

# Data as a Feature

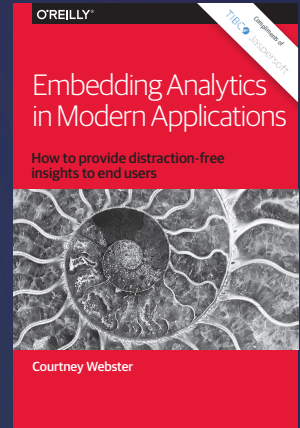
## A Guide for Product Managers



Alice LaPlante & Matt LeMay

# TAKE THE NEXT STEP...

LEARN **HOW** TO INCLUDE DATA AS A  
FEATURE IN YOUR SOFTWARE PRODUCT



GET KEY CONSIDERATIONS AND BEST PRACTICES  
FOR EMBEDDING ANALYTICS IN MODERN APPLICATIONS

---

**Download your free eBook, compliments of TIBCO JasperSoft.**

[www.jaspersoft.com/embedded-analytics-ebook](http://www.jaspersoft.com/embedded-analytics-ebook)

**TIBCO**  **Jaspersoft**<sup>®</sup>

---

# Data as a Feature

## *A Guide for Product Managers*

*Alice LaPlante and Matt LeMay*

Beijing • Boston • Farnham • Sebastopol • Tokyo

**O'REILLY®**

## **Data as a Feature**

by Alice LaPlante and Matt LeMay

Copyright © 2018 O'Reilly Media, Inc. All rights reserved.

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (<http://oreilly.com/safari>). For more information, contact our corporate/institutional sales department: 800-998-9938 or [corporate@oreilly.com](mailto:corporate@oreilly.com).

**Editors:** Tim McGovern

and Rachel Roumeliotis

**Production Editor:** Justin Billing

**Copyeditor:** Octal Publishing, Inc.

**Proofreader:** Amanda Kersey

**Interior Designer:** David Futato

**Cover Designer:** Karen Montgomery

**Illustrator:** Rebecca Demarest

January 2018: First Edition

### **Revision History for the First Edition**

2018-01-11: First Release

The O'Reilly logo is a registered trademark of O'Reilly Media, Inc. *Data as a Feature*, the cover image, and related trade dress are trademarks of O'Reilly Media, Inc.

While the publisher and the authors have used good faith efforts to ensure that the information and instructions contained in this work are accurate, the publisher and the authors disclaim all responsibility for errors or omissions, including without limitation responsibility for damages resulting from the use of or reliance on this work. Use of the information and instructions contained in this work is at your own risk. If any code samples or other technology this work contains or describes is subject to open source licenses or the intellectual property rights of others, it is your responsibility to ensure that your use thereof complies with such licenses and/or rights.

This work is part of a collaboration between O'Reilly and TIBCO JasperSoft. See our *statement of editorial independence*.

978-1-492-02533-7

[LSI]



---

# Table of Contents

<b>Data as a Feature: Turning Data into Your Product’s Most Potent Asset.</b>	<b>1</b>
Overview: Making Sense of Data Overload	1
Data as a Feature	3
Understanding Your Customers’ Goals Through Personas	9
Precisely Crafting the UX	15
Make Your Data “Over the Counter”	19
Managing Your Data Roadmap and Feature Requests	20
Conclusion	22



---

# Data as a Feature: Turning Data into Your Product's Most Potent Asset

## Overview: Making Sense of Data Overload

We live in a world where people expect answers at their fingertips. They want their bank balances to reflect their financial transactions in real time. They want their smart electrical meters to tell them to the penny how much energy they've consumed. So where do they go? To applications! Consumers fire up their banking application on their smartphones or log into their energy company's website on their laptops. There they find the answers they need.

On the business side, companies want to know how well they're serving their customers, how many pallets of inventory are left in a warehouse, or what treatment is proving most effective for patients at a hospital.

We're a long way from the 1970s, when only academics and specialized operational workers—for example, air traffic controllers—understood how to use data correctly (see [Figure 1-1](#)).

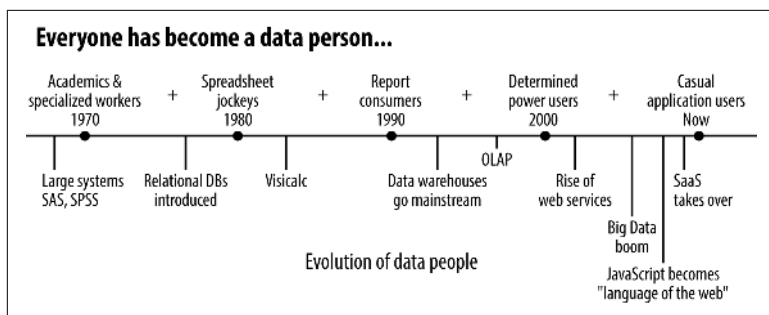


Figure 1-1. Evolution of data people (source: Jaspersoft)

Over the past decade, everybody from nontechnical business leaders to casual consumer technology users have discovered ways in which they can use data to improve their work and their lives. As might be expected, as soon as this change was underway, people began demanding access to ever more data. And until fairly recently, “too much data” was never a concern.

Fast-forward to today. Business users as well as consumers have data bombarding them from all angles: structured data, unstructured data, so-called big data, social media data, data from the Internet of Things (IoT). The challenge for people has shifted from not getting enough data to getting too much. They want to get value out of it but are overwhelmed.

And the challenge for people and organizations building applications mirrors that: how can they help their users get value out of all this data?

