

CS445 / ECE451 / CS645 / SE463
Software Requirements Specification & Analysis

Estimating Effort



#1 Estimating Effort by Analogy

Comparison to Past Projects – compute an early estimate of effort (range of values) based on the effort from **analogous** past projects

$$\text{Effort} = \text{PastEffort} \times (\text{Size} / \text{PastSize})$$

| Project | Size (LOC) | Schedule (Calendar Months) | Effort (Staff Months) | Productivity (LOC/Staff Month) |
|-----------|------------|----------------------------|-----------------------|--------------------------------|
| Project A | 33,842 | 8.2 | 21 | 1,612 |
| Project B | 97,614 | 12.5 | 99 | 986 |
| Project C | 7,444 | 4.7 | 2 | 3,722 |
| Project D | 54,322 | 11.3 | 40 | 1,358 |
| Project E | 340,343 | 24.0 | 533 | 639 |

Steve McConnell, *Software Estimation: Demystifying the Black Art*, Microsoft Press, 2006. (Table 19.2)

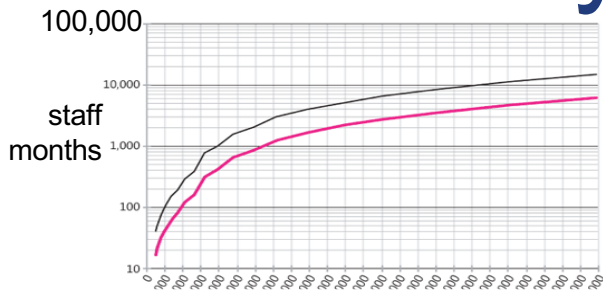
#1 Estimating Effort by Analogy

| Project | Size (LOC) | Schedule (Calendar Months) | Effort (Staff Months) | Productivity (LOC/Staff Month) |
|-----------|------------|----------------------------|-----------------------|--------------------------------|
| Project A | 33,842 | 8.2 | 21 | 1,612 |
| Project B | 97,614 | 12.5 | 99 | 986 |
| Project D | 54,322 | 11.3 | 40 | 1,358 |

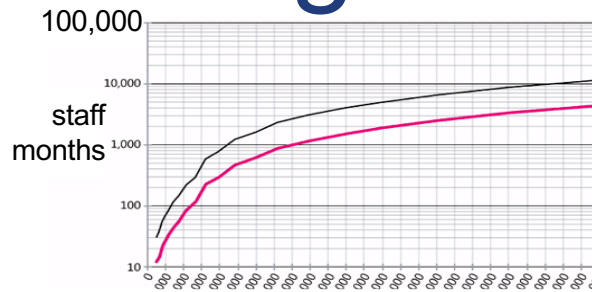
| New Project | Estimated Size (LOC) | Estimated Effort (staff month) |
|-------------------|----------------------|--------------------------------|
| Best Estimate | 65,000 | $65,000/986 = 65$ |
| Worst Estimate | 100,000 | $100,000/986 = 101$ |
| Expected Estimate | 80,000 | $80,000/986 = 80$ |

Steve McConnell, *Software Estimation: Demystifying the Black Art*, Microsoft Press, 2006. (Table 19.2)

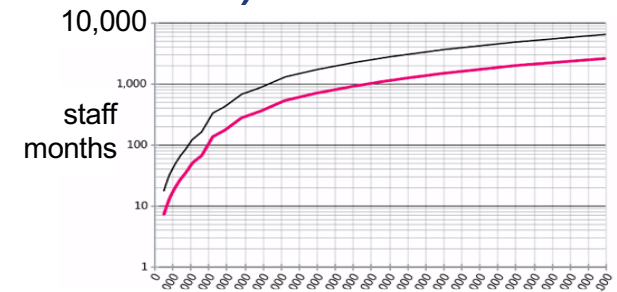
#2 Industry-Average Effort(2006)



