



UI Density

What UI density means and how to design for it

May 20, 2024 · 20 min read

Interfaces are becoming less dense.

I'm usually one to be skeptical of nostalgia and "we liked it that way" bias, but comparing websites and applications of 2024 to their 2000s-era counterparts, the spreading out of software is hard to ignore.

To explain this trend, and suggest how we might regain density, I started by asking what, exactly, UI density is. It's not just the way an interface looks at one moment in time; it's about the amount of information an interface can provide over a series of moments. It's about how those moments are connected through design decisions, and how those decisions are connected to the value the software provides.

I'd like to share what I found. Hopefully this exploration helps you define UI density in concrete and useable terms. If you're a designer, I'd like you to question the density of the interfaces you're creating; if you're not a designer, use the lens of UI density to understand the software you use.

Visual density

We think about density first with our eyes. At first glance, density is just how many things we see in a given space. This is **visual density**. A visually dense software interface puts a lot of stuff on the screen. A visually sparse interface puts less stuff on the screen.

Bloomberg's Terminal is perhaps the most common example of this kind of density. On just a single screen, you'll see scrolling sparklines of the major market indices, detailed trading volume breakdowns, tables with dozens of rows and columns, scrolling headlines containing the latest news from agencies around the world, along with UI signposts for all the above with keyboard shortcuts and quick actions to take.

BA US \$ ↑ 131.46 -1.91 N131.38 / 131.39N 20x223
 At 18:30 d Vol 30,703,029 0 127.115D H 131.50K L 124.35K Val 4.272B

BA US Equity 90 Actions 97 Export 98 Settings Financial Analysis

39) ADJ Boeing Co/The ASC 842 ? Periods 10 Annuals Cur FRC (USD)

Key Stats I/S B/S C/F Ratios Segments Add ESG Custom

By Measure By Geography By Segment

In Millions of USD except Per Share

	2016 Y		2017 Y		2018 Y		2019 Y	
12 Months Ending	12/31/2016		12/31/2017		12/31/2018		12/31/2019	
Revenue	93,496.0	100.0%	94,005.0	100.0%	101,127.0	100.0%	76,559.0	100.0%
Commercial Airplanes	59,378.0	63.4%	58,014.0	62.1%	57,499.0	56.8%	32,255.0	41.8%
Defense, Space, & Security	20,180.0	21.5%	20,561.0	22.0%	26,392.0	26.1%	26,227.0	34.0%
Global Services	13,819.0	14.8%	14,581.0	15.6%	17,056.0	16.9%	18,468.0	23.9%
Boeing Capital Corporation	298.0	0.3%	307.0	0.3%	274.0	0.3%	244.0	0.3%
Accounting differences/elimin...	-179.0				-94.0		-635.0	
Boeing Capital Corporation								
Commercial Airplanes								
Accounting differences/elimin...			542.0					
Other								
Boeing Defense, Space & Secu...								
Revenue by End User	93,496.0	100.0%	94,005.0	100.0%	101,127.0	100.0%	84,818.0	100.0%
Total Revenue	93,496.0	100.0%	94,005.0	100.0%	101,127.0	100.0%	84,818.0	100.0%
Operating Income	6,527.0	100.0%	10,344.0	100.0%	11,987.0	100.0%	-1,975.0	
Global Services	2,159.0	36.7%	2,246.0	22.4%	2,536.0	21.0%	2,697.0	-203.0...
Defense, Space, & Security	1,678.0	28.6%	2,193.0	21.9%	1,657.0	13.7%	2,608.0	-197.0...
Boeing Capital Corporation	59.0	1.0%	114.0	1.1%	79.0	0.7%	28.0	-2.1%
Unallocated Expenses	650.0				-115.0		-651.0	
Commercial Airplanes	1,981.0	33.7%	5,452.0	54.5%	7,830.0	64.7%	-6,657.0	502.8%
Boeing Capital Corporation								

DJ 18:47 *Management and Network Services, LLC Notifies Patients of Data Se
 BN 18:47 *QANTAS SECURED A\$550M IN FUNDING AGAINST 3 OF ITS BOEING 787-9
 902 NS1 18:47 Fox Business: Coronavirus-hit economies spark US, UK trade negotia
 BN 18:47 *DEXUS: REDUCTION IN SALARIES, FEES FOR DIRECTORS, CEO, EXECS

A screenshot of Terminal's interface. Via Objective Trade on YouTube

Craigslist is another visually dense example, with its hundreds of plain links to categories and spartan search-and-filter interface. McMaster-Carr's website shares similar design cues, listing out details for many product variations in a very small space.

