Dashboard Design Patterns

Dashboard Design Workshop

Benjamin Bach, University of Edinburgh

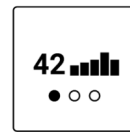*https://ashboarddesignpatterns.github.io*

Static	Analytical	Magazine	Infographic	Mini	Slideshow	Repository

Content Dashboard Design Patterns

Data

Less Data

- Single value
- Derived values
- Thresholds
- Filtered
- Aggregated
- Detailed Data

More Data

Meta Data

- Data Source
- Disclaimer
- Data Description
- Update Information
- Annotations

Visual Representation

More Detail

- List
- Table
- Detailed Visualization
- Miniature Chart
- Progress Bars & Gauge
- Trend arrow
- Pictogram
- Number

Less Detail

Composition Dashboard Design Patterns

Page Layout

- Open
- Table
- Stratified
- Grouped
- Schematic

Screenspace

- Screenfit
- Overflow
- Detail on demand
- Parameterization
- Mulitple pages

Structure

- Single Page
- Parallel
- Hierarchic
- Open

Interaction

- Exploration
- Navigation
- Personalization
- Drilldown

Color

- Distinct
- Data Encoding
- Semantic
- Emotive

Dashboard Genres

Static Dashboard

This is the traditional notion of a dashboard as a *non-interactive* and *flat* structured display of information. These are less common now than one might imagine, which we attribute to the fact that modern dashboards are digital and it is easy to support interaction and drill-down tasks through more complex structures. Another reason might be that the range of display sizes on desktop computers, tablets, and mobiles encourages adaptive solutions (e.g., use of overflow or paginated structures).
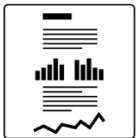
Magazine Dashboard

Many dashboards relating to Covid-19, climate change, politics, etc, are typically created by news agencies and similar media outlets. These dashboards are found as an integral part of journalistic articles and resemble visualizations of the *magazine* genre. The text goes beyond basic meta information to provide additional commentary and storytelling about the data. These dashboards are often broken into several pages and have an **overflow** page structure with **linear layout**, with visualizations positioned at appropriate points in the text to tell a story about what the data shows.

Infographic Dashboard

These dashboards are similar to infographics in that they focus on narration. They can include decorative graphical elements shown alongside data representations. Similar to magazine dashboards, they use non-data media to annotate and embellish data. These dashboards may thus have a different intended use, with an audience expected to discover them over a longer period of time, rather than checking in frequently for updates.

Mini-dashboard

Dashboards can be embedded into other applications such as news websites. These concise *mini* dashboards only occupy a small area on screen and usually come with a range of interactive features for navigation, or to parameterize the content.

Analytic Dashboard

This type generally uses complete visualizations (rather than simpler signature charts and trend arrows). Many of the dashboard elements are fully interactive, providing for pan+zoom, focus+context, tooltips, brushing+linking and other exploration and navigation strategies. These dashboards can also provide parameterization, and use tabs or linking to switch between *multiple pages* of the dashboard.

Repository dashboard

Many dashboards list a multitude of charts on a single website, with overflow page structures that make proper analytics difficult, i.e., making it more challenging to compare views. Their charts often lack textual or other narrative explanations, except for meta data information (which is often extensive). Charts may provide some interaction and usually provide links to *explore*, *filter*, and eventually *download* the raw data. Extensive meta information is often provided for transparency and to support reuse. The images below show two examples of repository dashboard.

Data Abstraction

| Detailed Dataset | Aggegated Data | Thresholds & Filters | Derived Data | Individual Values |

More Data ⟷ **Less Data**

This group captures the type of information shown in a dashboard. We found that information in dashboards ranges from the presentation of *raw data* to several levels of *abstractions* which simplify and reduce the amount of information shown. In other words, given a data set with multiple data elements (e.g., rows in a data table) and attributes (columns), dashboards can provide many different abstractions of this data set.
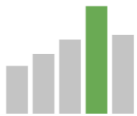
Individual values are specific data points in a data set, e.g., the most recent value in a time series. Showing an individual value (compared to the entire or a filtered data set) implies the importance of the single, e.g., most recent, or highest, value in a given context.

