Managing Managerless Processes

Fred George

fredgeorge@acm.org

@fgeorge52

Managing Managerless Processes

Fred George

fredgeorge@acm.org

🐦 *@fgeorge52*

goto; conference

Click 'Rate Session' to rate session and ask questions.

follow us @gotochgo

Fred George

- Programmer
  - Since 1968 (Basic)
  - 65,000 hours experience
  - 70+ languages
  - Computer Science Degree 1973
- Manager
  - 17 years IBM
  - Business degree, MIT Sloan School 1986
  - Product Owner, IBM
  - VP, ThoughtWorks
  - Co-founder, Outpace (Silicon Valley)
  - Senior Advisor to 3 tech companies
- Technologist
  - Computer networks – 70's
  - Token Ring LAN – 80's
  - GUI's – late 80's
  - OO – late 80's
  - Agile – late 90's
  - MicroServices – mid-2000's

Fred George

Consulting Roles

- Change Agent
- Disruptor
- “Hand grenade I am throwing into development”
  - CTO describing Fred to his Vice President

Why Managerless?

Y2k Agile

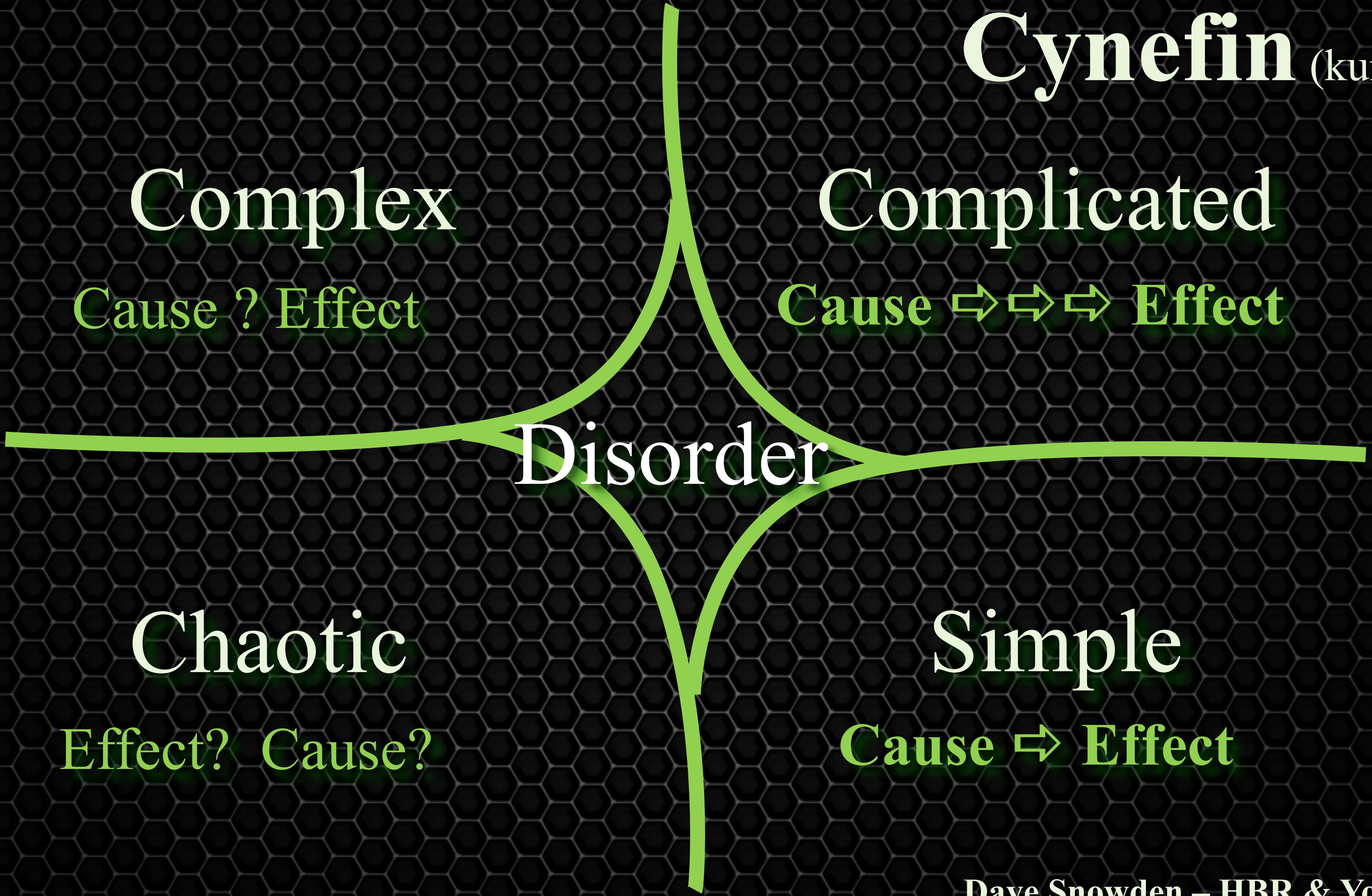
Requirements Uncertainty

Uncertainty

Certainty

Fuzzy vs. Traditional Problems

Cynefin (kun-**ev**-in)



