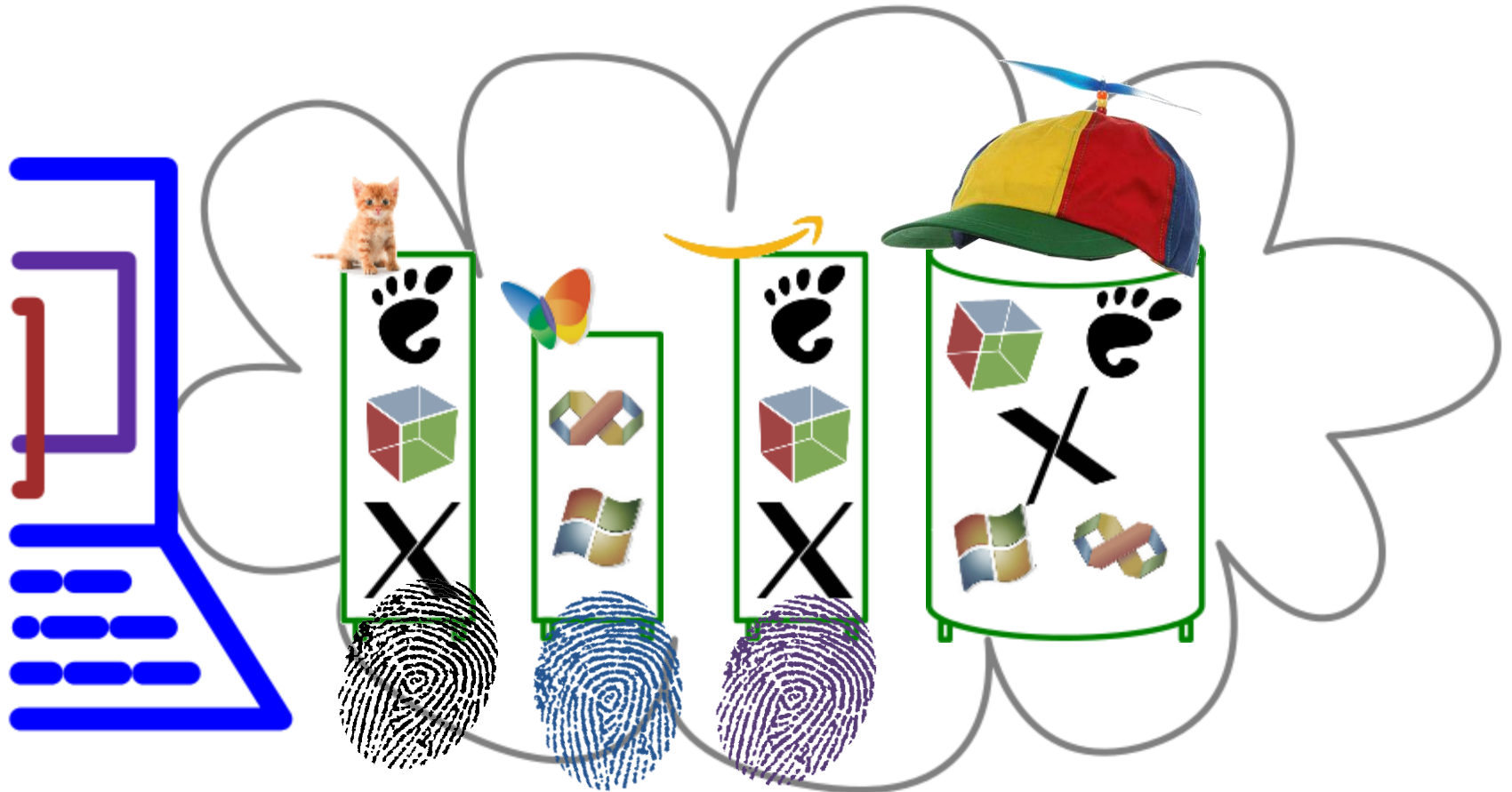
vendor control  
doesn't require  
unbounded divergence