driver software

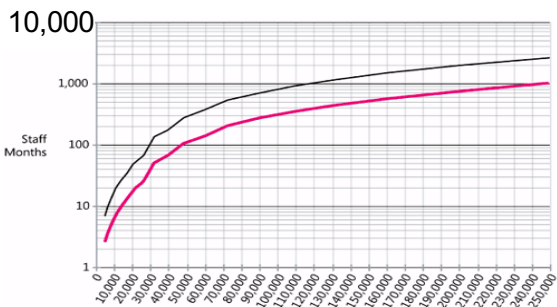


scientific research software

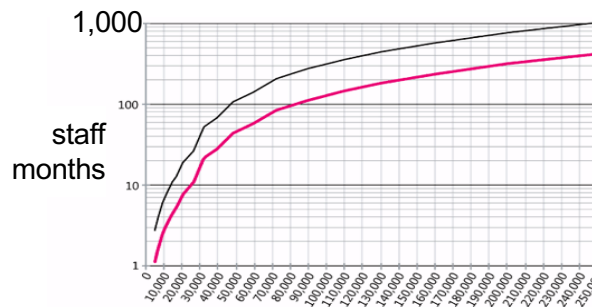


user applications

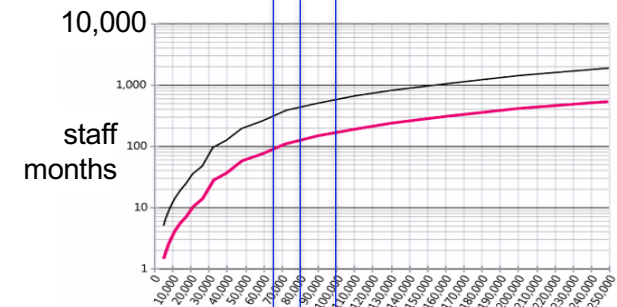
| New Project | Estimated Size (LOC) | Industry-Average Effort (staff month) |
|-------------------|----------------------|---------------------------------------|
| Best Estimate | 65,000 | 85 |
| Worst Estimate | 100,000 | 170 |
| Expected Estimate | 80,000 | 120 |



public internet software




private intranet projects



business applications

Steve McConnell, *Software Estimation: Demystifying the Black Art*, Microsoft Press, 2006. (Tables 19.1-19.9)

#3 ISBSG Estimating Tool



International Software Benchmarking Standards Group
Comparative Estimating Tool V6.0
 using the ISBSG project data

Inputs

Filters

Functional size (function points):

Development platform:

Development type:

Parameters

Functional size (function points):

Range minimum (percentile):

Range maximum (percentile):

Project attributes

Primary programming language:

Organisation type:

Application type:

Maximum team size:


Used prototyping:

Used JAD:

Web development:

| Number of matches | Project Delivery Rate | | | Speed of Delivery | | |
|-------------------|-----------------------|---------------|--------------|-------------------|---------------|------------|
| | Optim 20 % | Likely Median | Conserv 80 % | Conserv 20 % | Likely Median | Optim 80 % |
| 43 | 8.2 | 16.5 | 31.4 | 19.5 | 30.7 | 55.5 |
| 256 | 6.7 | 15.9 | 33.7 | 18.3 | 44.7 | 105.5 |
| 10 | 9.3 | 21.1 | 31.2 | 11.0 | 23.2 | 34.6 |
| 164 | 3.0 | 6.2 | 13.9 | 16.7 | 36.0 | 83.0 |
| 0 | | | | | | |
| 76 | 5.3 | 13.5 | 35.4 | 16.3 | 37.9 | 79.0 |
| 341 | 4.8 | 7.3 | 15.8 | 18.3 | 34.2 | 91.4 |

| Estimates | Project Delivery Rate | Project Work Effort | Speed of Delivery | Project Duration |
|--------------------|----------------------------|---------------------|-----------------------------|------------------|
| Level 1 - Dev Team | (hours per function point) | (hours) | (function points per month) | (months) |
| Optimistic | 6.2 | 621 | 74.8 | 1.3 |
| Likely | 13.4 | 1,342 | 34.4 | 2.9 |
| Conservative | 26.9 | 2,691 | 16.7 | 6.0 |

Developed in conjunction with  **CHARISMATEK**
 SOFTWARE METRICS

ISBSG Comparative Estimating Tool User Guide (2015)

#1 Estimating Schedule

There is a **Basic Schedule Equation** that produces good-enough estimates of schedule early in a medium-to-large sized project.

$$\text{ScheduleInMonths} = 3.0 \times \text{StaffMonths}^{1/3}$$

(where multiplier ranges from 2.0 to 4.0).

