

1968:

*„It seems almost automatic that software is **never produced on time, never meets specification, and always exceeds its estimated cost.**”*

Die Komplexität der Softwareentwicklung

- 1968: Als Geburtsstunde des Software Engineering
 - „Software Crisis“ als Auslöser
 - NATO-Konferenz von Garmisch
 - Siegeszug des Software Engineerings
- Aber: Es gibt keine „Silver Bullet“
 - Komplexität der Softwareprojekte steigt immer weiter an

Wer sind die Menschen hinter der Komplexität?

Software Engineer



<https://www.spiegel.de/karriere/programmieren-haben-sie-das-zeug-dazu-a-b568bf83-46aa-423a-b02a-0c35b7ee4738>

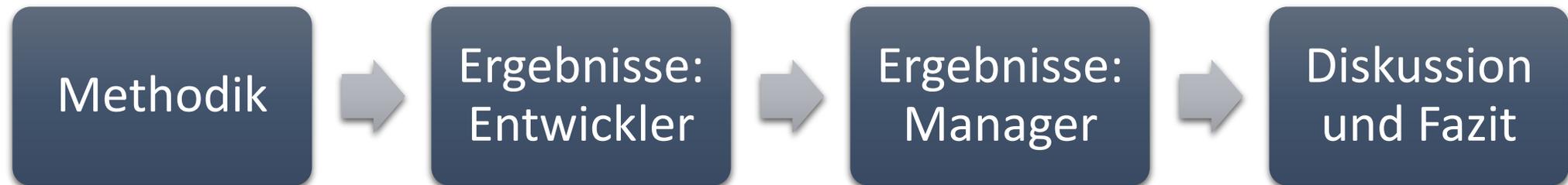
Software Engineering Manager*



<https://yieldhub.com/what-we-do/engineering-manager-semiconductor-yield-management-resources-2/>

Beiträge zum Software Engineering:

Die Eigenschaften großartiger Entwickler und Manager



Vorgestellte Forschung

Entwickler



- ① P. L. Li, A. J. Ko, and J. Zhu. **What makes a great software engineer?** In 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, 2015
- ② P. L. Li. What Makes a Great Software Engineer. PhD thesis, 2016

Manager



- ① Kalliamvakou, Eirini, et al. **What makes a great manager of software engineers?** In Proceedings of the 40th International Conference on Software Engineering, 2018
- ② D. A. Garvin, A. B. Wagonfeld, and L. Kind. **Google's project oxygen: Do managers matter?** 2013

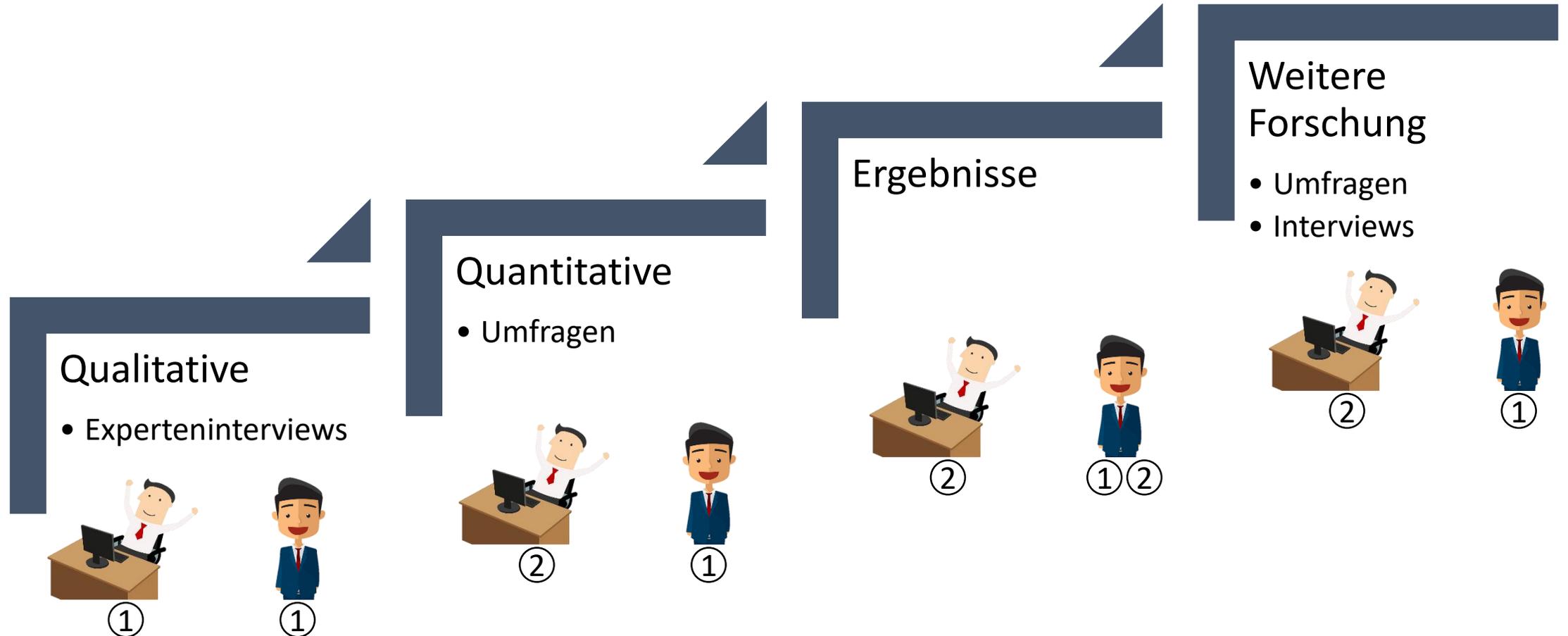
Was genau wird vorgestellt?