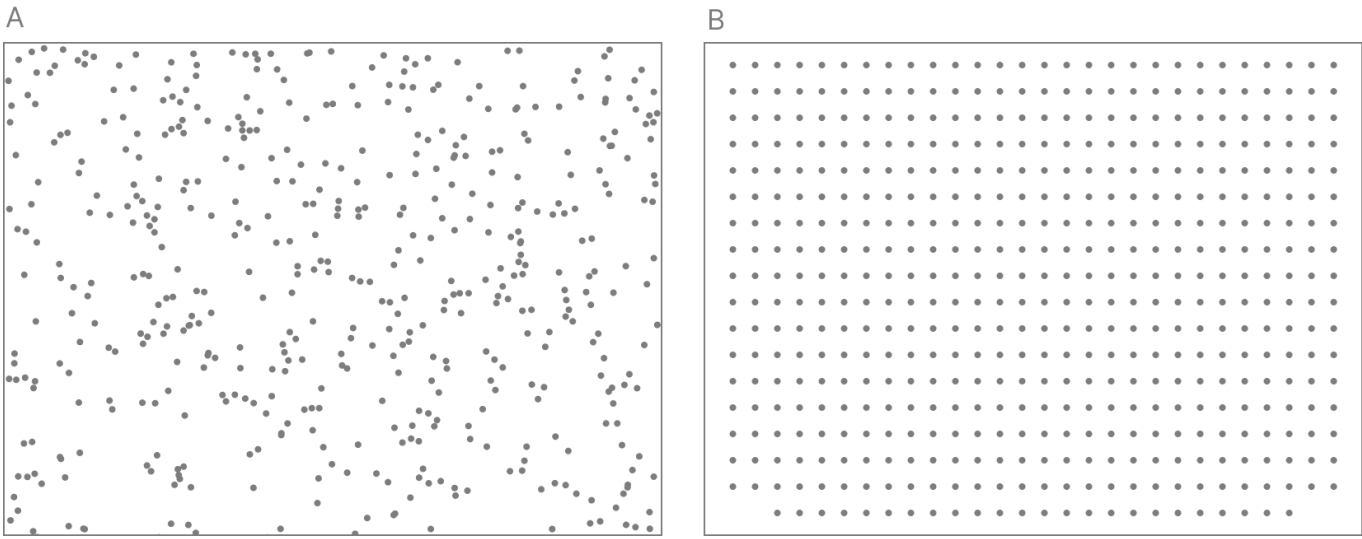
The screenshot shows the Craigslist homepage for New York. It features a top navigation bar with various filters and a main content area with a grid of category links. The categories include: community, housing, jobs, services, discussion forums, and gigs. Each category has a list of sub-links, such as 'last-found', 'missed', 'connections', 'artists', 'childcare', 'classes', 'events', 'general', 'groups', 'local news', 'automotive', 'beauty', 'cell/mobile', 'computer', 'creative', 'cycle', 'event', 'farm/garden', 'financial', 'health/well', 'household', 'legal', 'lessons', 'marriage', 'pet', 'real estate', 'skilled trade', 'sm biz adv', 'travel/vac', 'write/ed/tran', 'apts / housing', 'housing swap', 'office / commercial', 'parking / storage', 'real estate for sale', 'rooms / shared', 'rooms wanted', 'sublets / temporary', 'vacation rentals', 'accounting+finance', 'admin / office', 'arch / engineering', 'art / media / design', 'biotech / science', 'business / mgmt', 'customer service', 'education', 'etc / misc', 'food / bev / hosp', 'general labor', 'government', 'human resources', 'legal / paralegal', 'manufacturing', 'marketing / pr / ad', 'medical / health', 'nonprofit sector', 'real estate', 'retail / wholesale', 'sales / biz dev', 'salon / spa / fitness', 'security', 'skilled trade / craft', 'software / rpl / dba', 'systems / network', 'technical support', 'transport', 'travel', 'tv / film / video', 'web / info design', 'writing / editing', 'computer', 'event', 'creative', 'labor', 'talent'.

The screenshot shows the McMaster-Carr product page for 'drill bushings'. The page includes a search bar, a filter system, and a detailed product listing. The filter system allows users to filter by 'Inch' and 'Metric'. The product listing table includes columns for 'Size', 'Equivalent', 'Overall', 'OD', 'Hex Key Size', 'Material', and 'Each'. The table lists various drill bit sizes and their corresponding specifications.

