Contribution of AUEB to OPTET

Costas Kalogiros, Michael Kanakakis
Athens University of Economics and Business

OPTET Kick-off Meeting
12th-13th November 2012

Network Economics and Services Group

Athens University of Economics and Business

http://nes.aueb.gr

Costas Courcoubetis (Professor)

Costas Kalogiros (Phd)

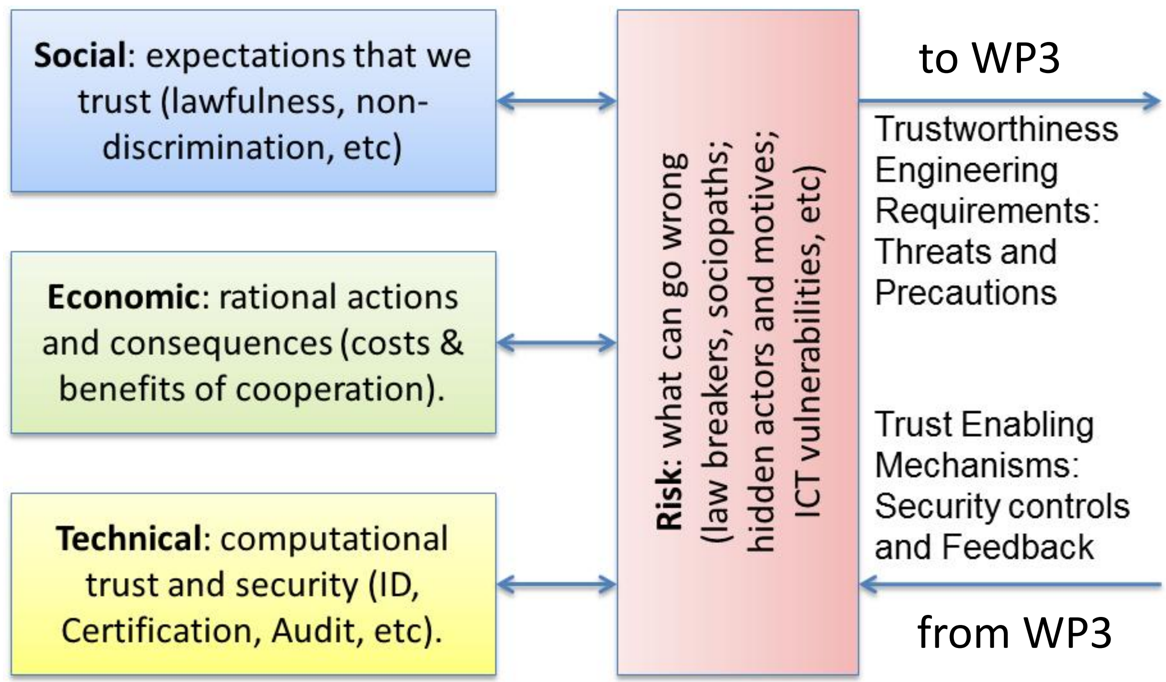
Michael Kanakakis (MSc)

High-level Contribution of AUEB

- Task 2.2: What is the economic basis for trust?
  - Will develop models of the economic risks and benefits on the Internet allowing the behaviour of rational actors to be predicted and outcomes to be evaluated.
- Task 3.3: How to assess the trustworthiness of the target **collaborative** system?
  - How trustworthiness metrics should be defined and monitored?
- Task 6.1: What business processes are effective for Trust Management (at run-time)?
  - to study trust compromising activities and trust recovery actions
  - to create a set of business processes for the measurement and management of trust and trustworthiness
- Task 8.1: Development of use cases and application concepts
  - Provide details for the use cases (value networks, contribution to stakeholders, stakeholder concerns, etc.)

