

JOSH CLARK AND @GLOBALMOXIE PRESENT

CAGE MATCH



SENSATIONAL MAIN EVENT



HEAD
TO
HEAD



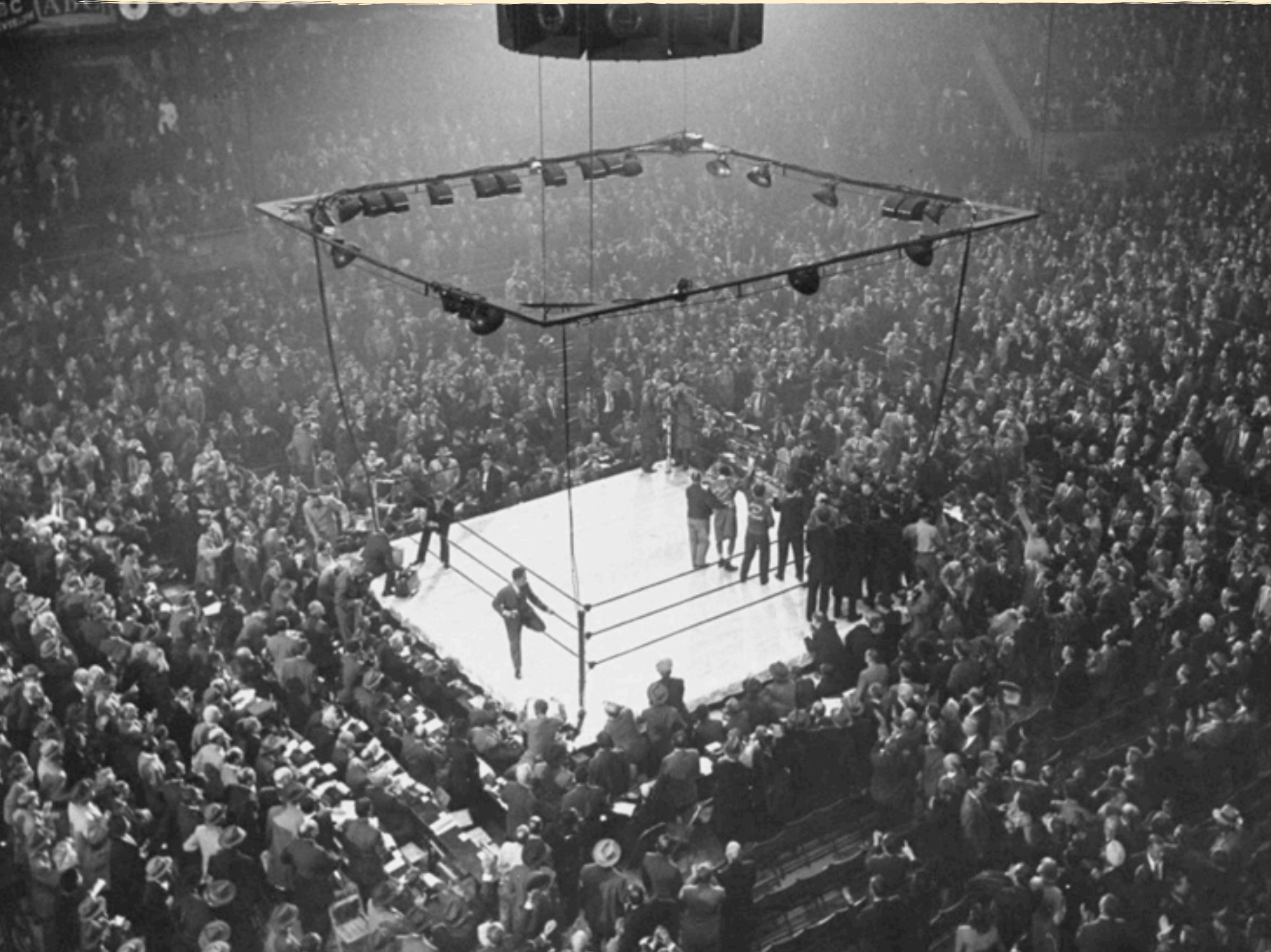
**MOBILE
WEB**

VS

**NATIVE
APPS**

Ladies and gentlemen
Battle royale
Head to head matchup

How do you decide web or native?
If native, which platform?



My friends, I give you the mobile arena.
Used to be uncomplicated.
5 yrs ago: Nokia's Symbian and Windows Mobile 80%.
All with some incredibly crummy web browsers.

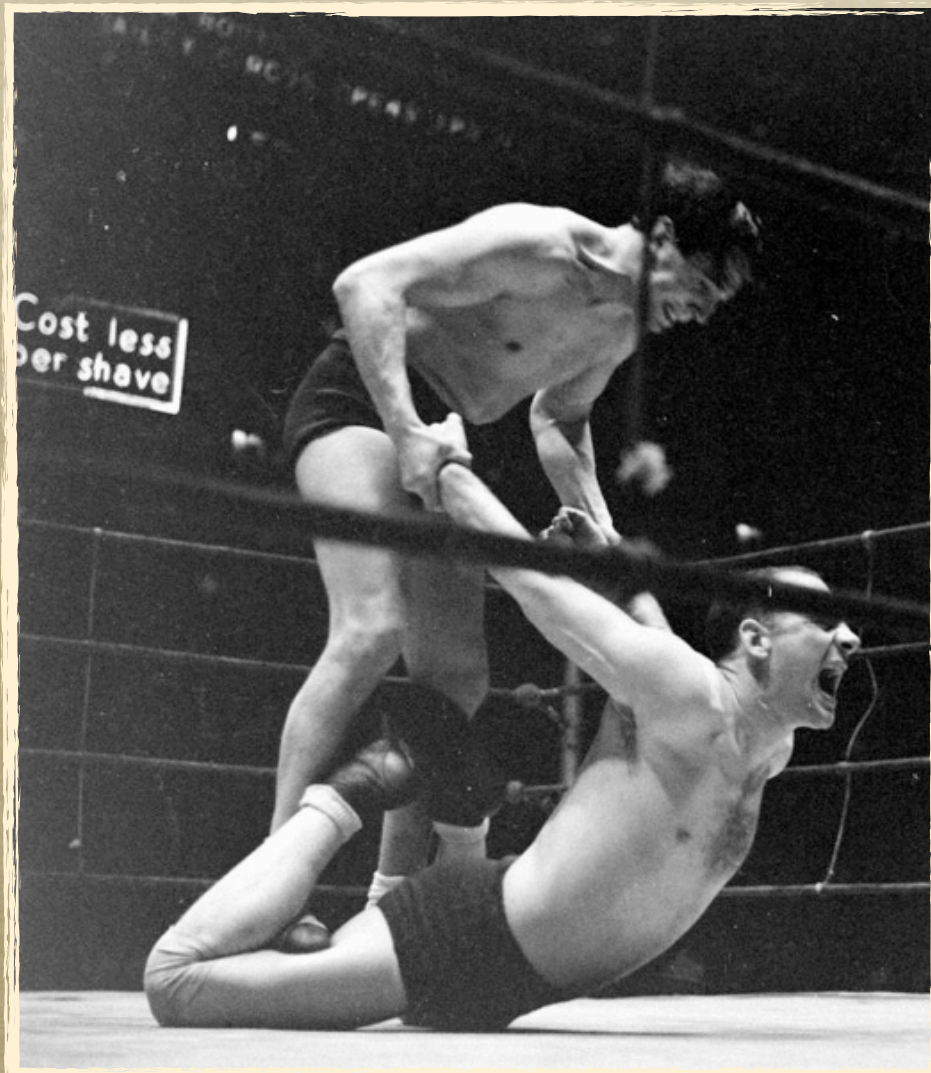
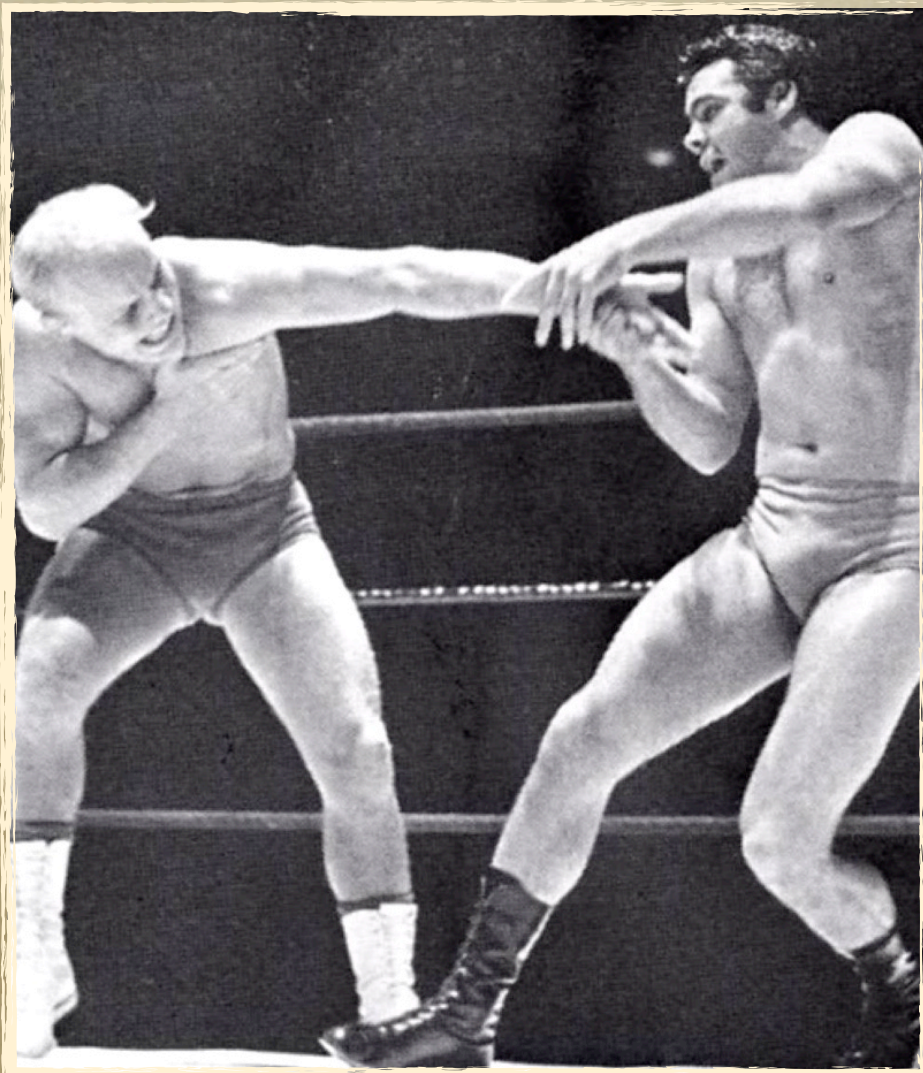
Now: 10 major mobile operating systems.
Plus code for different devices and screen sizes.
The combinations of Android alone stagger the mind.

Exciting times: huge amounts of innovation,
But we're splintering into all these different directions,
all these different devices.

FEAR! PAIN!



TEN ROUNDS OF **MOBILE MAYHEM**



And friends, that means fear and pain.
I think all of are a little nervous about how to keep up with mobile,
with this wild pace of change and growth.

Google Maps

Some kind of focus is required.
Think in more sophisticated ways about platforms we build for and why.

Not an easy, neat answer to how to juggle the available platforms
to provide the best experience for your audience.

Always some pain.
Difficult compromises -- customer vs biz priorities

But with a plan, perhaps we can at least get rid of the fear — and maybe a little bit of
pain, too.

CHALLENGERS

★ **MANY PLATFORMS, MANY CULTURES** ★



Thinking a lot about platforms as cultures.
Helps explain diffs between mobile platforms.
Why iPhone might appeal to some people, Android to others.

More important:
helps figure out how to develop mobile apps/websites
Which platforms to develop for, to design for?

This cage match billed as native apps vs mobile web.
Before that: dip into these platform cultures.

As weigh native vs web: important to realize...
Not all native platforms are the same.
There's not any one mobile culture, no single mobile context.

Choosing platform isn't just a technical question.
Each mobile operating system is its own little civilization,
with its own honest-to-god, distinct culture and personality.
Your design has to fit that culture, adapt to its expectations.

Understand these cultures = understand what means
to develop iPhone app vs Android app.
Or website for phone vs tablet (7" vs 10")

Before get to the main event, start with preliminaries
Dip into just a few of the platforms to show how your approach might be quite different
depending on the platform you target.

82%

★ **OF US ADULTS HAVE CELL PHONES** ★

★ **36% USE THE WEB**

★ **34% USE APPS**



FROM THE WIRE

82% of US adults have cell phones.

How do they use them?

36% of those folks use the web on their phones.

34% use downloaded apps.

It's not either-or, right? People of course use BOTH web and apps, use them evenly.

SOURCES ----

Pew Research Center:

In US, 82% of adults have cell phones.

