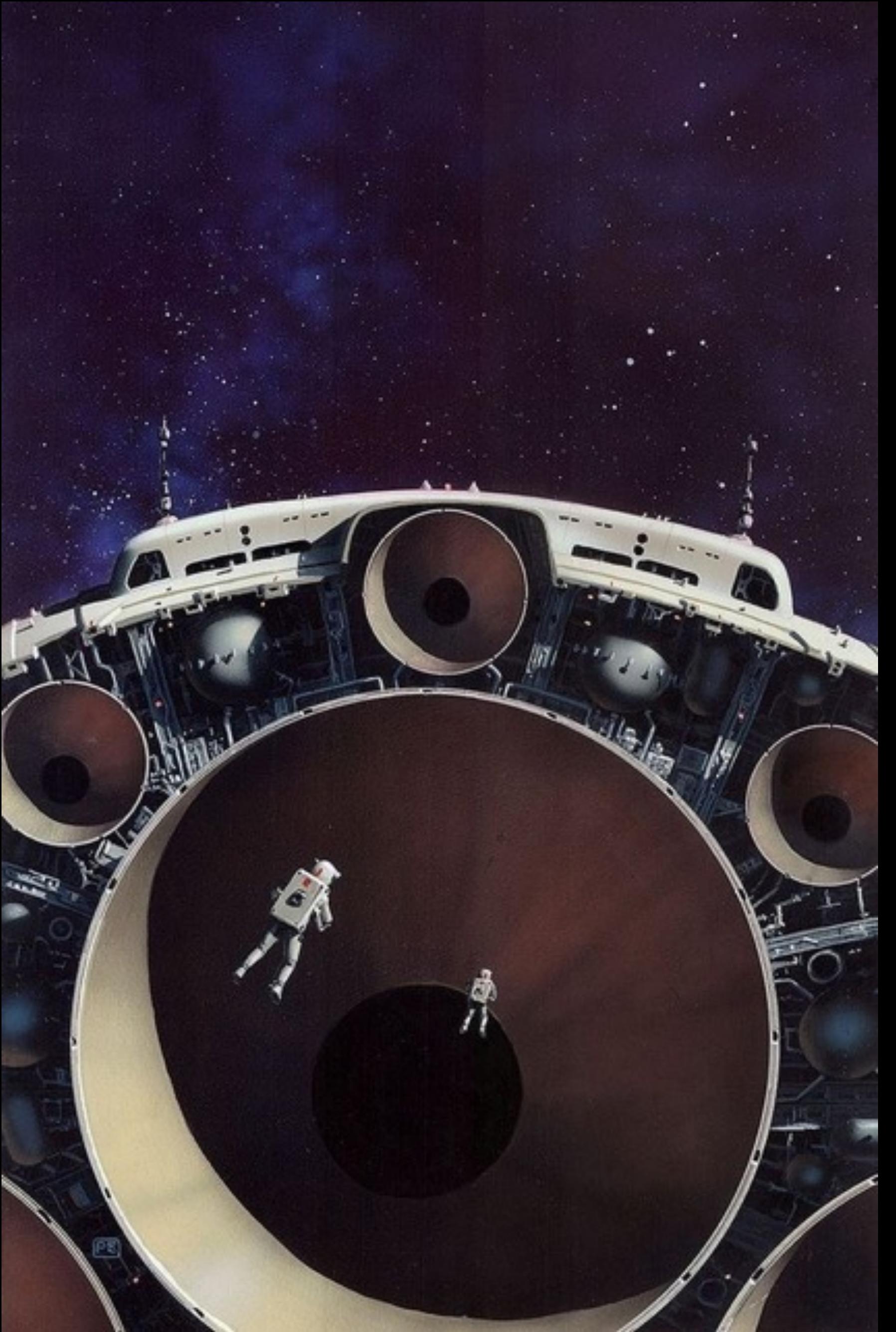
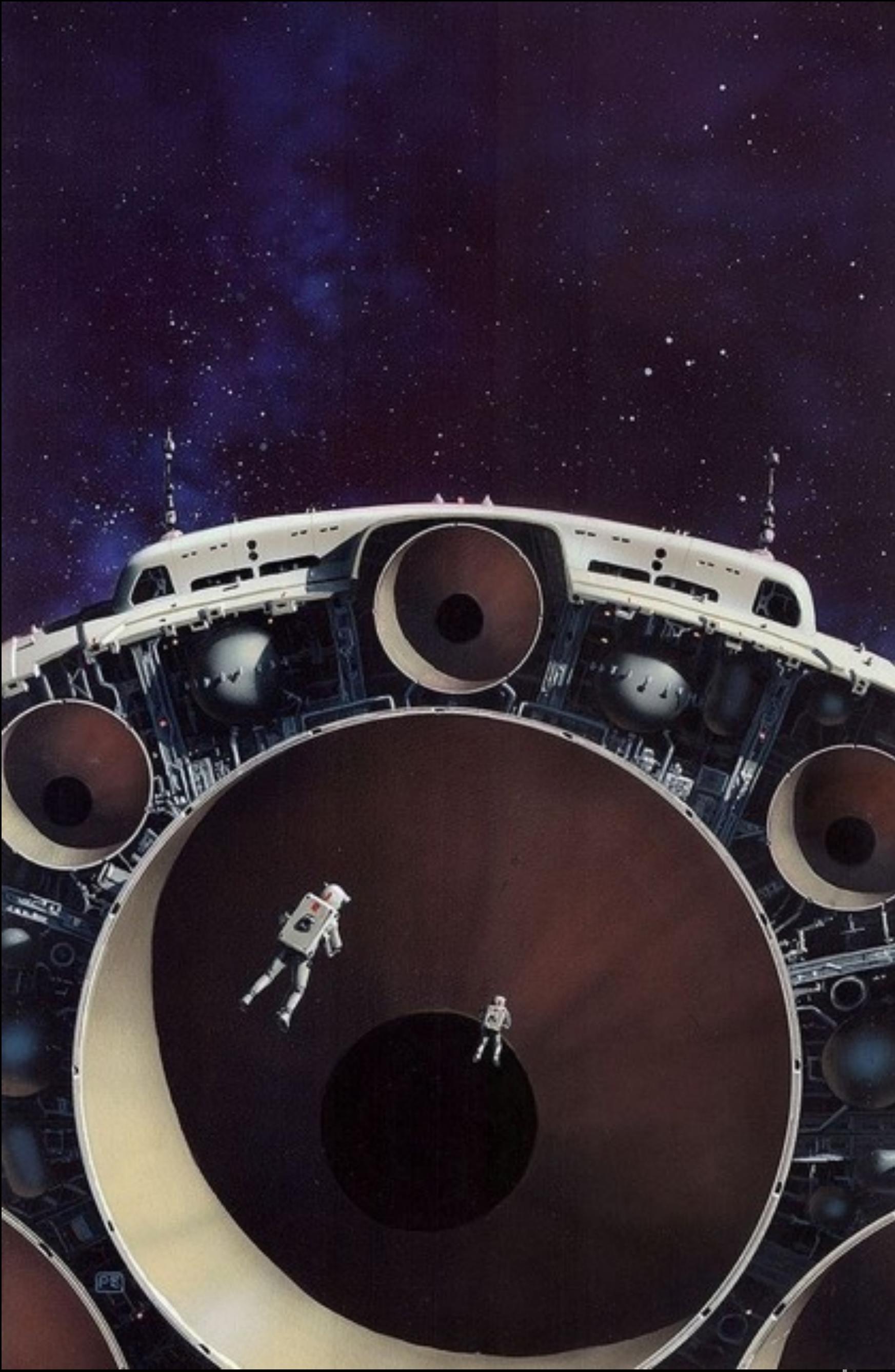


