

How To Talk Technical

Without Talking Technical

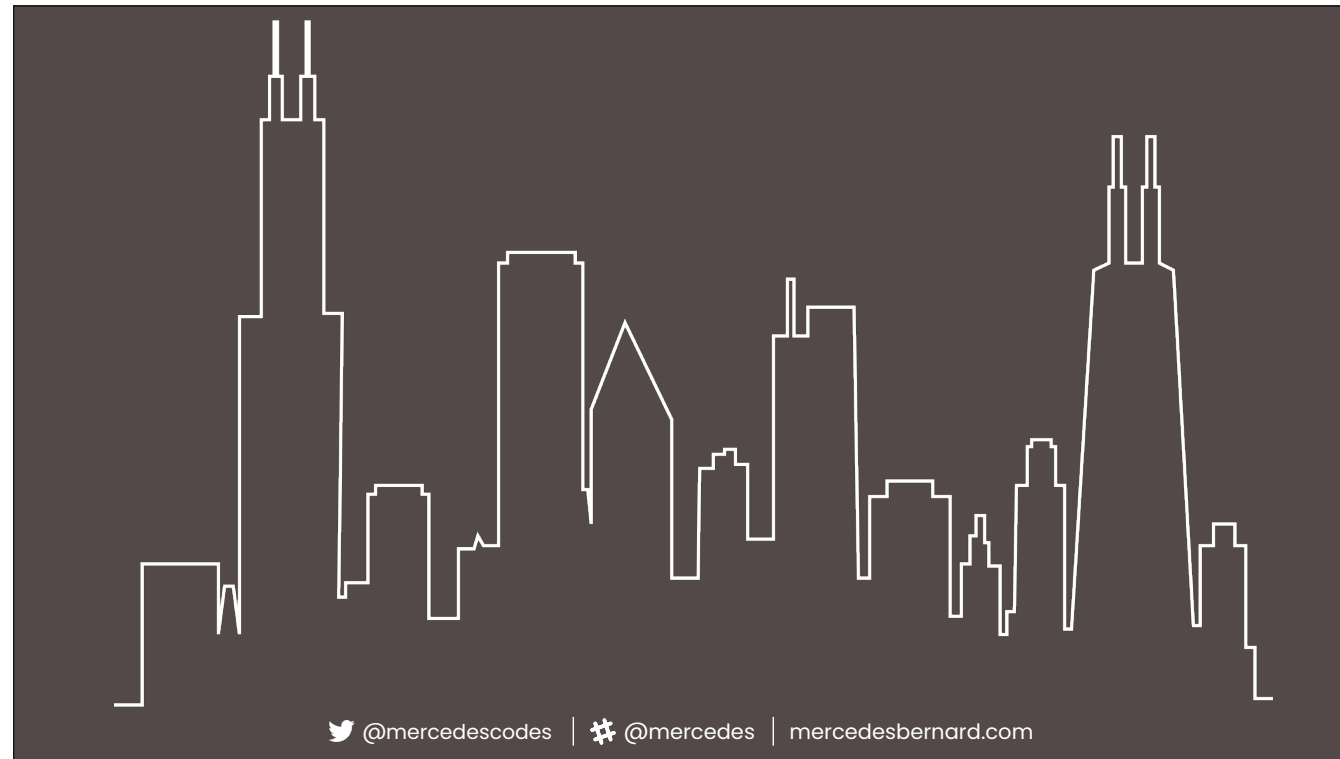
 @mercedescodes |  @mercedes | mercedesbernard.com

We've all been there.

We need to persuade our manager to give us time to build a proof of concept for why we should migrate our front end from Angular to React.

Or we need to explain to our client why we want to break apart their monolithic application and use microservices for their real-time notifications.

Or we just graduated from school and were never taught how to explain these concepts and someone just asked us what an API is.



My name is Mercedes Bernard and I've been a consultant for 7-ish years. I enjoy variety and working with clients to help them understand and get excited about what we're building.