#2 Estimating Schedule by Analogy

If you have historical data, you can compute a schedule estimate based on the schedules from similar past projects

Medium-to-large projects (more than 50 staff months)

$$\text{ScheduleInMonths} = \text{PastSchedule} \times (\text{EstimatedEffort} / \text{PastEffort})^{1/3}$$

Small projects

$$\text{ScheduleInMonths} = \text{PastSchedule} \times (\text{EstimatedEffort} / \text{PastEffort})^{1/2}$$

#2 Estimating Schedule by Analogy

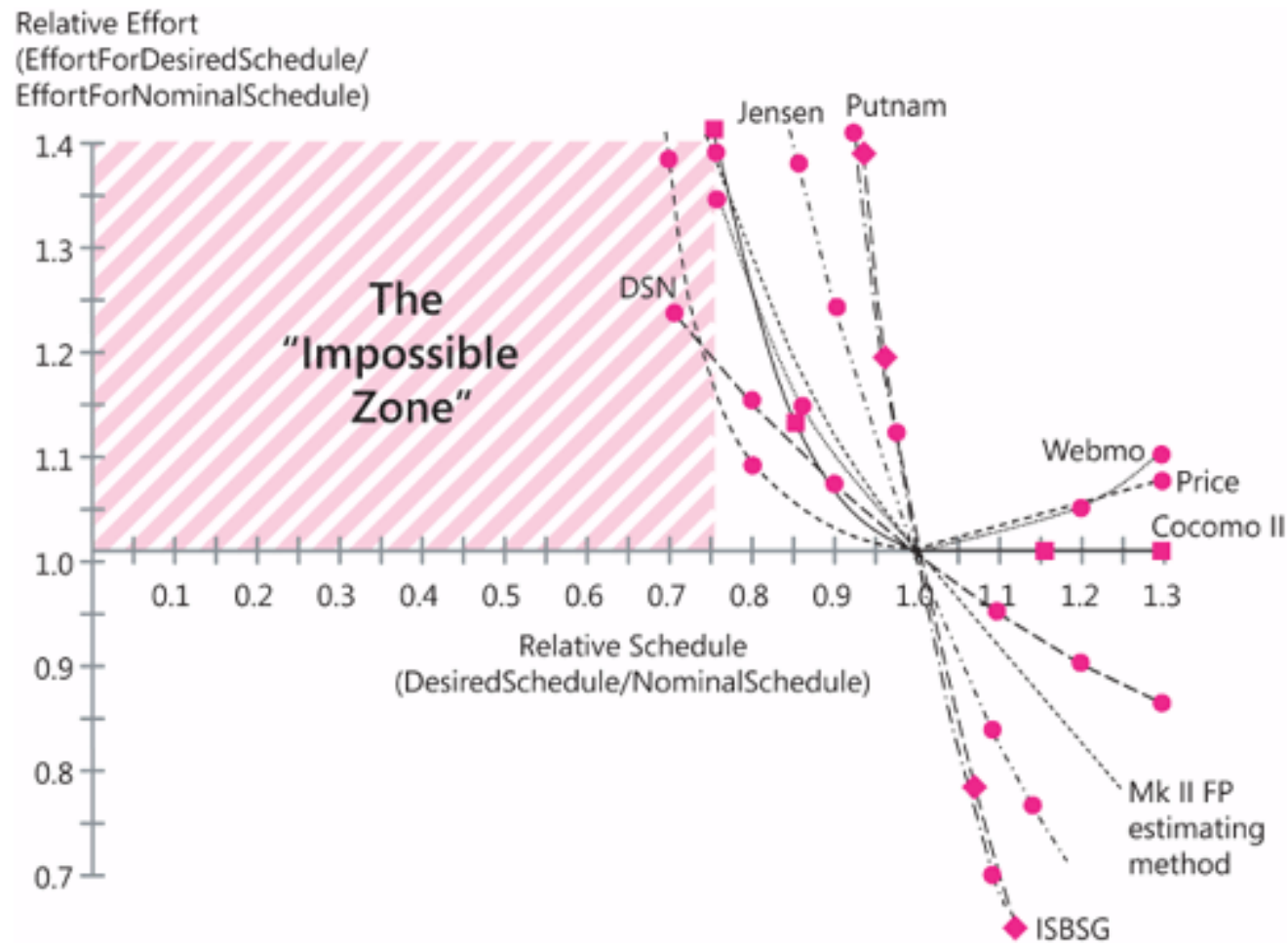
Recall our example new project whose estimated effort ranges 65 to 100 staff months, with an expected effort of 80 staff months