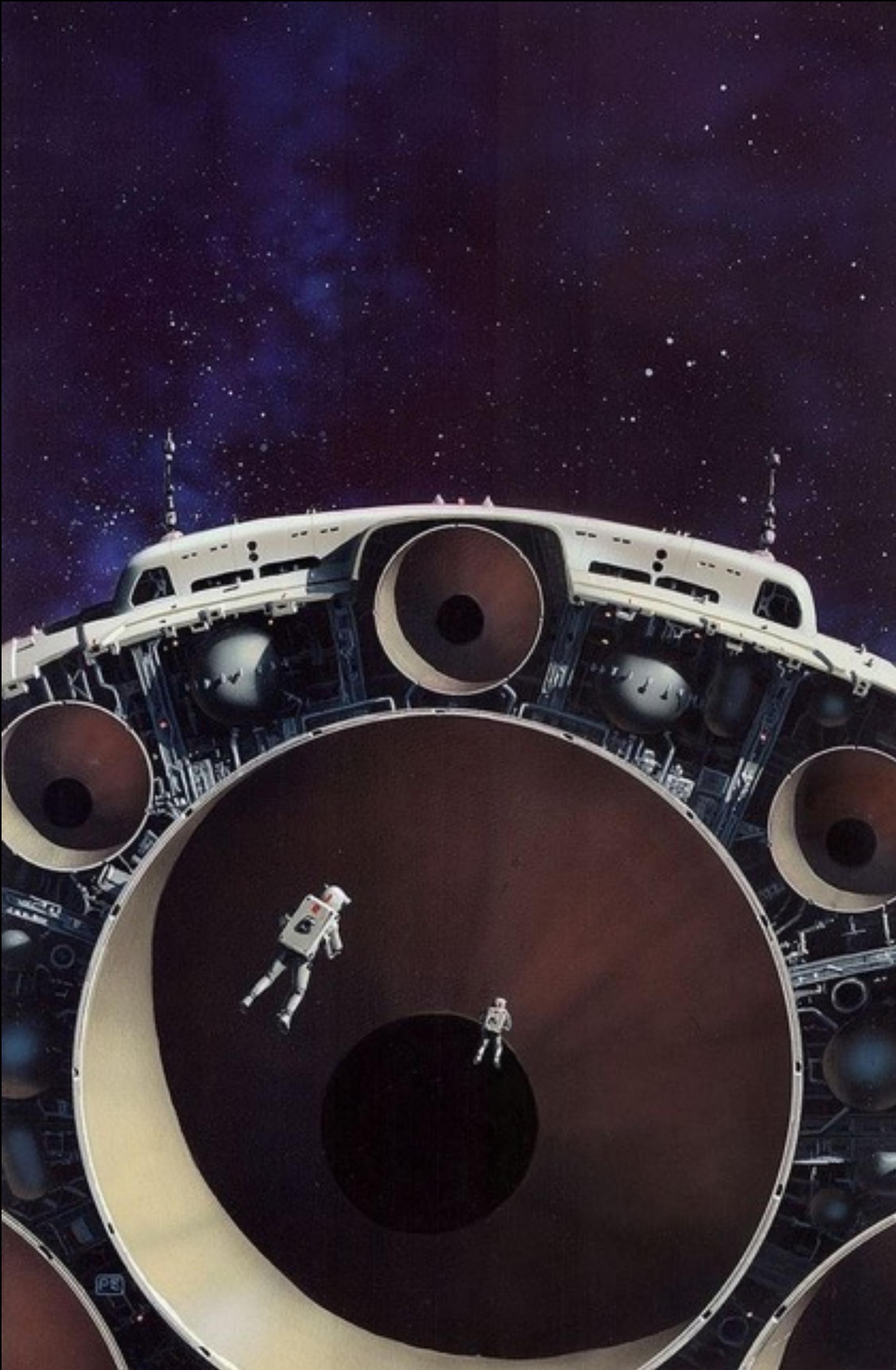


*Designing With Web Standards
in 2016*

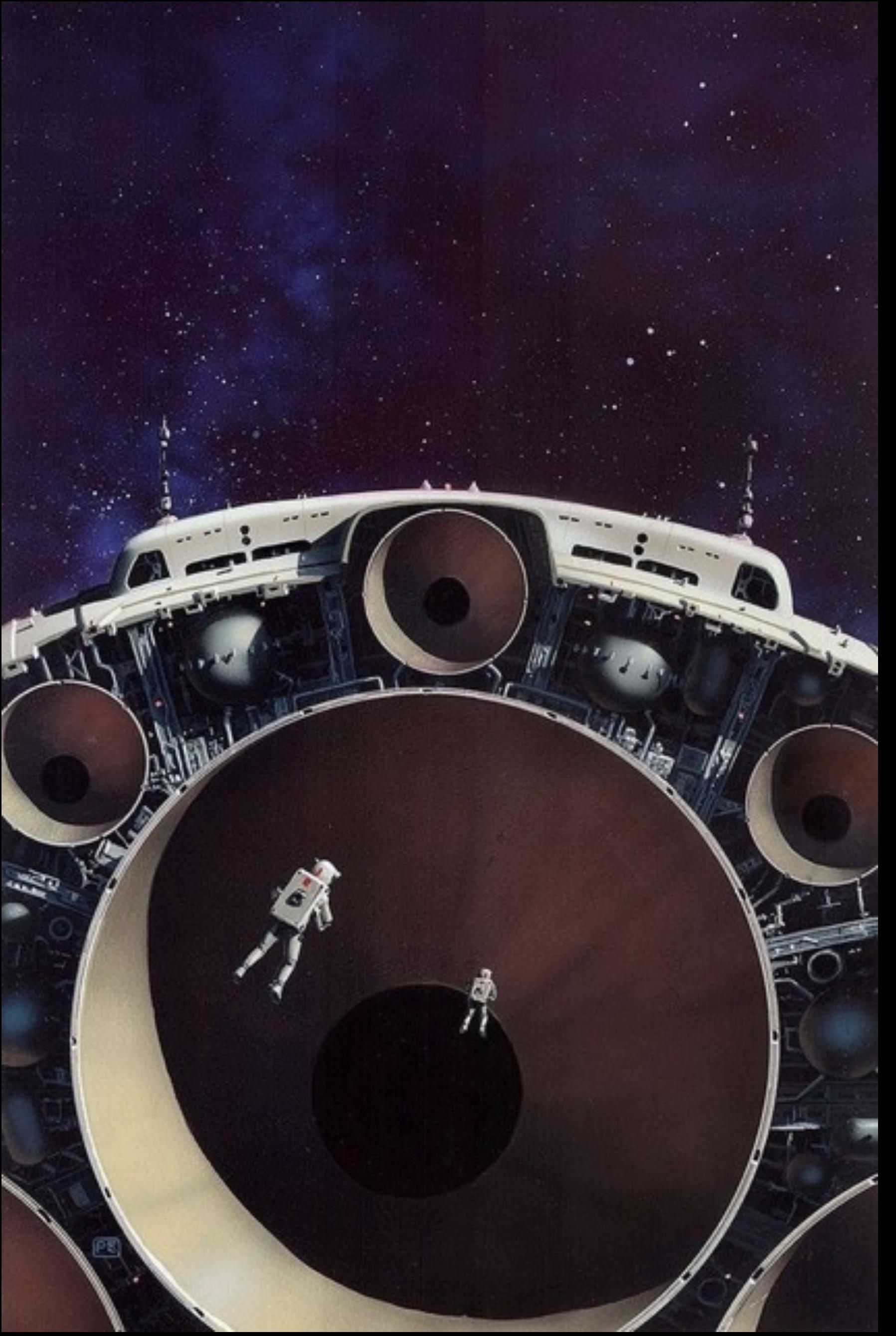


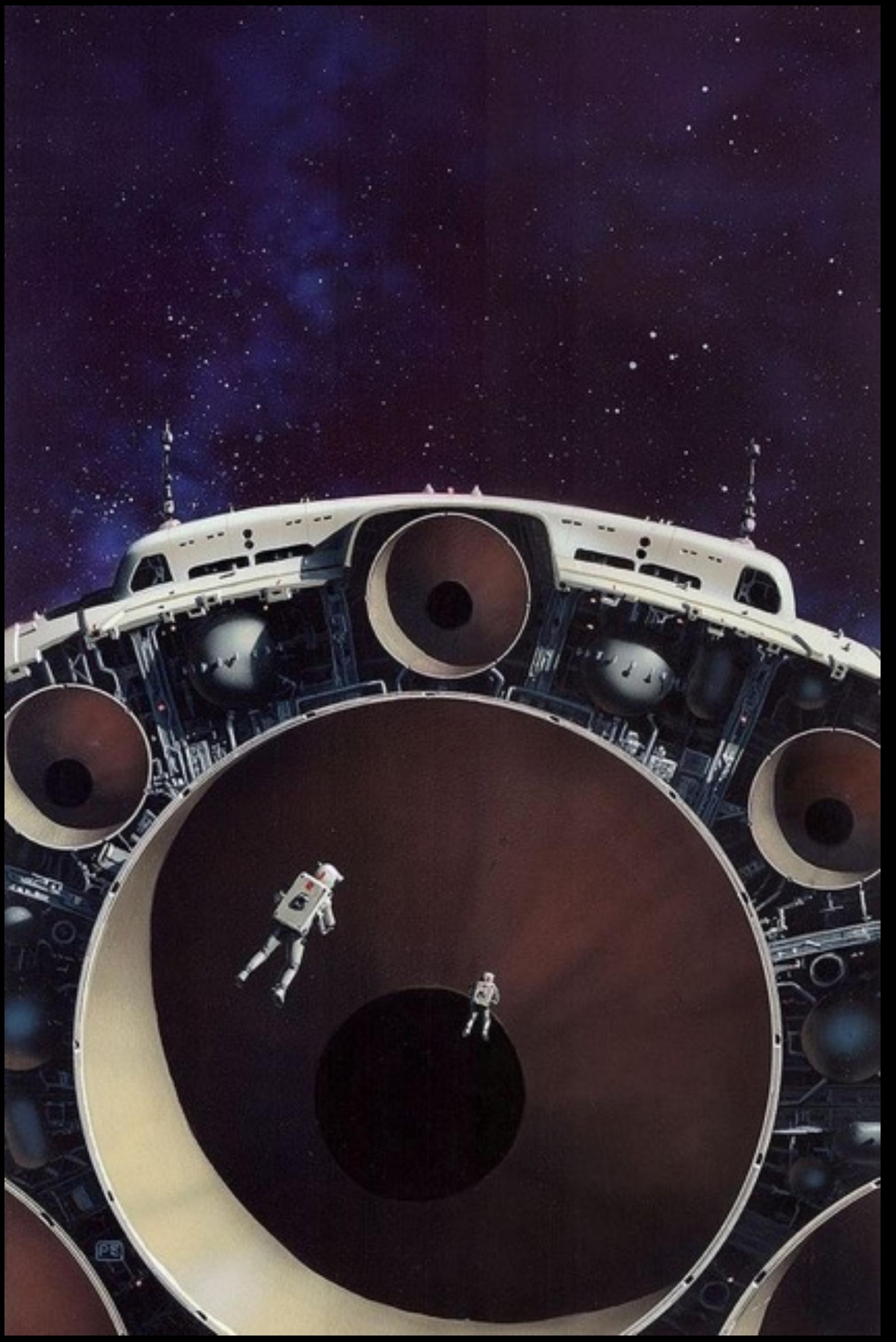
A Challenging Time

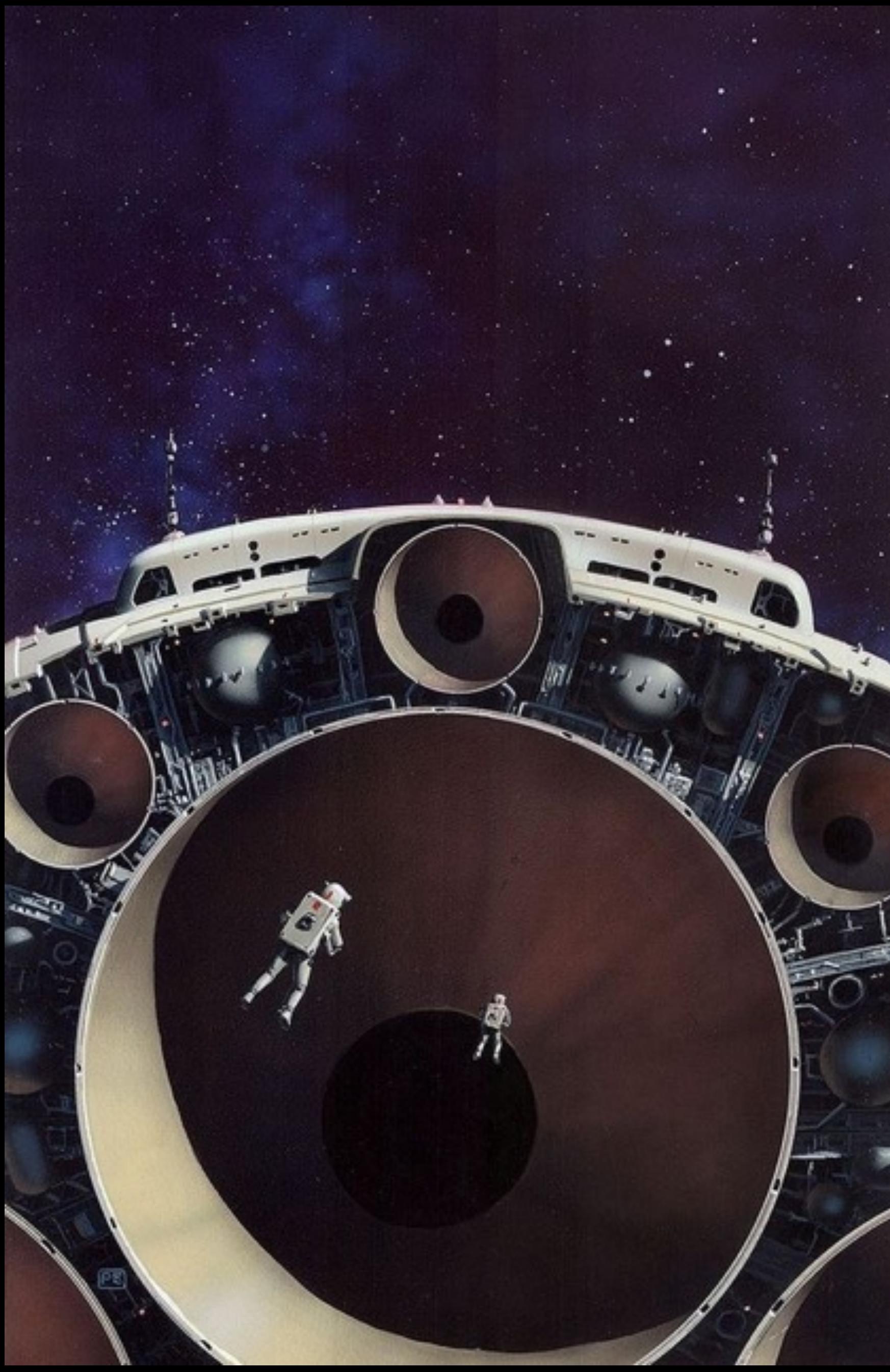


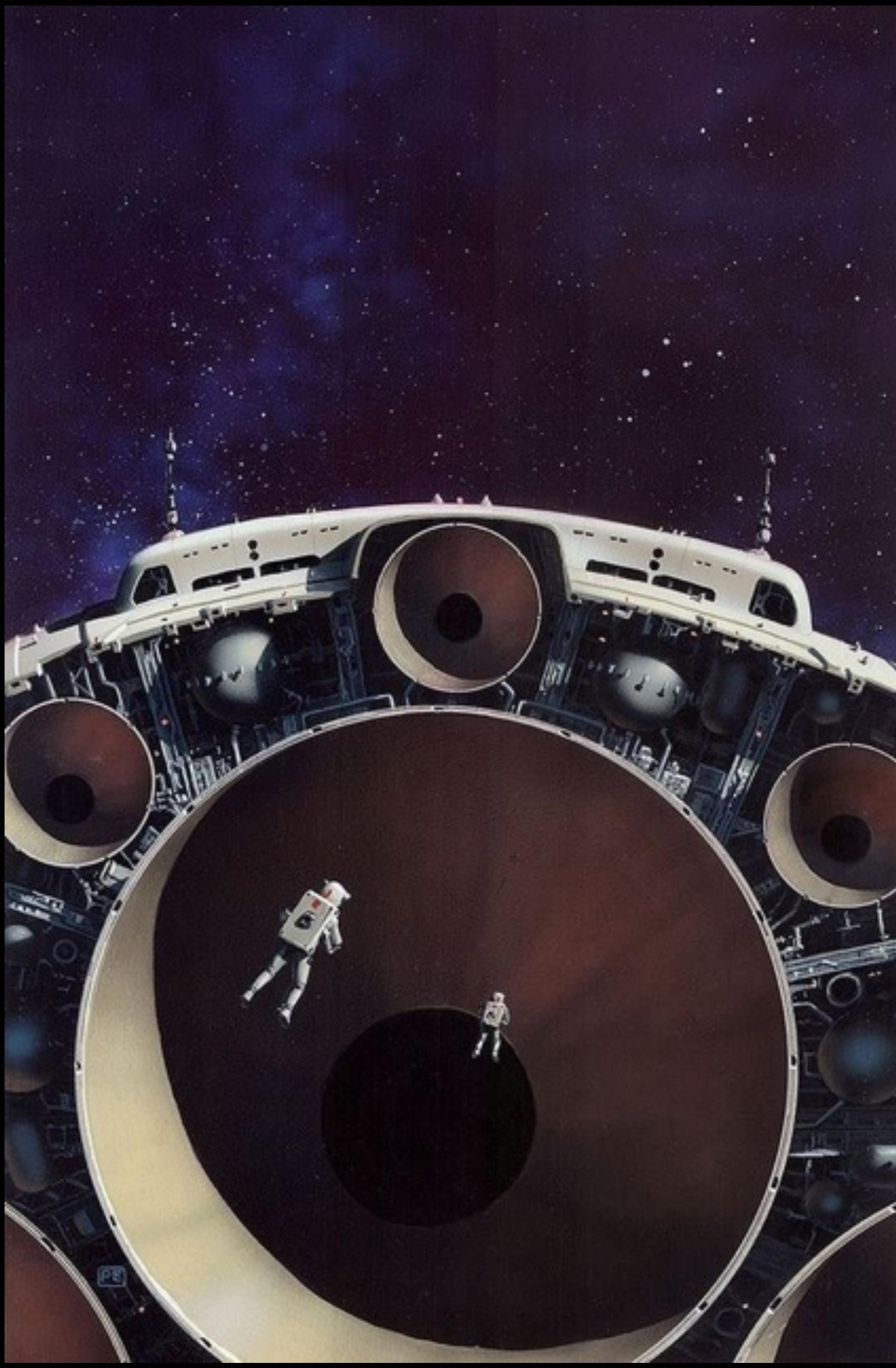


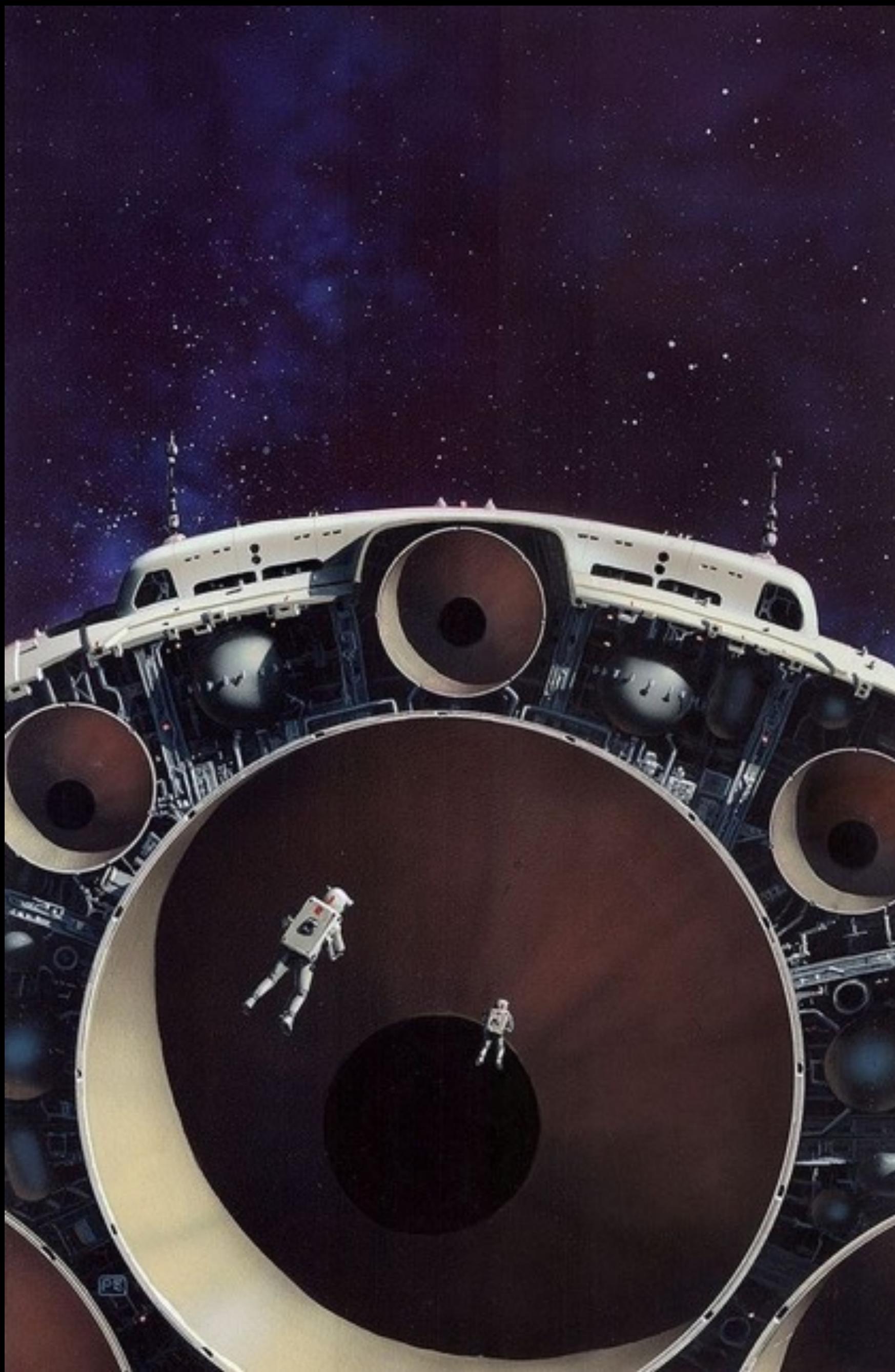
A Challenging Time











NVIDIA - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites

Address http://www.geforce.com/GamesandApps/games/battlefield-3/GPUAnalyzer Go

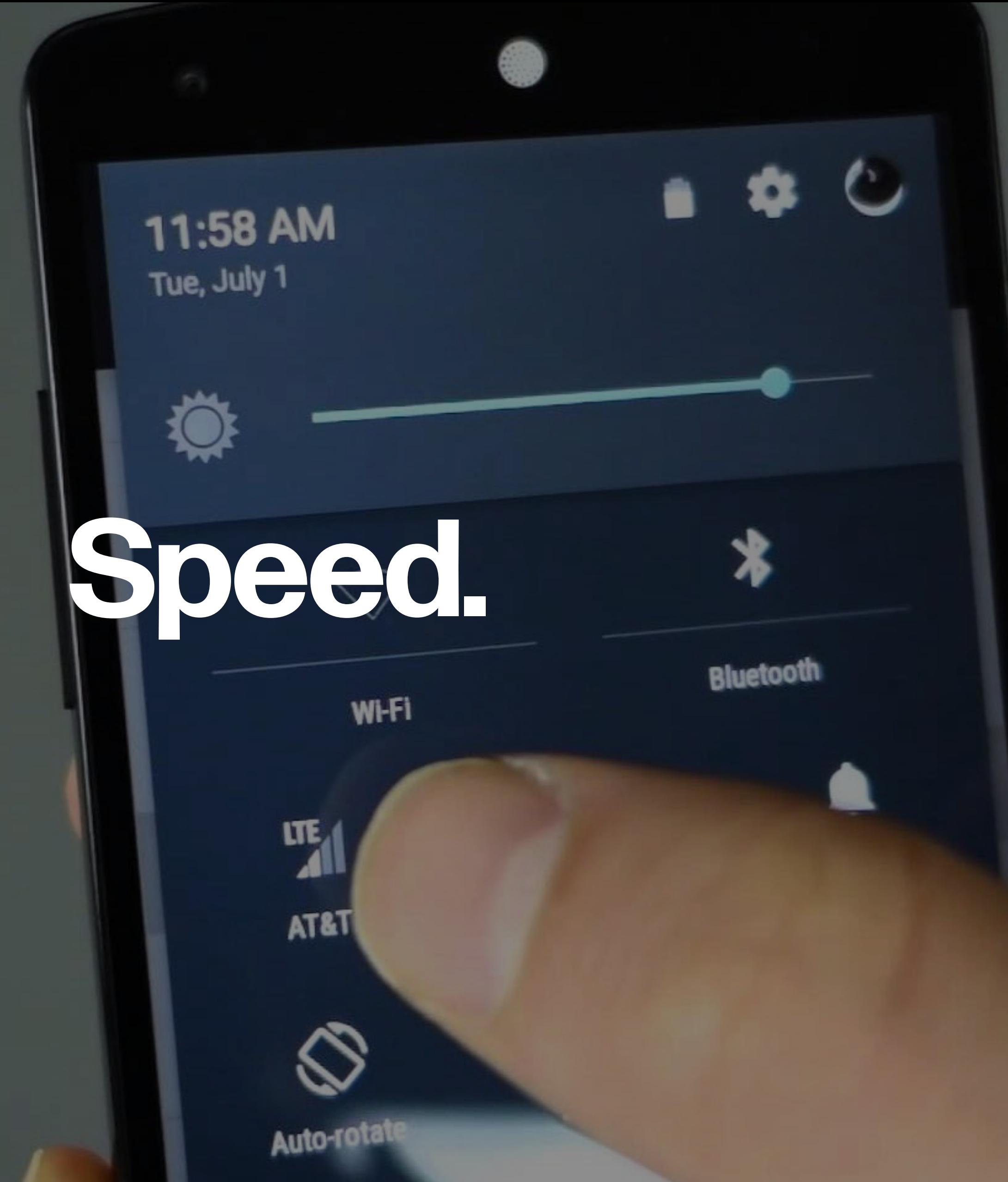


Internet Explorer 6, you cannot be serious?