<http://www.pewinternet.org/Reports/2010/Mobile-Access-2010.aspx>

COMSCORE: 36% web and 34% apps as of Mar 2011

http://www.comscore.com/Press_Events/Press_Releases/2011/5/comScore_Reports_March_2011_U.S._Mobile_Subscriber_Market_Share

82%

★ OF US ADULTS HAVE CELL PHONES ★

30%

ARE SMARTPHONES



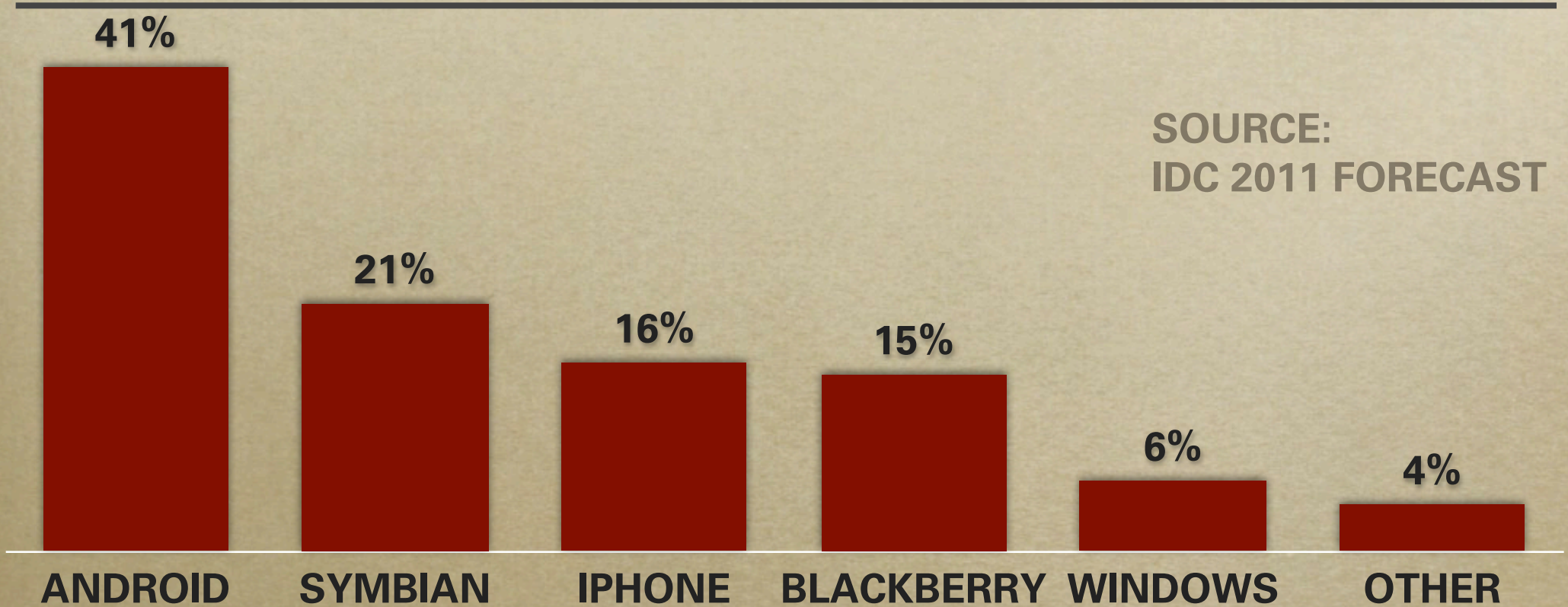
FROM THE WIRE

So: 30% of US phones are smartphones, or “app phones”
On the one hand, that might seem kind of low, only 30%
But three years ago, that number was essentially zero.
Pretty amazing uptake in three years.

But WHICH smartphones are we talking about here?

WORLDWIDE

★ MARKET SHARE ★



FROM THE WIRE

Start with worldwide view.

IDC forecast for 2011.

By end of the year, Android will have biggest marketshare.

Already fastest selling smartphone platform.

(This is just phones, if you count all of iOS, including iPad and iPod Touch, different story).

Nokia's Symbian until now has been the leader, by far.

They've abandoned ship and announced that Nokia's new platform will be Windows Phone 7.

7 billion people on planet—men, women, children—and over 25% have a Nokia phone.

but there's a lot of ground to chip away when Nokia's lead.

But in a market as changeable as this one,
where these platforms didn't even exist three years ago,
important to remember that they may not be the future either.

KEEP AN OPEN MIND ABOUT PLATFORMS.

As designers, we tend not to be people who follow market share numbers.

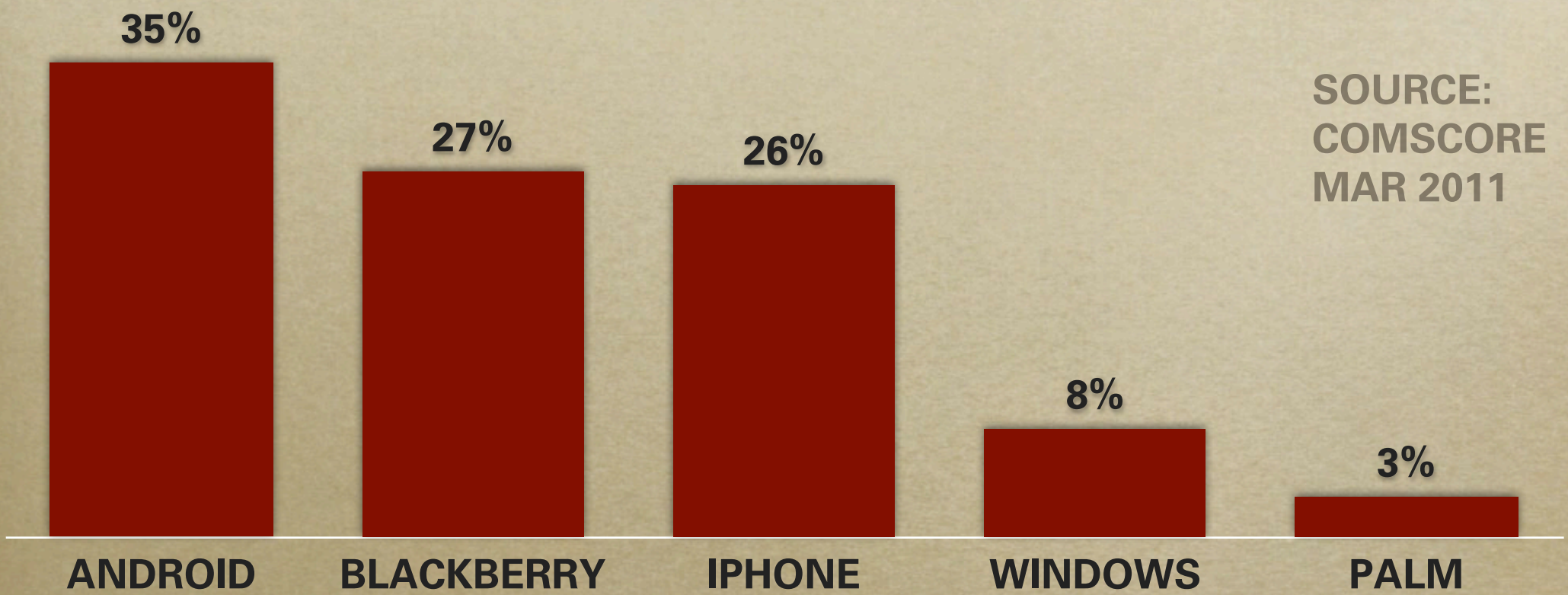
With mobile, it's important. It's changing fast, and it's important to understand where people are flowing.

So this is the worldwide picture. Very different in the US.

SOURCE: <http://www.idc.com/getdoc.jsp?containerId=prUS22762811>

UNITED STATES

★ MARKET SHARE ★



SOURCE:
COMSCORE
MAR 2011



FROM THE WIRE

January, this happened.

Android moved to top spot with the most smart phones in the US. For a long, long time it was BlackBerry, which dropped to second.

BlackBerry, wow, right?
iPhone's never been in the top spot for OS market share.

Now: The momentum is with Android.
Then again, fold in iPad and iPod Touch, iOS takes the lead outstripping Android by over 50%.
Even wider in Europe.

Raw numbers: Android would be your choice.
Numbers important. Help us think where to invest design/dev.
But mobile culture is more than numbers.

Source:
http://comscore.com/Press_Events/Press_Releases/2011/3/comScore_Reports_January_2011_U.S._Mobile_Subscriber_Market_Share

BLACKBERRY



40% OF GLOBAL ENTERPRISE



FEATURING
THE SENSATIONAL SUIT!
KEYBOARD KOMMANDO!
CORPORATE CRUSHER!

Wow, BB has 40% of the GLOBAL enterprise market.
Much higher in the US and in Europe, well over half.
If that's YOUR market, pay attention there.

As might expect, heavily text-centric culture.
Studies: far more email and texting than other devices.

But: Lower browsing activity. No wonder.
Miserable browser. JavaScript turned off.
Setting to turn it on is buried.

Situation's getting better, though, adopting WebKit browser.
More on WebKit in a bit.
For now, though, simply say: very text-driven culture.
Consider: how does your app fit into that culture?

IPHONE

★ **VIM! VIGOR! VITALITY!** ★



ACTIVE!

SPENDY!

WEALTHIER!

By contrast. Incredibly active users, active browsers.
33% of worldwide mobile browser traffic,
despite marketshare that's less than half that.

iPhone users are also the biggest buyers:
70% of eBay's mobile commerce comes from iPhone.
Just one company, sure, but...
eBay alone accounts for 1/4 of mobile e-commerce sales in US.

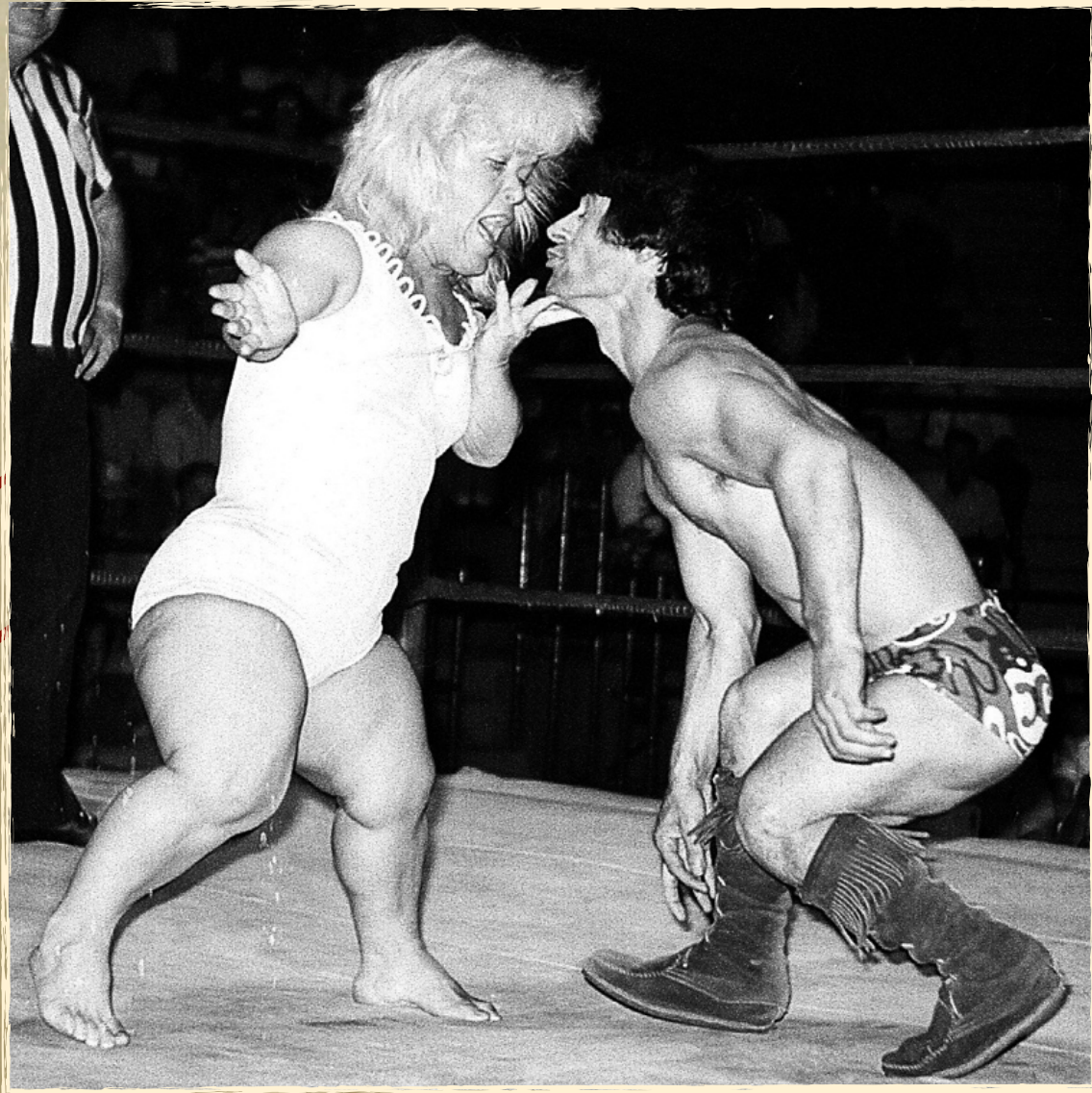
iOS users are more willing to buy.
When you look at the demographics makes sense...
iPhone skews older, wealthier, and more educated than Android.
Not surprising considering the higher typical cost of the devices.

But my favorite fact...