Entwickler

- ① P. L. Li, A. J. Ko, and J. Zhu. **What makes a great software engineer?** In 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, 2015
- ② P. L. Li. **What Makes a Great Software Engineer.** PhD thesis, 2016

Manager

- ① Kalliamvakou, Eirini, et al. **What makes a great manager of software engineers?** In Proceedings of the 40th International Conference on Software Engineering, 2018
- ② D. A. Garvin, A. B. Wagonfeld, and L. Kind. **Google's project oxygen: Do managers matter?** 2013



Methodik: Zielgruppe

Entwickler



- All jene Leute, die im Rahmen der Softwareentwicklung Code als Ergebnisse ihrer Arbeit produzieren.
- „Programmierer“

Manager



- Leiter von Entwicklerteams
- Programmieren meist nicht
- „Teamleiter“

Methodik: Interviews

- Befragte:
 - Softwareentwicklung bei Microsoft (Entwickler, Architekten, Manager, etc.)
 - Unterschiedliches Level an Expertise
 - Unterschiedliche Hierarchiestufen
- Unterschiede:
 - Explorative Interviews 
 - Vorangehende Recherche + Befragung / Diskussion innerhalb der Interviews
- Ergebnis: Liste von Attributen



Methodik: Umfragen

- Befragte: äquivalent zu Interviews
- Teilnehmer:
 - Ca. 2000 zu Entwicklern
 - Ca. 600 zu Managern
- Signifikanz der in Interviews gefundenen Attribute wird von Befragten bewertet
- Unterschiede:
 - Fragen zu 53 Attributen 
 - 15 Attribute + weitere Fragen zur Wahrnehmung von Managern 

Methodik: Umfragen - Beispiele

A great engineering manager...*

	Critical									Not important
mediates inter-team interaction Acting as a buffer with other teams/disciplines/managers, negotiating what her team can provide when, and mediating her own team's requests to other teams.	<input type="radio"/>									



Systematic

A systematic developer does not rush to conclusions or jump to conclusions; they address problems in a systematic and organized manner.

28) If an experienced developer---whose primary responsibility is developing software---did not have this attribute, could you still consider them a great developer?*

- Cannot be a great developer if they do not have this
- Very difficult to be a great developer without this, but not impossible
- Can be a great developer without this, but having it helps
- Does not matter if they do not have this, it is irrelevant
- A great developer should not have this; it is not good
- I do not know

Methodik: Weitere Forschung

- Befragung angrenzender Fachbereiche (Finance, Operations, etc.)

- In Form von Interviews



- In Form einer weiteren Umfrage



Was genau wird vorgestellt?

Entwickler

- ① P. L. Li, A. J. Ko, and J. Zhu. **What makes a great software engineer?** In 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, 2015
- ② P. L. Li. **What Makes a Great Software Engineer.** PhD thesis, 2016

Manager

- ① Kalliamvakou, Eirini, et al. **What makes a great manager of software engineers?** In Proceedings of the 40th International Conference on Software Engineering, 2018
- ② D. A. Garvin, A. B. Wagonfeld, and L. Kind. **Google's project oxygen: Do managers matter?** 2013

Fragen?

Ergebnisse

Weitere Forschung

- Umfragen
- Interviews

Qualitative

- Experteninterviews



①



①



②



①



②



① ②



②



①

Ergebnisse

What Makes a Great Software Engineer?

- 5 Kerneigenschaften eines exzellenten Entwicklers

What makes a great manager of software engineers?

- Framework eines großartigen Managers

What Google's Best Managers Do:

- 8 Schlüsseigenschaften der effektivsten Manager

5 Kerneigenschaften eines exzellenten Entwicklers:



5 Kerneigenschaften eines exzellenten Entwicklers

①

Der Entwickler als
kompetenter
Programmierer

- Programmieren als zentrale Aufgabe des Entwicklers
- Essentielle Grundlage
- Andere Attribute kommen untergeordnet zum Tragen

Attribute:

Auf Details im Code achten

Nicht schlampig arbeiten

Code fügt sich gut ein

5 Kerneigenschaften eines exzellenten Entwicklers

②

Der Entwickler als
Maximierer des
Wertes

- Berücksichtigt den Kontext bei der Entwicklung
- Schätzt Risiken ein
- Handelt „ökonomisch“

Attribute:

Sieht Anforderungen voraus

Denkt langfristig

Systematisch

Findet elegante Lösungen

5 Kerneigenschaften eines exzellenten Entwicklers

3

Der Entwickler als informierter Entscheidungsträger

- Entwickler trifft Vielzahl von Entscheidungen auf technischer Ebene
- Berücksichtigt alle Informationen

Attribute:

Systematisch

Kann um Hilfe bitten

Offen für andere Vorschläge

Daten als Entscheidungsgrundlage

5 Kerneigenschaften eines exzellenten Entwicklers

4

Der Entwickler als
Unterstützer der
Anderen

- Unterstützt Kollegen (Entwickler, Manager, andere Abteilungen)
- Macht die Arbeit anderer nicht schwieriger
- „Manages Expectations“

Attribute:

Ehrlich

Unterstützt andere

Selbstreflektiert

Erklärt verständlich

Kann um Hilfe bitten

5 Kerneigenschaften eines exzellenten Entwicklers

5

Der Entwickler als
stetig Lernender

- Hohe Frequenz an neuen Methoden, Trends, etc.
- Um effektiv Arbeiten zu können, muss stetig Neues gelernt werden

Attribute:

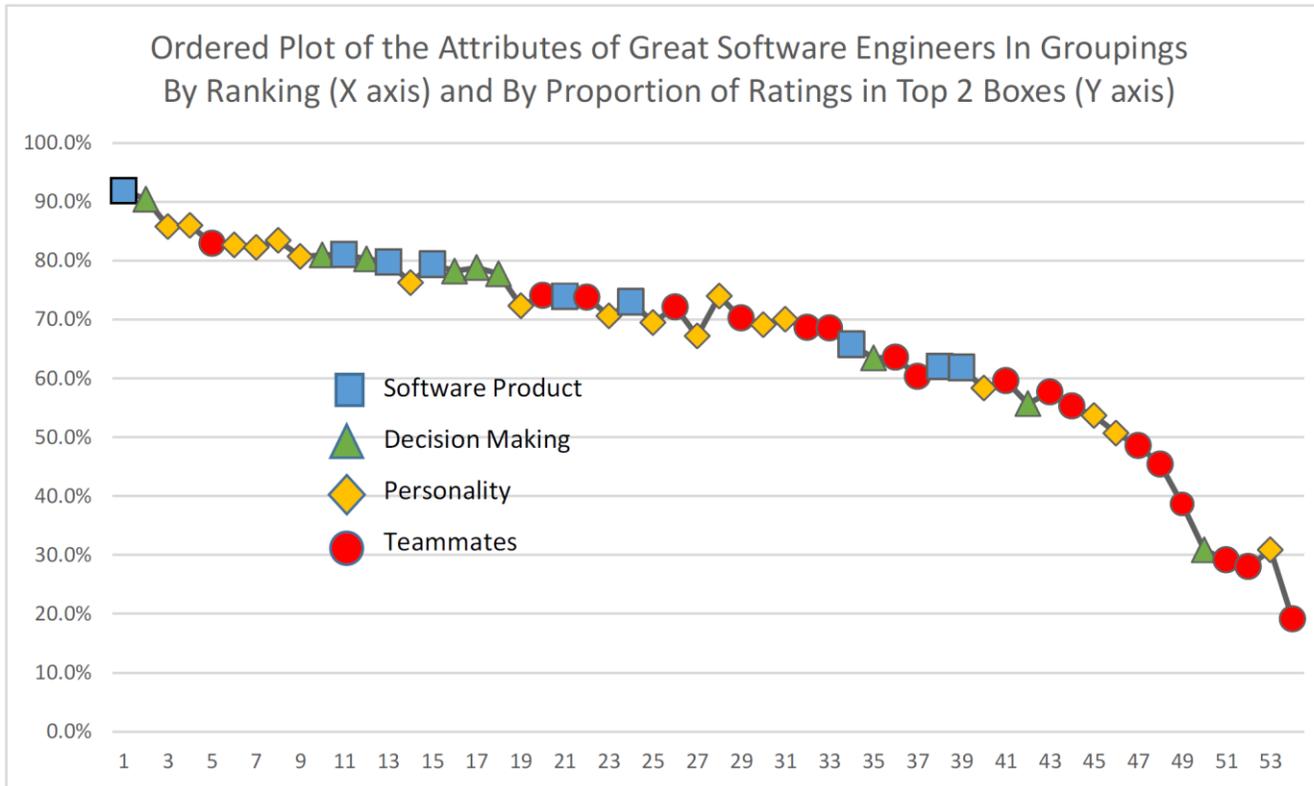
Neugierig

Sich stetig verbessernd

Lernt von anderen

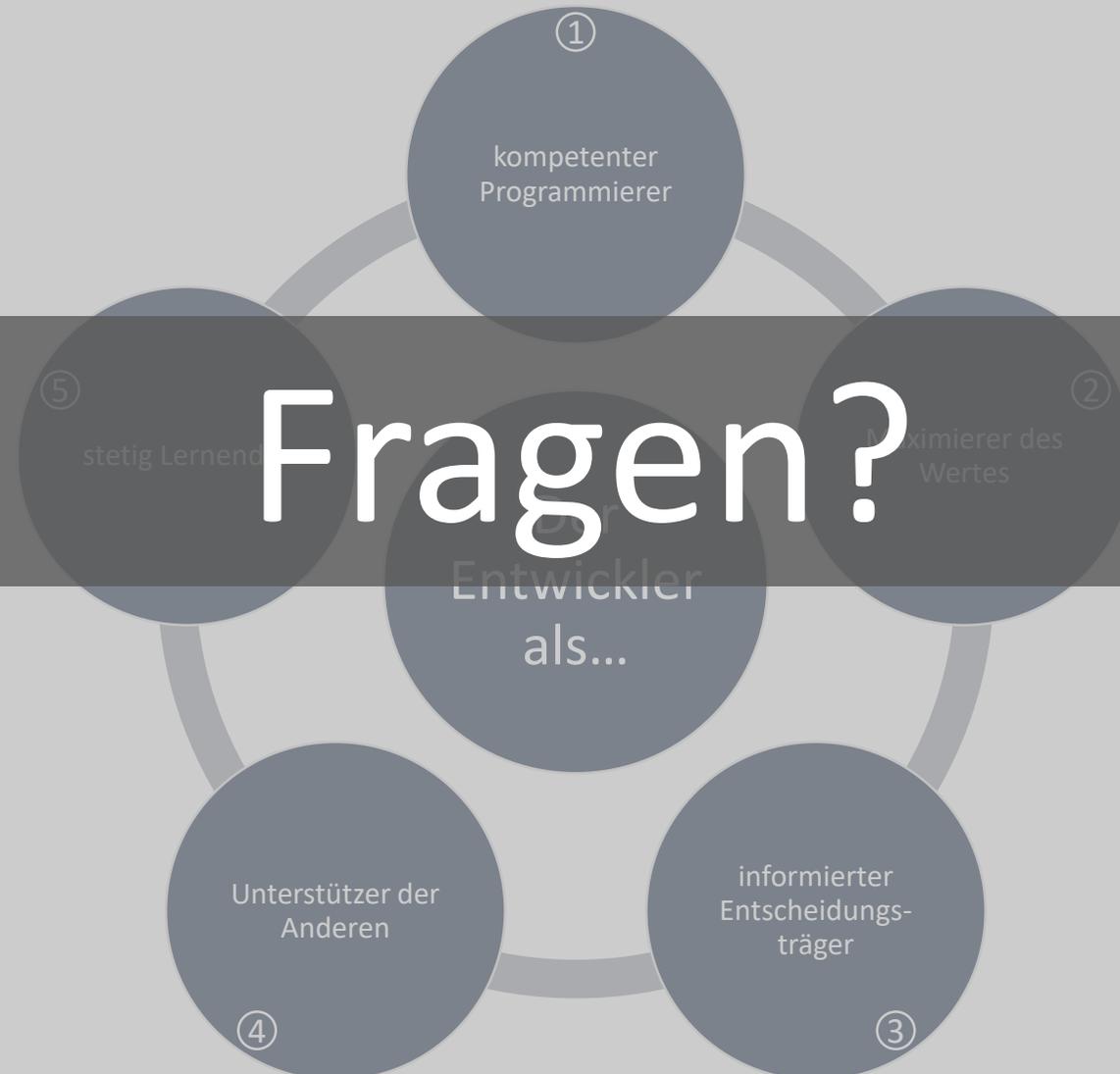
Berücksichtigt verschiedene Perspektiven

