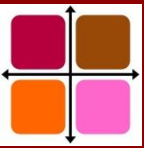


# Vergangenheit, Gegenwart und Zukunft der Agilität

Karlsruher Entwicklertage, 22.05.2017

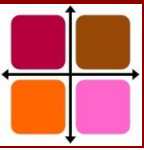
*Dr. Andrea Herrmann*

*[herrmann@herrmann-ehrich.de](mailto:herrmann@herrmann-ehrich.de)*

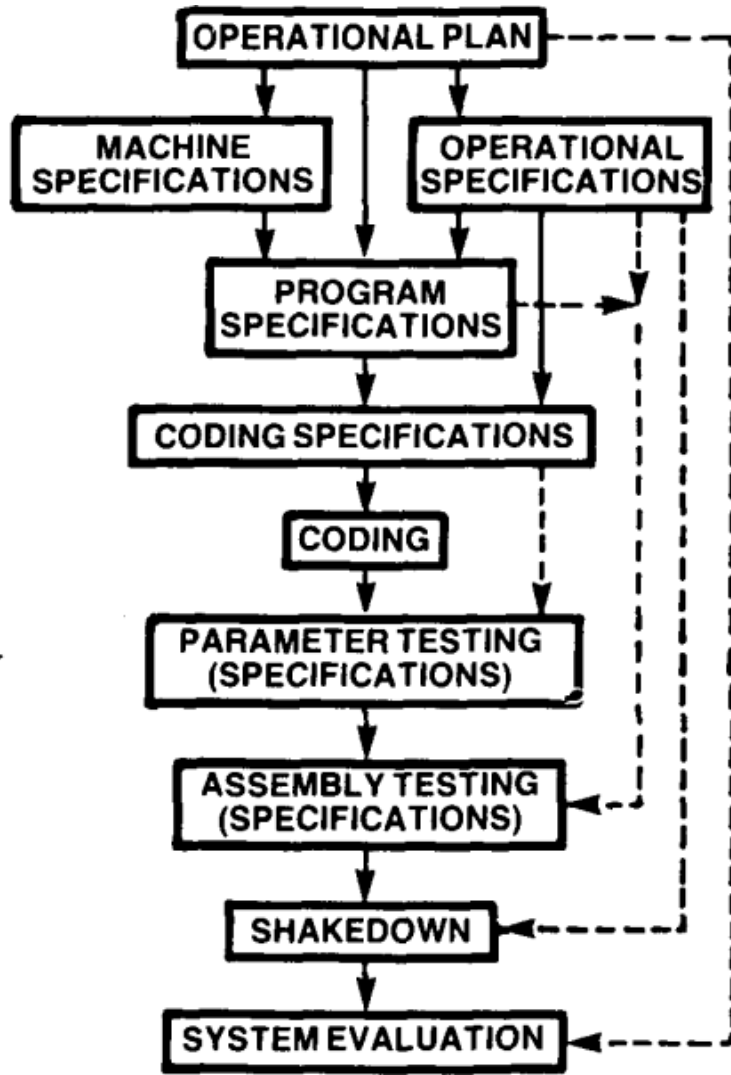


# Vergangenheit: historische Forschung

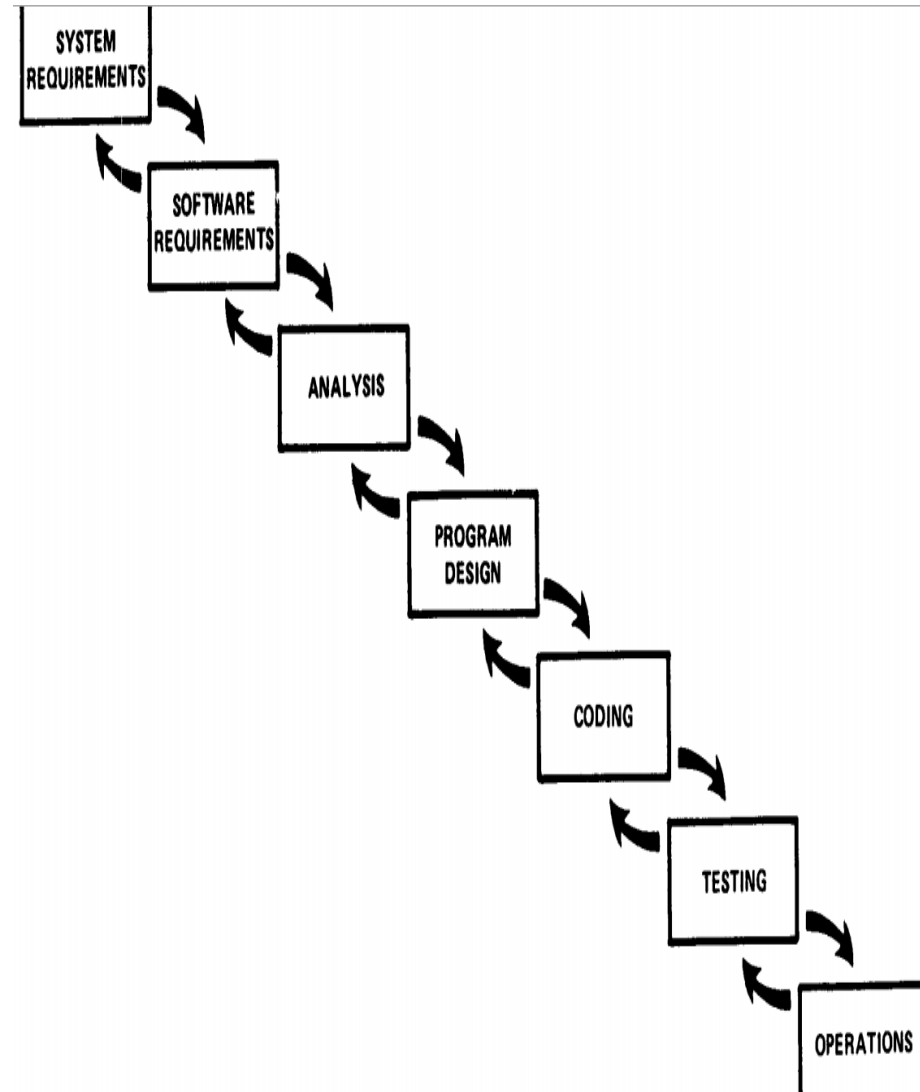




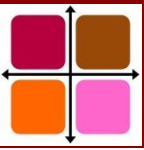