Source:
U.S. customers generated \$850 million in sales in 2010 (\$2billion worldwide)
Total was \$3.4 billion (ABI Research)
[http://mobile-financial.com/node/13353/eBay-hits-nearly-\\$2-billion-in-m-commerce-sales-worldwide-in-2010](http://mobile-financial.com/node/13353/eBay-hits-nearly-$2-billion-in-m-commerce-sales-worldwide-in-2010)
<http://www.pcworld.com/article/204970/ebay-bringing-mobile-apps-into-the-future.html>
<http://www.mobilecommercedaily.com/2010/09/29/ebay-ipad-users-spend-50pc-more-than-pc-web-users>

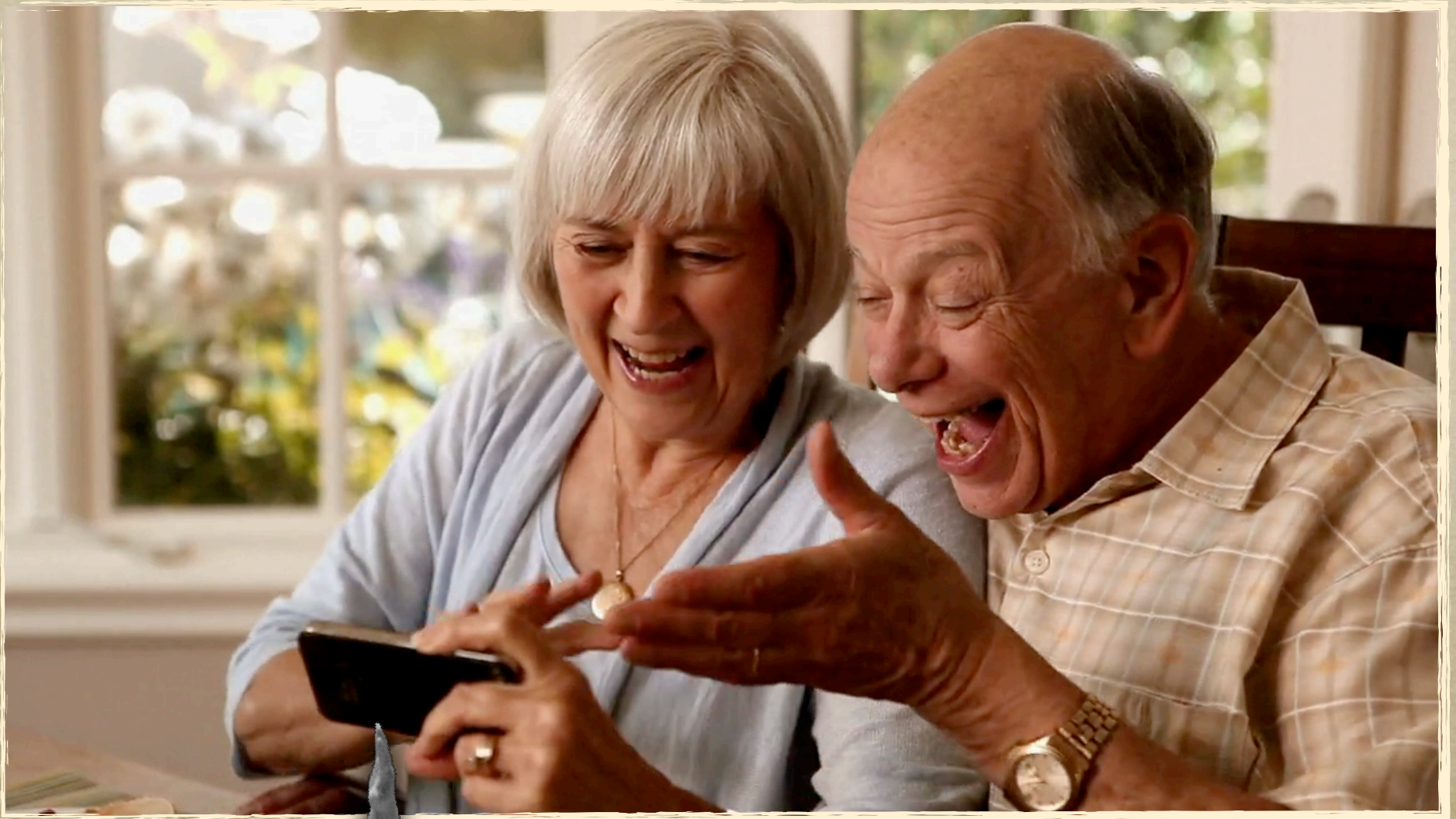
IPHONE

★ SO YOU KNOW... ★



iPhone users also have more sex than other mobile users.

That's according to a study by OKcupid, the dating site.
Apparently studied all their mobile traffic,
statistically significant difference.
Go older, educated, wealthy people!



Speaking of which... Can tell a lot about a culture by its ads.
Says who companies think their audience is,
shapes who buys the product going forward.

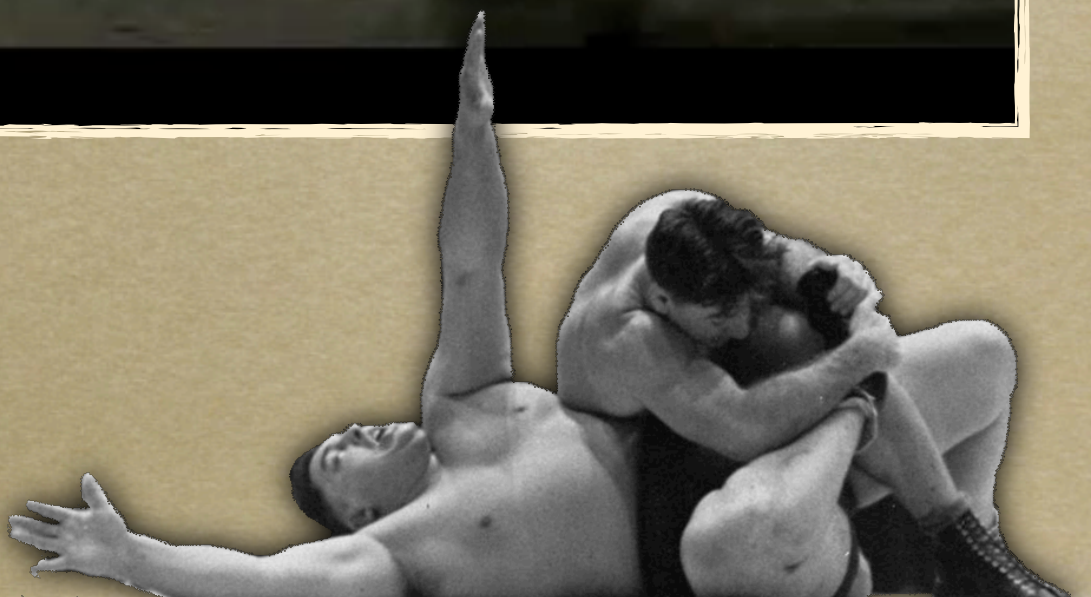
Marketing inventions, but give hints.
So I want to look at a couple of ads today to get a sense of personality.
Here's the ad Apple released right after iPhone 4 came out.

[view]
It's all about warmth, personal connection, emotion.
They're smiling at the device the whole time,
smiling at the person on the other side.
Device imbued with these personal connections.

Apple's marketing, design, and experience emphasize:
beauty, personal connection, polish, pleasure
human feeling, fun, even emotional

Emotional connection to hardware and software.
When a lot of people got their iPhones,
surprised by how personally attached they were
Not just to the phone but to the apps.
Suddenly apps were content, not just tools.

Entertainment & social connection
A new way to think about software.
Not only were phones accessories but apps, too.
Bag/desk



Compare that personality and message to this ad for Droid X, Android phone in US.

Droid ads like a sci-fi horror movie.
Cold technology versus warmth of human connection.
Tools versus content.

ANDROID

★ IT'S THE TECHNOLOGY ★



Android emphasizes technology: tools and features.
It does MORE than other phones.

iOS emphasizes polish, refinement, friendliness

Android emphasizes its tools, customization.

It's YOUR phone. It's your arm!

Experimentation. Cutting edge also means rough edges.

It's not always the most comfortable experience.

Feature count and flexibility at expense of polish.

Android users would welcome improved user exp,
but it's not highest priority for them or the platform.

Best suited for technically proficient high-end users
who don't mind tinkering to get past hiccups.

Nielsen: Android users customize phones more than others.
Lead iPhone in ringtones, wallpaper, picture downloads.

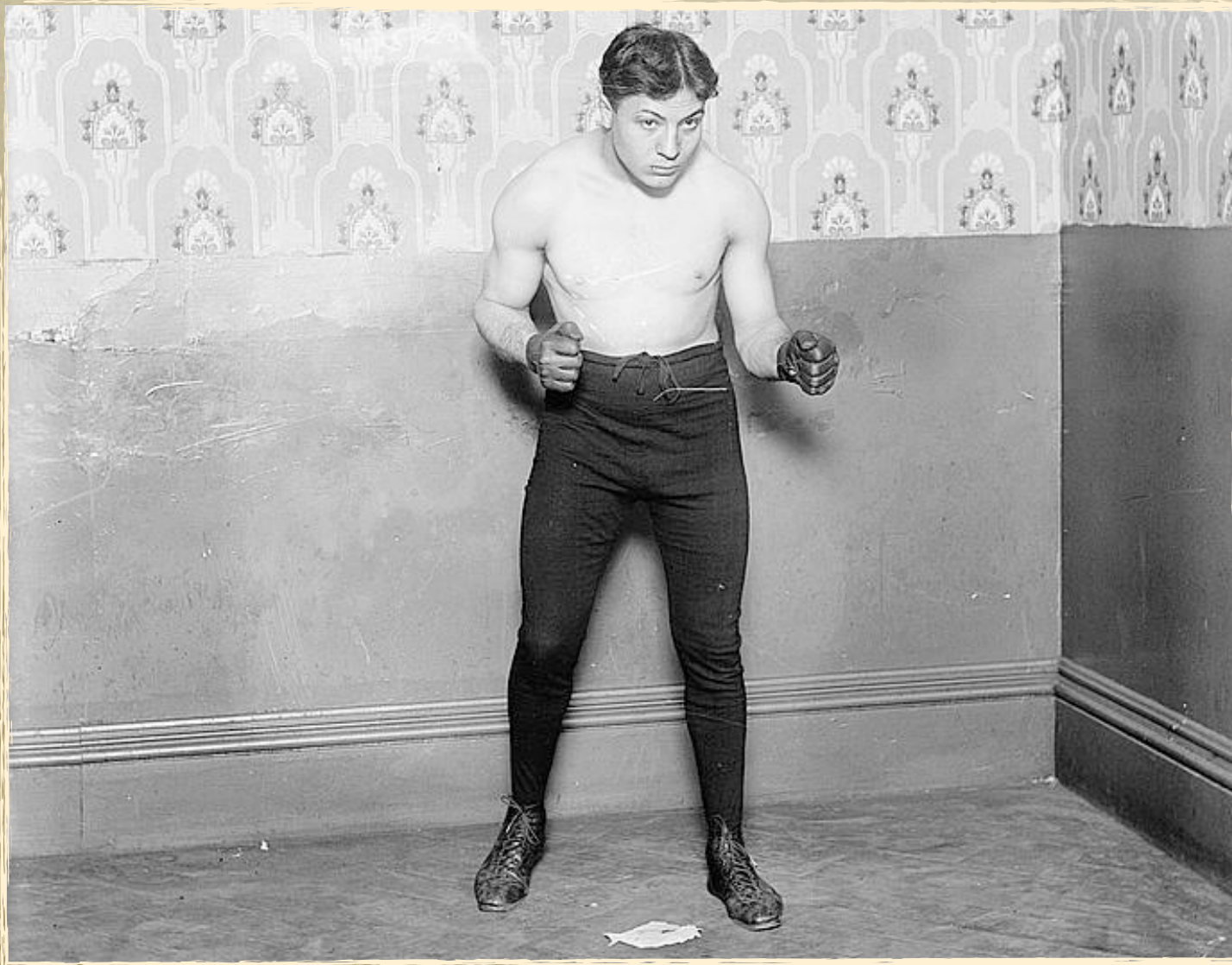
Also more popular for straight communication:
More text and instant messaging than iPhone.

Curious:
Fewer app and game downloads than on iPhone;

Again, iPhone users see apps as content. iPhone is a content/media device.
Likely related to superior app store.
Android is tool and task-driven.

ANDROID

★ SKEWS YOUNGER ★



Android tends to be slightly younger
In part, because phones often cheaper.

While it has lots of rough edges, it's a "good enough" phone.
Inexpensive, widely available.
Great recipe for spreading far and wide.
Top-selling smartphone platform around world.

Designed for geeks, by geeks.
But because of its price, also getting into the hands of regular, ungeeky folks,
A little bit of a disconnect, audience changing.

Causing some worrying trouble.

Need threshold know-how to stay out of trouble on the device.
It's no accident that antivirus software starting to come out for Android.
We're going to see some nasty surprises as nongeeks
find out how geeky their devices are.
We'll see.

WINDOWS MOBILE



THE FORMER CHAMP



Ohhh!! Poor Windows Mobile.

Windows Mobile used to dominate market,
along with Symbian.

Microsoft finally brought us something new late last year.
They're trying to work a spiffy transformation.

WINDOWS PHONE 7



CLASSY! URBAN! MODERN!



Windows Phone 7 has a whole new look
Classy UI design language: Metro
Very interesting ideas; as name suggests, urban and modern.

Design team focused on a very specific persona:
young socially active couple, geographically mobile
active life, juggling lots of people and media.

aiming for a very personal experience
that helps you make connections among people/events in your life.
Social media plugged into everything.

Highly customizable, a phone that looks and feels like yours.
Merges personality traits from both Android and iPhone:
About personal connection like Apple,
but also about personalization like Android.
Make the phone yours.

My point:
No matter what technology you use to build,
be aware that there's no single mobile culture.
Platforms and their users have different priorities.
Think about those priorities as you go to design for them.

LITERACY!



PERSONAL LITERACY IN PHONES



This is tough.
In order to design for an individual operating system,
gave to be fluent in that operating system.
Have to know its conventions and customs.

Android: Gotta be able to really grok that platform.
Lots of designers only know iPhone.
See iPhone worldview creeping into Android designs.

That's no good. You need to know what you're designing for.
Unfortunately, that's REALLY difficult to do unless you live with that phone.
That's inconvenient, and it's expensive.

You can use emulators, but not ideal.
You don't get the sense of the ergonomics,
of the screen resolution,
of the true performance.

If you have the means to buy and live with a different phone for a while, then great.
But I think that we're likely headed to a different answer.

SIGNATURE MOVE



PLATFORM SPECIALIZATION



And that's specialization.
In wrestling parlance,
develop your signature move, your awesome takedown.
(This is apparently called the nanny goat.)

