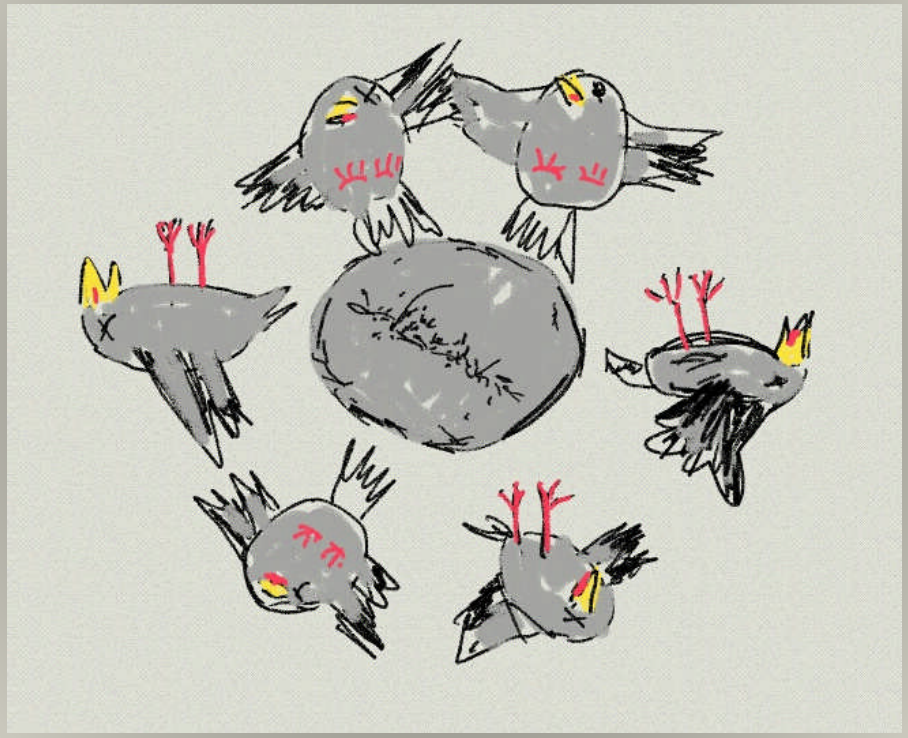
Killing Several Birds with One Stone:

Using SEMAT's ESSENCE in Teaching Software Engineering



KTH Royal Institute of Technology
Stockholm, Sweden
mekm2@kth.se

SEMAT

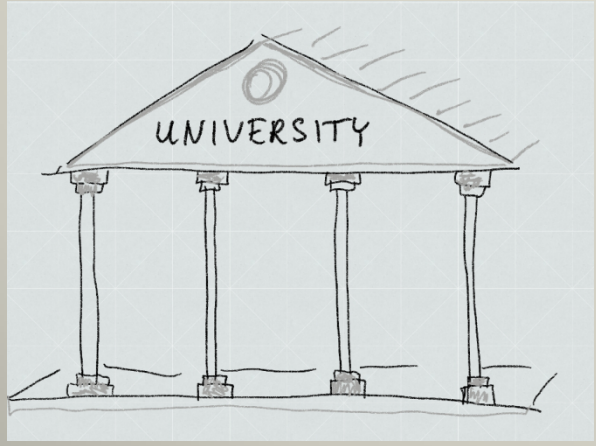
Outline

- Problems within education
- The SEMAT community
- ESSENCE
- SE education at KTH
- So how many birds….

Graduated students are poorly equipped for their future careers

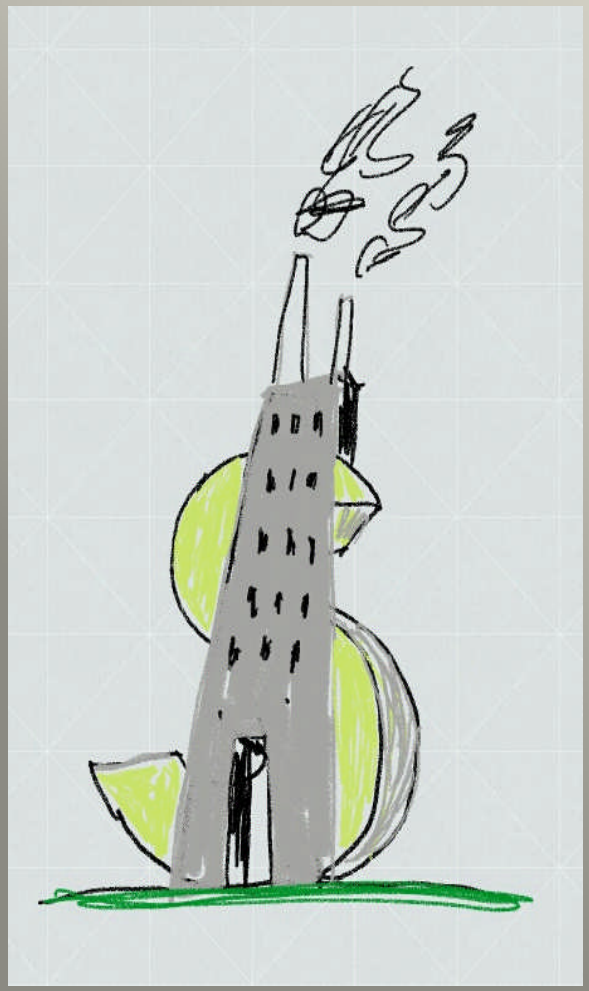
- Software engineering is difficult to learn in a classroom environmnet.
- Next to impossible to gain experience.
- Difficulties to assess students' progress and competence.
- Students have become nationally and internationally mobile.

Problems at universities



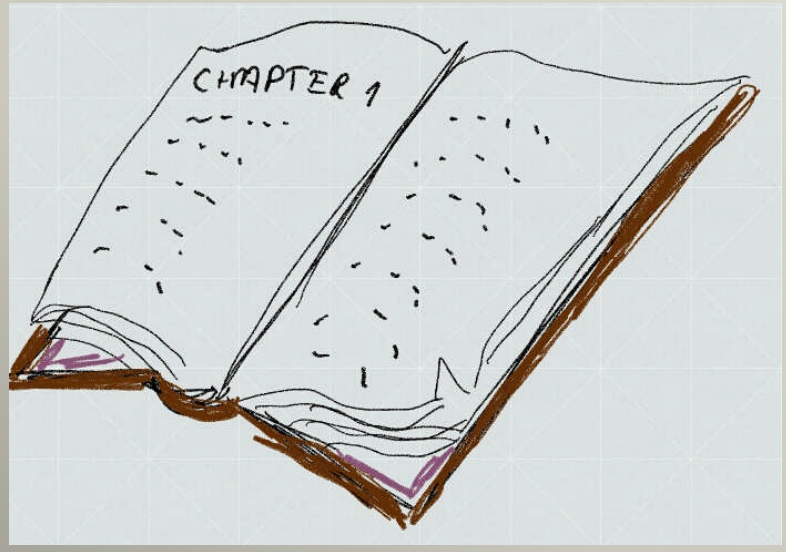
- Software engineering is often sqeezed into one course.
- You cannot grasp all within one course.
- No way of assessing current educators wrt how capable they are to deliver competencies.
- Need for a global approach for evaluating the educators, not just school reputation.

Problems within industry



- Shortage of employee candidates.
- Graduate students are poorly equipped for their future careers.
- Graduate students do not possess enough knowledge about and skills within software engineering.

Problems with educational material



- No interest in writing educational books.
- Different terminology used.
- Different understanding of software engineering
- Books too difficult for students to understand.
- There are no books that provide a good high-level overview of software engineering.

Outline

- Problems within education
- The SEMAT community
- ESSENCE
- SE education at KTH
- So how many birds….

We have no widely accepted common ground

- We do not lack methods or practices.
- Everyone of us knows how to develop **our own** software, but as a community we have **no** widely accepted common ground.

Some common problems

- Software development methods are all unique in their design and use of terminology so they cannot be easily compared.
- We do not know which methods we have in a large company.
- We have no solid knowledge which we can take from job to job.
- We have no common platform on which we could base
  - development of our methods
  - improvement of our methods
  - planning
  - Project status evaluation
  - risk identification
  - ……

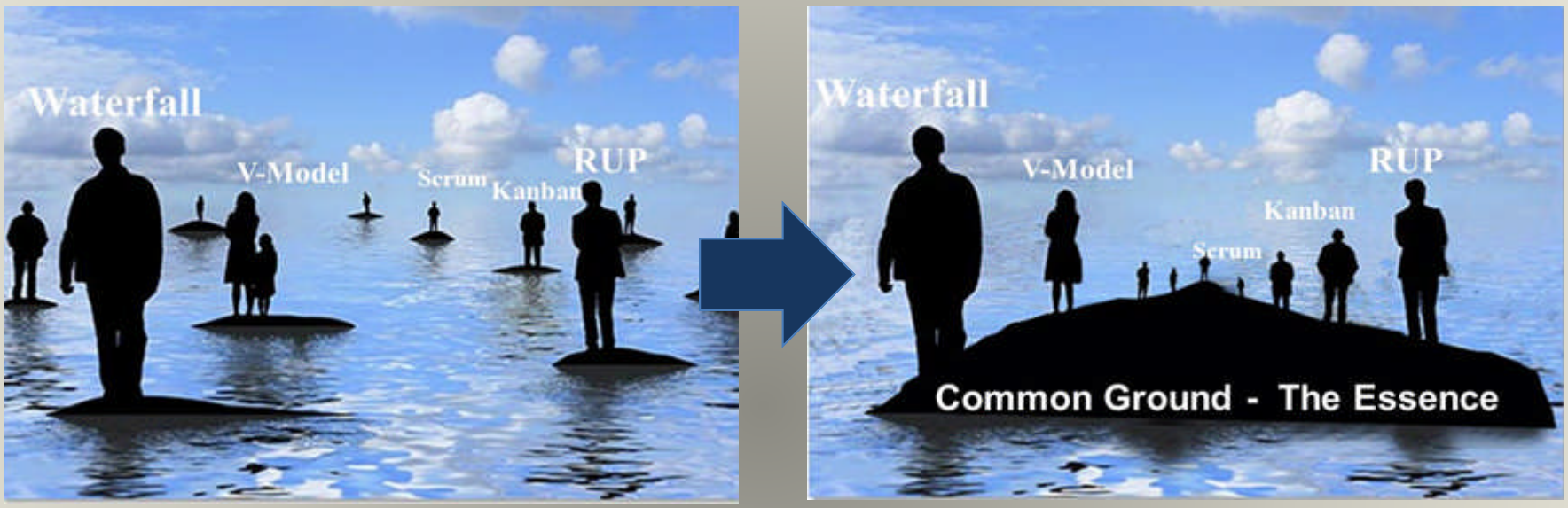
SEMAT

SEMAT: Software Engineering Method and Theory

Founded by Ivar Jacobson, Bertrand Meyer, Richard Soley in 2009

Re-found software engineering as a rigorous discipline based on a general theory of software engineering and a unifying process framework

Common Ground



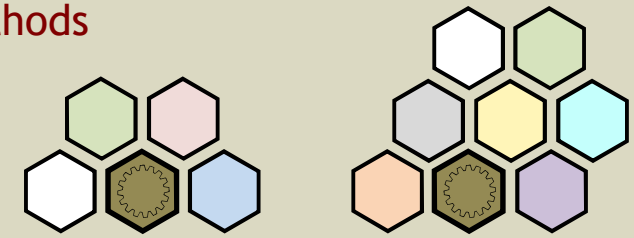
Find a kernel (essence) of widely agreed elements within software engineering

Outline

- Problems within education
- The SEMAT community
- ESSENCE
- SE education at KTH
- So how many birds….

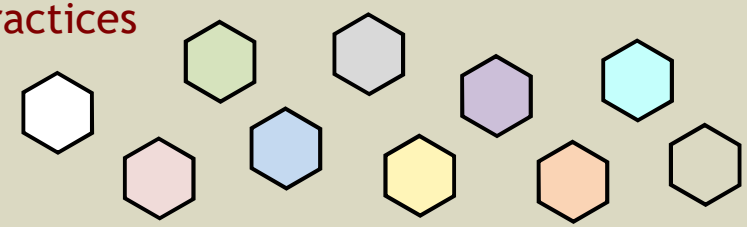
What is Essence?