# Definition: Wasserfall / Phasenmodell



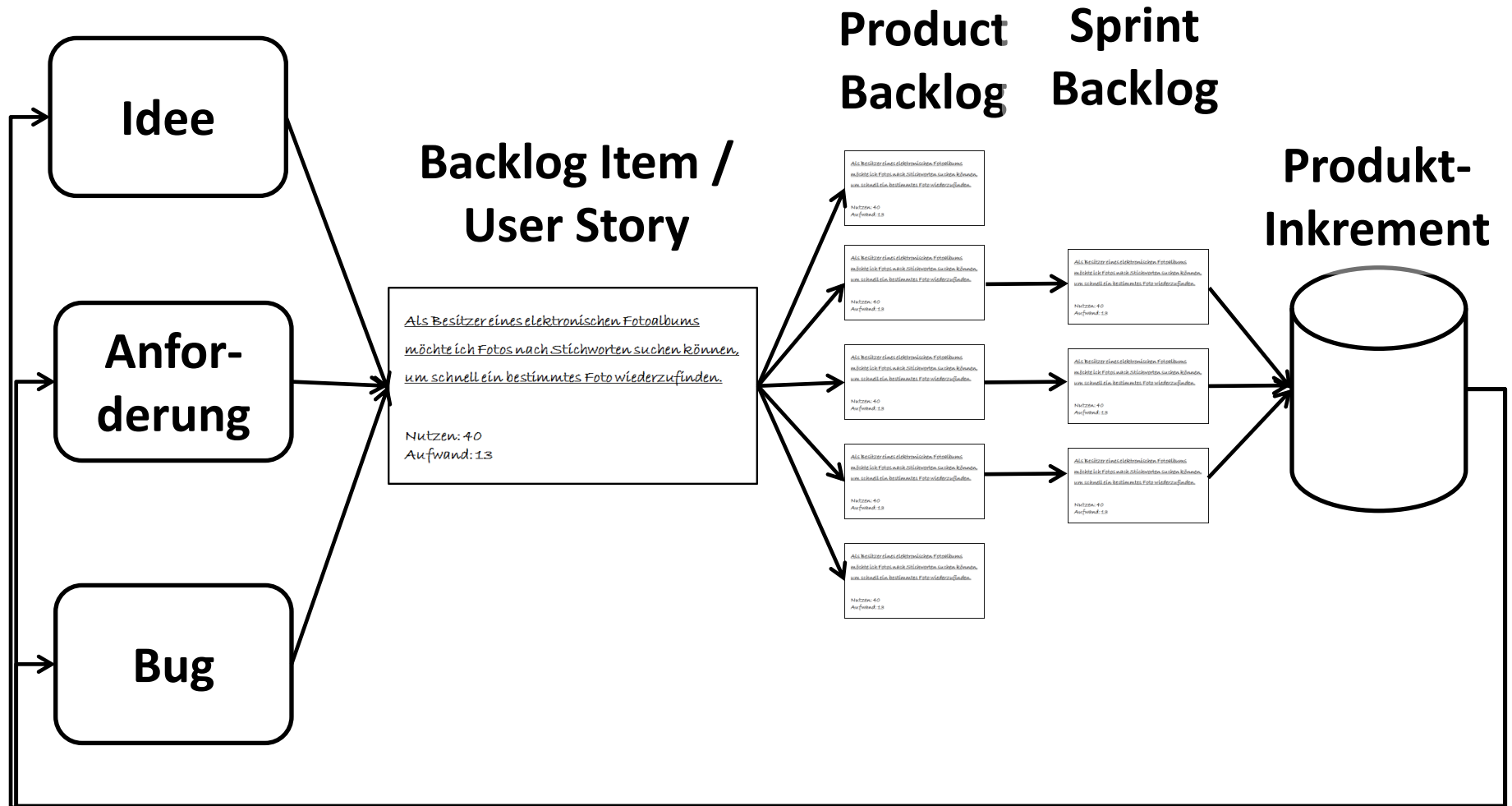
Benington, 1956

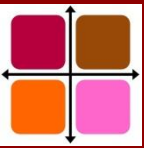


Royce, 1970

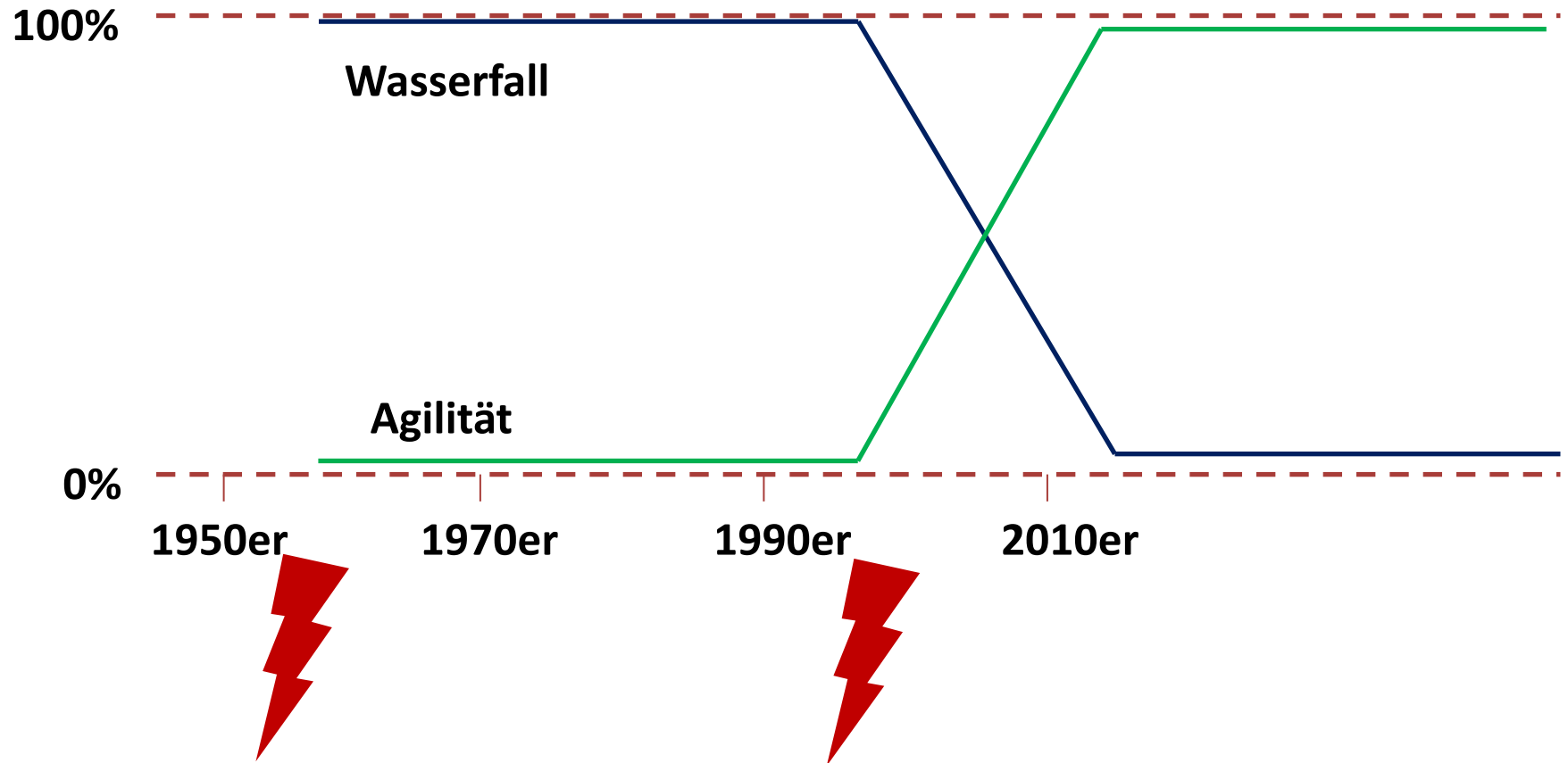


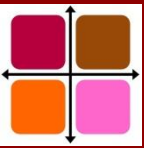
# Definition: Agilität





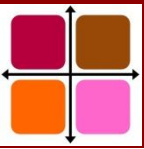
# Vergangenheit: übliche Geschichte





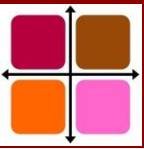
# Falsche Hypothesen

1. Früher hielt man Wasserfall für das einzig wahre Vorgehensmodell
2. Agile = leichtgewichtige, iterative Entwicklung erst seit 90er



# Richtiger: Hypothese 1

1. Früher hielt man Wasserfall für das einzig wahre Vorgehensmodell
  - Kein Hinweis darauf, dass Wasserfall funktioniert hat
  - Erste Nennung: Benington 1956, Wasserfall als Idee
  - Royce 1970: „risky and invites failure“
  - In 80ern verstärkte Kritik
2. Agile = leichtgewichtige, iterative Entwicklung erst seit 90er