Goals

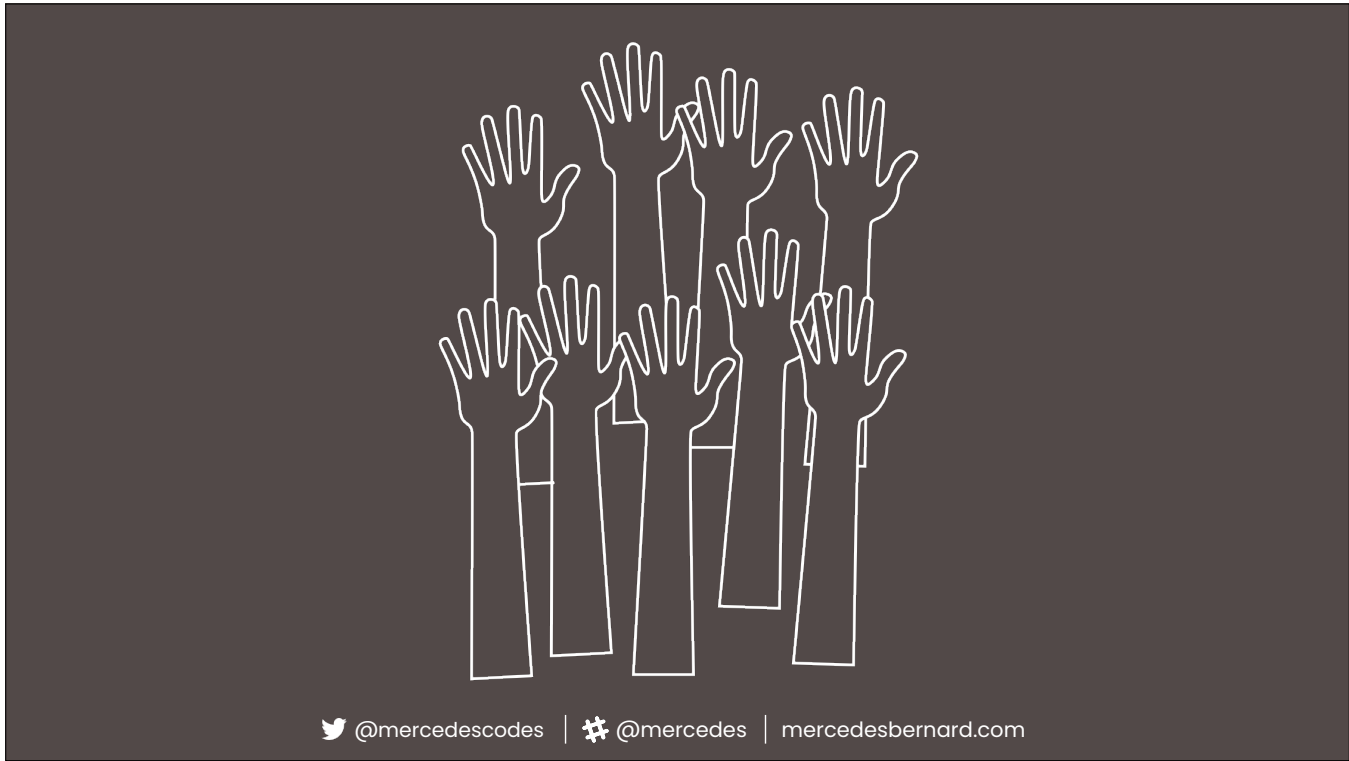
- Provide techniques for conveying complex technical concepts to non-technical audiences **in a way they understand**

Goals

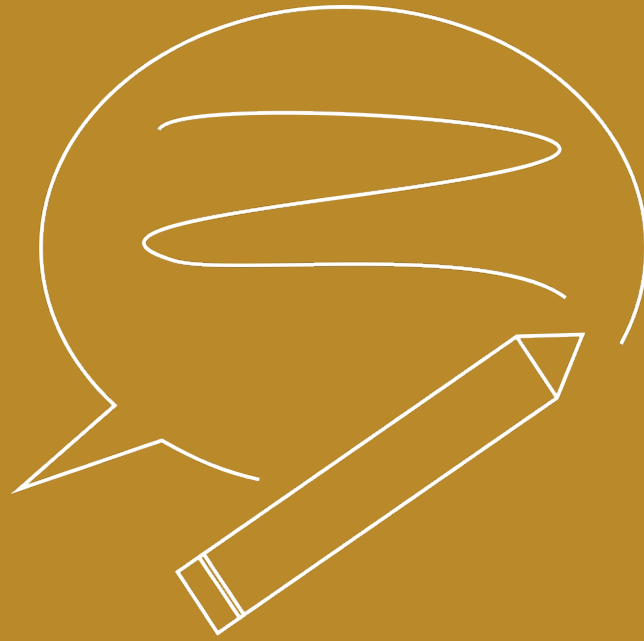
- Practice these techniques

Goals

- Increase confidence in technical communication



Consultant? Product or internal development?
Individual contributor? Manager?



 @mercedescodes |  @mercedes | mercedesbernard.com

1 min

Write down an idea you recently
advocated for

 @mercedescodes |  @mercedes | mercedesbernard.com

1 min

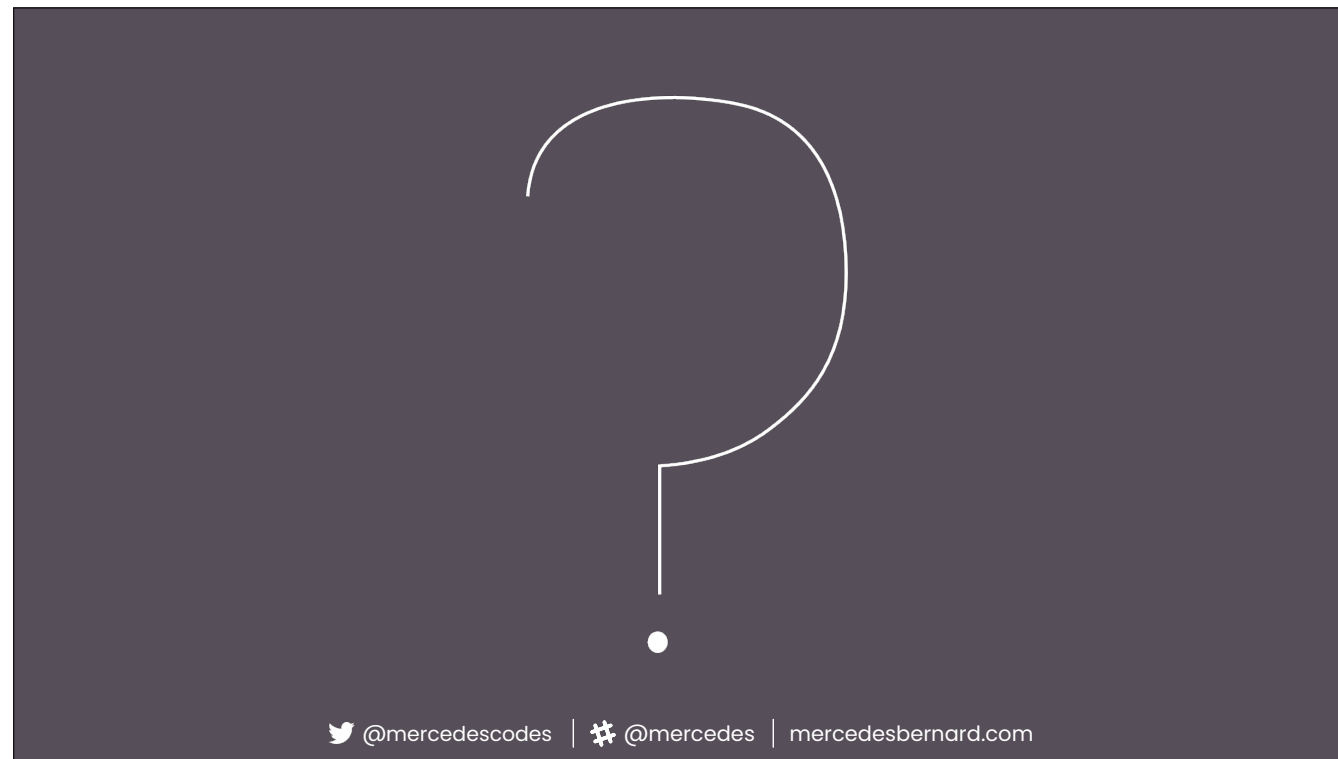
Write down something technical that
you recently learned