The last thing they want is to simply add more data for its own sake. Then, data becomes a burden, not an advantage. Now that data is ubiquitous, the value that products bring is not simply the presence of data. App builders would simply be selling more haystacks for their customers to sift through.

Today, it’s all about making the data work for users, whether business or consumer, to take action, to make decisions, to reach goals.

What does this mean for product managers? Plenty.

Product managers must piece it all together. They need to understand the data requirements of customers while keeping the apps they’re developing aligned with the goals of the business. They need to deliver exemplary user experiences within the confines of their

technical capacities. And they must balance the inputs of their developers, UX designers, and customers with their own visions for their products so that they provide real value for users.

Product managers today thus face a significant addition to their duties. On top of all their other responsibilities, they need to begin treating data as a feature in the products that they're building. In other words, not viewing data as just a byproduct of the apps, but a prominent *feature* of them.

Yes, so-called *design thinking* is important, even critical. But even more so is *goal thinking*: What are users' goals, and how do you present data in a way that helps them to achieve those goals?

In this book, you'll learn why treating data as a feature in your products is a way to make them stand out from the crowd. But standing out is more than just providing beautiful data visualizations. The data needs to help users take action, make decisions, or reach goals. Data for data's sake is no longer good enough. This book will explain why. It will also show you how to use personas, uncover your assumptions, and make your data "over the counter" so that it can be easily understood and valuable to any user.

## Data as a Feature

What is *data as a feature*?

Before we answer that question, let's first define what a software feature is. According to the team behind popular product roadmapping software **Aha!**, a software feature is "a slice of business functionality that has a corresponding benefit or set of benefits for that product's end user."

Data as a feature is then the act and process of treating data as a core feature of a software product in a way that delivers value to the user. By packaging and delivering data effectively in a product, app developers give users critical information, ready them to take action, and help them make better decisions. For product managers who want to treat data as a feature, it's critical to understand the users for whom they are building a product, what their data needs are, and how a specific "slice of business functionality" could help those users meet their needs.

Taking this definition a step further, a product with data as a feature delivers that data in a way that helps the user *meet a goal*.

To help users meet their goals, data as a feature has four attributes: it needs to be *intuitive* (easy to understand), *convenient* (accessible in the right context), *customizable* (viewable in ways unique to each user), and *actionable* (easy to apply insights to produce intended outcomes). The financial management app Mint is a good example of this.

Mint is a personal finance app that helps users meet a pressing and common goal: to gain control of their personal finances. Consumers give Mint access to their financial accounts—banks, credit cards, investments, loans, and other debts. Every transaction is labeled and logged. Mint then automatically pulls and tracks spending activity. Built-in rules and intelligence identify when an activity is potentially suspicious or hazardous, and the app then alerts the user.

Consumers can also create budgets and set personal financial goals. If, for instance, you establish a budget, and a purchase causes you to go over, you receive a message instantly, as shown in [Figure 1-2](#). You can look at a chart and know that you're \$35 over budget in food spending this week, or that you lowered your debt by \$400 this month. And with that easily gleaned information, you can take action. Either that action happens within the app—for example, you can drill into the data to see how you overspent—or in life by causing you to make a behavioral change because you've been enlightened.

Mint.com goes to great lengths to ensure that the complex world of financial data will be both accessible and actionable for all of its users. “Power users” are able to tag and categorize their transactions, set budgets for different types of spending, and manage a portfolio of investments. And the data generated by these “power users” helps improve the experience for casual users. For example, a merchant whose transactions have been manually tagged by a more active Mint user can be automatically categorized when a casual user makes a purchase from the same merchant.

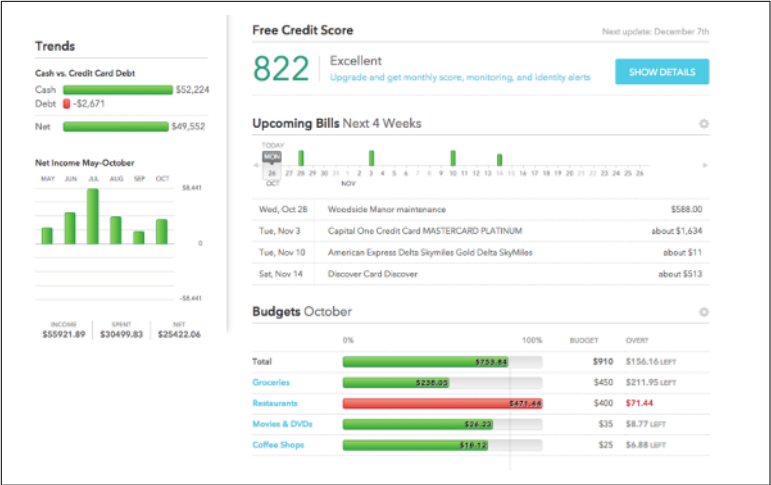


Figure 1-2. Mint.com displays easily absorbed information

The education app Waggle also uses data to create a goals-first experience for its users. Waggle was designed to help educators meet an important objective: to deliver individualized education to students in a world of standardized curricula. Waggle’s visual interface provides educators with immediate information about which students need help. The educators can then drill down to better understand the specific challenges facing each individual student, as demonstrated in Figure 1-3. Waggle is a powerful example of how data as a feature can meet the specific needs of nontechnical users.