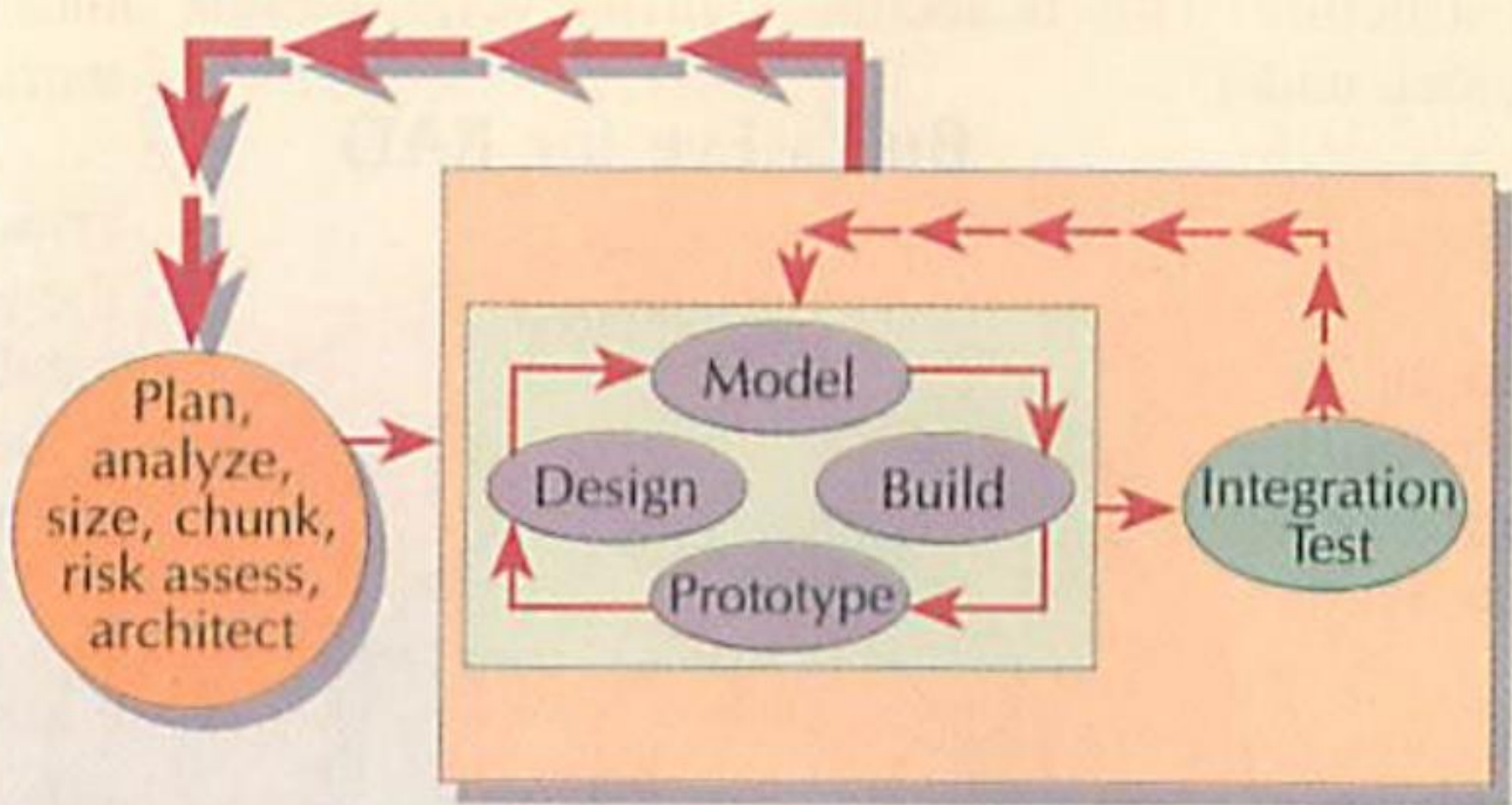
## Richtiger: Hypothese 2

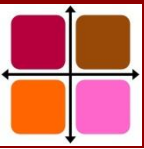
1. Früher hielt man Wasserfall für das einzig wahre Vorgehensmodell
2. Agile = leichtgewichtige, iterative Entwicklung erst seit 90er
  - Agilität in der heutigen Form (XP, Scrum) aus 90ern
  - 80er: RAD (Rapid Application Development)
  - davor: Cowboy Coding, Code-and-fix



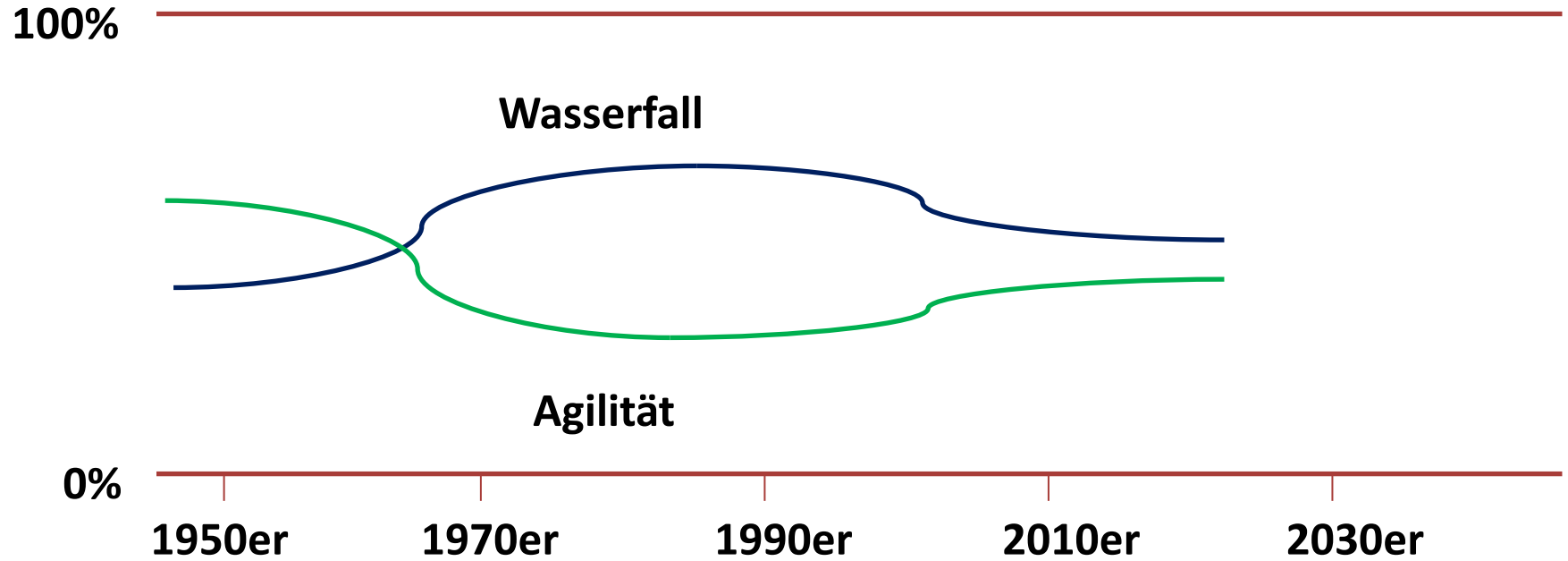
# RAD $\approx$ Scrum

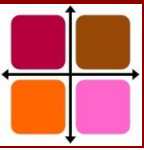
## Timebox for Each "Chunk"