Methods



Custom Method M Custom Method N

Practices



Kernel

Essence Kernel

Language

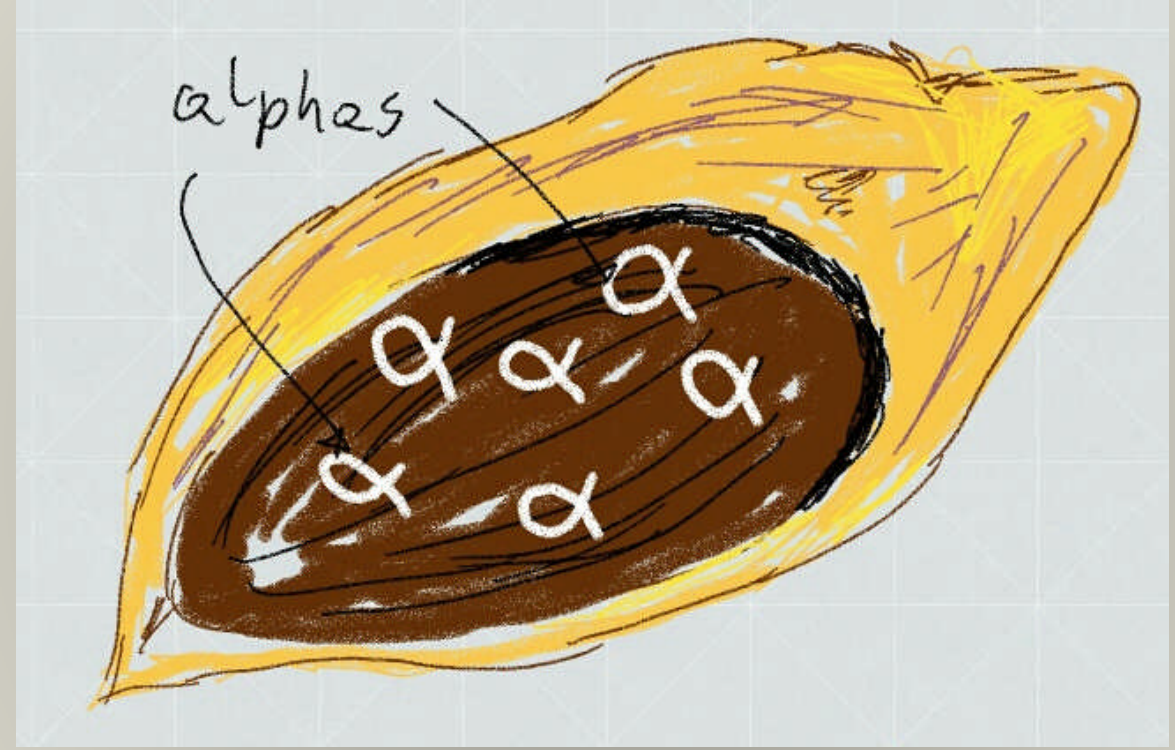
Essence Language

OBJECT MANAGEMENT GROUP

Essence - Kernel and Language for Software Engineering Methods

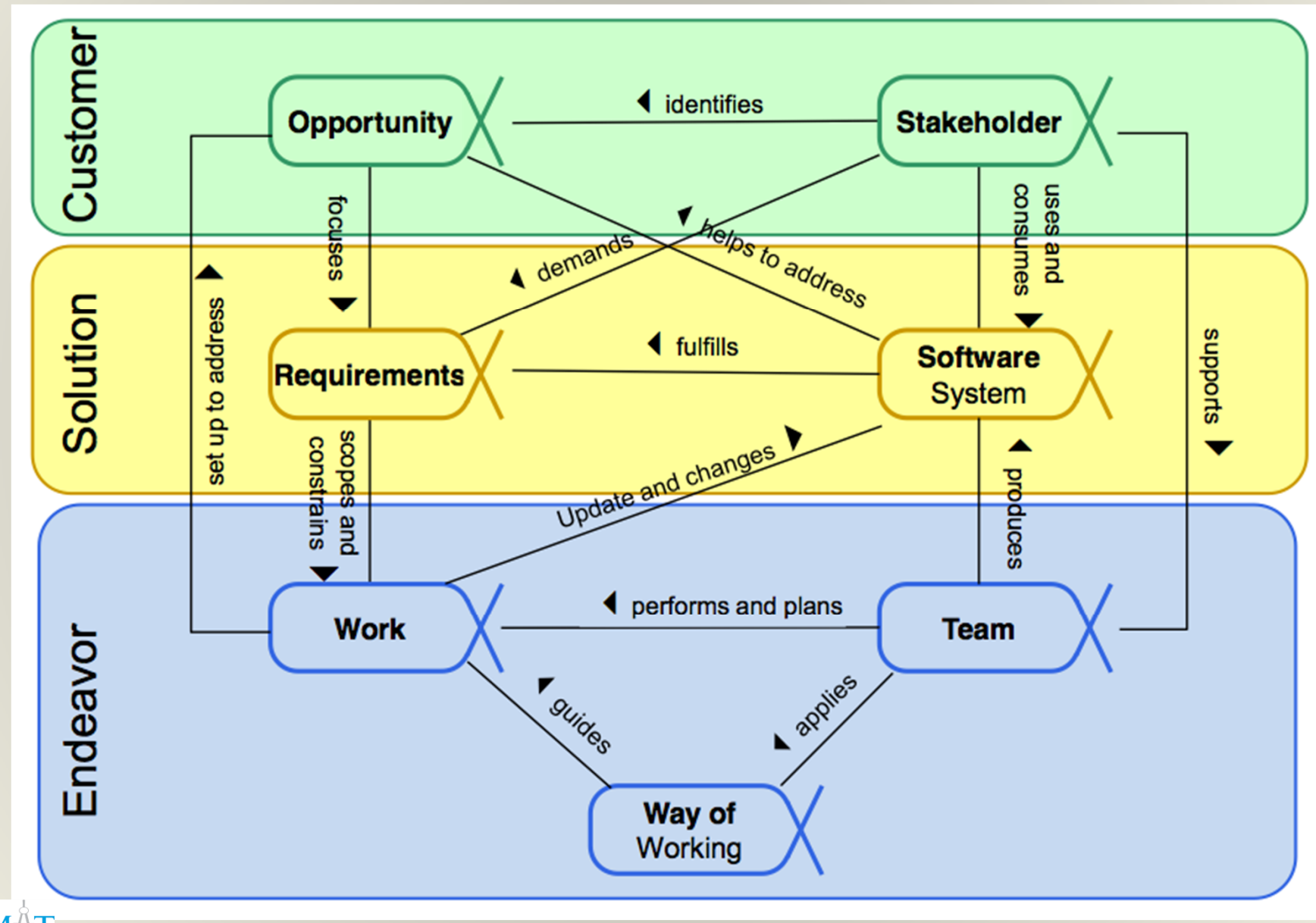
http://www.omg.org/spec/Essence/Current

The Kernel

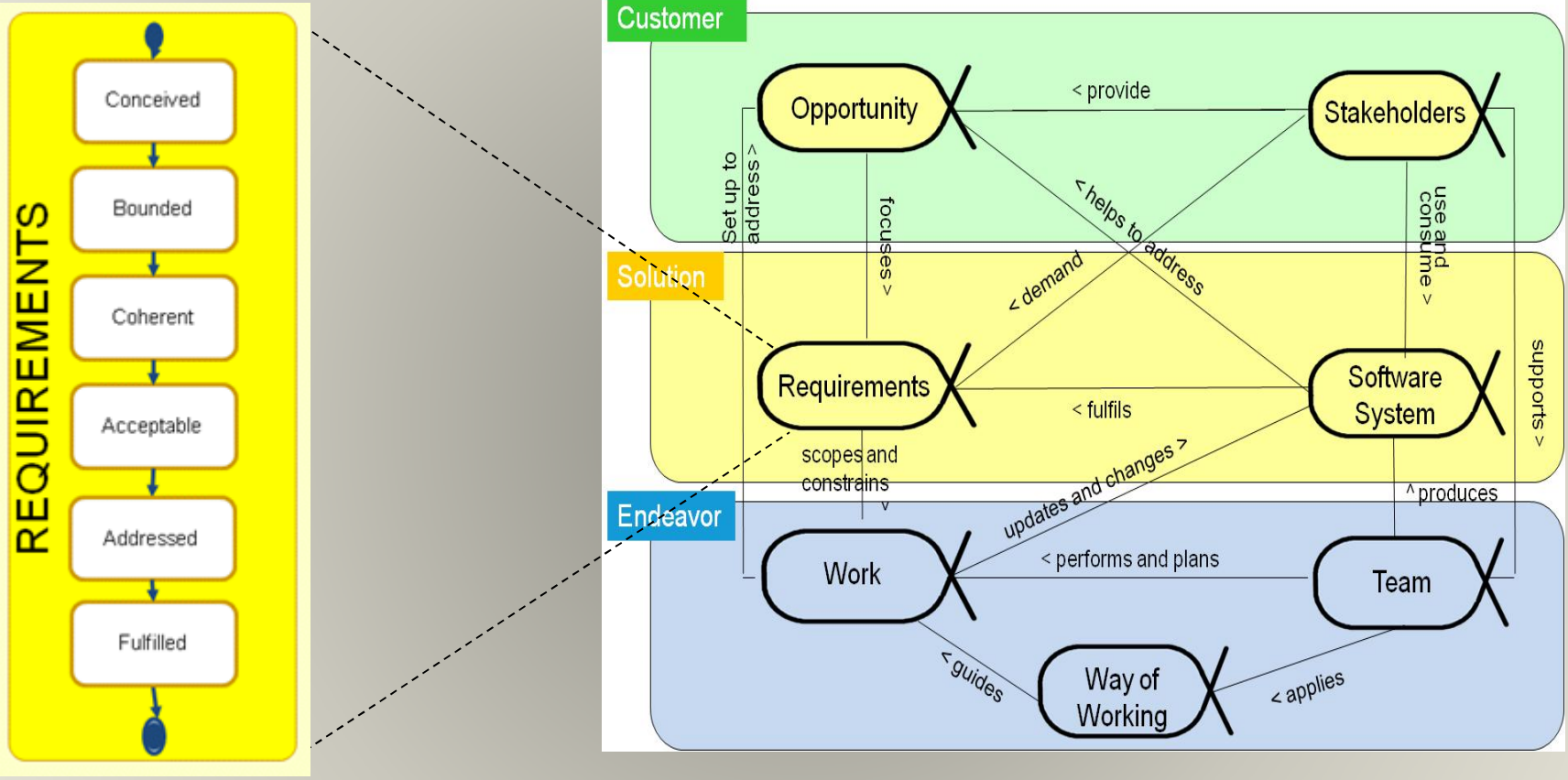


- Captures the essence of software engineering with the aid of essential properties called Alphas (**A**bstract-**L**evel **P**rogress **H**ealth **A**ttribute).

Essence Kernel Alphas

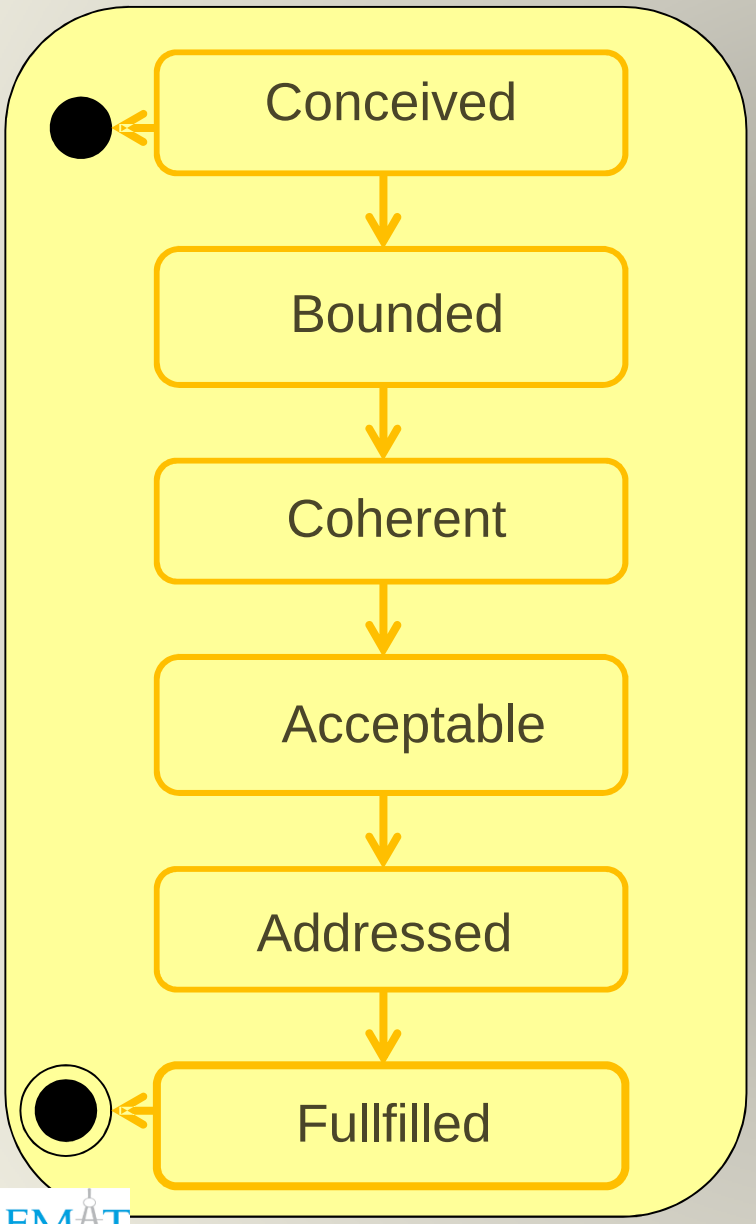


Requirements– one of the Alphas



Requirements Definition: What the software system must do to address the opportunity and satisfy the stakeholders.