We don't support old technologies and neither should you!
Please upgrade to the latest version of Internet Explorer or use a different browser to visit our site.

Done Internet

Speed.





Speed.



- ▲ You need to set the widths on your cells. Outlook doesn't know how wide to make them.
- Your header code should look something like this. I simplified it a bit for you:
 - ▼

```
<table border="0" cellpadding="0" cellspacing="0" width="100%>
  <tr>
    <td width="40%" align="center" valign="top" style="padding-top: 10px; padding-bottom: 10px;">
      
    </td>
    <td width="60%" align="right" valign="center" style="padding-top: 10px; padding-bottom: 10px;">
      &nbsp;&nbsp;
      &nbsp;&nbsp;&nbsp;
      &nbsp;&nbsp;&nbsp;
      
    </td>
  </tr>
</table>
```

Because you have ample space for the social icons, I just put them inline. You could keep them in cells though, but it was more code to do it that way.

Email Standards Project

- [About the Email Standards Project](#)
- [Why Web Standards Matter for Email](#)
- [What You Can Do](#)
- [Current Email Client Standards](#)
- [ESP Blog](#)

State of the clients

Results based on our [acid test](#)

- [AOL Webmail Full Report »](#)
- [Apple iPhone, iPad & Touch Full Report »](#)
- [Apple Mail Full Report »](#)
- [Apple MobileMe Full Report »](#)
- [Eudora \(Penelope\) Full Report »](#)
- [Google Gmail Full Report »](#)
- [Lotus Notes 8 Full Report »](#)
- [Entourage Full Report »](#)
- [Outlook 2007 Full Report »](#)
- [Thunderbird Full Report »](#)
- [Windows Live Hotmail Full Report »](#)
- [Windows Live Mail Full Report »](#)
- [Windows Mail Full Report »](#)
- [Yahoo! Mail Beta Full Report »](#)

The Email Standards Project works with email client developers and the design community to improve web standards support and accessibility in email.

Our goal is to help designers understand why web standards are so important for email, while working with email client developers to ensure that emails render consistently. This is a community effort to improve the email experience for both designers and readers alike.

What you can do to support the Email Standards Project

Whether it's spreading the word about our cause, helping out with additional research or telling us the secret handshake for the Gmail team, we'd love your help. This truly is a community effort and every little bit helps.

[Learn more about how you can help »](#)

- ▲ You need to set the widths on your cells. Outlook doesn't know how wide to make them.
- Your header code should look something like this. I simplified it a bit for you:
 - ▼

```
<table border="0" cellpadding="0" cellspacing="0" width="100%>
  <tr>
    <td width="40%" align="center" valign="top" style="padding-top: 10px; padding-bottom: 10px;">
      
    </td>
    <td width="60%" align="right" valign="center" style="padding-top: 10px; padding-bottom: 10px;">
      &nbsp;&nbsp;
      &nbsp;&nbsp;&nbsp;
      &nbsp;&nbsp;&nbsp;
      
    </td>
  </tr>
</table>
```

Because you have ample space for the social icons, I just put them inline. You could keep them in cells though, but it was more code to do it that way.

“If there is to be a loser in the battle of the layouts, my hope is that it will be the layout frameworks that tie our design to our markup. They have been a necessary placeholder while we waited for a true web layout system, but I believe that in a few years time we’ll be easily able to date a website to circa 2015 by seeing <div class=”row”> or <div class=”col-md-3”> in the markup.”

—Rachel Andrew

<https://24ways.org/2015/grid-flexbox-box-alignment-our-new-system-for-layout/>



Is CSS Necessary?

It's a question that has been asked by many web developers over the years. Is CSS necessary? The answer is yes, but it depends on what you're trying to achieve.

CSS stands for Cascading Style Sheets, and it's a language used to describe the presentation of a document written in HTML. It allows you to define styles such as colors, fonts, and layout for different elements on a page.

While it's not strictly necessary to use CSS, it can greatly simplify the process of styling your website. By separating the presentation from the content, you can make it easier to maintain and update your site. You can also reuse styles across multiple pages, which can save you time and effort.

However, there are some cases where CSS might not be the best choice. For example, if you're creating a simple one-page website with basic styling, you might be better off using inline styles or a preprocessor like SASS or LESS. These tools can help you write more organized and maintainable CSS code.

In conclusion, while CSS is not strictly necessary, it can greatly simplify the process of styling your website. It's a valuable tool that can help you create professional-looking websites with ease.

Is CSS Necessary?

Then: TABLE LAYOUTS

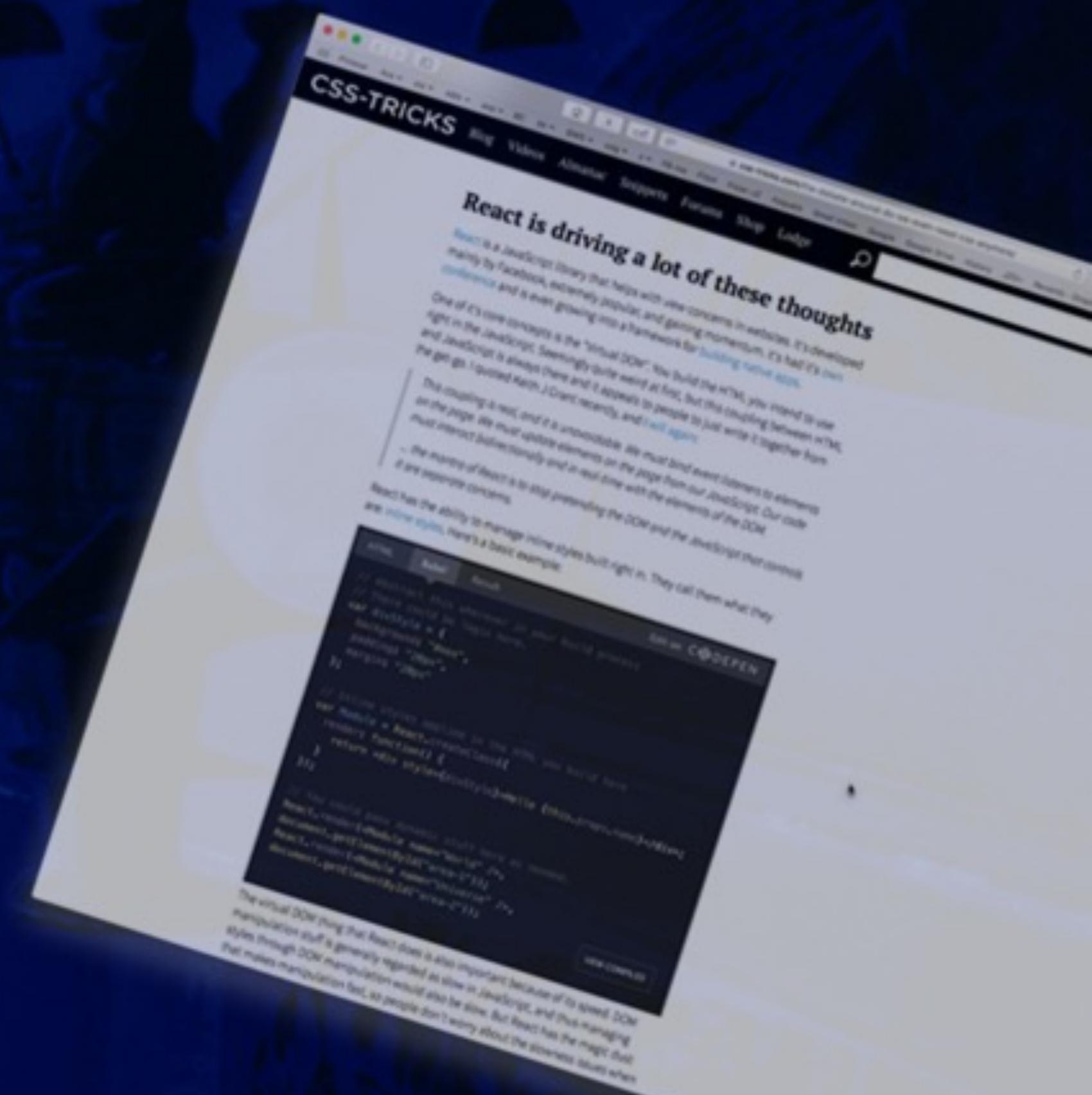
Is CSS Necessary?

“We’re standardized on Flash.”

Is CSS Necessary?

NOW: REACT JS

<https://css-tricks.com/the-debate-around-do-we-even-need-css-anymore/>



Bloat: THEN

DW

Bloat: NOW

A screenshot of a web browser displaying the Bootstrap website. The browser's address bar shows 'getbootstrap.com'. The page features a large purple header with the text 'Bloat: NOW' in white. Below the header is a navigation bar with links for 'Bootstrap', 'Getting started', 'CSS', 'Components', 'JavaScript', 'Customize', 'Themes', 'Expo', and 'Blog'. A blue banner at the top of the main content area says 'Aww yeah, Bootstrap 4 is coming!'. The main content area has a purple background and contains a large white square with a bold white letter 'B' in the center. Below this is a paragraph of text: 'Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile first projects on the web.' At the bottom of the page is a button labeled 'Download Bootstrap'.

(113) Pinterest: Discover and save creative ideas.

Bootstrap · The world's most popular mobile-first and responsive front-end framework.

Aww yeah, Bootstrap 4 is coming!

Bootstrap Getting started CSS Components JavaScript Customize Themes Expo Blog

B

Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile first projects on the web.

Download Bootstrap

Currently v3.3.6

There's nothing new under the suns.

The Dawn of Modern Web Design

Web Standards for
the Web

Jeffrey Zeldman
with Eric Meyer

© 2001 ADOBE



designing with web standards

third edition

jeffrey zeldman
with ethan marcotte



99.9% of Websites Are Obsolete

An equal opportunity disease afflicts nearly every site now on the web, from the humblest personal home pages to the multi-million-dollar sites of corporate giants. Cunning and insidious, the disease goes largely unrecognized because it is based on industry norms. Although their owners and managers may not know it yet, 99.9% of all websites are obsolete.

99.9% of Websites Are ~~Obsolete~~ **Problematic**

An equal opportunity disease afflicts nearly every site now on the web, from the humblest personal home pages to the multi-million-dollar sites of corporate giants. Cunning and insidious, the disease goes largely unrecognized because it is based on industry norms. Although their owners and managers may not know it yet, 99.9% of all websites are obsolete.

The perils of browser sniffing

All browsers have a user agent (UA) string, a short string of text that contains the browser's name, version, and platform, and acts as a digital fingerprint for the browser. Web servers frequently log the UA, providing valuable information to site owners hoping to better understand their audience. But all too frequently, site owners use JavaScript or server-side code to parse the UA string so they can serve platform-appropriate or version-specific content. The problem with this approach is that the User Agent string is unreliable. Depending on the user's security or network settings, a browser may not relay its UA to the server, causing these browser-sniffing scripts to fail, preventing users from reaching their destination.

Furthermore, all modern browsers allow the user to change the UA, whether

That a company as smart as Volkswagen misses this opportunity says everything you need to know about the entrenched mindset of brand managers who treat the web as if were a print ad, and of developers who hold “backward compatibility” in higher esteem than reason, usability, or their own profits.

The Hidden Cost of Bloated Markup

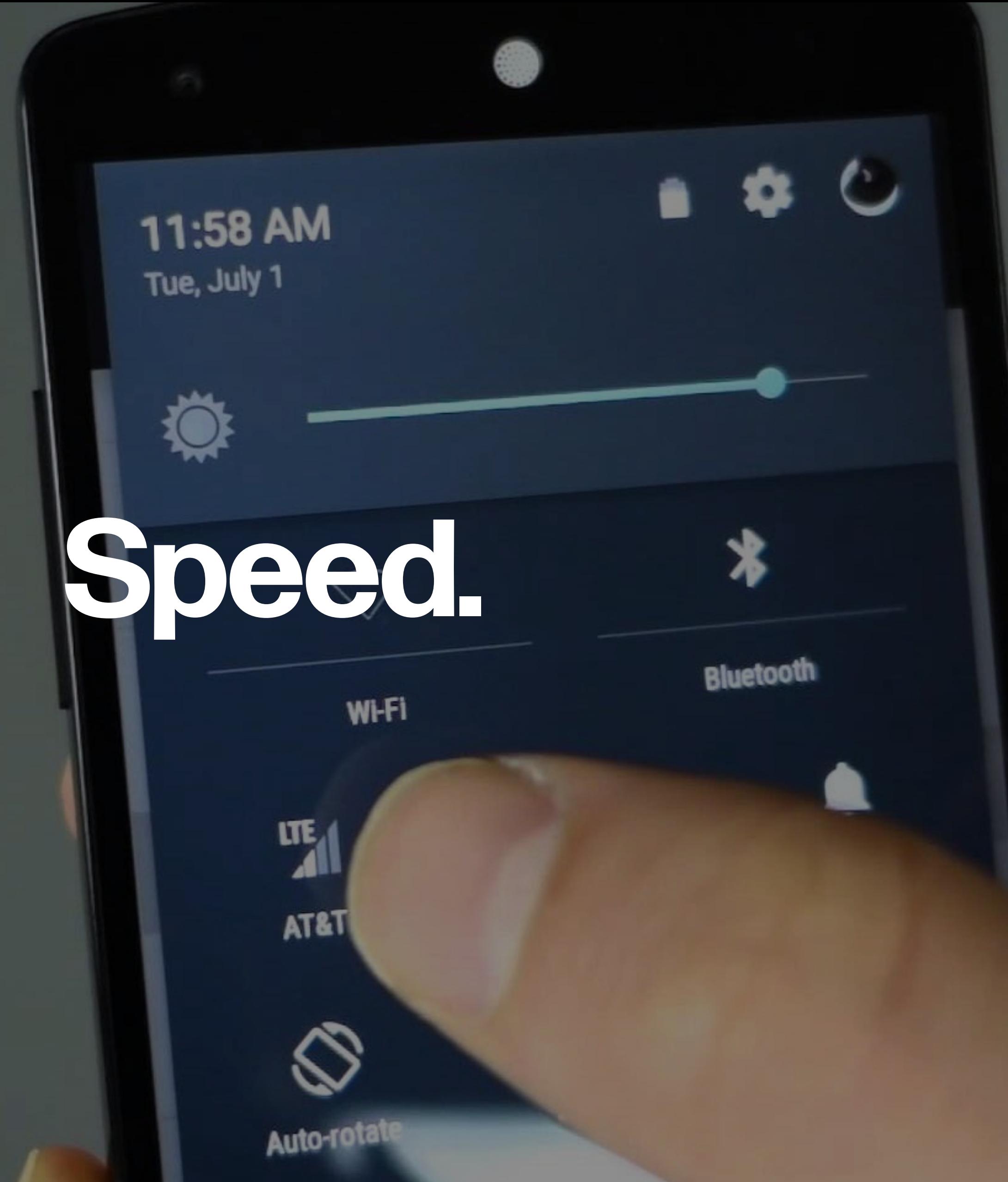
Suppose the code and markup on one old-school web page weighs in at 60K. Say that, by replacing outdated font tags and other presentational and proprietary junk with clean, structural markup and a few CSS rules, that same page can weigh 30K. (In our agency’s practice, we can often replace 60K of markup with 22K or less. But let’s go with this more conservative figure, which represents bandwidth savings of 50%.) Consider two typical scenarios, detailed next.

T1 Terminator

Scenario: A self-hosted business or public sector website serves a constant stream of visitors—several hundred at any given moment. After cutting its page weight in half by converting from presentational markup to lean, clean, structural XHTML, the organization saves \$1,000 a month.

