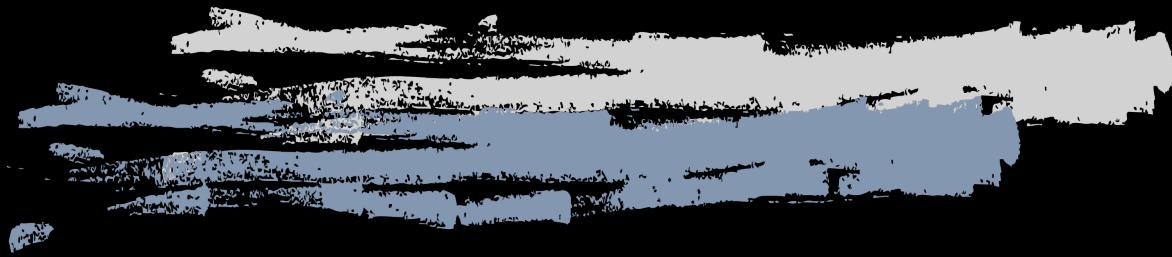
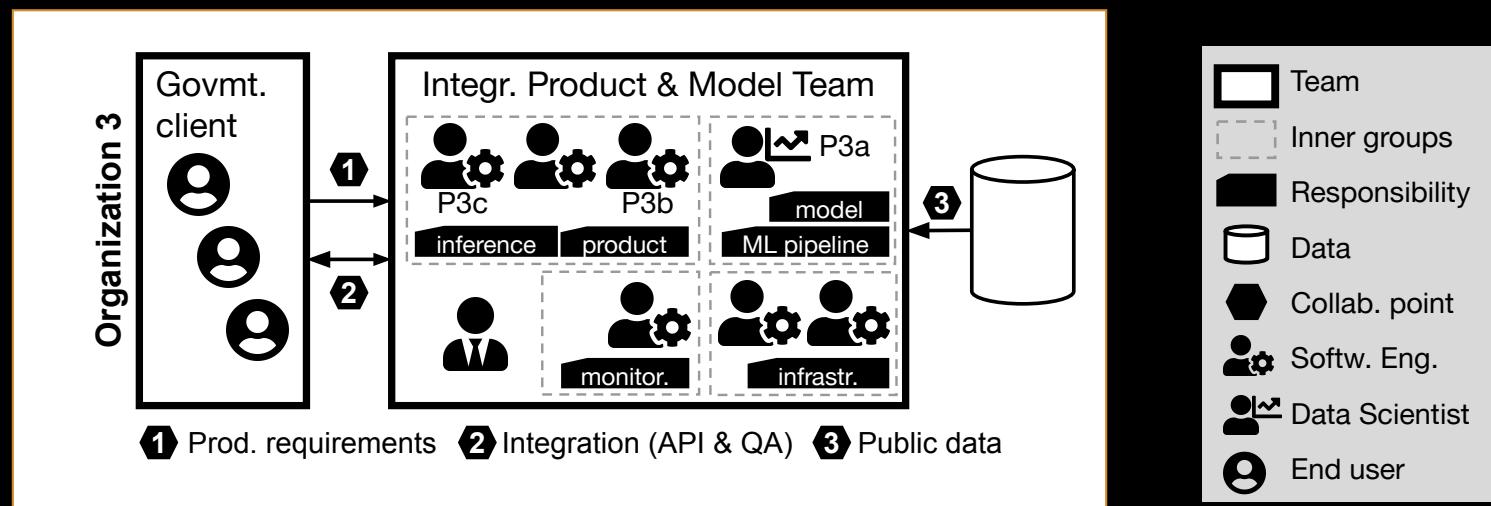