Complex

Cause ? Effect

Complicated

Cause ⇨⇨⇨ Effect

Disorder

Chaotic

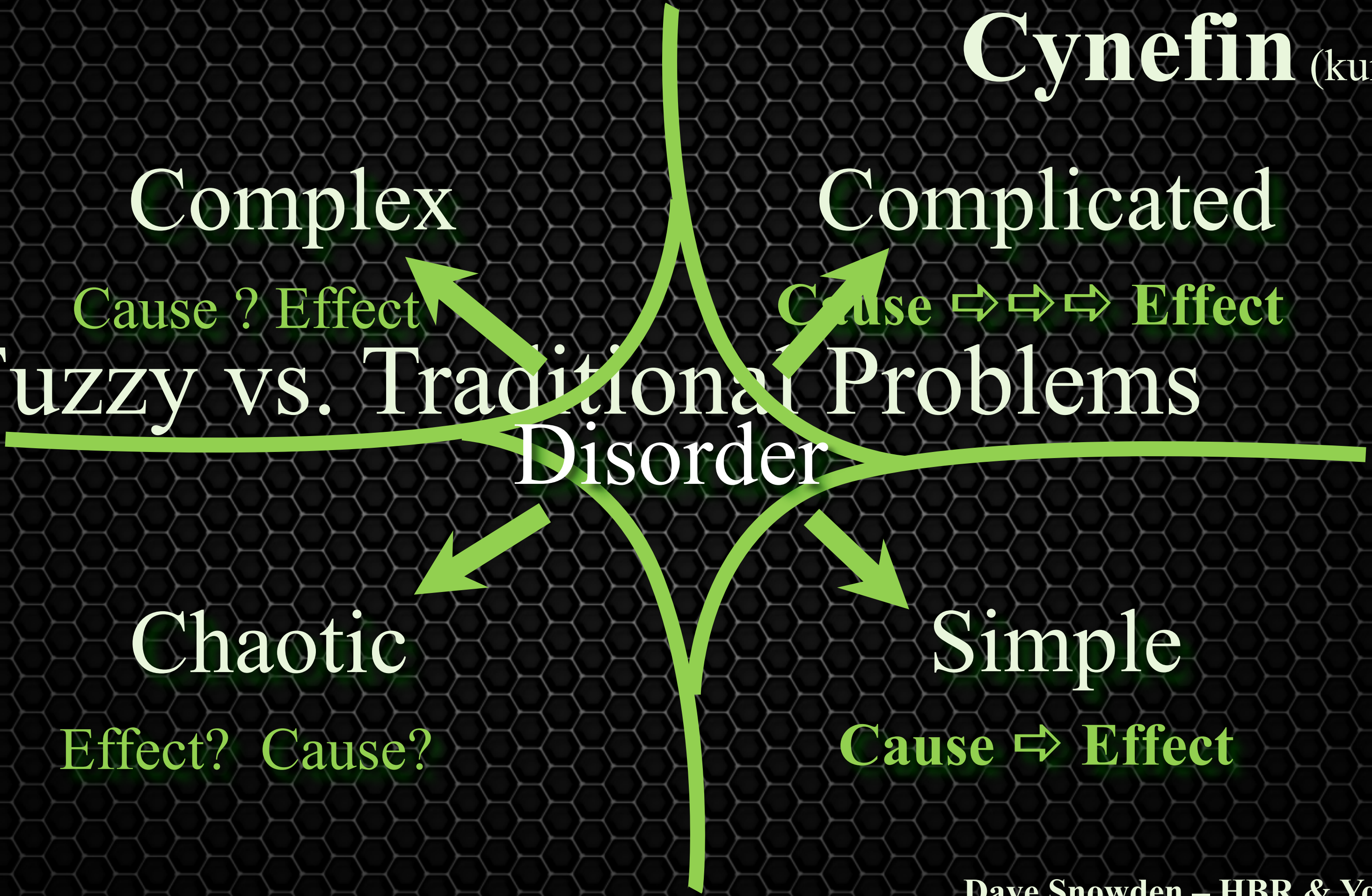
Effect? Cause?

Simple

Cause ⇨ Effect

Dave Snowden – HBR & YouTube

Fuzzy vs. Traditional Problems



Cynefin (kun-**ev**-in)

Complex

Cause ? Effect

Complicated

Cause ⇨⇨⇨ Effect

Disorder

Chaotic

Effect? Cause?