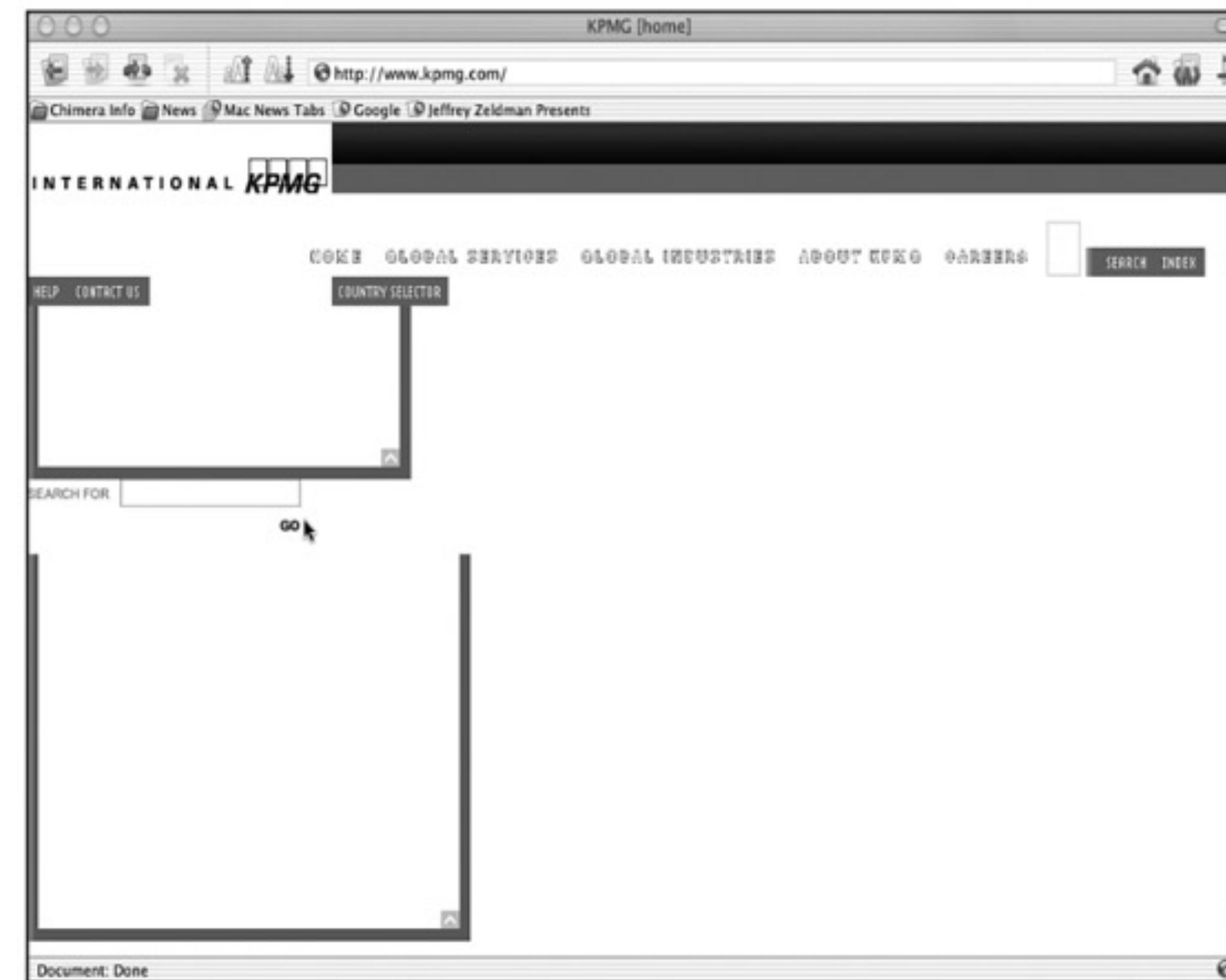
How it works: To serve its audience prior to the conversion, the self-hosted

Speed.



Blocking Users Is Bad for Business

While some companies undercut their own profitability trying to ensure that even the oldest browsers display their sites exactly as new browsers do, others have decided that only one browser matters. In a misguided effort to reduce expenses, many sites are designed to work only in Internet Explorer, and sometimes only on the Windows platform, thus locking out 15–25% of their potential visitors and customers [1.6, 1.7, 1.8, 1.9, 1.10].



1.6

The home page of KPMG (www.kpmg.com), circa 2003, as seen in Netscape Navigator. Or rather, as not seen in Navigator, thanks to IE-only code.

words, taking countless sites down with it. If you own, manage, design, or build websites, the bell tolls for you.

The Cure

“Write once, publish everywhere,” the promise of web standards, is more than wishful thinking; it is being achieved today, using methods we’ll explore in this book. Although today’s leading browsers finally support these standards and methods, the message has not yet reached many designers and developers, and new sites are still being built on the quicksand of nonstandard markup and code. This book hopes to change that.

Crafted by the members of the World Wide Web Consortium (W3C) and other standards bodies and supported in all post-2000 browsers, technologies like CSS, XHTML, standard JavaScript, and the W3C DOM enable designers to do the following:

One Document Serves All

The Web Standards Project was built with XHTML 1.0 Strict. CSS was used for layout. There is no Palm version or WAP version, although creating such versions was common practice at the time. Multiple versions were not needed; when you design and build with standards, one document serves all.

Figure 2.14 shows [webstandards.org](http://www.webstandards.org/) as seen in a Palm Pilot. Figure 2.15 shows how it looks in Microsoft's PocketPC. Most uncannily of all, Figure 2.16 shows the site working just as fine as you please on a Newton handheld, Apple's long-discontinued proto-PDA. (Think of it as the iPhone's great-grandfather.) Grant Hutchinson, who captured the Newton screenshot, told us: "There's nothing like viewing a modern site using a piecemeal browser on a vintage operating system."

That should be music to the ears of any designer or site owner who wants to reach the greatest number of visitors with the least effort. Strict compliance with (X)HTML and intelligent use of CSS frees designers and developers from the need to create multiple versions.



2.14

(left) The same site on a different day, as seen in a Palm Pilot. Look, Ma, no WAP! Screenshot courtesy of Porter Glendinning (www.g9g.org).

Forward Compatibility Ingredients

- Full separation of structure from presentation and behavior
- Valid CSS used for layout. Tables used only for their true and original purpose: the presentation of tabular data such as that found in spreadsheets, address books, stock quotes, event listings, and so on
- Valid XHTML 1.0 Strict or Transitional (or HTML 4.01 or 5) used for markup
- Emphasis on structure. No presentational hacks in markup (Strict) or as few as possible presentational hacks in markup (Transitional)
- Structural labeling/abstraction of design elements—“Menu” rather than “Green Box”
- DOM-based scripting for behavior: if you need to fork your code, then sniff for object support, not browser versions
- Accessibility attributes and testing

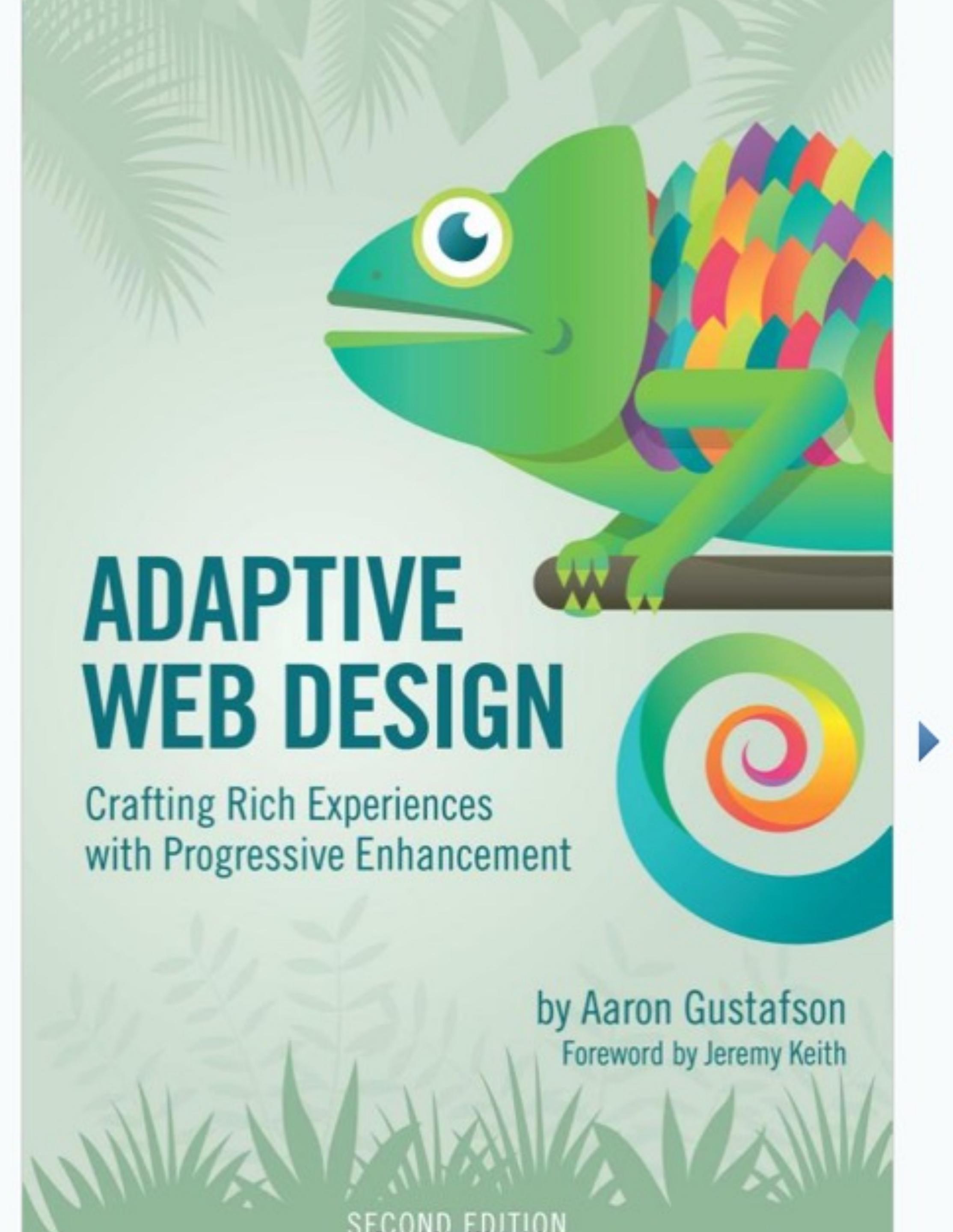
Forward Compatibility Ingredients

- Full separation of structure from presentation and behavior
- Valid CSS used for layout. Tables used only for their true and original purpose: the presentation of tabular data such as that found in spreadsheets, address books, stock quotes, event listings, and so on
- Valid ~~XHTML 1.0 Strict or Transitional (or HTML 4.01 or 5)~~ used for markup **HTML5**
- Emphasis on structure. No presentational hacks in markup (Strict) or as few as possible presentational hacks in markup (Transitional)
- Structural labeling/abstraction of design elements—“Menu” rather than “Green Box”
- DOM-based scripting for behavior: if you need to fork your code, then sniff for object support, not browser versions
- Accessibility attributes and testing

ay, Feb.
4 hrs 54 mins
Day Shipping.
m \$18.22

This Book

GO



SECOND EDITION

> Edit your book

Customers A

Page

Don't Ma
by Steve
★★★★★
Look In

The Elen
by Jesse
★★★★★
Look In

CSS Sec
by Lea V
★★★★★
Look In

Adaptive
by Aaron
★★★★★
Look In

Respon
by Ben F
★★★★★
Look In

Articulat
by Tom I
★★★★★
Look In

UX Strat
by Jaime
★★★★★
Look In

Design S
by Richa
★★★★★
Look In

JavaScri
by Jon D
★★★★★
Look In

Eloquent

chapter three

Gentle Persuasion

This may be the shortest book chapter you'll ever read. It is certainly the shortest thing I've written, if you don't count that note

HTML, CSS, and JavaScript, it will always be there to fall back to you.

Objection: “We are standardized on IE6.”

What to say: “Wonderful! Then we can use standard HTML, CSS, and JavaScript in our site.”

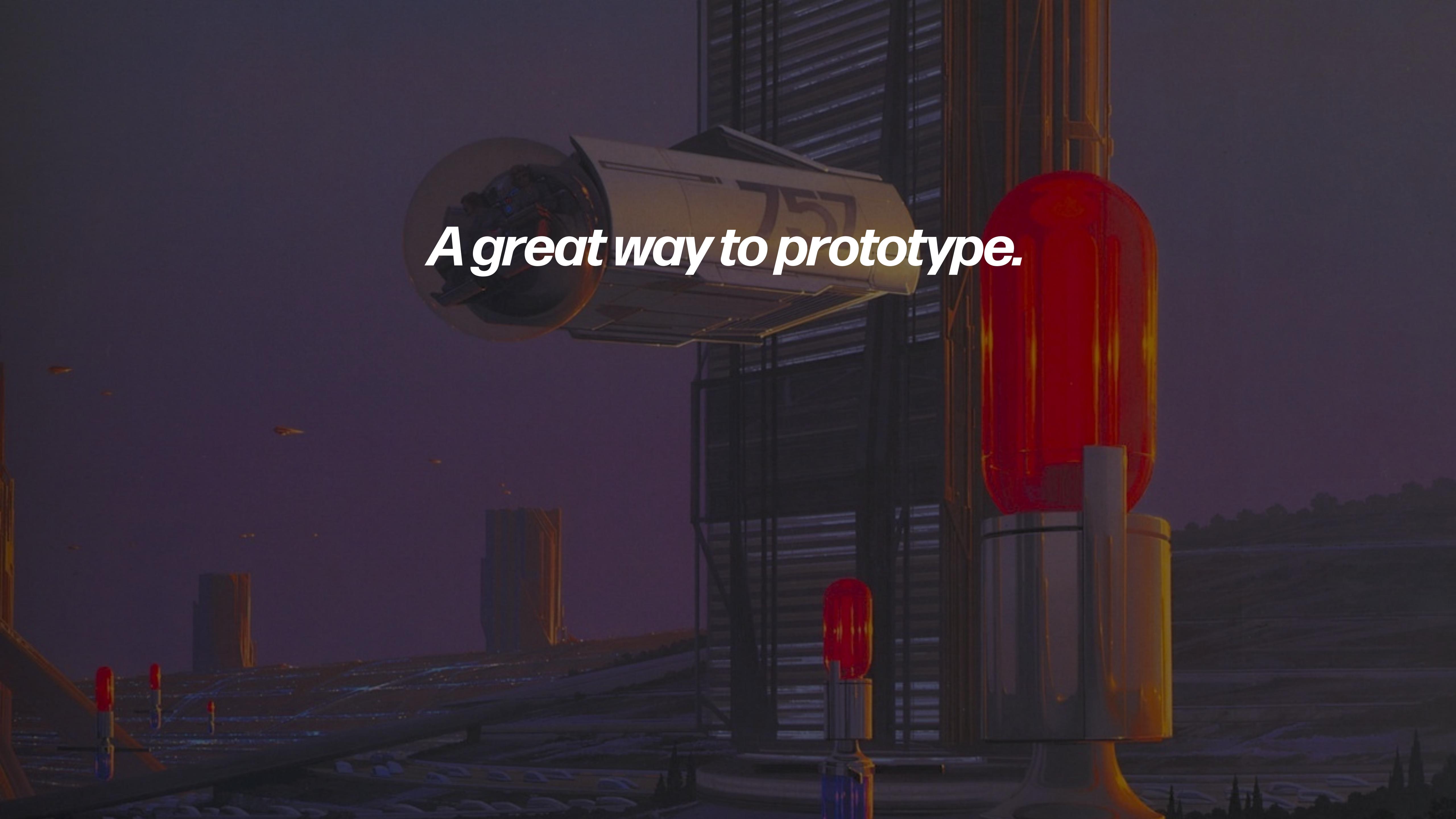
What to do: Design with web standards, as this book advises.

Develop your layout in a competent browser such as Firefox 3+, IE8+, Opera 10+, or Safari 4+, and then add the minimum number of workarounds necessary to make the design—or a close proximity of

the design—work in older versions of IE. (Such workarounds are described in Part II of the book.) By “close proximity of the design,” I mean make sure the

A large rocket stands on a launch pad at night, its side featuring a prominent NASA logo. The scene is illuminated by the rocket's own lights and the ambient light of the night sky. In the background, a satellite is visible in orbit.

“We are standardized on Bootstrap.”

