Object-Oriented Software Re-engineering

Universität Bern, IAM, SCG

Object-Oriented Metrics in Industry

Dr. Simon Moser

Version: November 1998

Release note

This 2 hour lecture emerged from previously developed tutorials held over the years 1994-1998.

It is a development of

Dr. Simon Moser (moser@acm.org).

Outline

Measurement-Based Estimation in Industry

The Effects of Reuse

Industry Benchmarking

Object-Oriented Metrics

Sources

[1] Moser S., Measurement and Estimation of Software and Software Processes, Ph.D. thesis (http://www.iam.unibe.ch/~scg/Archive/PhD/moser-phd.pdf)

[2] Moser S., Nierstrasz O., The Effect of Object-Oriented Frameworks on Developer Productivity, IEEE Software, September 1996, pp. 45-51

[3] IFPUG, International Function Point User Group (http://www.ifpug.org)

[4] Jones C., Programming languages Table (http://www.spr.com/library/0langtbl.htm)

[5] Moser S., Misic V., Class Coupling and Cohesion: A Formal Metamodel Approach, Proc. Of APSEC'97, Hong Kong, Dec. 1997, pp. 31-39

Measurement-Based Estimation in Industry (1/10)

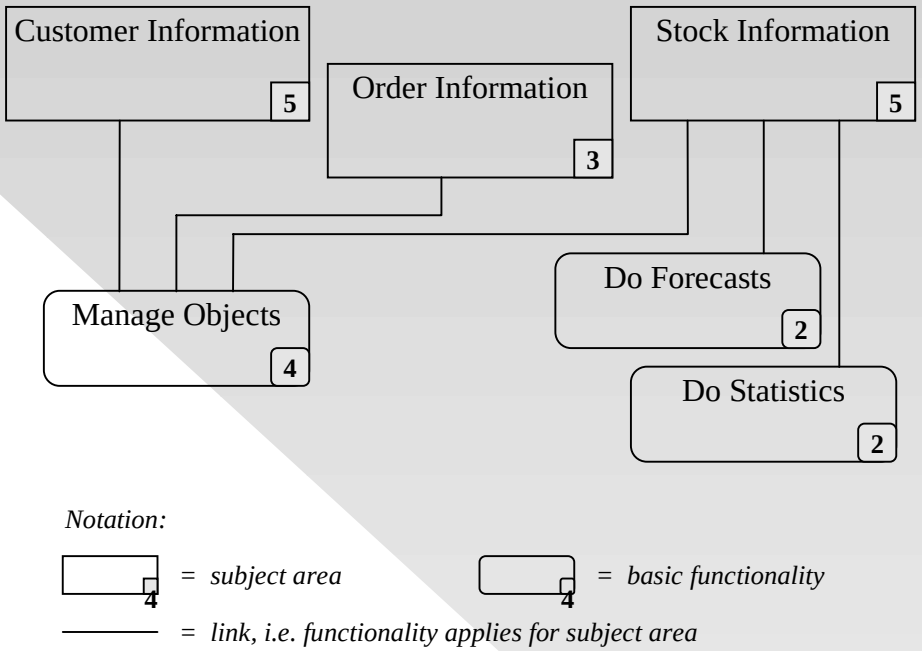
- Example:

  Estimate the development effort for a **„Tiny Order Information System“** (TOIS).

- The steps (cf. foil 28):

  (A) Measure requirements model
  (B) Derive standard development effort
  (C) Adapt with respect to tailored development plan

Measurement-Based Estimation in Industry (2/10)



Graphical notation of the TOIS preliminary requirements model

Measurement-Based Estimation in Industry (3/10)

```
; PRE description of TOIS, the tiny order information system
Subject Area ”Customer Information” 5 .
Subject Area ”Order Information” 3 .
Subject Area ”Stock Information” 5 .
Functionality ”Manage Objects” 4 .
Functionality ”Do Statistics” 2 .
Functionality ”Do Forecasts” 2 .
Subject Area ’Customer Information’ contains ’Manage Objects’ .
Subject Area ’Order Information’ contains ’Manage Objects’ .
Subject Area ’Stock Information’ contains ’Manage Objects’,
             ’Do Statistics’, ’Do Forecasts’ .
```

