

Theory Based Software Engineering with the SEMAT Kernel: Preliminary Investigation and Experiences

Pan-Wei Ng

www.ivarjacobson.com

My Background and Objective

- I am a coach – help teams and organization improve
- Making theory practical to practitioners

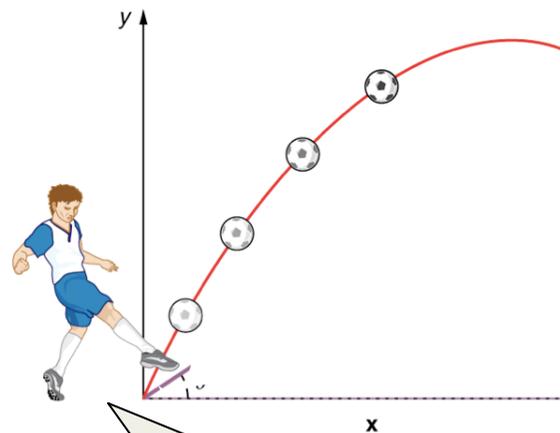
What is Theory? Why talk About Theory?

- Wikipedia: **Theory** is a group of ideas meant to explain a certain topic, such as a single or collection of fact(s), event(s), or phenomenon(a)(on).
- My version: Theory is a set of statements that relates variables and cause and effect.
- Why talk about theory? It is all about improvement, becoming better at what we are doing and to help others become better too
 - Explain (the theory) how we become better
 - Adapt the theory to another but similar context

Einstein: “Theory defines what we observe
(behave)”

Light hearted analogy: World Cup Fever (1/3)

- Soccer: Goal is to get the ball between the posts?
- How should I kick? How much force? Which angle?