 @mercedescodes |  @mercedes | mercedesbernard.com

1 min

Write down the most complex technical
solution that you recently worked on

 @mercedescodes |  @mercedes | mercedesbernard.com



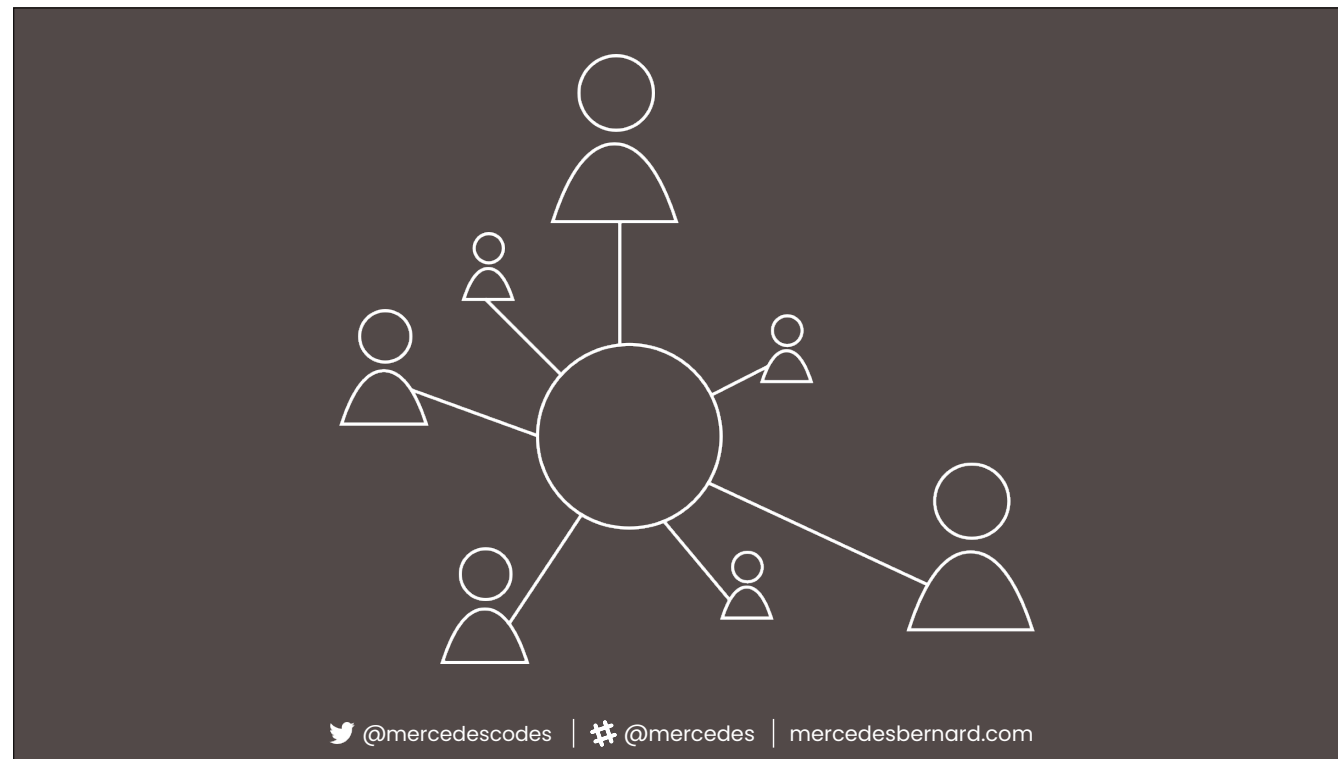
Why is the skill to translate technical concepts into understandable language so important?

If you do it well you can build credibility with your audience and create a shared understanding. Then your audience will be more likely to accept your ideas, invest in your ideas, and ultimately adopt your ideas.

Know Your Audience

 @mercedescodes |  @mercedes | mercedesbernard.com

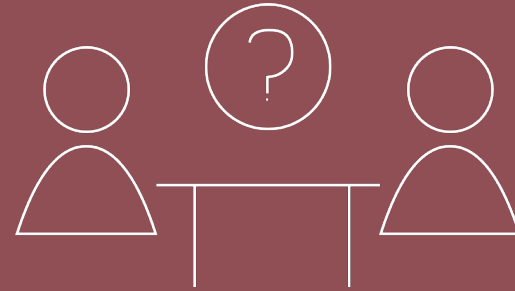
The first step to building this credibility is to know your audience.