WP2 - Socio-economics of Trust

Task 2.2
(Economic basis for trust)



- Will develop models of the economic risks and benefits on the Internet allowing:
  - the behaviour of rational actors to be predicted, i.e. when should people and businesses trust each other, and what incentives could ensure this?
  - the effectiveness of proposed trust-enabling technologies to be evaluated

Intro to Game Theory

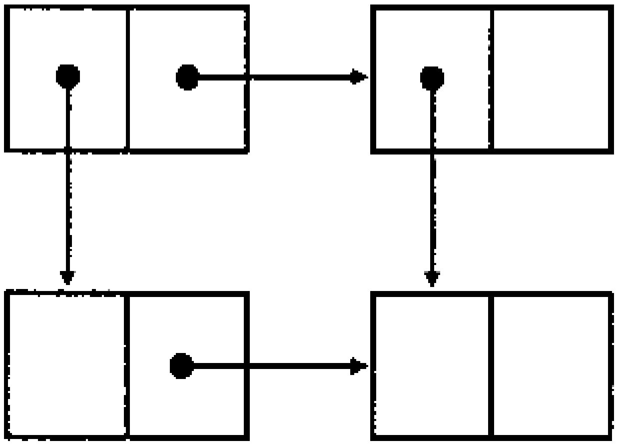
- Game theory is a mathematical tool for analyzing the interactions among a group of rational participants who behave strategically.
- The essential elements of game theoretic models are:
  - Players
  - Actions
  - Payoffs
  - Information

Classification of Games

- There are several types of games:
  - symmetric and asymmetric
  - zero sum and non-zero sum
  - simultaneous and sequential
  - perfect information and imperfect information
  - single-shot and iterative
  - cooperative and non-cooperative

SW Outsourcing as a Prisoner's Dilemma Game

- At T=0:
  - Client decides whether:
    - to pay the developer (cooperate),
    - or not (compete)
  - Developer chooses whether:
    - to implement the functionality (cooperate),
    - or not (compete)



Pay-off matrix		Developer	
		cooperates	competes
Client	cooperates	2 2	0 3
	competes	3 0	1 1

Having no information about the other party's behaviour leads to the **worst** outcome for both players!

N. Oza, **Game theory perspectives on client: vendor relationships in offshore software outsourcing**, International workshop on Economics driven software engineering research, 2006

Evaluating SW Correctness as a Prisoner's Dilemma Game

- Developer chooses effort level (high, low) and management pays only for bug-free SW components
- Tester chooses effort level (high, low) and management pays only for SW components found to be faulty

Average Pay-off matrix		Tester	
		Effort=L	Effort=H
Developer	Effort=L	2 2	0 3
	Effort=H	3 0	1 1

This particular incentive scheme enables competition (which is in favour of management, but not ideal for the players)!

L. Feijs. **Prisoner dilemma in software testing**, Computer Science Reports, 1:65-80, 2001

WP2 Remarks

- Game Theory is a powerful tool for predicting rational behaviour and evaluating outcomes.
  - But very sensitive to the assumptions.
- Which case studies should be modeled? These should not be restricted to SW engineering.
  - How can we induce SW users to participate in the reputation scheme and provide high effort?
  - How can we induce the White hat community to assist in dealing with cyber attacks?

WP3/Task 3.3

Task 3.3: How to assess the trustworthiness of the target **collaborative** system?

System made out of n Components

- Define the indicator variable x_i, to indicate whether the *i*-th component is functioning or has failed.

$$x_i = \begin{cases} 1, & \text{if the } i\text{th component is functioning} \\ 0, & \text{if the } i\text{th component has failed} \end{cases}$$

- The state vector: $\mathbf{x} = (x_1, \ldots, x_n)$

- The functionality of a system is determined by a function with regard to the state vector:

$$\phi(\mathbf{x}) = \begin{cases} 1, & \text{if the system is functioning when the state vector is } \mathbf{x} \\ 0, & \text{if the system has failed when the state vector is } \mathbf{x} \end{cases}$$

The series system

- It is functioning if and only if all of its components are functioning.