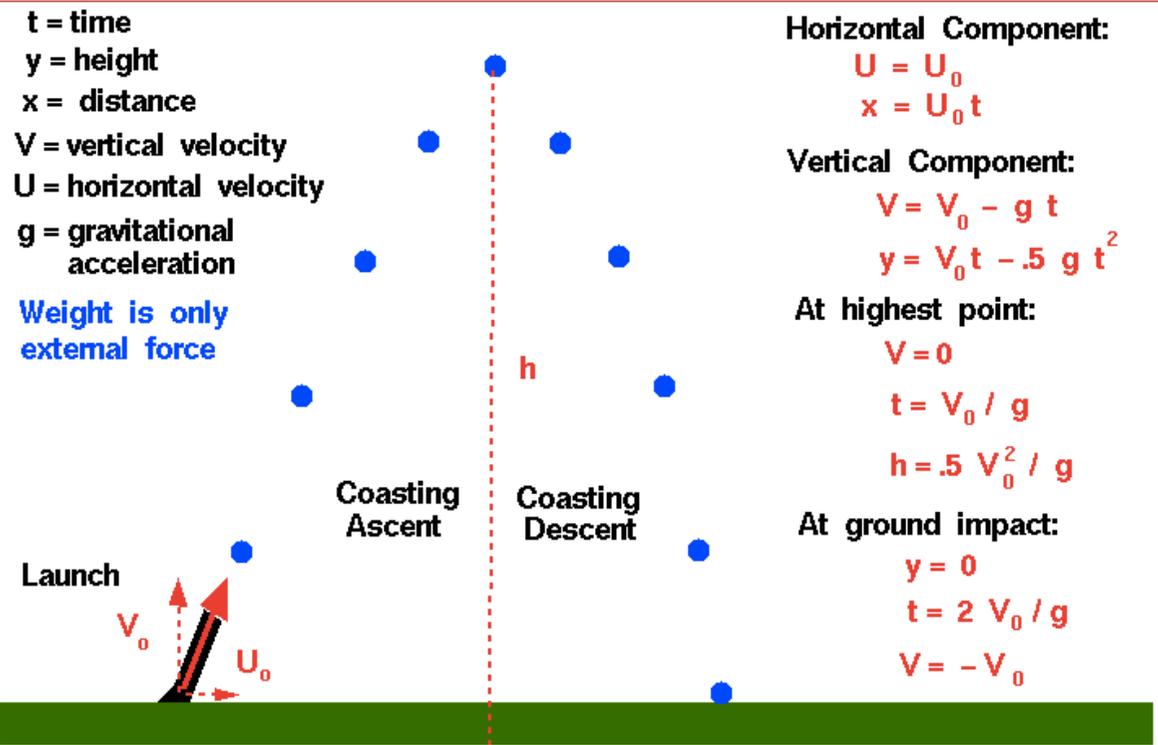


I followed your theory, it did not work in practice

The theory

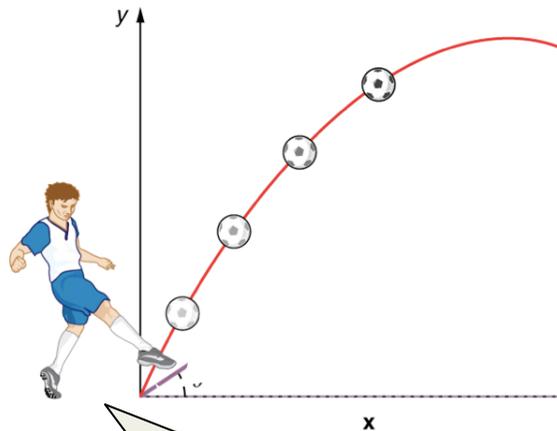
t = time
y = height
x = distance
V = vertical velocity
U = horizontal velocity
g = gravitational acceleration

Weight is only external force



Light hearted analogy: World Cup Fever (2/3)

- Soccer: Goal is to get the ball between the posts?
- How should I kick? How much force? Which angle?



I followed your theory, it did not work in practice

The reality (considering drag)

Vertical Ascent

$F_{net} = -W - D$

$a = -g - \frac{Cd A \rho V^2}{2m}$

$V = V_t \frac{V_0 - V_t \tan(t g / V_t)}{V_t + V_0 \tan(t g / V_t)}$

$y = \frac{V_t^2}{2g} \ln \left(\frac{V_0^2 + V_t^2}{V^2 + V_t^2} \right)$

$y_{max} = \frac{V_t^2}{2g} \ln \left(\frac{V_0^2 + V_t^2}{V_t^2} \right)$

$V_t = \sqrt{\frac{2m g}{Cd A \rho}}$

$t_{(v=0)} = \frac{V_t}{g} \tan^{-1} \left(\frac{V_0}{V_t} \right)$

Vertical Descent

$F_{net} = -W + D = 0$

$a = 0$

$V = V_t$

Horizontal:

$F_{net} = -D$

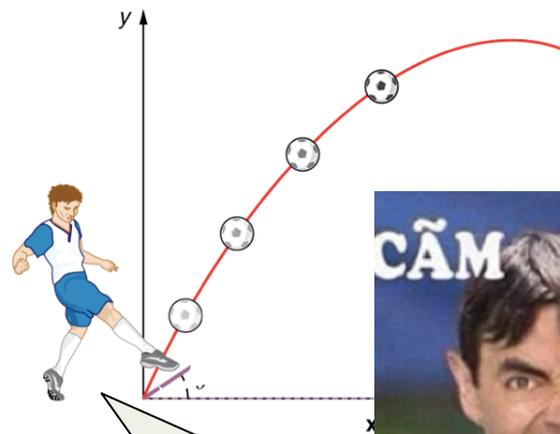
$a = -\frac{Cd A \rho U^2}{2m}$

$U = \frac{V_t^2 U_0}{V_t^2 + g U_0 t}$

$x = \frac{V_t^2}{g} \ln \left(\frac{V_t^2 + g U_0 t}{V_t^2} \right)$

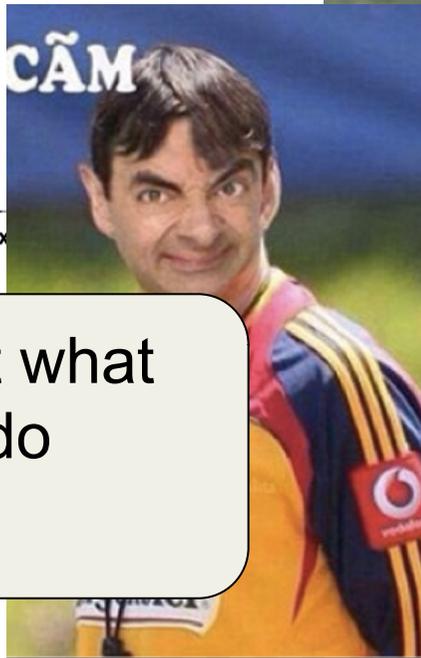
Light hearted analogy: World Cup Fever (3/3)

- Soccer: Goal is to get the ball between the posts?
- How should I kick? How much force? Which angle?



I see, but what should I do now?