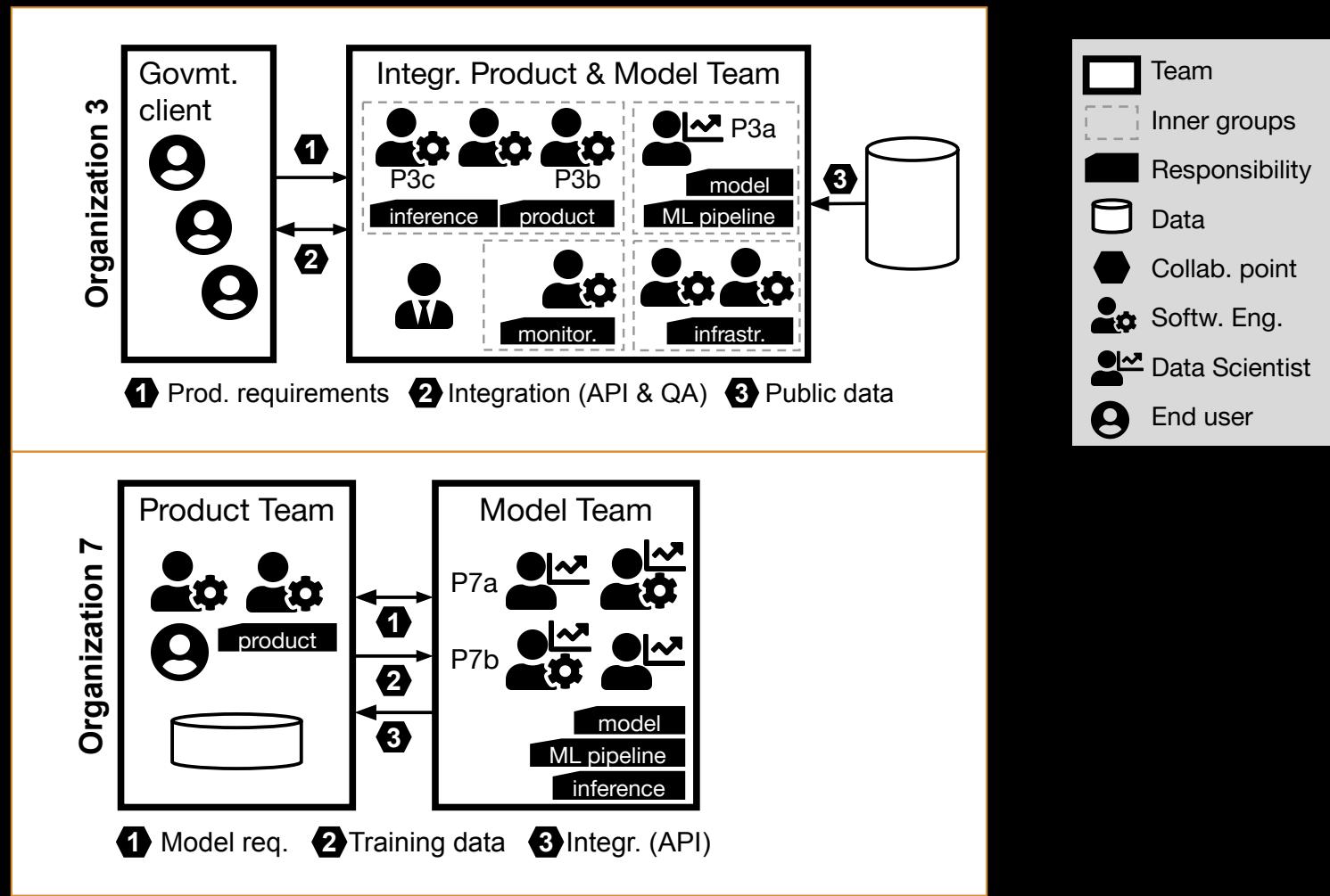
# Team and Collaboration



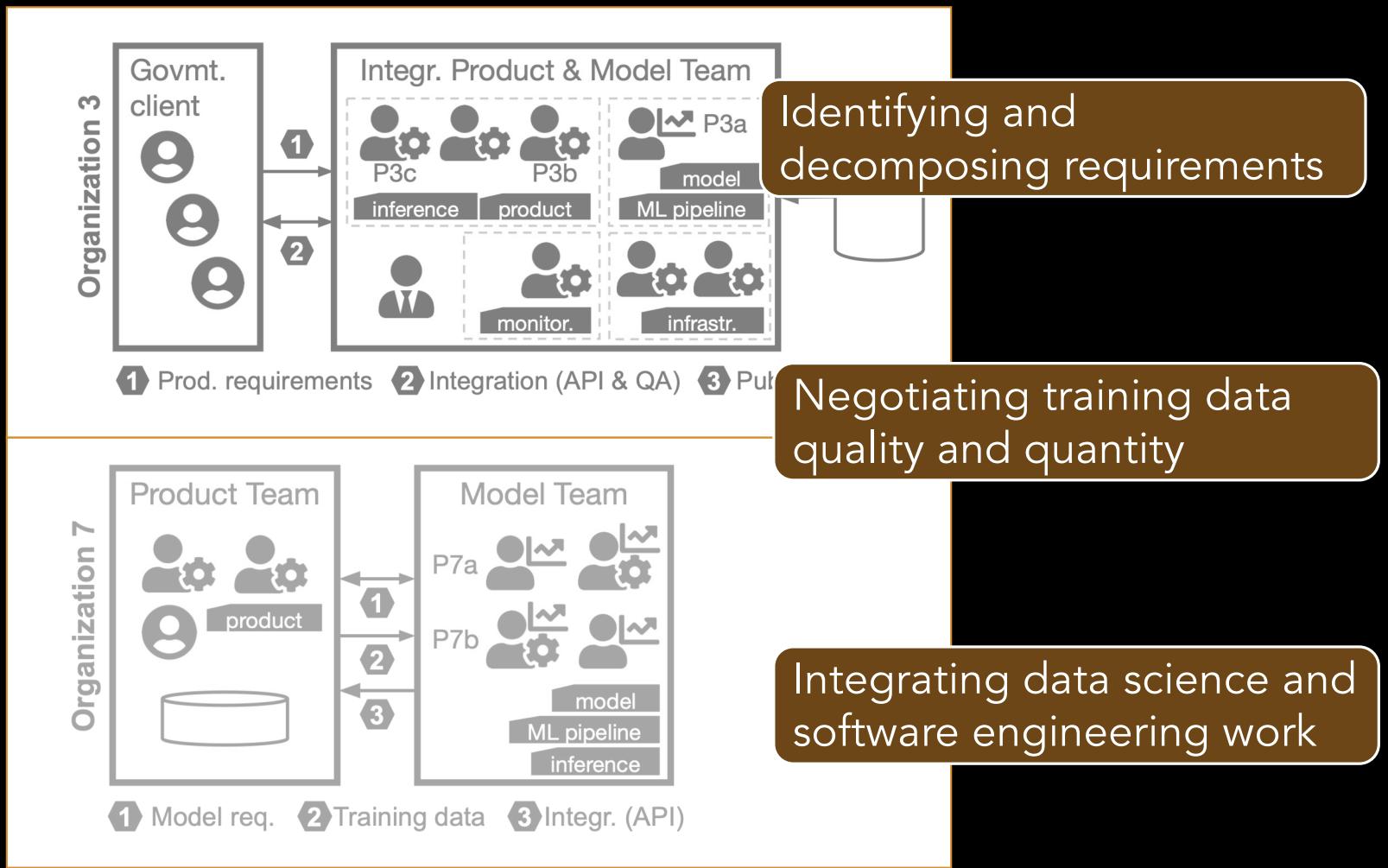
Jin Guo  
SOCS McGill University



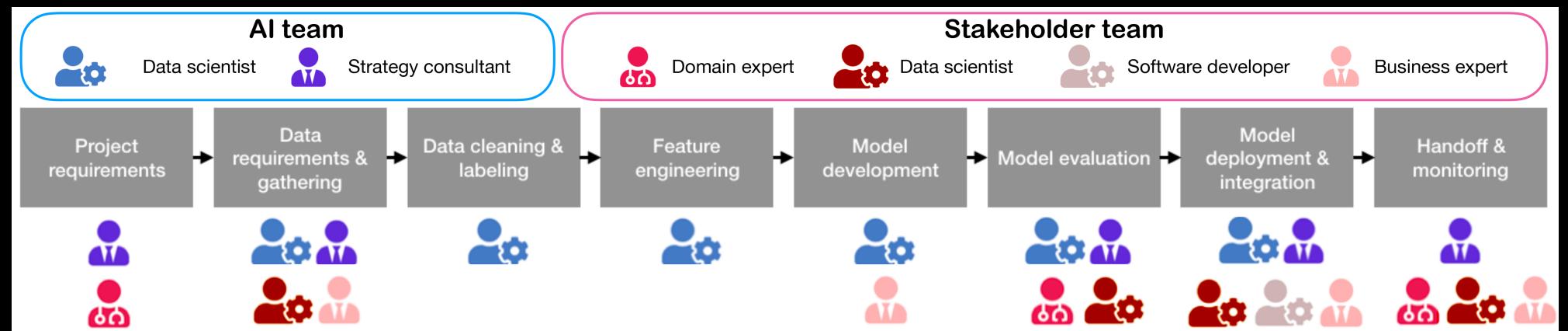
N. Nahar, S. Zhou, G. Lewis and C. Kästner, "Collaboration Challenges in Building ML-Enabled Systems: Communication, Documentation, Engineering, and Process," 2022 IEEE/ACM 44th International Conference on Software Engineering (ICSE), 2022, pp. 413-425



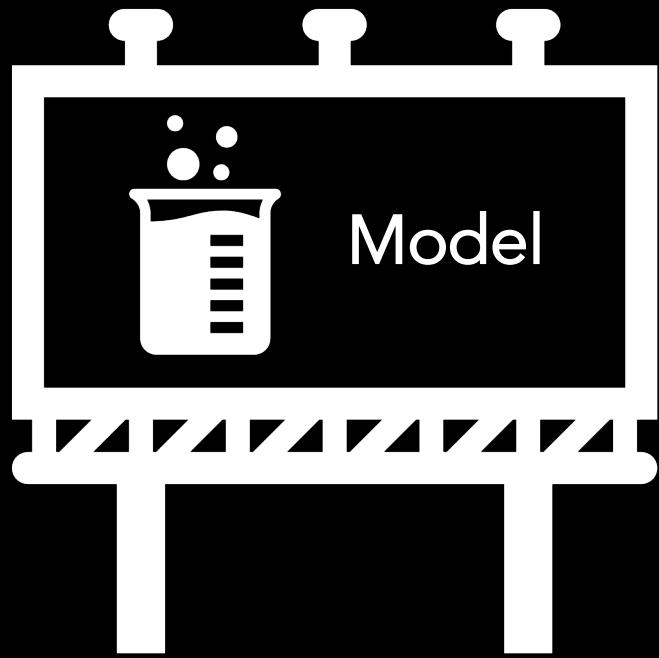
N. Nahar, S. Zhou, G. Lewis and C. Kästner, "Collaboration Challenges in Building ML-Enabled Systems: Communication, Documentation, Engineering, and Process," 2022 IEEE/ACM 44th International Conference on Software Engineering (ICSE), 2022, pp. 413-425



N. Nahar, S. Zhou, G. Lewis and C. Kästner, "Collaboration Challenges in Building ML-Enabled Systems: Communication, Documentation, Engineering, and Process," 2022 IEEE/ACM 44th International Conference on Software Engineering (ICSE), 2022, pp. 413-425



Piorkowski, D. et al. (2021) 'How AI Developers Overcome Communication Challenges in a Multidisciplinary Team: A Case Study', *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW1), pp. 1–25. doi:[10.1145/3449205](https://doi.org/10.1145/3449205).

