Tids- och kostnads-uppskattning av RUP Projekt

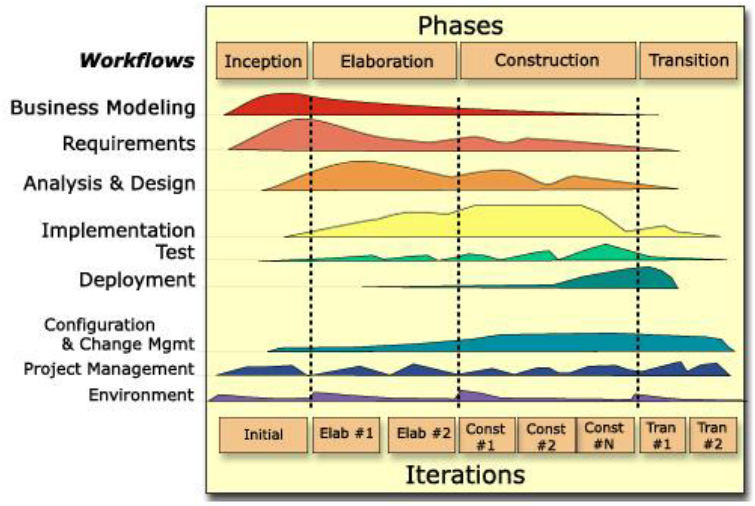
Micael Erneborg
micael.erneborg@kvadrat.se

Patrik Jonsson
patrik.jonsson@it-arkitekterna.se

Content

- **Context: RUP & Project Management Models**
- **Project Phases & Iteration**
- **Definitions**
- **What to Estimate**
- **Estimation Methods overview**
- **Practical examples**
- **Next step**

Rational Unified Process (RUP)



Source: RUP2000, Rational Software

KVADRAT

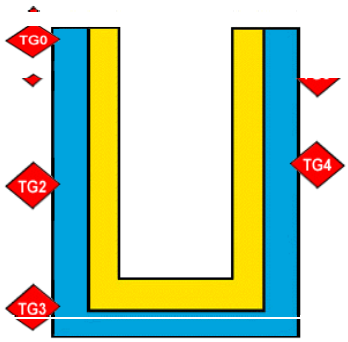
IT ARKITEKTERNA

PROPS

- **PROPS is a generic project model, developed by Ericsson Project Management Institute**

PROPS consists of three parts:

- ***Blue U*, project management**
- ***Yellow U*, development process**
- ***Red tollgates*, commercial decision points**



KVADRAT

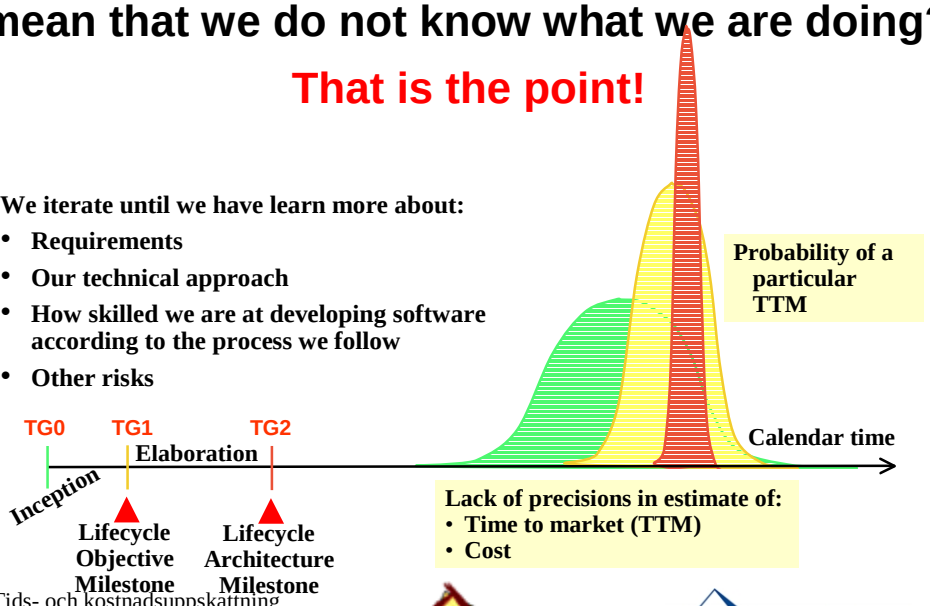
IT ARKITEKTERNA

Does not the Iterative Approach simply mean that we do not know what we are doing?