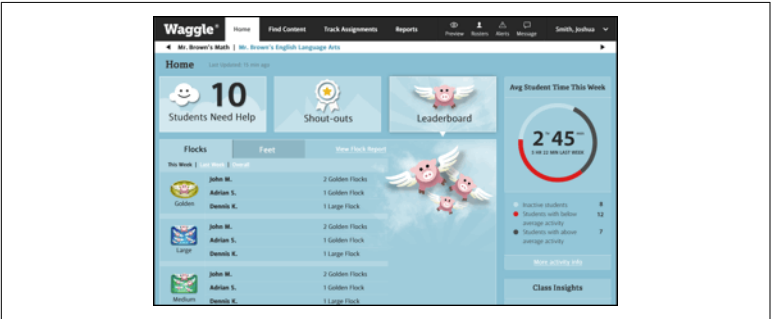


Figure 1-3. Waggle uses data to help educators provide individualized learning to students

Although this kind of easy-to-digest user experience (UX) started in the consumer world, it is spreading. On the business side, Workday



has done the same thing by building data features into its human management application that are explicitly designed to guide users toward making better decisions, as depicted in **Figure 1-4**.



*Figure 1-4. Workday uses intuitive data visualizations and labeling to make complex data easy to understand*

## The Importance of Goal-Based Thinking

No matter how elegant, mathematically advanced, or visually compelling data might be, it cannot succeed unless it helps users to meet a specific set of goals or needs. In other words, data as a feature should make people's lives easier, not more complicated.

Capital One Financial Corporation knew that its customers were concerned about fraudulent transactions. **But simply providing a list of monthly charges wasn't enough.** Customers found it unhelpful—and tedious—to be presented with long lists of alphanumeric data on charges every month. They wanted to be able to click and immediately stop a fraudster from making erroneous charges against their credit cards or checking accounts. Enter Second Look, an app Capital One developed to identify and flag suspicious charges.

Second Look sends notifications to users alerting them when potentially suspicious charges accrue—for example, a charge in a distant city or a sudden rise in a utility bill. But the app does more than notify the user. It displays the suspect item visually on the screen, and, better yet, it asks whether the user is OK with the charge and provides two large and intuitively colored buttons with which the

user can take an action. If the user selects the red “This looks wrong” button (Figure 1-5), the app provides instructions on how to dispute the charge.

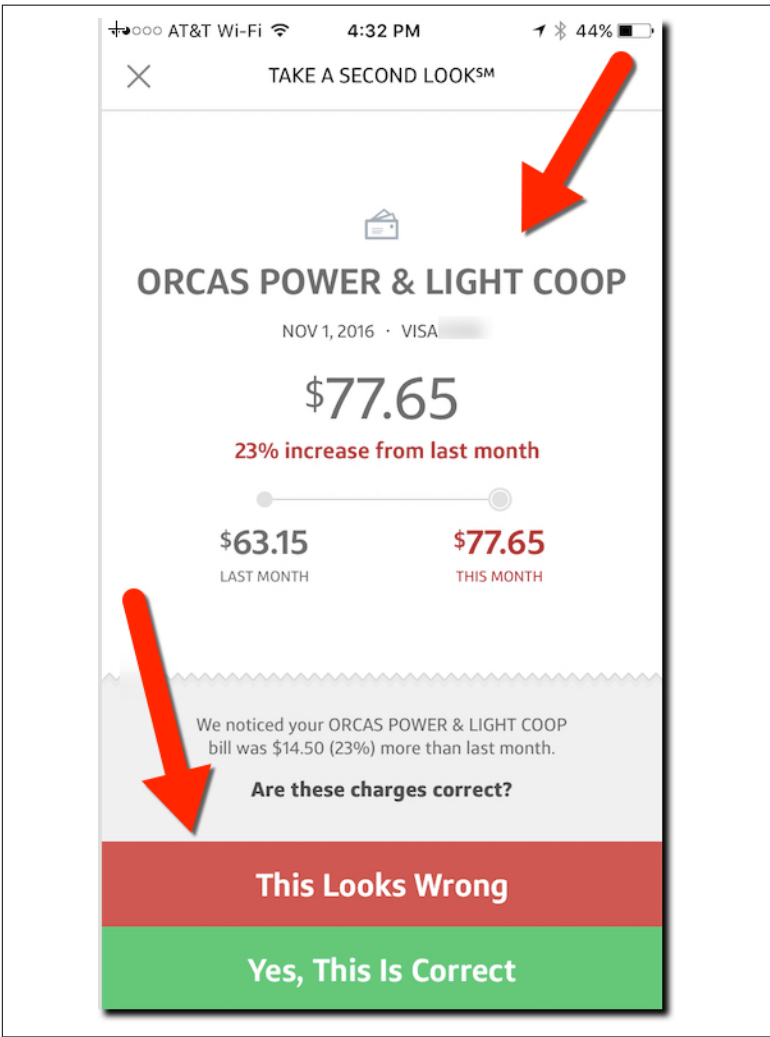
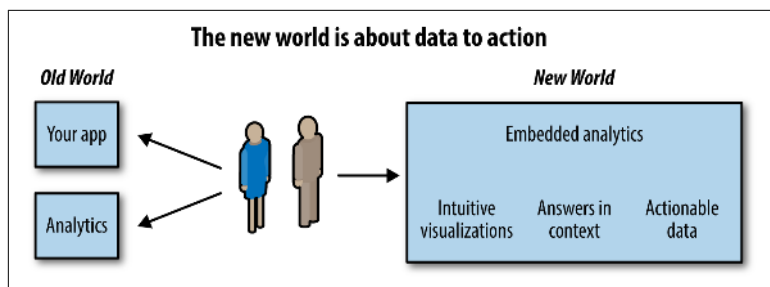


Figure 1-5. Second Look flags suspicious charges

This sort of UX requires design thinking, of course. But it also requires the product manager to do a bit of reverse engineering: What are the actual decisions that people want to make? What are the goals they are trying to achieve? Working one step back from that, how do you create a visualization that helps motivate and guide

your customers to make those decisions? And finally, what data must you capture to create those visualizations?