The states of Requirements



The need for a new system has been agreed.

The purpose and theme of the new system are clear.

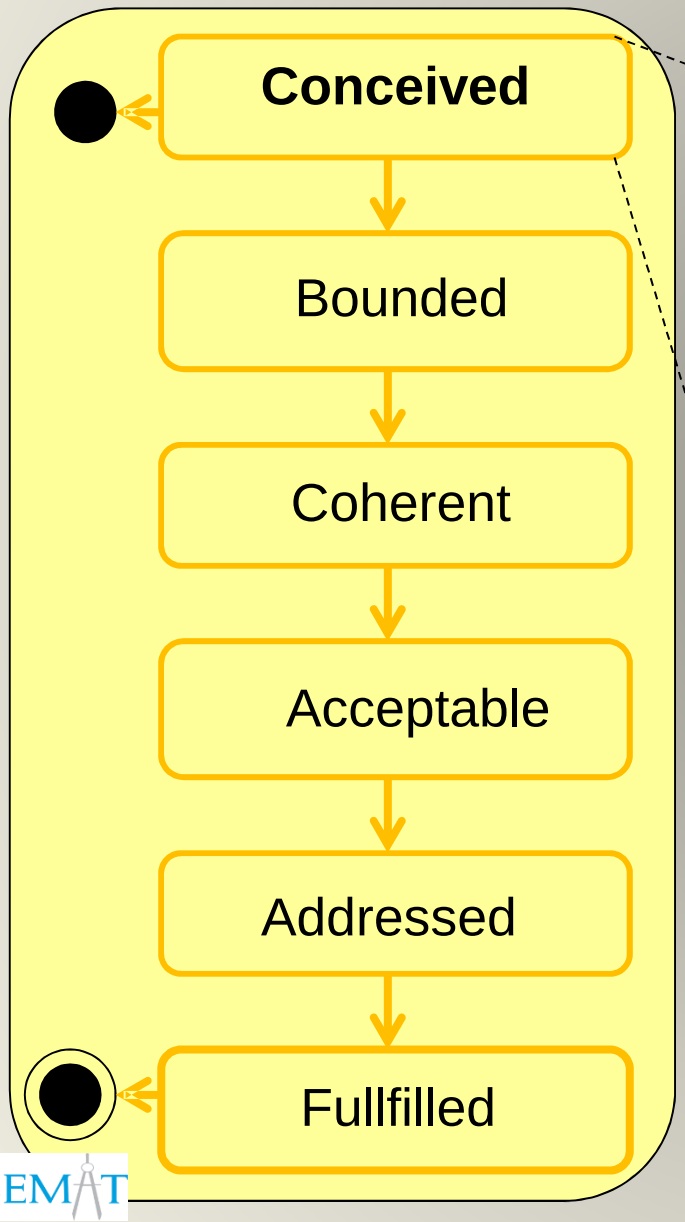
The requirements provide a coherent description of the essential characteristics of the new system.

The requirements describe a system that is acceptable to the stakeholders.

Enough of the requirements have been addressed to satisfy the need for a new system in a way that is acceptable to the stakeholders.

The requirements have been addressed to fully satisfy the need for a new system.

Checklist for achieving the Conceived state of Requirements



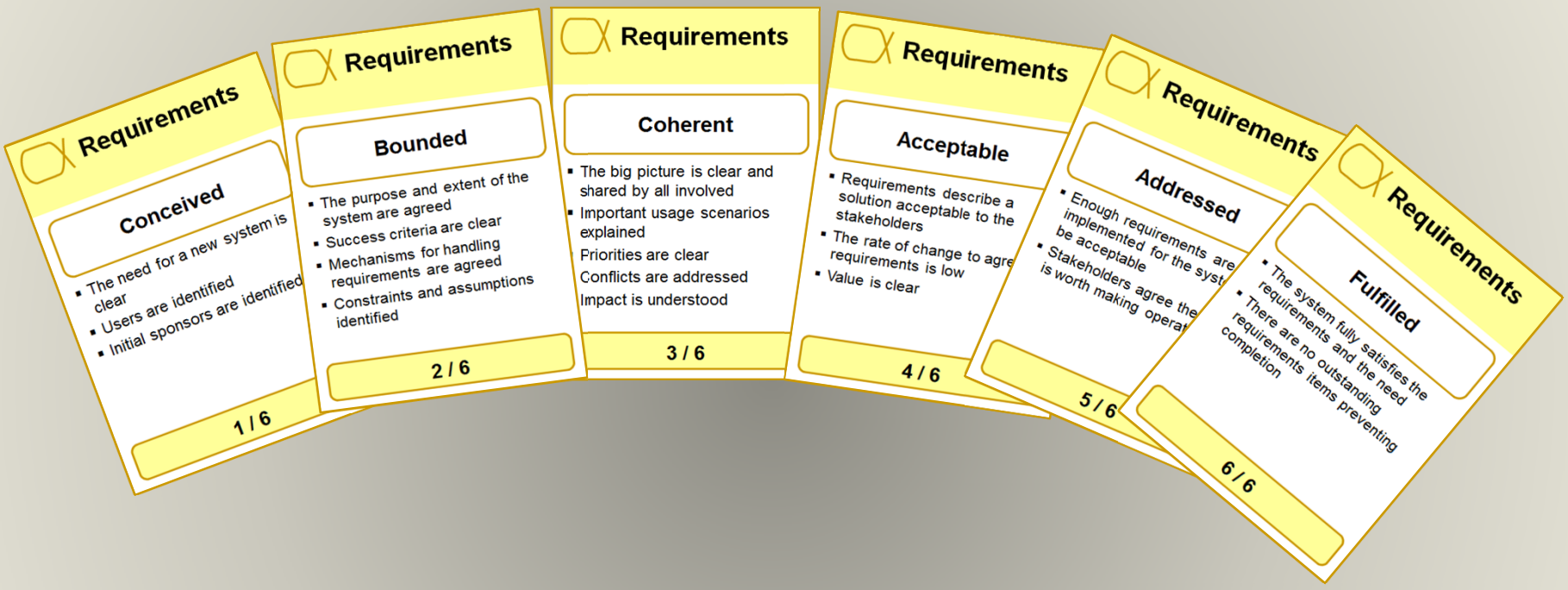
- ❑ The initial set of stakeholders agrees that a system is to be produced.
- ❑ The stakeholders that will use the new system are identified.
- ❑ The stakeholders that will fund the initial work on the new system are identified.
- ❑ There is a clear opportunity for the new system to address.

SEMAT

An example of a checklist for the Stakeholder Alpha

State	Checklist
Recognized	All the different groups of stakeholders that are, or will be, affected by the development and operation of the software system are identified. There is agreement on the stakeholder groups to be represented. At a minimum, the stakeholders groups that fund, use, support, and maintain the system have been considered. The responsibilities of the stakeholder representatives have been defined.
Represented	The stakeholder representatives have agreed to take on their responsibilities. The stakeholder representatives are authorized to carry out their responsibilities. The collaboration approach among the stakeholder representatives has been agreed. The stakeholder representatives support and respect the team's way of working.
Involved	The stakeholder representatives assist the team in accordance with their responsibilities. The stakeholder representatives provide feedback and take part in decision making in a timely manner. The stakeholder representatives promptly communicate changes that are relevant for

Essence Cards



- Each Alpha is materialized in a set of cards.
- One card represents one state of the Alpha.

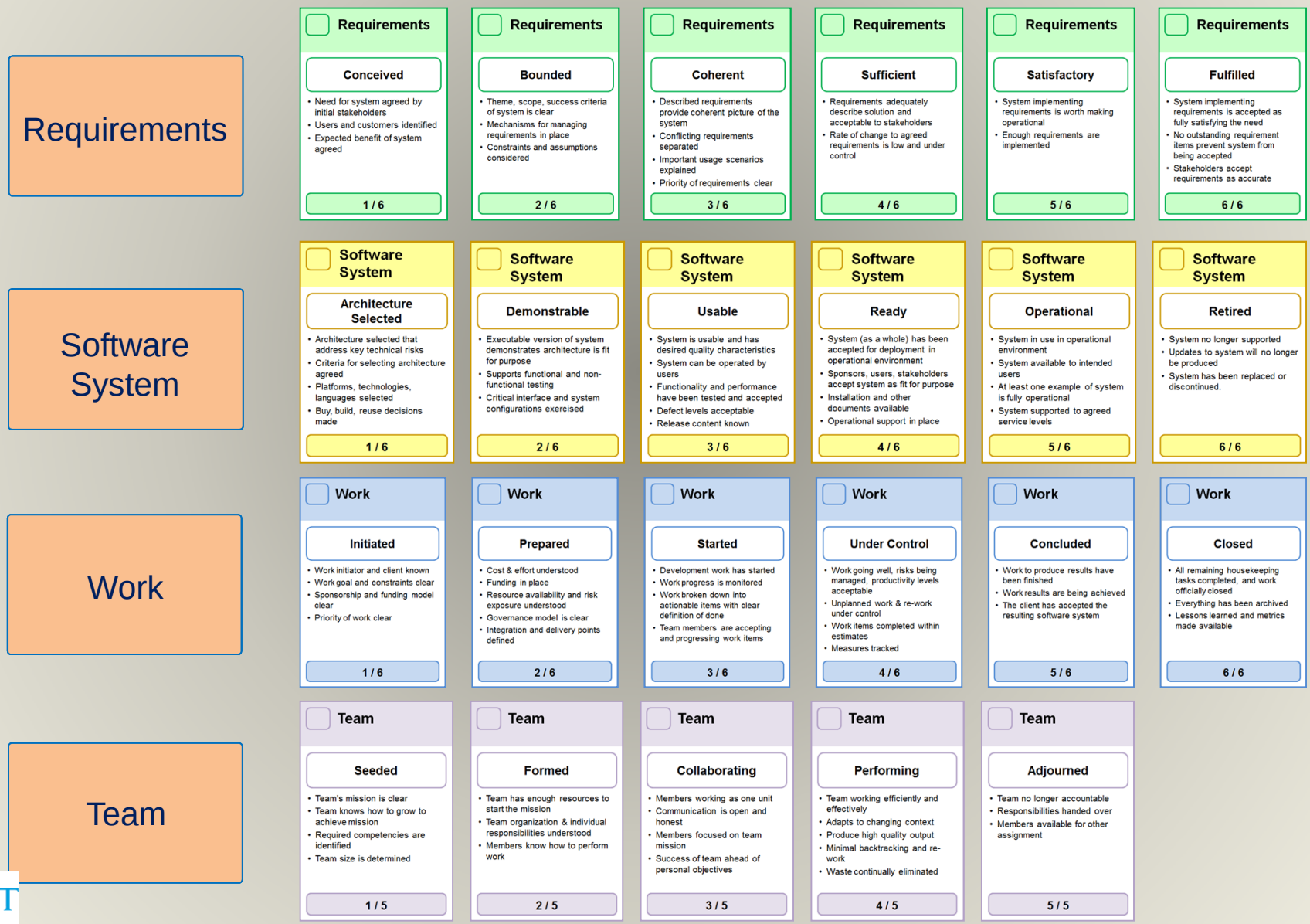
Abbreviated checklist items

Recognized: Stakeholders have been identified.	
☐ Stakeholder groups identified	All the different groups of stakeholders that are, or will be, affected by the development and operation of the software system are identified.
☐ Key stakeholder groups represented	There is agreement on the stakeholder groups to be represented. At a minimum, the stakeholder groups that fund, use, support, and
☐	The responsibilities of the stakeh
	lving the stakeholders are a
	The stakehold
	stakeh
	col
	The st
	es are active
☐	The sta
☐ Timely feedback and decisions provided	The stake timely mann
☐ Changes promptly communicated	The stakeholder stakeholder groups.
In Agreement: The stakeholder representatives are in agreement.	
☐ Minimal expectations agreed	The stakeholder representatives have agreed upon their deployment of the new system.
☐ Rep's happy with their involvement	The stakeholder representatives are happy with their invo lvement in the work.
☐ Rep's input valued	The stakeholder representatives agree that their input is valued by the team and treated with respect.
☐ Team's input valued & respected	The team members agree that their input is valued by the stakeholder representatives and treated with respect.
☐ Priorities clear & perspectives balanced	The stakeholder repre sentatives agree with how their different priorities and perspectives are being balanced to provide a clear direction for the team.
Satisfied for Deployment: The minimal expectations of the stakeholder representatives have been achieved.	
☐ Stakeholder feedback provided	The stakeholder representatives provide feedback on the system from their stakeholder group perspective.
☐ System ready for deployment	The stakeholder representatives confirm that they agree that the system is ready for deployment.
Satisfied in Use: The system has met or exceeds the minimal stakeholder expectations.	
☐ Feedback on system use available	Stakeholders are using the new system and providing feedback on their experiences.
☐ System meets expectations	The stakeholders confirm that the n ew system meets their expectations.

