



Software Engineering

Lets not forget the quality

A brief re-look into the interaction of the software testing discipline and the Software engineering academic world

1

Dani Almog – Ben Gurion
University of the Negev Israel

Agenda

2

- ❑ What is happening out there (results from surveys)
- ❑ Industry needs for software quality engineers
- ❑ Current programs
- ❑ Comments about Software Quality & Testing Engineer
- ❑ Where are we heading – a vision of quality driven software development.
- ❑ Highlights on new research thread (topics)
- ❑ What next

The following information was collected from

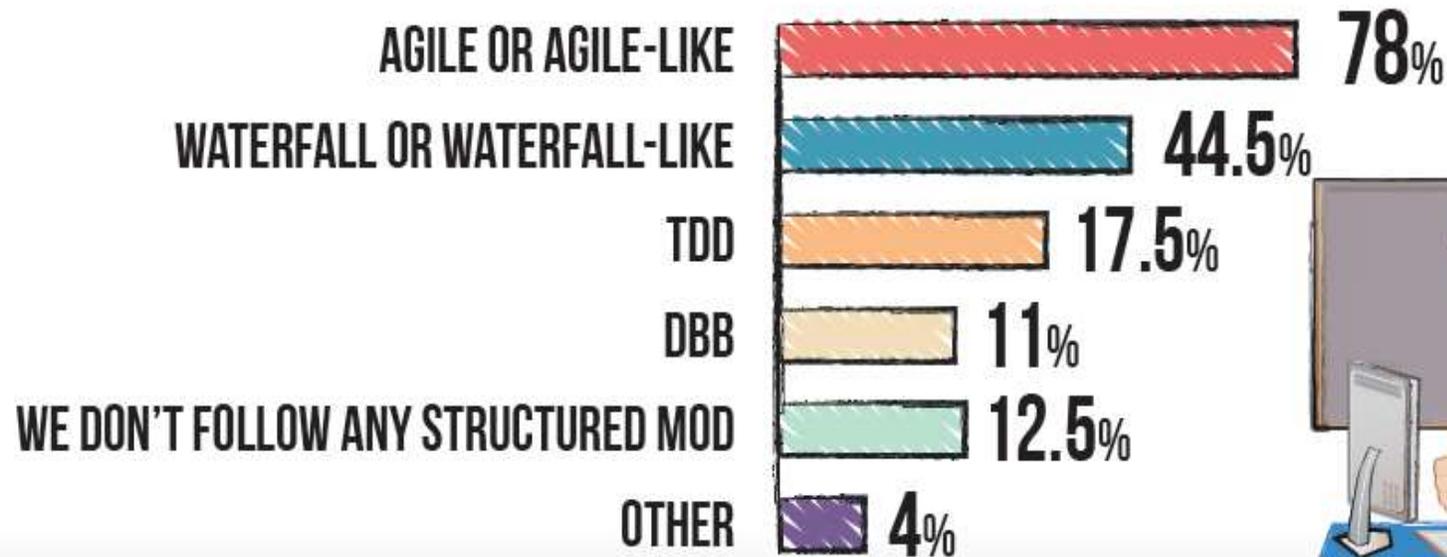
- ❖ **Self-Perceptions about Software Engineering:** A Survey of Scientists and Engineers survey 2012 The University of Alabama
- ❖ **The state of mobile Apps development and testing:** - test complete (by smart bear 2012)
- ❖ **State of testing survey 2013:** - Practitest & Tea Time with testers 2014
- ❖ **The ACM Guide to Computing Literature** (Bibliographic citations from major publishers in computing) 2013