That is the point!

We iterate until we have learn more about:

- **Requirements**
- **Our technical approach**
- **How skilled we are at developing software according to the process we follow**
- **Other risks**



Lack of precisions in estimate of:
- **Time to market (TTM)**
- **Cost**

KVADRAT

IT ARKITEKTERNA

Estimation

Estimate, Guess, Guestimate, …

People are
Optimistic, Scared, Want to look good,
Forget the history, Have selective memories, and
Think in single point numbers

We do better if we
Get a *clear picture*
Use *Team* and several experts,
Use *range estimates and statistics*,
Use a formal *estimation method*, and
Insist on *realistic estimates*

We assume you're
in control of your *requirements*, and
have a sound *Project Management process*

Unit of estimation:
- **Sweden: man-hours**
- **US: size**

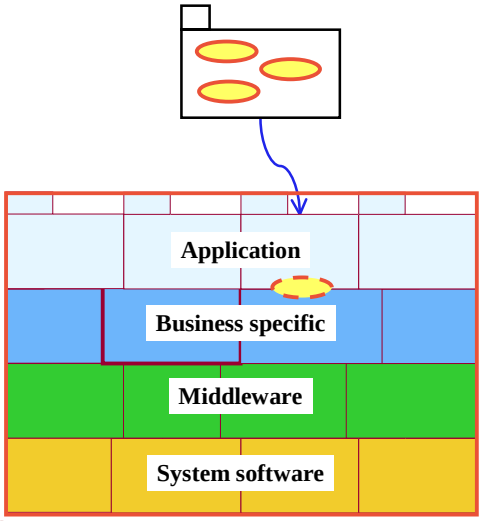
KVADRAT

IT ARKITEKTERNA

What to estimate for?

Everything that moves ☺

- **The complete system**
  - **Rough estimates**
- **Use Cases**
  - **A good foundation for estimation. Important to include challenges with the technology**
- **Use Case Realizations**
  - **We know more than the requirements**
- **Subsystems**
  - **The chosen technology is integrated in the sub-systems.**



KVADRAT

IT ARKITEKTERNA

Real world case 1

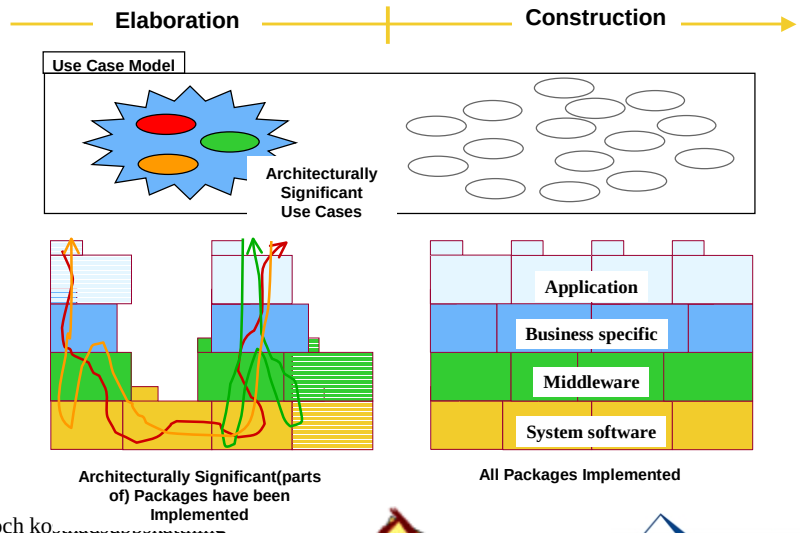
- **Estimates done with a matrix containing modules (part of the architecture) and use cases**
- **Estimates in execution**
  - **Chief architect (1)**
  - **Module responsibles (2)**
  - **Estimates 100%-400% larger**
- **Positive**
  - **Project Managers can easily plan which use cases to implement**
- **Negative**
  - **New method -> hard to determine if the estimates are resonable**
- **No outcome yet**

Module/ Req	A	B	C	D	E	F	Σ
UC1	32				8		40
UC2			45				45
UC3			23		10		33
UC4		37					37
R5			23			28	51
R6					34		34
R7					92		92
Σ	32	37	91		144	28	332