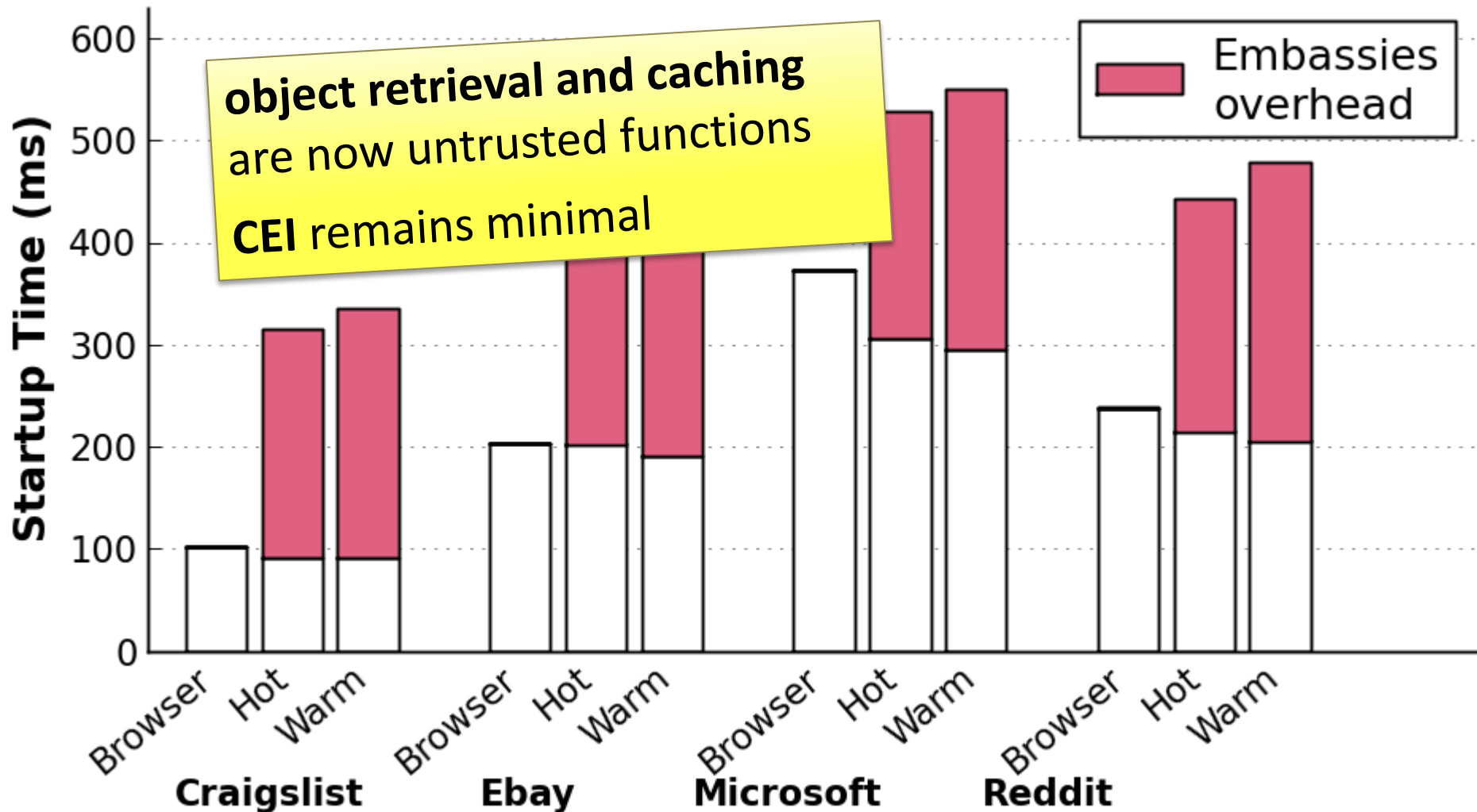
caching is effective



solution: untrusted cache



# startup caching is effective



# isn't 200 ms a lot?



we're only adding it when the user crosses over to a new site.