Textual notation of the TOIS preliminary requirements model (language PRE, cf. [1])

Measurement-Based Estimation in Industry (4/10)

issue

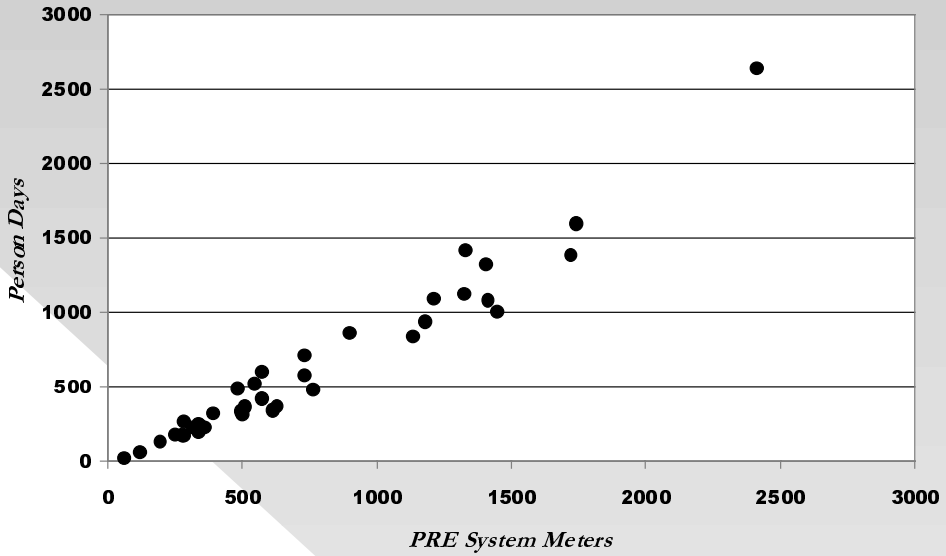
```
ma -v -f tois.sdf
```

in a DOS command shell from the **SEBT**-directory. After admittedly a bit lengthy phases of tokenising, scanning and finally counting, the tool will eventually finish its work by printing out a measured number:

```
System Meters = 563
```

Step (A): measurement of the TOIS preliminary requirements model using the utility „ma“ resulting in System Meters at PRE level (PRE SM)

Measurement-Based Estimation in Industry (5/10)