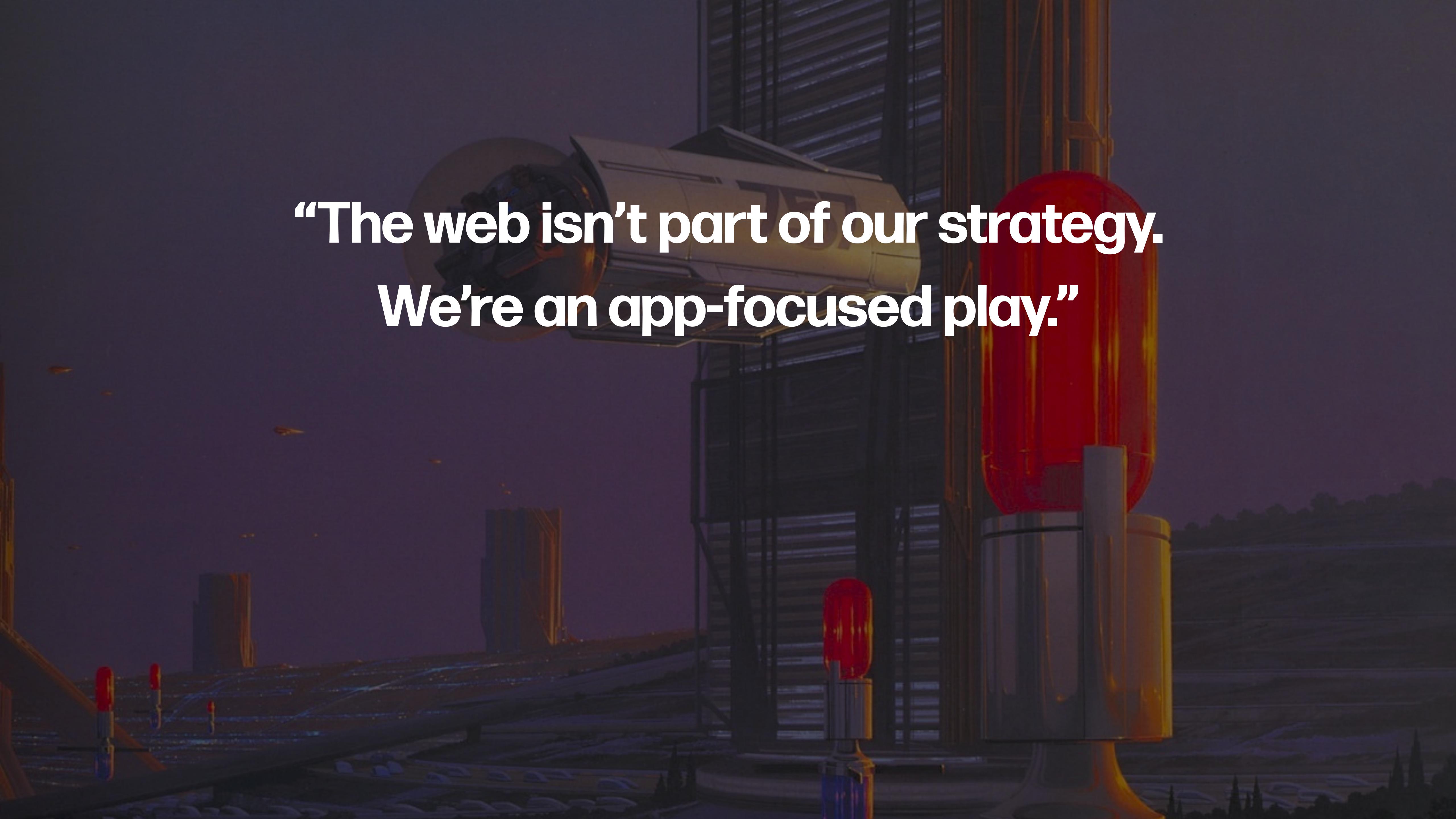


A great way to prototype.

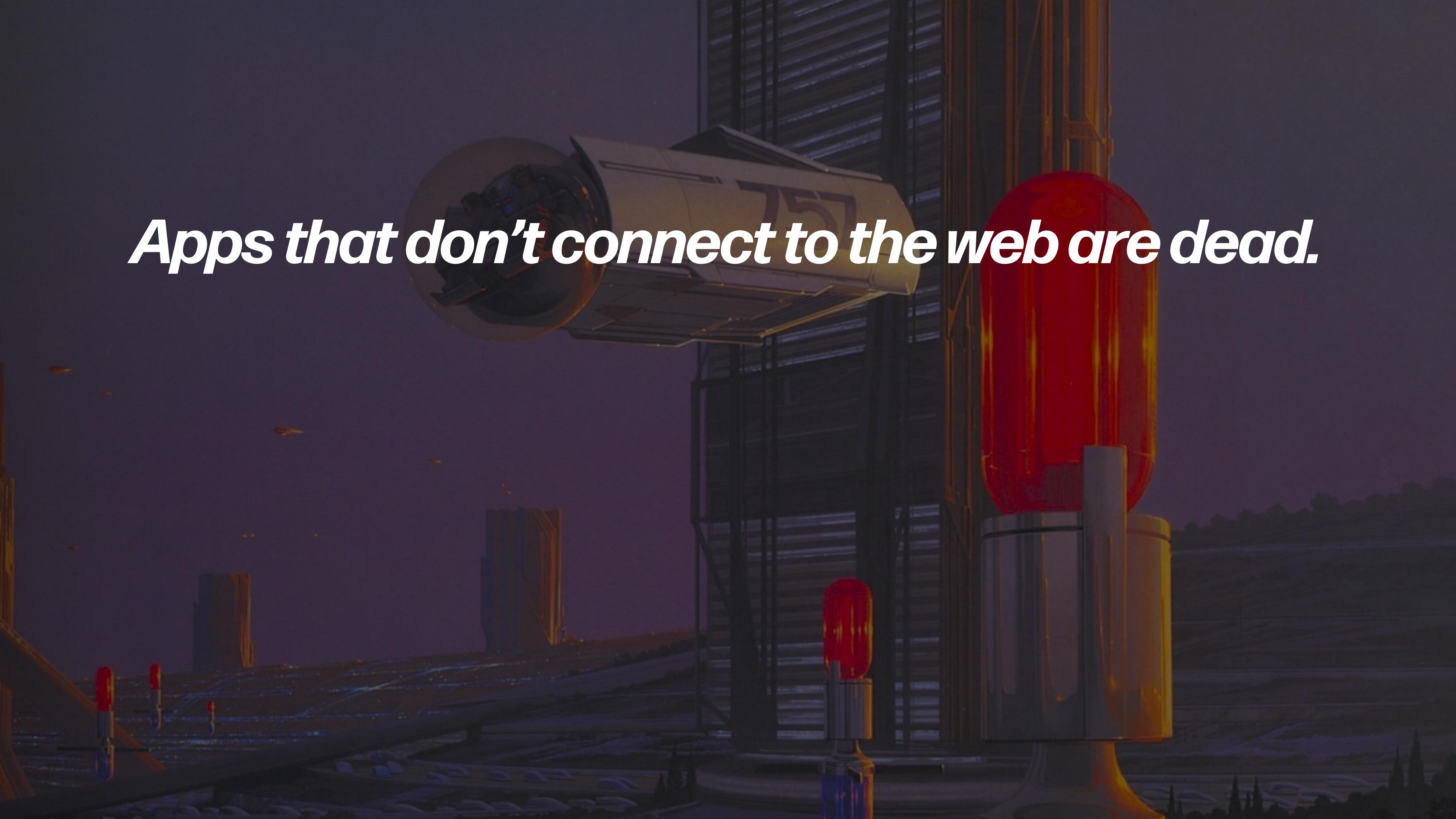
results. (If you can't explain it, photocopy a few pages out of this book—just don't tell my publisher.)

Objection: “Isn’t Flash (or other proprietary technology) better supported across platforms?”

What to say: “That’s a common misconception. Flash certainly has its place as a video delivery system or as a component in the experience layer, but you can’t beat (X)HTML, CSS, and JavaScript for maintainability and extensibility. In today’s browsers, design and scripting are as well supported across platforms with standards as they are in Flash. And you can’t beat web standards for SEO and accessibility.”



**“The web isn’t part of our strategy.
We’re an app-focused play.”**

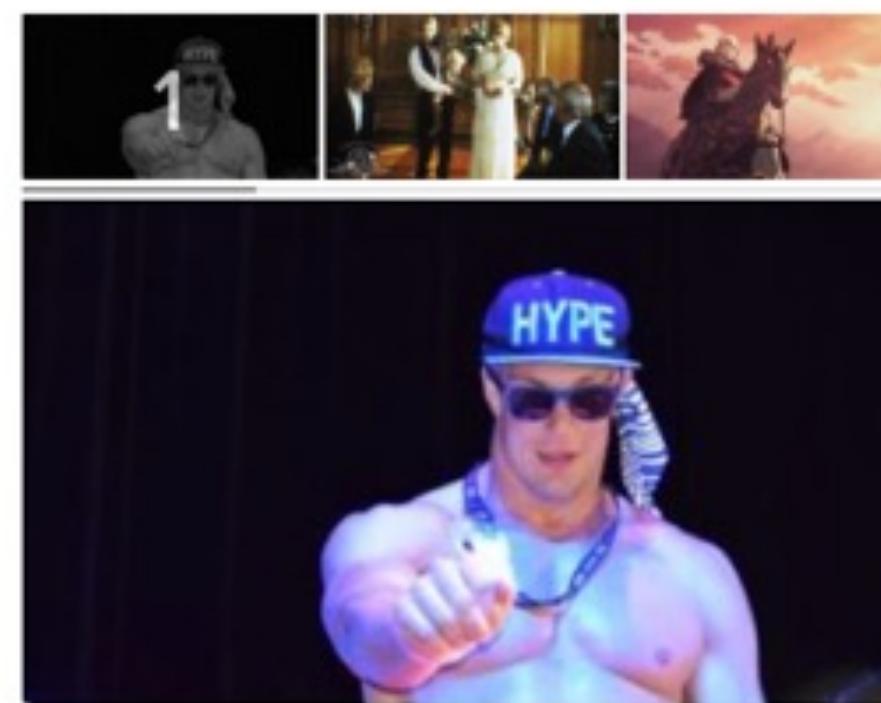
A dark, atmospheric night scene at an airport. In the foreground, a runway is visible with several red and blue light poles. In the background, there are several large, modern airport buildings with illuminated windows. The sky is dark, suggesting it's nighttime.

Apps that don't connect to the web are dead.

A night photograph of a modern building, possibly a hotel or airport, featuring a distinctive architectural design with large, illuminated spherical structures on its roofline. The sky is dark, and the building's lights are reflected on the ground.

Your brand happens between devices.

You may also like



Deadspin - Barry Petchesky

Imagine Being One Of The 1,600 People Who Didn't Realize They Were On The Gronk Cruise

Today 9:38am

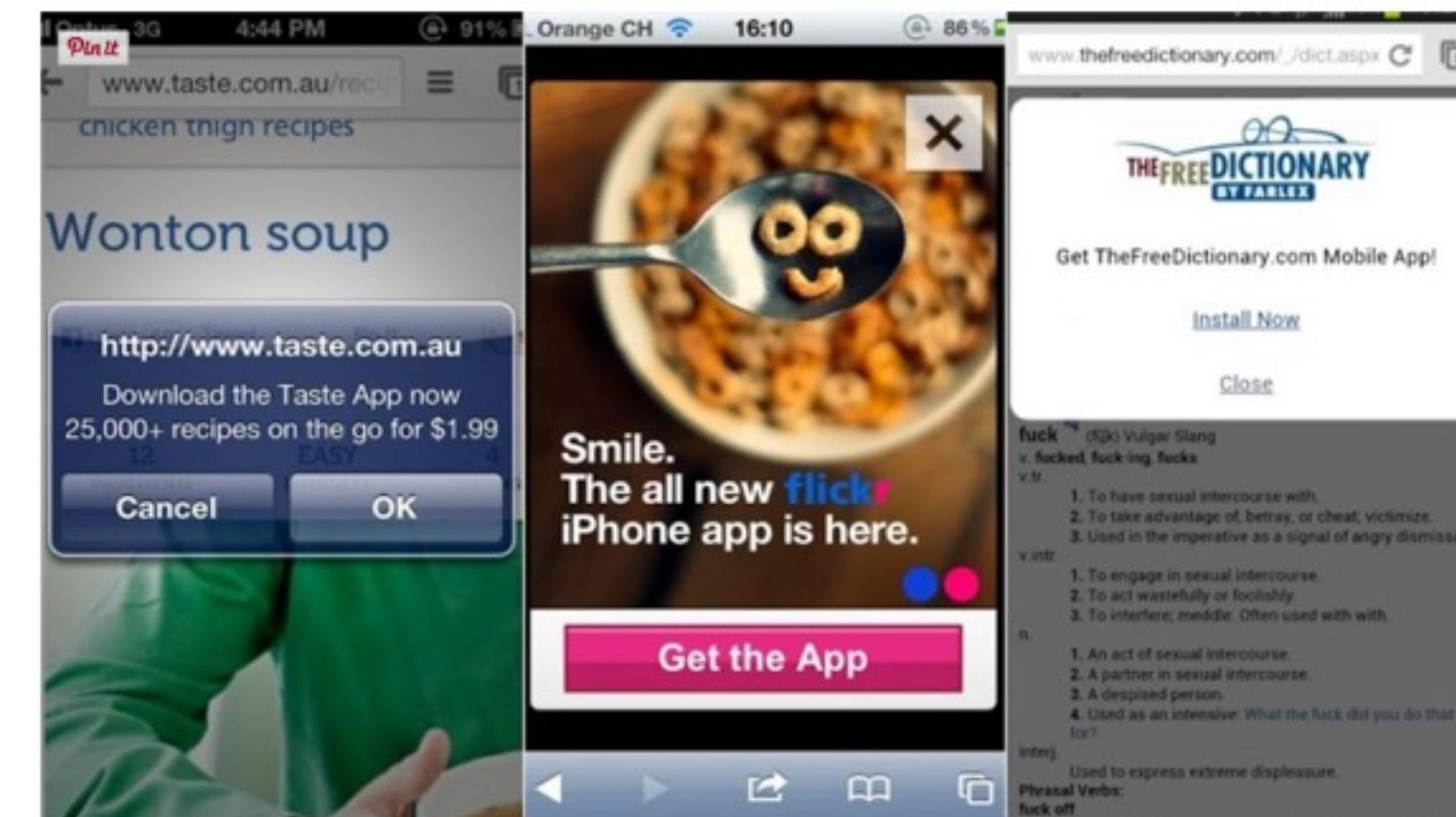
No One Wants to Download Your App When They Go to Your Website



Casey Chan

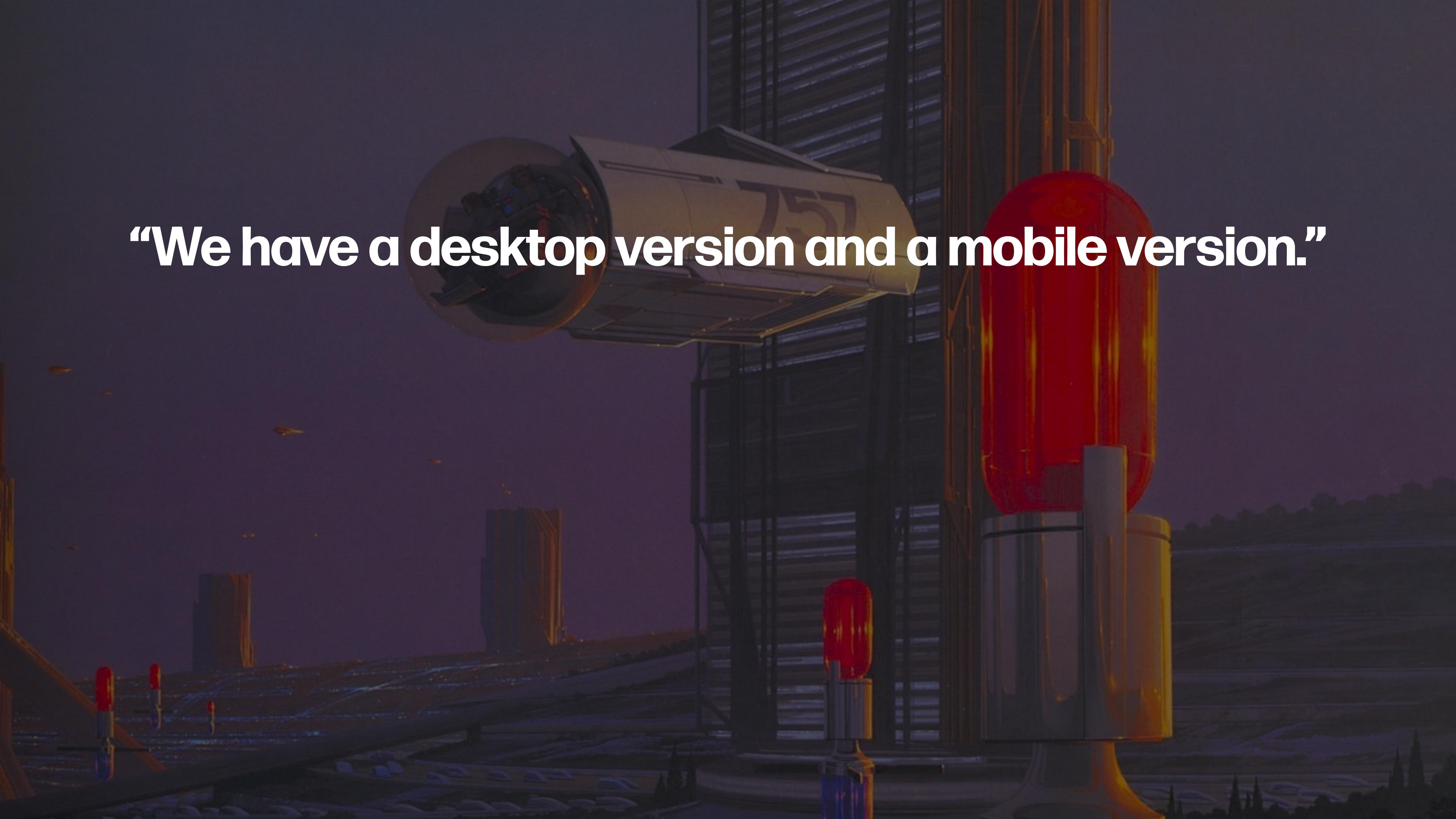
5/14/13 10:00pm · Filed to: HUMOR

13.3K 67 4



There's a tedious balance between using apps on your smartphone or just using the browser. Sometimes like in the case of Google Maps or The Weather Channel, apps are better. Other times when you're just dropping quickly into an online dictionary or a silly link, you just want to use the browser. The problem with that though is those stupid websites stupidly bombard you with stupid notifications to download their stupid app!