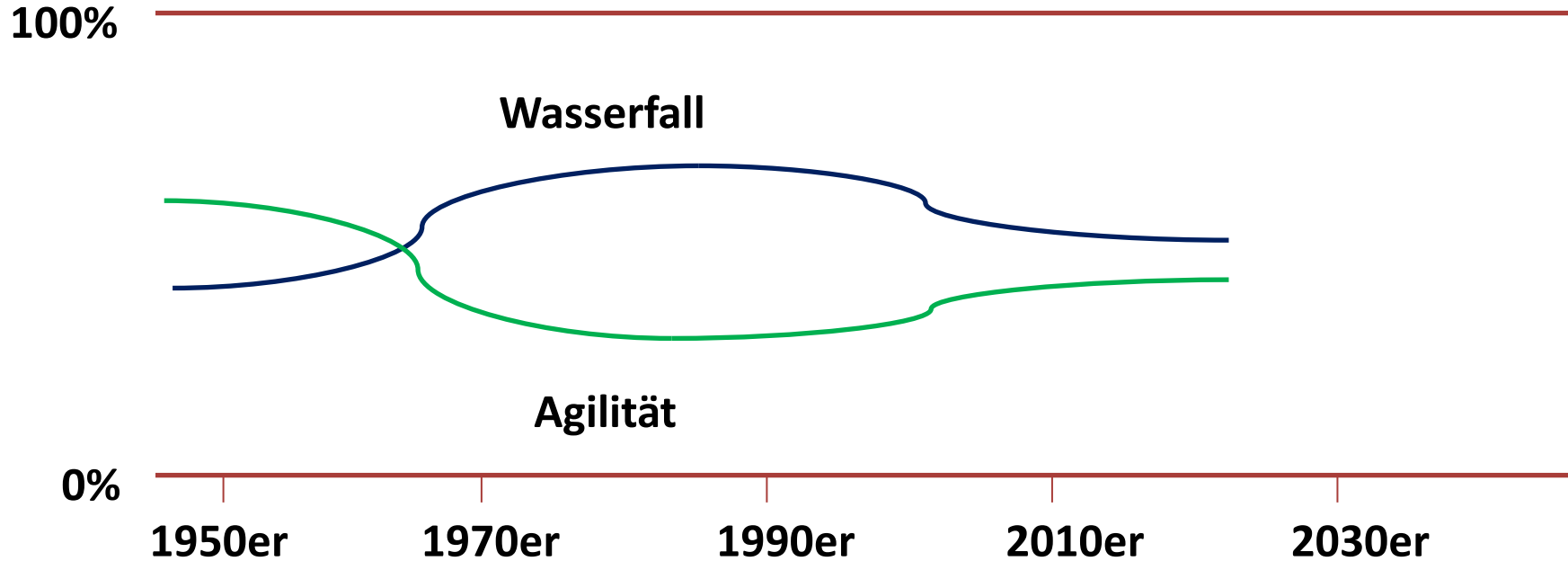


# Vergangenheit richtig



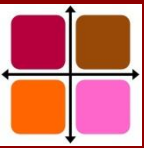


# Vergangenheit richtig



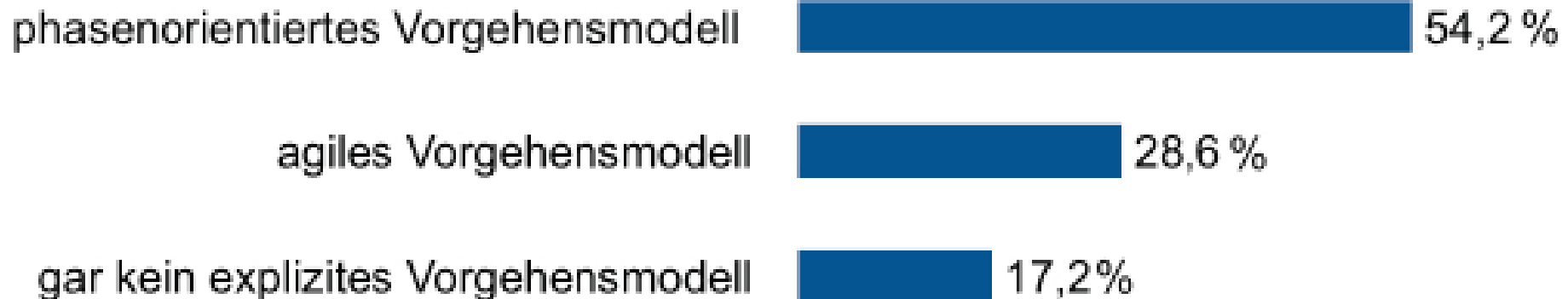
Standardisierung von Wasserfall und Agilität:

- Agile Manifesto, Scrum, XP
- Rational Unified Process RUP, V-Modell XT

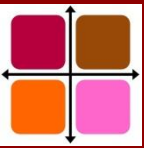


# Gegenwart: Vorgehensmodelle

*Wie ordnen Sie Ihr Vorgehensmodell in der Softwareentwicklung ein, eher als ...*



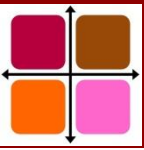
© Softwaretest-Umfrage 2011: HS Bremen, HS Bremerhaven, FH Köln, ANECON, GTB, STB