People have been talking about making the shift from data to insight for a long time. But it's really data to *action* that matters, as illustrated in [Figure 1-6](#).

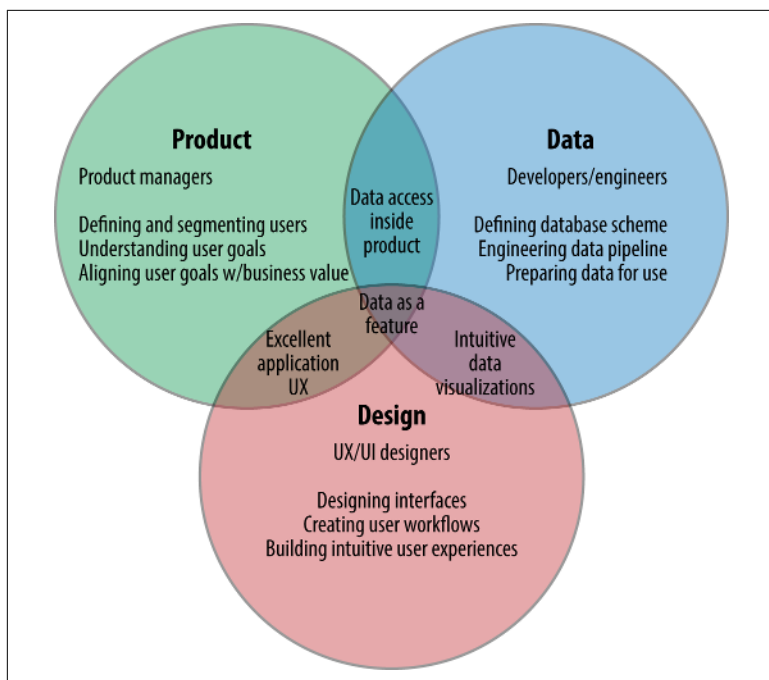


*Figure 1-6. Building actionable data as a feature into products means embedding analytics into applications*

How do product managers accomplish this? First, as we discuss in more detail in the next section, they need to develop a deep understanding of who they are building for and what the goals and motivations of those users might be. Then, they must use the product, data, and design expertise at their disposal to create actionable data-driven experiences in their applications, as shown in [Figure 1-7](#).

Remember that it's not enough to simply supply your users access to data in your product. Nor is it sufficient to just have a product that's technically robust or has beautiful data visualizations. It's the integration of all these things in the service of helping your users achieve a goal that makes data into a feature.

Most important, this type of treatment of data enables users to act. As we've mentioned, this is less about design-first thinking and more about goal-first thinking. Design is just a means to an end. You're looking for data to help users take action and make decisions.



*Figure 1-7. How different roles in a team converge around data as a feature*

## Understanding Your Customers' Goals Through Personas

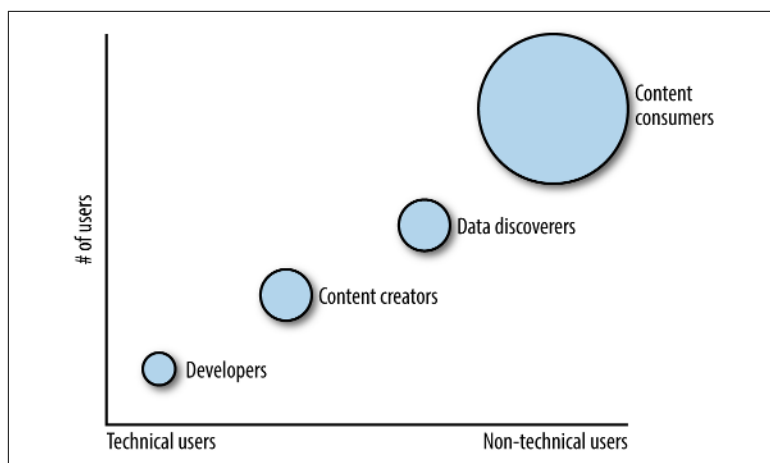
Personas are a very common tool used to figure out both the “who” and the “why.” They are detailed profiles of fictional characters, composited from real-world user research, that represent the users of your product. Product managers and designers depend heavily on personas to develop products of all kinds, and they can be particularly valuable when developing products that intend to treat data as a feature.

For example, personas could help Mint’s product managers ensure that both power users and more casual users are able to use the product to meet their goals. It’s clear that Mint caters to several types of users, each with specific goals and expectations of the product. By identifying and carefully profiling these users, the product team gets the direction needed to build features into the product that accommodate everyone.

To be useful, personas should be based on both quantitative and qualitative research to include such things as the goals, motivations, behaviors, and workflows of the user. And although personas should include some fictionalized personal details to bring your users to life, it is important to keep them firmly grounded in specific, well-understood, and prioritized needs and goals. When it proves difficult to understand and prioritize these needs and goals, it is often a sign that personas have not been sufficiently researched.