It's a pain in the ass when you have to come to a full stop because of an alert trying to get you to download their app instead of use their perfectly fine mobile website. Terence Eden feels your pain. He created the hilarious [I Don't Want Your Fucking App](#) tumblr to keep track of all the horrors of invasive app advertising. He explains:

A night photograph of a modern building with large illuminated circular windows and a red cylindrical structure.

“We have a desktop version and a mobile version.”

A wide-angle photograph of a large industrial or aerospace facility at night. In the foreground, a tall, cylindrical red structure, possibly a lightning rod or a large antenna, stands prominently. To its left, a large white satellite dish is mounted on a building. The building has several vertical panels and a flat roof. In the background, there are more industrial buildings and some smaller structures under a dark, hazy sky.

Does it scale? Are your assumptions correct?

The Future of Web Standards

Like the chapter we've just read, the first part of this chapter is for bosses, clients, the head of IT, the new director of marketing, and anyone else who doesn't yet see the connection between web standards and their organization's long-term health. For this is where we connect the dots between web standards, "Web 2.0," Search Engine Optimization (SEO), and other industry-leading ideas that can make even a businessperson who is indifferent to technology sit up and roll over.

Web 2.0-driven preference for *applications* over *documents*, and what happened next was inevitable.

HTML 5: Birth of the Cool

In 2005, under the leadership of Ian Hickson, engineers from the Mozilla Foundation and Opera Software formed the Web Hypertext Application Technology (WHAT) Working Group (www.whatwg.org), “a loose, unofficial, and open collaboration of Web browser manufacturers and interested parties” whose goal is “to address the need for one coherent development environment for Web applications, through the creation of technical specifications that are intended to be implemented in mass-market Web browsers.”

Although its parent organizations, including the Mozilla Foundation and the Opera Software company, are among the W3C’s greatest contributors, the engineers who formed WHAT were frustrated by the sometimes slow pace of W3C standards development. The group’s emphasis on practical, browser-related issues, and its preference for HTML over XML, initially set it apart from the W3C. But WHAT chose to work with the W3C, not against it, quickly submitting the first draft of its proposed HTML 5 language to the W3C for approval.

chapter five

Modern Markup

Part I outlined the creative and business problems engendered by old-school web design methods, sketched the benefits of designing with standards, and painted a cheery picture of standards-powered advances in the medium. The rest of this book will move from the general to the particular. The best way to start is by taking

A wide-angle photograph of a futuristic city skyline at sunset. The sky is filled with warm orange and yellow hues. In the foreground, a massive, multi-lane bridge or highway cuts through the scene, its lights reflecting on the water below. The city itself is built on a hillside, with numerous skyscrapers and buildings of varying heights. A person in a full-body spacesuit stands on the right side of the frame, looking out over the city. The overall atmosphere is one of a advanced, technologically advanced civilization.

Never underestimate
the power of HTML.

A wide-angle photograph of a futuristic city at sunset or sunrise. The sky is filled with warm orange and yellow hues. In the foreground, a massive, multi-layered bridge or highway structure stretches across the frame, its surfaces reflecting the light. Below the bridge, a river or body of water is visible. In the background, a range of mountains is partially obscured by mist or low-hanging clouds. On the right side of the image, a person wearing a full-body spacesuit and helmet is seen from behind, looking out over the city. The spacesuit has various panels and equipment attached. The overall atmosphere is one of a advanced, perhaps dystopian, urban environment.

Semantic markup, separating
content from presentation

A wide-angle photograph of a futuristic city at sunset or sunrise. The sky is filled with warm orange and yellow hues. In the foreground, there's a large, multi-lane highway or bridge structure that curves through the frame. To the right, a person wearing a full-body spacesuit and helmet is standing on a platform, looking out over the city. The city itself is built on a hillside, with numerous skyscrapers and buildings that have a metallic, reflective appearance. The overall atmosphere is one of a high-tech, advanced civilization.

*Now easier than ever
with CSS Flexbox!*



Putting My Patterns through Their Paces

Ethan Marcotte

<https://24ways.org/2015/putting-my-patterns-through-their-paces/>

10 December 2015

Published in [Code](#)

7 comments

Over the last few years, the conversation around [responsive design](#) has shifted subtly, focusing not on designing *pages*, but on *patterns*: understanding the small, reusable elements that comprise a larger design system. And given that many of those patterns are themselves responsive, learning to manage these small layout systems has become a big part of my work.

The thing is, the more pattern-driven work I do, the more I realize my design process has changed in a number of subtle, important ways. I suppose you might even say that pattern-driven design has, in a few ways, redesigned me.

MEET THE TEASER

Here's a recent example. A few months ago, some friends and I redesigned [The Toast](#). (It was a [really, really fun](#) project, and we [learned a lot](#).) Each page of the site is, as you might guess, stitched together from a host of tiny, reusable patterns. Some of them, like the search form and footer, are fairly unique, and used once per page; others are used more liberally, and built for reuse. The most prevalent example of these more generic patterns is the [teaser](#), which is classed as, uh, `.teaser`. (Look, I never said I was especially clever.)

In its simplest form, a teaser contains a headline, which links to an article:

Archives

Topics

What Wouldn't You Do With Significantly More Money Than You Have Now?

Tell me, what about you do you think would remain unchanged with endless funds?



By Mallory Ortberg 3 hours ago in BEAUTY

65

Places I Would Like To Live But Can't

The pocket of my own shirt, where I would be small and safe and warm and protected from wind and dust
 The inside of a big blue chair, like the meat insides, where all the stuffing and insulation lives
 Exactly on the place of any furniture where a small dog was just asleep on it so it's still warm and he's coming back any minute



MORE BY MALLORY ORTBERG



[Plausible Lines For Absolutely Any Episode Of Columbo](#)



[Bible Verses Where "Sinners" Has Been Replaced With "Posers"](#)



[Paintings Of The Nine Muses Dejectedly Performing Their Duties](#)



["And If You Threw A Party": The Full Golden Girls Theme Song Gospel Edition Is Finished](#)



3

[What Wouldn't You Do With Significantly More Money Than You Have Now?](#)



4

Misty Copeland Monday



5

[Link Roundup!](#)

6

John Keats' *La Belle Dame Sans Merci*

By Businesslady 4 hours ago in DEAR BUSINESSLADY

17

Dear Businesslady: Advice on Harassment and Prying

I've been at my job for two and a half years but I'm about to leave to move out of



1

2

MOST POPULAR

```
<div class="teaser">
  <p class="article-byline">By <a href="#">Author
Name</a></p>
  <a class="comment-count" href="#">126
<i>comments</i></a>
  <h1 class="article-title"><a href="#">Article Title</a></
h1>
  <p class="teaser-excerpt">Lorem ipsum dolor sit amet,
consectetur...</p>
</div>
```


The background of the slide is a dark, blue-toned aerial photograph of a complex highway interchange at night. The roads are illuminated by streetlights, creating a grid-like pattern of light against the dark sky. The interchange features multiple levels and ramps, forming a dense web of lines.

**Use Flexbox to further separate
structure from presentation**



**Since the basic design works
well, you're covered in any device**



Essence of Designing With Web Standards

* 5 January 2016 3pm America/New_York

OF PATTERNS AND POWER: WEB STANDARDS THEN & NOW



IN “CONTENT Display Patterns” (which all front-end folk should

Modular/**atomic design** doesn't change this truth, it just reinforces its wisdom. **Flexbox** and **grid layout** don't change this truth, they just make it easier to do it better. **HTML5** doesn't change this truth, it just reminds us that the separation of structure from style came into existence for a reason. A reason that hasn't changed. A reason that cannot change, because it is the core truth of the web, and is inextricably bound up with the promise of this medium.

Separating structure from style and behavior was the web standards movement's prime revelation, and each generation of web designers discovers it anew. This separation is what makes our content as backward-compatible as it is forward-compatible (or "**future-friendly**," if you prefer). It's the key to re-use. The key to accessibility. The key to the new kinds of CMS systems we're just beginning to dream up. It's what makes our content as accessible to an ancient device as it will be to an unimagined future one.

Every time a leader in our field discovers, as if for the first time, the genius of this separation between style, presentation, and behavior, she is validating the brilliance of web forbears like **Tim Berners-Lee**, **Håkon Wium Lie**, and **Bert Bos**.

Every time a Dan or an Ethan (or a **Sara** or a **Lea**) writes a beautiful

The Secret Shame of Rotten Markup

During our industry's first decade, designing for the web was like feeding a roomful of finicky toddlers. To build sites that worked, we dutifully learned to accommodate each browser's unique dietary requirements. Today's browsers all eat the same nutritious stuff, and have done since 2001; but many professionals *still* haven't grasped this and continue crumbling M&Ms into the soufflé.

Use Elements According to Their Meaning, Not Because of the Way They “Look”

Some of us have gotten into the habit of marking text as an `h1` when we merely want it to be large or as `li` when we just want to stick a bullet in front of it. As discussed in Part I, browsers have traditionally imposed design attributes

chapter seven

HTML5: The New Hope

HTML5 represents the first significant change to web markup since... well, forever. For one thing, it is the first web markup

HTML5 Elements on Parade

Beyond relaxing the rules that XHTML tightened, the big news in HTML5 is of course the creation of new elements. Some of these are long overdue application-focused shiny bits such as `video` and `audio`. As indicated earlier in the chapter, these elements allow video and audio to play in browsers without plugins, without previously invalid `embed` elements, and without nested `object` elements intermingled with scripts and fallbacks. In “A Preview of HTML5” (www.alistapart.com/articles/previewofhtml5), Lachlan Hunt provides these examples:

```
<video src="video.ogv" controls poster="poster.jpg"
width="320" height="240">
    <a href="video.ogv">Download movie</a>
</video>

<audio src="music.oga" controls>
    <a href="music.oga">Download song</a>
</audio>
```

Note the attributes `controls`, which tells the browser to create playback controls, and `poster`, which tells the browser what image to use as a placeholder until the user decides to play the video. Also note the use of simple fallback links for those whose browsers don’t support `video` or `audio`.

Sophisticated web page markup doesn’t get much simpler than that, nor will

aside, and especially with header and footer.

- **Section:** Like a div, a section contains and organizes related content, including headings, paragraphs, images, and so on. It normally has a header and may also have a footer. Section does not replace div, which remains a block-level organizer. A div contains items that may or may not relate to one another. A section, in contrast, may *only* contain thematically related items, such as an h1 headline advertising a house for sale, a photograph of the house, several paragraphs of text describing the house, a list containing the realtor's contact information, and so on. The enclosing section conveys the semantic information that the content of these items is related. In the hands of a standards-based designer, this relationship was implicit with the items contained in a div in XHTML. But HTML5 makes the connection explicit, and removes such a connection if a div is used. In HTML5, a div becomes simply a junk drawer (or "flow content" container in WHATWG parlance), and web designers are strongly encouraged to use section instead.
- **Article:** Self-contained content such as a blog post, news article, or reader comment. Like section, it may contain a header and footer. Although the specification doesn't explicitly call it this, an article may be viewed as a specialized form of a section. A blog post or news article would be marked up as an article, not as a section. Neither section nor article is "higher" in the stack. An article may contain sections; a section may contain articles. Like anything new, section and article are not yet widely supported by major browsers.

aside, and especially with header and footer.

- **Section:** Like a `div`, a `section` contains and organizes related content, including headings, paragraphs, images, and so on. It normally has a header and may also have a footer. `Section` does not replace `div`, which remains a block-level organizer. A `div` contains items that may or may not relate to one another. A `section`, in contrast, may *only* contain thematically related items, such as an `h1` headline advertising a house for sale, a photograph of the house, several paragraphs of text describing the house, a list containing the realtor's contact information, and so on. The enclosing `section` conveys the semantic information that the content of

Custom made for “atomic” design

tion if a `div` is used. In HTML5, a `div` becomes simply a junk drawer (or “flow content” container in WHATWG parlance), and web designers are strongly encouraged to use `section` instead.

- **Article:** Self-contained content such as a blog post, news article, or reader comment. Like `section`, it may contain a header and footer. Although the specification doesn't explicitly call it this, an `article` may be viewed as a specialized form of a `section`. A blog post or news article would be marked up as an `article`, not as a `section`. Neither `section` nor `article` is “higher” in the stack. An `article` may contain `sections`; a `section` may contain `articles`. Like anything new, `section` and `arti-`

friends with the unique identifier (`id`) attribute, which allows you to write ultra-compact, semantic markup; serve the correct layout to a specific page type without back-end scripting; and show users, without JavaScript, where they are in the site's hierarchy. And we'll add semantic richness to our sites via the `class` attribute, while avoiding the gross misuse of that attribute that wreaks semantic havoc on far too many modern sites.

div, id, and Other Assistants

This chapter and those that follow make much reference to the `div` element and the `id` attribute. Used correctly, `div` is the Hamburger Helper of structured markup, while `id` is an amazing little tool that permits you to write highly compact (X)HTML, apply CSS wisely, and add sophisticated behavior to your site via standard JavaScript. The W3C, in “The Global Structure of an HTML Document,” defines `div` and two other important HTML/XHTML components thusly:

The DIV and SPAN elements, in conjunction with the `id` and `class` attributes, offer a generic mechanism for adding structure to documents. These elements define content to be inline (SPAN) or block-level (DIV) but impose no other presentational idioms on the content. Thus, authors may use these elements in conjunction with style sheets...

friends with the unique identifier (`id`) attribute, which allows you to write ultra-compact, semantic markup; serve the correct layout to a specific page type without back-end scripting; and show users, without JavaScript, where they are in the site's hierarchy. And we'll add semantic richness to our sites via the `class` attribute, while avoiding the gross misuse of that attribute that wreaks semantic havoc on far too many modern sites.

The curse of divitis

`div`, `id`, and Other Assistants

This chapter and those that follow make much reference to the `div` element and the `id` attribute. Used correctly, `div` is the Hamburger Helper of structured markup, while `id` is an amazing little tool that permits you to write highly compact (X)HTML, apply CSS wisely, and add sophisticated behavior to your site via standard JavaScript. The W3C, in “The Global Structure of an HTML Document,” defines `div` and two other important HTML/XHTML components thusly:

The DIV and SPAN elements, in conjunction with the `id` and `class` attributes, offer a generic mechanism for adding structure to documents. These elements define content to be inline (SPAN) or block-level (DIV) but impose no other presentational idioms on the content. Thus, authors may use these elements in conjunction with style sheets...

friends with the unique identifier (`id`) attribute, which allows you to write ultra-compact, semantic markup; serve the correct layout to a specific page type without back-end scripting; and show users, without JavaScript, where they are in the site's hierarchy. And we'll add semantic richness to our sites via the `class` attribute, while avoiding the gross misuse of that attribute that wreaks semantic havoc on far too many modern sites.