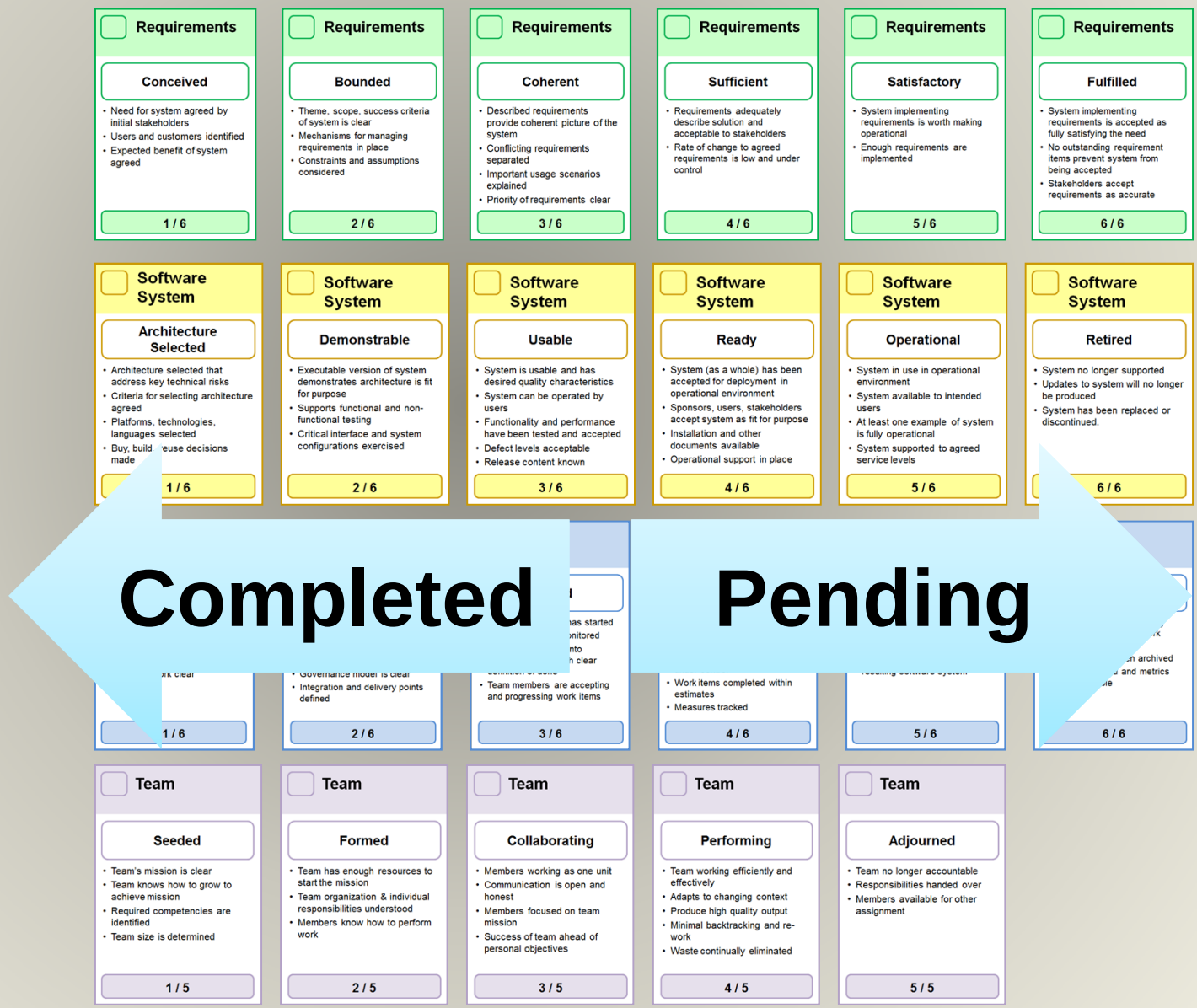
Stakeholder groups identified

All the different groups of stakeholders that are, or will be, affected by the development and operation of the software system are identified.

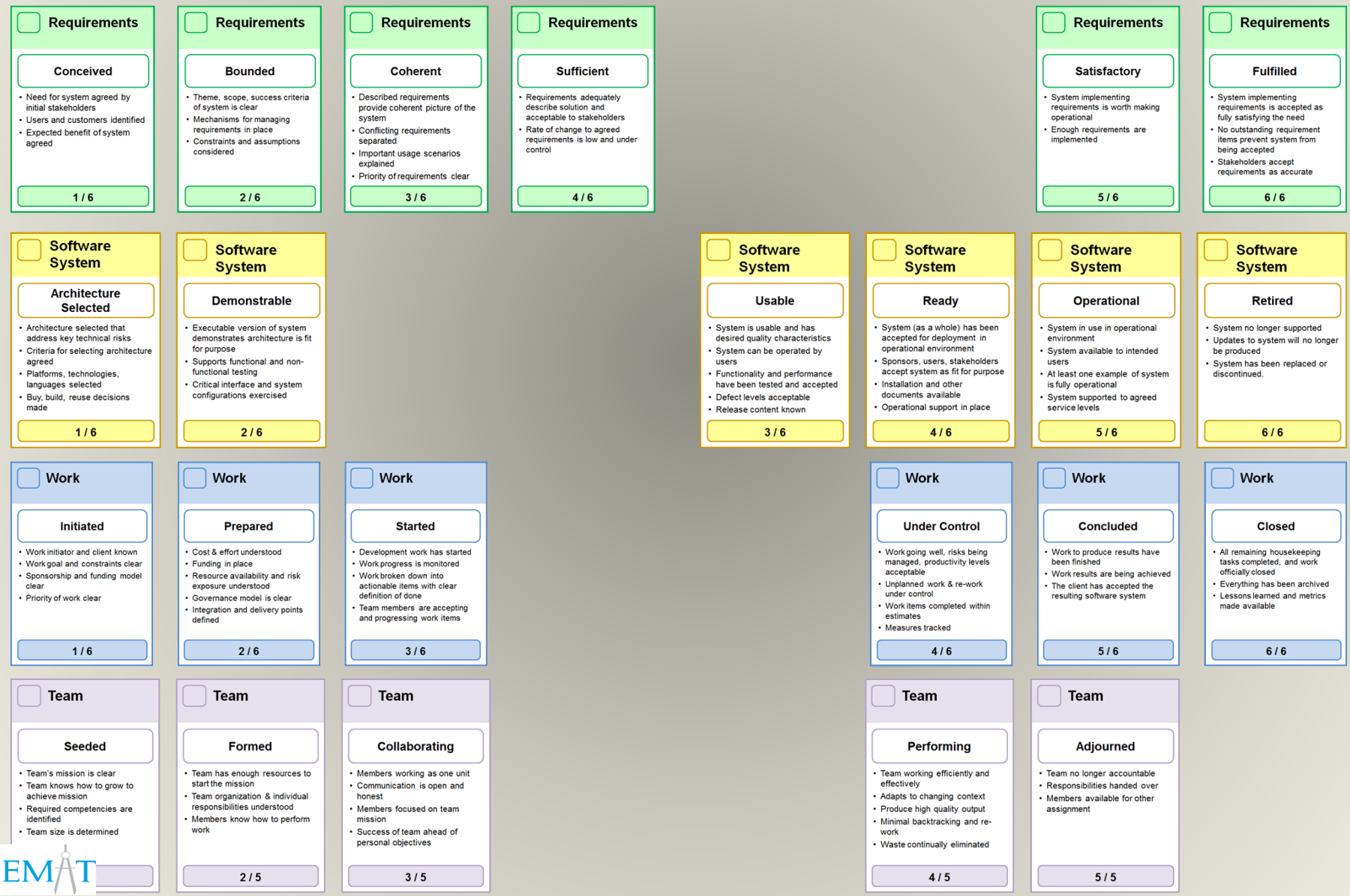
Working with the Alphas



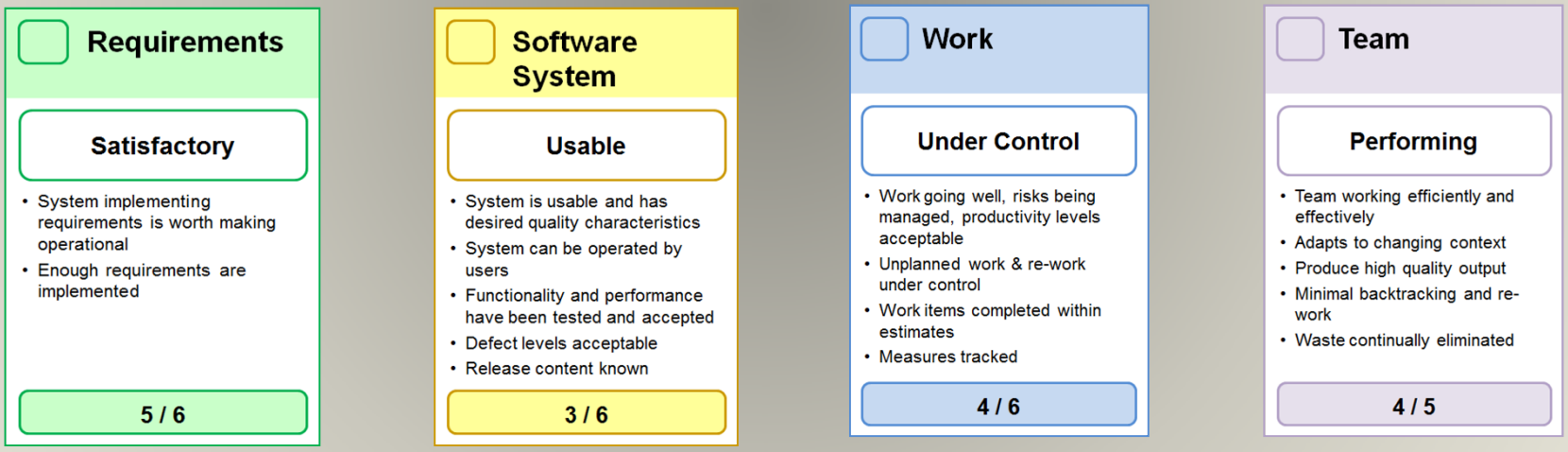
Determine Current State



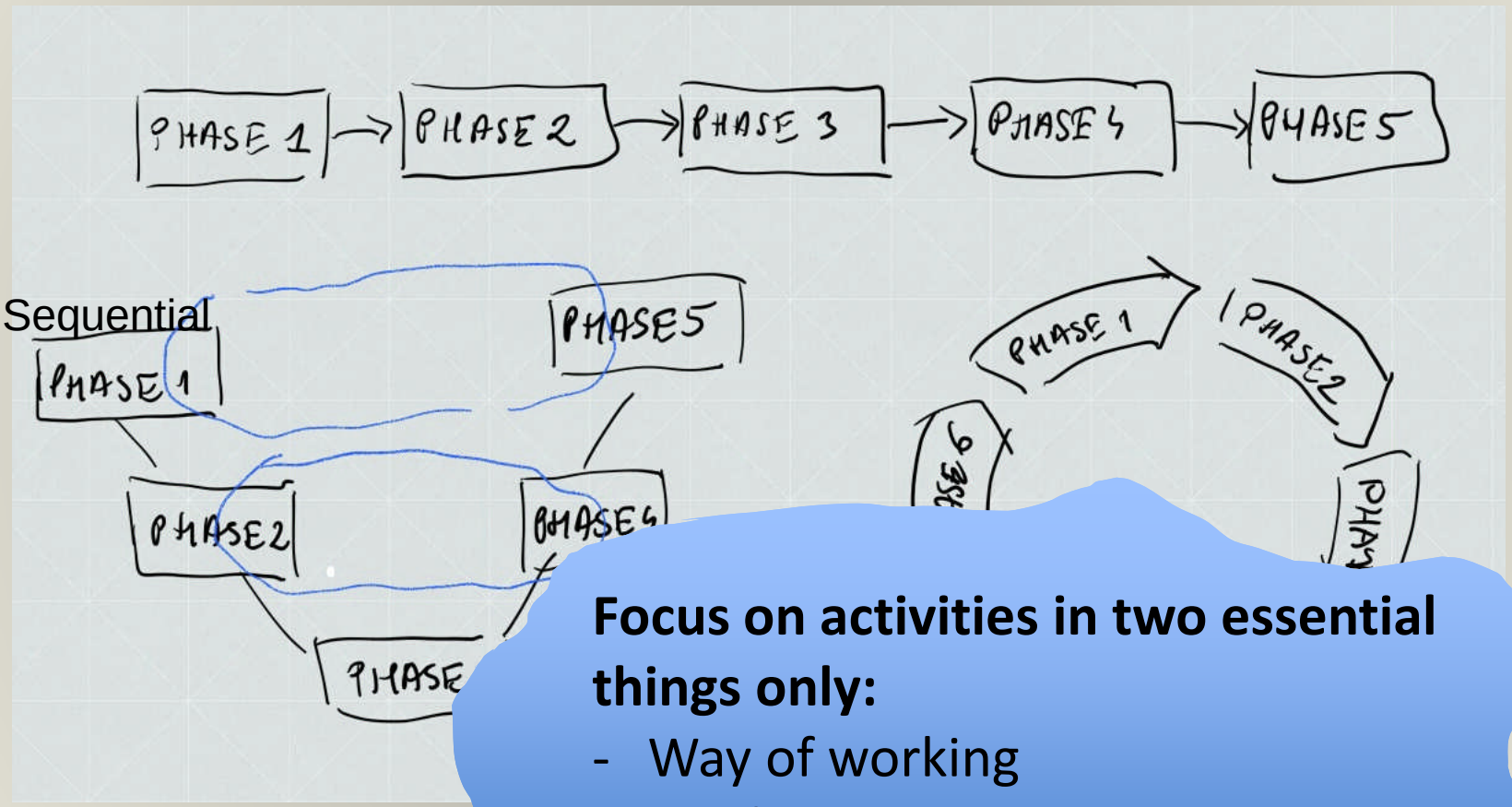
Determine Next State



Determine How to Achieve Next State



Some software development methods today



Focus on activities in two essential things only:

- Way of working
- Work

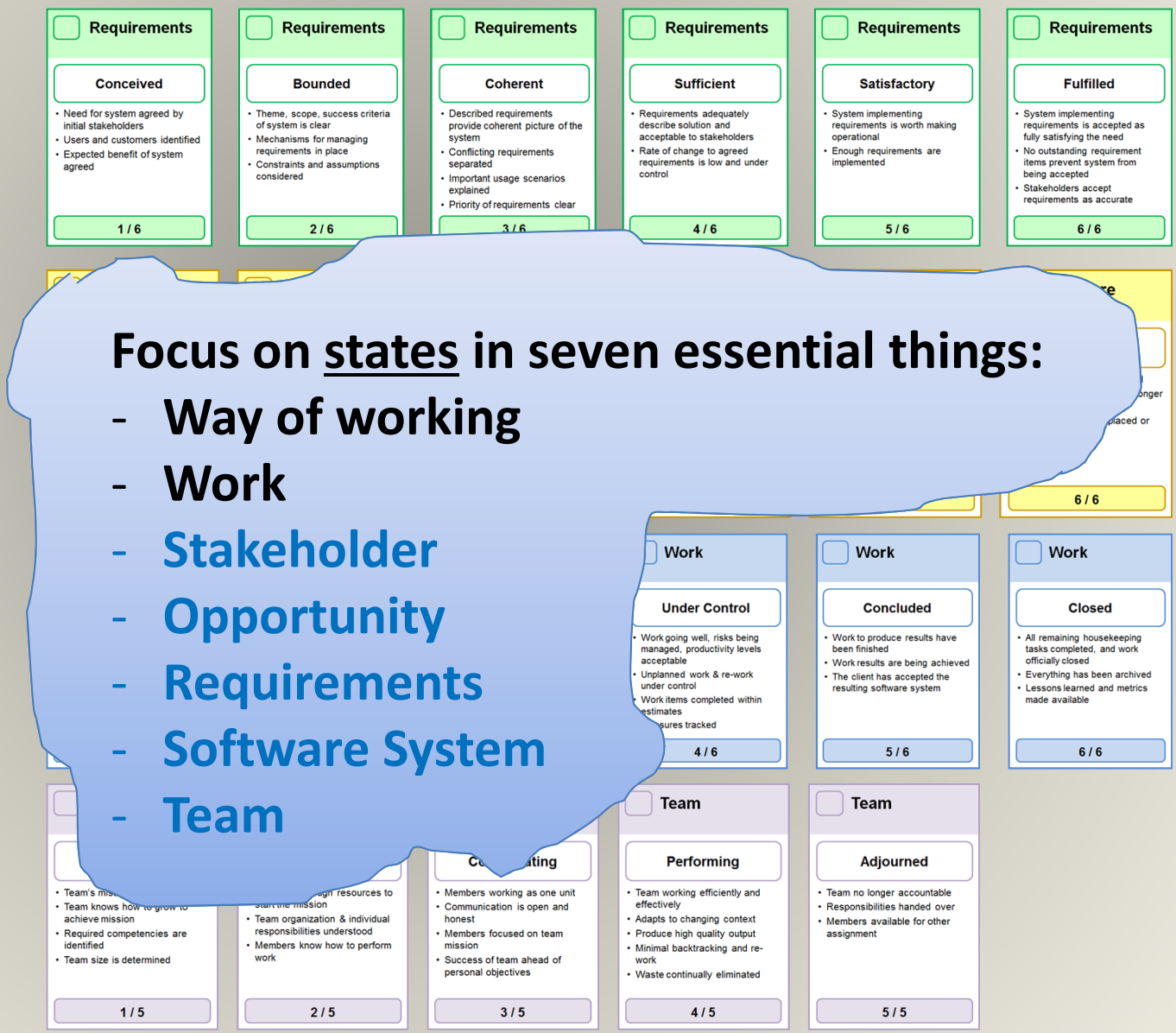
Essence Kernel

requirements

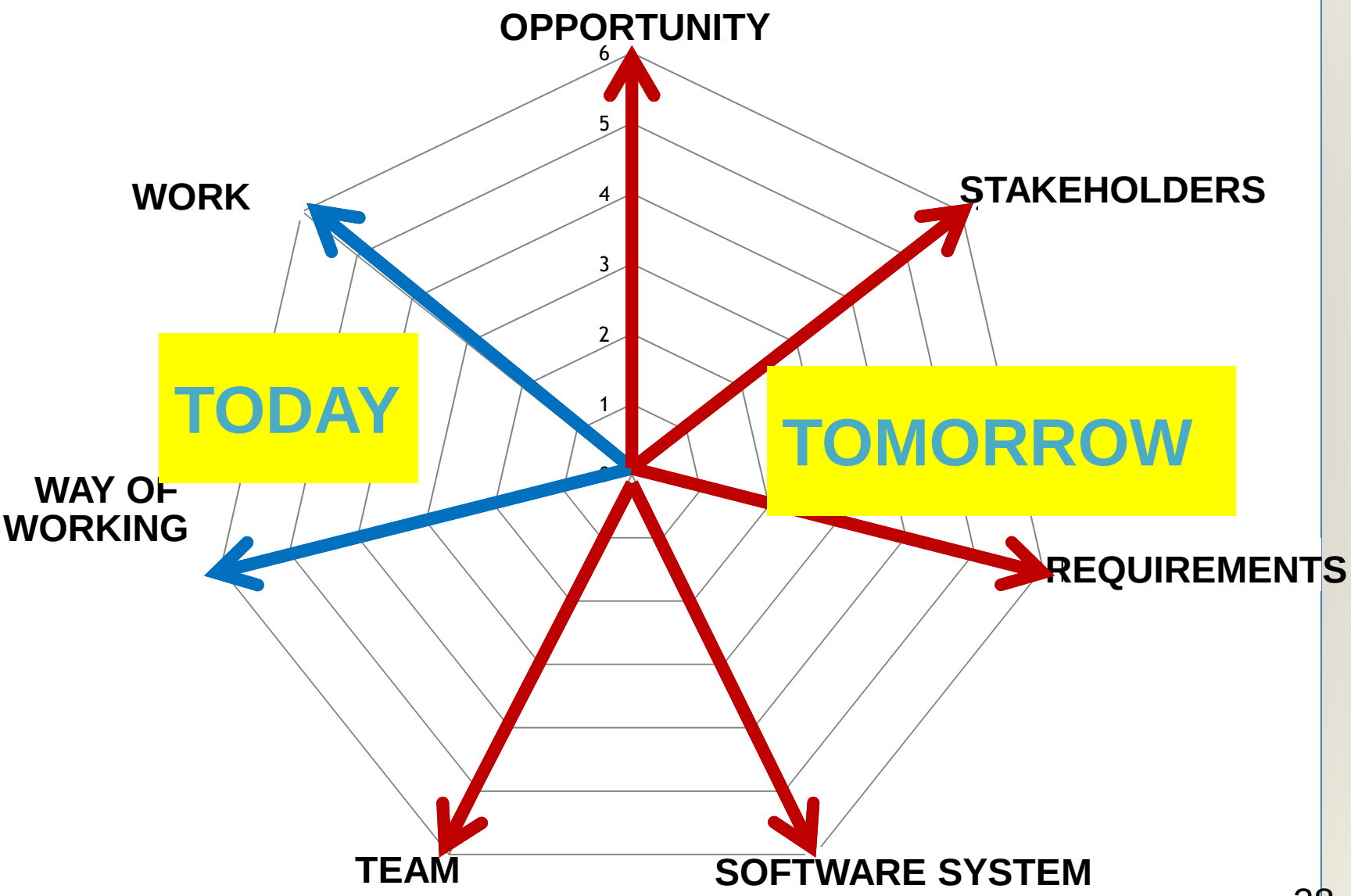
software system

work

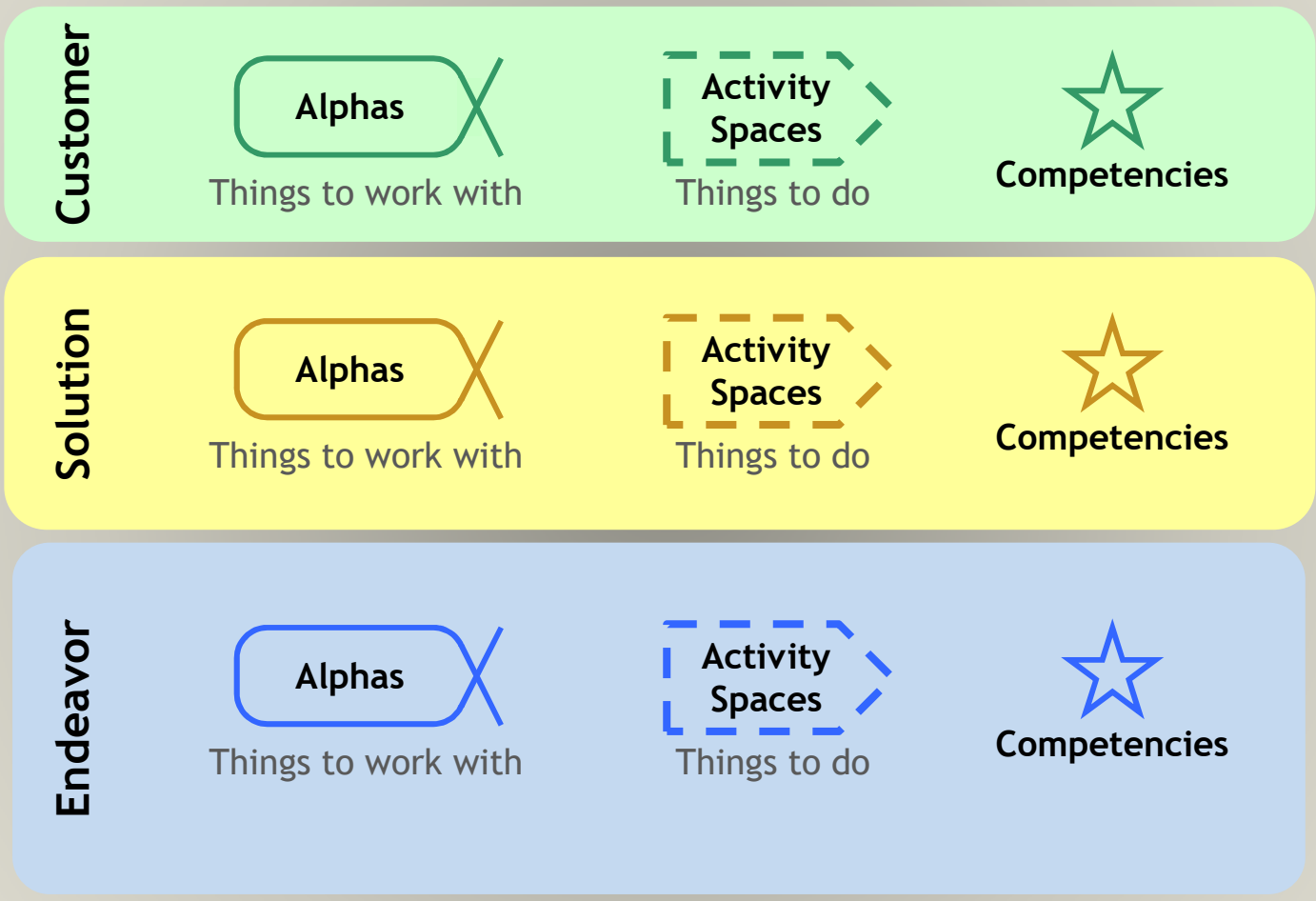
team



Following essential things



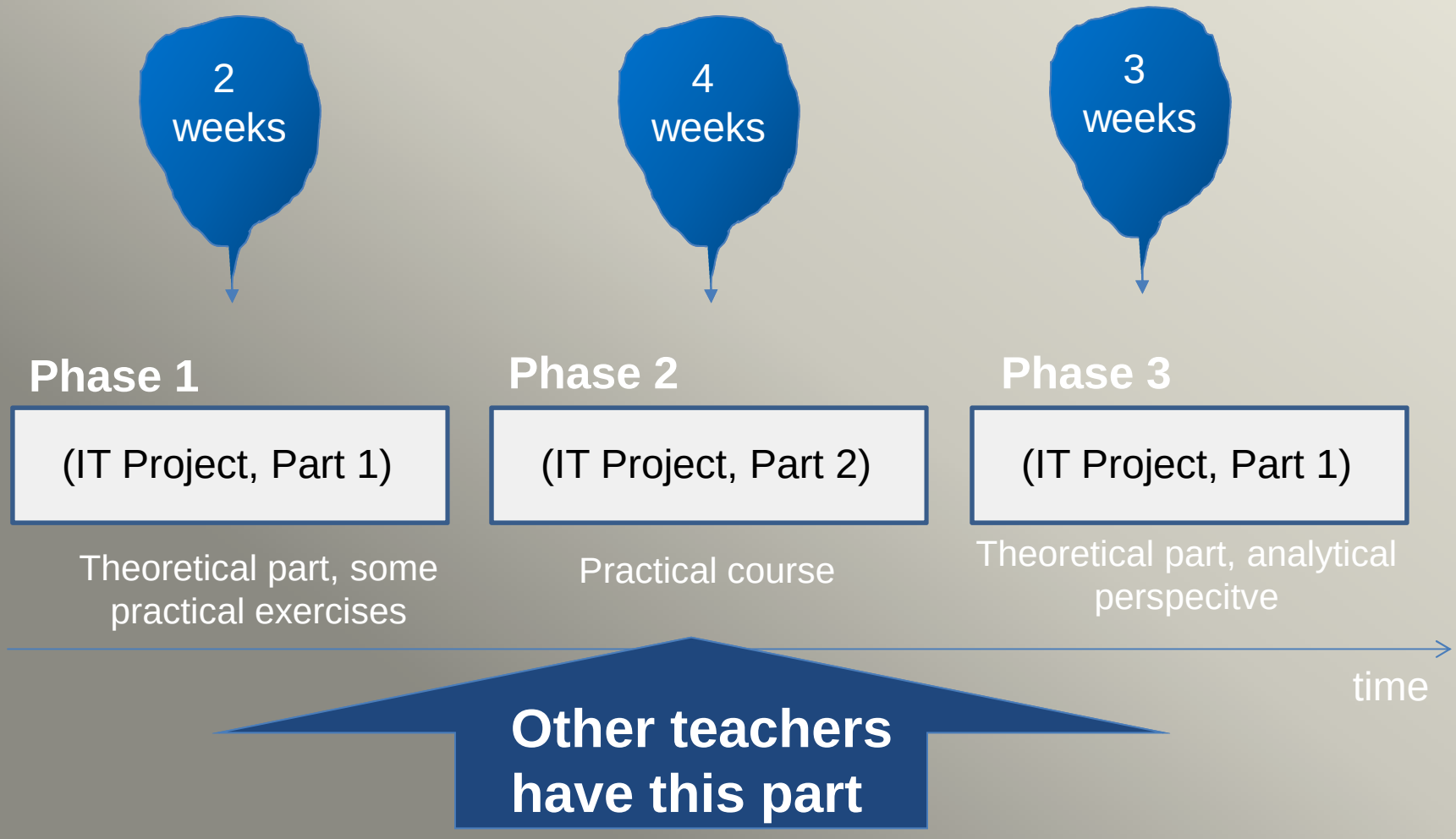
Essence Kernel



Outline

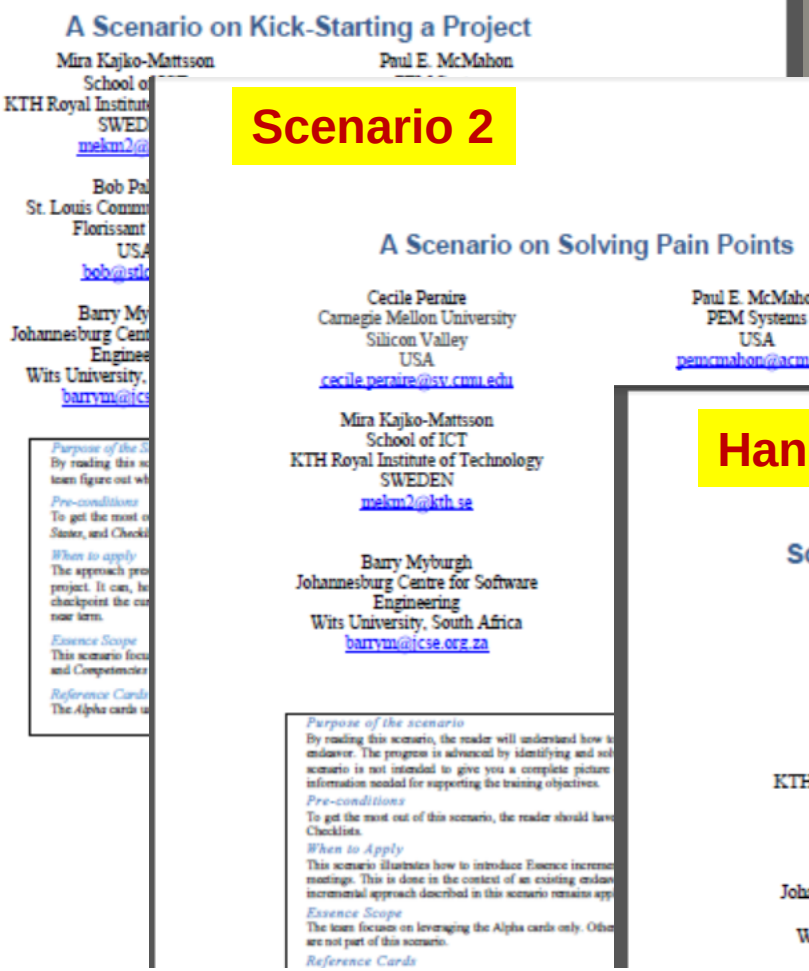
- Problems within education
- The SEMAT community
- ESSENCE
- SE education at KTH
- So how many birds….

IT-Project Course at KTH



Educational material

Scenario 1



A Scenario on Kick-Starting a Project

Mira Kajko-Mattsson
School of
KTH Royal Institute
SWED
mekm2@

Paul E. McMahon

Bob Pa
St. Louis Comm
Florissant
USA
bob@stlc

Barry My
Johannesburg Cent
Enginee
Wits University,
barrym@jcs

Scenario 2

A Scenario on Solving Pain Points

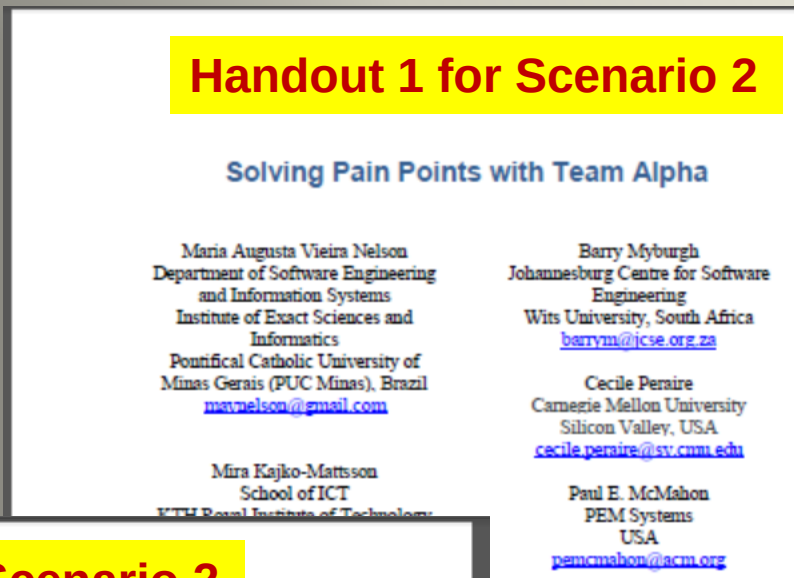
Cecile Peraire
Carnegie Mellon University
Silicon Valley
USA
cecile.peraire@sv.cmu.edu

Paul E. McMahon
PEM Systems
USA
pemcmahon@acm.org

Mira Kajko-Mattsson
School of ICT
KTH Royal Institute of Technology
SWEDEN
mekm2@kth.se

Barry Myburgh
Johannesburg Centre for Software Engineering