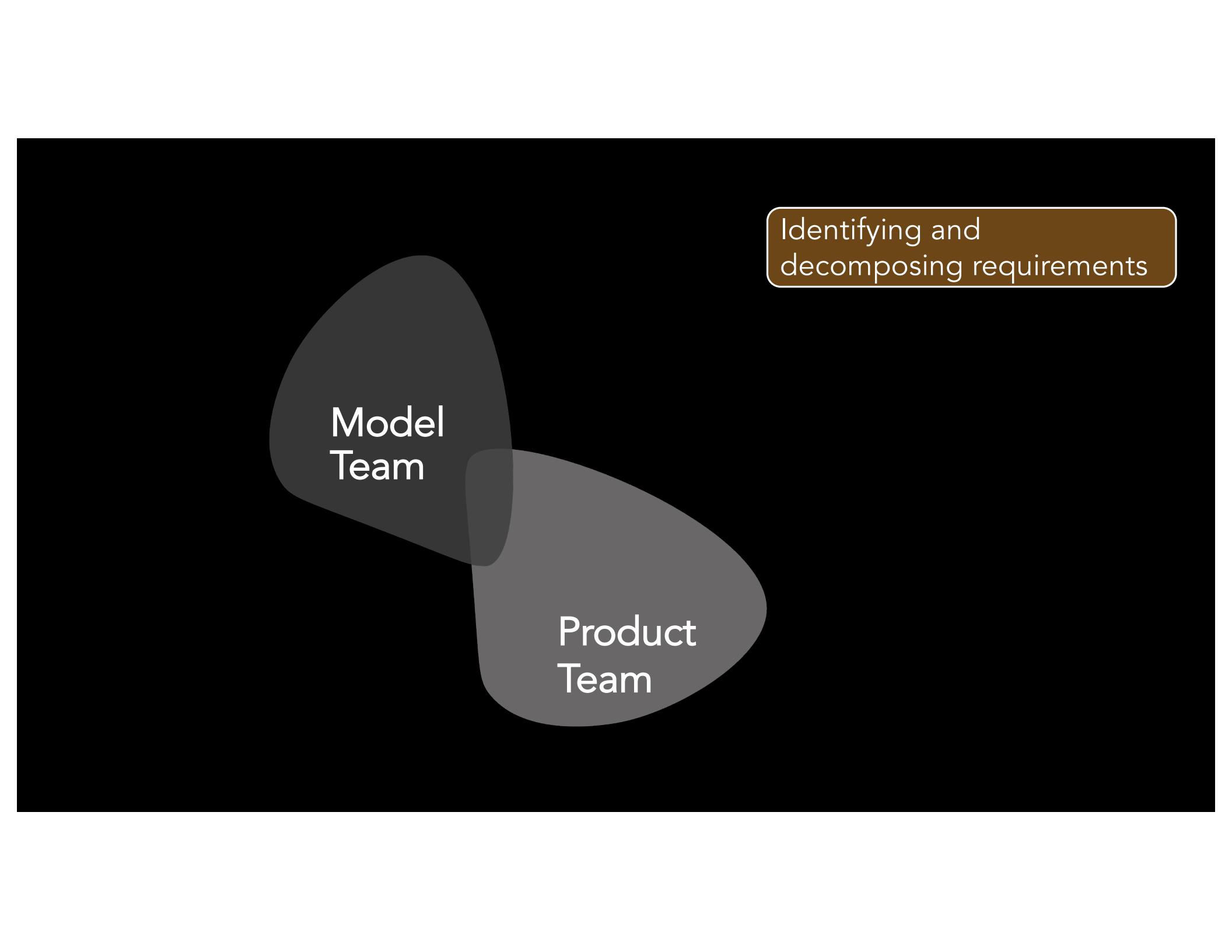


*Product-first*

*Model-first*

*Parallel*

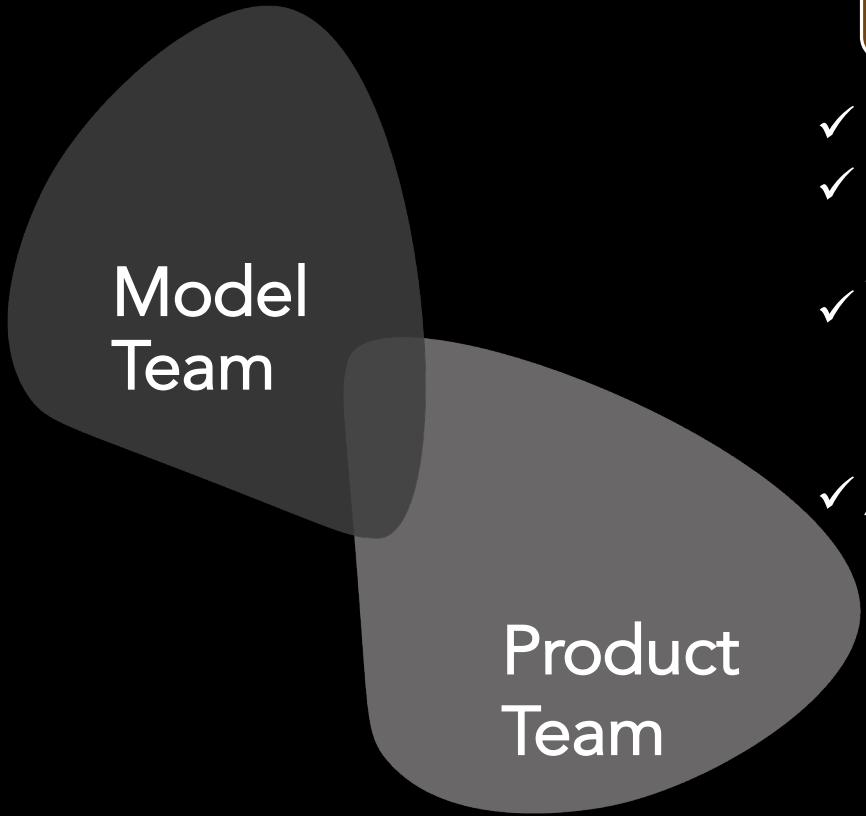
*Product*



Identifying and  
decomposing requirements

Model  
Team

Product  
Team

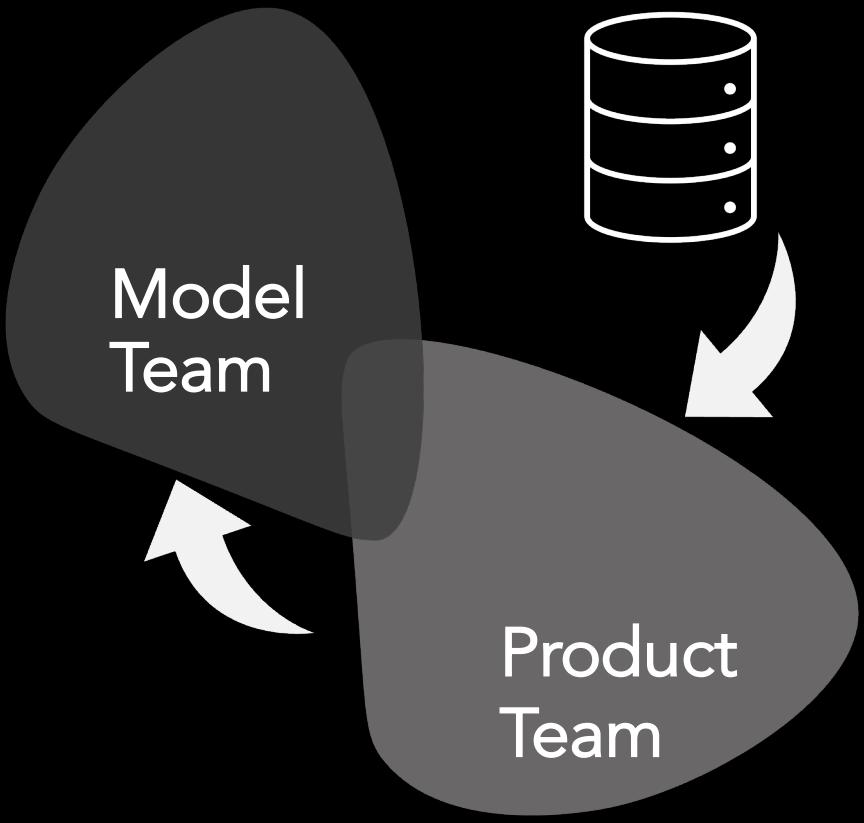


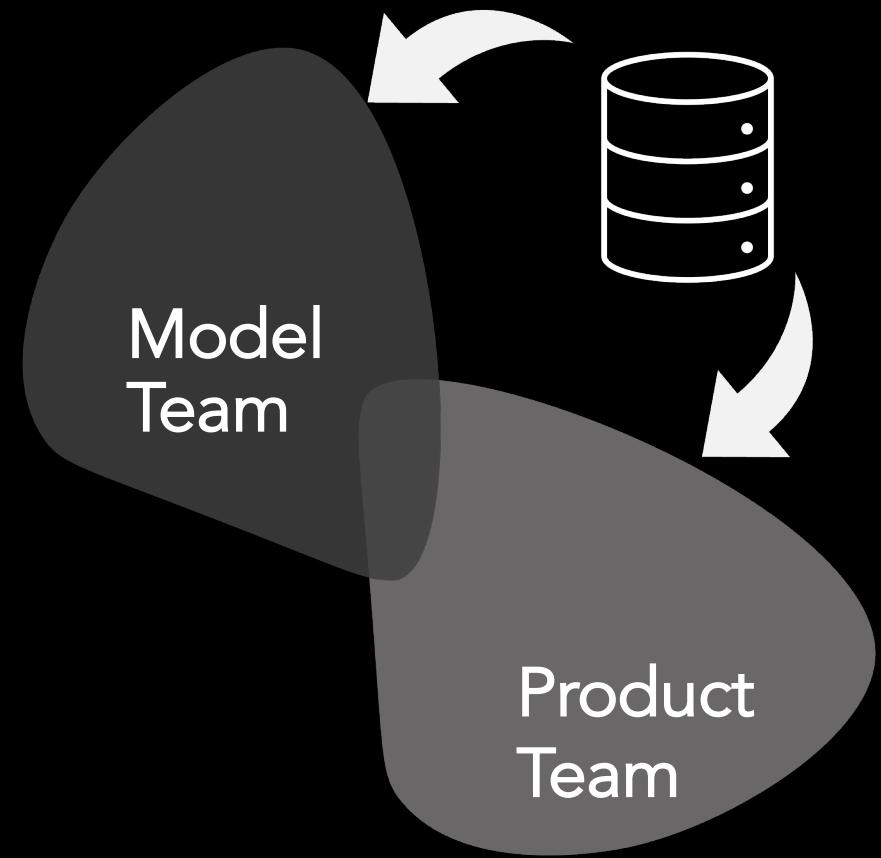
Model  
Team