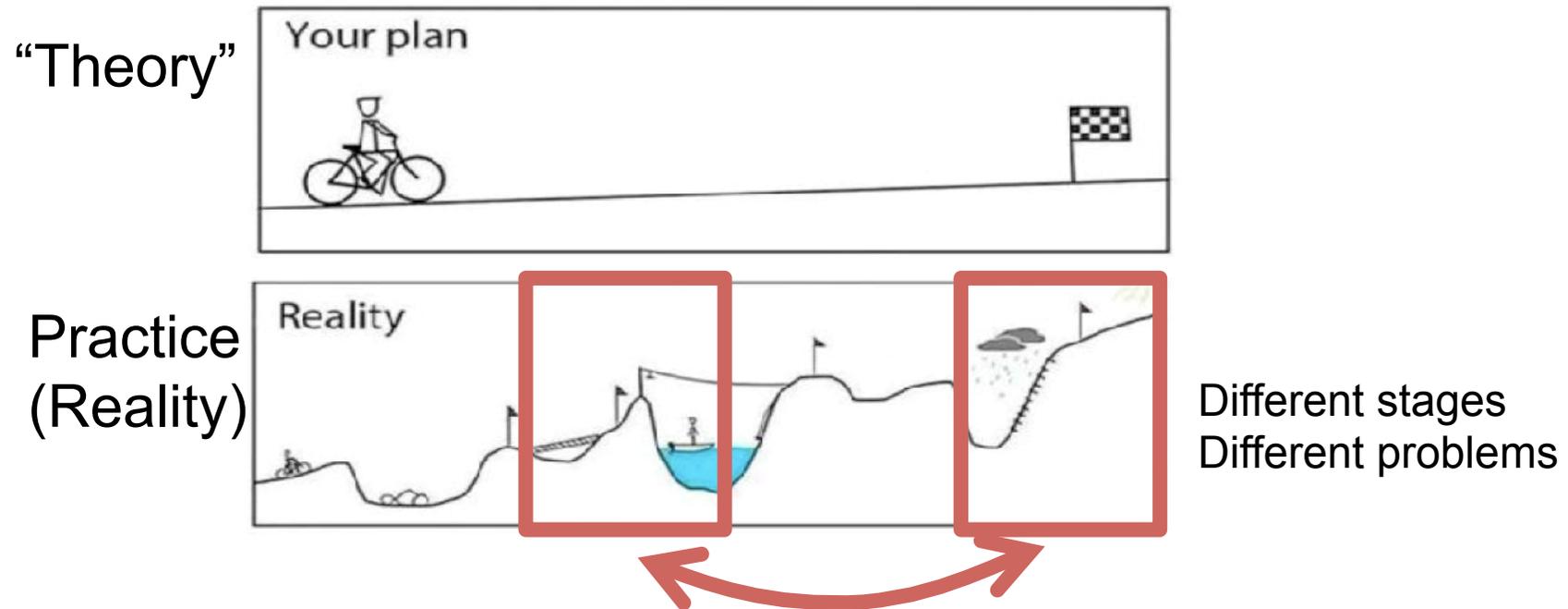
The reality (don't forget human factors)



Human factors are separate and distinct from physics, but still affects outcome

Software Engineering Theory and Practice (Reality)

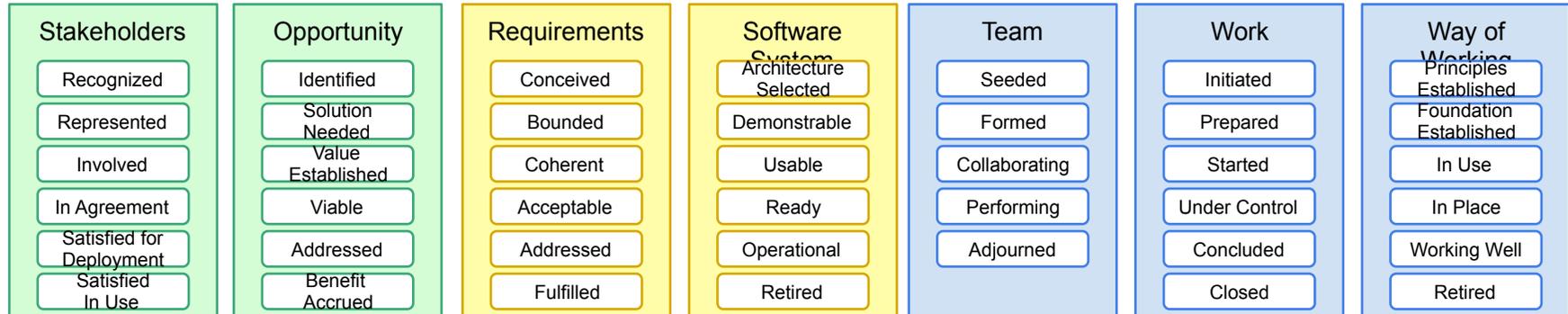
- Software development (engineering) is complex with many factors