- In terms of a diagram:

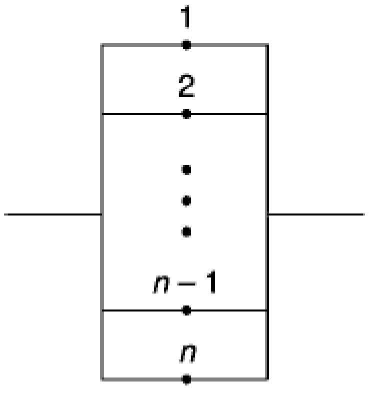
- The structure function:

$$\phi(\mathbf{x}) = \min(x_1, \ldots, x_n) = \prod_{i=1}^{n} x_i$$

The parallel system

- It is functioning if and only if at least one of its components is functioning.

- In terms of a diagram:



- The structure function:

$$\phi(\mathbf{x}) = \max(x_1, \ldots, x_n)$$

Reliability

The state of each component X_i is a random variable, such that.

$$P\{X_i = 1\} = p_i = 1 - P\{X_i = 0\}$$

- The value of p_i, equals the probability that the *i*-th component is functioning and is called the reliability of the *i*-th component.

- We define *r* as the reliability of the system:

$$r = P\{\phi(\mathbf{X}) = 1\}, \qquad \text{where } \mathbf{X} = (X_1, \ldots, X_n)$$

- We may express *r* as a (reliability) function of components reliability:

$$r = r(\mathbf{p}), \qquad \text{where } \mathbf{p} = (p_1, \ldots, p_n)$$

Systems' Reliability

- For the series system:

$$r(\mathbf{p}) = P\{\phi(\mathbf{X}) = 1\}$$
$$= P\{X_i = 1 \text{ for all } i = 1, \ldots, n\}$$
$$= \prod_{i=1}^{n} p_i \quad \blacksquare$$

- For the parallel system:

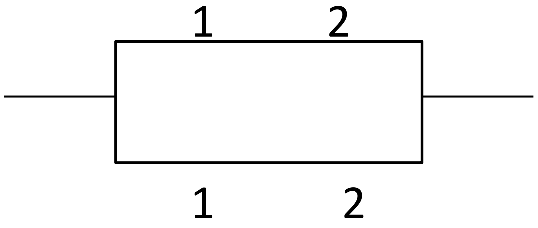
$$r(\mathbf{p}) = P\{\phi(\mathbf{X}) = 1\}$$
$$= P\{X_i = 1 \text{ for some } i = 1, \ldots, n\}$$
$$= 1 - P\{X_i = 0 \text{ for all } i = 1, \ldots, n\}$$
$$= 1 - \prod_{i=1}^{n} (1 - p_i) \quad \blacksquare$$

Example (WP5, WP6)

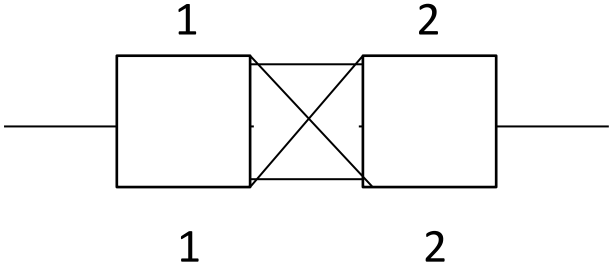
We want to build a **series** system of two different types of components and we have available two of each kind. Each of them is functioning with probability 0.5.

Should we build two separate systems, or replicate the components?

1. Reliability: $1-\left(\frac{3}{4}\right)^2=\frac{7}{16}$



2. Reliability: $\left(\frac{3}{4}\right)^2=\frac{9}{16}$



Examples of AUEB research work

1. Which is the minimum cost to make a system function with certain probability.

2. Each component functions properly for a random length of time.

Which is the **expected/upper bound** of a system **lifetime**. (WP6)

3. Once a component fails it need a random time period to be repaired.

How to count/increase the **availability** of a system at time t? (WP6)

Thank you!

Any Questions?