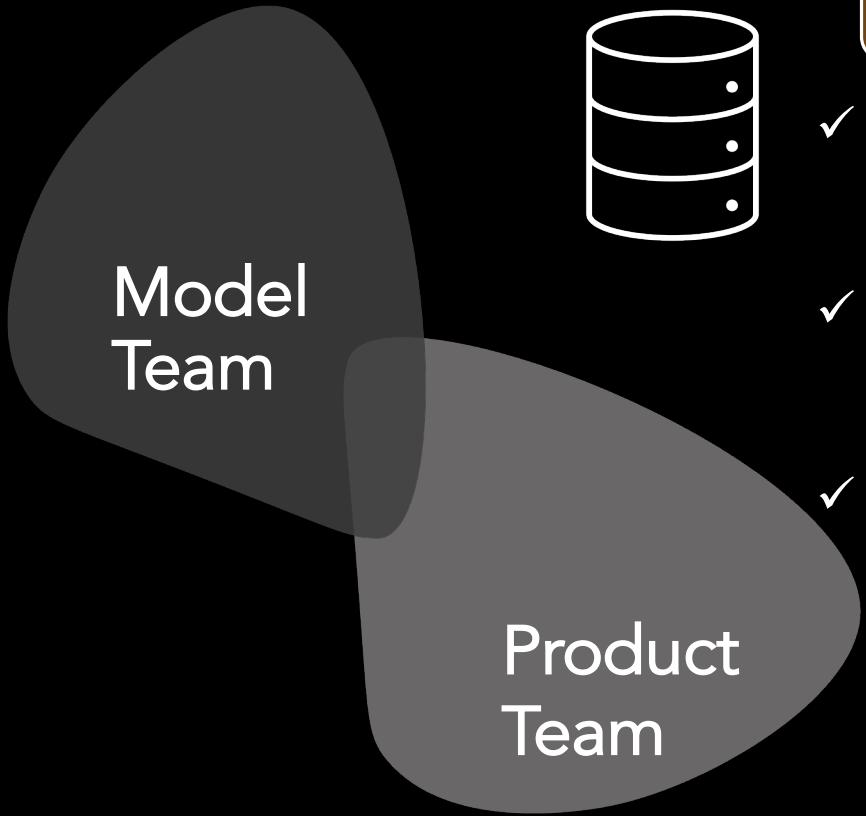
Product  
Team

### Identifying and decomposing requirements

- ✓ *Involving data scientists early*
- ✓ *Frequent interaction with the product team*
- ✓ *Clearly defining ML use cases and scope (with the help of other stakeholders)*
- ✓ *Adopt formal requirements documentation*