I'm using non-technical audience like a catch-all but there are many different types of audiences.

Including different seniorities like C-level executives, middle management, and entry-level folks; different departments like marketing, finance, legal, and HR; internal vs external audiences; and even clients across different industry domains.

Ask Questions



 @mercedescodes |  @mercedes | mercedesbernard.com

I once had a large hospitality holding company as a client. They owned restaurants, hotels, casinos, amusements parks, and more. We were redoing the websites for a bunch of their restaurants and my point of contact had been with the company for 15-20 years. He led their website team, including the content producers and the technical team that implemented it. He was really knowledgeable about their domain and all of the functional requirements as he had led their last implementation. We were having a conversation one day about some publishing requirements, and I was using the word “nullable” while talking about some implementation details. He was beginning to get very frustrated with me and it became clear that he didn’t understand me. Because he was so knowledgeable about the websites and the technical team he was leading, I made assumptions about his technical background. At that moment, I damaged some of the trust we had established by confusing him and making assumptions that he was technical.

If I would’ve taken the time to ask questions, I could’ve avoided that uncomfortable conversation and his frustration. Asking questions seems so basic, but we often skip this step and jump right into preparing what we want to say.

Who will be in the
audience?

 @mercedescodes |  @mercedes | mercedesbernard.com

Here are some of the questions I like to ask, whether it's before a presentation, a meeting, or a 1:1 conversation.

Who will be in the audience?

What are their names? What are their roles? How many people will be there? How do they work together?

Details about the individuals in the meeting are really important. The more you know about the individuals, the more you can personalize the conversation.

Who will be in the
audience?

What departments
will be represented?

 @mercedescodes |  @mercedes | mercedesbernard.com

What departments will be represented?

Marketing? Finance? Internal IT? HR?

Each department's concerns are going to be different. Imagine the things that marketing cares about and prioritizes compared with the legal department. They are probably very different. It helps to know the representation of your audience, and it can also help to know if it's going to be a mixed audience so you can be aware of how you present each concern and its priority.

Who will be in the audience?

How long have you been with [company]?

What departments will be represented?

 @mercedescodes |  @mercedes | mercedesbernard.com

How long have they been with the company?

Has the person you're talking to only worked there for just a few months? If so, they'll be a great resource for you to understand the current workflow pain points and problems to solve since they've learned it the most recently.

Or is the person part of the original team that started with the company? They'll be able to provide a lot of context about why things are the way they are and how decisions were made.

In both cases, you can learn from who you're talking to and what you learn can help strengthen your argument. It can also help inform how you position your idea when you present it.

Who will be in the audience?

How long have you been with [company]?

What is your background with [tech]?

What departments will be represented?

 @mercedescodes |  @mercedes | mercedesbernard.com

What is their background with [tech]?

If you're building them a web application, it can be helpful to know what their background is with basic web technology. How familiar are they with modern web practices? Do they use social media? Do they shop on Amazon?

When they are using the web, do they use a desktop computer? Do they mostly use their phone?

And if you're building a piece of software that they're going to interact with, such as a CMS solution, do they have experience writing content for the web? Have they ever used a WYSIWYG? Do they use an email client? Do they have familiarity with HTML?

Who will be in the audience?

How long have you been with [company]?

What is your background with [tech]?

What departments will be represented?

How long have you been working with [tech]?

 @mercedescodes |  @mercedes | mercedesbernard.com

How long have they been working with [tech]?

If they have a background in technology related to the software you're building them, how much experience do they have with it? Have they worked with Wordpress before? When? Was it when Wordpress was just becoming popular in the early 2000s? Or have they used it in the last 6 months?

All of this information will help you contextualize what you need to communicate with your audience.

Listen With Generosity

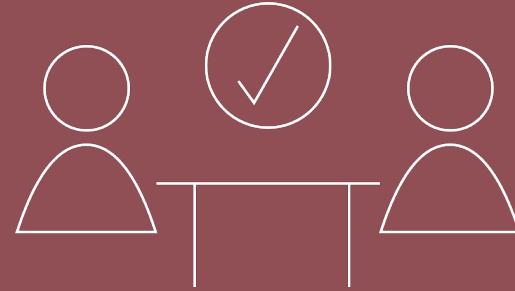
 @mercedescodes |  @mercedes | mercedesbernard.com

Take active listening a step further. It's important when you're asking these questions and gathering information to make sure that you are listening to understand. Don't make assumptions. Be open to all of the information they're sharing with you. Be willing to be surprised and to learn something. Be patient. Be curious. Be generous.

And use what you've learned to inform how you communicate later.

When someone says they don't know something, it's not because they're dumb. It's because it's never been explained before.

Understand Their Concerns



[@mercedescodes](#) | [@mercedes](#) | [mercedesbernard.com](#)

Next, work to understand their concerns. Developers love to talk about the latest and greatest tech. We get excited because it's new and shiny. Or we talk about performance and extensibility. But what does our audience actually care about?

The most common concerns I've experienced from stakeholders include the following.

Common Concerns

Understanding

 @mercedescodes |  @mercedes | mercedesbernard.com

In a majority of cases, when someone is concerned and asking a lot of questions, they just want to understand what it is that you're talking about. No one likes to feel left in the dark. If the person you're talking to will need to make decisions or relay information they've learned from you, it's often vital that they understand it so they can go into their conversations informed and equipped to talk about your solutions.