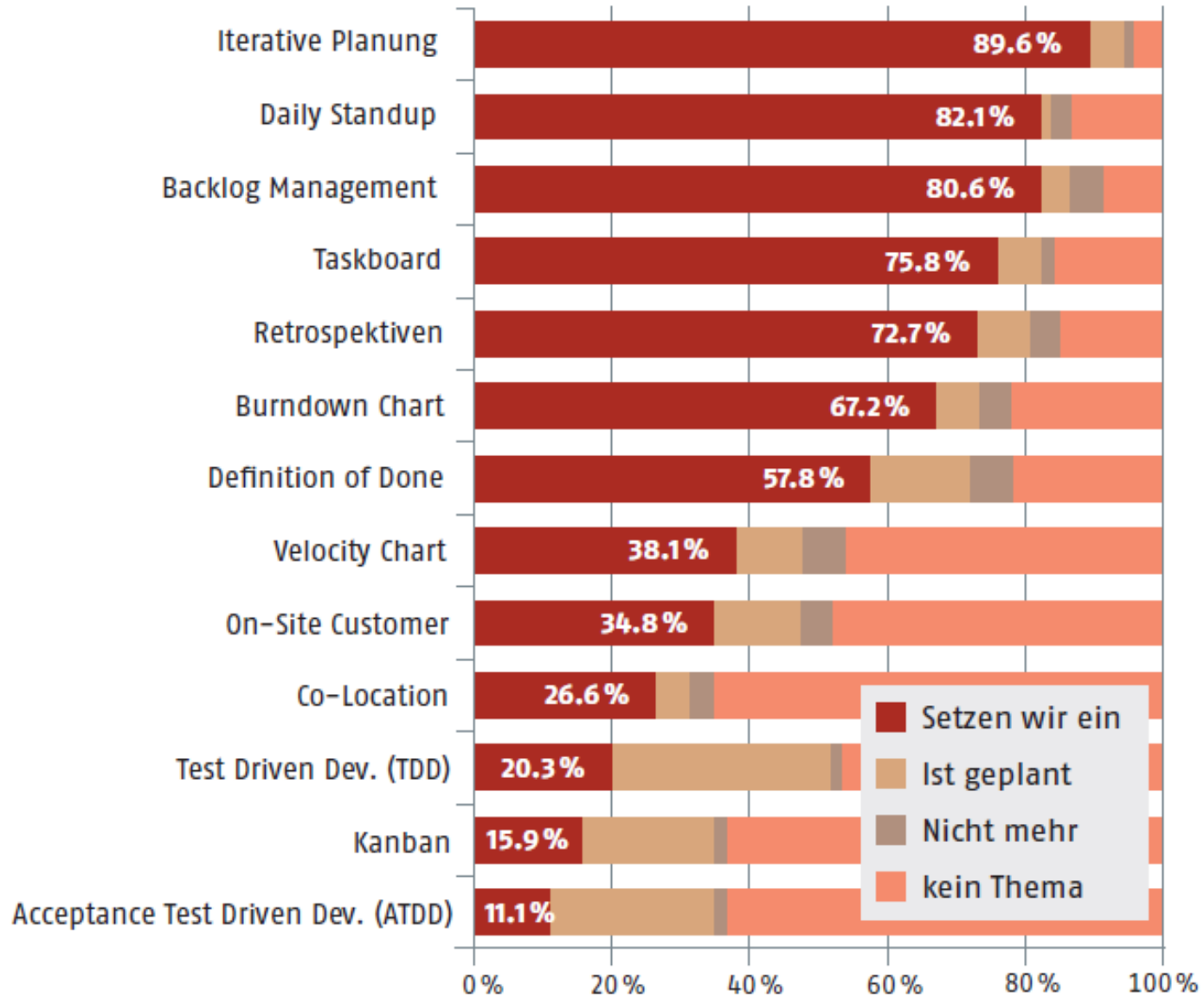
# Gegenwart: Scrum häufigste agile Methode

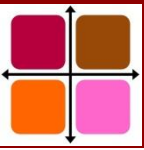
## 4. Please specify the dominant Agile approach that your project use

		Response Percent	Response Count
XP (Extreme Programming)		7.5%	35
Scrum		56.0%	262
XP-Scrum		14.5%	68
DSDM (Dynamic Systems Development Method)		0.9%	4
Crystal Clear		0.4%	2
FDD (Feature Driven Development)		1.3%	6
Lean		5.8%	27
Mixed/Other (please specify)		13.7%	64
answered question			468