| Project | Past Schedule (Months) | Past Effort (Staff Effort) | Low Estimate (Months) | Nominal Estimate (Months) | High Estimate (Months) |
|----------------|------------------------|----------------------------|--------------------------------|--------------------------------|---------------------------------|
| Project A | 8.2 | 21 | $8.2 * (65/21)^{0.33} = 12.0$ | $8.2 * (80/21)^{0.33} = 12.8$ | $8.2 * (100/21)^{0.33} = 13.8$ |
| Project B | 12.5 | 99 | $12.5 * (65/99)^{0.33} = 10.8$ | $12.5 * (80/99)^{0.33} = 11.6$ | $12.5 * (100/99)^{0.33} = 12.5$ |
| Project D | 11.3 | 40 | $11.3 * (65/40)^{0.33} = 13.2$ | $11.3 * (80/40)^{0.33} = 14.2$ | $11.3 * (100/40)^{0.33} = 15.3$ |
| Average | | | 12.0 | 12.9 | 13.9 |

$$\text{ScheduleInMonths} = \text{PastSchedule} \times (\text{EstimatedEffort} / \text{PastEffort})^{1/3}$$

$$\text{ScheduleInMonths} = \text{PastSchedule} \times (\text{EstimatedEffort} / \text{PastEffort})^{1/2}$$

Schedule Compression



Steve McConnell, *Software Estimation: Demystifying the Black Art*, Microsoft Press, 2006. (Figure 20.2)

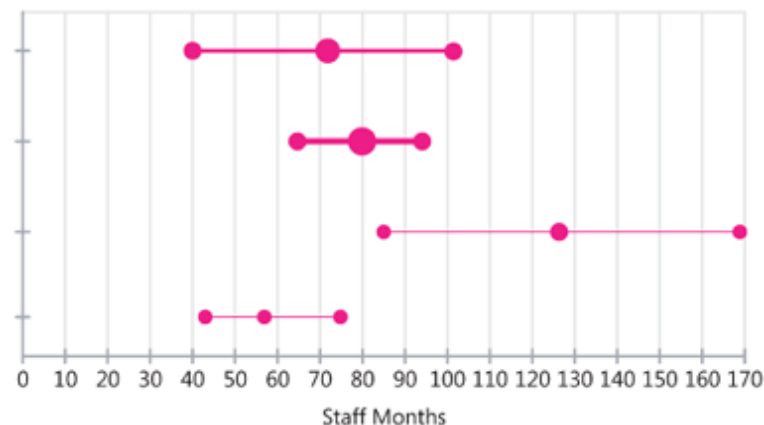
Best Practices

Decompose the estimation task into multiple subtasks and combine the results

- some estimation errors will cancel each other out

Use multiple estimation techniques and compare the results

- convergence suggests that you have good accuracy
- spread suggests some factors have been overlooked



Steve McConnell, *Software Estimation: Demystifying the Black Art*, Microsoft Press, 2006. (Figure 10.10)

Best Practices

Decompose the estimation task into multiple subtasks and combine the results

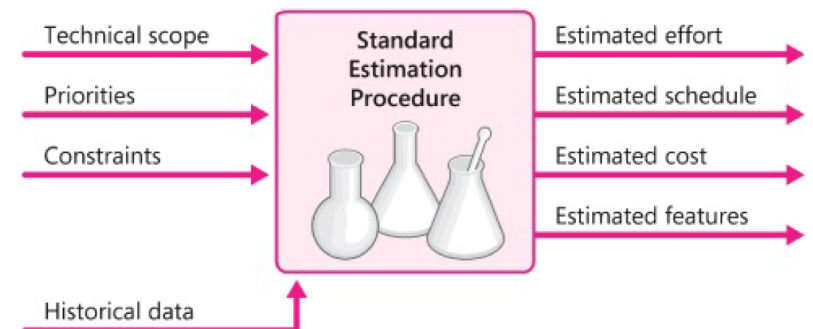
- some estimation errors will cancel each other out

Use multiple estimation techniques and compare the results

- convergence suggests that you have good accuracy
- spread suggests some factors have been overlooked