Weitere Ergebnisse: Rangliste aller 53 Attribute nach Kategorien

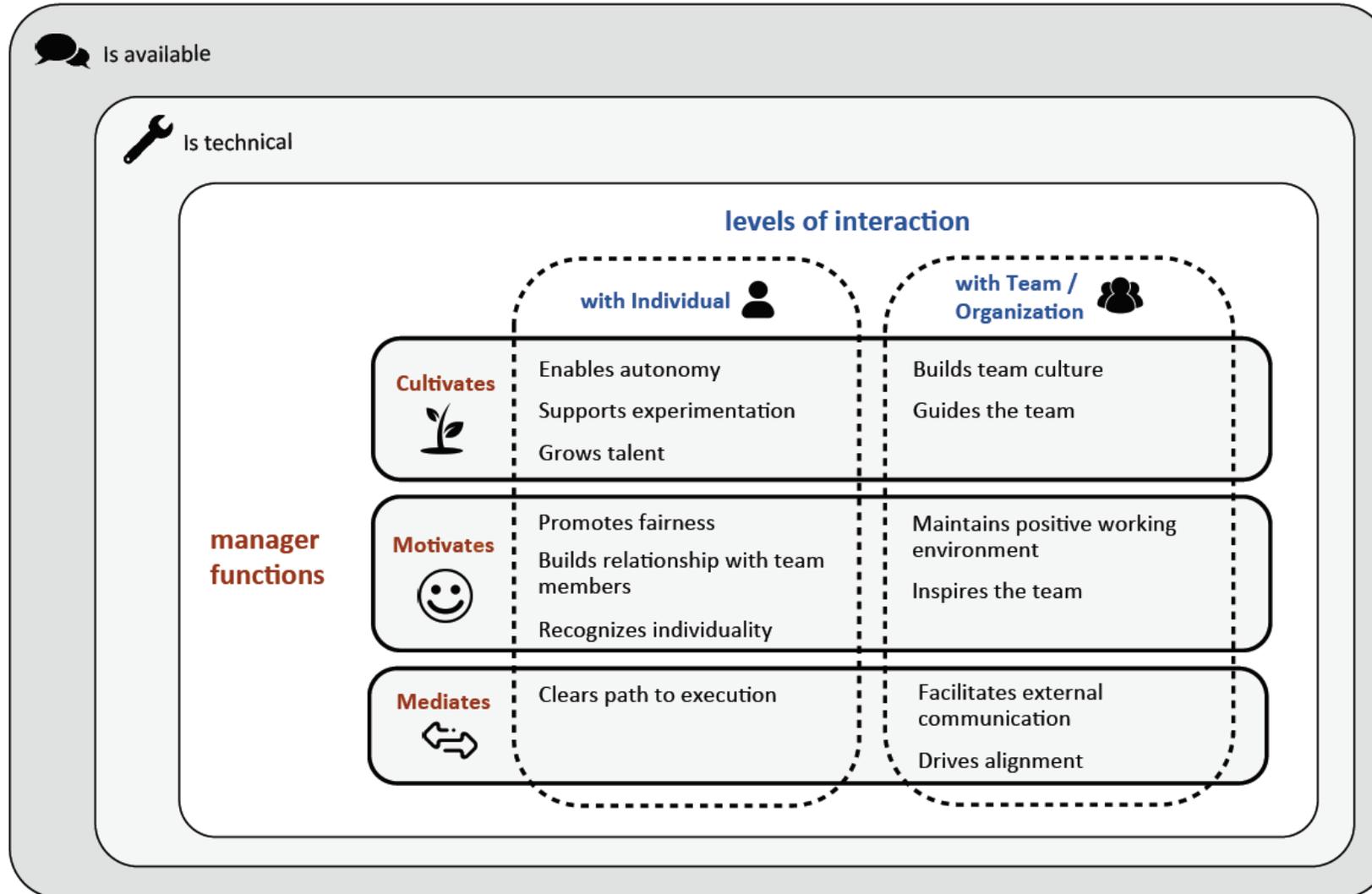


1. Decision making (\emptyset Rang: 17)
2. Software Product (\emptyset Rang: 17,5)
3. Personality (\emptyset Rang: 24)
4. Teammates (\emptyset Rang: 40)