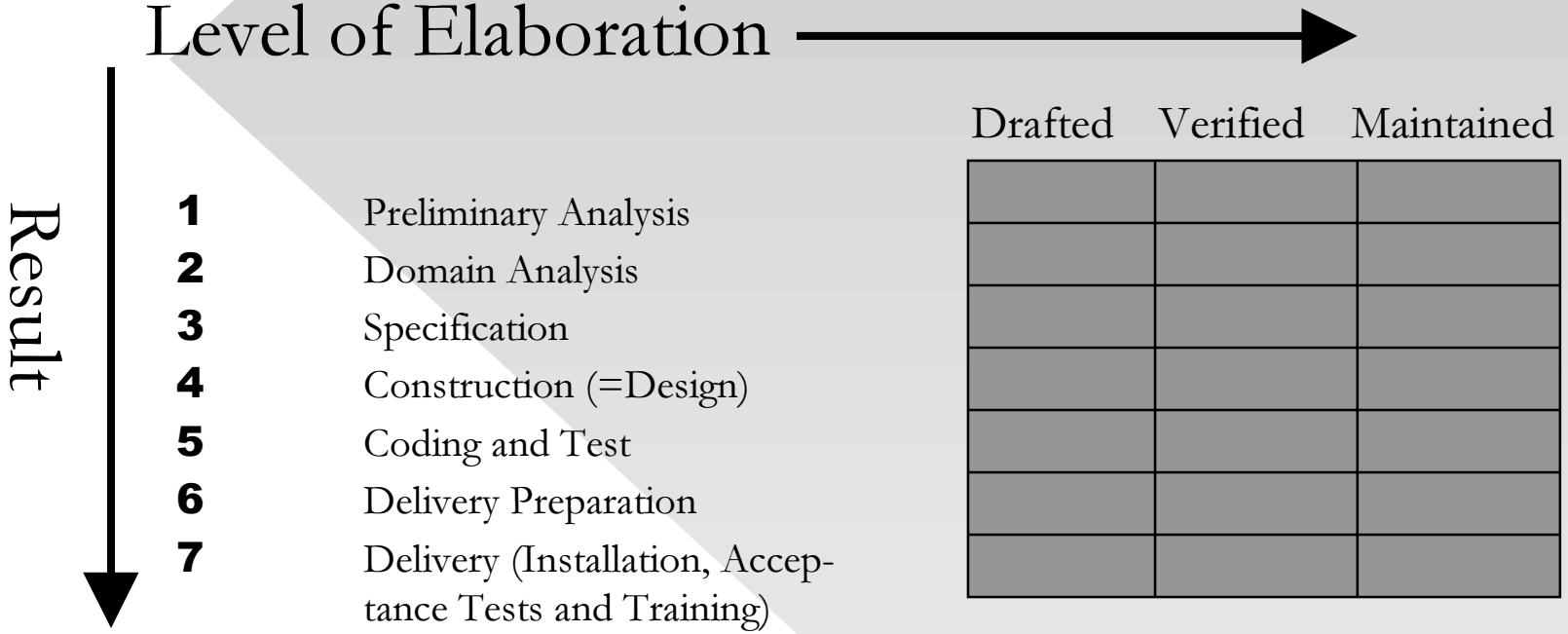


$$\mathbf{Effort_{Person\ Days} = Complexity_{PRE\ SM} \times 0.605 + Complexity_{PRE\ SM}^2 \times 0.00}$$

$563 \times 0.605 + 563^2 \times 0.0001779 = 340.6 + 56.4 = 397$ PD
(95%-error range: +/-33%)

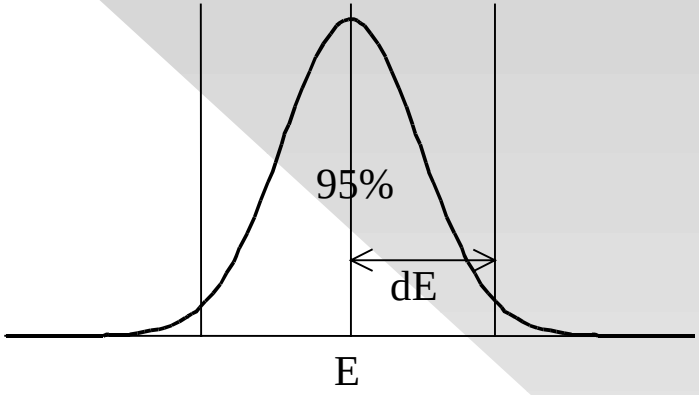
Step (B): Entering the empirical database to obtain an estimate for a standard full development process

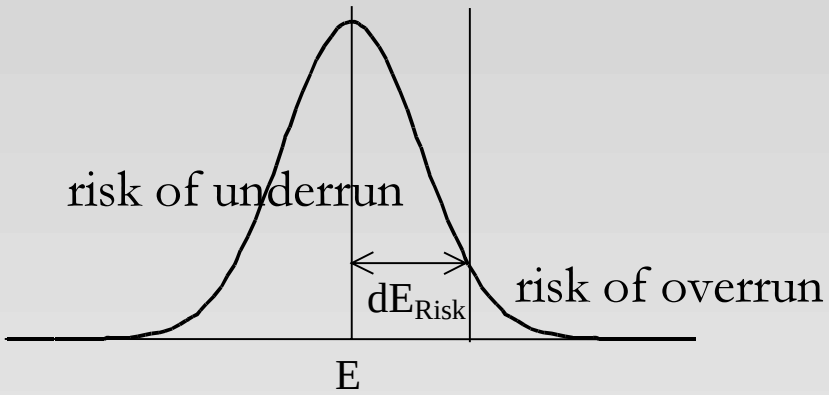
Measurement-Based Estimation in Industry (6/10)