Agile is here to stay

4

STRONG AGILE ADOPTION OF 78%

*(more than one answer per participant ->
total is more than 100%)*



The State of Mobile App Development & Testing

2014



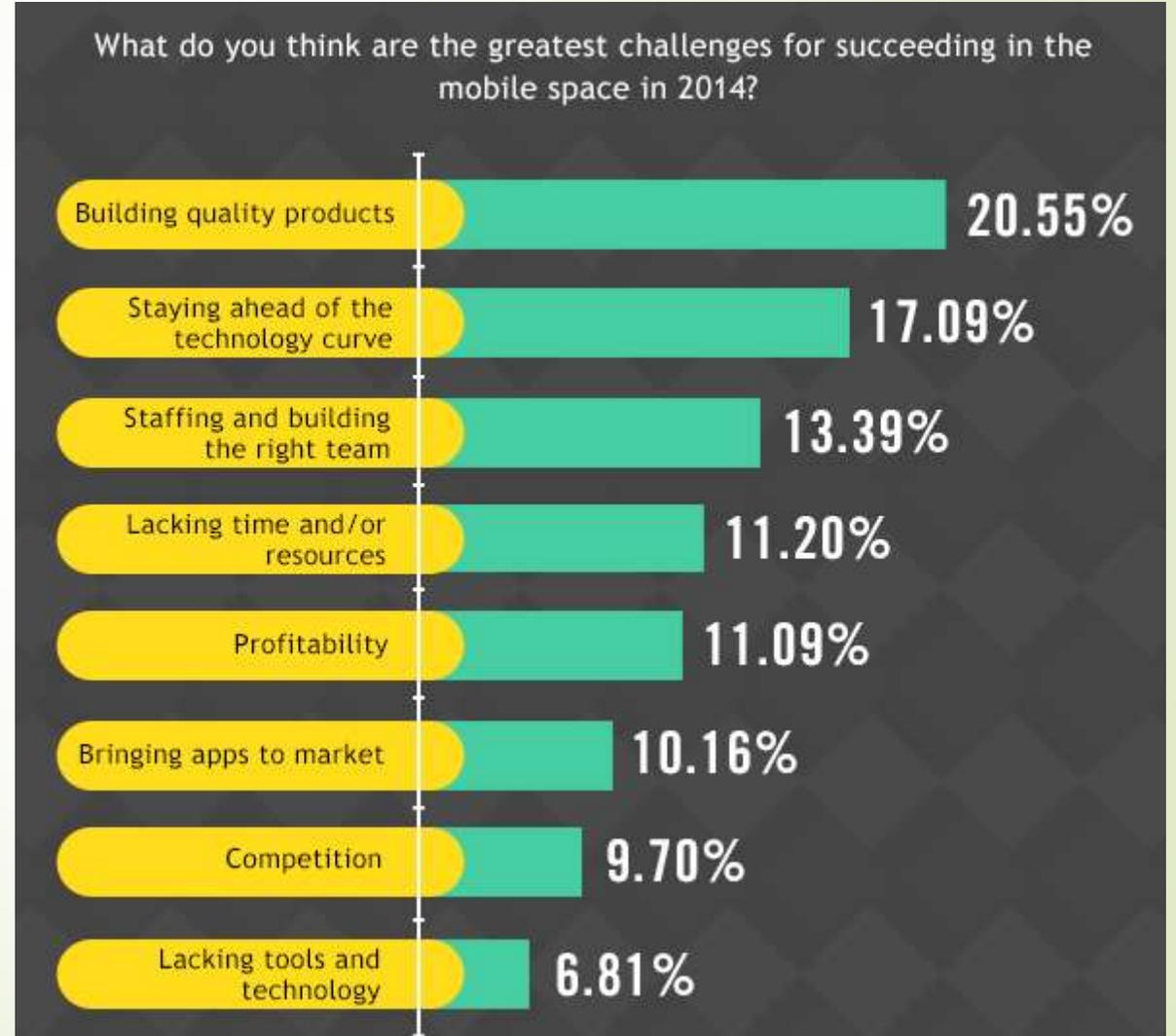
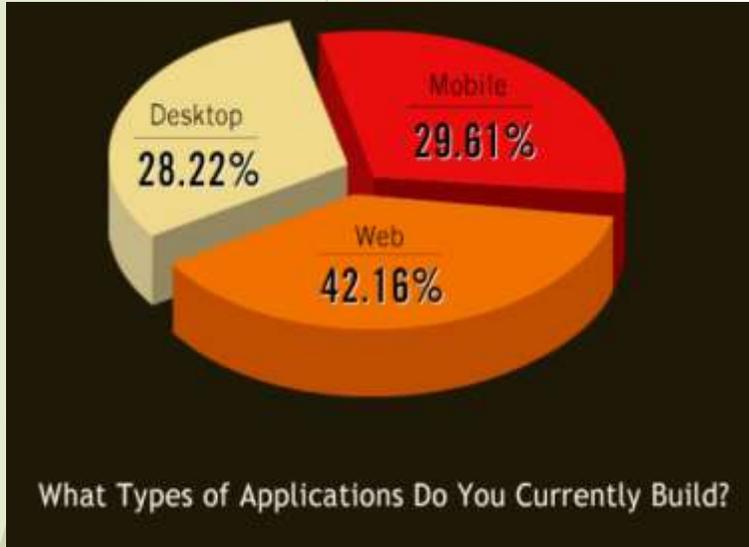
• ARE YOU READY? •

84% OF THOSE WHO ARE NOT CURRENTLY BUILDING MOBILE APPS **PLAN TO ENTER THE SPACE IN THE NEAR FUTURE.**



Do you plan to enter the mobile space in the near future?

Some results



Self-Perceptions about Software Engineering: A Survey of Scientists and Engineers

Department of Computer Science, The University of Alabama 2011

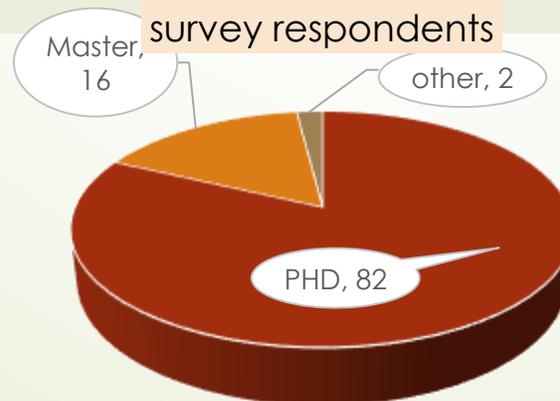
Jeffrey C. Carver, and Roscoe Bartlett, Dustin Heaton, Lorin Hochstein