5 Kerneigenschaften eines exzellenten Entwicklers:



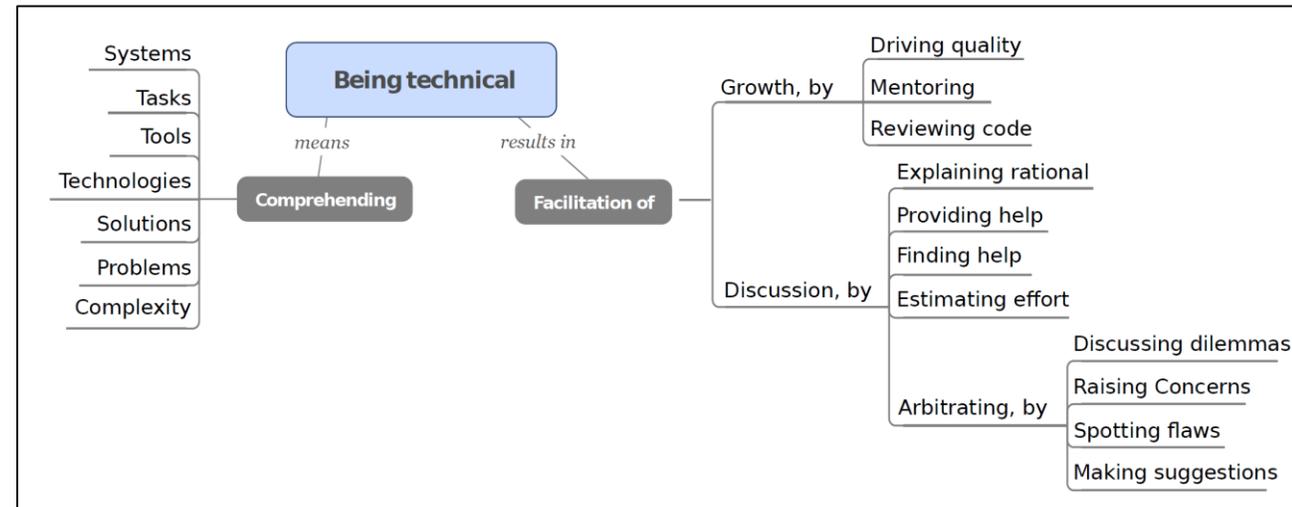
Framework eines großartigen Managers



Der Manager als technisch Erfahrener

- Technisches Wissen ist notwendig:

- Respekt des Teams
- Qualität und nicht funktionale Anforderungen kontrollieren
- Geleistete Arbeit bewerten
- Das Team repräsentieren
- Hineinversetzen in Teammitglieder



- Unterstützen des Teams im Vordergrund
- Manager sind i.d.R. keine Programmierer

Framework eines großartigen Managers: Der Manager als Förderer

Cultivates



- Entwickler wird als technischer Experte behandelt
- Überlässt technische Entscheidungen dem Team / Entwickler
- Unterstützt Entscheidungsfindung
- Schafft sichere Arbeitsatmosphäre

Attribute:

Autonomie ermöglichen 

Experimentierfreudigkeit unterstützen 

Talent fördern 

Kultur aufbauen 

Team leiten und unterstützen 

Framework eines großartigen Managers: Der Manager als Motivator

Motivates



- Wertschätzung und Lob für geleistete Arbeit
- „Praise publicly; correct privately“
- Kennt Stärken, Schwächen, Vorlieben etc.
- Persönliche Beziehung als Motivator
- Gesundes Arbeitsumfeld

Attribute:

Fördert Fairness



Baut Beziehungen auf



Erkennt Individualität



Schafft positive Arbeitsatmosphäre



Inspiziert das Team



Framework eines großartigen Managers: Der Manager als Mediator



- Agiert als Filter und Schutzschild des Teams
- Teilt Informationen und Vision
- Unterstützt Team innerhalb der Organisation

Attribute:

„clears the path to execution“



Erleichtert externe Kommunikation



Fördert Zusammenhalt



Weitere Ergebnisse: Unterschiede in den befragten Gruppen

Positive coefficients indicate higher importance. Negative coefficients indicate lower importance.

Demographic	Attribute	Change
Role: Managers (90) compared to Engineers (465)	😊 👤 builds a relationship with team members ↑	1.21
	😊 👤 inspires the team ↑	0.98
	🌱 👤 builds team culture ↑	0.97
	🗣️ is available ↑	0.60
	🌱 👤 grows talent ↑	0.54
Gender: Female (59) compared to Male (494)	🔧 is technical ↑	0.83
	😊 👤 inspires the team ↑	0.80
	↔️ 👤 facilitates external communication ↑	0.62
	🌱 👤 guides the team ↑	0.53
	🌱 👤 grows talent ↑	0.46
Region: China (32) compared to U.S. (410)	🔧 is technical ↑	1.02
	😊 👤 recognizes individuality ↑	0.70
	🌱 👤 guides the team ↑	0.65
Region: India (56) compared to U.S. (410)	🌱 👤 builds team culture ↑	0.99
	😊 👤 builds a relationship with team members ↑	0.85
	🌱 👤 grows talent ↓	-0.55
Region: Europe (50) compared to U.S. (410)	↔️ 👤 drives alignment ↓	-0.53
	😊 👤 recognizes individuality ↓	-0.50
Mgr Group Size (in people)	↔️ 👤 clears path to execution ↓	-0.01
	🌱 👤 enables autonomy ↓	-0.01
Years at Microsoft	🌱 👤 grows talent ↓	-0.03

- Manager legen mehr Wert auf:
 - Aufbau von Beziehungen / Kultur
 - Inspiration / Fördern
- Frauen im Vergleich zu Männern:
 - Technische Expertise
 - Inspiration des Teams
- Kulturelle / Geographische Unterschiede:
 - China: technische Expertise + Individualität
 - Indien: Kultur und persönliche Beziehung

Google: 8 Schlüsseleigenschaften der effektivsten Manager

Ein guter Manager:

ist ein guter Coach.

überträgt dem Team Entscheidungen und kontrolliert nicht bis ins Kleinste hinein.

zeigt Interesse und beschäftigt sich mit den einzelnen Teammitgliedern bezogen auf ihren Erfolg und ihr Wohlergehen.

ist produktiv und fokussiert auf Ergebnisse.

kann gut kommunizieren, also sowohl zuhören als auch Informationen teilen.