KVADRAT

IT ARKITEKTERNA

Use Cases Realizations connect Subsystems

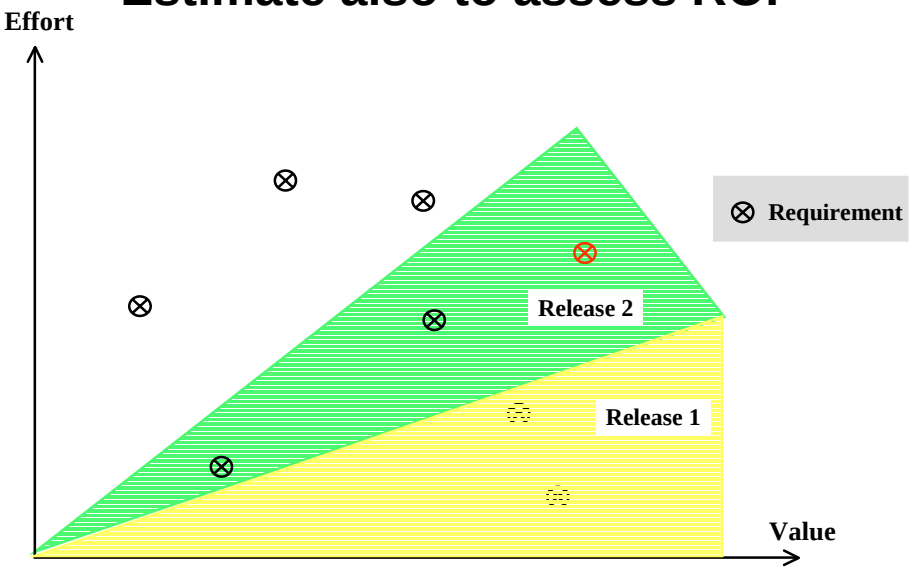


Tids- och kostnadsuppskattning
av RUP Projekt: # **9**
RAIN 2002

KVADRAT

IT ARKITEKTERNA

Estimate also to assess ROI



Tids- och kostnadsuppskattning
av RUP Projekt: # **10**
RAIN 2002

KVADRAT

IT ARKITEKTERNA

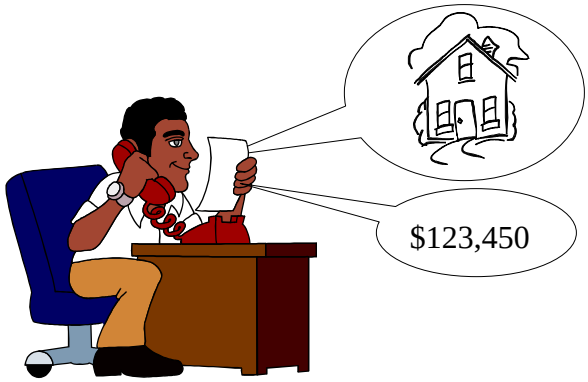
Methods overview

- **Expert Judgements**
- **Analogy**
- **Delphi**
- **Algorithmic Models**
  CoCoMo, Function Points, Object Points, UC Points
- **Lichtenberg**
- **Paired Comparison**
- **Critical Chain**

