



How much is a great developer worth?

NDC, 2013

Stein Grimstad
Magne Jørgensen

scienta

Salary of football players

- Daniel Berg Hestad (Molde):
1.6 mill NOK per year
- Lionel Messi (Barcelona):
\$40.3 mill per year
- Salary ratio of about 1:150



scienta

Software developers

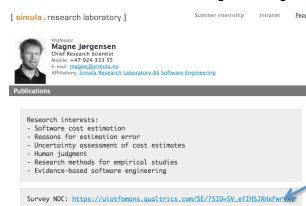
- Average (Norway): 600.000 NOK?
- Best paid (US): \$1.2 mill?
- Ratio of about 1:15
 - ✓ Norway-salary ratio: 1:2?
 - ✓ Norway-per hour ratio: 1:1.5?
- Is this reflecting the difference in value between a great and an average developer?



scientia

Before we move on ...

- If you are online (laptop or smartphone), please go to: **simula.no/people/magnej** and click this link



- You will then be asked 2-4 multiple choice questions about competence evaluation (takes 2-3 minutes).
- The result of the survey will be summarized as part of this presentation.

scientia

Research on productivity differences ...

- First study in 1966, with 12 experienced programmers (Sackman, Erickson & Grant):
 - ✓ Effort difference 1:16 and 1:25
 - ✓ Size difference 1:6 and 1:5
- Summary of individual programming productivity from 61 experiments (5-36 persons) (Prechelt, 1999)
 - ✓ Typical difference between best and worst about 1:15
 - ✓ Typical difference between one in "slower quarter" and one in "faster quarter" about 1:5
- Four companies developing the same system (Anda, Sjøberg et al., 2009)
 - ✓ Effort difference of about 1:3 (including client effort)
 - ✓ Size difference of about 1:2

scientia

Own research: The 6 best companies out of 16 companies bidding for our project

	Comp. A	Comp. B	Comp. C	Comp. D	Comp. E	Comp. F
Price	Very low	Low (2x)	Medium (3x)	High (5x)	Very high (12x)	Very high (14x)
Est. effort	Very low	Low (1.5x)	Medium (3x)	High (8x)	Medium (4x)	Very high (8x)
CV	OK	OK	Good	Good	Good	OK
Refs.	Very good	Very good	Very good	Very good	Very good	Very good
Proposal	OK	OK	Good	OK	OK	OK
Country	Finland	Malaysia	India	India	Canada	US

Which company would you select?

scientia

Before I give you the results ... It is not easy to be a client.

- As a client you have to decide whether a very low price or effort estimate (such as the one by Company A) indicates:
 - ✓ High productivity and skill (great developer)
 - ✓ High degree of over-optimism, leading to unrealistic plans
 - ✓ Low skill (the Dunning-Kruger effect, where those unskilled are less aware of their lack of skill)
 - ✓ Lower expected quality of the product
 - ✓ More problematic process with the provider (typical when fixed price projects and a bidder with low price is selected)
- In short, should we take the risk of selecting Company A with its low price and low effort estimates?

scientia

Our study of more than 800.000 projects at freelancer.com shows that ...

- Clients tend to avoid companies/developers with unusually low price, **even when the companies document the the same level of competence as the one selected!**
 - ✓ Experience from Norwegian software industry indicates that this does not necessarily hold for large scale projects costing millions, where they are more likely to select low price bidders ...
- A fear of low price is, to some extent rational. Our data shows that:
 - ✓ Low price makes, on average, good companies perform worse (due to overoptimistic estimates)
 - ✓ Low price correlates with higher risk of project failure

scientia

The study also showed that:

- The best predictor of non-failing projects, was "previous successful collaboration with the client"
 - ✓ Can be seen as a very realistic test of the provider
- Client skill was almost as important as the skill of the provider to predict project failure
- Systematic and large differences between project failure rates in different outsourcing countries.
- Among the larger outsourcing countries:
 - ✓ Lowest failure rates: Argentine, Eastern European countries
 - ✓ Highest failure rates: South Asia (India, Pakistan, Bangladesh)

scientia

So, the clients may be rational, BUT ...