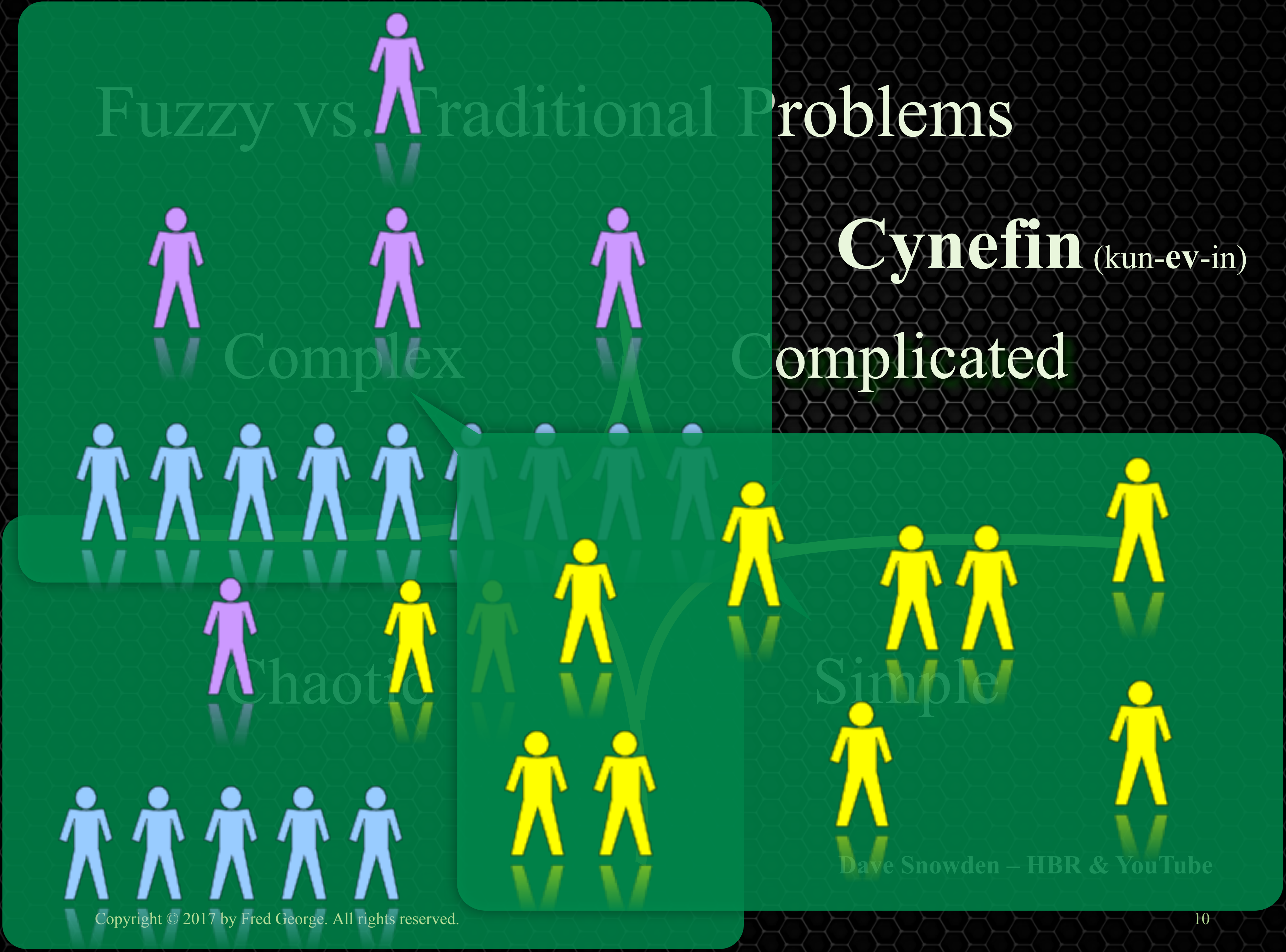
Simple

Cause ⇨ Effect

Dave Snowden – HBR & YouTube

Fuzzy vs. Traditional Problems

Cynefin (kun-ev-in)