Fear of “`id`”

div, id, and Other Assistants

This chapter and those that follow make much reference to the `div` element and the `id` attribute. Used correctly, `div` is the Hamburger Helper of structured markup, while `id` is an amazing little tool that permits you to write highly compact (X)HTML, apply CSS wisely, and add sophisticated behavior to your site via standard JavaScript. The W3C, in “The Global Structure of an HTML Document,” defines `div` and two other important HTML/XHTML components thusly:

The DIV and SPAN elements, in conjunction with the `id` and `class` attributes, offer a generic mechanism for adding structure to documents. These elements define content to be inline (SPAN) or block-level (DIV) but impose no other presentational idioms on the content. Thus, authors may use these elements in conjunction with style sheets...

friends with the unique identifier (`id`) attribute, which allows you to write ultra-compact, semantic markup; serve the correct layout to a specific page type without back-end scripting; and show users, without JavaScript, where they are in the site's hierarchy. And we'll add semantic richness to our sites via the `class` attribute, while avoiding the gross misuse of that attribute that we look at next.

Overuse of “`class`” in early standards-compliant sites kept HTML mired in layout

This chapter and those that follow make much reference to the `div` element and the `id` attribute. Used correctly, `div` is the Hamburger Helper of structured markup, while `id` is an amazing little tool that permits you to write highly compact (X)HTML, apply CSS wisely, and add sophisticated behavior to your site via standard JavaScript. The W3C, in “The Global Structure of an HTML Document,” defines `div` and two other important HTML/XHTML components thusly:

The DIV and SPAN elements, in conjunction with the `id` and `class` attributes, offer a generic mechanism for adding structure to documents. These elements define content to be inline (SPAN) or block-level (DIV) but impose no other presentational idioms on the content. Thus, authors may use these elements in conjunction with style sheets...

friends with the unique identifier (`id`) attribute, which allows you to write ultra-compact, semantic markup; serve the correct layout to a specific page type without back-end scripting; and show users, without JavaScript, where they are in the site's hierarchy. And we'll add semantic richness to our sites via the `class` attribute, while avoiding the gross misuse of that attribute that wreaks semantic havoc on far too many modern sites.

`id` and contextual selectors were the solution, but have a bad rep now because

`div`, `id`, and Other Assistants

of Object Oriented CSS (OOCSS)

This block demand that all we have in here refers to the `div` element and the `id` attribute. Used correctly, `div` is the hamburger helper of structured markup, while `id` is an amazing little tool that permits you to write highly compact (X)HTML, apply CSS wisely, and add sophisticated behavior to your site via standard JavaScript. The W3C, in “The Global Structure of an HTML Document,” defines `div` and two other important HTML/XHTML components thusly:

The DIV and SPAN elements, in conjunction with the `id` and `class` attributes, offer a generic mechanism for adding structure to documents. These elements define content to be inline (SPAN) or block-level (DIV) but impose no other presentational idioms on the content. Thus, authors may use these elements in conjunction with style sheets...

friends with the unique identifier (`id`) attribute, which allows you to write ultra-compact, semantic markup; serve the correct layout to a specific page type without back-end scripting; and show users, without JavaScript, where they are in the site's hierarchy. And we'll add semantic richness to our sites via the `class` attribute, while avoiding the gross misuse of that attribute that

1. Separate structure and skin.

2. Separate container and content.

This chapter and those that follow make much reference to the `div` element and the `id` attribute. Used correctly, `div` is the Hamburger Helper of structured markup, while `id` is an amazing little tool that permits you to write highly compact (X)HTML, apply CSS wisely, and add sophisticated behavior to your site via standard JavaScript. The W3C, in “The Global Structure of an HTML Document,” defines `div` and two other important HTML/XHTML components thusly:

The DIV and SPAN elements, in conjunction with the `id` and `class` attributes, offer a generic mechanism for adding structure to documents. These elements define content to be inline (SPAN) or block-level (DIV) but impose no other presentational idioms on the content. Thus, authors may use these elements in conjunction with style sheets...

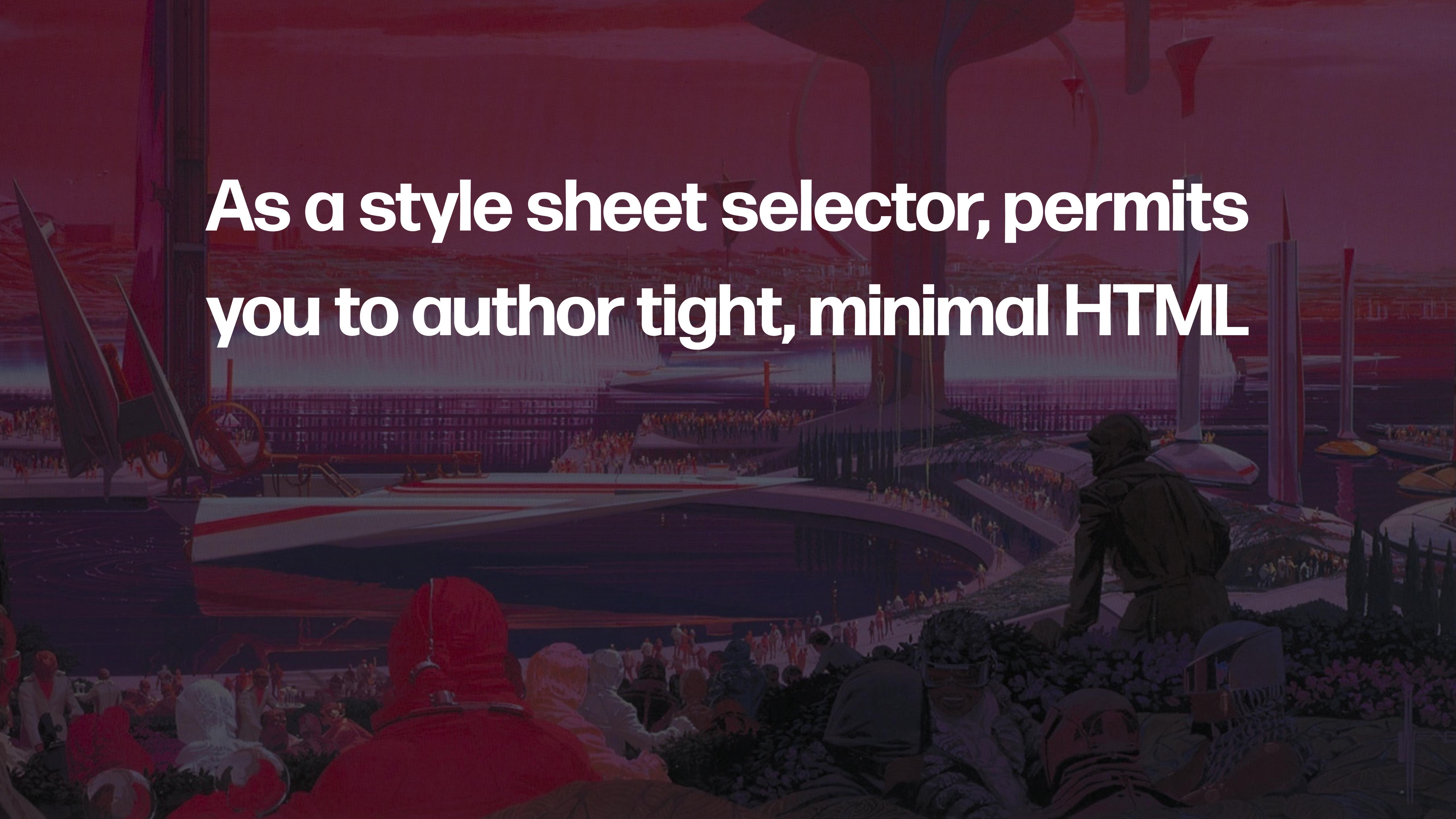
friends with the unique identifier (`id`) attribute, which allows you to write ultra-compact, semantic markup; serve the correct layout to a specific page type without back-end scripting; and show users, without JavaScript, where they are in the site's hierarchy. And we'll add semantic richness to our sites via the `class` attribute, while avoiding the gross misuse of that attribute that wreaks semantic havoc on far too many modern sites.

“ids are bad” :(

div, id, and Other Assistants

This chapter and those that follow make much reference to the `div` element and the `id` attribute. Used correctly, `div` is the Hamburger Helper of structured markup, while `id` is an amazing little tool that permits you to write highly compact (X)HTML, apply CSS wisely, and add sophisticated behavior to your site via standard JavaScript. The W3C, in “The Global Structure of an HTML Document,” defines `div` and two other important HTML/XHTML components thusly:

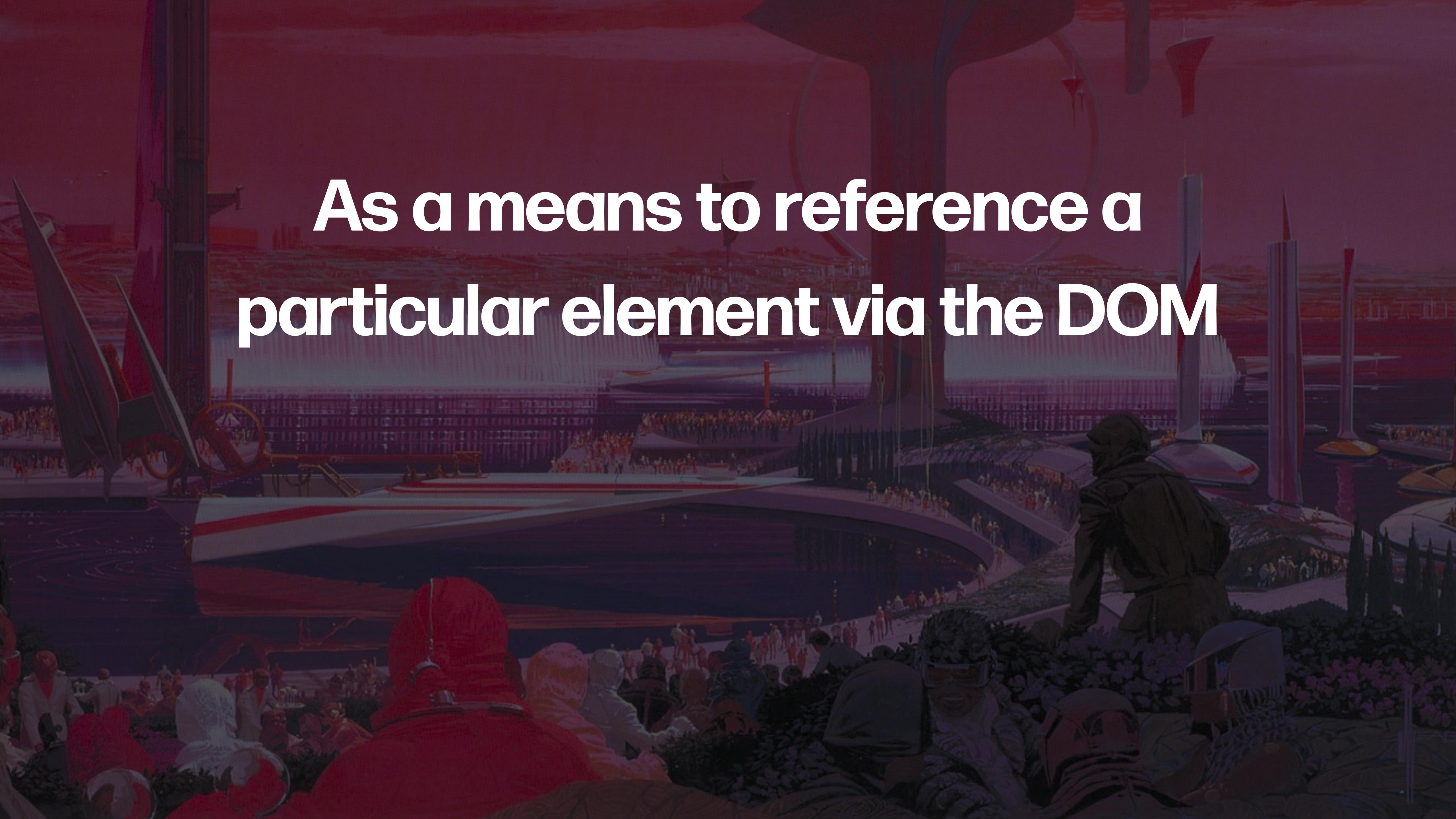
The DIV and SPAN elements, in conjunction with the `id` and `class` attributes, offer a generic mechanism for adding structure to documents. These elements define content to be inline (SPAN) or block-level (DIV) but impose no other presentational idioms on the content. Thus, authors may use these elements in conjunction with style sheets...



**As a style sheet selector, permits
you to author tight, minimal HTML**

A vibrant, futuristic cityscape at night. In the foreground, a large crowd of people in colorful clothing is gathered, some wearing headsets. In the background, a massive stadium with a curved roof is illuminated, and numerous flying cars and drones are visible against a dark sky.