KVADRAT

IT ARKITEKTERNA

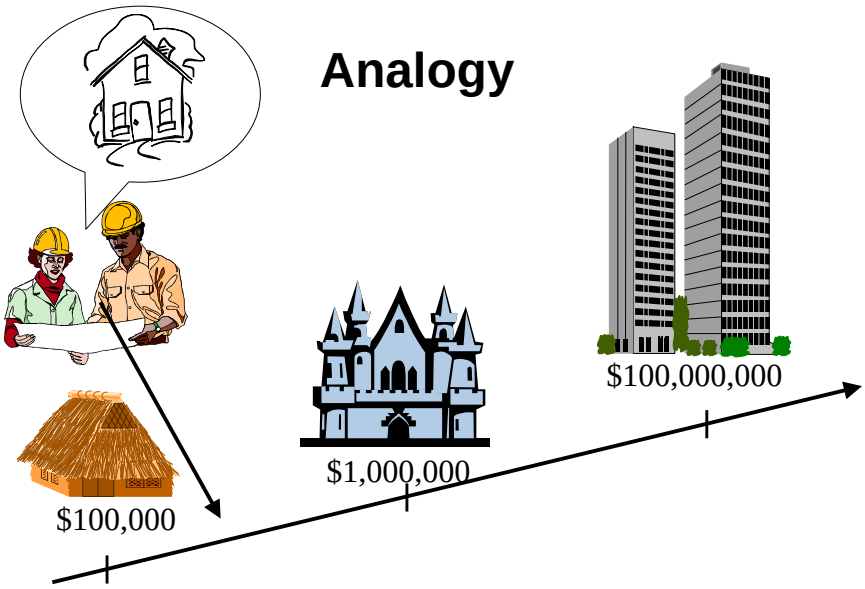
Expert judgements



KVADRAT

IT ARKITEKTERNA

Analogy

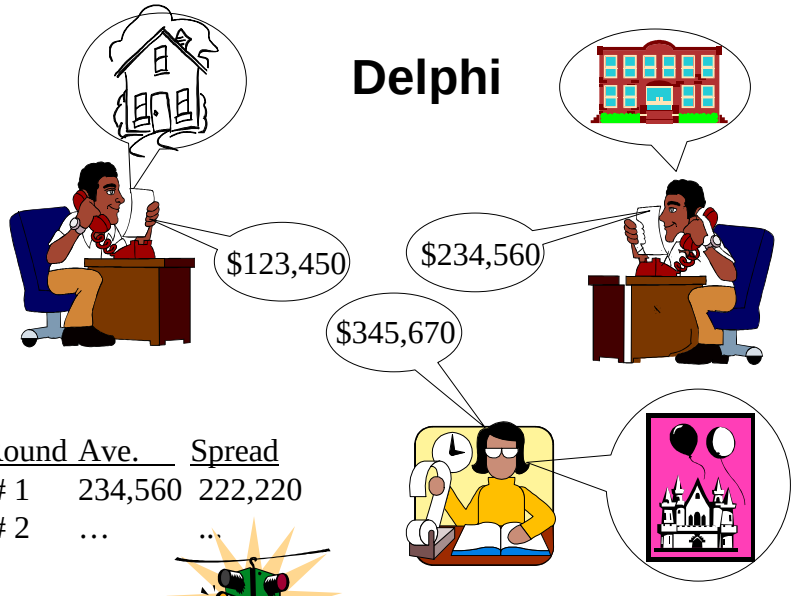


Tids- och kostnadsuppskattning
av RUP Projekt: # **13**
RAIN 2002

KVADRAT

IT ARKITEKTERNA

Delphi



Round	Ave.	Spread
# 1	234,560	222,220
# 2	…	…

Tids- och kostnadsuppskattning
av RUP Projekt: # **14**
RAIN 2002

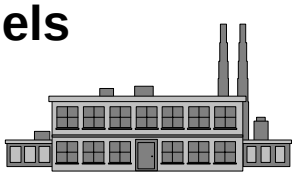
KVADRAT

IT ARKITEKTERNA

Algorithmic Models

Scaling Factor

Effort Multiplier



$$PM = A \times Size^{E} \times \prod_{i=1}^{n} EM_i + PM_{Auto}$$

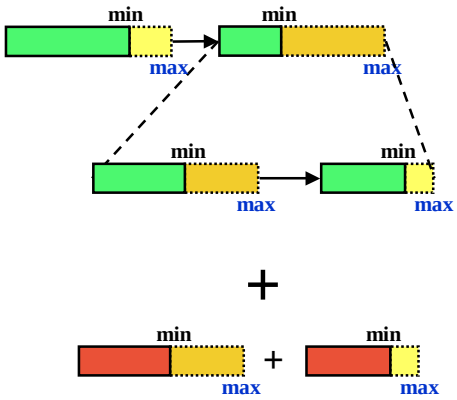
- **Cost in Effort (Person Months), Calendar Time, or Size (CoCoMo II)**
- **Function Points, Object Points, Use Case Points, Boehm, …**
- **Up to within 20% in 70% of the cases**

KVADRAT

IT ARKITEKTERNA

”Lichtenberg-method”

- **Estimate with uncertainty :**
  - **Min, max, most likely**
  - **Average = (min+max+3*likely)/5**
  - **Standard deviation = (max-min)/5**
- **Successive break-down**
  - **Break-down the activities with the largest uncertainty**
  - **Estimate the parts and add up:**
    - » ***Schedules***
    - » ***Costs***
    - » ***Standard deviations***
- **The Red/Green world**
  - **Ignore risks and crucial uncertainties first, then**
  - **Add the corresponding activities**