Common Concerns

Budget

 @mercedescodes |  @mercedes | mercedesbernard.com

For people who make budget decisions or who write the checks, there are a lot of questions about cost. Why does adding that feature cost so much? How can we finish this project faster or for less money? Is there a way to gradually implement this so that the entire cost doesn't land in this fiscal year?

Common Concerns

Changing business process

 @mercedescodes |  @mercedes | mercedesbernard.com

If the person you're talking to isn't making approval decisions and isn't signing the checks, they are often concerned with how your solutions will affect their day to day workflows. Whenever there are proposed changes, the transition can often be tedious and uncomfortable so there is understandable resistance. If you are proposing some kind of new data storage, who is responsible for migrating that data? Will it fall on someone you're talking to manually input the data? (an all too common cost-cutting measure I see)

Common Concerns

Added value

 @mercedescodes |  @mercedes | mercedesbernard.com

And finally, related to the changing business process is the concern about added value. It's normal to wonder what this new solution solves that the current working solution doesn't. For all of the investment in time, money, and effort that your stakeholders are making, what added value are they going to see? Will they be able to work faster? Will they have increased cross-department transparency? The value doesn't always need to be dollars and cents. Improved efficiency, communication, and culture are all valuable.

Address the Actual Concerns

 @mercedescodes |  @mercedes | mercedesbernard.com

Now it may seem obvious, but when we're communicating our ideas to them, make sure to address their ****actual concerns**** instead of trying to sell them on why the tech is so great.

Common Concerns

- Understanding → Teach the concept
- Budget → Address cost and potential savings
- Changing business process → Show time savings, increased transparency or other process improvements
- Added value →

 @mercedescodes |  @mercedes | mercedesbernard.com

If the person you're talking to is mainly concerned with understanding what you're explaining, take the time to teach them the concept. We'll talk about some strategies for this later in this post, but relate the concept to things they understand in their daily lives.

If their main concern is budget, address the cost and potential savings. Acknowledge the upfront cost, but be prepared with ROI and timelines for the investment to make itself worth it.

And if they are concerned about their changing workflow or wanting to know what the added value is, talk about how your solution will save them time, shorten feedback loops, increase communication transparency, more informed decision making, or whatever other process improvements you are proposing implementing.

5 min

Using the 1st item you wrote down at the start of this talk, turn to someone near you and discuss the stakeholder's underlying concern and what you did to address it. If you didn't, how could you have?

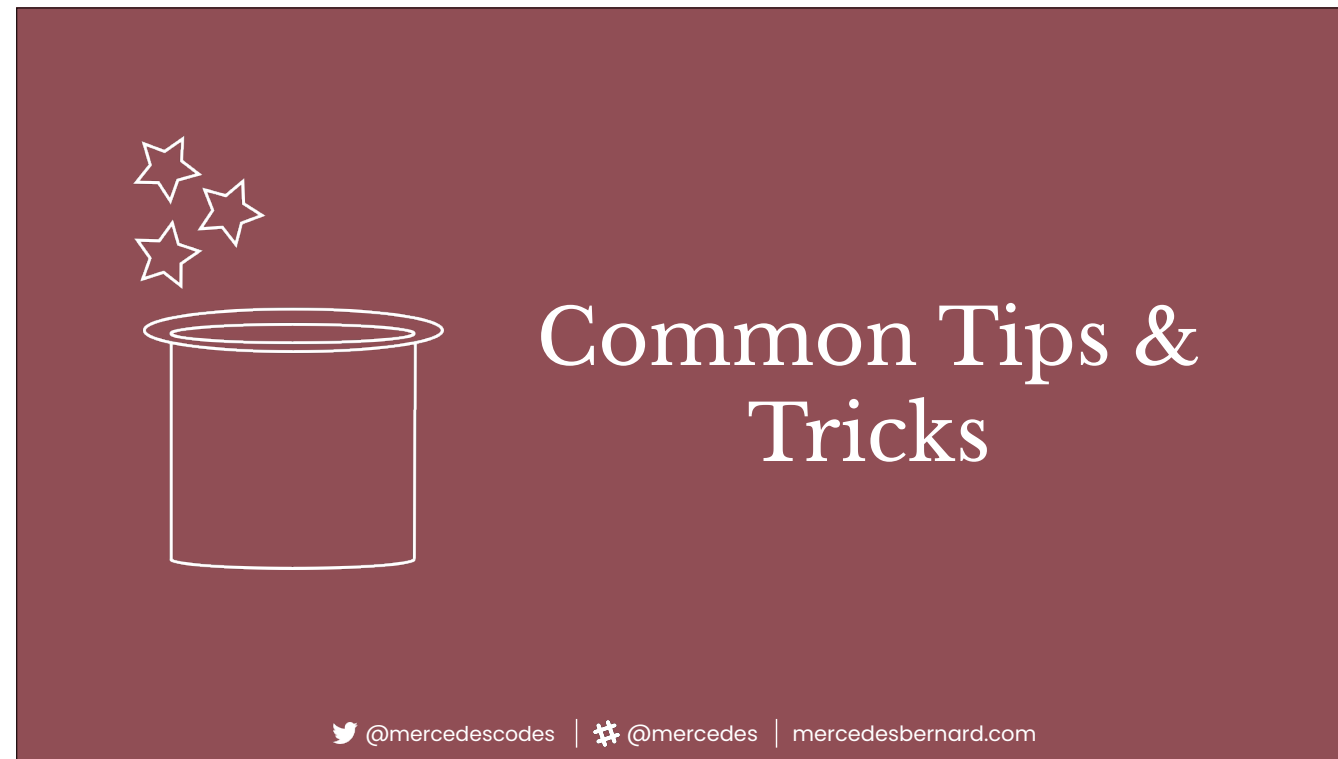
 @mercedescodes |  @mercedes | mercedesbernard.com

So How Do I Actually Explain the Thing?