Derived values include information derived from datasets or other values (e.g., trends). These provide a degree of abstraction over a data set, since they imply a specific interest in these derived values rather than the raw data. For example, derived values can be Key Performance Indicators in a given context. In many cases, derived values have been calculated independently of the dashboard (e.g., incidence values per fixed population).
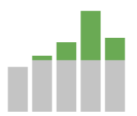
Thresholds make an explicit judgment about a data point, e.g., indicating whether a situation is *good*, *bad*, or *neutral*, according to objective thresholds for subjective labels. Thresholds are one of the highest levels of abstraction found in dashboards, as thresholds can have specific semantic meaning and implications. **Filters** show only relevant parts of a larger data set.

Aggregated data are the result of bringing together numerous data points into a new and more concise data point. For example, data values from the individual regions of a country may be aggregated into a sum or mean value about the country as a whole. Aggregation often implies the presence of multiple related data values, which may be of particular interest to a dashboard user.

Detailed datasets offer a more complete presentation of data. This can include multiple data elements and attributes, time series, and detailed geographic information which all show raw data. In the case of time series, a dataset is often filtered to a time range of relevance, e.g., the *last 14 days*. Showing raw data implies lots of details, but little "judgment" about which characteristics of this data set are important. Datasets are shown to provide an overview, support comparison of multiple data sets, and to encourage individual exploration and analysis.

Visual Representation

List

Table

Detailed Visualization

Miniature Chart

Progress Bars Gauges

Indicators

Trend arrow(s)

Pictogram

42

Numbers

More Detail ←————————————————→ **Less Detail**
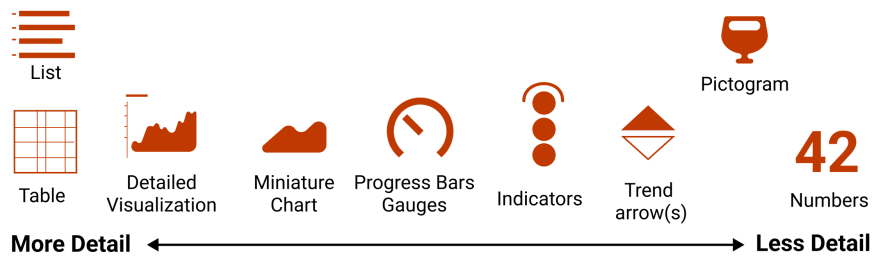
42 **Numbers:** individual numbers placed prominently on a dashboard, mostly used to indicate single key values. Numbers can represent absolute values, proportions (e.g., percentage or normalized *per million*) and have specific units. Numbers are usually large in text size, compared to, e.g., titles and descriptions, to draw attention to their value. They are sometimes colored to attract attention.

Trend-arrows: small arrows pointing up or down, used to indicate the direction of change in a data value. Some trend arrows are binary (up/down/neutral), whilst others include more fine-grained slopes. Trend arrows are often red or green, depending on semantics (positive/negative). These are often shown alongside *numbers* or included in *tables*.

Pictograms: abstract representations or symbols that illustrate concepts in the data. Can be used to represent data (i.e., conveying specific data information), or to designate the type of data value found near the pictogram (but not conveying specific data information).

Gauges and progress bar: any sort of visualization used to represent a scale or range of a value. Specific solutions include semicircular gauges, linear progress bars, thermometers, etc. Values represented by range-representations can refer to both values that only grow (e.g,. vaccination of population), as well as fluctuating values (e.g., COVID-19 cases per 1000 people). Some gauges come with an indication of 'critical' ranges, i.e,. a *threshold* indicating if values are positive/negative.

Signature Charts: small and concise visualizations without axis descriptions, labels, or tickmarks. The idea is to give a quick understanding of a trend, akin to sparklines, rather than allowing the reading of precise values.