Complex

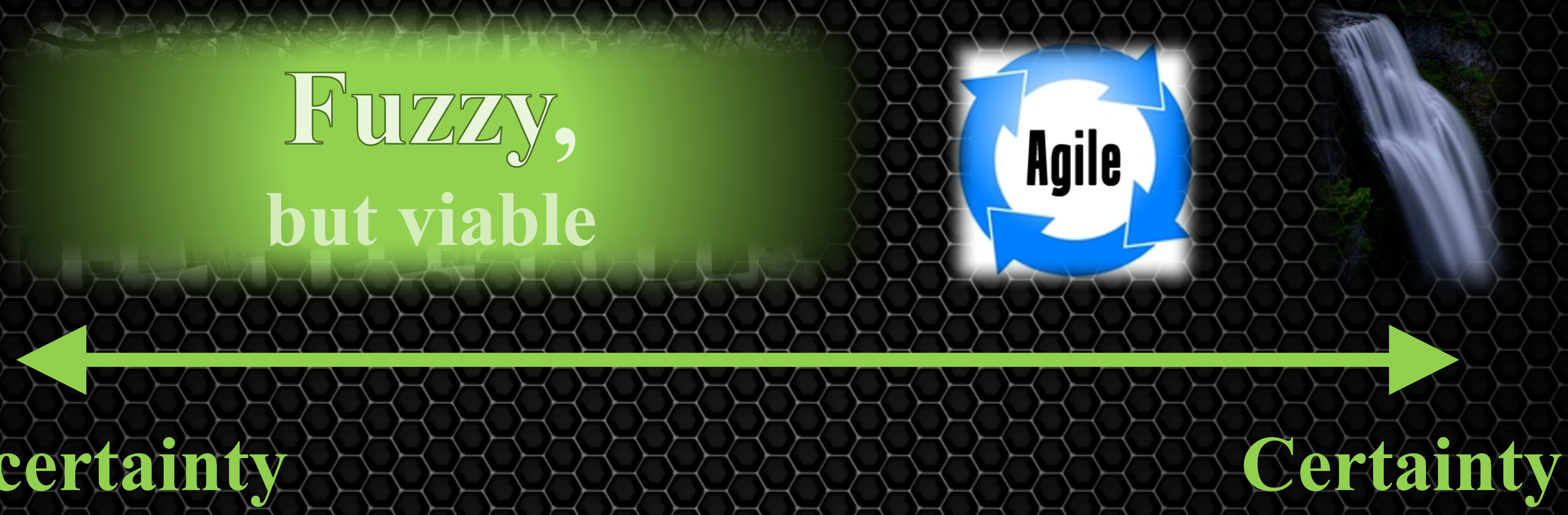
Complicated

Chaotic

Simple

Dave Snowden – HBR & YouTube

Requirements Uncertainty



Fuzzy,
but viable

Uncertainty

Certainty

Age of Agents

- **Assistive technologies abound**
- **Google Home, Siri, Alexa, Cortana, …**
- **Real innovation not voice, but *interaction* of *backing services***
- **Welcome to *fuzzy* world!**

Industry "Hot" Topics

Go Faster

One Hacker Way

Lean startup

Cassandra

Programmer anarchy

MicroServices

Docker

Dev/Ops

Cloud

Full-stack developer

Agile

Event bus

MVP (minimum viable product

No-SQL

XP Values: Cornerstones of Morale

- **Feedback**
- **Communication**
- **Simplicity**
- **Courage**
- **Respect**

Agile Practices and Roles

- Stand ups
- Story narratives
- Retrospectives
- Estimates
- Iterations
- Mandatory pairing

- Unit tests
- Acceptance tests
- Refactoring
- Patterns
- Continuous integration

- BA
- Developer
- Architect
- SCRUM master
- QA
- Manager
- Customer

Fuzzy

~~Agile~~ Practices and Roles

- ~~Stand ups~~
- ~~Story narratives~~
- ~~Retrospectives~~
- ~~Estimates~~
- ~~Iterations~~
- ~~Mandatory~~ pairing

- ~~Unit tests~~
- ~~Acceptance tests~~
- ~~Refactoring~~
- ~~Patterns~~
- ~~Continuous integration~~

- ~~BA~~
- Developer
- ~~Architect~~
- ~~SCRUM master~~
- ~~QA~~
- ~~Manager~~
- Customer

	Manager	Tech Lead	Designer	Developer	Tester	Analyst
Dave	✓				✓	✓
Fred	✓	✓	✓	✓		✓
George		✓		✓		
~~Jeremy~~						~~✓~~ ✖
Matt (cust)			✓		✓	✓
Paul		✓		✓		
Sarah				✓		

Actions to Complement Fuzzy

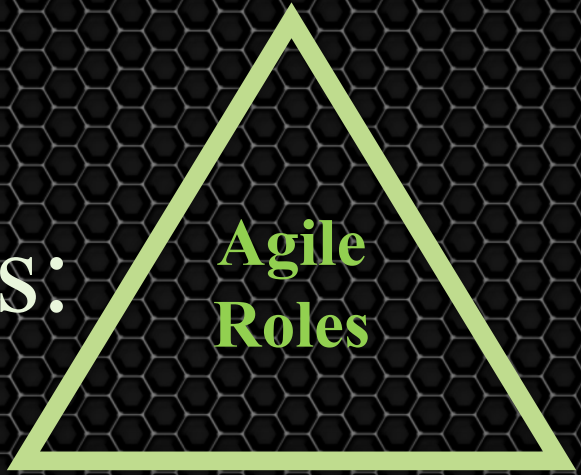
- Fixed price project (budget constrained)
- Co-located team
- Iterations == Daily Standups
- Discarded most original backlog (J-I-T)
- Result: Overdelivered by 20% on-time

Inhibitors for Managerless

Over-Specialization

- Theory:
  - Specialist are more productive
- Practice:
  - Overhead of communication is under-estimated
  - Unbalanced workload creates delays

Agile Roles:



Agile Roles

management

project manager

iteration manager

business

customer

QA

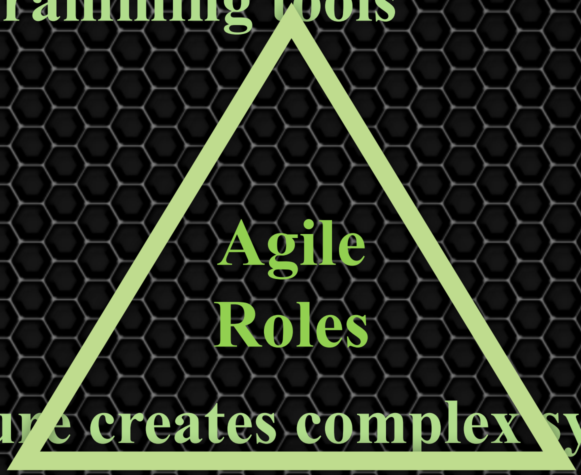
BA

development

developer

Fate of Roles: QA

- QA tools are Programming tools
  - Selenium
  - Cucumber
- Service Architecture creates complex systems
  - Need architecture skills to properly test
- Shift toward monitoring over acceptance testing



Agile Roles

management

project manager

iteration manager

business

customer

BA

QA

development

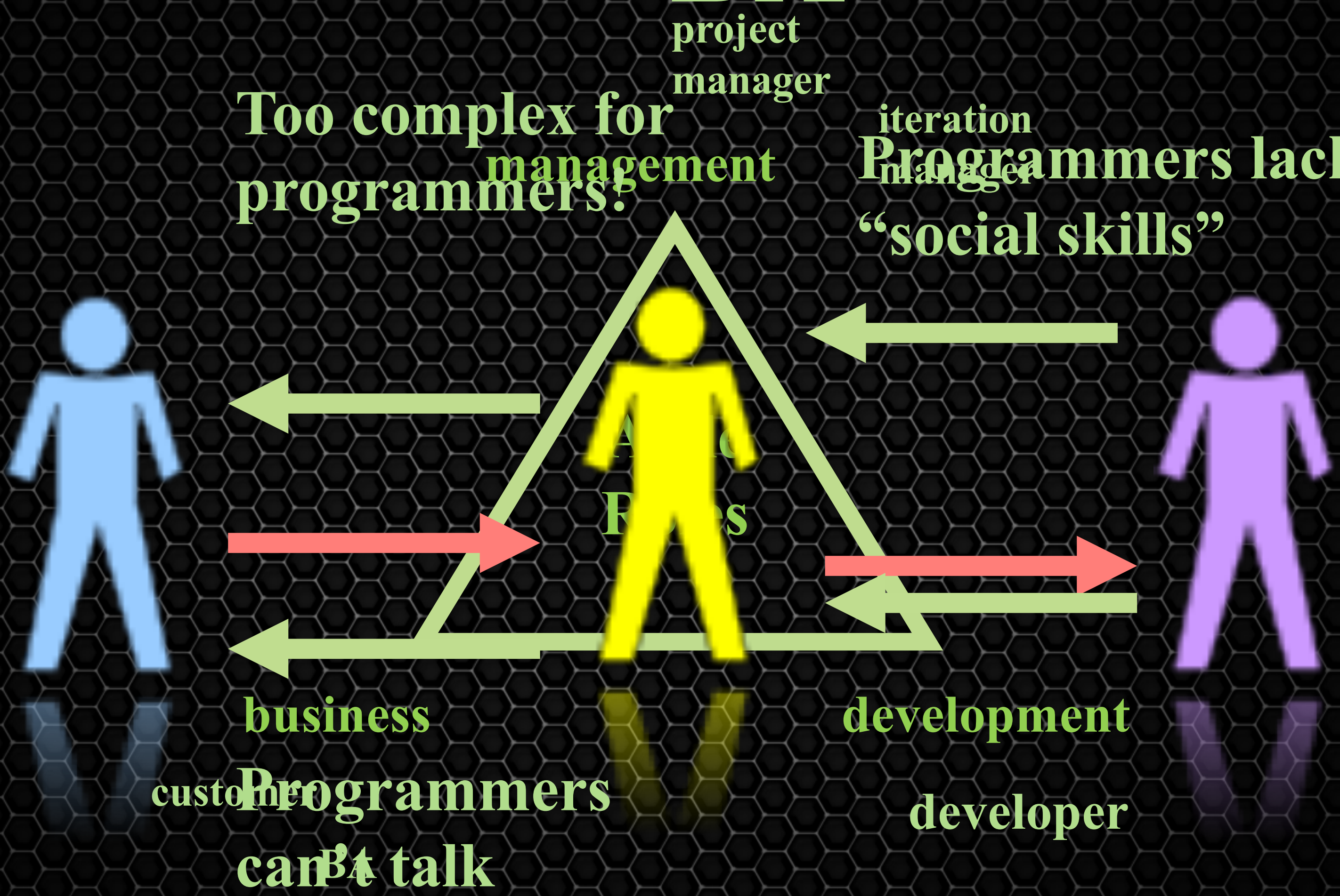
developer

Fate of Roles: BA

Too complex for programmers!