Define the estimation procedure in advance

- what granularity of task to count
- what multiple techniques to use
- when to re-estimate
- what project data to collect



Steve McConnell, *Software Estimation: Demystifying the Black Art*, Microsoft Press, 2006. (Figure 16.2)

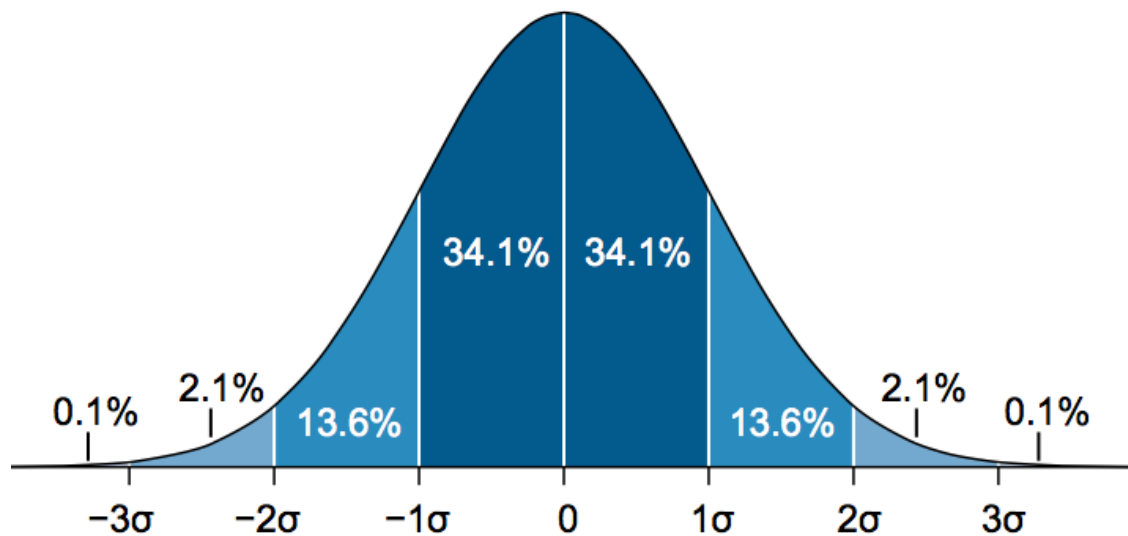
Recombining Piecewise Estimates

Suppose you have a software estimate for a new project or iteration

| Use Case | Best Case (staff months) | Expected Case (staff months) | Worst Case (staff months) |
|--------------|-----------------------------|---------------------------------|------------------------------|
| UC 1 | 4.1 | 5.0 | 6.0 |
| UC 2 | 5.2 | 6.4 | 7.7 |
| UC 3 | 6.0 | 7.1 | 8.2 |
| UC 4 | 2.4 | 3.1 | 3.6 |
| UC 5 | 5.8 | 7.0 | 8.2 |
| TOTAL | 23.5 | 28.6 | 33.7 |

Standard Deviations

A common approximation is to assume that the range between a minimum and a maximum is ± 3 standard deviations.



Aggregate Standard Deviation

Combine the standard deviations of the piecemeal estimates by summing their variances and taking the square root.

| Use Case | Best Case (BC) | Worst Case (WC) | Standard Deviation (WC-BC)/6 | Variance (SD ²) |
|--|----------------|-----------------|---------------------------------|--------------------------------|
| UC 1 | 4.1 | 6.0 | 0.317 | 0.100 |
| UC 2 | 5.2 | 7.7 | 0.417 | 0.174 |
| UC 3 | 6.0 | 8.2 | 0.367 | 0.135 |
| UC 4 | 2.4 | 3.6 | 0.20 | 0.040 |
| UC 5 | 5.8 | 8.2 | 0.40 | 0.160 |
| TOTAL | --- | --- | | 0.608 |
| Combined Standard Deviation | | | | 0.78 |

Precision

More realistic estimates of the SDs of the piecemeal estimates will consider the best-case to worst-case range to have a smaller confidence interval (and thus a SD is a larger fraction of the range)

| Confidence Interval | Divisor in calculations of SDs of piece estimates |
|---------------------|---|
| 10% | 0.25 |
| 20% | 0.51 |
| 30% | 0.77 |
| 40% | 1.0 |
| 50% | 1.4 |
| 60% | 1.7 |
| 70% | 2.1 |
| 80% | 2.6 |
| 90% | 3.3 |
| 99.7% | 6.0 |