Size	Equivalent	Overall	OD	Hex Key Size	Material	Each
1/16"	0.0625"	1/2"	3/8"	3/32"	Steel	8865A101 \$3.51
5/64"	0.0781"	1/2"	3/8"	3/32"	Steel	8865A107 3.55
3/32"	0.0938"	1/2"	3/8"	3/32"	Steel	8865A116 3.51
7/64"	0.1094"	1/2"	3/8"	3/32"	Steel	8865A124 3.55
1/8"	0.1250"	1/2"	3/8"	3/32"	Steel	8865A132 3.51
9/64"	0.1406"	1/2"	7/16"	3/32"	Steel	8865A136 3.80
1/8"	0.1563"	1/2"	7/16"	3/32"	Steel	8865A143 3.51
13/64"	0.1719"	1/2"	7/16"	3/32"	Steel	8865A151 3.80
9/64"	0.1875"	1/2"	7/16"	3/32"	Steel	8865A157 3.51
13/64"	0.2031"	1/2"	1/2"	3/32"	Steel	8865A168 4.20
7/32"	0.2188"	1/2"	1/2"	3/32"	Steel	8865A172 3.51
15/64"	0.2344"	1/2"	1/2"	3/32"	Steel	8865A176 4.20
1/4"	0.2500"	1/2"	1/2"	3/32"	Steel	8865A182 3.51
9/32"	0.2813"	5/8"	5/8"	3/32"	Steel	8865A185 5.45
19/64"	0.2969"	5/8"	5/8"	3/32"	Steel	8865A192 5.45
5/16"	0.3125"	5/8"	5/8"	3/32"	Steel	8865A195 5.45
21/64"	0.3281"	3/4"	3/4"	3/32"	Steel	8865A198 5.45
11/32"	0.3438"	3/4"	3/4"	3/32"	Steel	8865A204 6.20
23/64"	0.3594"	3/4"	3/4"	3/32"	Steel	8865A208 5.45
3/8"	0.3750"	7/8"	7/8"	3/32"	Steel	8865A213 6.20
25/64"	0.3906"	7/8"	7/8"	1/8"	Steel	8865A216 5.45
13/32"	0.4063"	7/8"	7/8"	1/8"	Steel	8865A221 7.85
27/64"	0.4219"	7/8"	7/8"	1/8"	Steel	8865A225 7.85
7/16"	0.4375"	7/8"	7/8"	1/8"	Steel	8865A227 7.85
29/64"	0.4531"	7/8"	7/8"	1/8"	Steel	8865A229 7.85
15/32"	0.4688"	7/8"	7/8"	1/8"	Steel	8865A231 7.85
31/64"	0.4844"	7/8"	7/8"	1/8"	Steel	8865A232 7.85
1/2"						8865A234 7.85

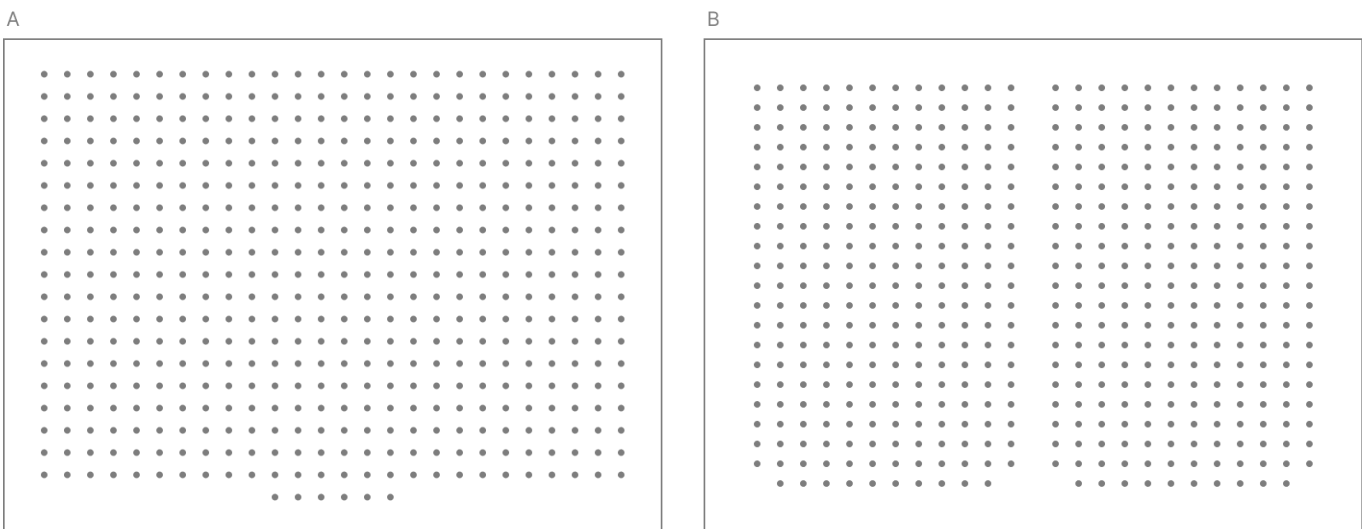
Screenshots of Craigslist's homepage and McMaster-Carr's product page circa 2024.