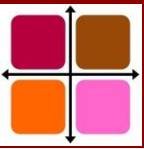
# Gegenwart: ScrumBut





# Erfolgsfaktoren und Nebenwirkungen





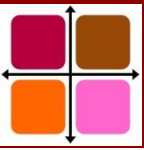
# Trends

**Größere Teams**

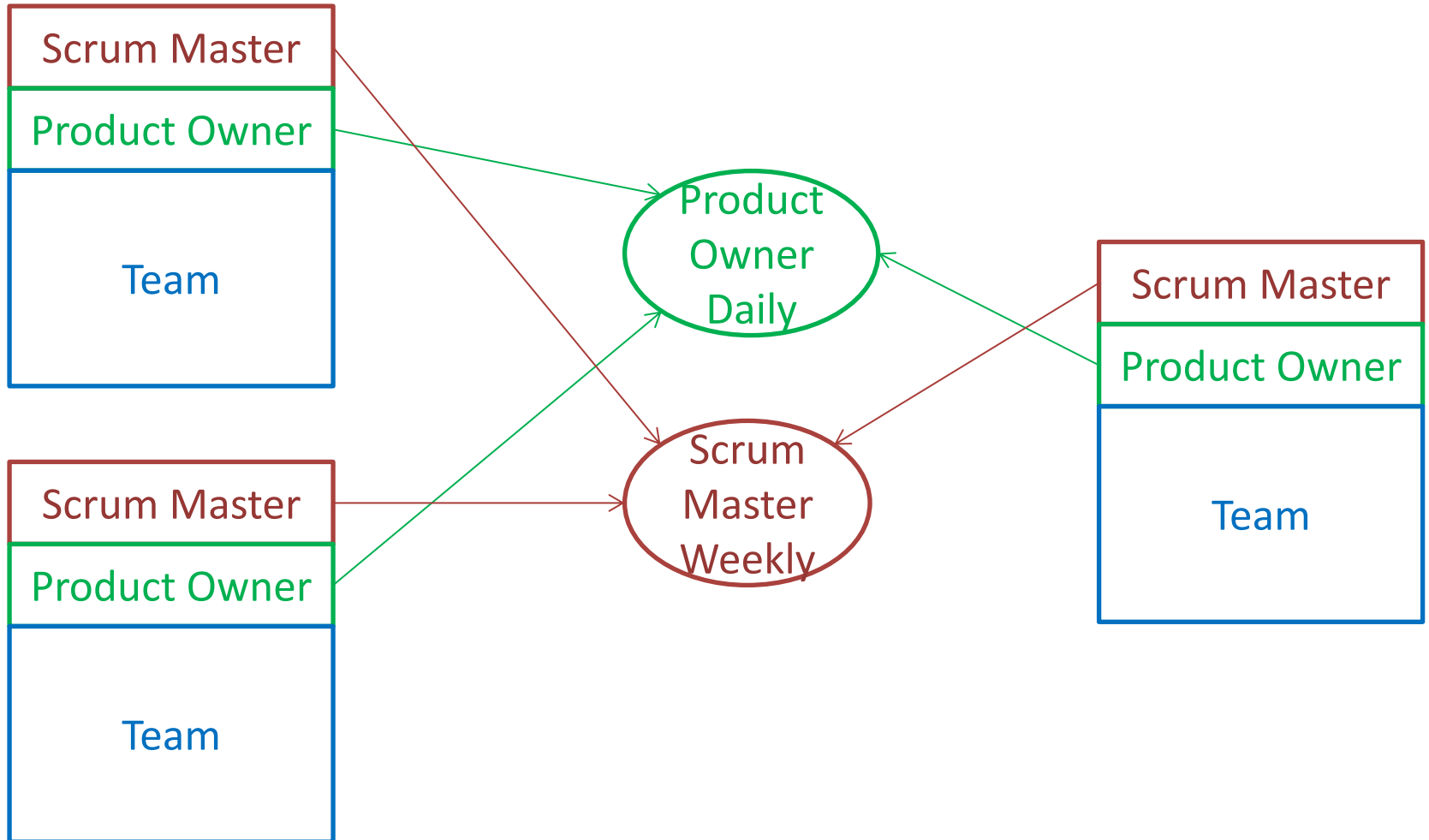
**Verteilte Teams**

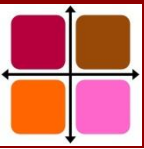
**Sicherheitskritische  
Software**



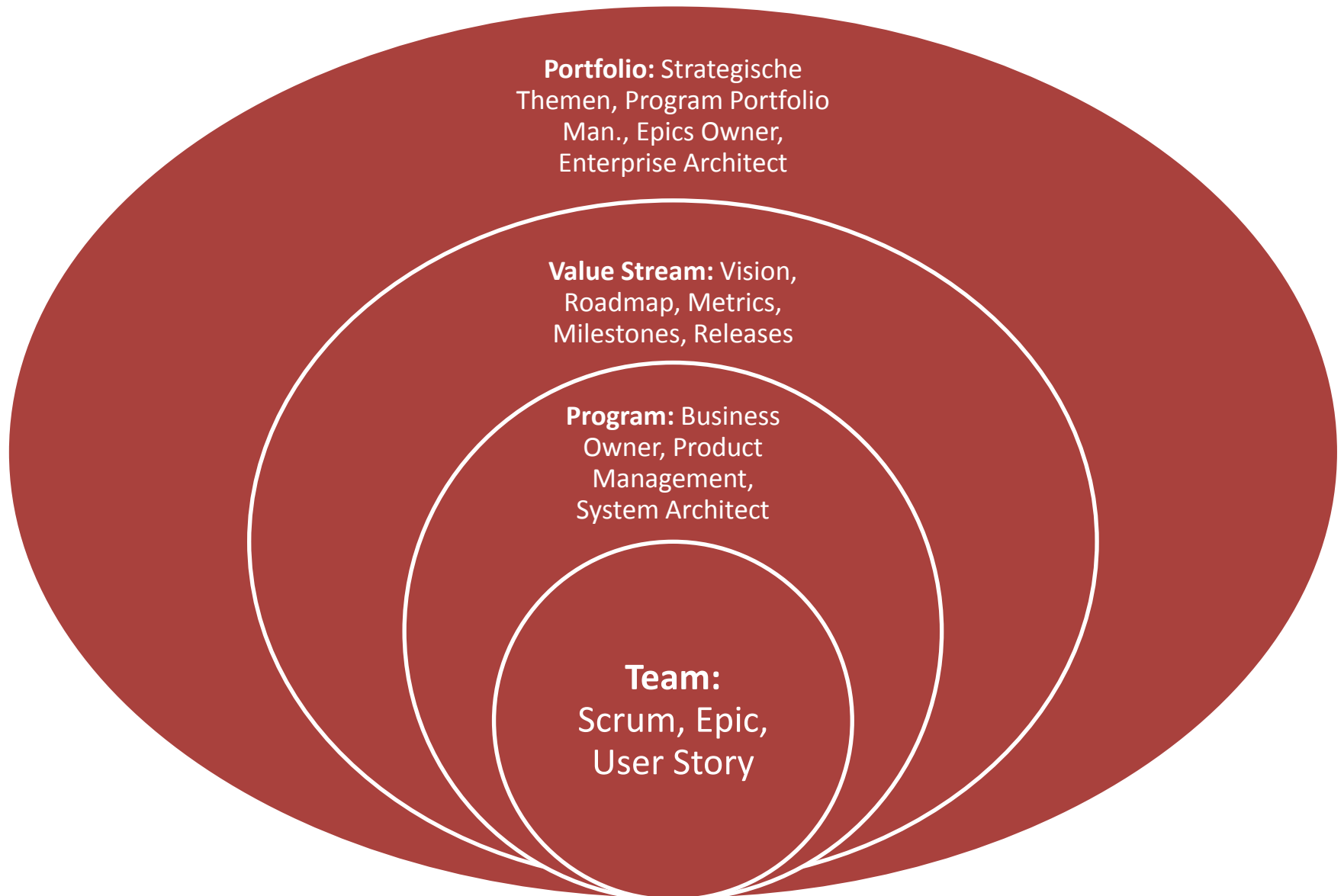


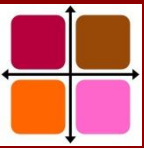
# Wie Scrum skaliert



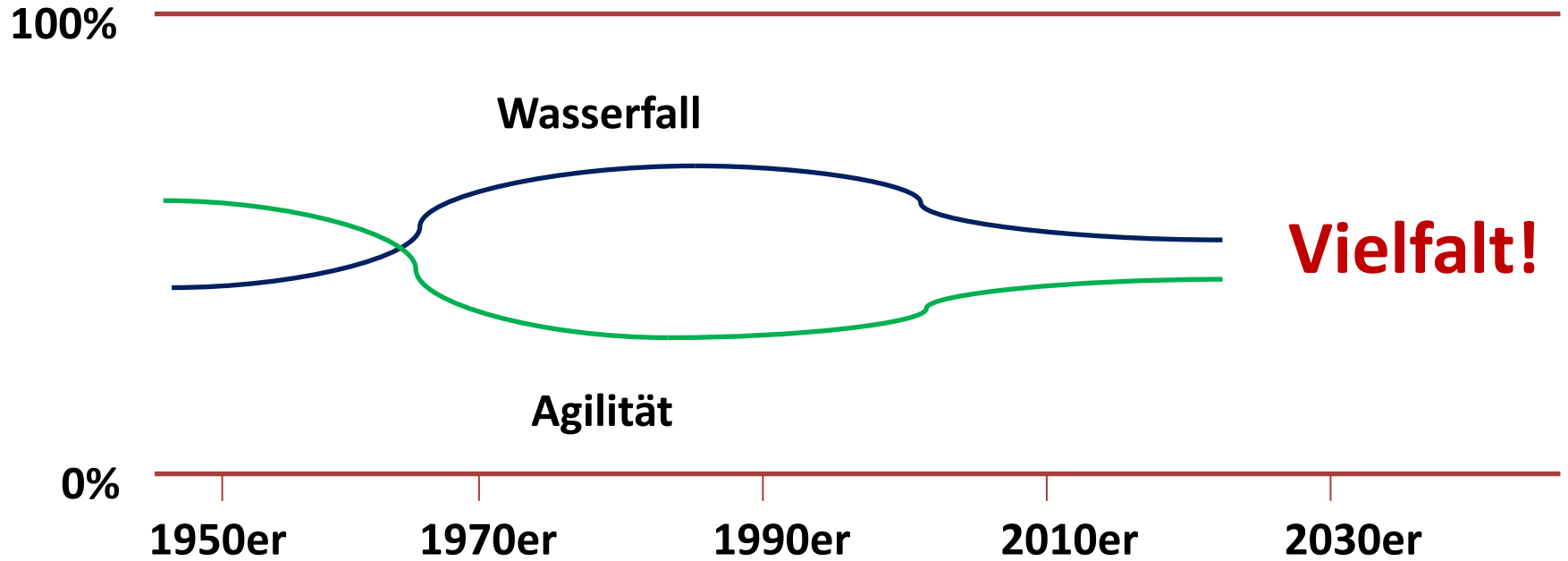


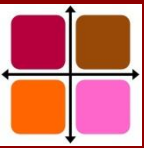
# Scaled Agile Framework: Ist das noch agil?





# Zukunft





# Vielen Dank für Ihre Aufmerksamkeit!

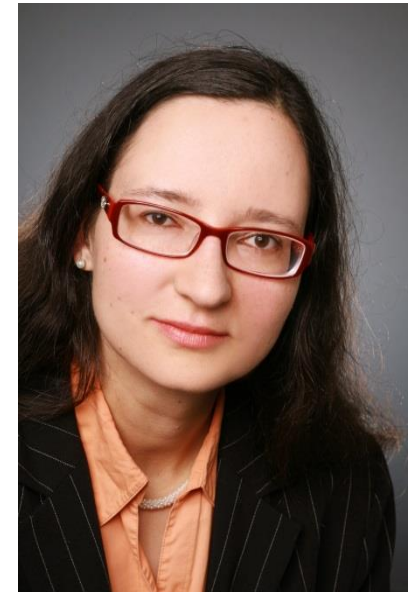
Dr. habil. Andrea Herrmann

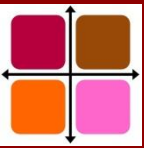
Trainerin und Beraterin

[herrmann@herrmann-ehrlich.de](mailto:herrmann@herrmann-ehrlich.de)

<http://www.herrmann-ehrlich.de>

<http://herrmannehrlich.twoday.net/>