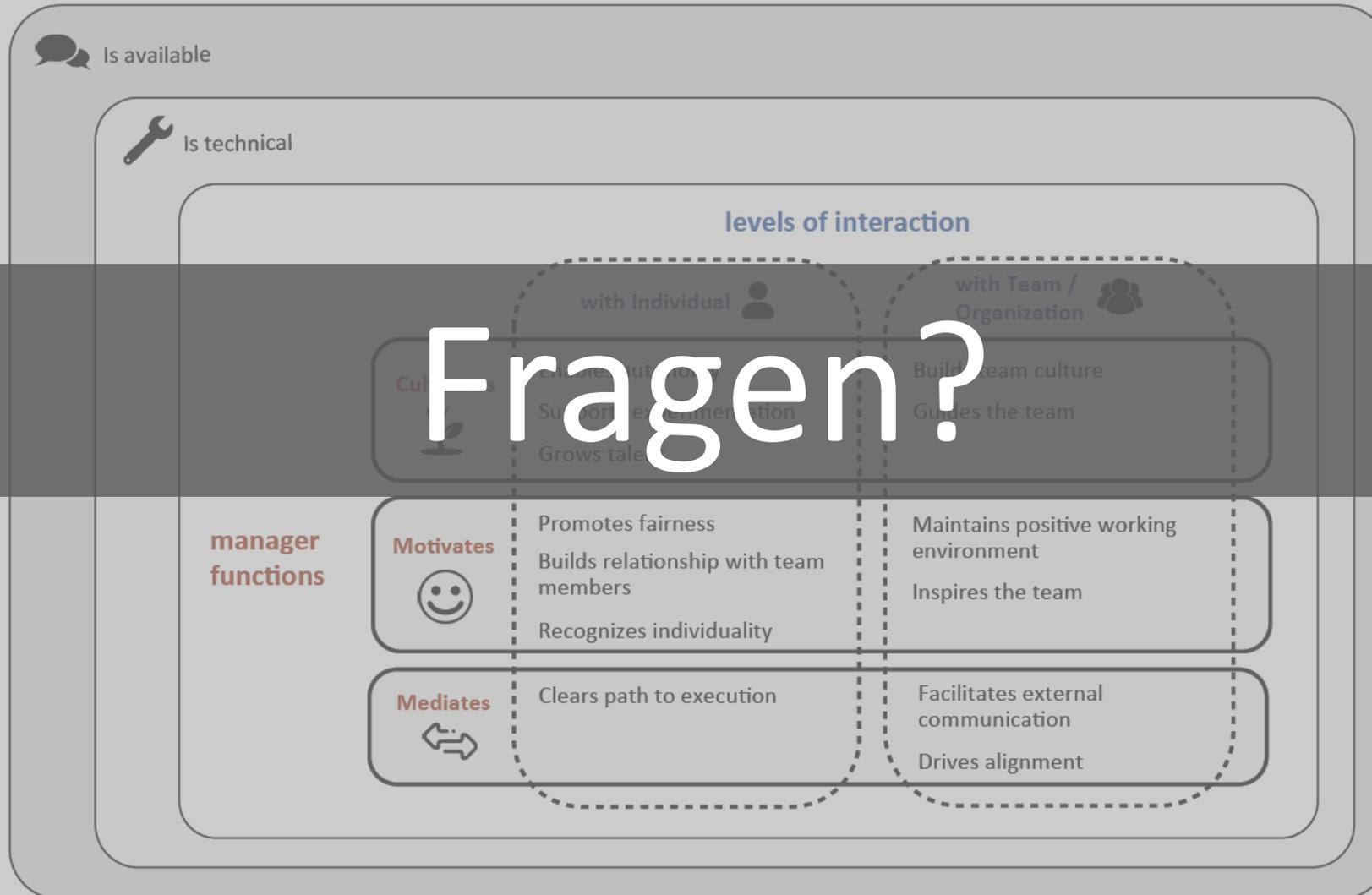
unterstützt die Entwicklung der Karriere.

hat eine klare Vision für sein Team.

hat die technische Expertise, um sein Team beraten und unterstützen zu können.

- Hier: Manager aller Hierarchien und Abteilungen bei Google
- Sehr ähnliche Ergebnisse
- Es fehlen domänenspezifische Eigenschaften

Framework eines großartigen Managers



Vergleich der Ergebnisse: Gemeinsamkeiten

- Technische Expertise als Grundlage
- Sozialer Umgang mit Kollegen von essentieller Bedeutung
- Unterstützen des Teams
- Fundierte Entscheidungen treffen
- Ehrlichkeit und Transparenz

Vergleich der Ergebnisse: Erkenntnisse

- Entwickler als Produzent von Code
- Manager als Leiter eines Teams, das Code produziert

- Hervorragende Entwickler bringen viele Eigenschaften mit, die das Ausfüllen der Managerrolle verlangen

Validität der Forschung

- Forschung nur bei Microsoft
 - Interviews und Umfragen bilden nur Meinung der Befragten ab, nicht zwangsläufig die Realität
 - Konzepte und Verständnis der Attribute und Rollen kann variieren
- Hohe interne und externe Validität und transparente Forschung

Fazit

- Ergebnisse interessant für Rekrutierung und Förderung im Bereich der Softwareentwicklung
- Wichtiges aber etwas unterrepräsentiertes Thema

Beiträge zum Software Engineering:
Die Eigenschaften großartiger
Entwickler und Manager

Vielen Dank!

Fragen / Anmerkungen / Kritik?