The goal of the survey was to gather the respondents' perception of software engineering knowledge and use at three different levels. First, we wanted to understand how the survey respondents assessed themselves in terms of their software engineering skills. Second, we were curious about how the survey respondents assessed their teammates' level of software engineering knowledge and skill. Finally, we were interested in the survey respondents' opinions

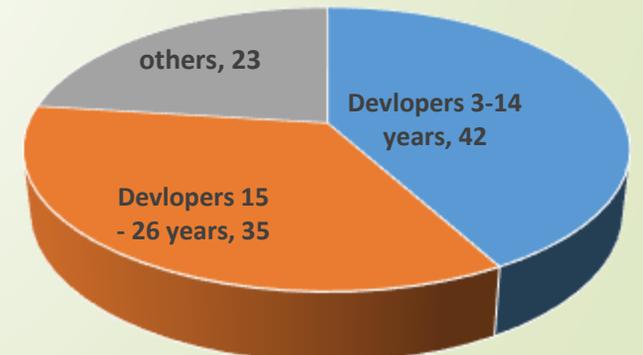
of the level of software engineering knowledge and skill for the CSE community as a whole.

7

- Relevance to my work
- personal level of use
 - personal familiarity
 - team level of use
 - team's familiarity



Respondent SE experience



Software Engineering & practices

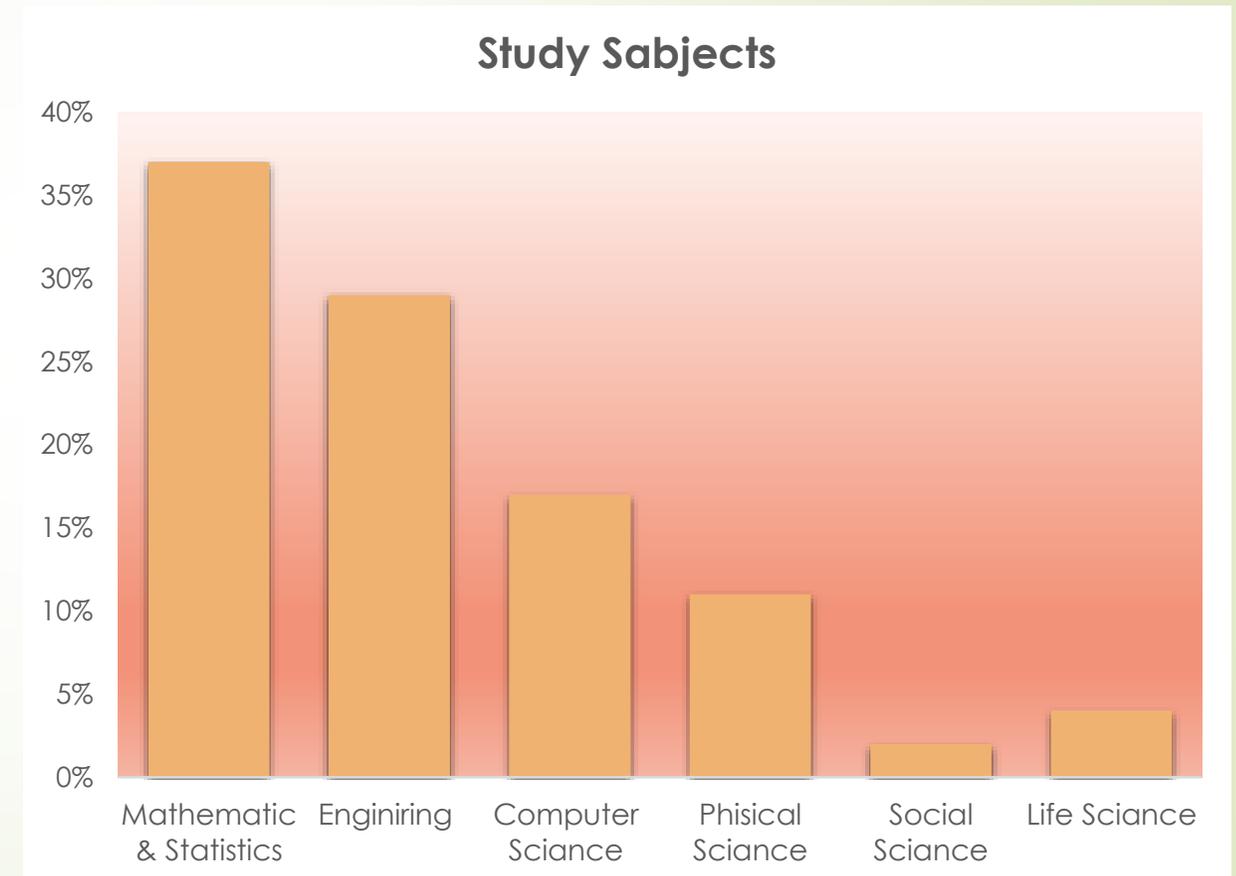
8

commonly used in the commercial software development community

What do we do in SE

1. Software Lifecycles
2. Documentation
3. Requirements
4. Basic Design
5. Intermediate Design
6. V&V (Verification & Validation)
7. Unit Testing
8. Integration Testing
9. Acceptance Testing
10. Regression Testing
11. Version Control/Change Management
12. Issue/Bug Tracking
13. Test-Driven Development
14. Structured Refactoring
15. Code Reviews
16. Agile Methods

