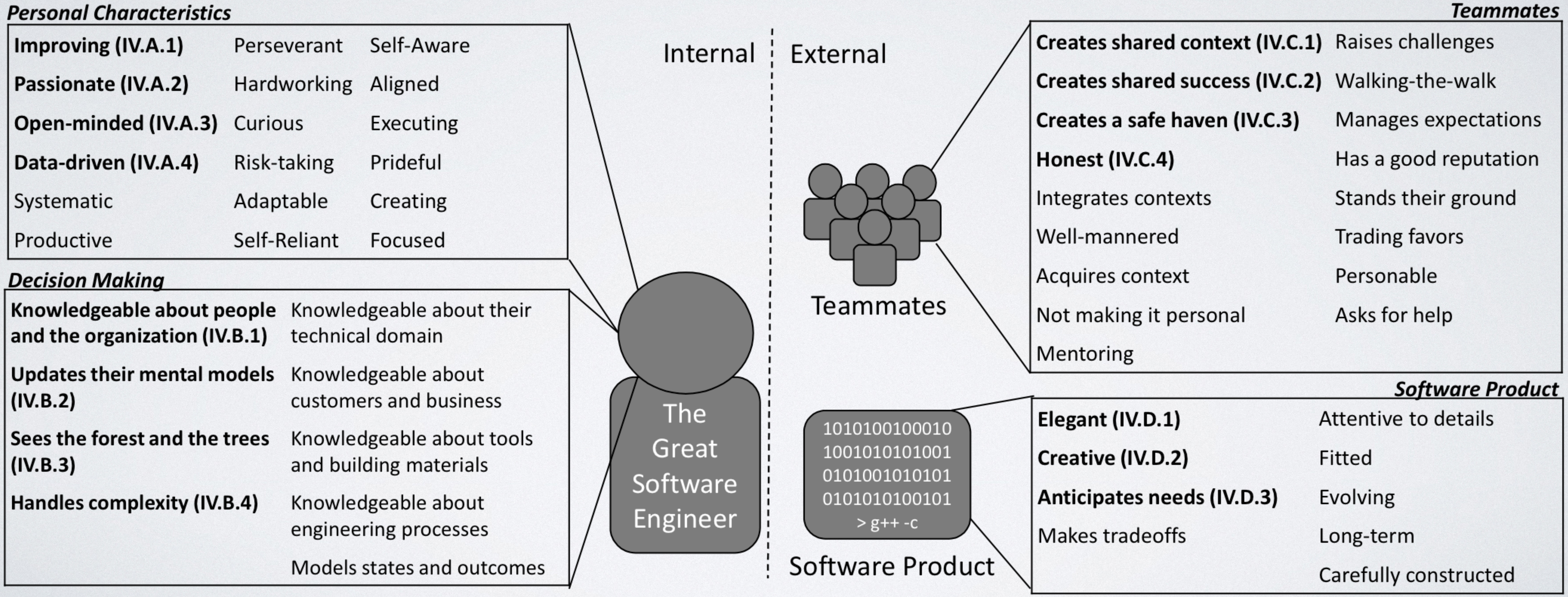
Hiring and Intervieweing

What Makes a Great Software Engineer?

- 2015 paper by Paul Luo Li, Amy J. Ko, and Jiamin Zhu
- Researchers interviewed 59 software engineers at Microsoft
  - Identified 53 attributes of great engineers
- Use this list for personal development and hiring
  - Both as a hiring manager and as a candidate

Categories of Characteristics

Internal attributes of people (left) lead to external benefits (right)



Personal Characteristics

- **Improving (IV.A.1)**
- **Passionate (IV.A.2)**
- **Open-minded (IV.A.3)**
- **Data-driven (IV.A.4)**
- Systematic
- Productive
- Perseverant
- Hardworking
- Curious
- Risk-taking
- Adaptable
- Self-Reliant
- Self-Aware
- Aligned
- Executing
- Prideful
- Creating
- Focused

Decision Making

- **Knowledgeable about people and the organization (IV.B.1)**
- **Updates their mental models (IV.B.2)**
- **Sees the forest and the trees (IV.B.3)**
- **Handles complexity (IV.B.4)**
- Knowledgeable about their technical domain
- Knowledgeable about customers and business
- Knowledgeable about tools and building materials
- Knowledgeable about engineering processes
- Models states and outcomes

Teammates

- **Creates shared context (IV.C.1)**
- **Creates shared success (IV.C.2)**
- **Creates a safe haven (IV.C.3)**
- **Honest (IV.C.4)**
- Integrates contexts
- Well-mannered
- Acquires context
- Not making it personal
- Mentoring
- Raises challenges
- Walking-the-walk
- Manages expectations
- Has a good reputation
- Stands their ground
- Trading favors
- Personable
- Asks for help

Software Product

- **Elegant (IV.D.1)**
- **Creative (IV.D.2)**
- **Anticipates needs (IV.D.3)**
- Makes tradeoffs
- Attentive to details
- Fitted
- Evolving
- Long-term
- Carefully constructed

Fig. 1. Model of attributes of great software engineers, with attributes we discuss in detailed in bold.