Body Level One

 @mercedescodes |  @mercedes | mercedesbernard.com

We've talked about how to get to know your audience and how important it is to understand their cares and concerns, but how do you actually explain your solution?



I'm going to start with some common tips and tricks when explaining technical concepts to a non-technical audience.

No Jargon

- Avoid acronyms and other technical language except where necessary
- Always define technical language prior to using it
- Saying what the acronym stands for does not count as defining it

 @mercedescodes |  @mercedes | mercedesbernard.com

The first tip is widely known and accepted, but warrants repeating. Don't use technical jargon. Avoid acronyms and any other technical vocabulary except where absolutely necessary. When you do need to use technical vocabulary, make sure you clearly define it in everyday language before you use it. And make sure everyone understands the term before you move on. If you need to use an acronym, saying what it stands for does not count as defining it.

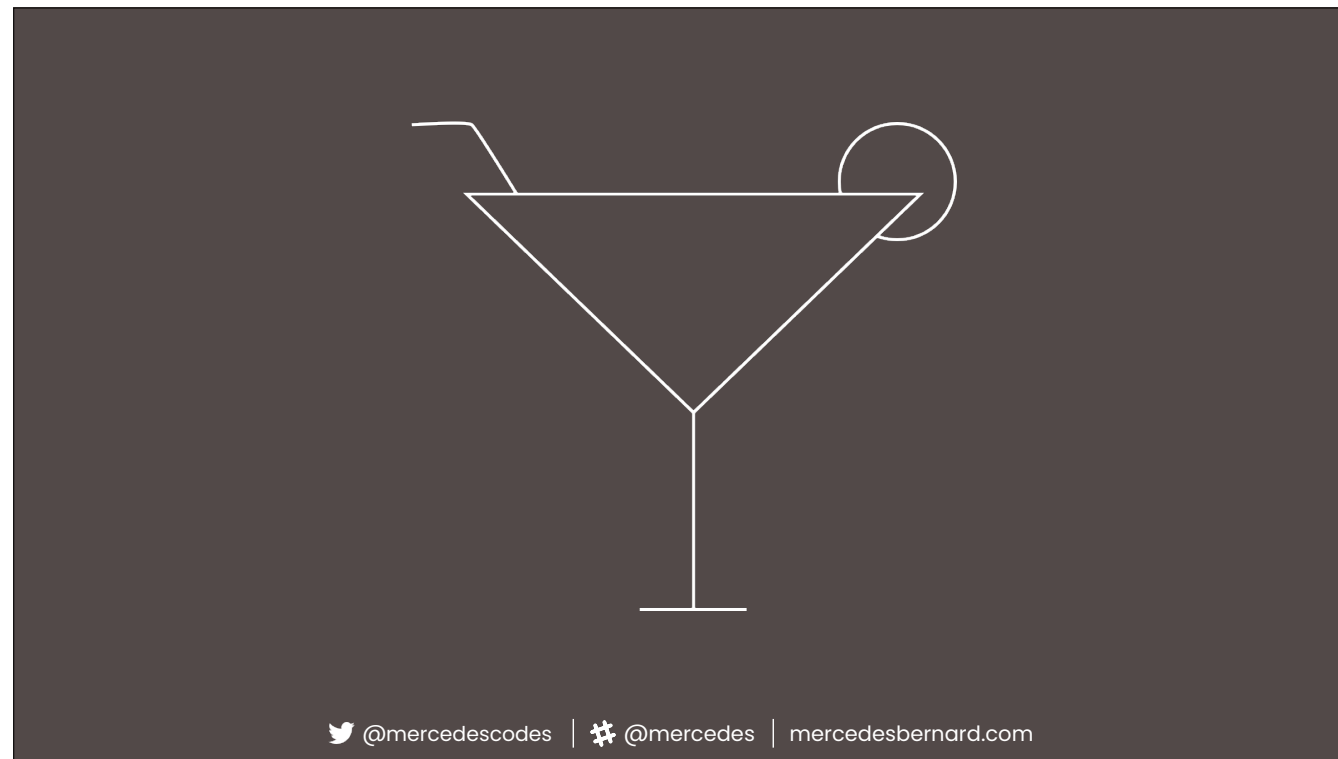
Consider the acronym "API." Your audience is not going to understand what you mean if you say, "I need to integrate with this third-party API. That's an application programming interface." Instead, you should say, "I need to integrate with this third-party API. That means that I need to write code to send and receive information in a way the third party understands."

Metaphors

- Make them accessible
- Every day situations: mailing a letter, starting your car, ordering a drink

 @mercedescodes |  #mercedes | mercedesbernard.com

Another really common tip is to use metaphors. And metaphors are an extraordinary tool for your technical communication toolbox, but people often don't realize how challenging it can be to come up with a really useful metaphor. The key to a metaphor helping your audience understand something is that it's accessible. Regardless of your audience's background, experience, culture, where they grew up, or what domain they work in, they should have familiarity with the subject of your metaphor. Everyday situations like mailing a letter, starting your car, or ordering a drink are all examples of accessible metaphors.



One of my coworkers recently gave a technical presentation to our entire company which includes both our design team and our sales and growth team. In order for everyone to understand his presentation, he needed to make sure everyone understood the architectural pattern MVC. He used the metaphor of ordering a drink at a bar where the customer is the user.