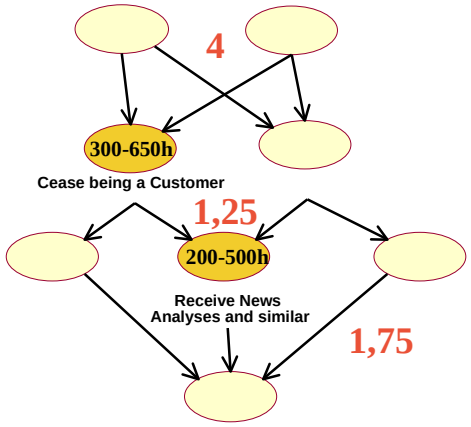
KVADRAT

IT ARKITEKTERNA

Paired Comparison

- Compare pairwise all ”objects” that we estimate for X_i and X_j – are they:
  - $X_i >> X_j$
  - $X_i > X_j$
  - $X_i = X_j$
- This yields a *relative comparison* of sizes
- Estimate the absolute value for a number of ”fix points”
- Calculate the *total*, here:

  +350/1,75
  +3*350
  +2*475
  +2*4*475
  = 6000h



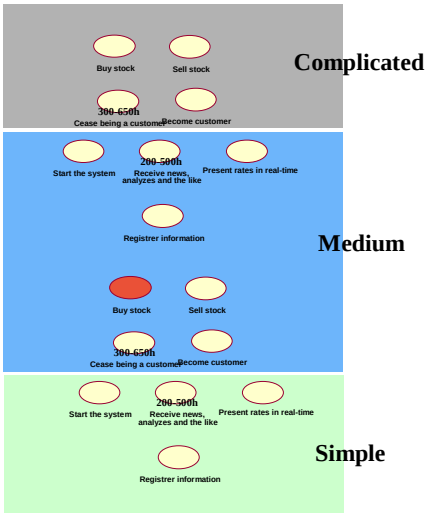
Tids- och kostnadsuppskattning
av RUP Projekt: # **17**
RAIN 2002

KVADRAT

IT ARKITEKTERNA

Real world case 2

- Administrative system with 110-120 use cases
- Migrating legacy to modern framework
- Implemented *1 use case* and measured time for that
- Arranged all use cases i 3 sets:
  - Complicated
  - Medium
  - Simple
- Compared to implemented use case and developed estimates during Elaboration
- Adjusted new estimates to the fact that next use cases should be quicker since experience with new framework was building
- Presented estimates to management
- Project was cancelled – should have taken to much time and effort!

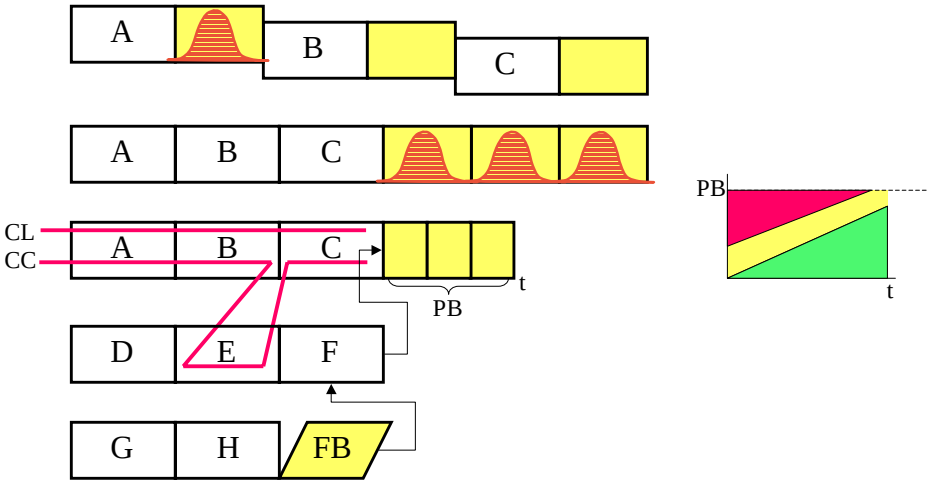


Tids- och kostnadsuppskattning
av RUP Projekt: # **18**
RAIN 2002

KVADRAT

IT ARKITEKTERNA

Critical Chain



KVADRAT

IT ARKITEKTERNA

Critical Chain

- **Based on Critical Chain by Goldratt**
- **Theory of constrains**
- **Aggressive planning & Reduce Multitasking**
- **Critical Chain vs. Critical Line**
- **Buffer management**
  - **The Project Buffer protects the delivery date**
  - **The Feeding Buffer protects the Critical Chain**
  - **The Capacity Buffer protects the critical resources in a multi-project environment**