The chief benefit of using personas is to promote a user-centric approach to product development. All stakeholders can discuss whether a particular piece of data—and how it is presented graphically—meets the needs of “Tania the Technician,” for example. Personas force you to place names, characteristics, and even faces on users so you can keep them in mind when making critical development and design decisions.

Following are personas that represent some common types of data users. Note that this is just one way to create data-user personas. You can be much more detailed and personal based upon the targeted customers for the product you are building, as demonstrated in [Figure 1-8](#).



*Figure 1-8. Different types of personas and their relative size*

### *The content consumer*

This persona represents the majority of users today, both business and consumer. Content consumers are nontechnical, non-analyst users who want at-a-glance insights, delivered at the

right time, presented in the right context, in a manner with which they can interact. Most important—as we’ve previously discussed—the data being presented must be tied to specific actions that this user is already looking to take or goals that they are already looking to meet.

Users who fit into this category will usually consume prebuilt reports and dashboards, adjusting and filtering their data as needed. Note that nearly every persona will have at least some use for prebuilt reports and visual summaries; your job is to determine what exactly those needs are, and how they differ from persona to persona.

#### *The data discoverer*

These are people (often data analysts) who use dashboards to monitor performance, drill down to greater details, and discover hidden relationships within their data. Whereas the content consumer is looking for data to provide answers to finite and well-understood questions like “what are sales numbers for the past two quarters?”, the data discoverer finds value in more open-ended questions like “what are interesting patterns in sales activity over the last six months?”

Users who fit into this category are likely to dig deeper into reports and dashboards. But even the most open and curious data discoverer still has specific goals and needs, and these vary greatly across domains and industries, so it is critical that you take the time to discover these for your specific users.

#### *The content creator*

These people are often classified as power users. Content creators have the desire and ability to create their own visualizations, reports, and dashboards—either for their own consumption or the consumption of others. These users are generally more comfortable combining and synthesizing data from multiple sources, and generally seek a greater degree of customizability and portability in their data experiences.

Users who fit into this category can be critical advocates for your product because they are often creating reports and visualizations for the explicit purpose of sharing their creations with others. This means that you must understand not only the needs of these users, but also the needs of the users with whom they are sharing the reports and visualizations that they create.

## *Developers*

Perhaps the most important personas of all are developers. This group is responsible for writing complex queries and creating reports and dashboards that are consumed by many different types of users. Developers also play the key role of preparing data for use by content creators, which often involves defining a metadata layer. This metadata layer abstracts the complexity of the database away from the user, making it possible for nondeveloper users to query the data without knowing how to code.

Though the technical needs of developers will differ in fairly obvious ways from those of other personas, it is important to remember that developers still have nontechnical goals and needs. Any steps you can take to understand the needs of specific groups of developers beyond “technical access to data and basic documentation” will help differentiate your product in a crowded market.

## **Limitations of Personas**

For the product manager, user personas are useful tools, but they can be limiting in some ways. For starters, we run the risk of designing our personas around the things we want to build, rather than taking the opposite—and correct—approach. When creating data-rich products, it is tempting to define each persona by the data we think they might want, and to think that our job is done when we have provided that data. But we must be very cautious of defining a persona in terms of the data that a customer needs. Customers don’t need data; they need to make decisions. Data is simply a means for customers to make better decisions.

For example, if designing a fitness app, you would start with saying, “As a fitness enthusiast, I need to understand how my daily fitness activities are helping me achieve a healthier lifestyle.” Be wary of designing a persona who says, “I need data about my health.”

A phrase that is often repeated in product management circles is *outcomes over outputs*. So, when designing a persona, first consider what outcome that persona is seeking. Then you can look at the outputs. Again, outputs are not our end results, but part of a journey of decision making that our users are on.

Another potential pitfall of using personas is that we can let them become static. We need to be rigorously researching our customers



on an ongoing basis. Unless we do the difficult work of talking to our users, directly interacting with them, and understanding their needs, our personas represent nothing more than untested assumptions based on speculation.

Rob Pierry, CTO at experience-driven software design and development firm **projekt202**, suggests that a robust user research practice, focusing on discovery-level methodologies like ethnography, can help differentiate useful personas from “creative writing personas” that are based more on assumptions than actual research.

A robust user research practice can help turn the specific challenges facing users into powerful competitive differentiators. For example, when projekt202 **led a digital transformation initiative** for Southwest Airlines, it was able to make data a powerful feature for users by designing a data-driven experience to help those users meet their goals and by making this experience as accessible and actionable as possible. Starting with a clear and well-researched understanding of airline employees’ most pressing need—keeping flights running smoothly and on time—projekt202 made it as easy as possible for Southwest’s personnel to get an at-a-glance visual understanding of flight status: red = bad/delayed; green = good/on time, as illustrated in **Figure 1-9**.

Beyond that, they created a customized visual language to capture the specific information, such as pilot status, that is most important to airline employees and customers. Every step taken by projekt202 added to this highly functional, meticulously designed application that was tailored to help the users it spent so much time researching achieve their goals. As this example shows, understanding the goals and needs of your specific users and composite personas—such as airline employees—can and should guide the entire product development process from ideation to execution.