**As a target for hypertext links,
works in all HTML-capable devices**



**As a means to reference a
particular element via the DOM**



YOU DECIDE

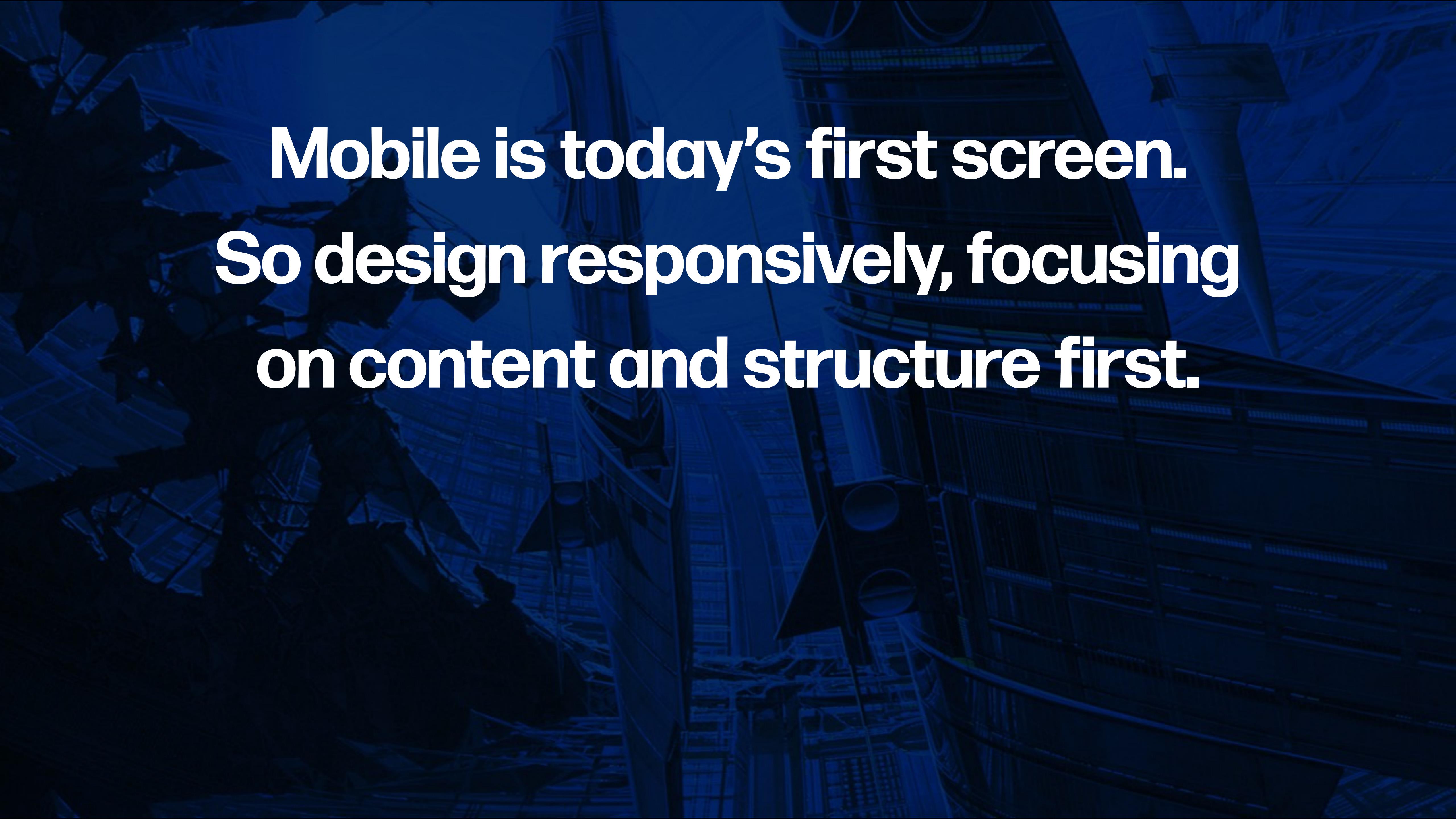


Design Responsively

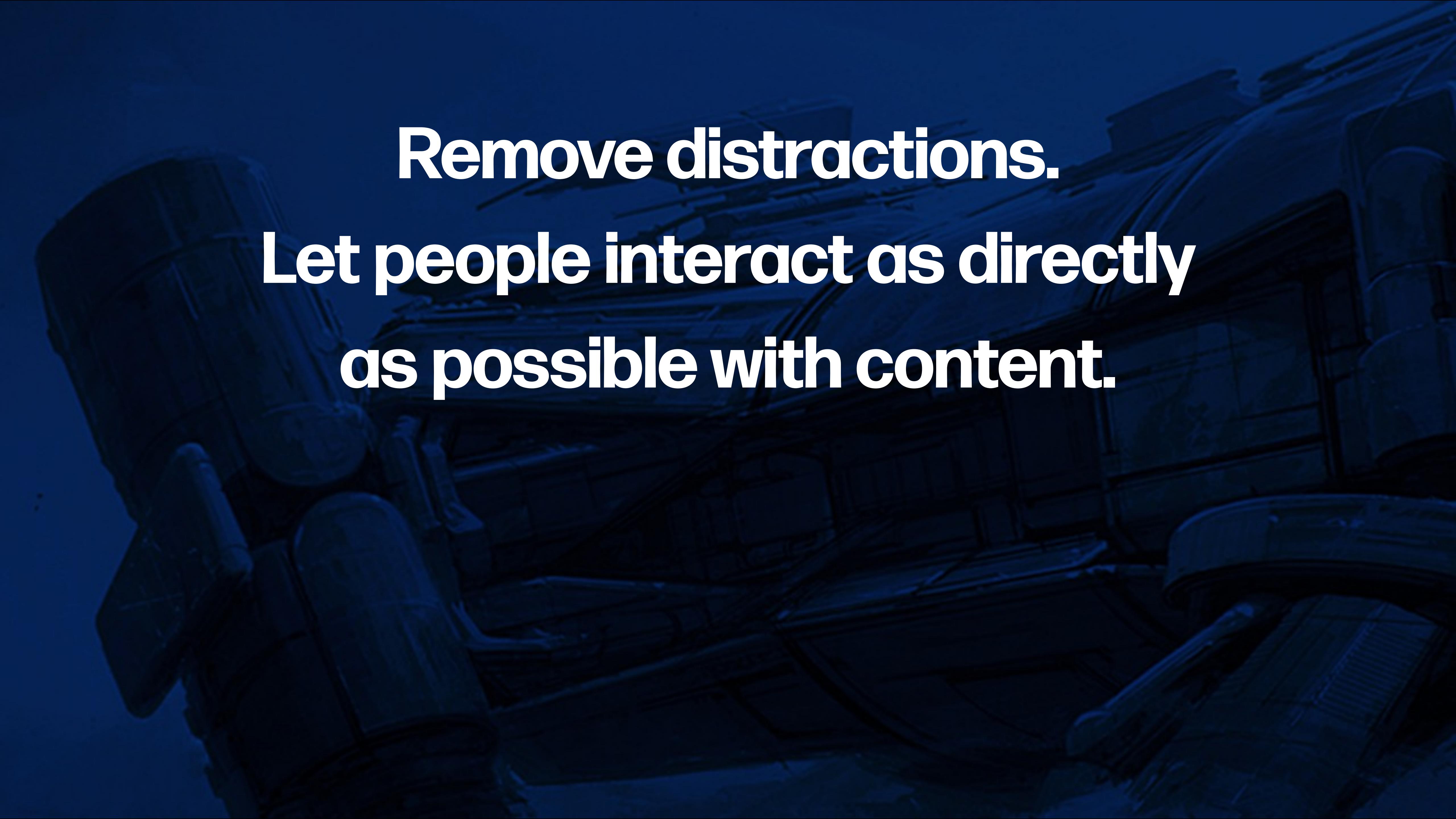
A dark, atmospheric scene showing a train car with solar panels and a person's legs.

Song remains the same.

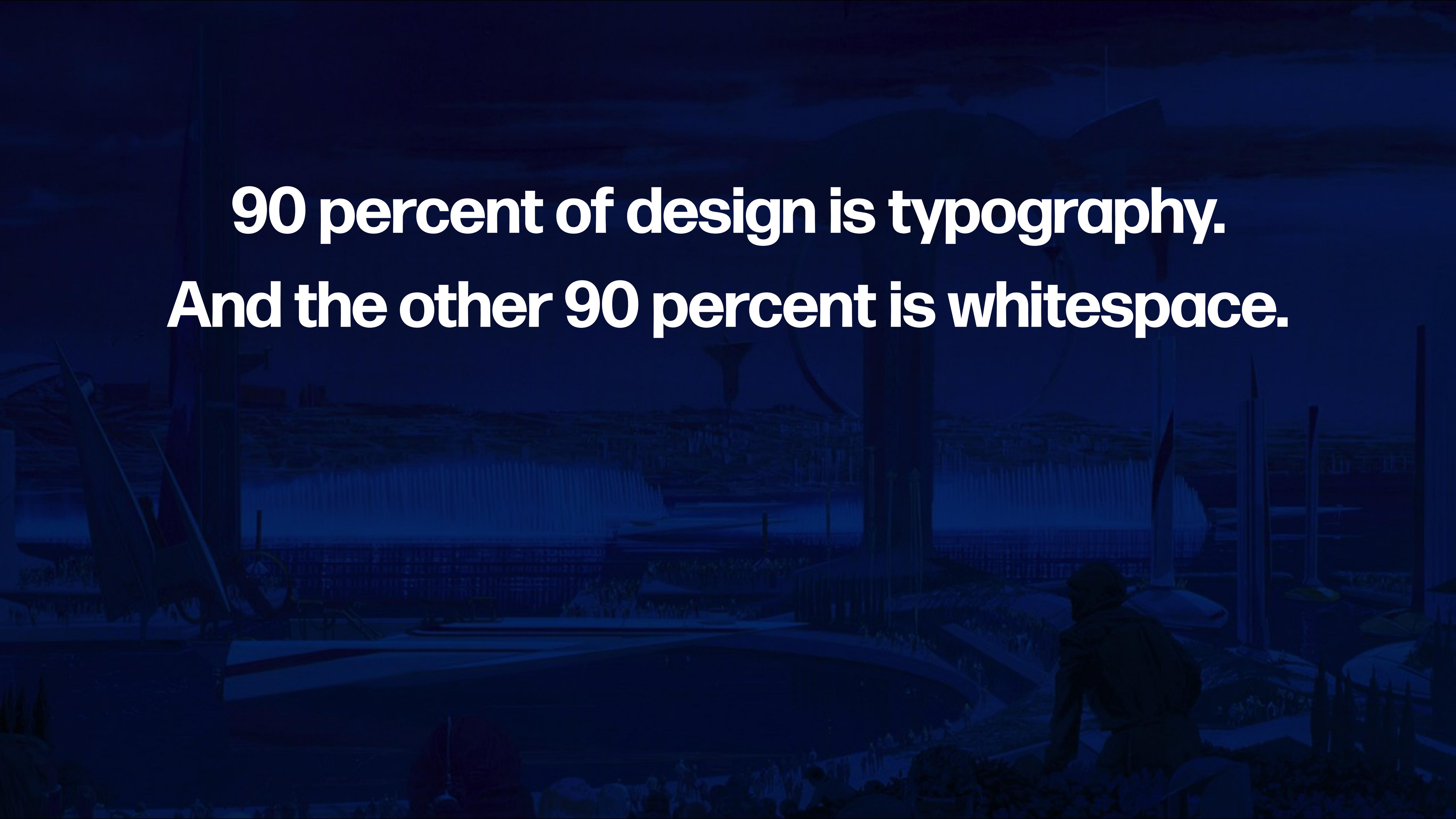
Rules For The Road



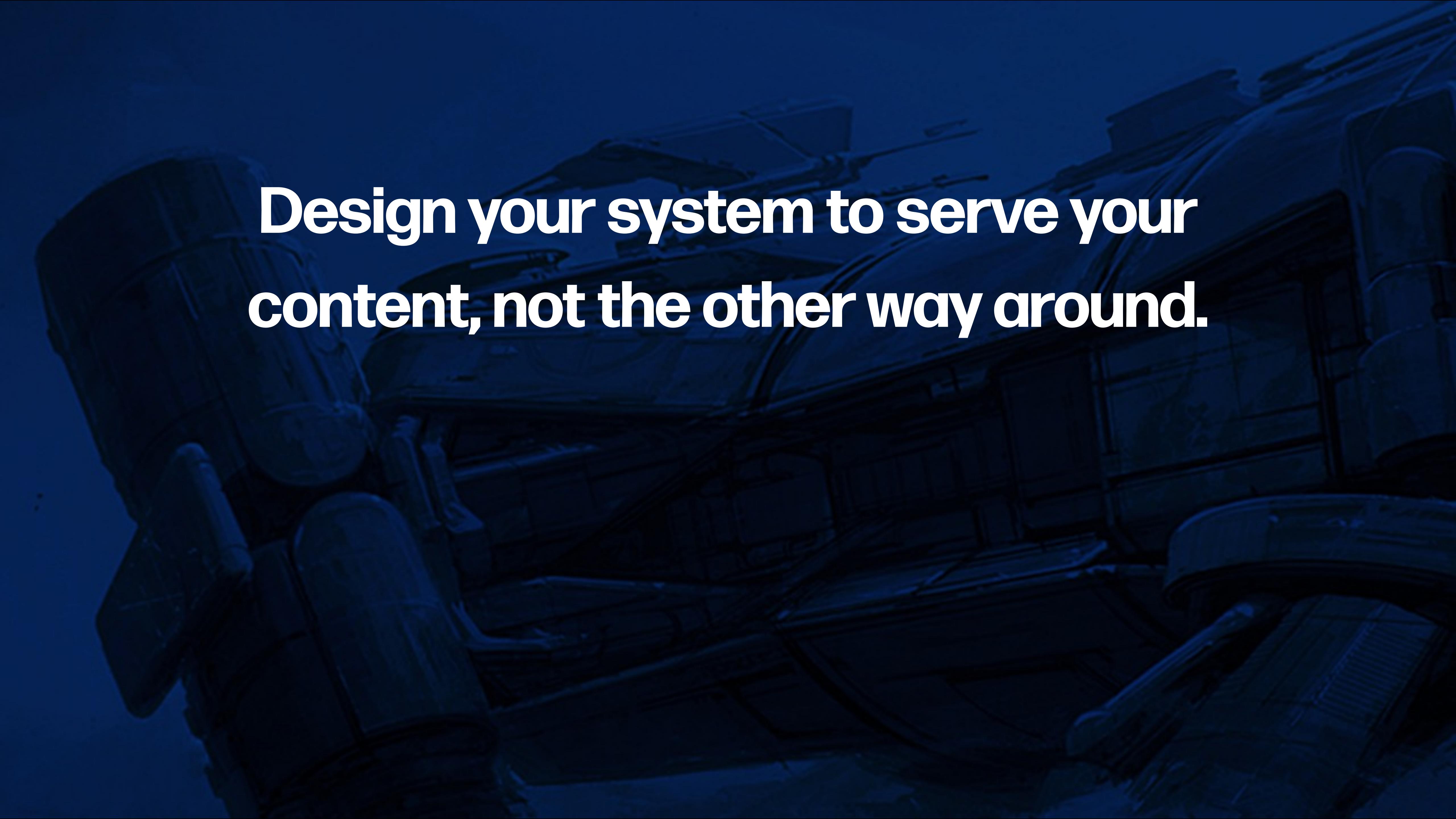
**Mobile is today's first screen.
So design responsively, focusing
on content and structure first.**



**Remove distractions.
Let people interact as directly
as possible with content.**



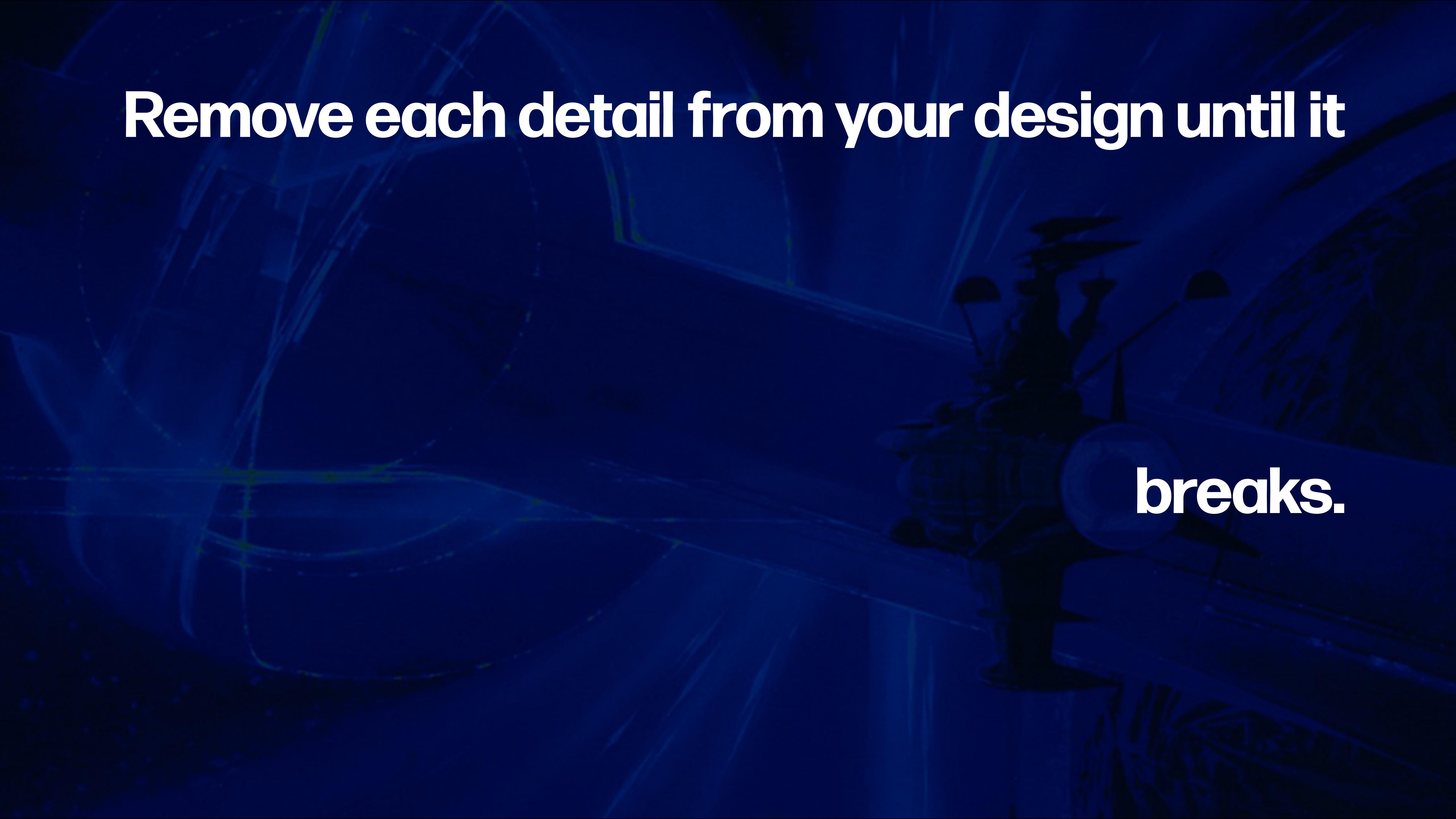
**90 percent of design is typography.
And the other 90 percent is whitespace.**

A dark, atmospheric photograph showing a close-up view of a stack of numerous old books. The books are bound in worn, textured covers, many of which are a faded yellow or brown color, suggesting significant age. The lighting is low, creating deep shadows and highlighting the ridges between the spines of the books.

**Design your system to serve your
content, not the other way around.**



Style is the servant of brand and content.
Style without purpose is noise.



Remove each detail from your design until it

breaks.

Nobody waits.
Speed is to today's design what
ornament was to yesterday's.



**Don't design to prove you're clever.
Design to make the user think she is.**

The background of the image is a detailed illustration of a futuristic city at sunset. The city is built on a steep hillside, with numerous skyscrapers and a complex network of elevated roads and highways. In the distance, a massive, multi-tiered space station or orbital platform hangs in the sky, its structure illuminated by the setting sun. The sky is a vibrant orange and yellow, transitioning into a darker blue at the top. The overall atmosphere is one of a advanced, sprawling urban environment.

Type family: Forma (Font Bureau)

Illustrations: Syd Mead