within a site, vendors can go *faster*:  
SPDY++?



we're loading unoptimized WebKit



this **modest performance problem**  
resolves a **bucket of security problems**



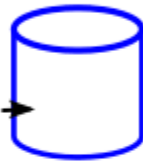
# fixing flaws: history leaks



[kittens.com](http://kittens.com) rabbits



# fixing flaws: cross-site scripting (XSS)



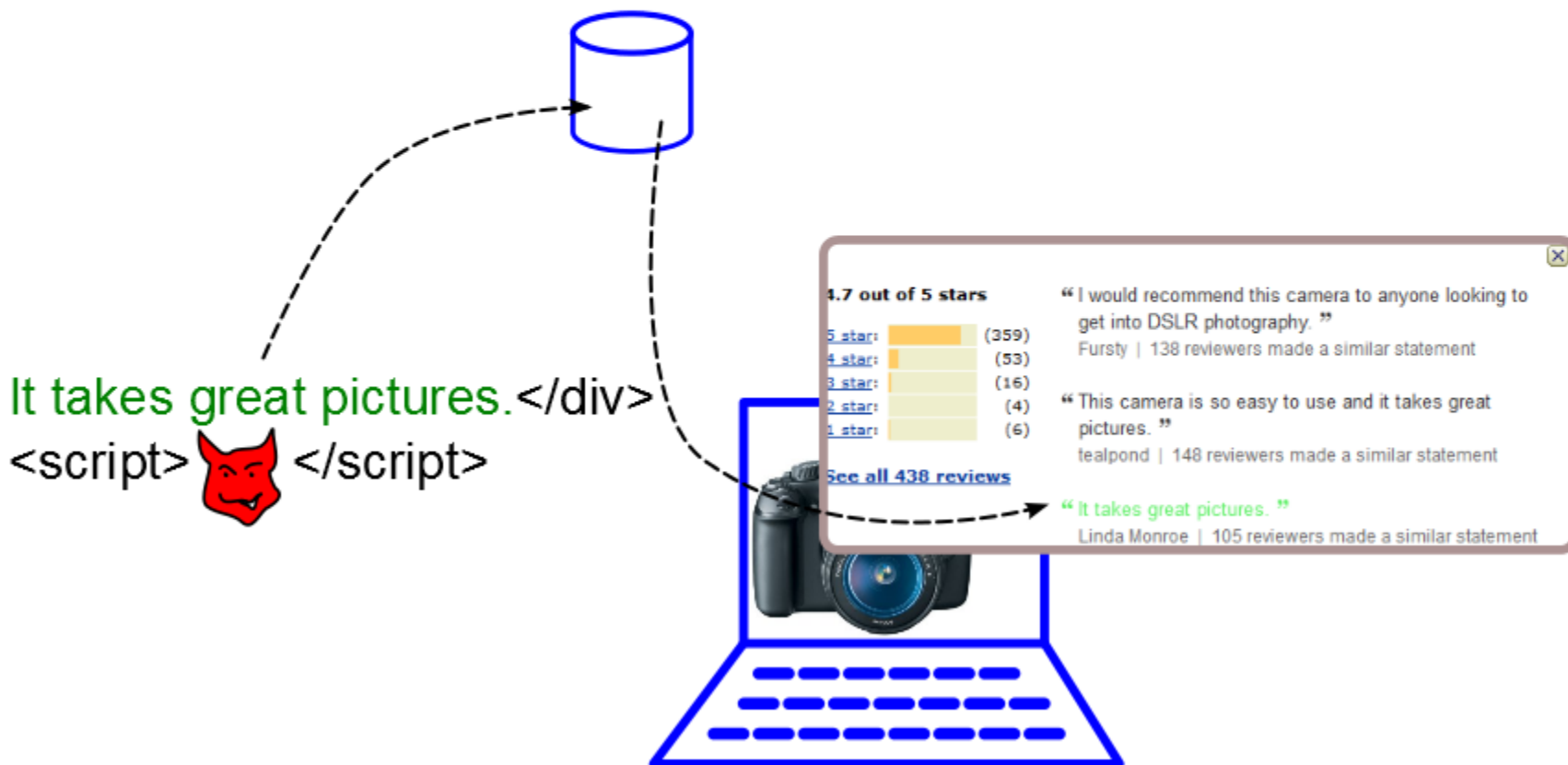
It takes great pictures.</div>

<script>



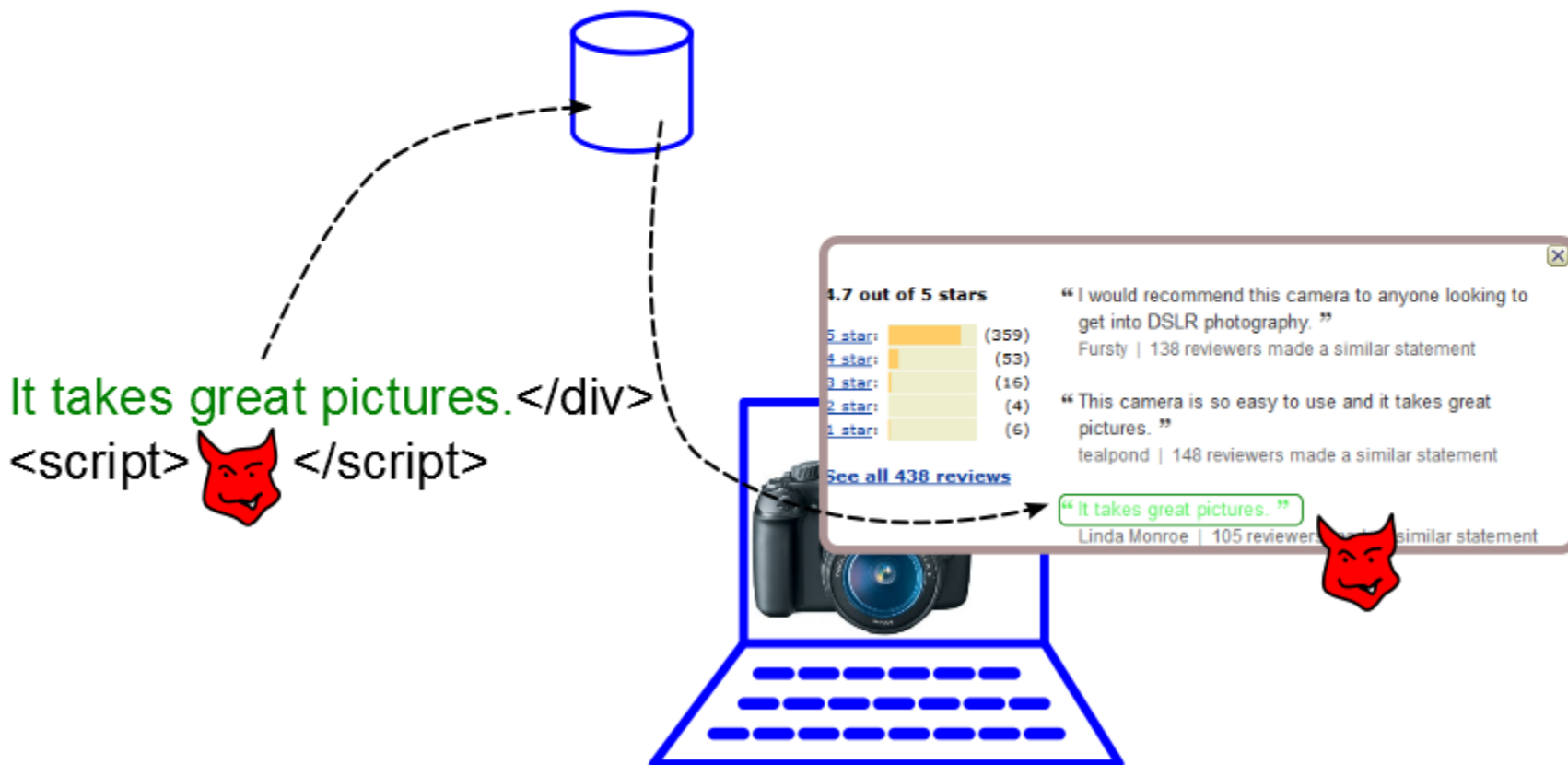


# fixing flaws: cross-site scripting (XSS)

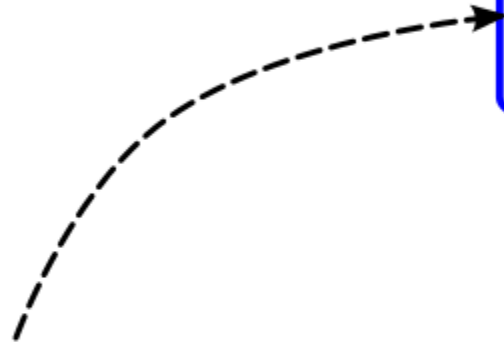
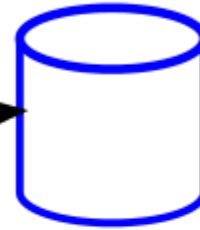




# fixing flaws: cross-site scripting (XSS)



# server