You can form an opinion about the density of these websites simply by looking at an image for a fraction of a second. This opinion is from our subconsciousness, so it's fast and intuitive. But like other snap judgements, it's biased and unreliable. For example, which of these images is more dense?



Both images have the same number of dots (500). Both take up the same amount of space. But at first glance, most people say image B looks more dense.¹

What about these two images?



Again, both images have the same number of dots, and are the same size. But organizing the dots into groups changes our perception of density. Visually density — our first, instinctual judgement of density — is unpredictable.

It's impossible to be fully objective in matters of design. But if we want to have conversations about density, we should aim for the most consistent, meaningful, and useful definition possible.

Information density

In *The Visual Display of Quantitative Information*, Edward Tufte approaches the design of charts and graphs from the ground up:

Every bit of ink on a graphic requires reason. And nearly always that reason should be that the ink presents new information.

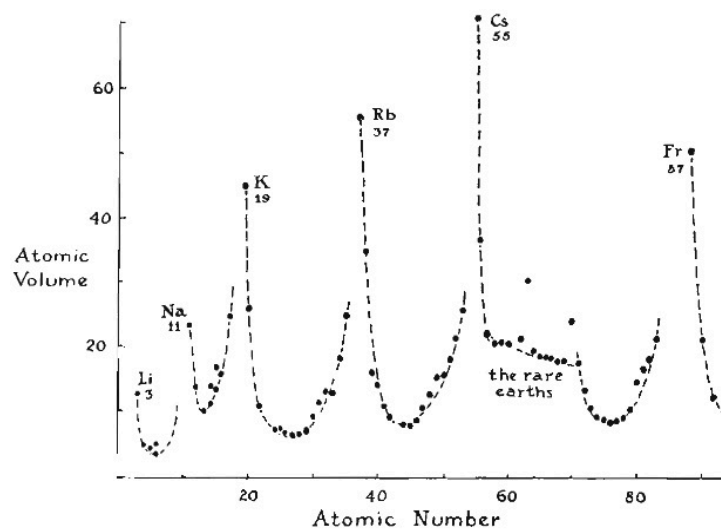
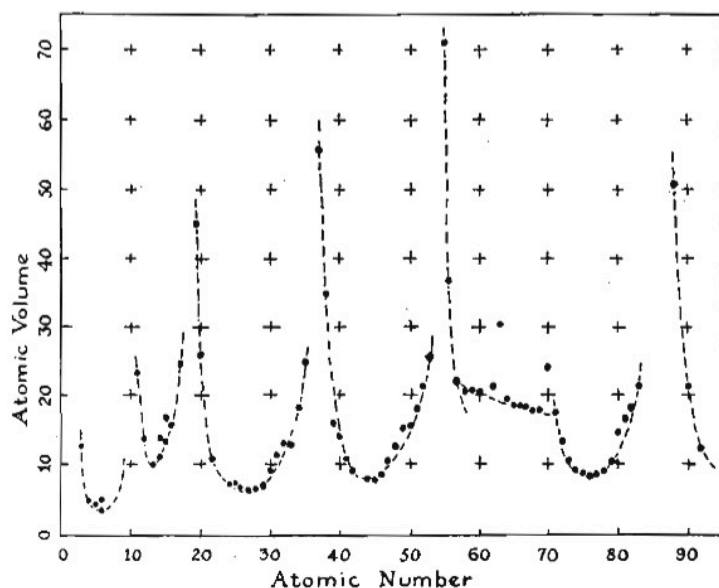
Tufte introduces the idea of “data-ink,” defined as the useful parts of a given visualization. Tufte argues that visual elements that don’t strictly communicate data, whether it’s a scale value, a label, or the data itself — should be eliminated.