Step (C): Adapting the „normal effort“ to the tailored plan:
C.1 Process Completeness Assessment (PCA) ⇨ %C

Measurement-Based Estimation in Industry (7/10)





E = estimate, dE = error range of estimate

Step (C): Adapting the „normal effort“ to the tailored plan:

C.2 Risk Adaptation ⇨ $\%dE_{Risk}$, $E_{C.2} = E \times (1 + \%dE_{Risk})$

Measurement-Based Estimation in Industry (8/10)

Step (C):

C.3: Project Cost Calculation (PCC):

Input:
- risk adapted „normal effort“
- PCA per project phase (optional: per contractor)
- degree of completeness per subsystem per phase
- span activities (project mgmt, quality management)
- tariffs per activity

Output:
- efforts per subsystem and phase
- costs per subsystem and phase

Measurement-Based Estimation in Industry (9/10)

How to use measurement-based estimation in Re-Engineering:

(A) Getting the System Size backwards (Backfire-method):
1) measure LOC (or KLOC) per language
2) derive functional size by using empirical DBs per language (cf. [4])

(B) Deriving „normal effort“

(C) Doing PCA for the Re-Engineering process:
- What do we have in maintained form? Code? Design? Specs? Domain models?
- What do we want? In what form?
- What can we obtain in an automated way? (assess as draft)

Measurement-Based Estimation in Industry (10/10)

More topics...

- Using more detailed models (e.g. business models using UML, cf. [1])

- Deriving ideal team size and elapsed time from the adapted effort (cf. [1])

- Adaptations for very tight schedules (cf. [1])

- Using other functional measures (e.g. Function Points, cf. [3])

The Effects of Reuse (1/4)

- Example:

  For some parts of the TOIS we have reusable parts at hand...

- possible strategies:

  1 Heuristically assess completeness percentages (step C.3 PCC)

  2 Rely on special empirical database (step B)

  3 Modify the requirements model (chosen approach...)

The Effects of Reuse (2/4)

```
; PRE description of TOIS, the tiny order information system, with reuse modelling
;ma-entry: category library
Subject Area    "Customer Information"  5 .
;ma-entry: category project
Subject Area    "Order Information"  3 .
Subject Area    "Stock Information"  5 .
;ma-entry: category library
Functionality   "Manage Objects"  4 .
;ma-entry: category project
Functionality   "Do Statistics"  2 .
Functionality   "Do Forecasts"  2 .
;ma-entry: category library
Subject Area    'Customer Information' contains 'Manage Objects'.
;ma-entry: category project
Subject Area    'Order Information' contains 'Manage Objects'.
Subject Area    'Stock Information' contains 'Manage Objects',
                'Do Statistics', 'Do Forecasts'.
```

Reuse-modified TOIS preliminary requirements model
(language PRE, cf. [1])

The Effects of Reuse (3/4)

```
invoking ma...

System Meters = 415

415 × 0.605 + 415² × 0.0001779 = 251 + 30.6 = 281.6 PD
```

Steps A and B for the reuse-adapted TOIS

The Effects of Reuse (4/4)

Project	Function Point Productivity	Productivity Increase (%)	PRE SM Productivity	Productivity Increase (%)
C1	1.1	N/A	2.47	N/A
C2	1.6	45	2.30	-7
C3	1.6	45	2.12	-14
C4	1.5	36	2.64	7
C5	1.7	55	2.30	-7
C6	1.5	36	2.13	-14
C7	1.6	45	2.42	-2
C8	1.6	45	2.69	9

C1 = Framework project, C2-8 = Reuse projects

Productivity rates of reuse-adapted (PRE SM) and non-adapted (Function Point) model sizes (cf. [2])

Non-adapted sizes show „magical“ increase...