Programmers lack “social skills”

Programmers can’t talk



project manager

management

iteration manager

Agile Roles

business

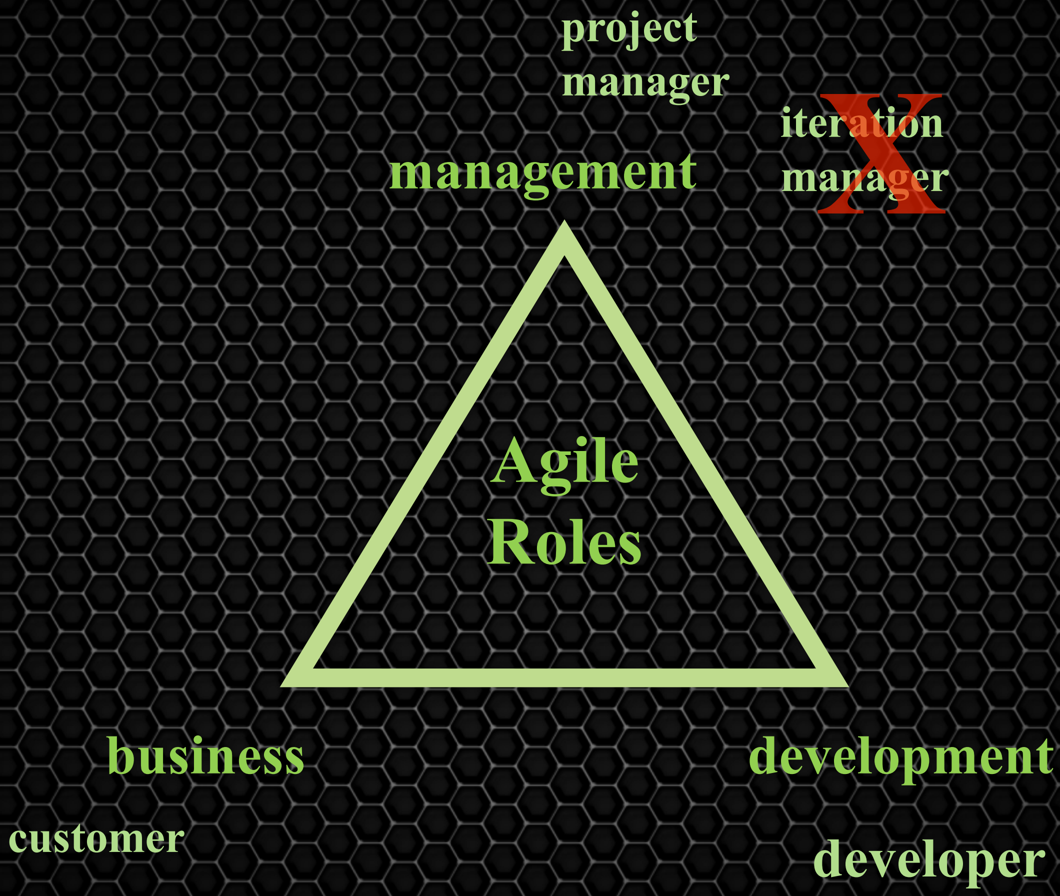
development

customer

BA

developer

Fate of Roles:



project manager

management

~~iteration manager~~

Agile Roles

business

customer

development

developer

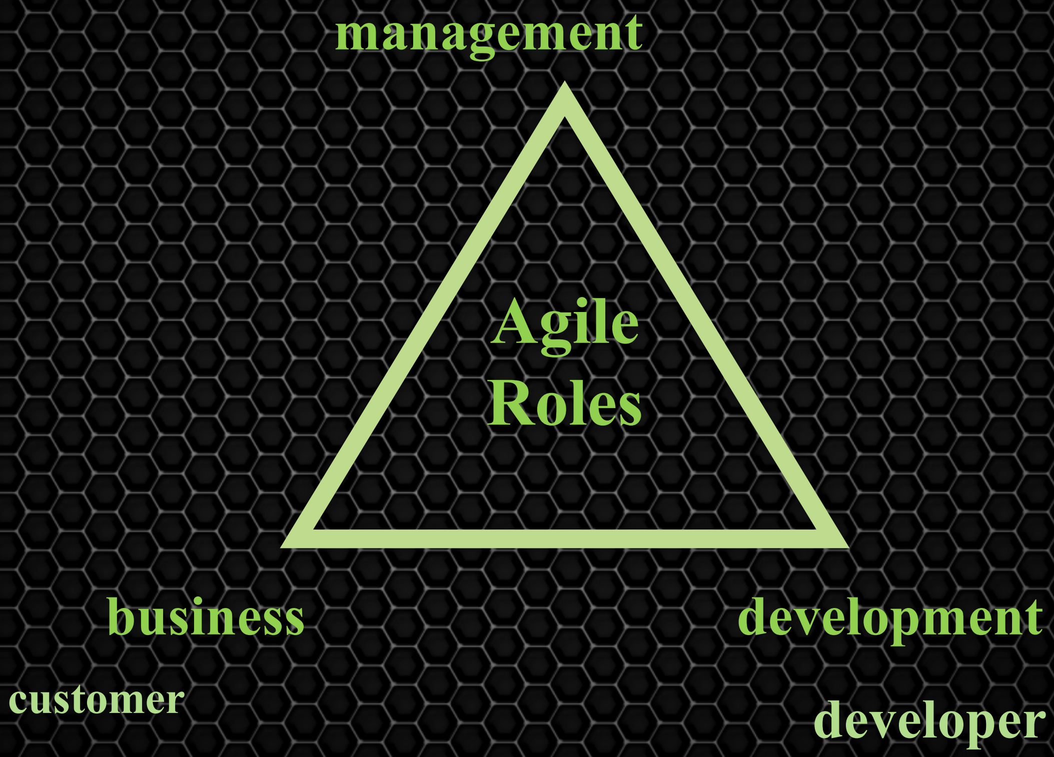
Fate of Roles:

Power-Hungry Boss

Manager

[illegible]

Anarchy Roles



management

Agile
Roles

business

customer

development

developer

Creating Managerless

Trust

- **Programmers care**
- **Leaders will emerge**
- **Problems will arise and get solved**
- **Team composition will (should) morph**

Delivery Focus

- **All roles support delivery**
- **Complete focus on cycle time**
- **Constantly reiterated from the top**

Management Inversion Enforced

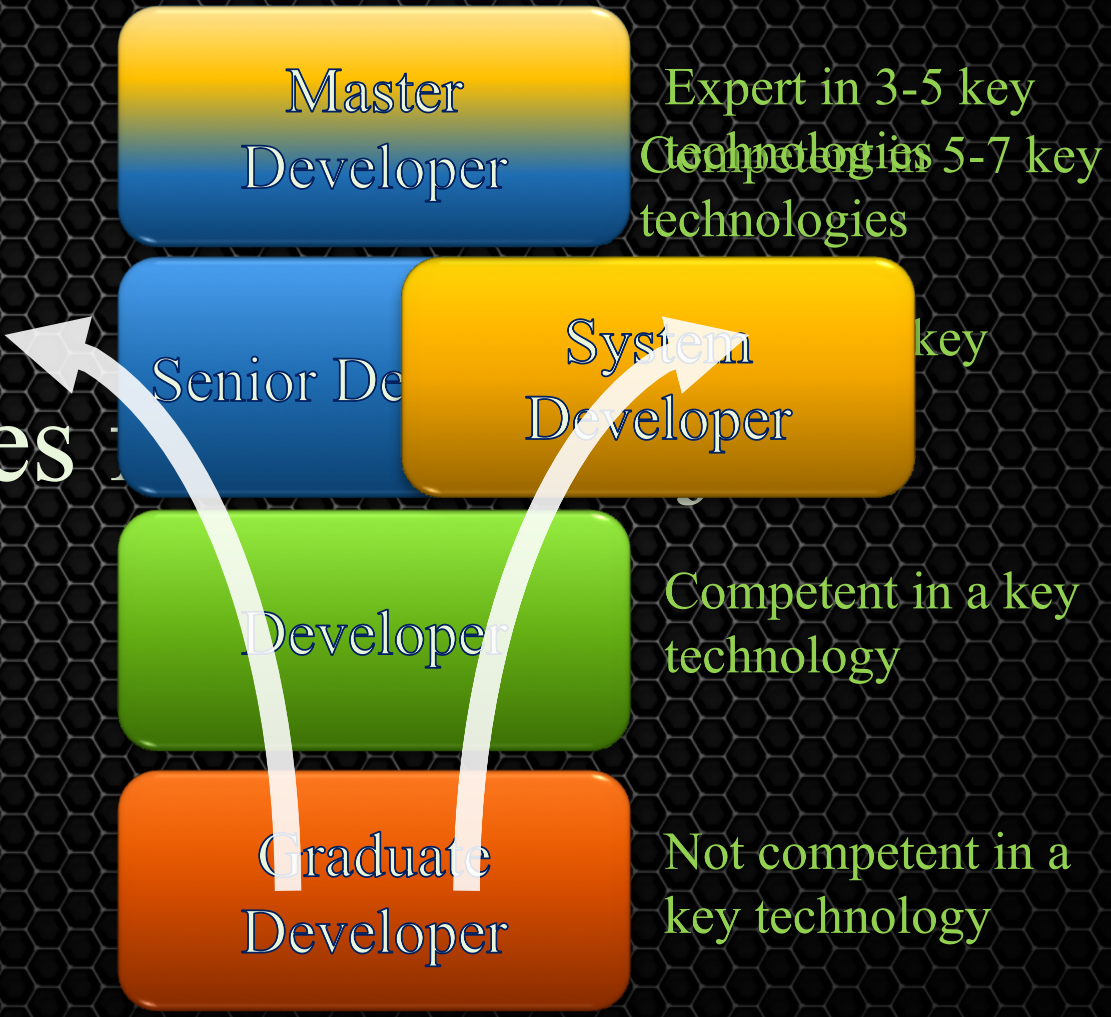
- **Concierge (ombudsman) key**
- **React to team needs (however trivial)**
- **Respect team decisions**
    - **Leadership, ambassadors, membership**
- **Continual insight to team inner workings**
    - **Not necessarily “fun” role**