Methodik

Entwickler

Manager

Diskussion
und Fazit

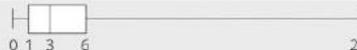
Appendix: Entwickler - 53 Attribute



# Higher	Ratings distribution	Attribute and description
53	■ ■ ■ ■ ■	<u>Pays attention to coding details</u> : including error handling, memory consumption, performance, and style
52	■ ■ ■ ■ ■	<u>Mentally capable of handling complexity</u> : able to comprehend and understand complex situations, including multiple layers of technology and interacting/intertwining software
49	■ ■ ■ ■ ■	<u>Continuously improving; constantly looking to become better</u> : improving themselves, their product, or their surroundings
49	■ ■ ■ ■ ■	<u>Honest</u> : truthful, not sugar coating or spinning the situation for their own benefit. They provide credible information and feedback that others can act on
49	■ ■ ■ ■ ■	<u>Open-minded</u> : willing to let new information change their thinking; do not believe they know everything and will consider new information
46	■ ■ ■ ■ ■	<u>Executes</u> : does not have analysis paralysis; knows when to stop thinking and to start doing
45	■ ■ ■ ■ ■	<u>Self-reliant</u> : gets things done independently and does not get blocked easily; they get around problems by themselves
45	■ ■ ■ ■ ■	<u>Self-reflecting</u> : recognize when things are going wrong or when their current plan is not going to work, and then self-initiate corrective actions.
43	■ ■ ■ ■ ■	<u>Persevering</u> : not dissuaded by setbacks and failures; keeps on going, keeps on trying
41	■ ■ ■ ■ ■	<u>Fits together with other pieces around it</u> : such as environmental constraints, complementary components, and other products.
41	■ ■ ■ ■ ■	<u>Knowledgeable about their technical domain</u> : thoroughly conversant about their software product, technology area, and competitors
39	■ ■ ■ ■ ■	<u>Makes informed trade-offs</u> : in their software (e.g. code quality for time to market), meeting critical needs of the situation
39	■ ■ ■ ■ ■	<u>Updates their decision making knowledge</u> : does not let their understanding and thinking stagnate
36	■ ■ ■ ■ ■	<u>Curious</u> : desires to know why things happen and how things work
36	■ ■ ■ ■ ■	<u>Evolving</u> : structured to be effectively built, delivered, and updated in pieces.
35	■ ■ ■ ■ ■	<u>Knowledgeable about tools and building materials</u> : knows the strengths and limitations of technologies used to construct their software
35	■ ■ ■ ■ ■	<u>Grows their ability to make good decisions</u> : builds their understanding of real world situations, including alternative, outcomes, and values of the outcomes.
34	■ ■ ■ ■ ■	<u>Sees the forest and the trees</u> : reasons through situations and problems at multiple levels of abstraction
31	■ ■ ■ ■ ■	<u>Craftsmanship</u> : takes pride in their work; wants their output to be a reflection of their skills and abilities
30	■ ■ ■ ■ ■	<u>Does due diligence beforehand</u> : searches for and examines available information before engaging. They are prepared when they discuss situations and do not waste others' time.
30	■ ■ ■ ■ ■	<u>Elegant</u> : intuitive software (i.e. minimum complexity) design solutions that others can understand
29	■ ■ ■ ■ ■	<u>Asks for help</u> : will find and engage others with needed knowledge and information. They know the limits of their knowledge and supplement their knowledge with the knowledge of others
28	■ ■ ■ ■ ■	<u>Desires to turn ideas into reality</u> : takes pleasure in building, constructing, and creating software
28	■ ■ ■ ■ ■	<u>Long-termed</u> : considers long-term costs and benefits in producing software and designs, not just short-term gratification
25	■ ■ ■ ■ ■	<u>Willing to go into the unknown</u> : willing to step outside of comfort zone to explore a new area, even when risky and benefits uncertain
24	■ ■ ■ ■ ■	<u>Is a good listener</u> : effectively obtains, comprehends, and understands others' knowledge about the situation
22	■ ■ ■ ■ ■	<u>Passionate</u> : intrinsically interested in the area they are working in; not just in it for a pay check
22	■ ■ ■ ■ ■	<u>Manages expectations</u> : sets forth what they are going to do and by when, updates expectations (e.g. explaining impacts and implications of unexpected problems), and then delivers on promises
22	■ ■ ■ ■ ■	<u>Focused</u> : allocates and prioritizes time for the most impactful work; is not overwhelmed by daily distractions and tasks
21	■ ■ ■ ■ ■	<u>Systematic</u> : does not rush to conclusions or jump to conclusions; address problems in a systematic and organized manner
21	■ ■ ■ ■ ■	<u>Adapts to new settings</u> : continues to be valuable to the organization even with changes in their environment
19	■ ■ ■ ■ ■	<u>Integrates understandings of others</u> : can combine and integrate the knowledge of others into a more complete understanding, noticing and asking questions about the gaps
19	■ ■ ■ ■ ■	<u>Does not make it personal</u> : avoids personal biases. They act and react based on fact and reason, avoiding dysfunctional behaviors based on personal feelings and perceived slights
19	■ ■ ■ ■ ■	<u>Creative</u> : novel and innovative solutions based on understanding the context and limitations of existing solutions.
18	■ ■ ■ ■ ■	<u>Walks-the-walk</u> : is an exemplar for others: being a great developer themselves, letting others see their actions, and inspiring others to follow.
18	■ ■ ■ ■ ■	<u>Knowledgeable about software engineering processes</u> : knows the practices and techniques for building a software product: purposes, how to, costs, and when best to use
13	■ ■ ■ ■ ■	<u>Anticipates needs</u> —problems and needs not explicitly known at the time of creation—based on their knowledge and understanding
13	■ ■ ■ ■ ■	<u>Uses the right processes during construction</u> : using the right processes (e.g. unit testing and code reviews) to construct their software and designs, in order to deal with potential problems
13	■ ■ ■ ■ ■	<u>Resists external pressure for the good of the software product</u> : will articulate and advocate actions that are for the good of software product, being firm against outside pressures
11	■ ■ ■ ■ ■	<u>Has a good reputation</u> : has the belief, respect, and confidence of others. They have a track-record of success such that they are trusted with current and future decisions
11	■ ■ ■ ■ ■	<u>Productive</u> : achieves the same results as others faster, or takes the same amount of time as others but produces more
10	■ ■ ■ ■ ■	<u>Knowledgeable about customers and business</u> : understands the role their software product plays in the lives of their customers and the business proposition that it entails
10	■ ■ ■ ■ ■	<u>Creates shared understanding with others</u> : molds another person's thinking of the situation: tailoring the communication to be relevant and comprehensible
9	■ ■ ■ ■ ■	<u>Creates shared success for everyone</u> : win-win situations that is beneficial to everyone, commonly involving establishing a common big picture or long-term goals that everyone can buy into
8	■ ■ ■ ■ ■	<u>Aligned with organizational goals</u> : takes actions for the good of the product and the organization, not just what interests them
7	■ ■ ■ ■ ■	<u>Well-mannered</u> : developer treats others with respect: not obnoxious about titles, accolades, or knowledge
7	■ ■ ■ ■ ■	<u>Data-driven</u> : measures their software and the outcomes of decisions; let actual data drive actions, not depending solely on intuition
6	■ ■ ■ ■ ■	<u>Creates a safe haven for others</u> : where others are not afraid of being blamed for mistakes, empowering others to do what they feel is right, and to learn and grow
5	■ ■ ■ ■ ■	<u>Mentoring</u> : teaches, guides, and instills knowledge to others, helping others—often new team members—to improve and to be more productive.
4	■ ■ ■ ■ ■	<u>Knowledgeable about people and the organization</u> : informed about the people around them: responsibilities, knowledge, and tendencies.
2	■ ■ ■ ■ ■	<u>Challenges others to improve</u> : challenges others to take action (e.g. doing something new or taking on more responsibilities), expanding others' limits and capabilities
2	■ ■ ■ ■ ■	<u>Personable</u> : others enjoy interacting with; they establish good personal relationships with others
1	■ ■ ■ ■ ■	<u>Hardworking</u> : is willing to work more than 8 hr days to deliver the software product
0	■ ■ ■ ■ ■	<u>Trades favors</u> : builds personal equity with others, such that the developer can call upon others to do them personal favors