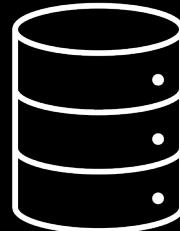






Model  
Team

Product  
Team



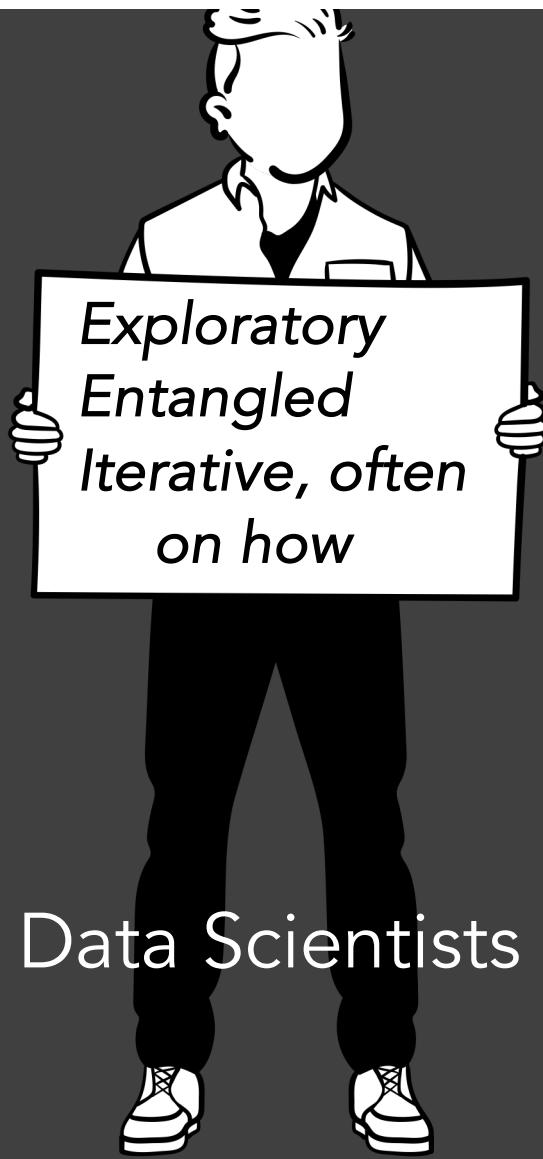
Negotiating training data quality and quantity

- ✓ *Plan budget for data collection, access to domain experts, or even a dedicated data team*
- ✓ *Written agreement on expectations of data quantify and quality*
- ✓ *Investigate on data validation and monitoring infrastructure*

Integrating data science and  
software engineering work

Model  
Team

Product  
Team



Data Scientists

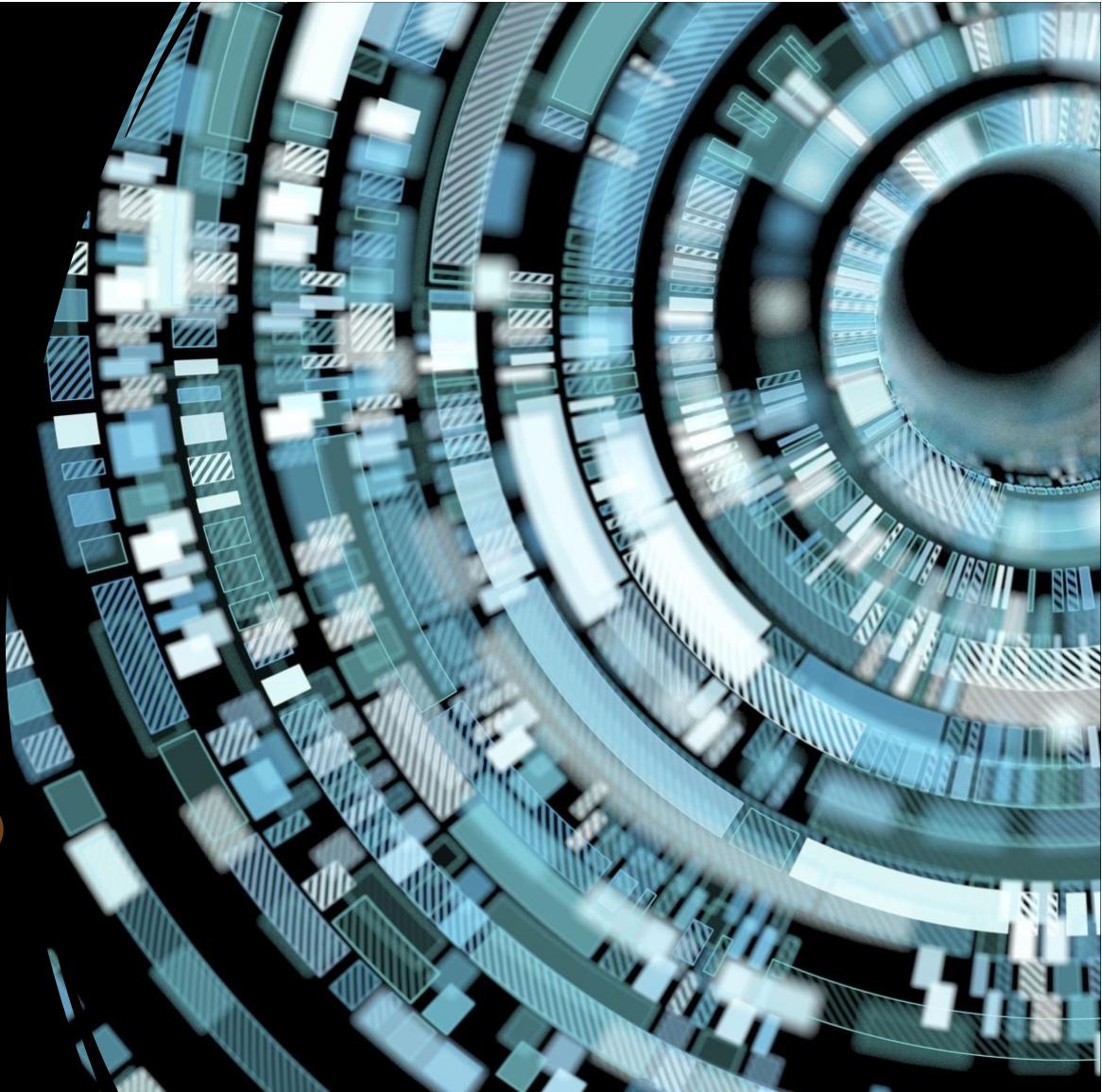
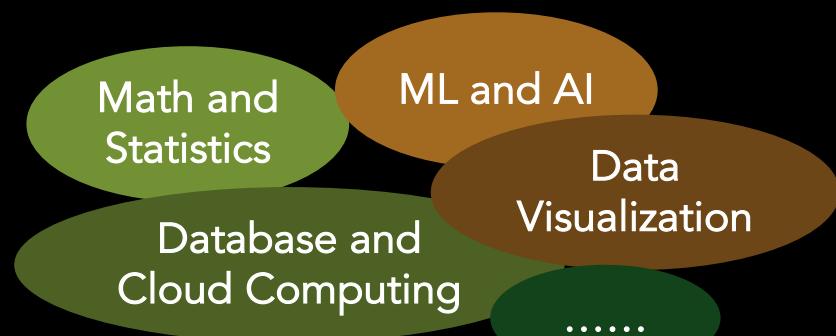