What do we teach in SE



Selective conclusions (Out of many)

- CSE practitioners have very little formal software engineering training and tend to be mostly self-taught
- Most of practitioners believed that their software engineering knowledge and skills were at least “mostly sufficient” to achieve the goals of their CSE projects
- Majority of the individuals in the CSE community “don’t know what they don’t know.”
- More than a third of the respondents thought that overall the CSE community’s skills were not adequate to
- Collaborative development practices which are necessary for achieving sufficient quality in many situations

My personal intake from this survey is:

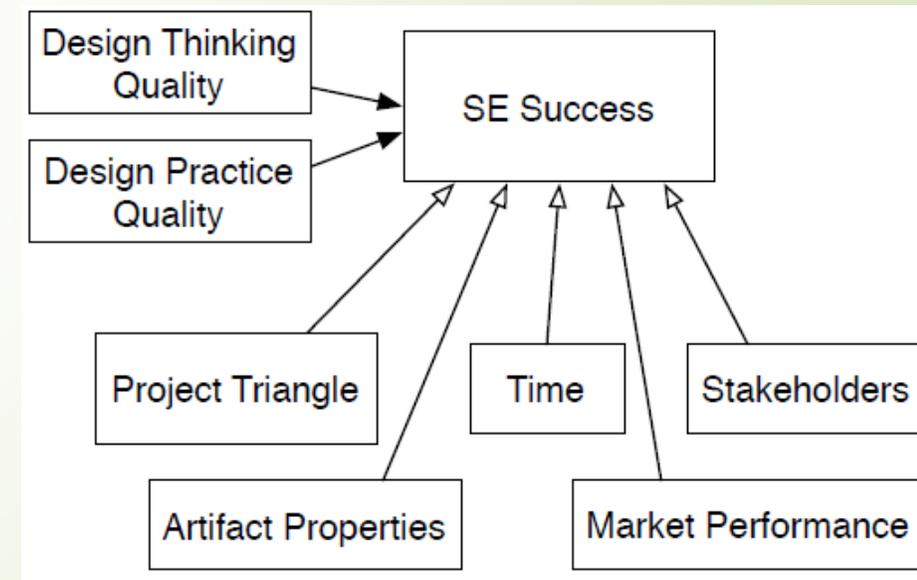
- Big Gap between what we teach and the practice
- We tend to neglect all Quality related aspects
- This survey must be read and evaluated by all SE policy leaders

Software Engineering challenges

10

The modern SE requires fast development and adaptation skills following rapid changes on market trends

- New software Languages
- New development platforms
- Rapidly changed target platforms – today mobile – tomorrow?
- New development techniques & tools
- Development life cycle agile dictate openness & cooperation



Academia = Education or research?

11

- What is really motivating academia (Publish or Parish)
- In between CS and Software Engineering
- Difference between 'undergrad' and 'graduate'
- Industry & Academy - this is not only a terminology gap

"Universities are producing IT graduate students, but whether the degrees fit with business needs is another question. We need a wider conversation between business and universities," - *Gareth Preece, skills specialist at UK Trade & Investment*

"What we need is a qualification that is really fit for purpose, universities, business and the government need to work together to create a worthwhile qualification".

Revision of the SE 2004 Curriculum Model

12

[Software Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering](#) (SE 2004) is one volume in a set of computing curricula adopted and supported by the ACM and the IEEE Computer Society

2011 Survey conducted by:

Gregory W. Hislop	Drexel University
Mark J. Sebern	Milwaukee School of Engineering
Mark Ardis	Stevens Institute of Technology
Jeff Offut	George Mason University
David Budgen	Durham University
Willem Visser	University of Stellenbosch