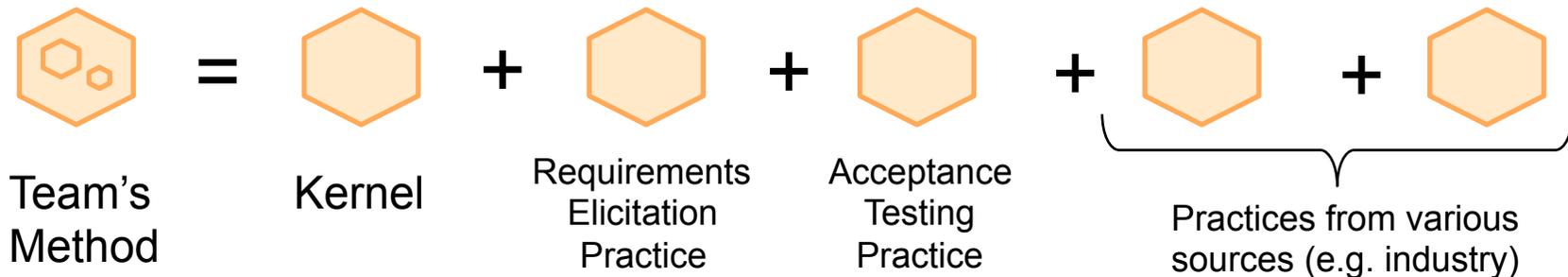
Kurt Levin: “There is nothing as practical as good theory”

What is Essence? Two Main Ideas

1. Alphas and states – aspects of progress and health



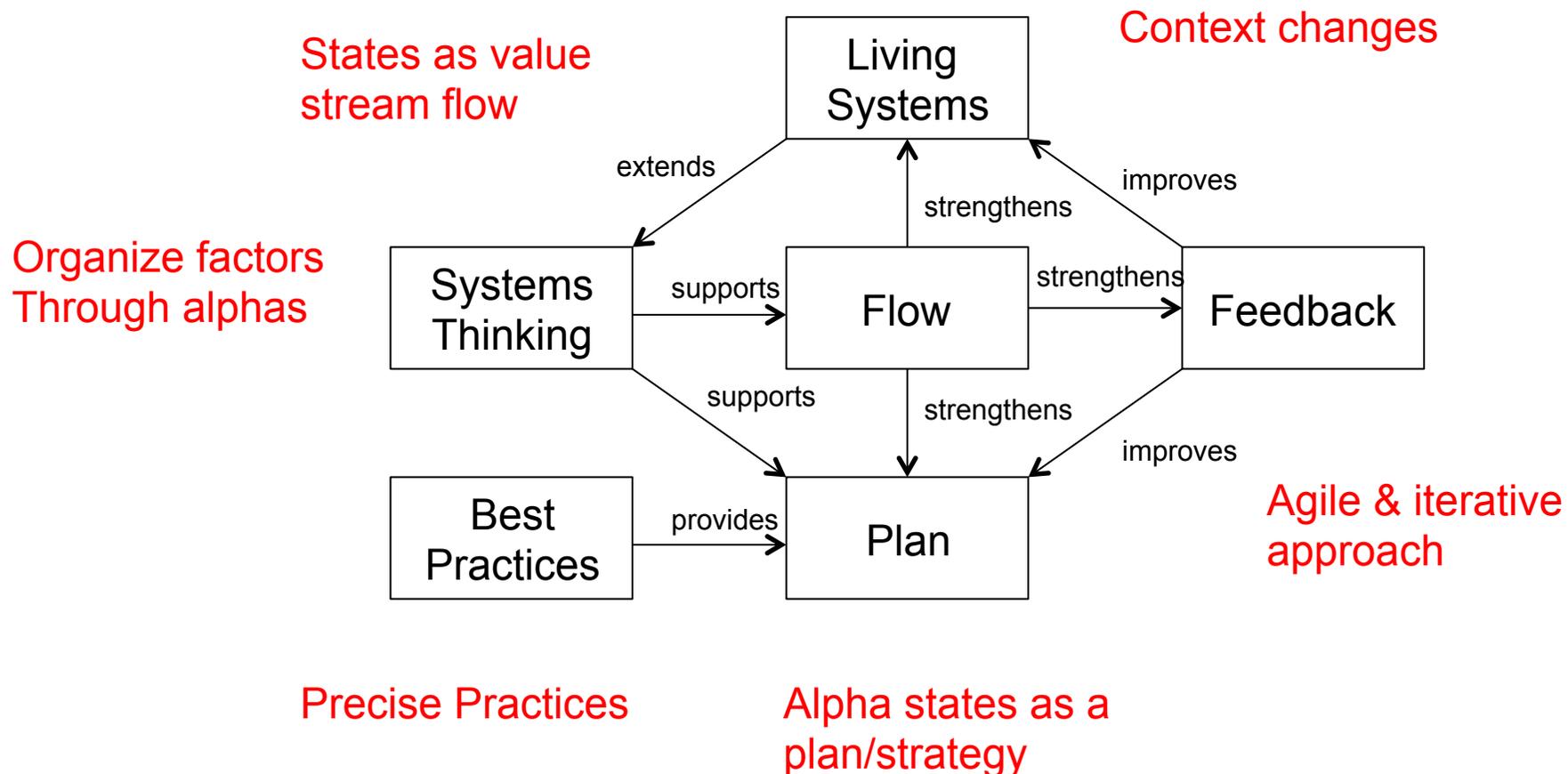
2. Separation of concerns – Methods are a composition of practices on top of the kernel