# Bildquellen

[https://upload.wikimedia.org/wikipedia/commons/e/ec/Bookshelf\\_Prunksaal\\_OeNB\\_Vienna\\_AT\\_matl00786ch.jp](https://upload.wikimedia.org/wikipedia/commons/e/ec/Bookshelf_Prunksaal_OeNB_Vienna_AT_matl00786ch.jpg)

g

Wasserfall: Benington, Herbert D.: "Production of Large Computer Programs," Proc. ONR Symposium on Advanced Programming Methods for Digital Computers, June 1956, pp. 15-27. Also available in Annals of the History of Computing, Oct. 1983, pp. 350-361, and Proc. Ninth Int'l Conf. Software Engineering, Computer Society Press, 1987.

Royce, Winston W.: Managing the Development of Large Software Systems, Proceedings of the 9th international conference on Software Engineering ICSE '87, IEEE Computer Society Press, Los Alamitos, CA, USA, 1987, S. 328-338

[https://commons.wikimedia.org/wiki/File:Extreme\\_programming\\_in\\_action.jpg](https://commons.wikimedia.org/wiki/File:Extreme_programming_in_action.jpg)

RAD: Gottesdiener, Ellen: RAD Realities: Beyond the Hype to How RAD Really Works. Application Development Trends, August 1995, S. 28-38