analogue: SQL injection

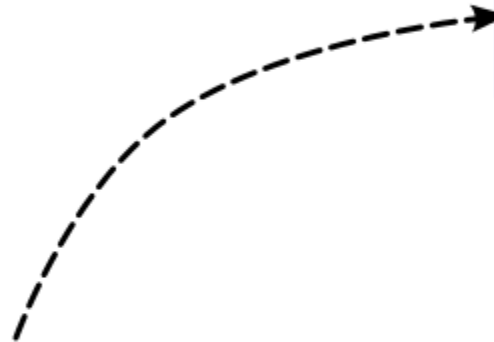


```
Robert'); DROP  
TABLE Students;--
```

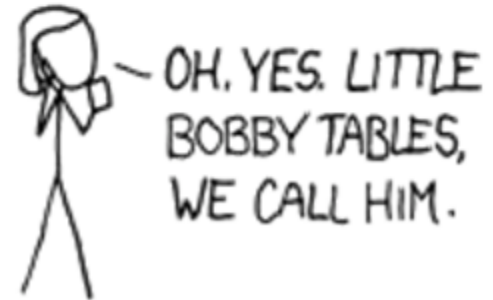


OH, YES. LITTLE  
BOBBY TABLES,  
WE CALL HIM.

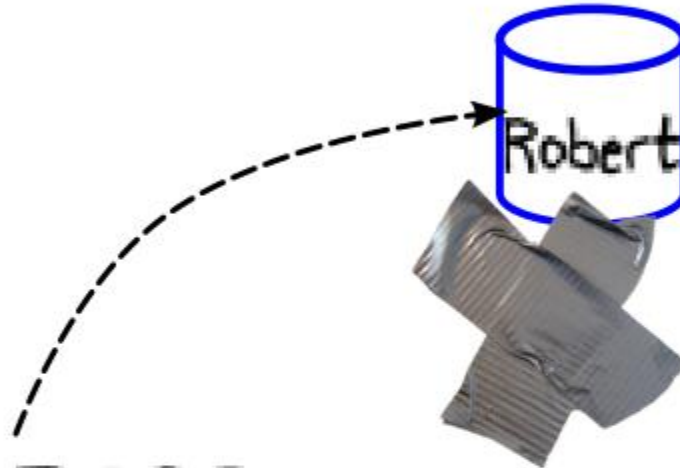
# server analogue: SQL injection



```
Robert'); DROP  
TABLE Students;--
```