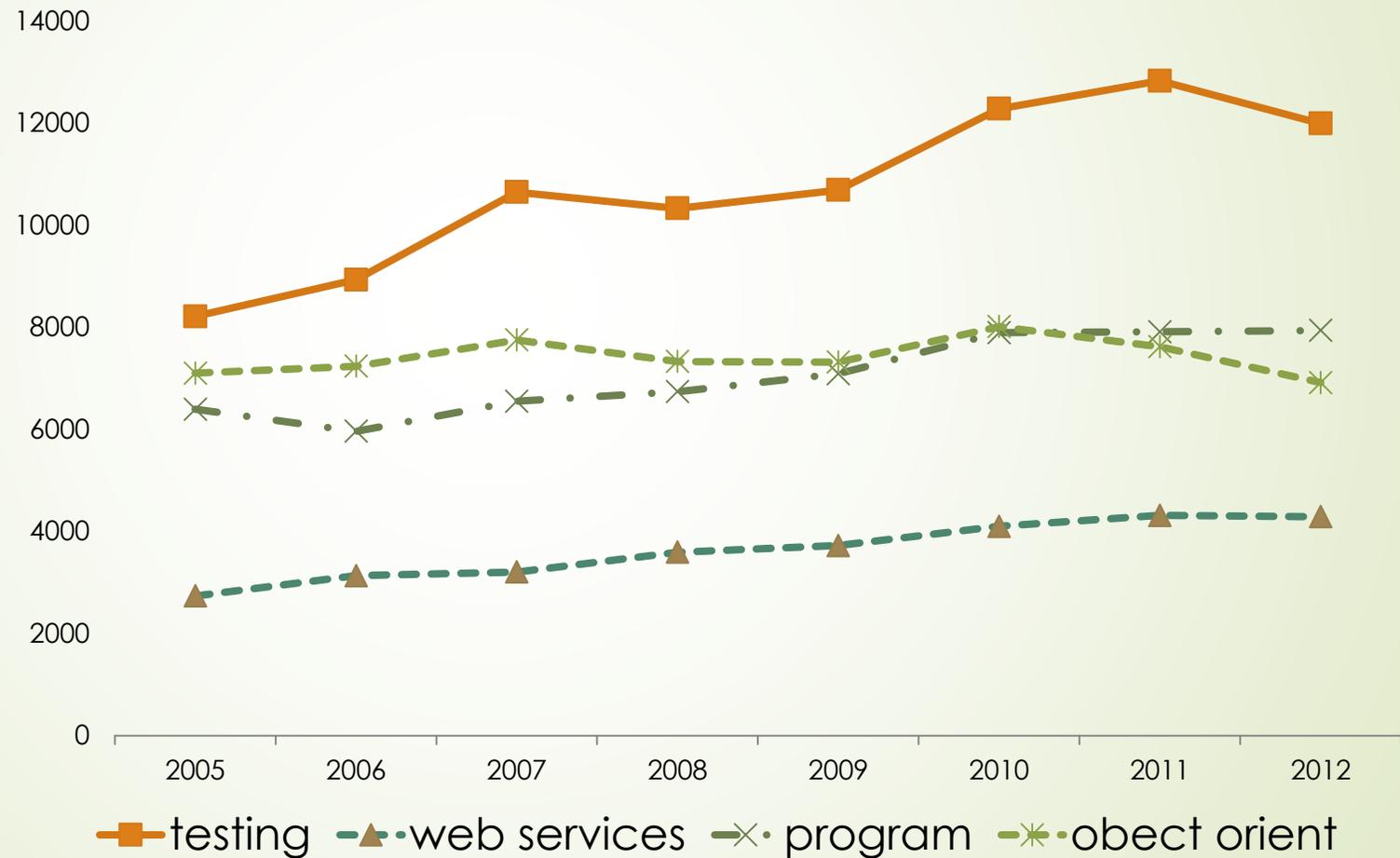
we found that the overall structure of SE 2004 is sound and does not need to be changed. The SEEK was found to need only minor revisions. The areas of revision identified include:

- ▶ agile methods: these have become more popular and successful
- ▶ security: increasingly important as more services are exposed to attack
- ▶ service-oriented computing: these have become more popular and important

Testing and Quality Assurance is a mandatory part of the Core Software Engineering Sequences