Case Study:

Specialization Institutionalized with Titles

- •50 **IT professionals**
- •25+ **titles**
- •0 **people understanding projects**

New Titles



Expert in 3-5 key technologies

Competent in 5-7 key technologies

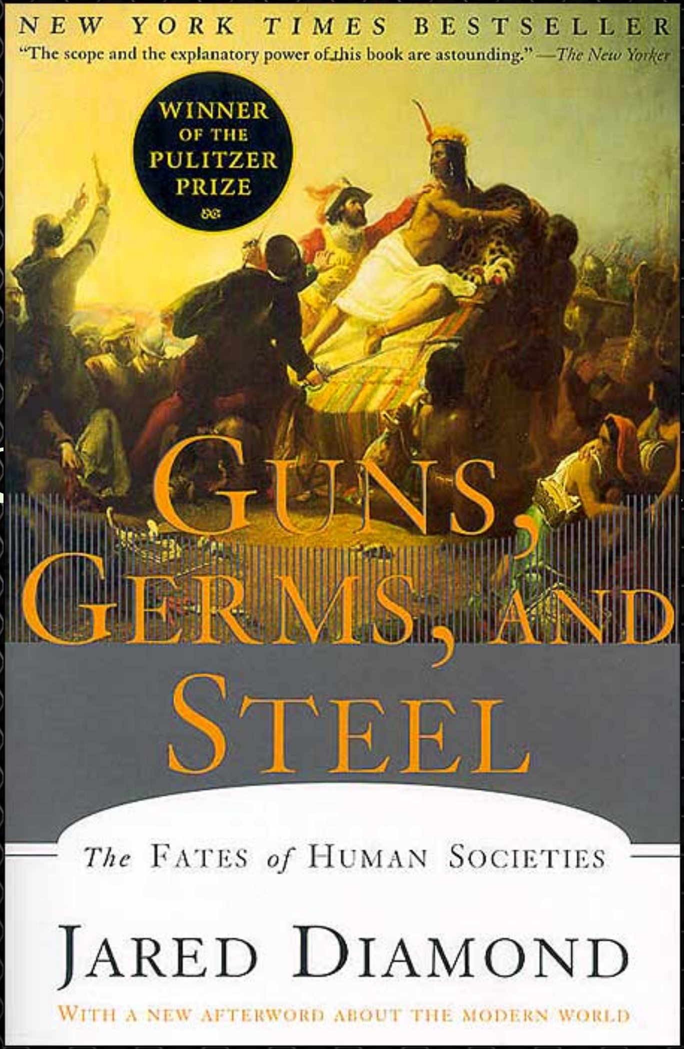
key

Competent in a key technology

Not competent in a key technology

*Kent Beck, XP Explained

Non-dedic ip



NEW YORK TIMES BESTSELLER

"The scope and the explanatory power of this book are astounding." —*The New Yorker*

WINNER OF THE PULITZER PRIZE

GUNS, GERMS, AND STEEL

The FATES *of* HUMAN SOCIETIES

JARED DIAMOND

WITH A NEW AFTERWORD ABOUT THE MODERN WORLD

Ambassador Role

- **Ambassador represents team to outside groups**
- **Member of the team**
    - **Rotation allowed**
- **Not empowered to commit**
    - **Just the ambassador after all!**
- **Speaking skills valuable; EQ very valuable**

Bring Work to the Team…

Not musical chairs for each small project

Appraisals & Coaching

- **At conflict in employee's mind**
- **Appraisal**
  - **Measuring contribution (vs. Others)**
  - **Effects promotions (so what?) and salary**
  - **Perception vs. data conflict**
- **Coaching**
  - **Honest, critical is valuable**
  - **Shortcomings are the focus**

Appraisals in Mangerless Teams

- Define relevant metrics
- All individual metrics are subjective
- Trust team allocation of “success” to individuals
- Allow variation across groups
  - Including one-time experiments

Coaching in Mangerless Teams

- **Separate timing from Appraisals**
- **Allow variation (including experiments)**
- **Mentor model:**
  - **Mutual selection of relationship**
  - **Cap individual mentors as necessary**
  - **Outside mentors fine (preferred?)**
  - **Firewall information from mentors to appraisal process**

Anti-Patterns in Mangerless Teams

- **Status reports**
  - **Beware engineering desire to answer queries**
  - **“How are you going to use this information?”**
- **Sign-offs**
- **“Just try again to work with John…”**
- **Over-reaction to failures / mistakes**
  - **“If we just had more planning meetings…”**

Requirements Uncertainty

Fuzzy,
but viable

Uncertainty

- **Idea-focused features**
- **Full-stack developers**
- **Fast failure systems**
- **MicroServices**
- **Event-based architecture**
- **Continuous deployment**

- **Stories**
- **Specialists**
- **TDD**
- **Acceptance tests**
- **Migration scripts**

Certainty

goto; conference

Please

Remember to rate this session

Thank you!

follow us @gotochgo

Managing Managerless Processes

Fred George

fredgeorge@acm.org

@fgeorge52