Clients avoiding companies with low price or low effort estimates may also avoid the companies with low price due to great developers!

Let's go back to our 6-company study ...

scientia

Repetition: The six good looking companies

	Comp. A	Comp. B	Comp. C	Comp. D	Comp. E	Comp. F
Price	Very low	Low (2x)	Medium (3x)	High (5x)	Very high (12x)	Very high (14x)
Est. effort	Very low	Low (1.5x)	Medium (3x)	High (8x)	Medium (4x)	Very high (8x)
CV	OK	OK	Good	Good	Good	OK
Refs.	Very good	Very good	Very good	Very good	Very good	Very good
Proposal	OK	OK	Good	OK	OK	OK
Country	Finland	Malaysia	India	India	Canada	US

scientia

We selected all six ...
Here is how they performed

	Comp. A	Comp. B	Comp. C	Comp. D	Comp. E	Comp. F
Actual effort	Very low	Low (3x)	High (6x)	High (8x)	Very high (18x)	Very high (16x)
Error fixing effort	Very low	High (4x)	Medium (2.5x)	High (4x)	Very high (8x)	Extr. high (20x)
Maintenance effort	Very low	High (6x)	Very high (11 x)	High (8x)	Extr. high (26x)	Extr. high (20x)
Lines of code	Very low	Low (2x)	Low (1.5x)	Medium (3x)	High (4x)	Low (1.5x)

Company A had a great developer, but we would probably not have chosen that company in the normal case when selecting only one developer. Simply too risky without knowing more about the competence. Middle is more safe ...

scientia

What can we learn?

- Huge differences in software development productivity, quality and maintenance cost for even simple systems
- Not easy to identify great developers from CVs, satisfaction of previous clients and quality of proposals
- The real differences will typically remain unknown to the clients, the managers of the developers and probably to the developer themselves, as clients select only one provider

scientia

Consequences

- The salaries of developers and payment by clients are not even close to reflecting the real differences in performance
- We need better ways to assess the competence of developers and companies

- Stein?



scientia

Hiring : State of practice

- Many employers are currently using suboptimal selection methods for hiring of software developers
 - ✓ Both for consultants and permanent employment
- Studies shows that when recruitment personnel are updated on relevant academic research, their companies perform better economically

scientia

Interviews (unstructured)

- Often used for hiring developers
 - ✓ Cheap and straight-forward method
 - ✓ Interviewer are often over-confident in their interviewing skills
- Research has repeatedly documented that this is a poor selection mechanism
 - ✓ Over-emphasize irrelevant information and contextual knowledge
 - ✓ Difficult to compare candidates
 - ✓ Probably even worse for selection in offshoring contexts



scientia

“Clouds Make Nerds Look Better”



- We know that interviews are influenced by the candidate's weight, attractiveness, speed of speech, etc
- Study of university applicants:
 - ✓ 12% higher chance when sunshine compared to worst cloud cover
 - ✓ Sunshine means more focus on social skills, cloudy means more focus on academic skills
 - Nerd-factor measured as academic rating divided by social rating (e.g., leadership).

scientia

Structured interviews

- Good selection method for hiring top performers
 - ✓ Unlike unstructured interviews
- How does structured interviews differ?
 - ✓ Questions determined by a careful analysis of the job in question
 - ✓ Usually the same questions to all candidates
 - ✓ Predefined scoring of responses and rules for candidate evaluation
- In practice, structured interviews are very similar to testing of candidates

scientia

Biographical information (CV)

- Useful for initial screening
- Research suggest that you should emphasize:
 - ✓ University grades
 - ✓ Past job performance (preferable from similar jobs)
 - ✓ Relevance of experience / education
- ...and don't emphasize:
 - ✓ Years of experience
 - ✓ Knowledge of specific technologies & frameworks
 - ✓ "Buzzword compliance"
 - ✓ Activities unrelated to work
- Research shows that strong candidates benefit from excluding less relevant information in their CVs

scientia

References

- Checking references and networks may be a valuable if you can trust them to give you honest and complete information
 - ✓ NB! Most people find it difficult to reveal negative information
 - ✓ NB! Sometimes job performance is strongly dependent on job environment
 - ✓ NB! Expertise can be surprisingly narrow

scientia

Test of programming skills

- Useful for filtering out candidates that lack programming skills
 - ✓ Lots of tests are available
 - ✓ Typically "programming puzzles"
- ... but remember that many other factors also impact software development performance
 - ✓ Ability to share/reuse code
 - ✓ Team work / communication
 - ✓ Requirement engineering skills
 - ✓ Etc

```

1 for i in range(1,101):print 'FizzBuzz' if i%3==0 else i