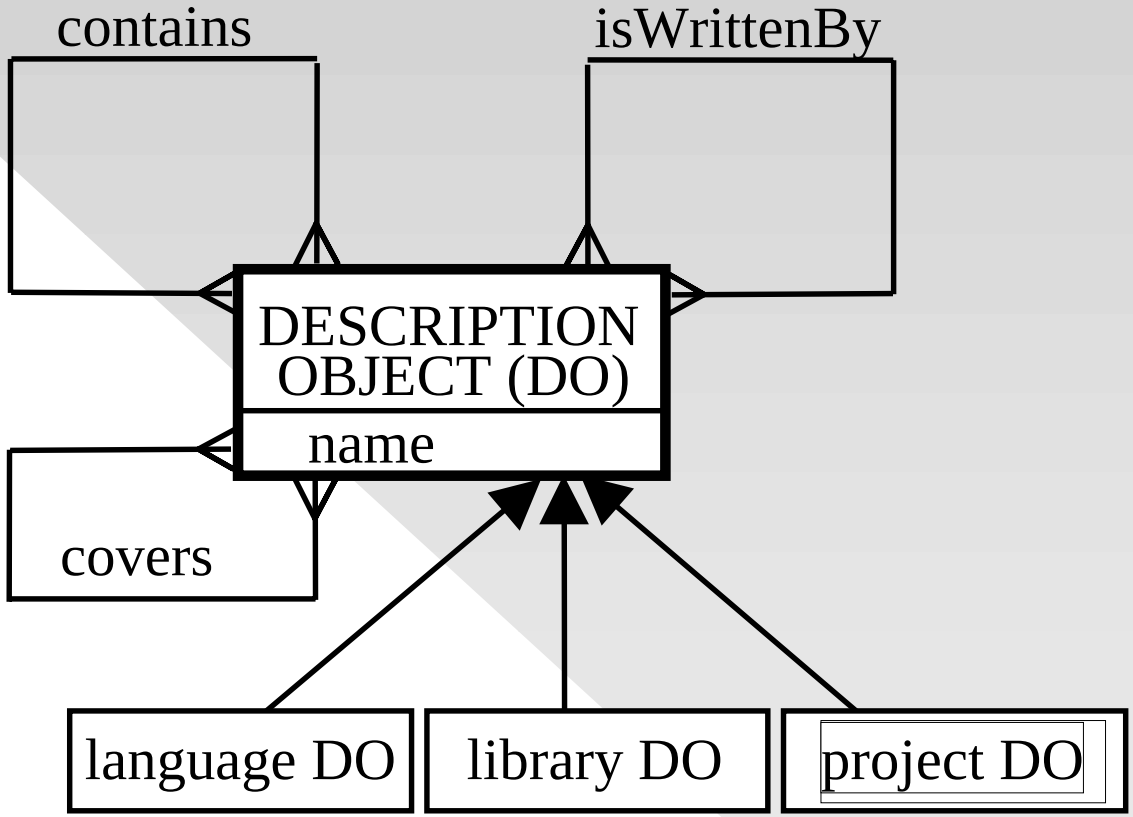
Industry Benchmarking (1/1)

- Compare productivity rates (or other target key figures) among companies...