Separation of concerns apply to theories too

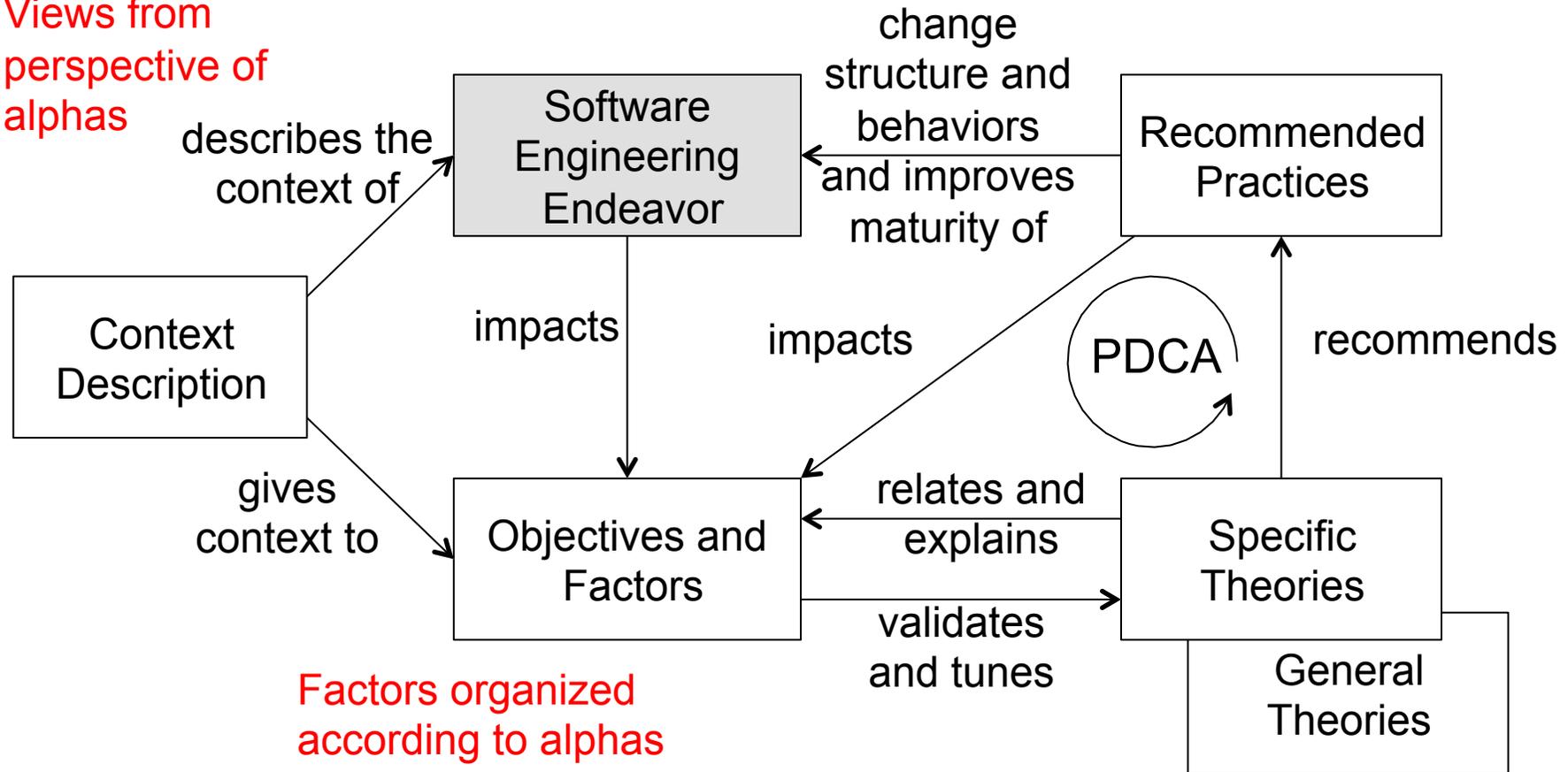
Underlying foundations of process improvement theories