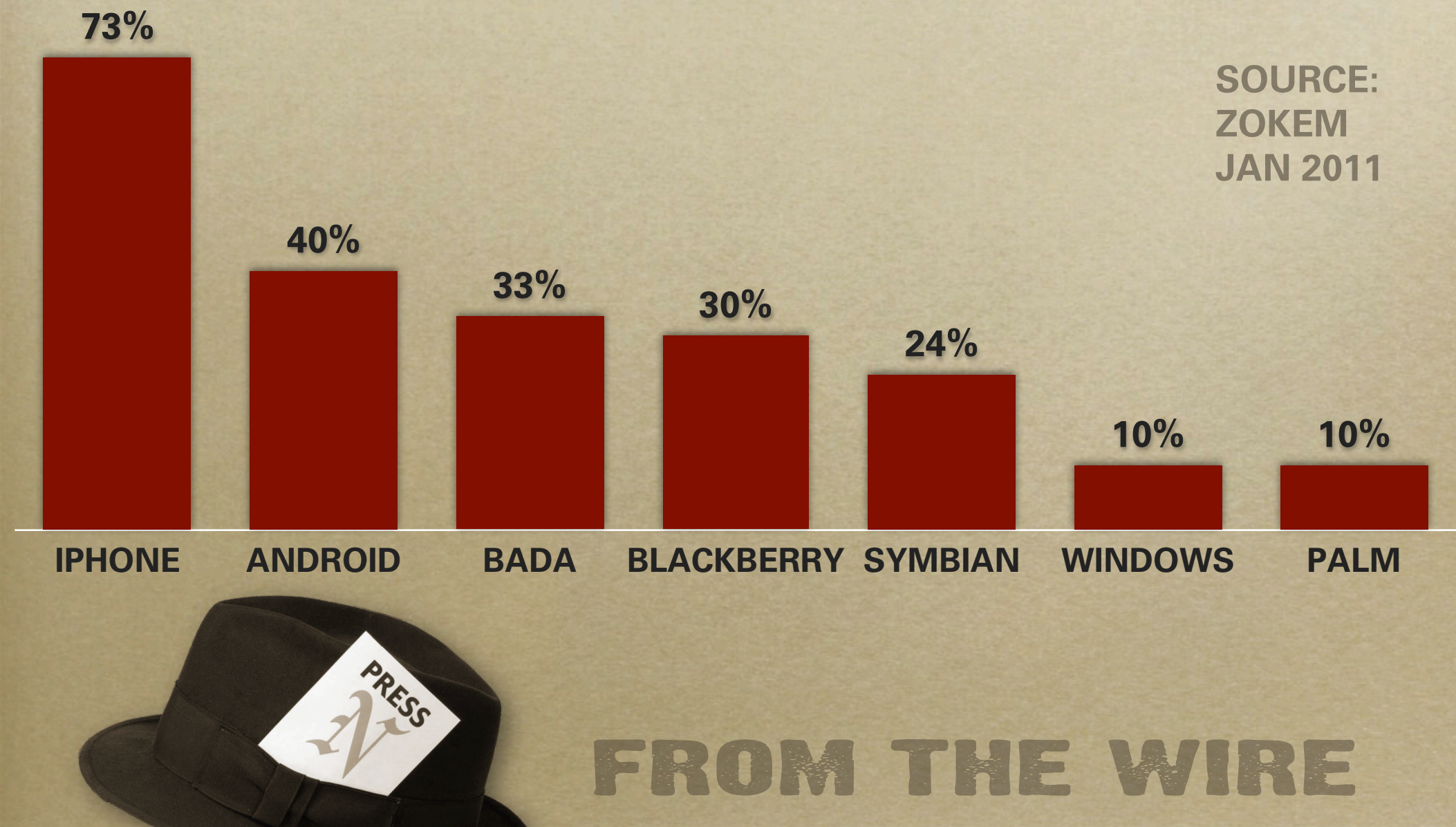
The web has fostered a culture of generalists and that's worked well there.
But I believe that for native app designers,
we're going to have to become specialists in individual platforms.

Go deep,
Really develop a nuanced understanding of
interface conventions and use patterns of that platform.

What's your signature move?
What's your deep mobile know-how?

SATISFACTION

★ WANT NEXT PHONE TO USE SAME OS? ★



FROM THE WIRE

So which are people most happy with? There's a survey for that. Asks if want their next phone to use the same OS. iPhone users far and away show the most loyalty.

After that, things go over a cliff. For all the rest, less than half want to continue. Bada (3 people using it) Don't count those other platforms out just yet, Windows Phone 7, a little early to tell. Those numbers mix Windows Mobile.

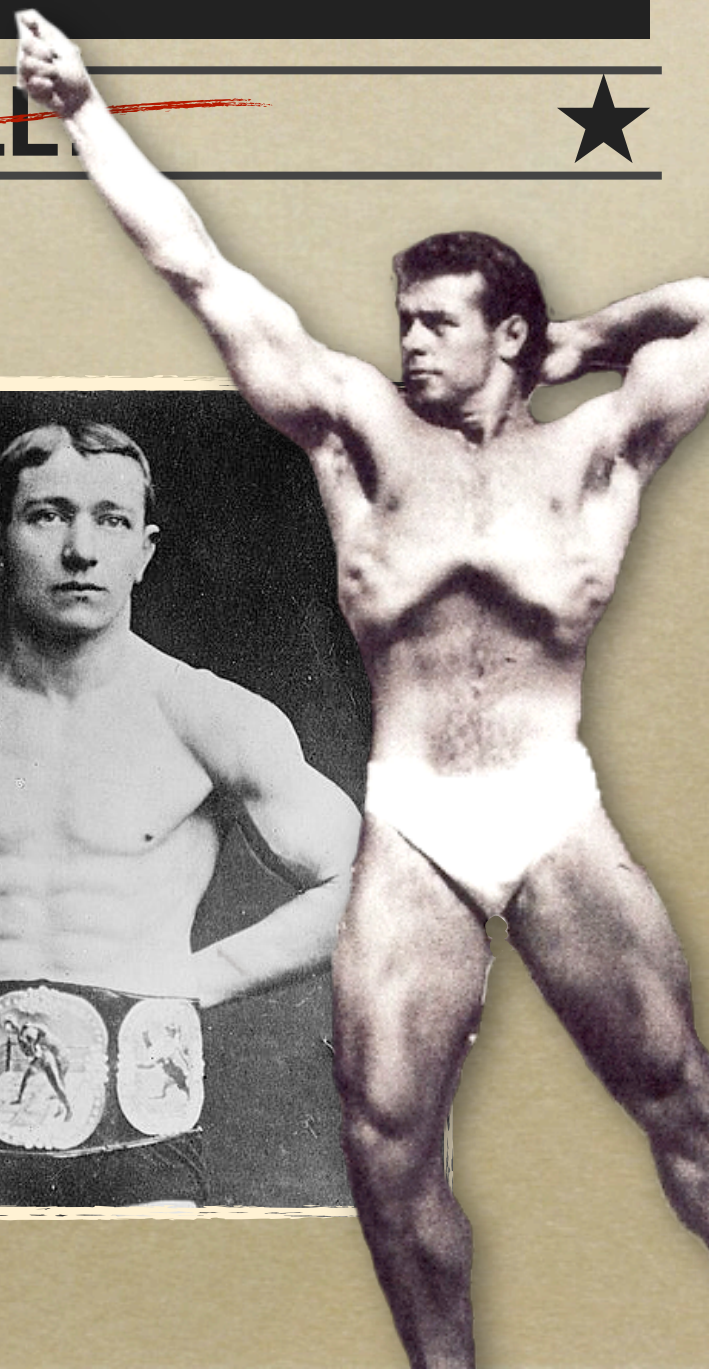
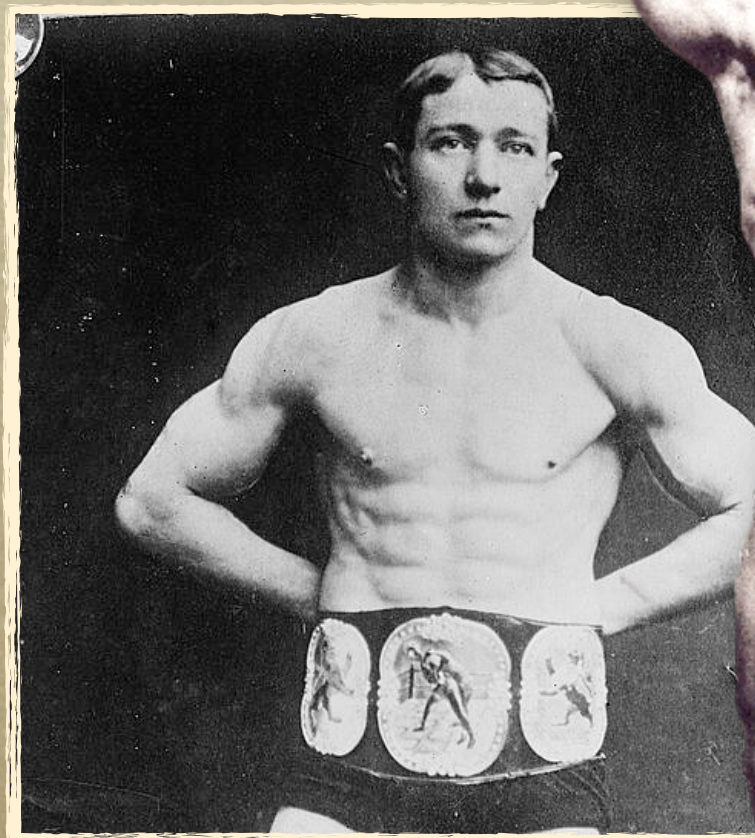
But iPhone (Android to degree) setting pace of user experience. And by user experience I mean exactly that. A great experience. Do you want more of this or not? Nobody comes close to iPhone on that.

But it's not all about numbers.

<http://www.zokem.com/2011/01/in-the-us-market-iphone-outperforms-other-mobile-platforms-in-user-loyalty-by-a-wide-margin-android-is-second-blackberry-fourth/>

SPLIT DECISION

★ ~~WINNER TAKE ALL~~ ★



All these different platforms and personalities, who wins?
Nobody. No single mobile platform winner in near term.

Android best selling type of smartphone.
By end of year, will have most global market share.

But room for more than just one winner.
So much churning innovation, investment, experimentation
Idea that any one platform is going to win
in near future is naive at best.

More important, these are very personal devices.
The personality of the software matters.
Android FEELS different from iPhone, which FEELS different from Web OS.

This market is not like Mac vs Windows.
Not about TOOLS, where compatibility is most important.
Compatibility isn't even an issue. More and more, data stored in cloud.

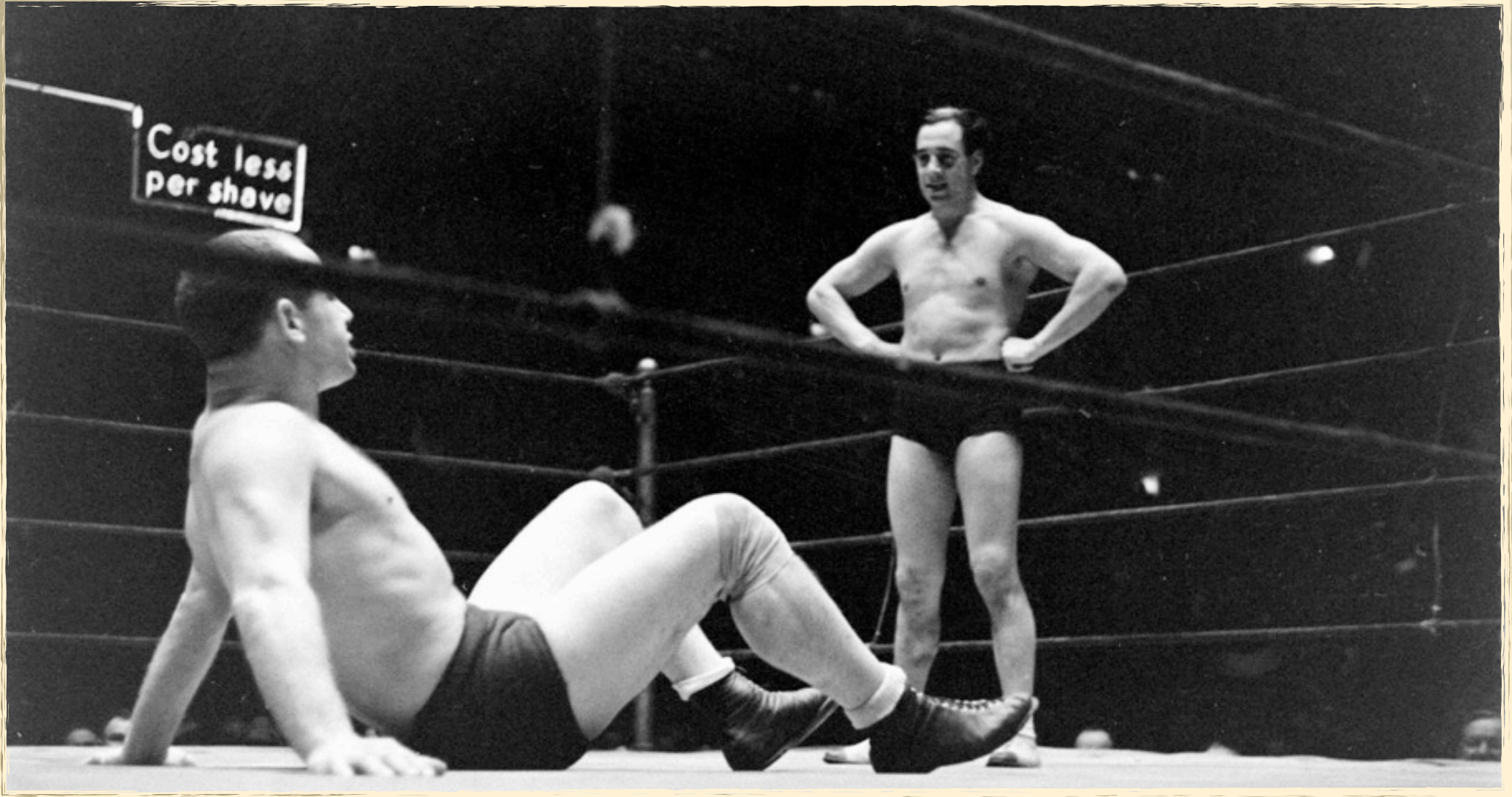
Personal vibe and emotional attachment counts more.
Personal nature of these devices means more cultures can thrive.

Also important to remember, before we pick winners:
huge number of people aren't even in the match.

TECHNICAL K.O.



YES! WE HAVE **NO APPS!**



Only 30% of U.S. adults have apps on phones.
The number is a bit lower in Europe.
That means over 70% are still sitting this out. No web, no apps.

Compare that to text messages.
3/4 US adult cell phone users send & receive text messages
87% of teen cell users text.

At this moment, ours is a text culture more than an app or web culture.
That's changing, web and app use growing fast.
But that's the reality of our moment now.

So sometimes, it may make sense to...

THINK SMALL



CONSIDER SMS TEXT APPS



Think small.
Fancy mobile apps—whether web or native—aren't always the answer.