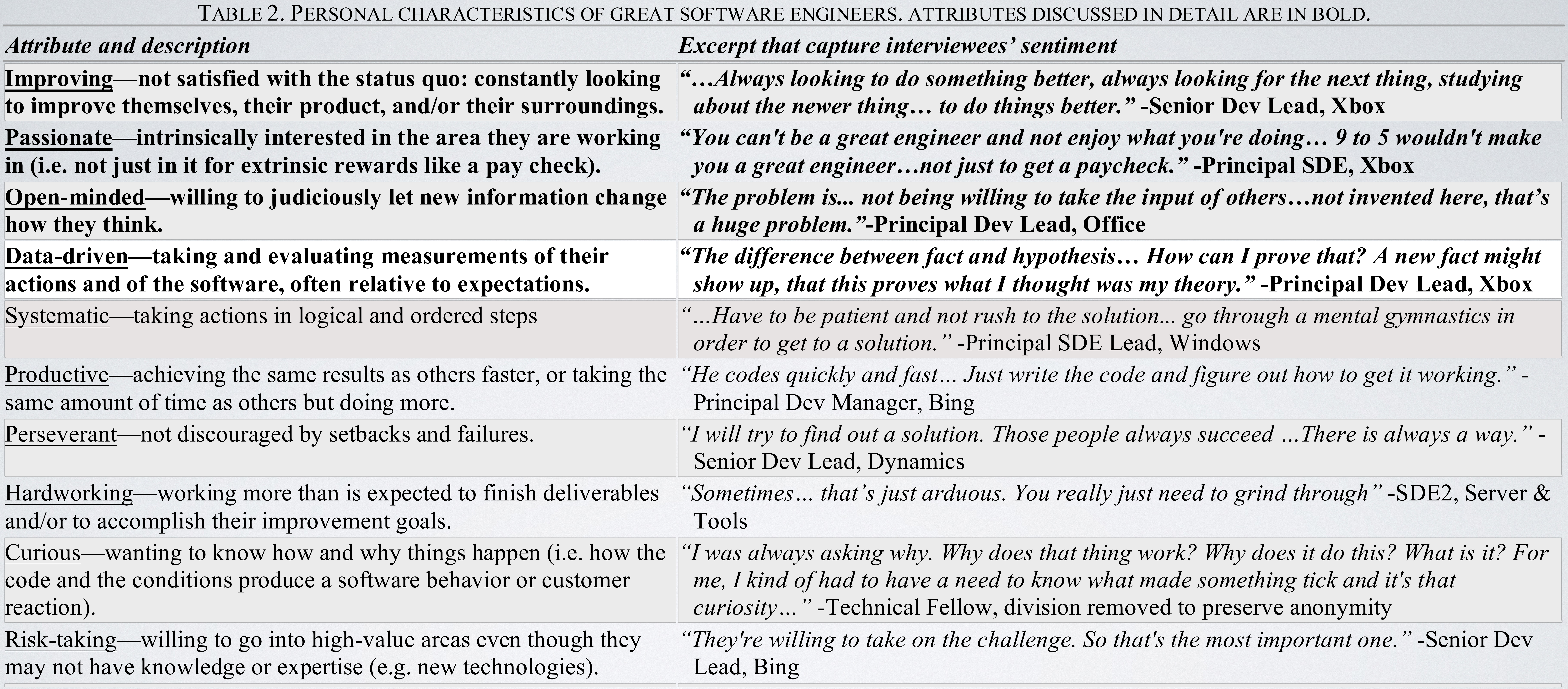


TABLE 2. PERSONAL CHARACTERISTICS OF GREAT SOFTWARE ENGINEERS. ATTRIBUTES DISCUSSED IN DETAIL ARE IN BOLD.

Attribute and description	*Excerpt that capture interviewees' sentiment*
Improving—not satisfied with the status quo: constantly looking to improve themselves, their product, and/or their surroundings.	***"…Always looking to do something better, always looking for the next thing, studying about the newer thing… to do things better."* -Senior Dev Lead, Xbox**
Passionate—intrinsically interested in the area they are working in (i.e. not just in it for extrinsic rewards like a pay check).	***"You can't be a great engineer and not enjoy what you're doing… 9 to 5 wouldn't make you a great engineer…not just to get a paycheck."* -Principal SDE, Xbox**
Open-minded—willing to judiciously let new information change how they think.	***"The problem is... not being willing to take the input of others…not invented here, that's a huge problem."*-Principal Dev Lead, Office**
Data-driven—taking and evaluating measurements of their actions and of the software, often relative to expectations.	***"The difference between fact and hypothesis… How can I prove that? A new fact might show up, that this proves what I thought was my theory."* -Principal Dev Lead, Xbox**
Systematic—taking actions in logical and ordered steps	*"...Have to be patient and not rush to the solution... go through a mental gymnastics in order to get to a solution."* -Principal SDE Lead, Windows
Productive—achieving the same results as others faster, or taking the same amount of time as others but doing more.	*"He codes quickly and fast… Just write the code and figure out how to get it working."* -Principal Dev Manager, Bing
Perseverant—not discouraged by setbacks and failures.	*"I will try to find out a solution. Those people always succeed …There is always a way."* -Senior Dev Lead, Dynamics
Hardworking—working more than is expected to finish deliverables and/or to accomplish their improvement goals.	*"Sometimes… that's just arduous. You really just need to grind through"* -SDE2, Server & Tools
Curious—wanting to know how and why things happen (i.e. how the code and the conditions produce a software behavior or customer reaction).	*"I was always asking why. Why does that thing work? Why does it do this? What is it? For me, I kind of had to have a need to know what made something tick and it's that curiosity…"* -Technical Fellow, division removed to preserve anonymity
Risk-taking—willing to go into high-value areas even though they may not have knowledge or expertise (e.g. new technologies).	*"They're willing to take on the challenge. So that's the most important one."* -Senior Dev Lead, Bing