Arrival	Turn	Departure
08:15	A7	08:55
08:20	A23	09:00
08:25	A21	09:00
08:50	B20	09:10
09:00	A16	09:20
09:08	A25	09:30
09:10	A9	09:50
09:20	A4	10:00
09:25	A18	10:00
10:00	A29	10:00
10:13	B7	10:00
10:25	B20	11:10
10:40	A21	11:05
10:45	B20	11:10
11:30	B16	11:20
11:30	A25	12:00
11:45	A9	12:10
11:50	B4	12:20
12:00	A16	12:35

Figure 1-9. A customized data experience, led by projekt202, that Southwest Airlines employees use to monitor and manage flight statuses

## Start with “Why,” Not with “How”

Another important consideration for product managers is that *how* people want data is not the same as *why* people want the data. After you’ve ascertained *how* your users want their data, whether it is a raw feed or a fully designed dashboard, you might begin to feel like you have all the information you need to launch your data as a feature initiative. But it is even more critical for you to understand *why* somebody wants data in the first place. Because the “how” alone, whether it is a fully designed dashboard or a raw data feed, doesn’t give you enough to necessarily deliver value to your customers.

For example, the user of a particular data-driven product or dashboard might ask for access to a raw data feed or API in addition to a more designed experience. Simply providing this additional resource might seem like a quick and easy way to meet a specific need for your user. But in this case, the user has only provided information about *how* she would like to receive the data, not *why*.

In some cases, a request for a raw feed or API might be a signal that your app’s user experience needs some more work. Perhaps there is a particular way of interacting with data that your user needs but you

have not built. Or, perhaps your user has built her own centralized system for consuming and storing external data feeds. Simply knowing *how* your user wants to consume your data is not enough, and can leave room for critical missteps.

This is just as true for users requesting fully designed, low-customization data experiences. From a user's perspective, asking for “a single report that tells me everything I need to know” is easy. But taking the time to fully understand what exactly it is that this user needs to know—and *why* this user needs to know it—is challenging but critically important. A user's specific, transactional needs around data can change very quickly, and fully grasping their motivations is key to future-proofing your product. If you truly understand why the data you are providing is valuable to the specific users that you are trying to reach, the data then becomes your product's most potent asset.

One of the fundamental goals of product management is to align the “why” with the UX and the business needs of the vision. How do you make sure that the work that your design team is doing is aligned with the work that your development team is doing? How do you know it is aligned with the needs of your users? What it comes down to is that design needs to have a purpose. A robust and dimensional understanding of the “why” behind data-driven experiences can create a common language that unites every aspect of the “how” from design to technical implementation.

## Precisely Crafting the UX

After you understand the *why*, or the desired outcomes for your personas, it is time to think about the output, or the UX (the “how”). Again, the key to successfully creating a great UX is to put yourself in your user's shoes. To help you to begin thinking through how these experiences might differ from user to user, here are four general ways to present the output:

### *No visualization*

There are certain personas that might not need their data visualized at all. They simply need an API or another technical system that will allow the personas to get the data they need, when they need it. Keep in mind that things like documentation and naming conventions can make a big difference in terms of creating a

good or bad UX for APIs and technical systems—just because there is no visual layer does not mean that there is no UX.

### *Nonrepresentational data experiences*

In some cases, data is used to create visual experiences that look nothing like a dashboard or a report. For example, Amazon's lists of product recommendations—presented simply as photos with corresponding prices (Figure 1-10)—are powered by enormous amounts of product and user data. Both the visual experience and the underlying data are designed with the user's goal in mind: to discover and purchase products that will be of value. And although data is being used to create a visual experience, that visual experience does not directly represent the underlying data. Nonrepresentational data experiences can be a powerful way to enhance or optimize a user's experience, but can prove very frustrating for users who are looking to understand and/or directly manipulate the underlying data.

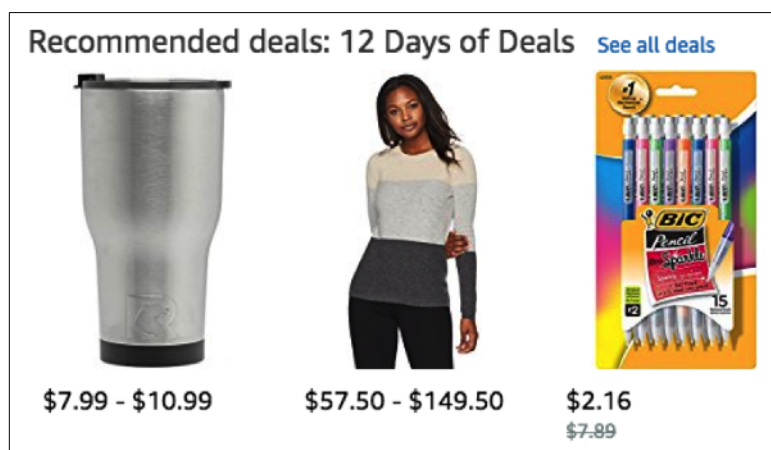


Figure 1-10. Amazon's recommended products: a nonrepresentational data experience

### *Canned reports and dashboards*

Most users currently use these types of outputs. In fact, the majority of the user population simply want a report that's built into the product. For example, Mint provides canned reports and dashboards that are built and presented in a way that is consumable out of the box for its audiences. Users can filter and interact with this data, based on built-in parameters, to find answers to the majority of their questions.

### *Data exploration*

Some people still have questions that canned reports and dashboards can't answer. Even if you've done all of the necessary user research and considered all of the possible scenarios, it's not practical to assume that your canned analytic content will cover every question from each user. For example, perhaps a user of your fitness application wants to see all of his bicycle rides over the past year with a distance of 20 miles or longer—which doesn't show up on the standard dashboard—or to manipulate the data in some other way. This type of output requires an element of self-service if those filters aren't built in to the initial report or dashboard.