SMS text apps work great in some situations.

Simplest solution almost always best.
I love the new. I love where mobile tech is headed.
But let's not forget the tried and true. Text works.

Face cremes.

Less is more in mobile apps.
Or as Milton Glaser says, just enough is more.
Whether website, app, or text service:
what's minimum I need to offer to let audience accomplish their goal,
to be more awesome?
Sometimes low-tech, small solutions are best.

WEBTASTIC!



EVERYONE LOVES HER!



So. What about the web?
Let's get down to the main event.

Varied platforms.
Web supposed to solve this.
Blind to platform.
Put up a web page once, shows up anywhere.

Is the Web strong enough to wrestle this chaos?

Well, sort of. It does some things really well.
Sidesteps app stores, gives you complete control over distribution.
Sidesteps carries dragging their heels with OS updates,
big problem for Android.

Don't have to build a separate app for every type of device:
web is available everywhere.

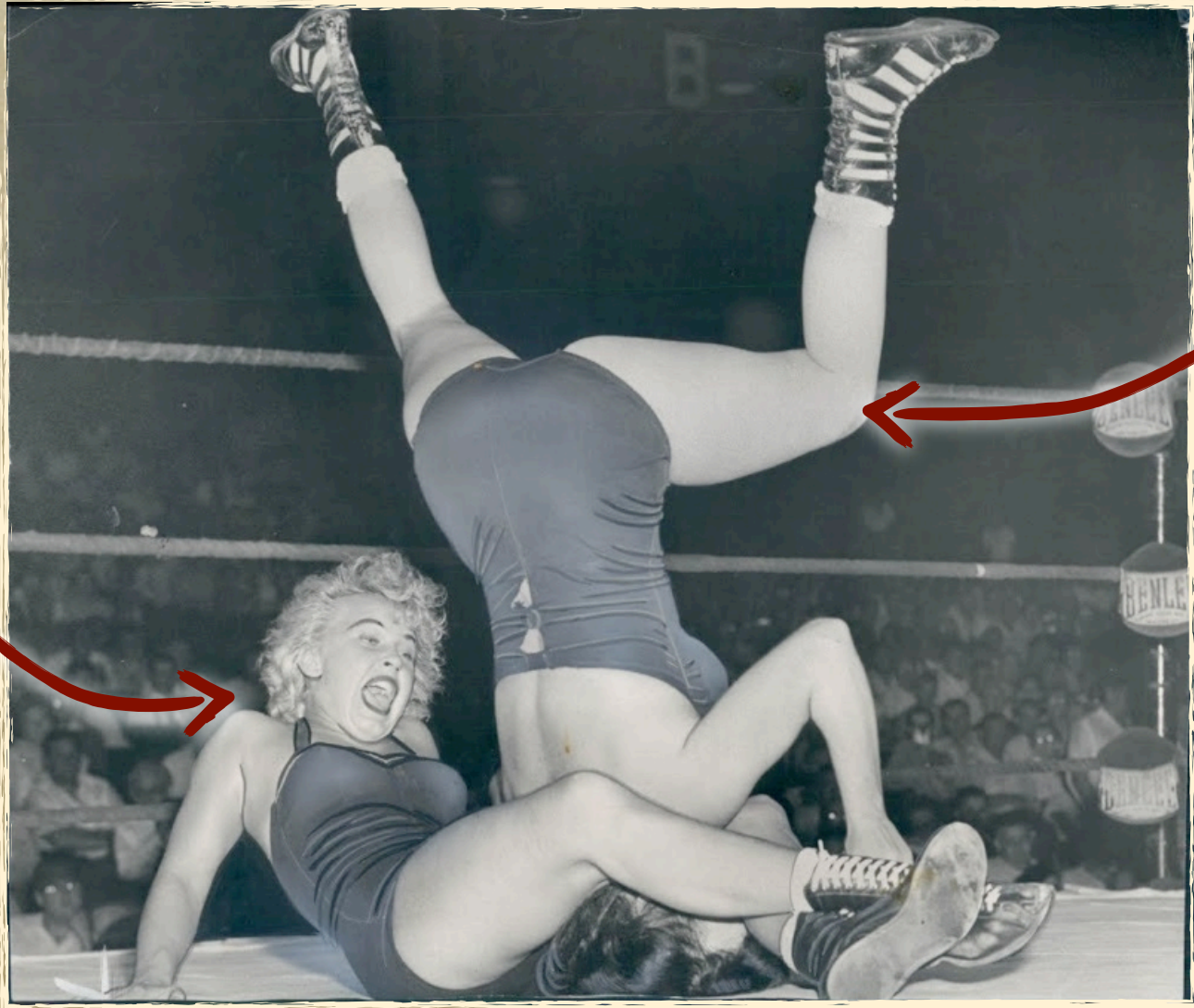
Good news: Nearly all have adopted WebKit engine.
It's a GREAT browser engine.
Web designers haven't experienced this kind
of browser ubiquity in a long, long time.

We say we like choice and competition.
But our dirty secret is that designers secretly love monopoly
For the first time since Mosaic, we've got an environment that resembles that, in mobile.

Windows Phone 7: IE7 — IE9 later this year.
But this is a... (text is cut off)

APP CULTURE!

★ APPS = DOING WEB = REFERENCE ★



App

Web

As capable as web is, apps have advantage in mindset and culture.
<3 yrs app stores, already have entrenched app culture.

User intvws: distinction between doing and reference.
When people simply want to look something up, turn to the web.
But when they have a recurring task, a thing to do, a game to play,
instinct is to turn to app stores, not to the web.

That's not to say that can't change. But that's the current culture.

The mindset, the advertising, the operating systems...
They all reinforce the idea that to do something, need an app.
To use an app, need app store.

Big hurdle to overcome for web apps if that's your only route.

WEAK SPOTS



THE WEB'S UX WEAKNESSES



★ **GREAT EXPECTATIONS!**

★ **HEAVYWEIGHT LIBRARIES!**

★ **ROPE-A-DOPE!**

★ **CLUNKY GRAPHICS TOOLS!**

★ **NOT ENOUGH HARDWARE!**

★ ~~**ONLINE ONLY!**~~

You can do remarkable things on the mobile web now.
Getting better as the browsers improve, devices get faster.

But: Can't match native apps in features, speed, polish; probably never will.
Native apps hold possibility of better, sleeker experiences than the web.
The web simply has some relative weaknesses. [list]

Great expectations: Devices created expectations of touch interaction: web not good
Complex gestures difficult enough in native apps.
Huge hassle on web. Drag and drop, good luck.
Swipes: Flickr and NYT on iPad.
Operating systems and native apps have leapfrogged web in interaction.

JS libraries trying to paper over that. (Sencha Touch, jQuery Mobile, jqTouch)
[click]
But they're heavy. Performance: both with caching and with processing.

Rope-a-dope: In general, just slower.
Leaning into ropes compared to native.
JS interpreted, native is compiled.
Speed is important in any software -- in any business -- but especially mobile.

Mobile apps = takeout restaurants. If not fast, not keeping promise.

Graphics tools. Getting better! SVG and Canvas.
Painful to work with and just not yet up to the standard that native SDKs offer.

Hardware: Browsers don't yet give you full access to all the hardware sensors
As important: system libraries like address book.
Coming. Android 3, Honeycomb.

Online only:[click]
Your app's files can be cached offline.

RAZZLE DAZZLE



WEB BOWS TO APPS



Native apps have edge with the fancy stuff.

Now, sure.

Can certainly build slow native apps, can certainly build awesome web apps.
But it's EASIER to build the awesome with native apps, though.

And it will have a natural-feeling experience,
with full access to native widgets that simply makes
iPhone app feel like an iPhone app, an Android app feel like an Android app, etc.

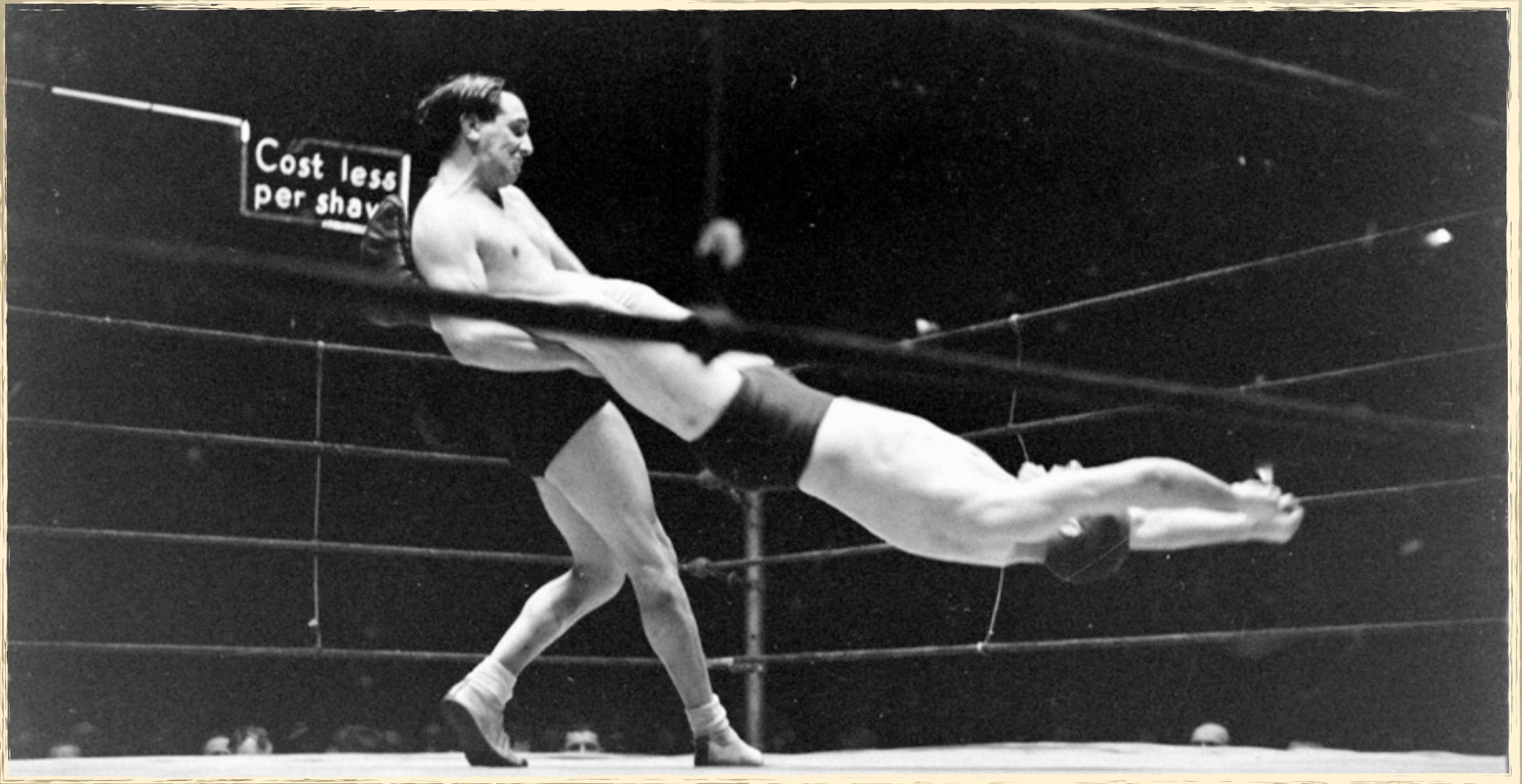
[slow]

It's hard for something developed in a cross-platform technology
to provide as great an experience as a similar native app.

BIG REACH



WEB TOUCHES EVERYONE



Then again, the most crucial aspect of user experience is:
Can the user have any experience at all?

Native apps are siloed to their platform.
The web of course touches everyone on all platforms.
So mobile web wins here over platform specific native apps.
Build the app once and it instantly hits all platforms.

[slow]
This is important:
Mobile website or web app is cost of entry. Table stakes.
I believe you must have a mobile site.
Whether or not you build a native app,
you have to have a mobile website.

Web is the hub: lowest common denominator of all platforms.
Your website should serve every device carried by people you care about.

Doesn't have to have all the bells and whistles of a native app
in many cases, it can't.
But you have to have one.

Web wins for reach.

THE MONEY



PAYMENT

FINDABILITY



Are you REALLY reaching everyone, though?
What if they can't find it? What if it's too hard to pay for it?

For mobile, web isn't the first place we look.
As I mentioned, there's a strong app store culture.
If you want an app, the default is to go to app store.

And apps easy to BUY in app store. One-click purchase.
Buying web apps trickier, or you get into subscriptions.

Findability tough for both. App stores are crummy for search.
400,000 iPhone apps, impossible!
We are in Altavista days of app stores.
In late 90s, few hundred thousand webpages and you couldn't find anything.

Then Google came around.
With billions of pages, you can find great resources.
Problem will be solved.

For now, surveys tell us that over half of users find apps by word of mouth.
And that the app store is where people look for apps.
Apps get the point for payment and findability.

BANTER!

★ APPS SPEAK! SWAP INFO! ★



Web apps can talk to each other cross platform.
URLs are awesome. They let you send data to any other webpage.
With native apps, meanwhile, it's much harder to share data.

The platforms do have internal URL schemes to share information among apps,
but only if those apps are actually installed.

Hard to link to info in an app on my phone to share with someone else.
Only reliable way is to send a link to a webpage.

Content that needs to be shared needs to be on the web.

Web is terrific at that.
The URL is an important innovation that we lose when we're rattling around individual devices.

THE BUSINESS



BATTLE OF THE BACK OFFICE!



All these categories so far = user experience
But what's YOUR experience of building and maintaining this thing.

These are the business considerations.

Platforms have different languages to build native apps, different code bases.
Web apps, obviously, familiar technology. Already have the skills
HTML, CSS, JavaScript
Makes development across mobile platforms easier: same code base.

Testing is super-easy.
A/B testing and beta distribution = pushing changes to server.
Hard for native.
Prototyping: faster to build a HTML app than an Objective C app, for example.

easier updates. just push the update.
don't have to worry about app store updates, customer upgrades.
Hugely easier for maintainance and customer support.

