

# Refocusing a company

from technical aspects to real customers



Matěj Procházka, Bohemia Interactive Simulations



You can act agile, be agile in a very unagile environment





# Bohemia Interactive Simulations - What do we do?

- Help train military and civilian personnel
- Provide tools for complex solutions integrating hardware and software
- Use game technologies for serious training
- Create AI to aid in training



# Complex business environment

- Extremely conservative customers
  - it takes up to a year to adopt a new version, the customer uses that version for years
- Strict and complex procurement laws and organizations
  - anti corruption laws
- Very formal agreements and requirements
  - same process as for building a fighter jet
- Great distance between actual users and the developer





# Complex business environment

- Extremely conservative customers - it takes up to a year to adopt a new version
- Strict and complex procurement laws and organizations - anti corruption laws
- Very formal agreements and requirements - same process as for building a fighter jet
- Great distance between actual users and the developer

Trainee

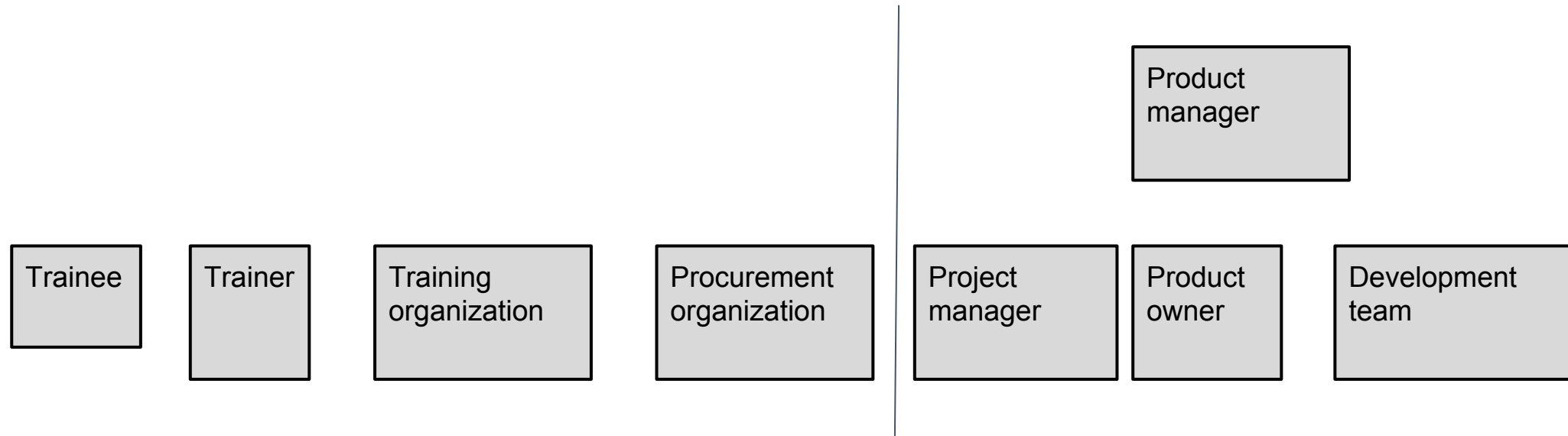
Trainer

Development  
team



# Complex business environment

- Extremely conservative customers - it takes up to a year to adopt a new version
- Strict and complex procurement laws and organizations - anti corruption laws
- Very formal agreements and requirements - same process as for building a fighter jet
- Great distance between actual users and the developer





# The tale of two projects

Two projects, one in 2016, one in 2018

- Same customer
- Same contract type (Fixed time, fixed budget)
- comparable effort (~4 team months)
- 2016
  - far above the budget
  - customer satisfaction low
- 2018
  - within the budget
  - customer satisfaction high



# The tale of two projects

Project 1 (2016)

Project 2 (2018)

Final delivery
ProjMan Interactions
Within budget
Sprints & demos
delivery





# The tale of two projects

Project 1 (2016)

Project 2 (2018)

	Project 1 (2016)	Project 2 (2018)	
Final delivery	6 months late	3 months early	from Period of performance to MVP thinking
ProjMan Interactions			
Within budget			
Sprints & demos			
delivery			



# The tale of two projects

Project 1 (2016)

Project 2 (2018)

	Project 1 (2016)	Project 2 (2018)	
Final delivery	6 months late	3 months early	from Period of performance to MVP thinking
ProjMan Interactions	terrible	great	more experienced PM, more business context to the team
Within budget			
Sprints & demos			
delivery			



# The tale of two projects

Project 1 (2016)

Project 2 (2018)

Final delivery	6 months late	3 months early	from Period of performance to MVP thinking
ProjMan Interactions	terrible	great	more experienced PM, more business context to the team
Within budget	No!	yes	established as KPI, visibility to the PO
Sprints & demos			
delivery			



# The tale of two projects

Project 1 (2016)

Project 2 (2018)

Final delivery	6 months late	3 months early	from Period of performance to MVP thinking
ProjMan Interactions	terrible	great	more experienced PM, more business context to the team
Within budget	No!	yes	established as KPI, visibility to the PO
Sprints & demos	yes	yes	teams using standard builds to conduct demos, not local builds
delivery			



# The tale of two projects

Project 1 (2016)

Project 2 (2018)

	Project 1 (2016)	Project 2 (2018)	
Final delivery	6 months late	3 months early	from Period of performance to MVP thinking
ProjMan Interactions	terrible	great	more experienced PM, more business context to the team
Within budget	No!	yes	established as KPI, visibility to the PO
Sprints & demos	yes	yes	teams using standard builds to conduct demos, not local builds
delivery	one final delivery	multiple deliveries	we found customer employees interested in the builds





# Bringing about change

- People with vision
- Agile transformation middle up & down
- Setting clear KPIs - establishing what is important
- Creating a better relationship with key people (project managers, customer)
- Clarifying expectations of quality - ask for forgiveness, not permission - less gates, more reviews
- Requirements management, confirmation of use cases



# Conclusion

You can act agile, be agile in a very unagile environment