(table is abbreviated; see the paper for the rest)

Improving

- Technology changes quickly

- “Computer technology, compared to other sciences or technology, it's pretty young. Every year there's some new technology, new ideas. If you are only satisfied with things you already learned, then you probably find out in a few years, you're out of date... good software engineer [sic], he keep investigate, investment. [sic]” -SDE2, Corp Dev

Passionate

- “I found that there's always a person who's passionate about every type of thing, you just have to find the right people... I ended up in the wrong job for six months. It was painful. People around me, they loved their work” - Principal Dev Lead, Phone

Open-Minded

- “You should be open... what you think need not be the right thing tomorrow... like the Facebook explosion, when Myspace was already there, but it exploded... no one knew that Facebook would explode when it started”. - Senior SDE, Windows Services

Decision Making

- Book knowledge is insufficient

- "Great engineers not only knew what should happen, but also what can and likely will happen."

Knowledgeable About People and the Organization

- Know who knows what

- “Make sure that you are aware of that big picture, you know where you fit in and how you interact with everyone else to optimize what you are doing.” - Principal Dev Lead, Ad Platform

- Enables asking for help

Sees the Forest and the Trees

- “What differentiated [this great engineer] from other people in management positions... capability to zoom into the details, and he was not just a high level guy, ...know the reality of the stack or the reality of the software...’’ -Senior Dev Lead, Ad Platform

Update Their Mental Models

- “You can always follow patterns too much... It's worked in the past, but conditions have changed. You always need to look and take a little bit of risk with each one of your tasks. If you're not then you're not really going to find out what's possible.” -Principal Dev Lead, Office

Handling Complexity

- “To solve the problem, [great engineers] have to have the ability to connect things... You are always debugging layers of stacks of code... this layer talks to some other layer in the horizontal... you need to solve the problem and you don't know what's going on.” -Senior SDE, Windows Services

Impact on Teammates

- Creates Shared Context
- Creates Shared Success
- Creates a Safe Haven
- Honest

Creates Shared Context

- “You perceive who you are talking to, and you are able to judge on those levels that they are, or you just ask important questions. Do you know about this? And then, be able to simplify the problem to the level that they’re working in, or you estimate the amount of information given to them.” -Senior SDE, Windows

Creates Shared Success

- The team succeeds or fails together. Personal compromise is often needed.

- “It's a two-way communication... there's something going to happen down the road, this piece of code or this feature going to have some issues, need to make your manager aware.” -SDE2, Phone

Creates a Safe Haven

- Learn and improve from mistakes without negative consequences

- But:

  - “I believe in having people feel the pain of their own mistakes... dealing with the ramifications of the decisions that are being made, I guess is the best way to learn.” -Principal Dev Lead, Office

Honest

- "'Influence comes to someone else trusting you, part of that trust is that they go, 'You know what? I know that this person always speaks the truth.' As a result of that, when they say something is good, I will totally believe them because they are not trying to kind of misrepresent something or make them look better or whatever." -Windows Services Principal Dev Manager

How Good Is a Candidate?

- Need to interview the candidate (maybe *you*) to find out!
- Two kinds of questions:
    - Technical
    - Behavioral

Behavioral Questions

- Premise: the best predictor of the future is the past
- Behavioral questions ask about your past experience
- Some intern interviews skip these (they assume you have no experience)
- "Tell me about a time when…"

Answering Behavioral Questions

- Standard advice: use "STAR" framework for answers
  - Situation: explain why the situation is relevant to the question
  - Task: describe your role in the situation
  - Action: say what you did to overcome the challenge
  - Result: explain the outcome. Include concrete examples and quantifiable achievements

Example

- Tell me about a time when you realized you needed to grow as a software engineer. What did you do to improve?

Activity

- Write a behavioral question that seeks information about one of the key SE skills from the paper.

Answering Questions

- Now, turn to a partner.
  - Ask your question
  - Answer theirs