Freedom to do what you want.
If it's possible in a browser, you can do it. Nobody will tell you no.
Apple: brouhahas over content, technology, possibly cynical business maneuvering.
For most, not terribly risky.
If you're pushing the limits, if you're in a gray area of what technology can do or trying a new business model, there's risk.

THE BUSINESS?



WHAT ABOUT YOUR AUDIENCE?



Better or worse, these biz cases tend to drive decisions.
Fair enough. Has to be worthwhile, have to resource the thing. Sure.

But man: how do we do right by our audience, by our customers?
UX designer job = champion of our customer.
To be all about the best experience.

Best EXPERIENCE will almost always be done in a native app.
Easier to find, easier to pay. Better, more polished, more native exp.

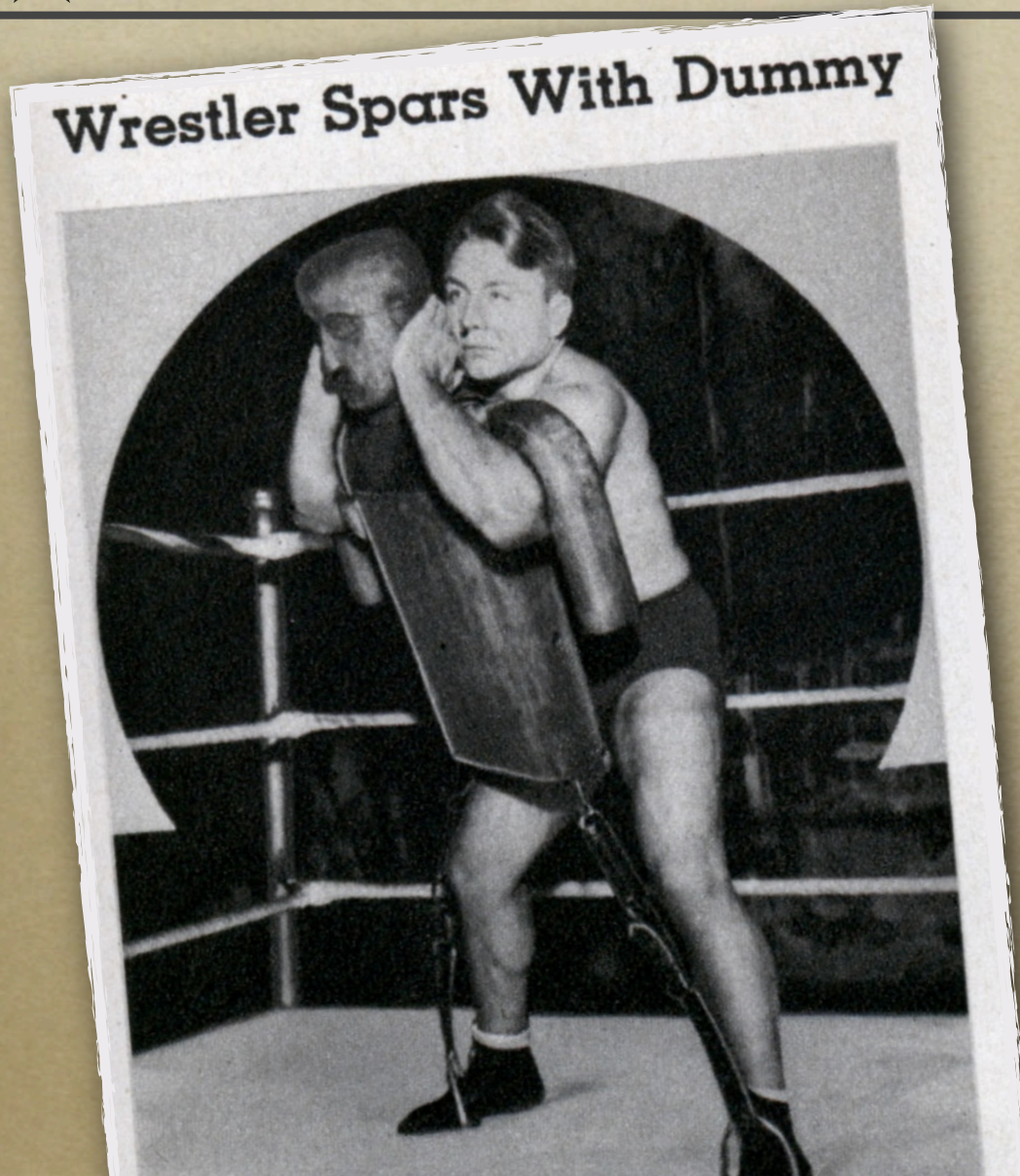
Awful tension between business expense/resources,
and best experience of your customers.
But that's nothing new. Tension always exists.

There's compromise here.
And that compromise means...

HOAX REVEALED



RED HERRING!



NOT A REAL FIGHT!

**“ENEMIES” ARE
FRIENDS!**

ALL A SHAM!

The winner here is neither the web nor native apps.
The mobile web vs native app cage match is a sham.
It's a false argument, a matchup that's been overhyped.

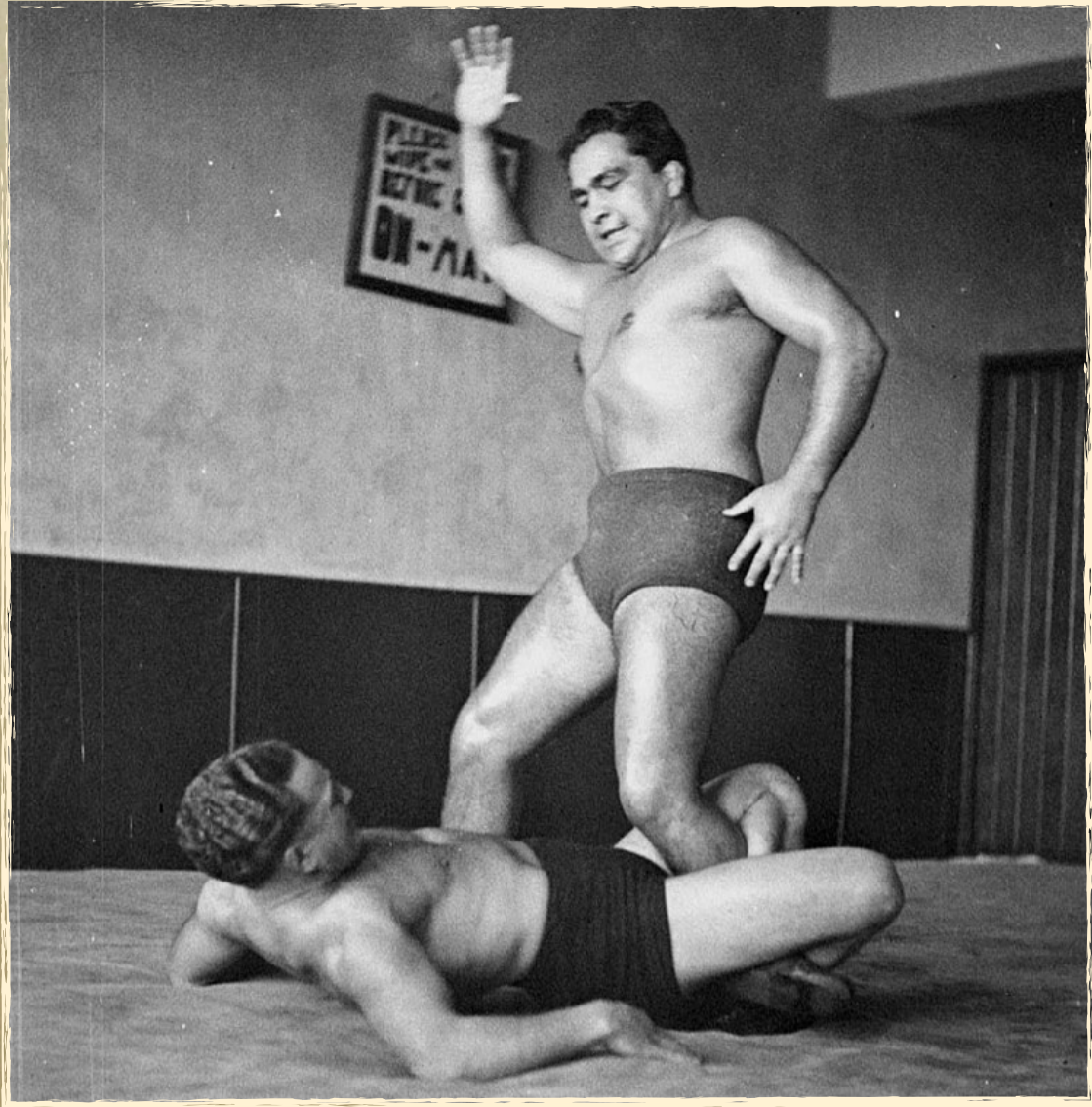
There is a winner here, a mystery contender, whom I'll reveal later.
But it's not web or native.

Web vs native is an implementation detail, a business consideration.
You can do well building apps in native code.
You can do well building apps with HTML.

But that's code, not user experience.

LOVEFEST!

★ WE'RE ALL IN THIS THING TOGETHER! ★



We're all in this thing together.

Build it with native code, build it with web technologies, it's all for the same goal. To help your user be more awesome.

There are some who fight web vs native like a religious war. Let it go. We need both.

How do you do that?
Web plus one or more native apps.

FLAGSHIP APPS



OBEY.

Do your mobile website, you have to have it,
but think about flagship apps.

Create something awesome for your best customers.
Reward them with native apps filled with
double rainbows all the way across the sky.

Choose one or two platforms.
Find out what devices your customers use.
Understand the personalities I mentioned earlier.
Aim for the mobile cultures & populations
that match your message and demographic.

Reward your best customers with an app that
does things a one-size-fits-all website can't do.

Make full and subtle use of sensors,
put graphics and transitions to work.
Make it fit in completely with their native app experience.

TAG TEAM!



HYBRID APPS! BEST OF BOTH!



Web app vs native app isn't always a clean distinction.
All kind of depends on what you mean by web and "native."
A quick nod to hybrid apps.

PhoneGap.

Lets you build web app, distribute as native app in app stores.

PhoneGap is basically a kick-ass browser.

The browser that SHOULD ship with your phone.

Gives javascript access to your address book, to the camera, to most sensors.

But it comes as the source code to that browser.

You just put your web app's html, css, and javascript into a folder in that code.

The developer compiles it like a native app.

Submits to store.

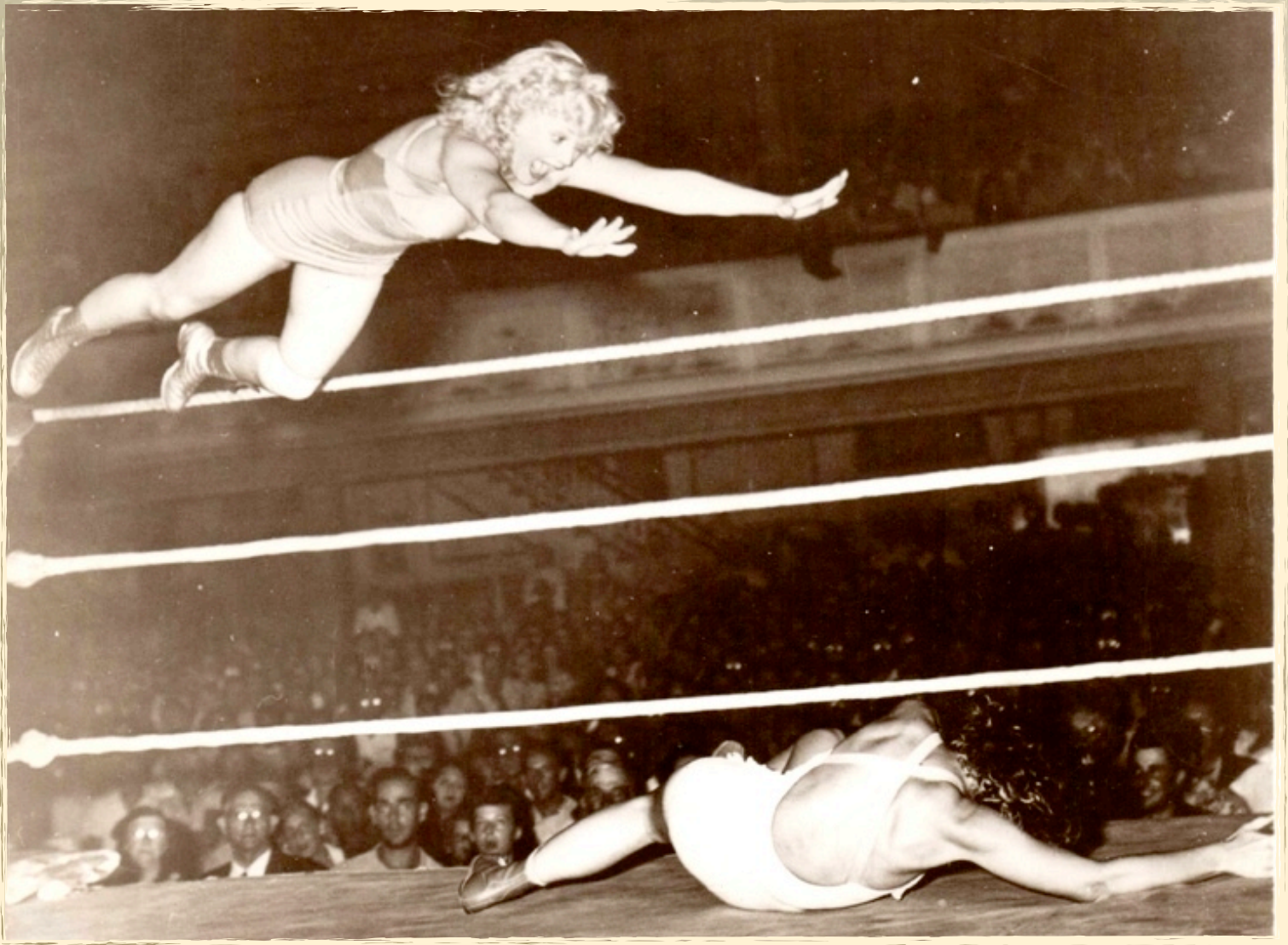
Addresses some problems of building web apps
(distribution, payment, findability, common code base), but they still don't give the
absolutely rockin' experience.