Wits University, South Africa
barrym@jcse.org.za

Handout 1 for Scenario 2



Solving Pain Points with Team Alpha

Maria Augusta Vieira Nelson
Department of Software Engineering and Information Systems
Institute of Exact Sciences and Informatics
Pontifical Catholic University of Minas Gerais (PUC Minas), Brazil
mavnelson@gmail.com

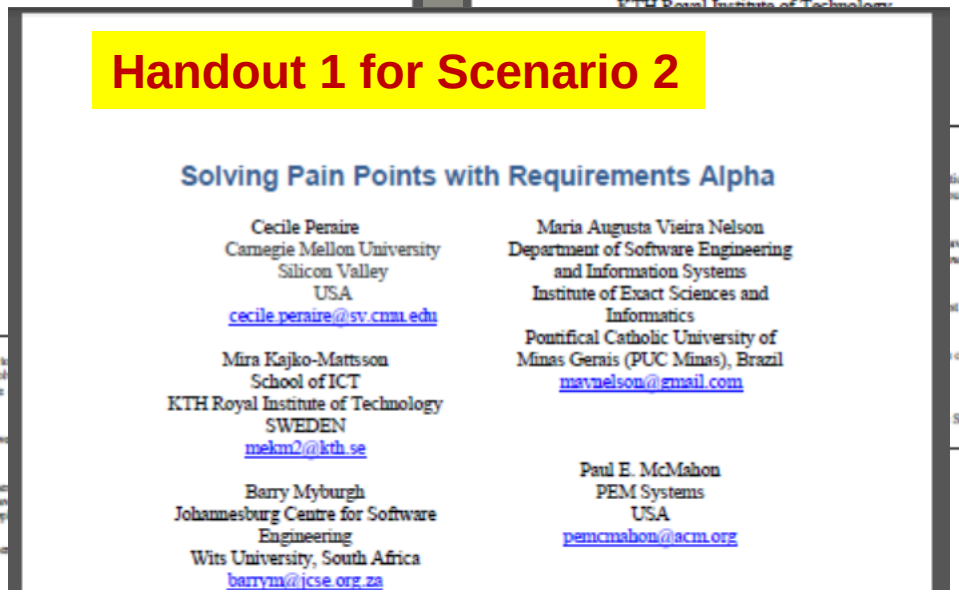
Barry Myburgh
Johannesburg Centre for Software Engineering
Wits University, South Africa
barrym@jcse.org.za

Cecile Peraire
Carnegie Mellon University
Silicon Valley, USA
cecile.peraire@sv.cmu.edu

Mira Kajko-Mattsson
School of ICT

Paul E. McMahon
PEM Systems
USA
pemcmahon@acm.org

Handout 1 for Scenario 2



Solving Pain Points with Requirements Alpha

Cecile Peraire
Carnegie Mellon University
Silicon Valley
USA
cecile.peraire@sv.cmu.edu

Maria Augusta Vieira Nelson
Department of Software Engineering and Information Systems
Institute of Exact Sciences and Informatics
Pontifical Catholic University of Minas Gerais (PUC Minas), Brazil
mavnelson@gmail.com

Mira Kajko-Mattsson
School of ICT
KTH Royal Institute of Technology
SWEDEN
mekm2@kth.se