Appendix: Entwickler - Unterschiede in den befragten Gruppen



Row	Contextual factor	Encoding	Distribution of data <small>Box plots are min, 25th, median, 75th, max</small>	Significant relationships with attributes (OLR, FDR, $q=0.1$)																								
1	Is very experienced	Categorical Based on title	Very experienced (1,101, 57.2%) Experienced (825, 42.8%)	<u>Executes</u> (+) <u>Knowledgeable about tools and building materials</u> (+) <u>Makes informed tradeoffs</u> (+)																								
2	Age	Numerical Optional response 1,512 responses		<u>Knowledgeable about customers and business</u> (+) <u>Knowledgeable about software engineering processes</u> (+) <u>Aligned with organizational goals</u> (+)																								
3	Years as a professional developer	Numerical		<u>Hardworking</u> (+) <u>Desires to turn ideas into reality</u> (+)																								
4	Years at Microsoft	Numerical		<u>Aligned with organizational goals</u> (+)																								
5	Number of software companies in career	Numerical		<u>Continuously improving</u> (+)																								
6	Years on current team	Numerical		—																								
7	Experience as manager	Yes/No	Manager (1,028, 53.4%)	—																								
8	Is female	Yes/No	Female (149, 7.7%)	<u>Uses the right processes during construction</u> (L)																								
9	Has bachelors/ associates degree	Yes/No	Yes (1,228, 63.8%)	—																								
10	Has non-MBA masters	Yes/No	Yes (762, 39.6%)	<u>Asks for help</u> (-)																								
11	Has MBA	Yes/No	Yes (49, 2.5%)	—																								
12	Has doctorate	Yes/No	Yes (49, 2.5%)	<u>Walks-the-walk</u> (+) <u>Challenges others to improve</u> (+)																								
13	Has other degree	Yes/No	Yes (103, 5.3%)	—																								
14	Has non-CS degree	Yes/No Based on self-reported degree	Yes (537, 27.9%)	—																								
15	Is not currently working in the US	Yes/No	Yes (351, 18.2%)	<u>Aligned with organizational goals</u> (+)																								
16	Work experiences in non-US countries	Categorical	<table border="1"> <thead> <tr> <th>Country</th> <th>Count</th> <th>%</th> </tr> </thead> <tbody> <tr> <td>India</td> <td>273</td> <td>14.2%</td> </tr> <tr> <td>China</td> <td>168</td> <td>8.7%</td> </tr> <tr> <td>Canada</td> <td>77</td> <td>4.0%</td> </tr> <tr> <td>UK</td> <td>63</td> <td>3.3%</td> </tr> <tr> <td>Israel</td> <td>51</td> <td>2.6%</td> </tr> <tr> <td>Other</td> <td>383</td> <td>19.9%</td> </tr> <tr> <td>None</td> <td>911</td> <td>47.3%</td> </tr> </tbody> </table>	Country	Count	%	India	273	14.2%	China	168	8.7%	Canada	77	4.0%	UK	63	3.3%	Israel	51	2.6%	Other	383	19.9%	None	911	47.3%	31 attributes (+) 9 attributes (+) <u>Hardworking</u> (-) <u>Hardworking</u> (-) <u>Aligned with organizational goals</u> (-)
Country	Count	%																										
India	273	14.2%																										
China	168	8.7%																										
Canada	77	4.0%																										
UK	63	3.3%																										
Israel	51	2.6%																										
Other	383	19.9%																										
None	911	47.3%																										
17	Non-native English speaker	Yes/No Based on first language	Yes (926, 48.1%)	<u>Passionate</u> (+)																								
18	Type of customer	Categorical	Internal (791, 41.1%) External (270, 39.1%) Both (865, 31.75%)	<u>Persevering</u> (+)																								
19	Server-side software	Categorical	Client-side (562, 29.2%) Server-side (754, 39.1%) Both (610, 31.7%)	—																								
20	Developers worked with in past year	Numerical	Min (0), 25th (8), median (15), 75th (25), max (1,000)	—																								

Appendix: Weitere Ergebnisse - Unterschiede in den Abteilungen



Attribute	Software Engineering	Business Programs & Operations	Finance	Marketing	Program Management	Sales
Respondents	563	193	106	184	265	266
↔👤 Facilitates external communication	7.86				-0.47	
🌱👤 Grows talent	8.93		+0.40			
🌱👤 Guides the team	8.19					+0.50
↔👤 Clears path to execution	8.30				-0.38	
😊👤 Inspires the team	8.43	+0.54	+0.48	+0.54	+0.27	+0.56
↔👤 Drives alignment	8.33	+0.68	+0.46	+0.58	+0.45	+0.67
😊👤 Builds a relationship with team members	7.41					+0.65
🌱👤 Builds team culture	8.13	+0.59	+0.53	+0.64	+0.36	+0.88
🔧 Is technical/is domain expert	7.89	-1.43	-0.63	-0.82	-1.12	-0.99