The thing about hybrid apps is that they still use
relatively slower and less capable web technologies.

LIGHTWEIGHT CHAMP



WEB FLOATS LIKE A BUTTERFLY



Great at relatively lightweight UI.

if you don't need fancy gestures, animations, or graphics.
If you're building a to-do list, for example.
Build it with web technologies, bundle in phonegap.
If you CAN do it with the web, then by all means, do it with the web.
It answers lots of business concerns—easier, cheaper, faster.

So, great: build one web app, distribute to all app stores.
Not so fast. Build once for everyone/everywhere
is a pipe dream, even for the mobile web.

Especially true when building hybrid apps distributed through app store.
IF downloading as native app, people expect it to LOOK like native app.
Otherwise, it feels wrong.
iPhone like iPhone, Android like Android.
In the browser, you have much more interface leeway.
Not when launching an app from your home screen.

Need different design for every platform.

ELEPHANTINE!



MANY DEVICES, DIFFERENT VIEWS



So you know the story of the blind men and the elephant.
One's got the trunk, another has the tail, another has the leg.
Very different views of the whole based on the context.

That's how you should think about content across devices/contexts.

This isn't just design & presentation.
Often need CONTENT specialized to the specific context.
I'm not just talking about responsive web design.
This is more than adding a CSS style sheet to reformat your desktop webpages.

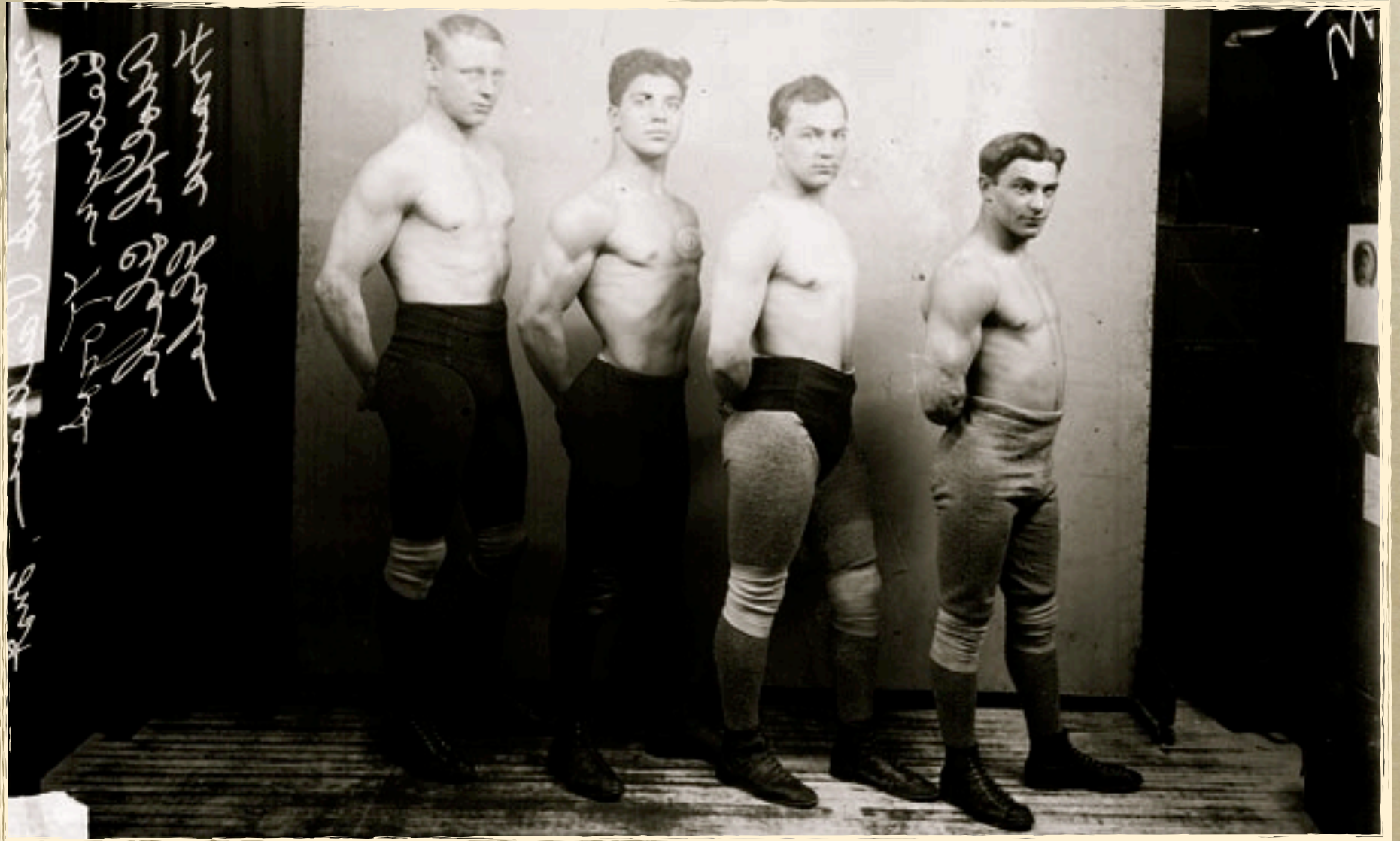
Mobile apps need mobile content.
Features, content, navigation need to be adapted to
non-traditional computing environments.

Content, or at least hierarchy, should usually differ on phone vs desktop,
often for tablet too.
Because my needs are different.

CONSISTENCY



CORE CONTENT ACROSS ALL APPS



Does not mean LESS content on mobile devices.
Kneejerk assumption Lmobile means lite version of desktop.
85% expect mobile experience to be at least as good as desktop.

Our job is not to willy-nilly strip out useful features.
We've all had exp: mobile browser bumps us to mobile website.
Feature/content we want "helpfully" stripped out.
Hunt for link to full desktop version.

There's myth of single mobile culture or context.
Mobile users are not always distracted and in a rush.
Mobile not just on go: couch, kitchen, bed.

using small screen \neq wanting to do less.
Like saying that because paperbacks have smaller pages, you have to remove entire chapters. It makes no sense.

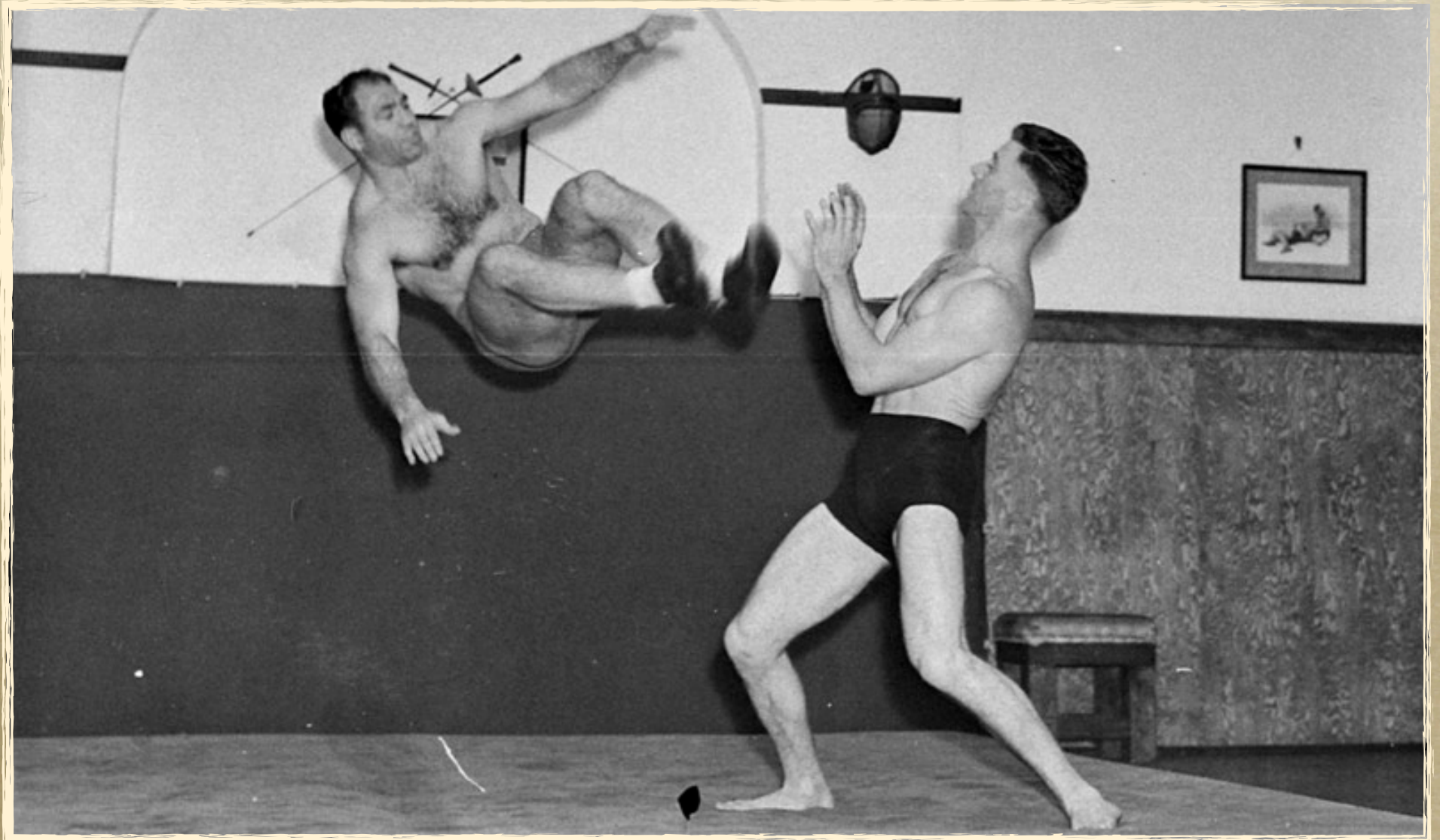
[slow] Don't confuse context with intent.

So: When I say consistency of content, DON'T mean the same design.
I mean similar content and messaging.
Your mobile native app may do things slightly differently from your mobile website.
An iPad app will likely address different needs than a mobile app.
Desktop different from all.

Core content should all be there, but it's hierarchy may change.
Just don't arbitrarily give me LESS.

ON THE GO

★ MIND YOUR MINDSETS! ★



You gotta think mobile.
As you consider your mobile app,
consider broad mindsets that cut across all mobile cultures.

What makes your app mobile? Why do I want to use it on the go?
Or more specifically: non-traditional computing environments.
Not just out and about: couch or kitchen.

Wherever you are: people fire up mobile apps in
one of three mindsets.

MINDSETS



THREE MOBILE MINDSETS!



★ I'M MICROTASKING

★ I'M LOCAL

★ I'M BORED

No matter what tech you use, how do you organize for mobile content and features?

Here's a framework I find useful.

By and large we launch mobile apps in one of three mindsets.

Microtasking = that commonly understood mobile user.

In a rush.

Microtasks are quick dashes of activity, frequently wedged between other tasks.

Efficiency: identify the most frequent recurring tasks, the primary thing your app does, and optimize the hell out of it to help people get it done quick.

I'm local. What's around or what's in front.