Paul E. McMahon
PEM Systems
USA
pemcmahon@acm.org

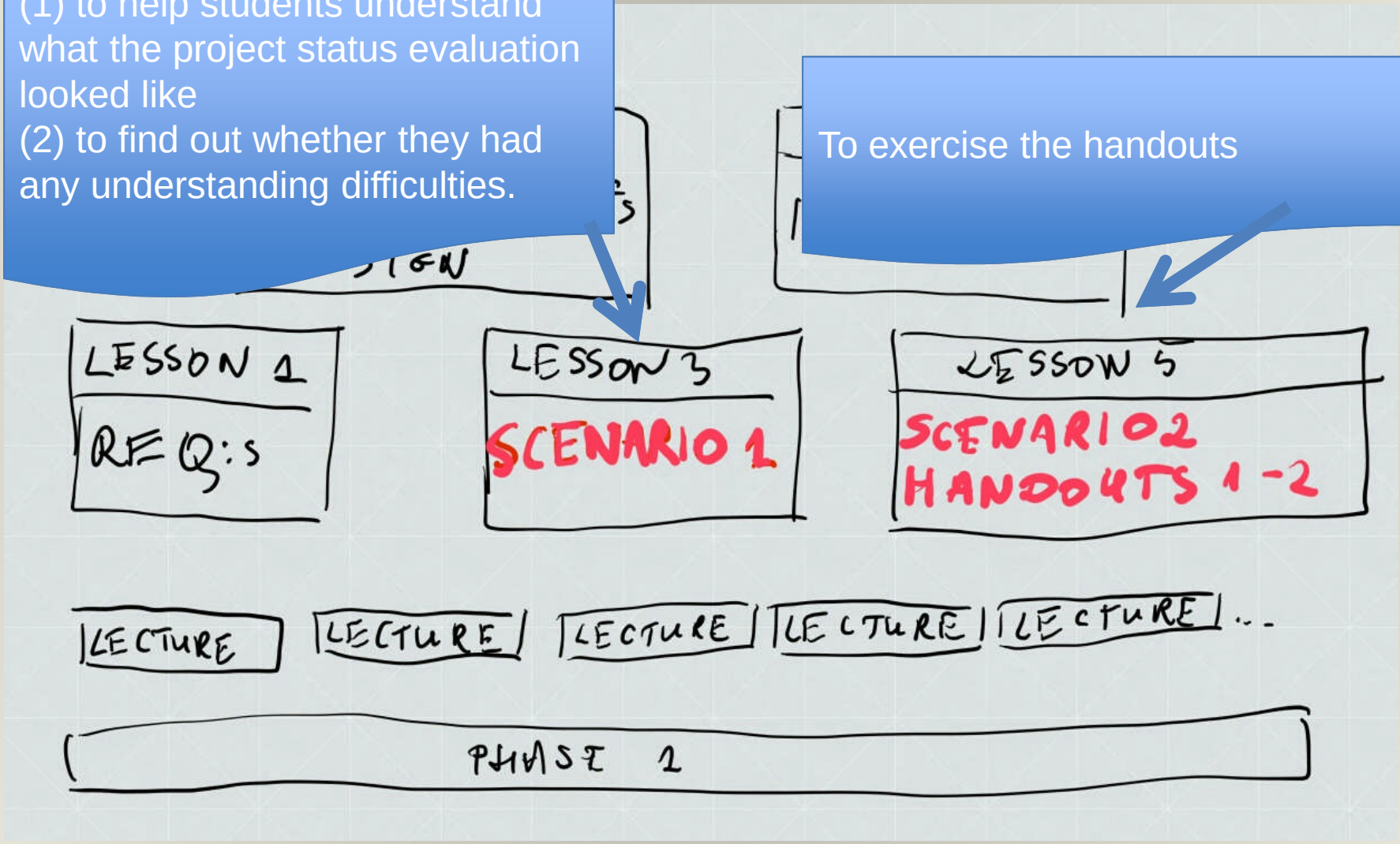
Barry Myburgh
Johannesburg Centre for Software Engineering
Wits University, South Africa
barrym@jcse.org.za

IT-Project Course at KTH

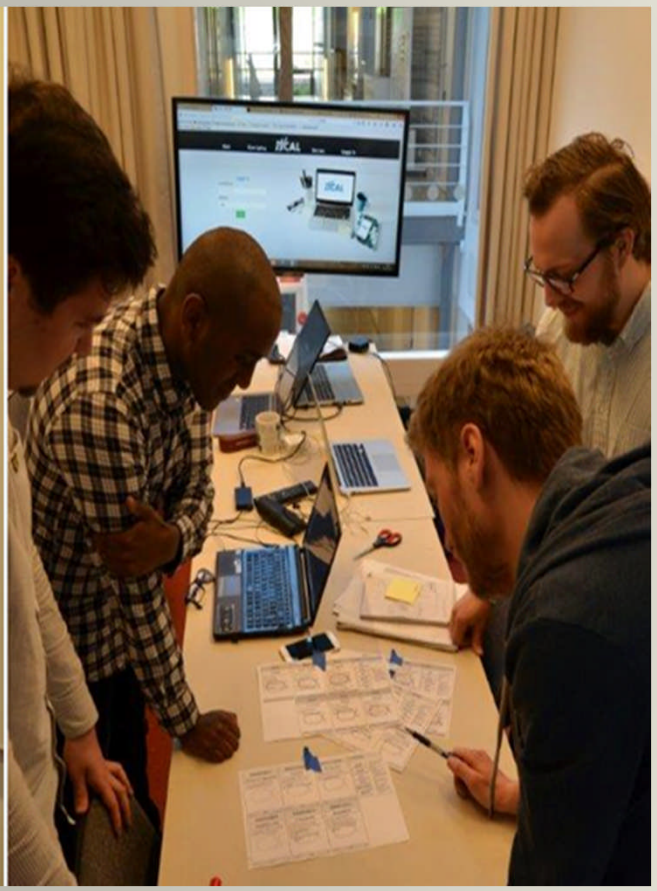
Year 2015

(1) to help students understand what the project status evaluation looked like
(2) to find out whether they had any understanding difficulties.

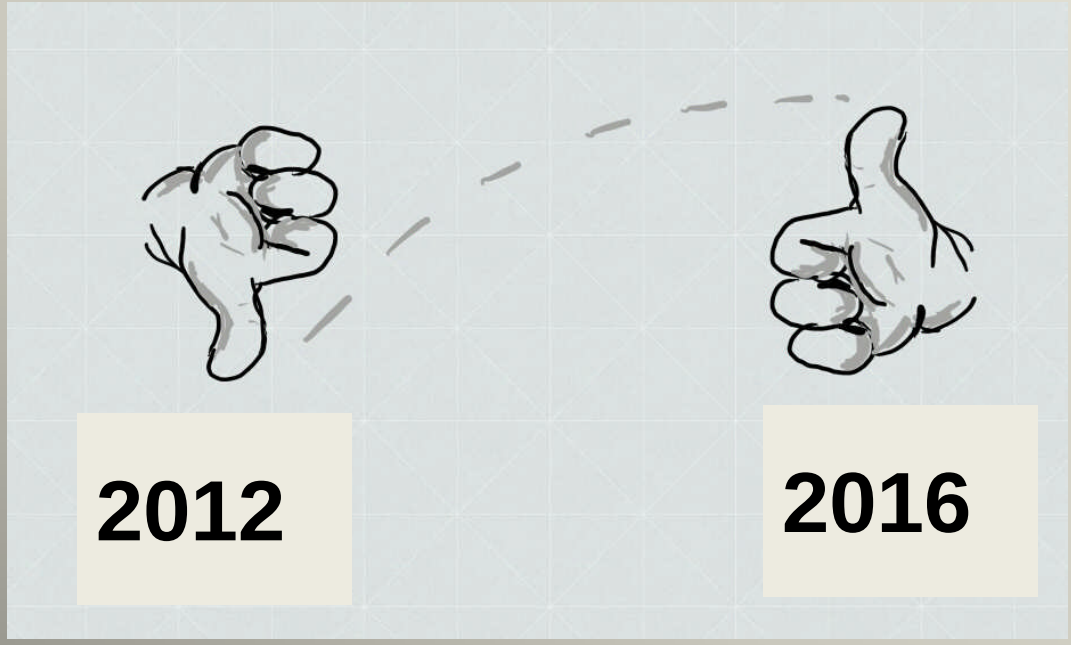
To exercise the handouts



Year 2015



Students' attitued towards ESSENCE



- 60 % of the students of the year 2012 were not directly positive.
- They treated it as a burden and as an unnecessary new method.
- They expressed that they felt like guinea pigs.
- Now, about 80-90% of the students are positive.

Some opinions about ESSENCE

Intuitively understood

superior to other methods thanks to its full coverage of the essential things

provides a stable platform to stand on

enables distribution of work

assurance of project quality

time to learn ESSENCE is well invested

makes project more visible

facilitates project communication

useful in projects lacking any methods

Having a list of items to be checked off was definitely much better than having nothing and trying to figure out what to do next!

Outline

- Problems within education
- The SEMAT community
- ESSENCE
- SE education at KTH
- So how many birds….

So how many birds

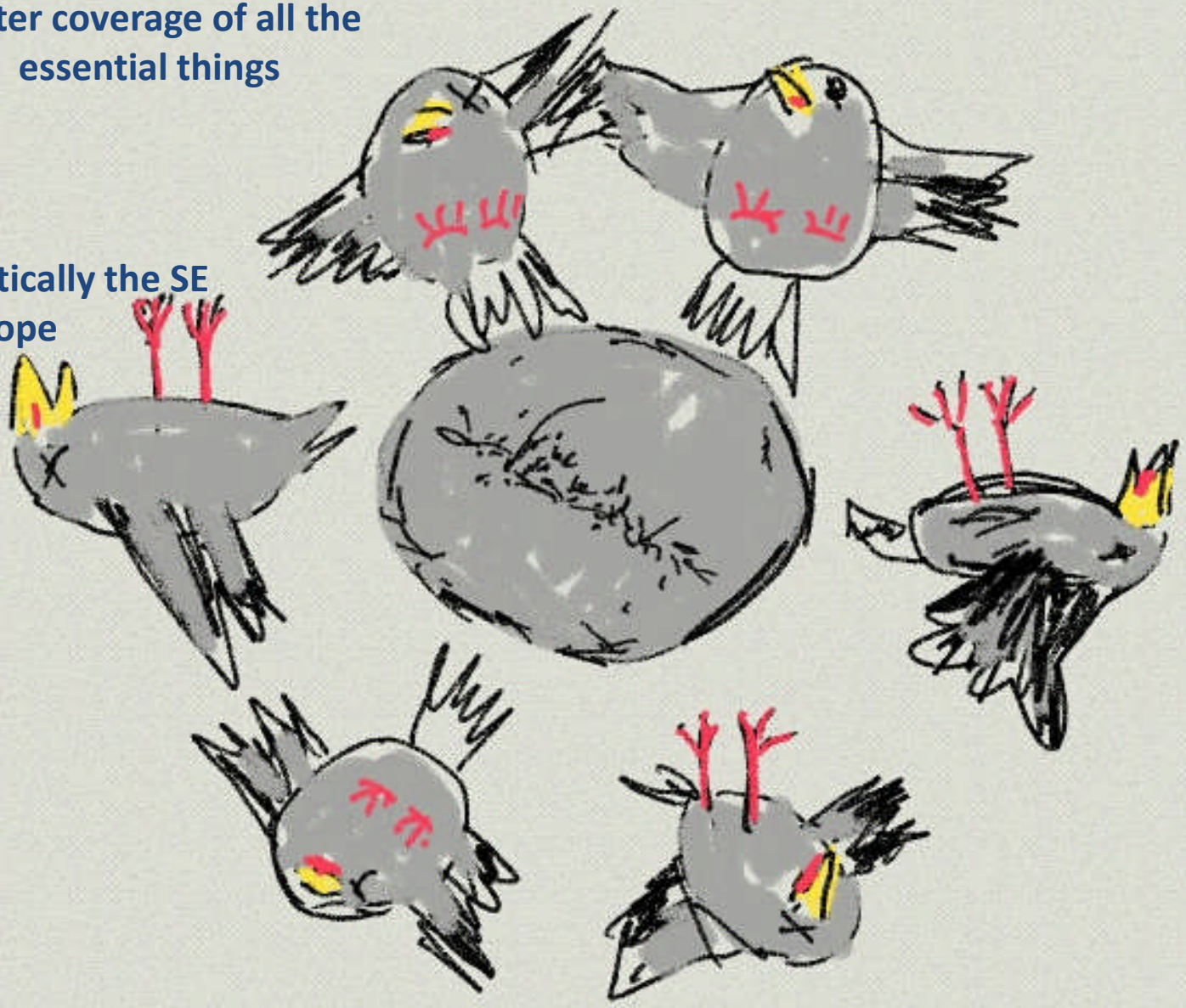
Complaints to the English
for their weird expression.

The Swedes are kinder. They only kill flies.