- Van Hilst and Fernandez's Pattern System of Underlying Theories of Software Process Improvement (2010)



Theory Based Software Engineering

Architecture
Views from
perspective of
alphas

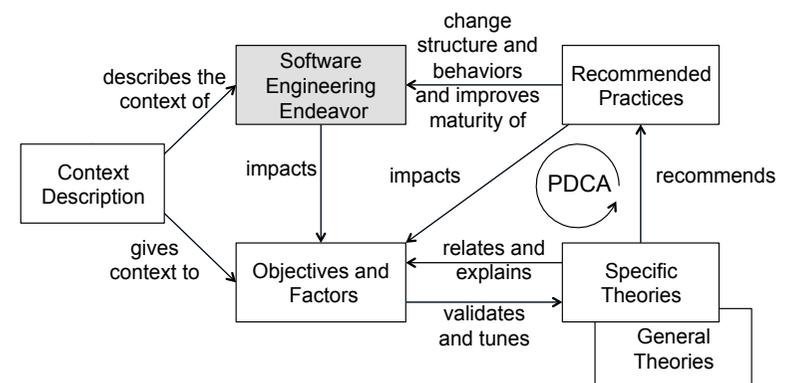


Factors organized
according to alphas

Separation of concerns
between theories

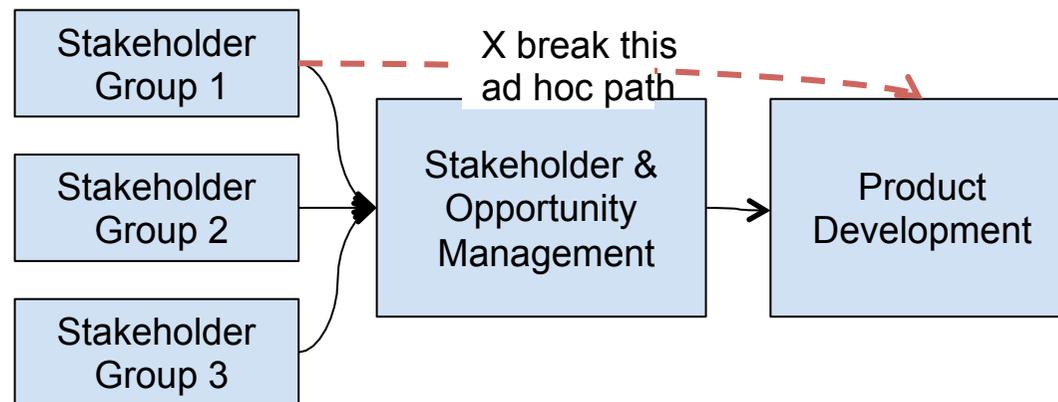
Steps to TBSE

1. Identify which aspect to improve
 - Alphas for selecting area(s) of improvement
2. Describe context – architecture views
 - Structural versus dynamic view
 - Gives factors context
3. Theorize the relationship between factors and outcomes
 - Specific theory and general (background) theory
4. Make recommendations
 - Recommendations affect factors
5. Act and observe behaviors
 - They may work according to or against recommendations
6. Validate/Tune the theory