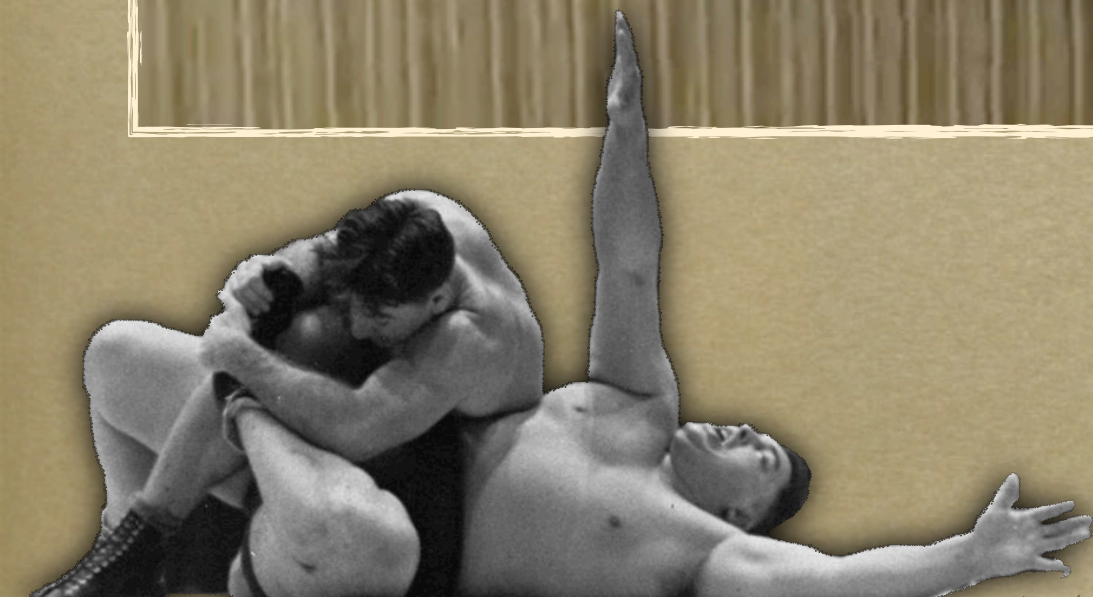
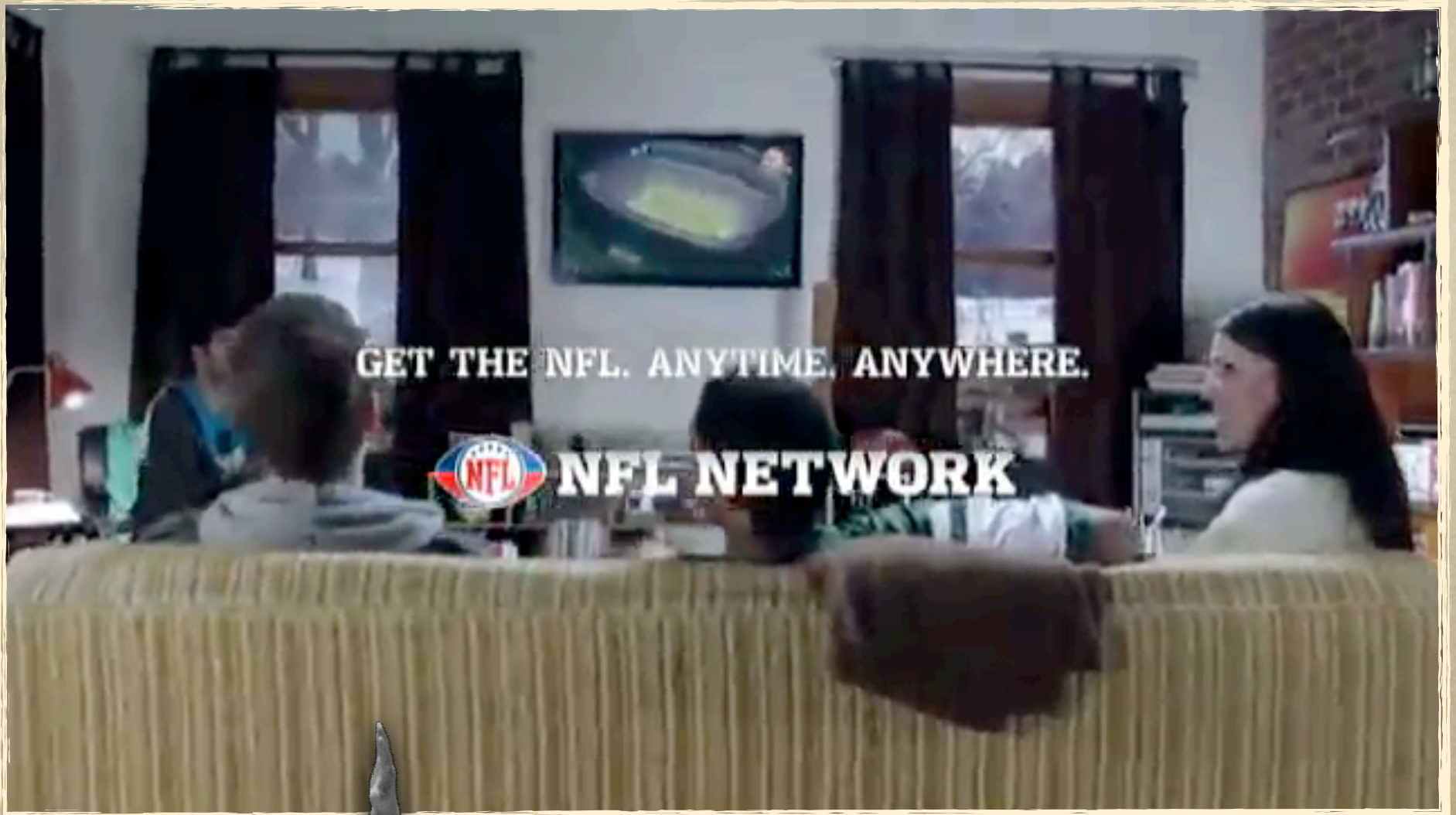
How can you use the phone's sensors to add important context to tasks and activities, to speed information input by scanning bar code with camera.

I'm bored. Thing that makes phone great for microtasking, same as what makes great for moments of dreary boredom. It's always there to entertain you. Whether that's a game, a book, a news feed.

But another way to put it is, you have my full attention. I will spend the next 30 minutes on my phone, because I am undistracted. This allows leisurely stroll through data. Allows close reading.

Efficient microtasking important,
but so is this high-attention "I'm bored" mindset.
Give people something to spend time with.

Best apps do all three.



But again, that doesn't mean that you should do less or even anything different.

There's a growing mainstream expectation that you can simply get all your content from any device.

We expect content to flow seamlessly, follow us throughout the day.

This ad for NFL.com, the national football league, is a crisp illustration of that shift.

[play]

We access the same content across multiple devices.

Phones, PCs, tablets, X-boxes, tv boxes

Kindle and Netflix

That expectation is going to spread to even the most modest apps.

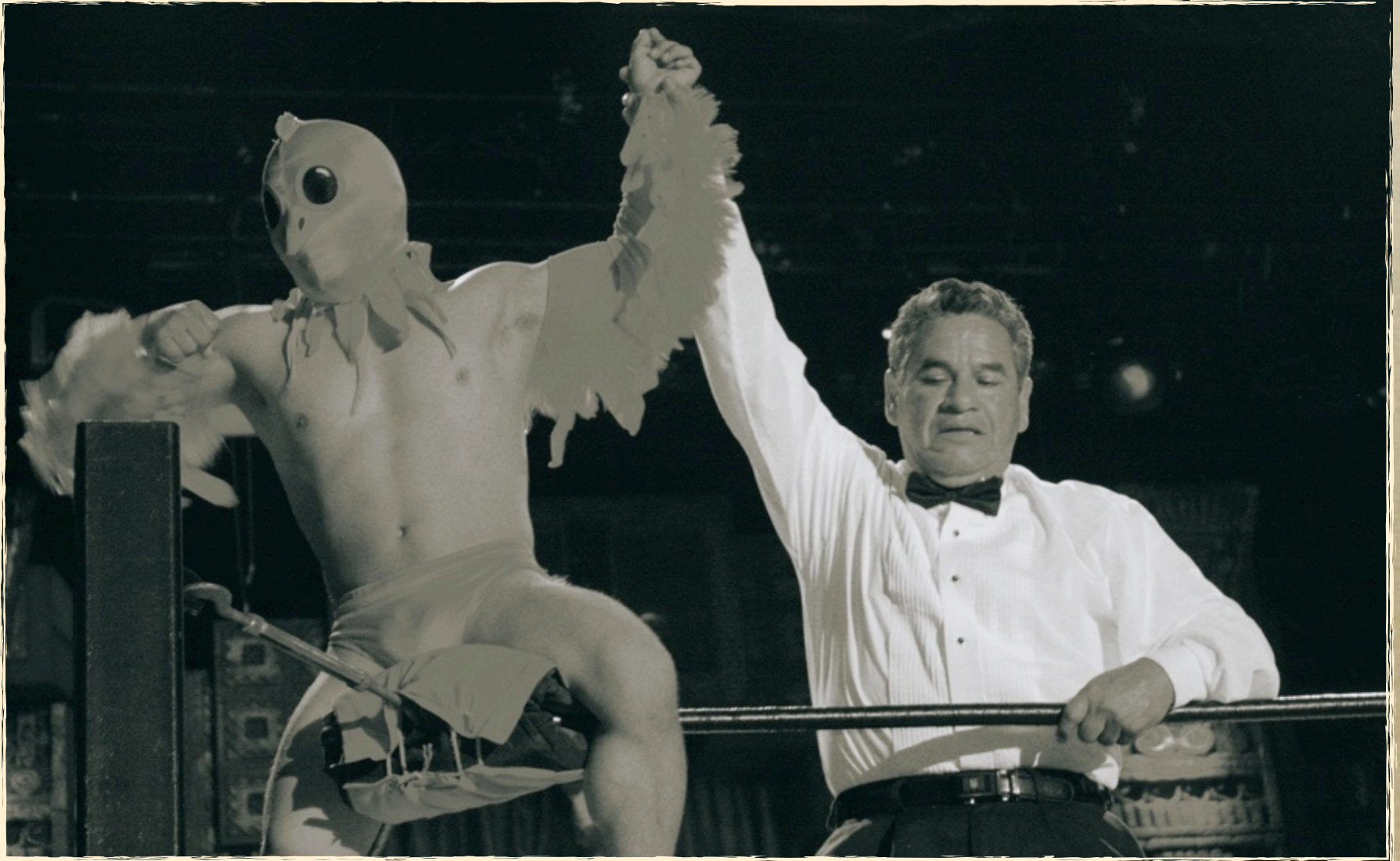
I want my stuff everywhere.

This finally brings us to the REAL winner of our cage match.

THE CHAMP!



#WINNING



Got all these devices.
You're going to be building several front ends.
For iPhone, for iPad, for web, for desktop apps, for Android, for SMS.
Step back from that whirlwind for a moment.
Stop focusing so much on those APPS.

You have to start with your CONTENT.

Look across the entire range of devices and interfaces,
and seek out the commonalities.
What's range of service you want to offer.

Tying all these interfaces together is you...
your company, your goals, your service.
And that's embodied digitally by the real champ...

BIG DADDY!



THE **API** WINS!



The Big Daddy! The API!

The winning approach here is thinking about your interfaces as a spectrum of apps that plug into a single wellspring of service.

Build a common back end that can serve all of these interfaces, that lets you turn and pivot to each culture, to each technology, to each device.

So:

What are the digital services and information that you provide?
Who are you? Who are your customers? What do you provide for them?

Not mobile first, not desktop first. Content and API first.



That means we're all cloud developers.

I don't care how you build it—native or web:

[slow]just about every app should be a web client of some kind.

Here's the thing:
An app is not a strategy. It's just an app.

In the end: it's not your mobile web strategy
or your app-store strategy.
or your tablet strategy
or your desktop strategy.

IT'S JUST PLAIN STRATEGY

Who are you to your customers?
What do you do for them?
How can you embody that digitally
and provide that service or information wherever they might need it?
And how can you be ready for the future?

Your API has to be agnostic to all of these platform machinations.
We came out of the 1990s with our data locked up in legacy systems.
Now we're turning out of the early web era with our data locked up in web content management systems.
Content that's stored as HTML.

Moving forward we need clean content repositories and APIs
to deliver more neutrally formatted content.

So there's some important design work to be done behind the scenes.
Careful decisions about the services we want to provide and how to build the data to send.

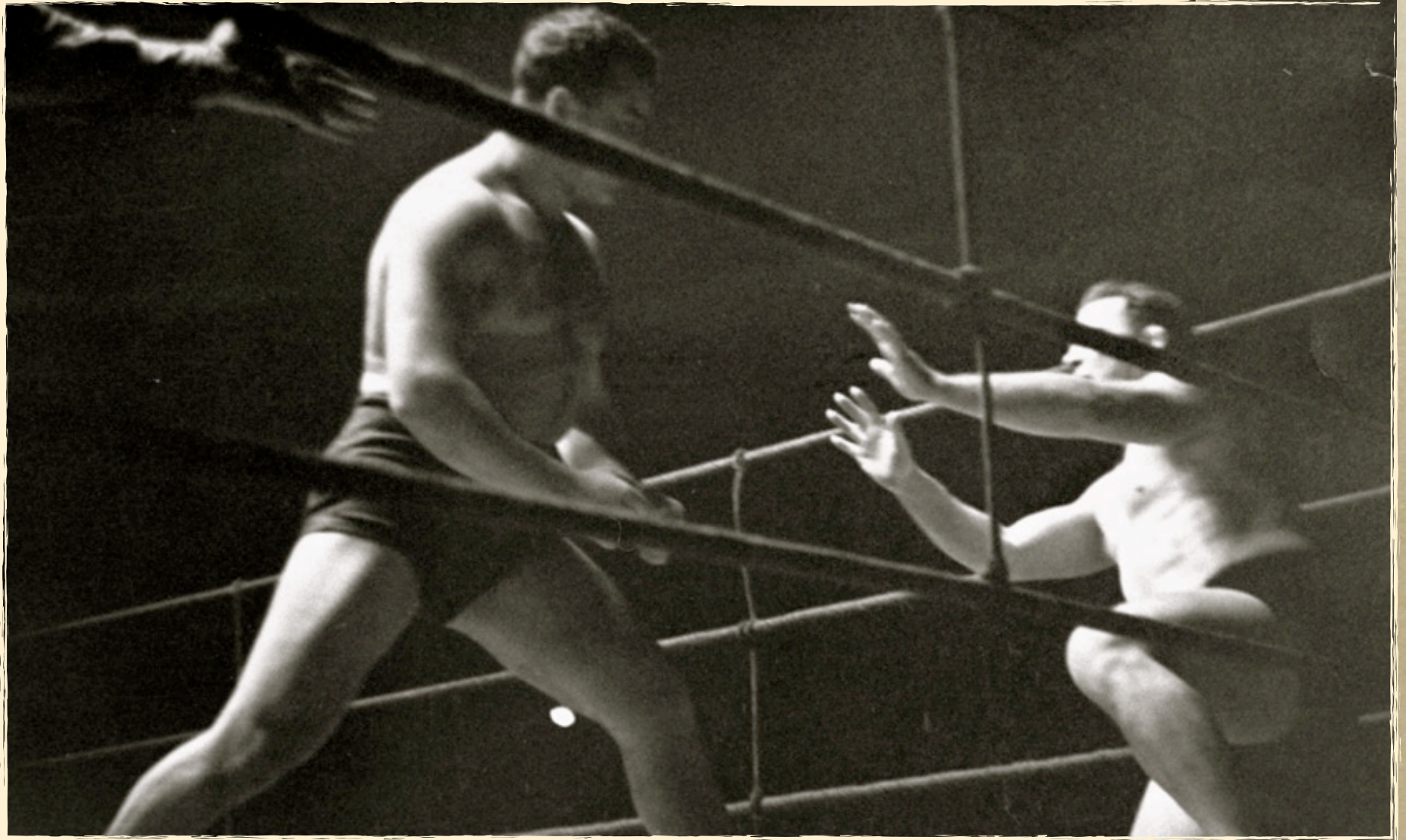
Because here's the thing:
This may seem really scary right now.
No way to keep up with all of these devices and platforms.

Friends, this is just the beginning.
So we have to stop thinking so rigidly about these separate independent channels.
Start thinking of them as a spectrum across which our content flows seamlessly.

There's no single mobile culture, no single mobile platform.
We as consumers know this. So let's be sure we do the same as designers.

FEAR NOT

★ DON'T BE SCARED! ★



And finally, don't be scared.

Easy to get intimidated by these rapid changes,
by this crazy lineup of devices and platforms.

[slow]
We are inundated by screens,
a rain of little glowing rectangles.

That's not going to change, so we can't sweat it.
This is simply the environment we work in now.
So accept it, understand it. And then rock it.

Go be awesome.



**I LOVE
YOU GUYS**

JOSH CLARK
@GLOBALMOXIE
WWW.GLOBALMOXIE.COM