If you are not sure which approach is best for a specific user or group of users, try creating a quick visual prototype or sketch that your users can interact with directly. You might discover, for example, that a user who initially described wanting an invisible “it just works” experience actually wants to engage with and manipulate data directly. Or, you might discover that a user who rattled off a million different custom feature and filter requests is primarily concerned with one particular visual element or interaction. You might even discover that a technical user who requested “just the raw data” is able to better articulate her specific needs and goals when presented with a visual mockup.

## **Asking the Right Questions**

How do product managers take a truly user-first approach to creating data-driven experiences? One way to do so is by asking these questions, in this order:

1. Who are my users?
2. What are my users' goals?
3. What data is needed to achieve those goals?
4. Where does the data need to come from?
5. How should it be collected?
6. How does the data need to be presented to them?

This approach requires expanding the notion of design to not just include visual design, but information design, as well. If you're truly putting the user at the center of your process, this an important

thing to keep in mind, because the easiest data to access and visualize is not always the data that is most important for helping your users achieve their goals.

Fitbit provides a really interesting example of this. People are trying to make decisions about exercising (their goals). Tracking how far their phones have moved on a certain day doesn't work because they might be in a car. So, Fitbit decided to gather data based on arm movements. That way, it avoided a potential disconnect between how the data was collected and its users' goals—even though it required designing a completely different product from a standard software-only smartphone app.

For product managers, this comes down to one thing: ask a lot of questions. Do not assume that the data immediately available to you is the data that your users will find most valuable. Be open to discovering new and unexpected solutions that might require an altogether different dataset, or an altogether different approach to visually representing data. Keeping the specific needs of your users, the data you provide, and the way you represent that data aligned with one another is critical for your product's success.

In fact, if you are working on a data-driven project, be as precise as possible when describing the specific information at hand, and why it is important. Avoid using the word “data” as a catch-all. If you're describing the number of steps a person has taken in the last week, say “steps.” If you're describing the balance in somebody's primary checking account, say “balance.” Don't say “data about fitness” or “data about banking.” Because when it actually comes time for you to help your user solve specific problems or meet specific goals, that disconnect, that ambiguity, can be really dangerous.

Another related piece of advice: whenever you're making a decision or doing anything driven by data, use a template to document all the assumptions you're making.

Say you're going to use banking balance and activity records to help customers make decisions about where to spend their money. What are some of the assumptions you're making? You're making the assumption that most of their transactions are present in their bank account. You're making the assumption that their banking data is up to date. You're making a lot of different assumptions.

Assumptions are unavoidable if we want to make a decision, even if that assumption is simply that our users and our market will be the same when our product launches as it is when our product is designed. Rather than pretending those assumptions don't exist, document them. Because as you get a little bit farther downstream, it might turn out that some of those assumptions are really important and might affect who you're building for and how you're building for them.

If you realize that most people are purchasing things using cash rather than credit cards, that changes everything. You have to ask: who this is going to work for? Who's going to be more or less successful in using your particular product to solve their particular problem?

That step of documenting assumptions is a powerful one that all product managers should undertake in their work, but especially when they're working with data. After you've documented an assumption, you can discuss it—and test it—with a broader group. This provides a critical chance to have a discussion with your developers and designers and actual users to make sure your assumptions are sound.

## Make Your Data “Over the Counter”

Data is often very high stakes. Decisions—sometimes business- and life-altering ones—are based on it. Because of this, it is essential that the meaning of data can be grasped easily and immediately, without errors, and without getting a PhD in advanced mathematics.

This is where over-the-counter data (OTCD) comes in. This research-based design approach helps users accurately and easily interpret data. Dr. Jenny Grant Rankin—the expert who created the **OTCD Standards**—synthesized the research literature on the subject to include best practices for data visualization.

OTCD involves embedding help for interpreting data directly in the visualization using five components, each of which has its own set of **standards**:

### *Label*

Just like over-the-counter medicine, data needs to be properly labeled to ensure that it is used easily and appropriately.



### *Supplemental documentation*

Not all information a user needs to know can fit on the label, so supplemental documentation offers further explanation for the analysis and use of data.

### *Help system*

An online help system, accessible via live links from the data, contains explanations and instructions to help users perform tasks and understand the data and analyses.

### *Package/display*

The manner in which data is packaged and displayed for users must promote easy and accurate understanding, analysis, and use of the data.

### *Content*

Just as over-the-counter medications must contain effective and unexpired ingredients to function properly, the data chosen and delivered to application users must be effective and timely.

All too often, companies simply throw data to users without these OTCD precautions in place to make sure that it's used correctly and easily. Indeed, Rankin's research uncovered data being misused across all major industries. She also found that people's own perceptions of their proficiency with data had no bearing on their ability to accurately interpret it.

In other words, those who thought they were data experts were just as likely to misunderstand the data as someone who thought, "Well, I don't know, I don't think I'm so good at it." There was no correlation.

Usability testing is critical, according to Rankin. Do lots of user testing, she advises. Be open to hearing things you don't want to hear. Reinforce the "why" and the users' goals every step of the way.

## **Managing Your Data Roadmap and Feature Requests**

Requests for new features come from all over. You have stakeholders from the business, engineers who are anxious to implement innovative new features, and designers who want to improve the UX. And then, of course, there are the users of your product.

Treating data as a feature means that you can—and should—subject all requests to incorporate data into your product to the same rigorous prioritization process as any other feature you are looking to build. Again, you cannot assume that data has intrinsic value—you must be able to make the case for why this data is essential for your users, will add value to your product, and is worth pushing aside other requests to build.