Data-ink isn’t just the space a chart takes up. Some charts use very little extraneous ink, but still take up a lot of physical space. Tufte is talking about **information density**, not visual density.

Information density is a measurable quantity: to calculate it, you simply divide the amount of “data-ink” in a chart by the total amount of ink it takes to print it. Of course what is and is not data-ink is somewhat subjective, but that’s not the point. The point is to get the ratio as close to 1 as possible.

You can increase the ratio in two ways:

1. Add data-ink: provide additional (useful) data
2. Remove non-data-ink: erase the parts of the graphic that don’t communicate data



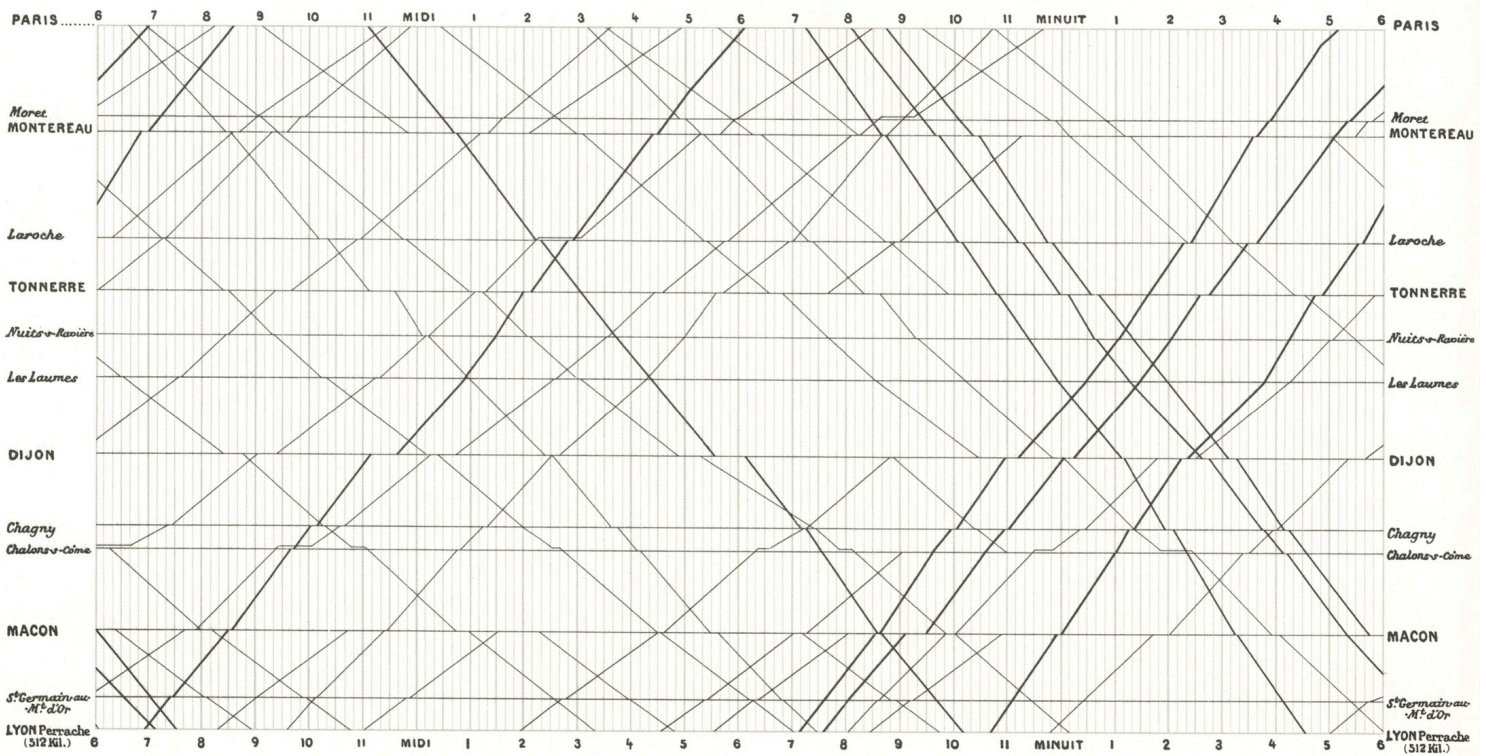
Tufte's examples of graphics with a low data-ink ratio (first) and a high one (second). Reproduced from Edward Tufte's *The Visual Display of Quantitative Information*

There’s an upper limit to information density, which means you can subtract too much ink, or add too much information. The audience matters, too: A bond trader at their 4-monitor desk will have a pretty high threshold; a 2nd grader reading a textbook will have a low one.