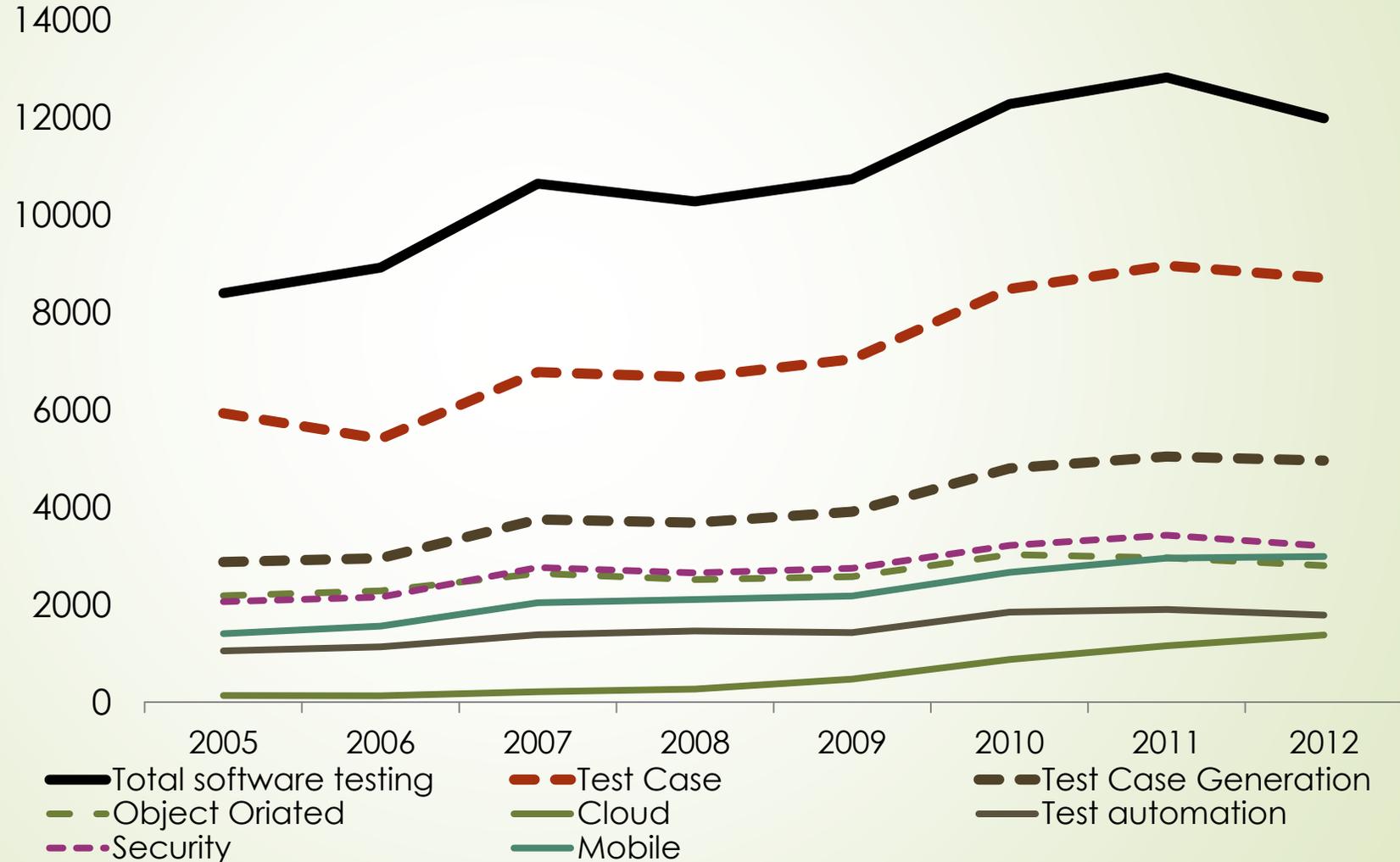
Academic Publication Trend

Software Subject Publication Trend



Academic Publication Trend

Software Testing related publications



Software Quality & Testing engineer

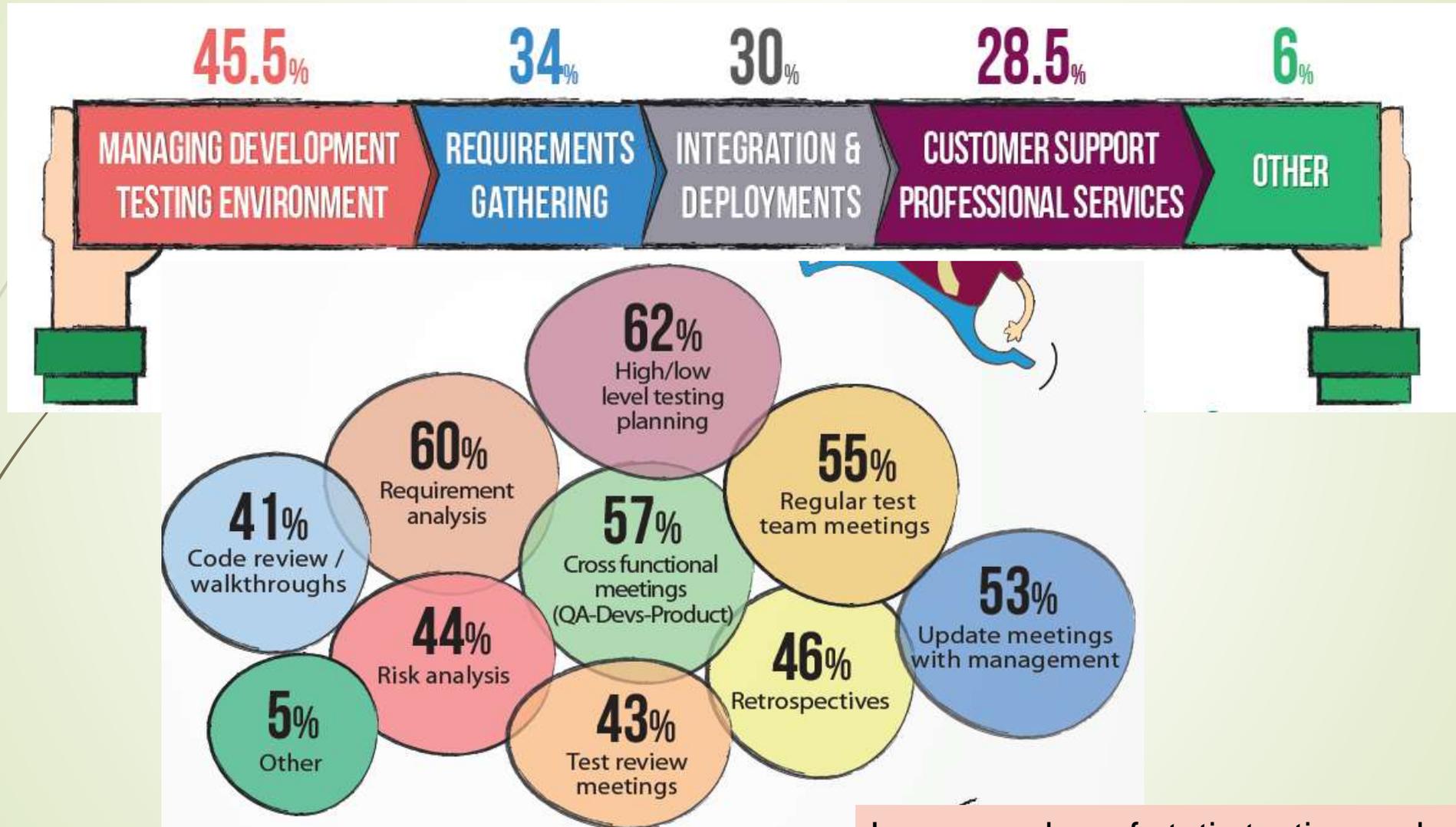
- No more traditional positions – the stage is constantly changing
- Integrate with development (agile)
- Very fast change rate – self learner – if you do not progress you'll disappear
- No more just manual (simple) work – you either Test Expert or Automation Expert
- Very challenging work environments
- Be organizational glue and a leader