```

Work sample tests

- Highly recommended method
 - ✓ Better than general programming tests
 - ✓ Typically small, but complex tasks
 - ✓ The more representative tasks, the better results in context-specific problems
 - ✓ Examples: Fix a bug in the system, design a new feature
- Take measures to avoid cheating
 - ✓ Change tasks frequently
 - ✓ Use pair-programming/blackboards

```

* @param ctx CosContext
* @param agreement The agreement for which we want to l
* @param searchCustUnitNumber customer unit number.
* @return Structure containing the hierarchy
* @throws COSFunctionalException any functional error.
* @throws COSTechnicalException any technical error.
* @methodAuthor Stein Grimstad
* @since COS 4.2
* /

```

General Mental Ability (GMA)

- Intelligence at work is not wholly different from intelligence at school
 - ✓ Intelligent people acquire job knowledge faster and acquire more of it
 - ✓ Inexpensive tests are available, e.g. Wonderlic tests
- Research shows that GMA nicely complements structured interviews and tests
 - ✓ ...but GMA is rarely used for hiring of software developers
 - ✓ Prejudice against high IQ (bad at communication, etc)
 - ✓ We may, wrongly, assume small differences within the same profession (e.g. software developers)

scientia

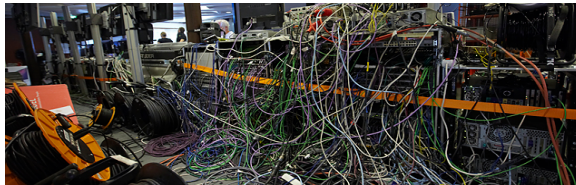
NB! Selection of top performers is not the only way to increase productivity

- We can also increase productivity by, e.g.
 - ✓ Reducing system complexity
 - ✓ Improving software development tools and methods
 - ✓ Improving the work environment
 - ✓ Improving processes analysis and specification work
- Hiring top performers is not even always wanted
 - ✓ Sometimes bad for team dynamics
 - ✓ Issues with cost and competition
 - ✓ Lack of challenges / more easily bored

scientia

Reduce system complexity

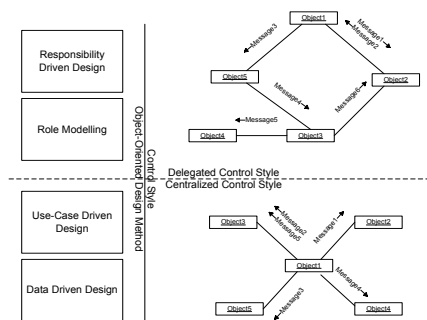
- Productivity differences correlates with job complexity
 - ✓ Reduce system complexity -> reduce differences
- Many well-known approaches to reduce complexity
 - ✓ Modularization
 - ✓ Consistency
 - ✓ Conventions and Documentation
 - ✓ Simple, easy to understand, design patterns



scientia

Centralized vs. Delegated design

- In a study by Erik Arisholm, 500+ performed maintenance tasks on two alternative designs of the same system
- Purpose: Study the effect of centralized vs. delegated design (the latter often considered better)



scientia

Results

- In the delegated design, the maintenance tasks took more time and had more errors
- Only the most experienced developers seemed to have the necessary skills to utilize the more elegant delegated design

scientia

Summary

- Productivity differences are huge among software developers
 - ✓ Even for developers with similar CV, experience, education, etc
- It is hard to select the top performers
- Recommended: GMA test in combination with either structured interviews and work-samples
 - ✓ The huge economic benefits of selecting top performers makes up for the additional costs of these selection methods

scientia

Questions and Comments?

Contact information:
magnej@simula.no
stein@scienta.no

scienta