KVADRAT

IT ARKITEKTERNA

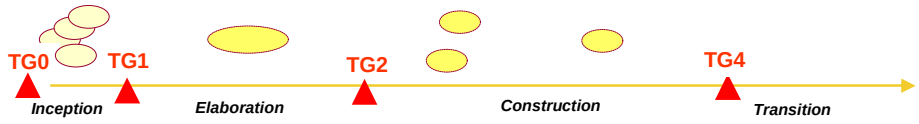
Technique	Advantage	Disadvantage
Analogy	Experience Based	Standard project?
Expert judgement	Fast, compensates for circumstances	Biases, incomplete recall of experiences
Delphi	Formal, Can use a mix of methods	As good as the experts
Algorithmic Models	Objective, Efficient, Calibrated	Subjective inputs, Inflexible, Must be calibrated
Critical Chain	Reduced time schedule	Extra pressure (?)
Lichtenberg	Top-down view, Risk Focus	Mentally demanding
Paired Comparison	Suggests high fidelity. Practical if comparisons can be reduced.	Cumbersome? Unproven in academia

KVADRAT

IT ARKITEKTERNA

Real world case 3

- **Estimated effort for each use case during Inception:**
  - **Requirements**
  - **Design**
  - **Implementation**
  - **Test**
- **Allocated use cases to iterations**
- **Experiences – Sufficiently good. Precision mostly OK, large deviations (up to +100%) sometimes due to:**
  - **New architecture**
  - **Use cases with complex business rules**
  - **Most deviations struck execution of Elaboration phase – iteration 1**
- **Caution:Early estimates may turn out to be completely wrong if elaboration fails to stabilize architecture**

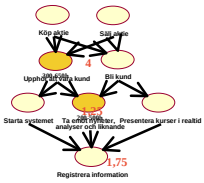


KVADRAT

IT ARKITEKTERNA

A sketch of a practical method for *Inception*

- **Order use cases with *paired comparison* according to effort**
- ***Detail a few use cases*, and analyze them in detail – estimate with Lichtenberg**
- **Use paired comparison to *estimate rest* of use cases**
- **Estimate *impact of mechanisms, risks and missing use cases* with Lichtenberg**
- **Make an iterative plan for use case and mechanism realizing**
- **Make schedule based on plan**
- **No commitment yet!**



+

Missing use cases: + 10%
Non-functional requirements: 690h
Mechanisms: 1550h
Technical risk impact: 1300h
Non-technical risk impact: 1200h

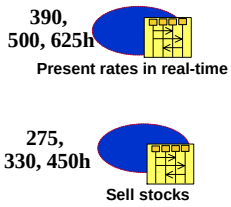
Tids- och kostnadsuppskattning
av RUP Projekt: # **23**
RAIN 2002

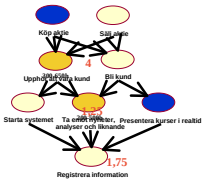
KVADRAT

IT ARKITEKTERNA

A sketch of a practical method for *Elaboration*

- **Learn from Inception time spent vs plan**
- **Improve estimates when there are use case realizations available**
- **Adjust to current knowledge of risks**
- **Study trends in growth in number of use cases**
- **Let the developers do the estimates!**
- **Manage the trend estimates**