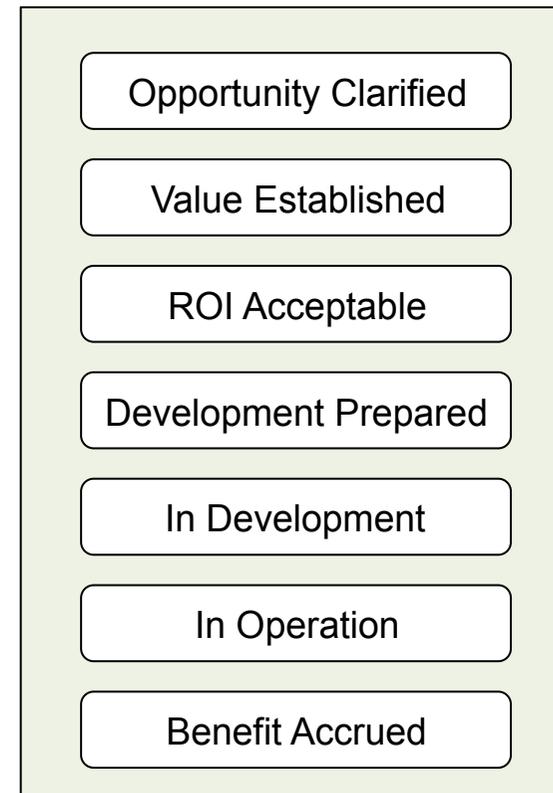


Case Study: Knowledge Management System

- Area to improve: Stakeholder and Opportunity Management
 - Symptoms: too many requirements, implemented requirements not being used by end-user community, development overload
- Architecture Views: Structural and Behavioral

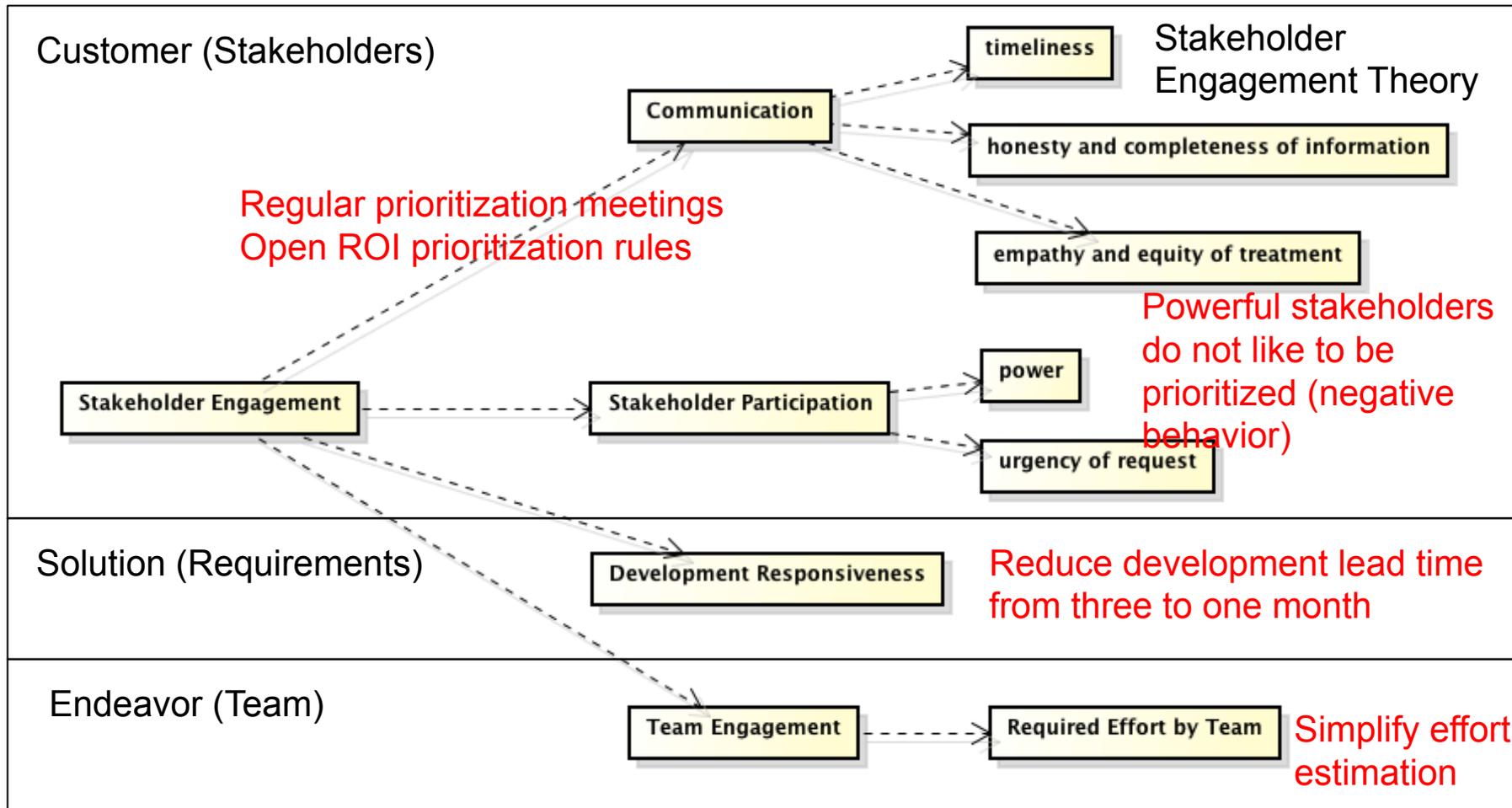


- Structural views constructed based on instance of alphas
- Behaviors defined through views of alpha states