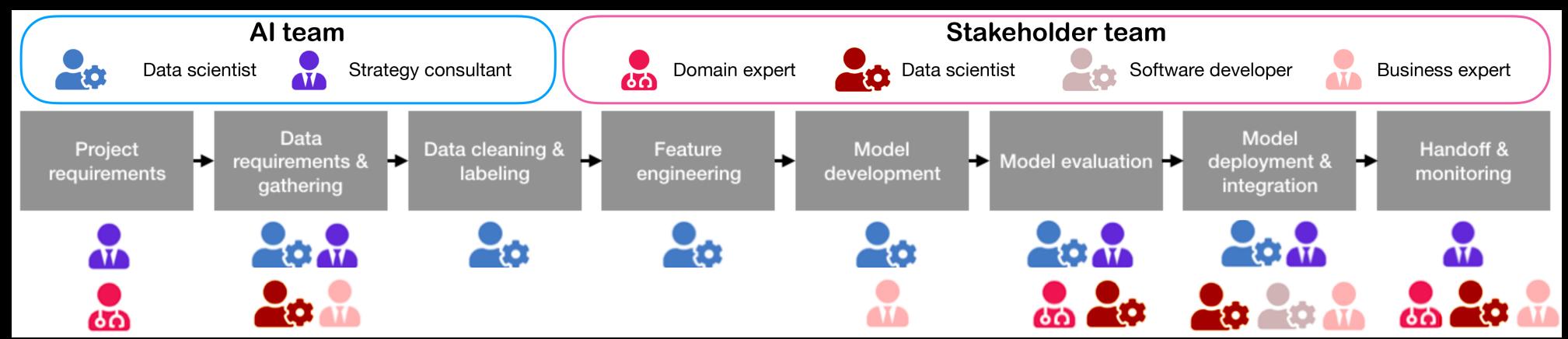
Software Engineers

# Data Scientists

---

leverage modern machine learning techniques to identify insights from data





Piorkowski, D. et al. (2021) 'How AI Developers Overcome Communication Challenges in a Multidisciplinary Team: A Case Study', *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW1), pp. 1–25. doi:[10.1145/3449205](https://doi.org/10.1145/3449205).

Table 2. Categories of data science tools and the number of times each tool was mentioned by respondents.

Tool Category	Tools Mentioned by Respondents (number of times mentioned)
asynchronous discussion	Slack (86), email (55), Microsoft Teams (1)
synchronous discussion	meeting (13), e-meeting (12), phone (1)
project management	Jira (8), ZenHub (2), Trello (1)
code management	GitHub (56), Git (5)
code	Python (42), R (9), Java (3), scripts (3)
code editor	Visual Studio Code (11), PyCharm (11), RStudio (8), Eclipse (1), Atom (1)
interactive code environment	Jupyter Notebook (66), SQL (6), terminal (4), Google Colab (4)
software package	Scikit-learn (3), Shiny App (2), Pandas (2)
analytics/visualization	SPSS (27), Watson Analytics (22), Cognos (7), ElasticSearch (4), Apache Spark (3), Graphana (2), Tableau (2), Logstash (2), Kibana (1)
spreadsheet	Microsoft Excel (22), spreadsheets (3), Google Sheets (1)
document editing	wiki (2), LaTeX (2), Microsoft Word (2), Dropbox Paper (2), Google Docs (1)
filesharing	Box (43), cloud (5), NFS (2), Dropbox (1), Filezilla (1)
presentation software	Microsoft Powerpoint (18), Prezi (1)

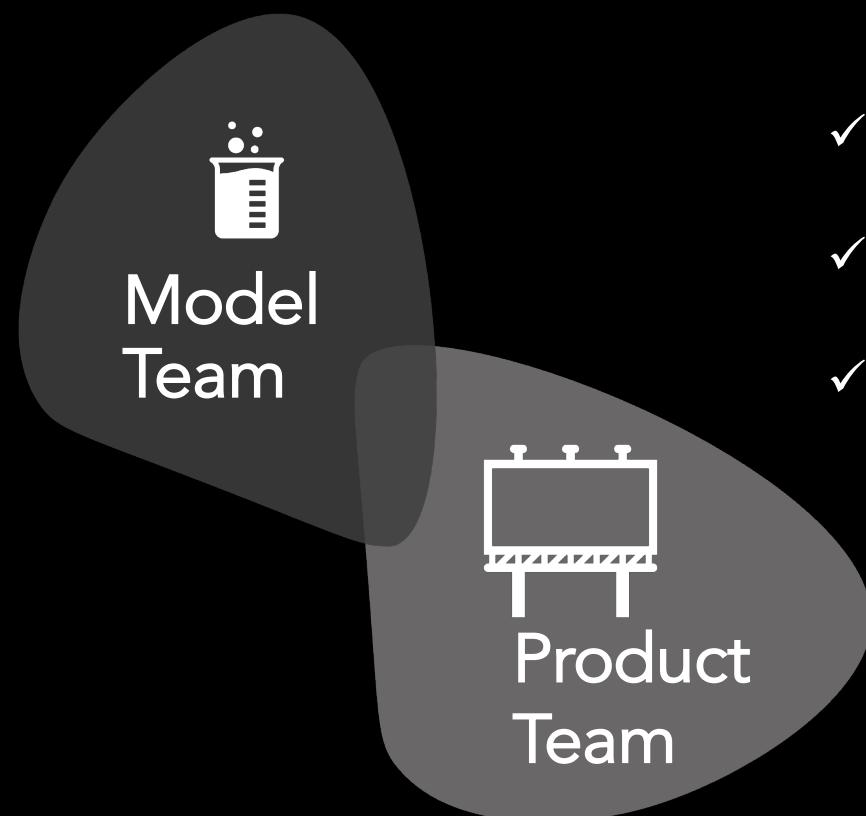
Note: **code** allows programmers to *write algorithms* for data science. **code editor** and **interactive code environment** provide a user experience for writing that code. **code management** is where the code may be stored and shared. By contrast, **analytics/visualization** provides “macro-level” tools that can invoke entire steps or modular actions in a data science pipeline.