The customer walks up to the bar and places their order, or `_makes a request_`, in a language that the bartender understands. The bartender, or `_controller_`, receives that request and uses the ingredients and tools behind the bar, the `_models_`, to create the drink. Notice that the customer never gets behind the bar and makes their own drink, or the `_view_`. In MVC, the user should never interact directly with models and instead, it is the controller's responsibility to interact with the models to compose the view. Then the bartender hands the drink to the customer, which is to say the controller returns the view back to the browser to be presented to the user.

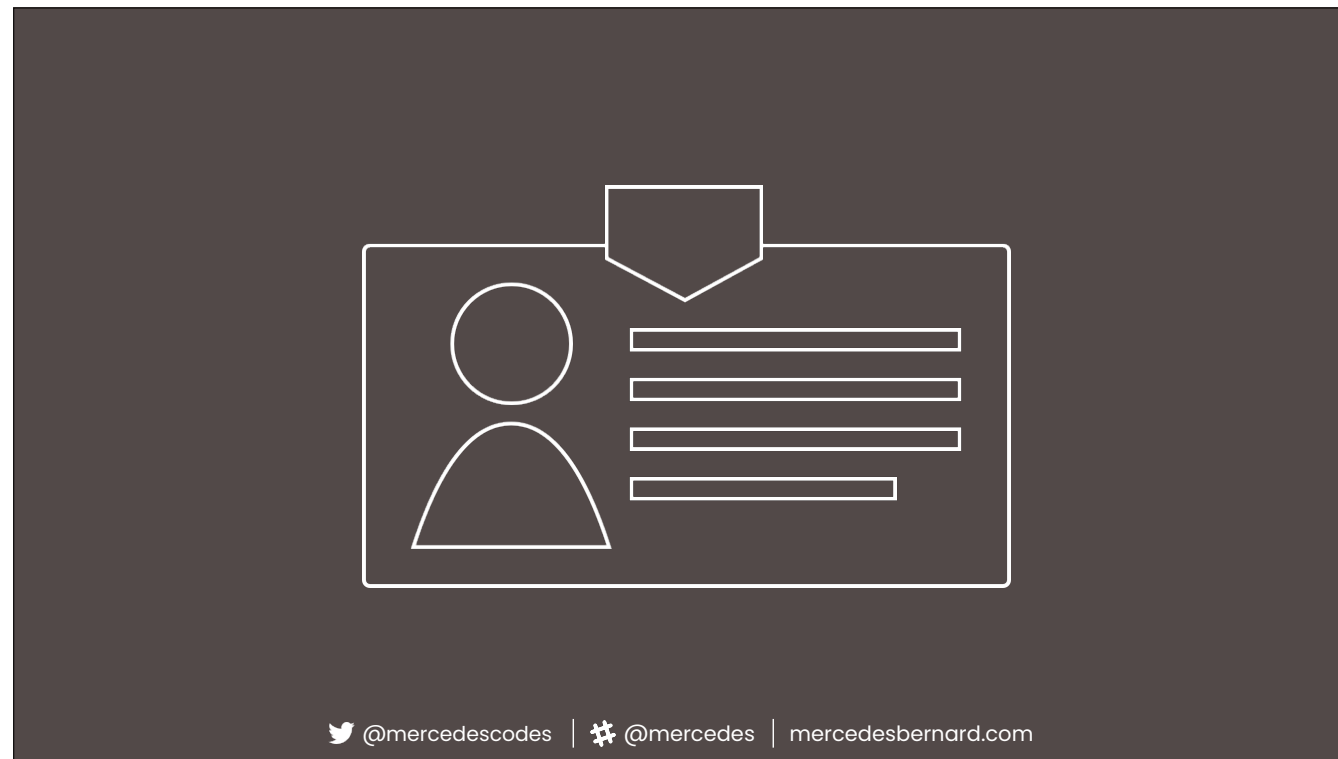
This metaphor is a great example of using a situation that everyone is familiar with to explain an abstract technical concept in enough detail that the audience understands all the key points without getting lost in technical language.

Metaphors

- Use analogies to your audience's domain

 @mercedescodes |  @mercedes | mercedesbernard.com

Another great way to create a metaphor is to use analogies specific to your audience's domain. This can be especially useful if your audience all works at the same company or all have similar job functions.



My current client is a government agency. I needed to explain various types of access control to a group of their employees and I was able to use security clearance as a metaphor explain mandatory access control.

Mandatory access control is where you give a resource (such as data) a security level and each user has a security clearance. In order to be granted access to that resource, the user must have a security clearance at or greater than the resource's security level. Because this is how both security clearances and mandatory access control work, my audience was able to understand the concept quickly and move on to more learning nuanced forms of access control.

Metaphors

- Stay away from sports

 @mercedescodes |  @mercedes | mercedesbernard.com

The last tip I like to give when using metaphors is to stay away from sports metaphors. Not everyone knows enough about sports to really understand the metaphor or for the metaphor to be useful in illustrating a point. Additionally, sports language tends to be competitive and individualistic which are not the values you want to emphasize when collaborating with your client and helping them learn something new.

Don't Be Afraid To Draw

Body Level One

 @mercedescodes |  @mercedes | mercedesbernard.com

My last trick is don't be afraid to draw. Visual language leaves less open to interpretation. It doesn't rely on using the right words to convey meaning. One of my favorite things to do after I've drawn a picture to explain something is to hand the marker to the person I'm talking to when they have questions so they can visually show me what they're confused about. This helps because most of the time they don't have the specific vocabulary to explain their confusion. They may just circle part of the drawing or they may add to it to clarify one part in the information flow, but either way, it's enlightening for me to see where they might be getting tripped up so I know where and how to elaborate.