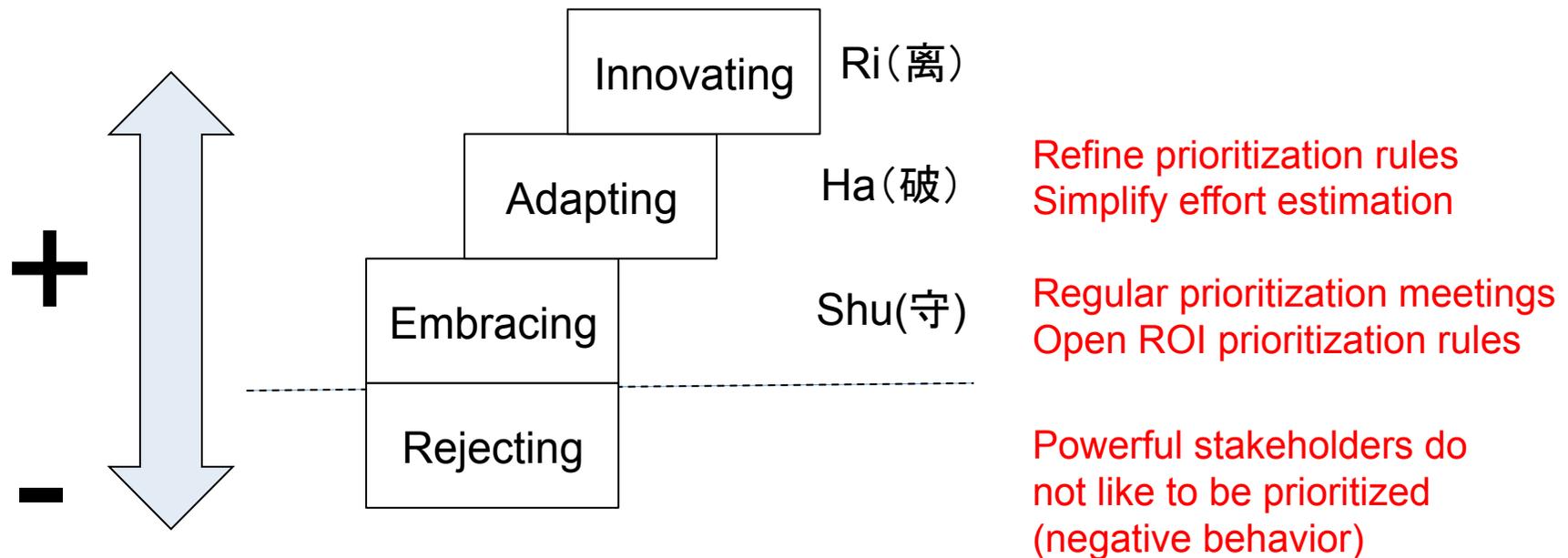
Identify success factors and recommendations

- Identified factors grow as more observations are made



Monitor and observe

- In the ideal world, everything works out as planned or recommended
- Negative behaviors often highlight missing factors or oversight in some aspects



Lessons learnt (1/2)

- Essence is an attractive candidate for organizing context descriptions and factors
 - Check assumptions from different aspects / perspectives
- Objective is to gradually and systematically engage practitioners to “theorize” their approach to development and process improvement
 - Theory defines what you observe
 - Specific theory versus general/background theory (assumptions)
 - Context is important
 - Organize and describe context

Lessons learnt (2/2)

- TBSE is very much like Systems Thinking
 - Actually, it is built on the underlying pattern system of process improvement (Van Hilst and Fernandez)
 - Differentiators:
 - having an agreed domain model (Essence) to begin with
 - architecture descriptions to give further context
 - Specific versus background theories (separation of concerns applied to theory)
- Training practitioners to “theorize” is challenging
 - It is not something they do naturally
 - They want answers fast (“you tell me” syndrome)
- Still very much work in progress

- Thank you