Information density is can be related to visual density. Usually, the higher the information density is, the more dense a visualization will look.

For example, take the train schedule published by E.J. Marey in 1885². It shows the arrival and departure times of dozens of trains across 13 stops from Paris to Lyon. The horizontal axis is time, and the vertical axis is space. The distance between stops on the chart reflects how far apart they are in the real world.

The data-ink ratio is close to 1, allowing a huge amount of information — more than 260 arrival and departure times — to be packed into a relatively small space.



The train schedule visualization published by E.J. Marey in 1885. Reproduced from Edward Tufte's *The Visual Display of Quantitative Information*

Tufte makes this idea explicit:

Maximize data density and the [amount of data], within reason (but at the same time exploiting the *maximum resolution* of the available data-display technology).

He puts it more succinctly as the “Shrink Principle”:

Graphics can be shrunk way down

Information density is clearly useful for charts and graphs. But can we apply it to interfaces?

The first half of the equation — information — applies to screens. We should maximize the amount of information that each part of our interface shows.

But the second half of the equation — ink — is a bit harder to translate. It’s tempting to think that pixels and ink are equivalent. But any interface with more than a few elements needs separators, structural elements, and signposts to help a user understand the relationship each piece has to the other.

It’s also tempting to follow Tufte’s Shrink Principle and try to eliminate all the whitespace in UI. But some whitespace has meaning almost as salient as the darker pixels of graphic elements. And we haven’t even touched on shadows, gradients, or color highlights; what role do they play in the data-ink equation?

So, while information density is a helpful stepping stone, it's clear that it's only part of the bigger picture. How can we incorporate *all* of the design decisions in an interface into a more objective, quantitative understanding of density?

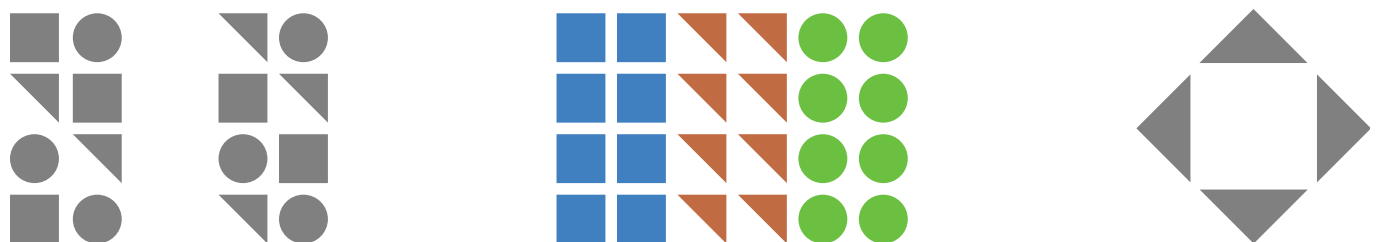
Design density

You might have already seen the first challenge in defining density in terms of design decisions: what counts as a decision decision?

In UI, UX, and product design, we make many decisions, consciously and subconsciously, in order to communicate information and ideas. But why do those particular choices convey the meaning that they do? Which ones are superlative or simply aesthetic, and which are actually doing the heavy lifting?

These questions sparked that led 20th century German psychologists to explore how humans understand and interpret shapes and patterns. They called this field “gestalt,” which in German means “form.” In the course of their exploration, Gestalt psychologists described principles that describe how some things appear orderly, symmetrical, or simple, while others do not. While these psychologists weren't designers, in some sense, they discovered the fundamental laws of design:

1. **Proximity:** we perceive things that are close together as comprising a single group
2. **Similarity:** objects that are similar in shape, size, color, or in other ways, appear related to one another.
3. **Closure:** our minds fill in gaps in designs so that we tend to see whole shapes, even if there are none
4. **Symmetry:** if we see shapes that are symmetrical to each other, we perceive them as a group formed around a center point
5. **Common fate:** when objects move, we mentally group the ones that move in the same way
6. **Continuity:** we can perceive objects as separate even when they overlap
7. **Past experience:** we recognize familiar shapes and patterns even in unfamiliar contexts. Our expectations are based on what we've learned from our past experience of those shapes and patterns.
8. **Figure-ground relationship:** we interpret what we see in a three-dimensional way, allowing even flat 2d images to have foreground and background elements.