10 min

Using the 2nd item you wrote down at the start of this talk, turn to someone near you (different person) and use a metaphor to describe the concept you learned so they will understand it

Tips & Tricks Are Great and all... Body Level One

 @mercedescodes |  @mercedes | mercedesbernard.com

But What About the Meat and Potatoes?

Body Level One

 @mercedescodes |  @mercedes | mercedesbernard.com

Explain Just Enough



 @mercedescodes |  @mercedes | mercedesbernard.com

When you're explaining something technical, make sure you explain just enough to highlight the key points. You don't want to go too deep into the details that you overwhelm your audience or lose them along the way.

Don't Go Too Deep



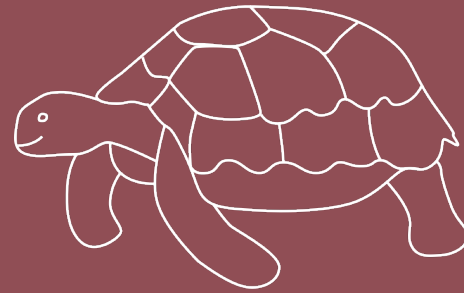
- More often than not, you need to explain the 'why,' not the 'how'



 @mercedescodes |  @mercedes | mercedesbernard.com

For example, if you need to explain the difference between SOAP and REST, don't spend your time explaining the details of messaging protocols and architectural styles. Your audience most likely just needs to understand that SOAP can be more secure but requires a higher level of effort to implement than REST which is lightweight and flexible. SOAP is like reading a letter that has been sealed in an envelope where REST is like reading a postcard.

Slow Down



[@mercedescodes](#) | [@mercedes](#) | mercedesbernard.com

Make sure that you slow down while you're explaining unfamiliar technical concepts to your audience.

Take Your Time

- Take more time than you think you need
- Use the time wisely

 @mercedescodes |  @mercedes | mercedesbernard.com

It's easy to forget what it feels like not to understand the topic and to move quickly through your explanation. Think of it like public speaking, you want to go slower than feels natural. Take more time than you think you need. But make sure that you're using this time wisely.

Mix It Up



 @mercedescodes |  @mercedes | mercedesbernard.com

And by that I mean, use multiple different strategies to teach your concept rather than repeating yourself over and over again

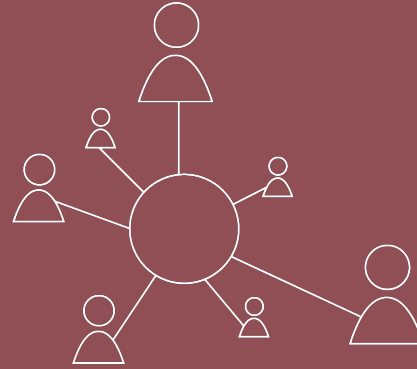
2 or More Descriptions

- Try to come up with 2 different ways to explain your concept
- A verbal description and a visual description are good go-tos

 @mercedescodes |  @mercedes | mercedesbernard.com

If a plain English verbal description isn't helping your audience, draw a picture instead. Whenever I'm explaining something new, I try to use two or more different strategies to explain the concept. It can be plain English and a picture, a picture and a metaphor, or whatever works for you and your audience.

Remember Your Audience



 @mercedescodes |  @mercedes | mercedesbernard.com

And above all, make sure that throughout this process you are remembering your audience. You took all that time, in the beginning, to learn about them and understand their concerns so make sure you are incorporating that into your explanation.

Use Your Audience's POV

- Think about how the person you are talking to will interact with this tech/solution every day
- Explain what happens from their point of view

 @mercedescodes |  @mercedes | mercedesbernard.com

Think about how the person you're talking to will interact with your solution every day and then explain your solution from their point of view.

If your stakeholders are in marketing and you're proposing that they spend some of their budget on implementing CI/CD for their project, consider their concerns and frame your proposal from their perspective. The marketing team probably cares about the time it takes for their feature requests to be implemented and reflected on the company's website. Talk about how an automated CI/CD pipeline removes the friction with the development and infrastructure teams to schedule after-hours deployments when someone is available to stay late, manually deploy, and then health check the site. With an automated CI/CD pipeline, deployment cycles shorten dramatically. Features can be automatically pushed to production after they've been successfully tested in staging no matter what time of day or who is available. Now the marketing team can get really excited about CI/CD and will be more likely to spend their budget on it than if you had talked only about the technology and reliability or uptime.

Pay Attention to Body Language



 @mercedescodes |  @mercedes | mercedesbernard.com

And finally, make sure that you're paying attention to your audience and their body language while you're speaking. If you notice crossed arms, lots of yawning, sleepy eyes, or slouching, it may be time to call it a day and table the discussion. Not every meeting is perfect and it's ok to say, "let me spend some time putting more information and we'll revisit this." It's better to revisit a discussion when people are fresh rather than trying to plow through to get your point across while your audience is exhausted.

Let's Tie It All Together

Body Level One

 @mercedescodes |  @mercedes | mercedesbernard.com

10 min

Using the 3rd item you wrote down at the start of this talk, turn to someone near you who you don't work with and using the techniques discussed, explain your solution

 @mercedescodes |  @mercedes | mercedesbernard.com

10 min

- Know your audience
- No jargon
- Metaphors
- Explain “just enough”
- Slow down
- 2 different ways of explaining

Questions

Body Level One

 @mercedescodes |  @mercedes | mercedesbernard.com

 @mercedescodes |  @mercedes | mercedesbernard.com