+

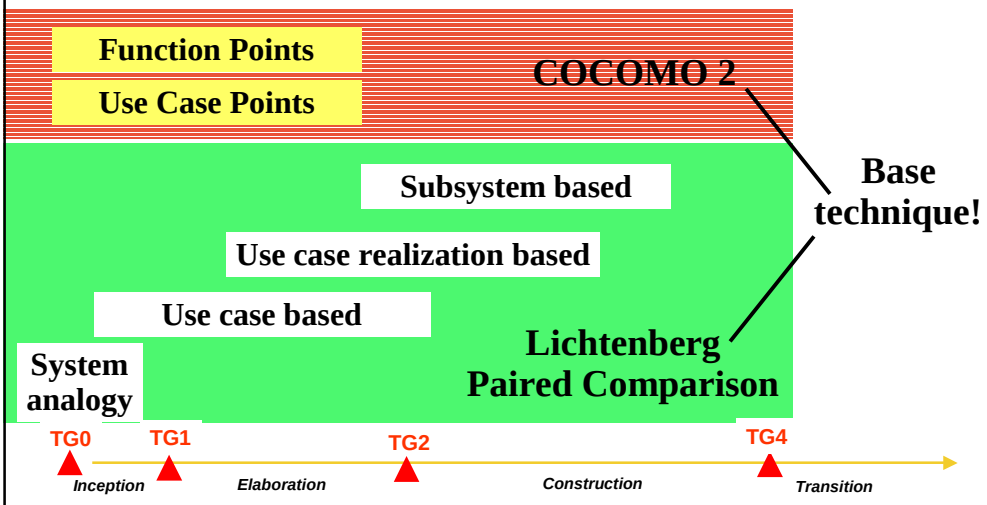
Non-functional requirements: 470h
Mechanisms: 250h
Technical risk impact: 3500h
Non-technical risk impact: 1000h

Tids- och kostnadsuppskattning
av RUP Projekt: # **24**
RAIN 2002

KVADRAT

IT ARKITEKTERNA

When to use different techniques?



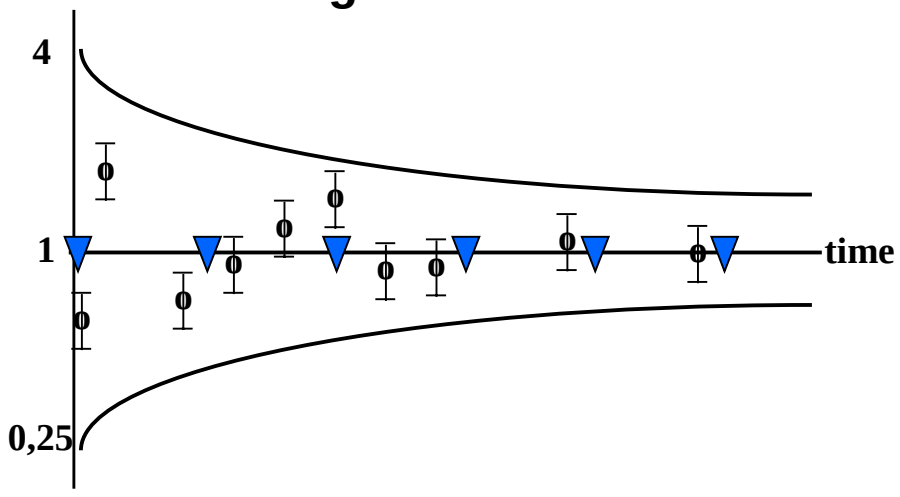
TG = Tollgate i PROPS

Tids- och kostnadsuppskattning
av RUP Projekt: # **25**
RAIN 2002

KVADRAT

IT ARKITEKTERNA

Manage the trends!



Tids- och kostnadsuppskattning
av RUP Projekt: # **26**
RAIN 2002

KVADRAT

IT ARKITEKTERNA

Next steps

- **Start using one technique**
- **Use it from day one in your project**
- **Let the project members do the estimation**
- **Collect experiences and improve**
- **Call us if you need more**

- **Thanks for your time!**

KVADRAT

IT ARKITEKTERNA

References

- Walker Royce, Software Project Management - A Unified Framwork, Addison Wesley Longman, Reading, Massachusetts, 1998, 406 pp. ISBN 0-201-30958-0
- Eduardo Miranda, Improving Subjective Estimates using Paired Comparison, IEEE Software, Jan/Feb 2001.
- Steen Lichtenberg, Projektplanlaeggning i en foranderlig verden, Polyteknisk Forlag, Lyngby, 1978
- Barry Boehm, et. al., Software Cost Estimation with COCOMO II, Prentice Hall PTR, 2000, ISBN: 0130266922
- Goldratt, Critical Chain

KVADRAT

IT ARKITEKTERNA