Examples of the principles of proximity (left), similarity (center), and closure (right).

Gestalt principles explain why UI design goes beyond the pixels on the screen. For example:

- Because of the **principle of similarity**, users will understand that text with the same size, font, and color serves the same purpose in the interface.
- The **principle of proximity** explains why when a chart is close to a headline, it's apparent that the headline refers to the chart. For the same reasons, a tightly packed grid of elements will look related, and separate from a menu above it separated by ample space.
- Thanks to our **past experience** with switches, combined with the figure-ground principle, a skeuomorphic design for a toggle switch will make it obvious to a user how to instantly turn on a feature.

So, instead of focusing on the pixels, we think of design decisions as how we intentionally use gestalt principles to communicate meaning. And like Tufte's data-ink ratio compares the strictly necessary ink to the total ink used to print a chart, we can calculate a gestalt ratio which compares the strictly necessary design decisions to the total decisions used in a design. This is **design density**.

Four different treatments of the same information, using different types and amounts of gestalt principles. Which is the most dense?

This is still subjective: a design decision that seems necessary to some might be superfluous to others. Our biases will skew our assessment, whether they're personal tastes or cultural norms. But when it comes to user interfaces, counting design decisions is much more useful than counting the amount of data or "ink" alone.

Design density isn't perfect. User interfaces exist to do work, to have fun, to waste time, to create understanding, to facilitate personal connections, and more. Those things require the user to take one or more actions, and so density needs to look beyond components, layouts, and screens. Density should comprise all the actions a user takes in their journey — it should count in space and time.

Density in time

Just like the amount of stuff in a given space dictates visual density, the amount of things a user can do in a given amount of time dictates **temporal** — time-wise — density.

Loading times are the biggest factor in temporal density. The faster the interface responds to actions and loads new pages or screens, the more dense the UI is. And unlike 2-dimensional whitespace, there's almost no lower limit to the space needed between moments in time.

Bloomberg's Terminal loads screens full of data instantaneously

With today's bloated software, making a UI more dense in time is more impactful than just squeezing more stuff onto each screen. That's why Bloomberg's Terminal is still such a dominant tool in the financial analysis space; it loads data almost instantaneously. A skilled Terminal user can navigate between dozens of charts and graphs in milliseconds. There are plenty of ways to cram tons of financial data into a table, but loading it with no latency is Terminal's real superpower.

But say you've squeezed every second out of the loading times of your app. What next? There are some things that just can't be sped up: you can't change a user's internet connection speed, or the computing speed of their CPU. Some operations, like uploading a file, waiting for a customer support response, or processing a payment, involve complex systems with unpredictable variables. In these cases, instead of changing the amount of time between tasks, you can change the **perception** of that time:

- Actions **less than 100 milliseconds** apart will feel simultaneous. If you tap on an icon and, 100ms later, a menu appears, it feels like no time at all passed between the two actions. So, if there's an animation between the two actions — the menu slides in, for example — the illusion of simultaneity might be broken. For the smallest temporal spaces, animations and transitions can make the app feel *slower*.³
- Between **100 milliseconds and 1 second**, the connection between two actions is broken. If you tap on a link and there's no change for a second, doubt creeps in: did you actually tap on anything? Is the app broken? Is your internet working? Animations and transitions can bridge this perceptual gap. Visual cues in these spaces make the UI feel more dense in time.
- Gaps between **1 and 10 seconds** can't be bridged with animations alone; research⁴ shows that users are most likely to abandon a page within the first 10 seconds. This means that if two actions are far enough apart, a user will leave the page instead of waiting for the second action. If you can't decrease the time between these actions, show an indeterminate loading indicator — a small animation that tells the user that the system is operating normally.
- Gaps between **10 seconds and 1 minute** are even harder to fill. After seeing an indeterminate loader for more than 10 seconds, a user is likely to see it as static, not dynamic, and start to assume that the page isn't working as expected. Instead, you can use a determinate loading indicator — like a larger progress bar — that clearly indicates how much time is left until the next action happens. In fact, the right design can make the waiting time seem shorter than it actually is; the backwards-moving stripes that featured prominently in Apple's "Aqua" design system made waiting times seem 11% shorter.⁵
- For gaps **longer than 1 minute**, it's best to let the user leave the page (or otherwise do something else), then notify them when the next action has occurred. Blocking someone from doing anything useful for longer than a minute creates frustration. Plus, long, complex processes are also susceptible to error, which can compound the frustration.