# server analogue: SQL injection



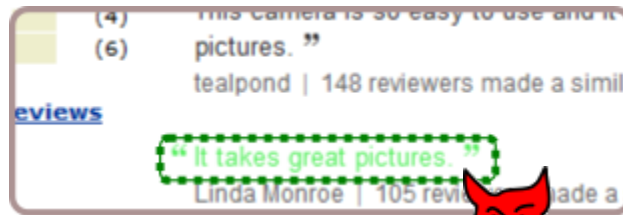
vendors fix their own **servers**

```
Robert'); DROP  
TABLE Students;--
```



OH, YES. LITTLE BOBBY TABLES, WE CALL HIM.

# fixing flaws: cross-site scripting (XSS)



vendors fix their own clients





# Summary

- The web API conflates **CEI** and **DPI**
- A **minimal** CEI can isolate correctly
- **native code** allows **rich** DPIs
- Launching big DPIs **isn't cost-prohibitive**
- The **pico-datacenter analogy** makes security tradeoffs obvious

**No more dangerous links!**

[research.microsoft.com/embassies/](https://research.microsoft.com/embassies/)

- linux & microkernel clients
- Webkit with protocol communication
- Gimp, Inkscape, spreadsheet, word processor
- untrusted app cache



# what about mashups and serendipitous interoperability?

- Today, servers speak open protocols like XML and JSON; we can scrape HTML
- A few standard stacks will use a few standard wire protocols
- Sure, adversarial vendors can obfuscate, but they can do that in JavaScript, too.



# shouldn't / control my browser?

- Shouldn't / get to control my browser?
  - ad blocker
- Letting a user give a third-party program (or plugin) full authority opposes vendor autonomy
  - Trojans / drive-bys
  - Autonomy means vendors can provide a predictable, safe experience



# Accessibility



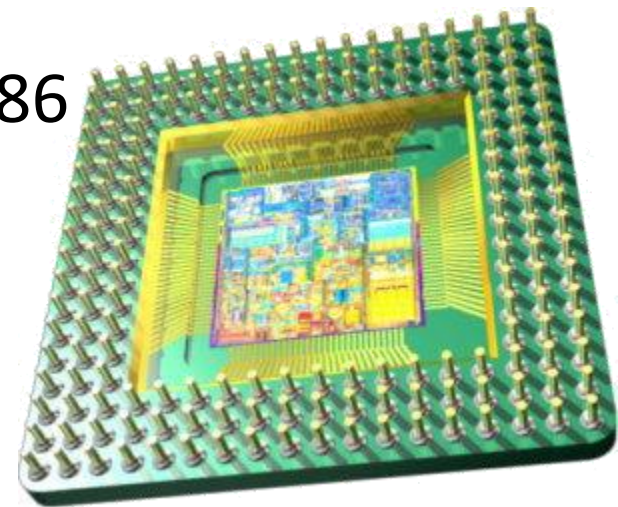
vendor control  
doesn't require  
**unbounded divergence**

Popular stacks (e.g. Windows, Gnome) include accessibility affordances.

# Cross-architecture compatibility

Three approaches:

- Managed code (JS, Java, C#) still a fine plan  
just deploy it from the vendor
- Cross-compile. Debian runs on a dozen archs.
- Binary rewriting  
got Apple from 68K to PowerPC to x86



# Peripherals

- Printers *already* speak IP
  - Google Cloud Print “IP-ifies” your legacy printer
- Same approach for GPS, cameras...
- Disks are easy
  - untrusted “Seagate” app exposes storage





# GPUs



- Long term:  
treat GPU like CPU
- Intermediate:  
exploit GPU segmentation as memory protection
- Near term:  
Even native CPU is pretty sweet

# Deployment

- Start with a browser plug-in  
users enjoy rich apps, like NaCl
- Embassies client with compatibility mode  
supply a default DPI for “legacy” sites;  
Embassies-aware sites explicitly disable legacy mode