Detailed Charts: detailed visualizations with sufficient details to read and understand precise values (e.g., axes, ticks, values, etc)

Tables: these show raw data values in a tabular format. The most simple form of a table is a simple key-value list, but tables can show several columns and may include *signature charts* and/or *trend-arrows*. Tables do not need to contain numerical values, but can also include information through color/darkness (e.g., heatmaps).

Text Lists: lists of textual information, e.g.,like tickers that show non-quantitative information.

Structure Patterns

These **page structure** patterns describe how a dashboard organizes information across multiple pages, and the implied relationships between those pages. These patterns may influence how a user navigates through the various pages of a dashboard, and may also be influenced by the intended use of the dashboard (e.g., providing a *hierarchical* page structure so that users can 'drill down' into increasing levels of detail, or using a *parallel* page structure when users only need a high-level overview of small number of views).

Single Page: a dashboard presents all information on one page, e.g., using *screenfit* or *overflow* to utilise available screenspace.

Hierarchical: a dashboard has multiple pages organized in a hierarchy that implies a structured relationship between pages and the information they convey. Dashboard pages at adjacent levels of the hierarchy might, for example, differ in the amount of detail they show and support users in *drilling down* to greater levels of detail (e.g., abstract representations in a top-level page vs detailed representations in a low-level page). Dashboard pages from the same level of the hierarchy will typically be related in some way (e.g., a top-level page showing data from a whole country, with multiple lower-level pages showing data from the individual regions of that country).

Parallel: a dashboard has information distributed across multiple pages, all considered to be at the same 'level' with no hierarchical relationships between them. For example, each page might represent distinct facets of the data set. Navigation between pages is typically unconstrained via top-level navigation widgets like *tabs* or *links*.

Open: a dashboard has multiple pages but with less rigid relationships between them. Contrast with a *hierarchical* or *parallel* dashboard structure, where there are more clearly defined relationships between pages and the data information they show.

Semantic: a dashboard with multiple pages, whose relationships are determined by the semantics of the information being shown. For example, a dashboard for an organization may have pages relating to distinct divisions within the company, with pages following the operational structure of those divisions.

Layout Patterns

Layout patterns describe how the widgets are laid out and sometimes implicitly grouped together in a dashboard view - i.e., the prominent layout decision used to group a potentially large set of visual components. We emphasize that none of these layout patterns are exclusive and combinations can be used. Note: when describing layout patterns, we define a *widget* as the smallest unit on screen.

Open Layouts: widgets of different sizes and aspect ratios are laid out in an open way without apparent specific rules. Often these widgets are aligned at a grid following classical design guidelines. There is no strong semantic associated with the location and adjacency of widgets and each widget seems to have equal importance. Widgets can be arranged closely if they are related (e.g., first and second COVID-19 vaccine dose) or they have similar visual types for aesthetics reason (e.g., top rows are all gauges and bottom rows are all line charts). Tiled layouts invite reordering and adjustment of widgets by the user.

Table Layouts: these align widgets (or groups of widgets) into columns and rows, each with a specific semantic and inducing a repetition of information and visual encoding. Rows and columns can represent facets of the data set such as time, data elements, or any other grouping. Table layouts make it easy to retrieve and relate specific information in a multidimensional dataset. Usually, the visual content along a column uses the same visual encoding. Table layouts imply direct comparison and juxtaposition of information across widgets.

Stratified Layouts: these emphasize a top-down ordering of widgets and their information. For example, dashboards may show general high-level information along the top, while showing more detailed information further down. A strata layout can be used to emphasize some information (e.g., further up) over other information (e.g., further down).

Grouped Layouts: these visibly group two or more widgets with a specific relation. For example, using diving lines, whitespace borders, or shaded background regions to emphasize grouping of related widgets. A dashboard with clear explicit grouping may help viewers observe the data more easily, especially for intended public audience.

Schematic Layouts: these place widgets in a dashboard informed by some external property such as a physical layout or other types of schematic relationships (e.g., widgets placed on a map to represent geographic location of data points). Such dashboards can leverage a user's spatial/geographical/etc knowledge to support visual analytics tasks.