In the end, though, making a UI dense in time and space is just a means to an end. No UI is valuable because of the way it looks. Interfaces are valuable in the outcomes they

enable — whether directly associated with some dollar value, in the case of business software, or tied to some intangible value like entertainment or education.

So what is density really about, then? It's about providing the highest value outcomes in the smallest amount of time, space, pixels, and ink.

Density in value

Here's an example of how value density is manifested: a common suggestion for any form-based interface is to break long forms into smaller chunks, then put those chunks together in a wizard-type interface that saves your progress as you go. That's because there's no value in a partly-filled-in-form; putting all the questions on a single page might look more visually dense, but if it takes longer to fill out, many users won't submit it at all.

This form is broken up into multiple parts, with clear errors and instructions for resolution.

Making it possible for users to get to the end of a form with fewer errors might require the design to take up more space. It might require more steps, and take more time. But if the tradeoffs in visual and temporal density make the outcome more valuable — either by increasing submission rate or making the effort more worth the user's time — then we've increased the overall value density.

Likewise, if we can increase the visual and temporal density by making the form more compact, load faster, and less error-prone, *without* subtracting value to the user or the business, then that's an overall increase in density.

Channeling Tufte, we should try to increase value density as much as possible.

Solving this optimization problem can have some counterintuitive results. When the internet was young, companies like Craigslist created value density by aggregating and curating information and displaying it in pages of links. Companies like Yahoo and Altavista made it possible to search for that information, but still put aggregation at the fore. Google took a radically different approach: use information gleaned by the internet's long chains of linked lists to power a search box. Information was aggregating itself; a single text input was all users needed to access the entire web.

Google and Yahoo's approach to data, design, and value density hasn't changed from 2001 (when the first screenshots were archived) to 2024 (when the second set of screenshots were taken). The value of the two companies' stocks reflect the result of these differing approaches.

The UI was much less visually dense, but more value-dense by orders of magnitude. The results speak for themselves: Google went from a \$23B valuation in 2004 to being worth over \$2T today — closing in on a 100x increase. Yahoo went from being worth \$125B in 2000 to being sold for \$4.8B — less than 3% of its peak value.⁶

Conclusion

Designing for UI density goes beyond the visual aspects of an interface. It includes all the implicit and explicit design decisions we make, and all the information we choose to show on the screen. It includes all time and the actions a user takes to get something valuable out of the software.

So, finally, a concrete definition of UI density: **UI density is the value a user gets from the interface divided by the time and space the interface occupies.**

Speed, usability, consistency, predictability, information richness, and functionality all play an important role in this equation. By taking account of all these aspects, we can understand why some interfaces succeed and others fail. And by designing for density, we can help people get more value out of the software we build.

Footnotes & References

1. This is a very unscientific statement based on a poll of 20 of my coworkers. Repeatability is questionable. ↩
 2. The provenance of the chart is interesting. Not much is known about the original designer, Charles Ibry; but what we do know points to even earlier iterations of the design. If you're interested, read [Sandra Rendgen's fascinating history of the train schedule](#). ↩
 3. I have no scientific backing for this claim, but I believe it's because a typical blink occurs in 100ms. When we blink, our brains fill in the gap with the last thing we saw, so we don't notice the blink. That's is why we don't notice the gap between two actions that are less than 100ms apart. You can read more about this effect here: [Visual Perception: Saccadic Omission — Suppression or Temporal Masking?](#) ↩
 4. Nielsen, Jakob. "How Long Do Users Stay on Web Pages?" Nielsen Norman Group, 11 Sept. 2011, <https://www.nngroup.com/articles/how-long-do-users-stay-on-web-pages/> ↩
 5. Harrison, Chris, Zhiquan Yeo, and Scott E. Hudson. "Faster Progress Bars: Manipulating Perceived Duration with Visual Augmentations." Carnegie Mellon University, 2010, <https://www.cbrisharrison.net/projects/progressbars2/ProgressBarsHarrison.pdf> ↩
 6. HackerNews as pointed out that this is a ridiculous statement. And it is. Of course, value density isn't the *only* reason why Google succeeded where Yahoo failed. But as a reflection of how each company thought about their products, it was a good leading indicator. ↩
-

April 28, 2024

The polish paradox

Join the mailing list

I'll send new posts to your inbox, along with links to related content and a song recommendation or two.

Email

Subscribe

[Twitter](#) · [RSS](#) · [Email](#)