Data Scientists



Software Engineers



## Integrating data science and software engineering work

- ✓ Define processes, responsibilities, and boundaries
- ✓ Recruit dedicated engineering support for model deployment
- ✓ Improve communication, and establish a team culture with mutual understanding and exchange

# Best Practices

- Education *about data science and SE practices*
- Documentation *a shared effort*
- Investment on engineering practices  
*data quality checks, deployment automation, testing in production, etc.*
- Integrated process life cycles  
*enable exploration and requirement refinement*

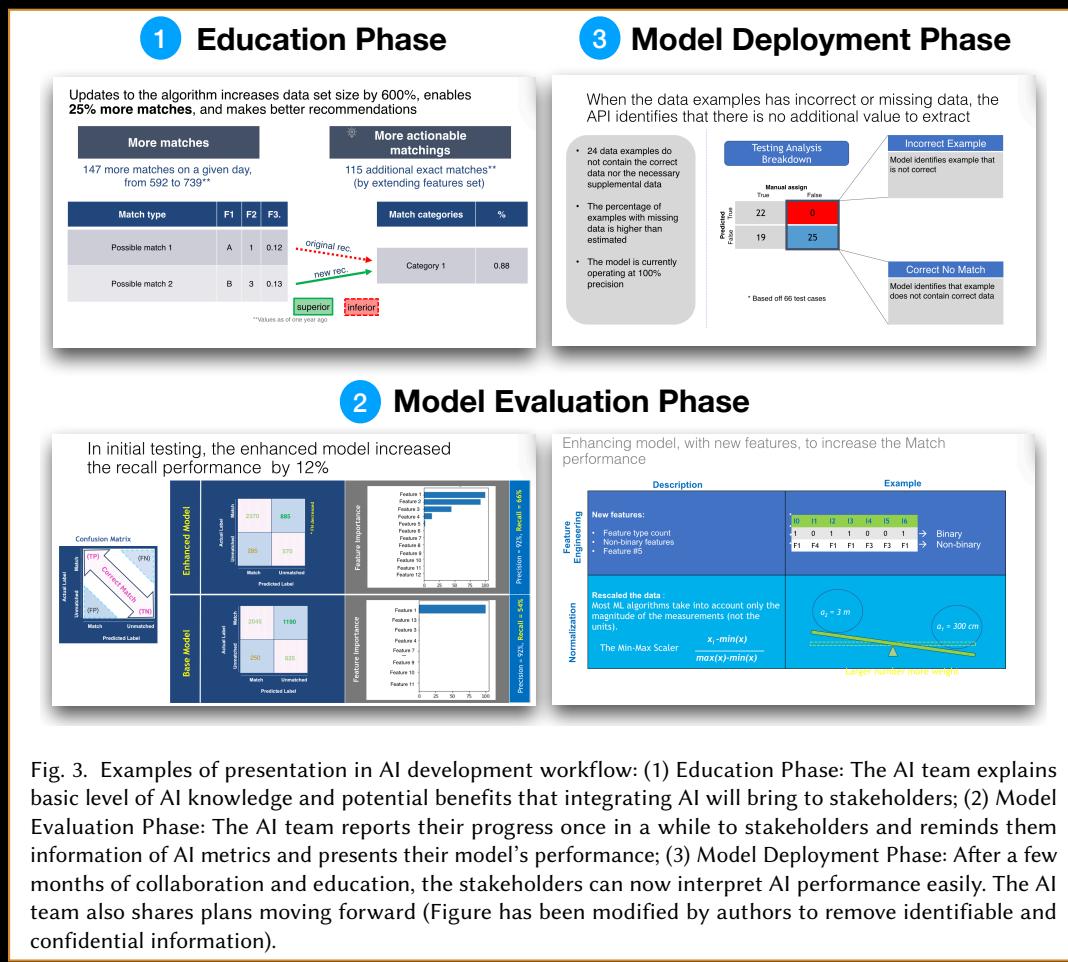


Fig. 3. Examples of presentation in AI development workflow: (1) Education Phase: The AI team explains basic level of AI knowledge and potential benefits that integrating AI will bring to stakeholders; (2) Model Evaluation Phase: The AI team reports their progress once in a while to stakeholders and reminds them information of AI metrics and presents their model's performance; (3) Model Deployment Phase: After a few months of collaboration and education, the stakeholders can now interpret AI performance easily. The AI team also shares plans moving forward (Figure has been modified by authors to remove identifiable and confidential information).

Piorkowski, D. et al. (2021) 'How AI Developers Overcome Communication Challenges in a Multidisciplinary Team: A Case Study', *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW1), pp. 1–25. doi:[10.1145/3449205](https://doi.org/10.1145/3449205).

On Thursday:

M1 Debriefing