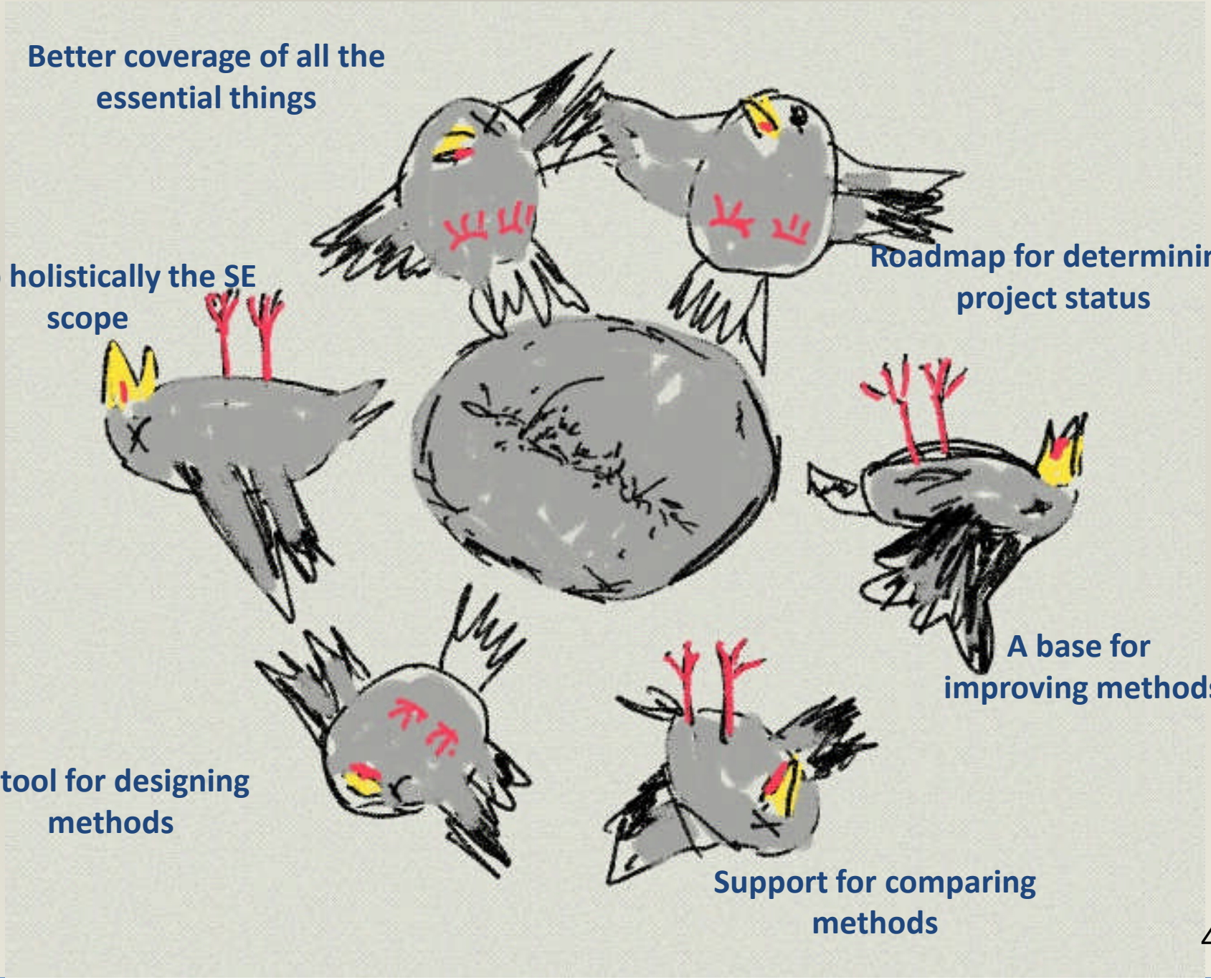
can we kill with one stone!

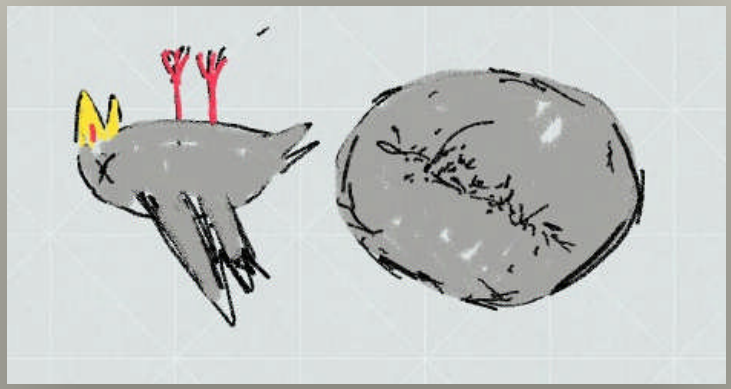
Better coverage of all the essential things

Grasp holistically the SE scope



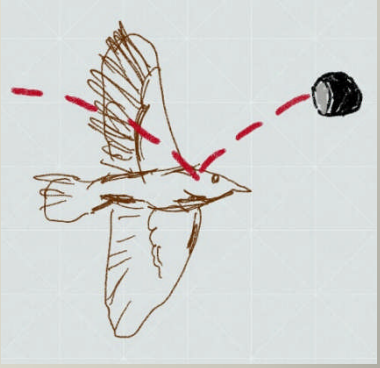
SEMAT


Better coverage of all the essential things
Grasp holistically the SE scope
Roadmap for determining project status
A base for improving methods
A tool for designing methods
Support for comparing methods

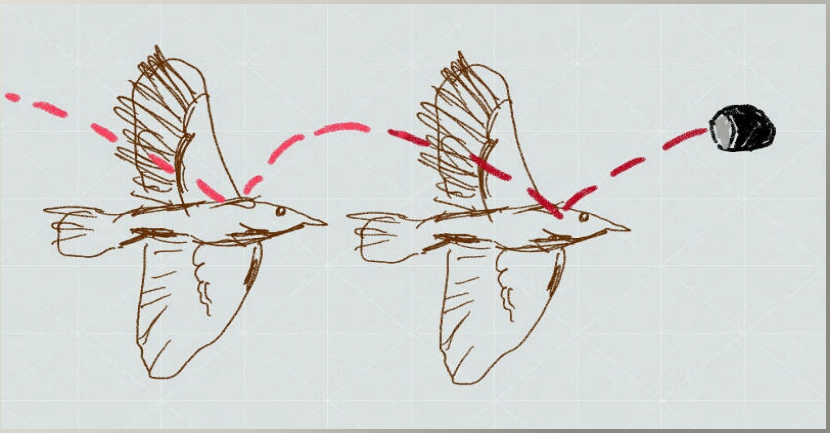


We are not ready yet!

We are

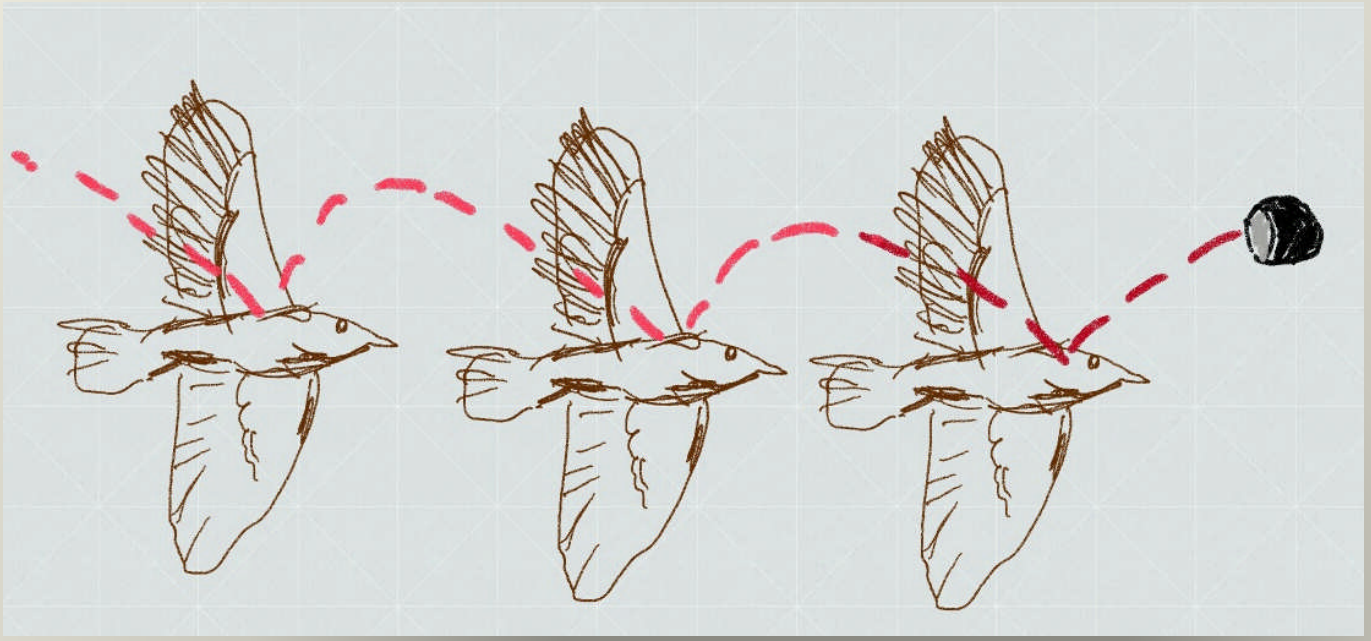


List for planning the work



List for planning the work

Aid in identifying and assessing risks



List for planning the work

Query for guidance

Aid in identifying and assessing risks

- Identify gaps in competencies
- Support tool building
- Structure for communication
- Template for teaching software engineering
- and other.

All this can be done in a holistic, simple, lightweight, non-prescriptive and method-agnostic fashion

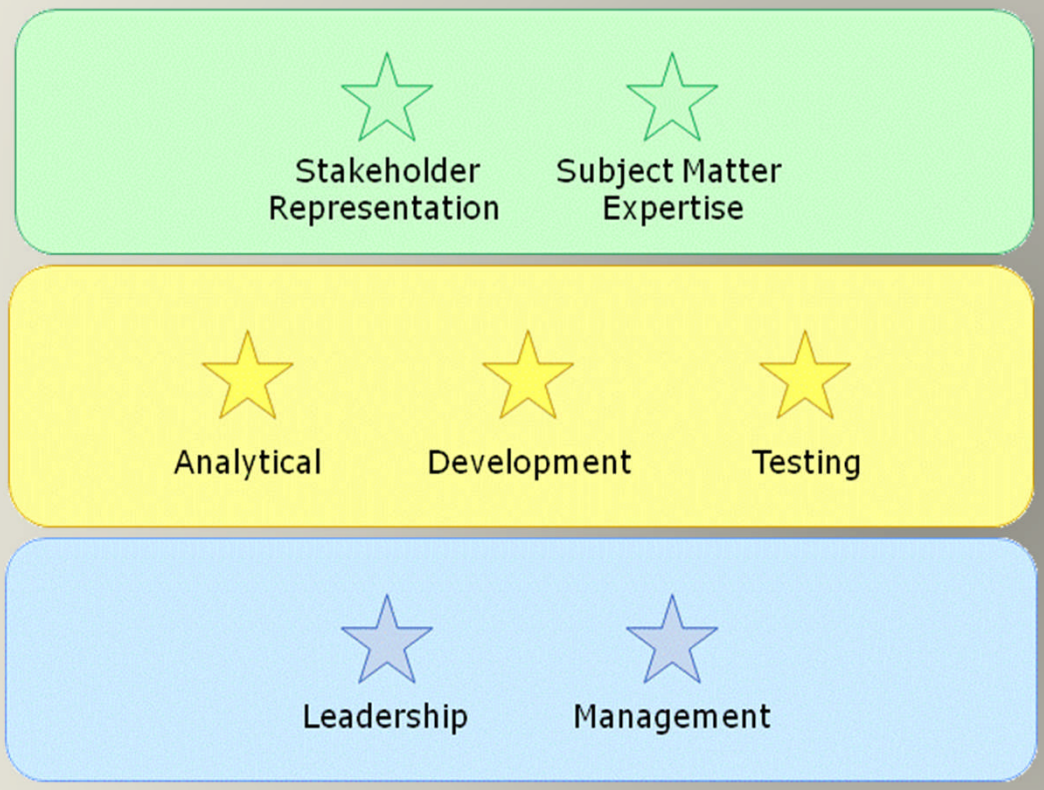
Future

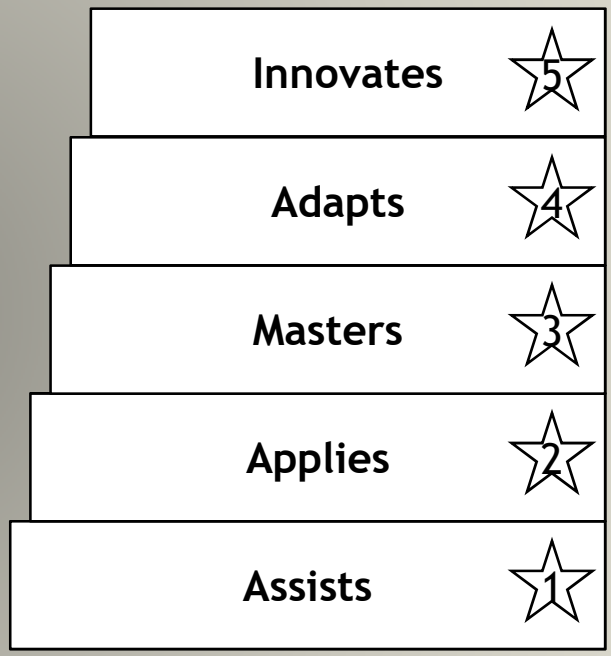
- ESSENCE is an excellent tool for squeezing software engineering education within a short period of time, even on an undergraduate level.
- Continue using ESSENCE within the education.
- Continue to develop educational material.

Anybody interested in cooperating with us?

SEMAT

Competencies





View of key competencies needed in software engineering