First, it's important to translate all requests into users' goals. Even users won't be able to tell you exactly what they want. It's your job to translate their requests—their “hows” or desires for certain outputs—into “whys” and outcomes (goals).

It can be tricky, however, because in a lot of cases, requests for more data and data features are, indeed, indicative of unmet needs. But it's very rare that customers will come to you and say, “I am having trouble making decisions about my personal finances.” They'll say, “I wish I could see this data” or, “I really need more of that data.” *A big aspect of your job as a product manager is to level up from feature requests to goals and needs.* And it can be difficult to do that when you're under pressure to put something on the roadmap.

This isn't the fault of the people making these requests. Making feature requests is a sign that somebody—whether an internal stakeholder or an external user—is invested enough in your product to care about its future direction. Figuring out the “why” behind that feedback is your job, not theirs. Getting outside that tactical and transactional space of, “I want this,” “OK, here it is” to, “Why do you want this? What is the current product not helping you do?” can be a painful step, but you must take it.

In some cases, what's being asked for isn't actually what people need. People might request a new report, when you simply need to change one word on a report you're already giving them. Heavyweight feature requests might come down to a simple product copy change. Still, you won't discover this unless you take the time to talk at length with your customers.

Even when a feature is added to your roadmap, the work of understanding its user- and business-facing impact is not finished. Short-term prioritization is much more difficult than long-term roadmapping because prioritization is when you actually must allocate resources. Prioritization is when you say, “Our team is working on this data feature and not this other data feature.” Even though

saying, “Sure, we’ll add that to the roadmap” doesn’t cost anything, prioritizing one feature over another is a zero-sum game, and in turn requires a laser-focused understanding of your users’ goals, your business goals, and how the two are aligned.

The trick of prioritization is that the work doesn’t actually go into prioritizing; it goes into setting your goals. And if you’ve done that well with your stakeholders, prioritization actually happens pretty easily. And all of the downstream decisions you must make to deliver a feature to your users—build versus buy, degree of technical scalability, and other tactical distinctions—can be guided by the goals you have set, not the whims of your stakeholders.

Product managers should periodically sit down with senior leaders in the organization to lay out the product requests and ask them to write out the business goals and vision clearly enough so that it’s easy to prioritize. And these conversations should be animated by a deep understanding of user needs and goals. Actually test your organizational goals against your organizational roadmap or backlog. Because, ideally, the goals that are set by senior leaders in concert with product managers should be so clear and so actionable that you’re prioritizing the same way every single time.

Ideally, your company’s goals and vision should reflect both your customers’ needs and what your business wants to accomplish. The best goals at an organizational level are aligned with the needs of your users, not at odds with them.

This can be very difficult to achieve, and in some cases, it doesn’t happen. You can get into a tug of war with the business. Or you’ll wind up with compromises and a product that doesn’t please anyone. As a product manager, you must keep your user’s needs and goals at the center of everything you do. If you don’t, your product ultimately cannot drive growth and success for your business, even if it pleases internal stakeholders in the short term.

## Conclusion

There’s too much data today. There are too many tools to slice and dice it. And it’s not very clear to users how all of this data and these tools relate to each other, and which things are the most important to track.

The job description of a product manager has changed significantly. Simply providing data to users is no longer enough. Instead, by understanding and supporting your users' needs to act and achieve desired goals, you provide actionable data as a feature.

The big shift in how product managers need to approach data is not assuming that data has intrinsic value, but rather that products need to clearly communicate the value of the data they're presenting, and—most important—make it easier for people to act on that data.

There are millions of ways you can visualize data, but unless you make it easy, intuitive, and convenient for users to consume, they're either not going to consume it, or—worse—they're going to interpret it incorrectly and act accordingly.

As in so many other aspects of innovation, consumer technology is leading the way. The best consumer applications are now differentiating themselves based on how much value they create for users with data. Many businesses are thus “consumerizing” data so that customers get easily absorbed, contextualized data presented in a way that helps them to take action and make better decisions. In this next generation of business applications, software builders are making data their products' most valuable asset.

## About the Authors

---

**Alice LaPlante** has more than 25 years experience as an award-winning journalist, corporate editorial consultant, writing coach, and university-level writing instructor. She has written for *Forbes*, *ASAP*, *Bloomberg BusinessWeek*, *Computerworld*, *InformationWeek*, *Discover*, and a host of other national publications. Her corporate clients include IBM, Microsoft, Oracle, Symantec, Deloitte, and HP. She is also an award-winning fiction writer.

**Matt LeMay** is a product management coach and consultant. He has helped build and scale product management practices at companies ranging from early-stage startups to Fortune 500 enterprises. Matt was selected as a Top 50 Product Management influencer by the PM Year in Review for both 2015 and 2016.

Matt is cofounder and partner at Sudden Compass, a consultancy that helps organizations take a cross-functional and customer-centric approach to working with data. In his work as a technology communicator, Matt has developed and led digital transformation and data strategy workshops for companies like GE, American Express, Pfizer, McCann, and Johnson & Johnson.

Previously, Matt worked as senior product manager at music startup Songza (acquired by Google), and head of consumer product at Bitly. Matt is also a musician, recording engineer, and the author of a book about singer-songwriter Elliott Smith. He lives in Brooklyn, NY, with his wife Joan and their turtle Sheldon. You can find more of his work online at [mattlemay.com](http://mattlemay.com).