Roles of tester

(Other than testing! Of course)

16

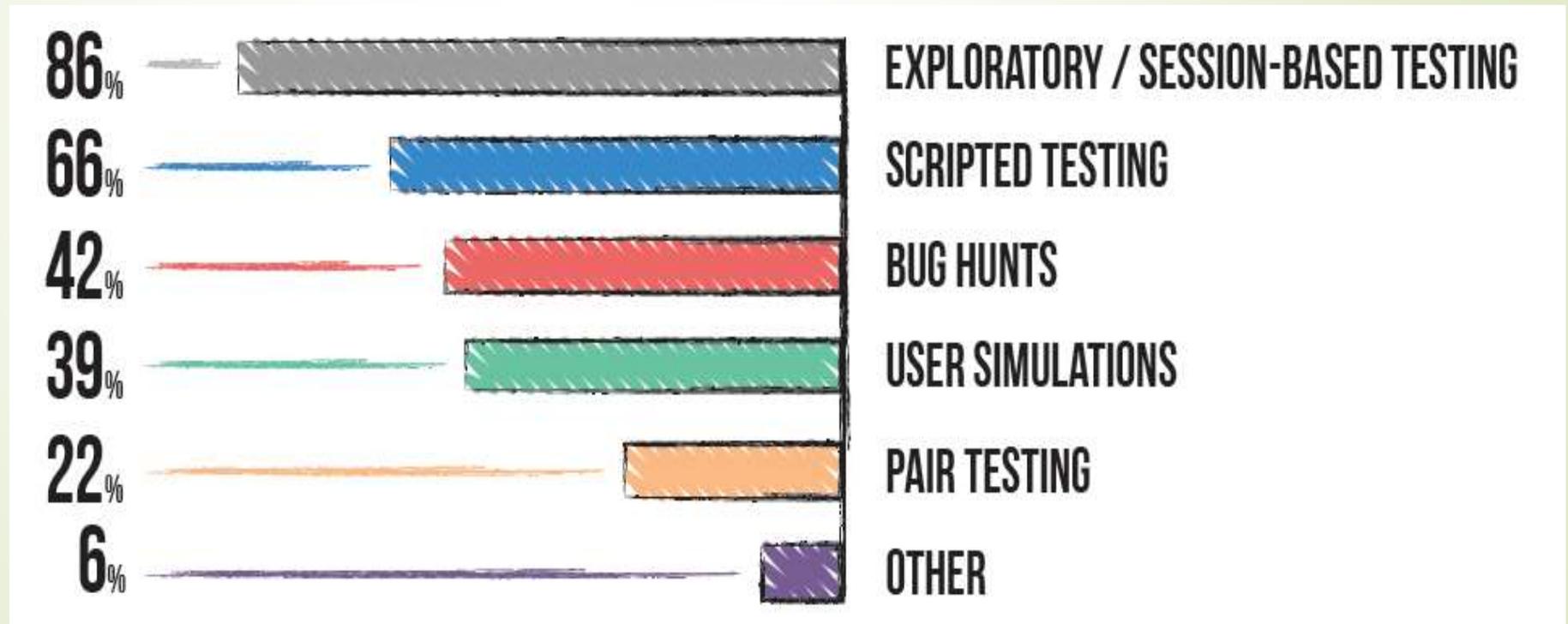


Large number of static testing and review activities

Testing & Development Practices

17

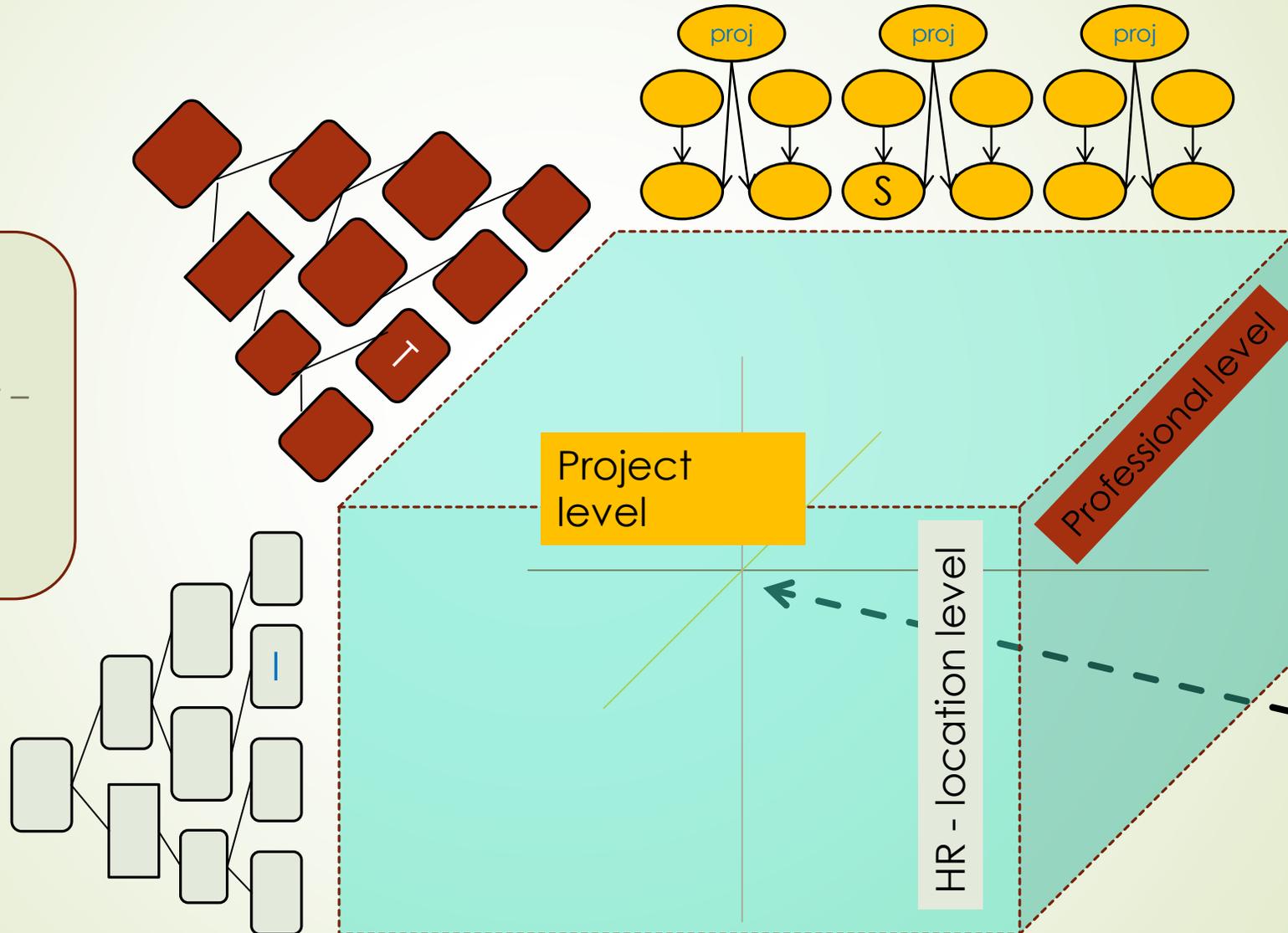
**86% USE EXPLORATORY OR
SESSION-BASED TESTING**



Who is my manager

18

Who tell me **what** to do –
Project Manager
Who tells me **how** to do it –
Professional Manager
Who pays my **salary** –
Location Manager



Research trajectories (threads)

- Search into the gap between development and testing (new software development/ testing paradigms)
- Security & Cyber Testing
- Test case generation
- SOA testing and testing as a service
- Testing models for the cloud and big data
- Testing of mobile technology related
- Lean & Exploratory Testing

Conclusions

20

- SW quality & Testing was almost ignored by the academic world until 1999
- SW Quality and Testing is currently getting more emphasis in SW engineering
- A gap between industry needs and academic education and research
- Academic Research is basically detached from industry achievements

What's next?

- Software Quality & Testing should be a **mandatory course** to every SE schools
- Get the professionals to contribute to universities/collages
- Open communication channels
- Expose students to industry real life cases and reality
- Generic research programs (jointly initiated by industry and academia)



Questions ??

Thank you for the opportunity
to share my vision

22

Dani Almog

Almog.dani@gmail.com