- Analyse company/project characteristics ⇨ possible reasons
- Define corrective actions... and iterate

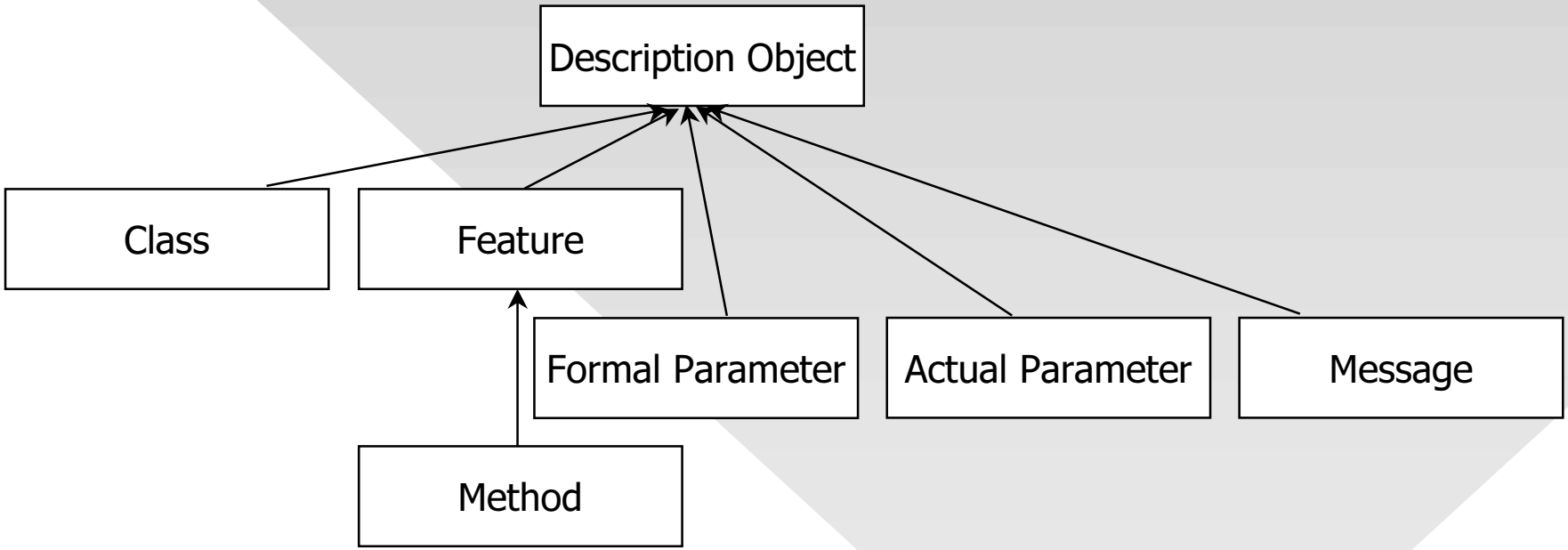
Object-Oriented Metrics (1/4)

The **generic metamodel** of system descriptions...



Object-Oriented Metrics (3/4)

The **metamodel hierarchy** for object-oriented code...



Object-Oriented Metrics (3/4)

Metrics of size, e.g. System Meter

- Definition (for single description object):
  Size ≡ externalSize + internalSize
  externalSize ≡ numberOfTokens (name)
  internalSize ≡ externalSize (isWrittenBy)
- Definition (for sets of description objects):
  Size ≡ externalSize (library) + Size (project)

(NB: The above notation is somewhat casual)

e.g. Number of Classes

(easy but correlates with +/-40% to effort only)

Object-Oriented Metrics (4/4)

Quality metrics, e.g. coupling (cf. Mišić, Moser, 1997)

– Definition on generic metamodel:

$$\textbf{coupling}\ (c1, c2) = 1 - \frac{\sum_{o \in c1.isDefinedWith \cap (\{c2\} \cup c2.contains^{+})} externalSize(o)}{\sum_{p \in c1.isDefinedWith} externalSize(p)}$$

– Informal definition: Coupling is the ratio of dependencies outside some composite description object

Object-Oriented Metrics in Industry - Summary

- **Measurement-based estimation**: transparent, improvable and adaptable (e.g. to re-engineering projects)

- **Reuse modelling**: enables extension of measurement-based techniques to framework and reuse projects

- **Benchmarking**: comparing measured performance, improve towards industry state-of-the-art

- **Object-oriented metrics**: KISS ... and wait for more research results!