Screenspace
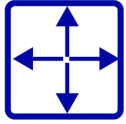
Screenfit   Overflow   Detail on   Parameter   Mulitple
                         demand   -ization   pages

More concise ←——————————→ **Less concise**

Dashboards can only show a certain amount of information on screen at a single time: available screen space is consumed by the amount of *data information* to be shown and the dimensions of their *visual encodings*. Whilst some dashboards can fit entirely on screen, it may be necessary to adopt display strategies that reduce the amount of information shown on screen at a given time. We identify a range of strategies for making use of available screen space, ranging from concise **screenfit** dashboards that fit entirely on screen, to less concise dashboards with **multiple pages** of information. These strategies are not mutually exclusive and can be combined, e.g., using **detail-on-demand** to make more economical use of screenspace in a **screenfit** dashboard.
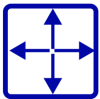
 Screenfit dashboards are fully visible on screen and all information is visible at all times. No interaction is needed to view any of the dashboard content, as all information fits on screen and is available at a glance.

 Overflow designs allow a dashboard to exceed the available screen boundaries, so that additional content is rendered off-screen and is revealed as the user scrolls. This approach allows all dashboard views to be made available on a single page, even if not all are visible on screen at once.

 Detail-on-demand interactions allow dashboards to reduce the amount of information shown on screen at once, by revealing more details only when the user asks (for example, using tooltips or pop-ups).

 Parameterized dashboards allow users to control what content is currently shown on screen. We found several approaches for providing dashboard parameters, including checkboxes, drop-down menus, lists and search fields for filtering data, sliders and pickers for selecting date ranges, or radio buttons and switches for changing the data. These typically allow users to choose which data is shown through the dashboard widgets.

 Multiple page dashboards are composed of elements distributed across several pages, with one page shown on screen at a time; here, 'page' refers to a single composite view within a dashboard and does not necessarily mean, e.g., distinct web pages. There were many strategies for presenting an overview of available dashboard pages and allowing viewers to navigate between them, including *tabs*, *navigation buttons* and *links*

Interaction

Exploration   Navigation   Personal-
ization   Filter &
Focus

This group describes common interaction approaches found within the dashboards. Interaction can exist through interactive data, user interface elements, and window-level interactions (e.g., scrolling). The patterns we highlight in this group refer to common roles that interaction could play in dashboard use, expressed through specific user interface components. These are defined broadly, to identify general usage patterns and how they can be implemented through dashboard designs.

Exploration interactions allow users to explore data elements and the relations between them. Exploration interactions can take on my forms, including *brushing and linking* interactions that link data representations across different views, and *detail-on-demand* interactions that provide additional information through pop-ups or tooltips.

Navigation interactions enable designers to direct users through facets of information in a particular order. Interaction elements can be used to navigate between different dashboard components and layout structures (e.g., pages, tabs). Interaction techniques and elements such as *scrolling*, *navigation buttons*, or *tab links* can be used to create a journey for the viewers through which they explore the information. Navigation can be *constrained* and follow a particular sequence, or can be *unconstrained* to allow free navigation. Examples of navigation interactions include:

- **Tabs** that inform the user about available content and allows navigation between views.
- **Links** that support navigation through clickable visual elements (e.g., charts, titles, clickable text).
- **Navigation buttons** that allow users to move forwards or backwards through the pages of a dashboard in (or out of) sequence.
- **Animated transitions** that are used to advance through dashboard pages automatically, like a slide show.

Personalization interactions allow users to redefine and reconfigure the information shown within a dashboard. For example, users may be able to add new representations (e.g., by choosing a new data feature to be visualized), resize dashboard elements, or reorder the existing dashboard elements. Such interactions may lead to more specialized and bespoke dashboard configurations, where the viewers are more actively engaged with the information content after the dashboard is designed and deployed.

Filter and Focus interactions allow users to find or focus on specific data, e.g., by searching for particular data values or applying filter criteria so that only relevant information remains. These interactions are typically facilitated by user interface elements like text fields, checkboxes, radio buttons, range sliders, etc.