Aggregate Standard Deviation

| Use Case | Best Case (BC) | Worst Case (WC) | Standard Deviation (WC-BC)/2 | Variance (SD ²) |
|--|----------------|-----------------|---------------------------------|--------------------------------|
| UC 1 | 4.1 | 6.0 | 0.95 | 0.902 |
| UC 2 | 5.2 | 7.7 | 1.25 | 1.56 |
| UC 3 | 6.0 | 8.2 | 1.1 | 1.21 |
| UC 4 | 2.4 | 3.6 | 0.6 | 0.36 |
| UC 5 | 5.8 | 8.2 | 1.2 | 1.44 |
| TOTAL | --- | --- | | 5.47 |
| Combined Standard Deviation | | | | 2.34 |

Estimate = 28.6 ± 3σ = 28.6 ± 7.02 staff months

Confidence Interval

Can use a table of standard deviations to compute percentage-confident estimates.

| Percentage Confident | Calculation | Percentage Confident | Calculation |
|----------------------|-----------------------------|----------------------|-----------------------------|
| 2 | Expected Case – (2 x SD) | 60 | Expected Case + (0.25 x SD) |
| 10 | Expected Case – (1.28 x SD) | 70 | Expected Case + (0.52 x SD) |
| 16 | Expected Case – (1 x SD) | 75 | Expected Case + (0.67 x SD) |
| 20 | Expected Case – (0.84 x SD) | 80 | Expected Case + (0.84 x SD) |
| 25 | Expected Case – (0.67 x SD) | 84 | Expected Case + (1 x SD) |
| 30 | Expected Case – (0.52 x SD) | 90 | Expected Case + (1.28 x SD) |
| 40 | Expected Case – (0.25 x SD) | 98 | Expected Case + (2 x SD) |
| 50 | Expected Case | | |

Significant Digits

| Percentage Confident | Effort Estimate (staff months) |
|----------------------|---|
| 25 | $28.6 - (0.67 \times 2.34) = \cancel{27.0322}$ 27 |
| 50 | 28.6 |
| 75 | $28.6 + (0.67 \times 2.34) = \cancel{30.1678}$ 30 |
| 90 | $28.6 + (1.28 \times 2.34) = \cancel{31.5952}$ 32 |

Report estimates in significant digits and units that are consistent with the estimate's underlying accuracy and precision

Project Control

Once we make an estimate and make a commitment to deliver functionality and quality by a particular date, then we **control** the project to meet the target.

- remove noncritical requirements
- redefine requirements
- replacing less-experienced staff with more-experienced staff

If the initial target and initial estimate are within about 20% of each other, the project manager should have enough maneuvering room to meet the project's business goals.

Software Estimations

Size Estimation (LOC)

->

Effort Estimation (Person Months)

->

Schedule (Months)

-
- | | | |
|--|--|---|
| <ul style="list-style-type: none">• Analogy (old project code size -> estimated code size)• Function Points -> LOC• Structured Expert Judgement (PERT)• Commercial Tool | <ul style="list-style-type: none">• Informal/Analogy (old project effort -> estimated effort)• Industry-Average Data• ISBSG Tool (FPs x MaxTeamSize -> Effort) | <ul style="list-style-type: none">• Basic Schedule Equation• Analogy (old project schedule -> estimated schedule) |
|--|--|---|
-
- Best Practices (piecewise estimation, multiple techniques, best/expected/worst estimations)
 - Percentage-Confidence Estimates
 - Project Control

Personal Software Estimation

Data Collection

In order to compute software estimates based on historical data, you need to be collecting data about your project (every 1-2 weeks)

Per requirement (e.g., use case, scenario, story, screen)

- lines of code (LOC)
- effort (staff hours)
- time (calendar hours)

Software Estimation

- effort of whole project (as sum of efforts of use cases, increments)
- effort of next increment (as sum of efforts of atomic requirements)

References

McConnell, S., *Software Estimation: Demystifying the Black Art*, 2006.



**UNIVERSITY OF
WATERLOO**

All rights, including copyright, in the content of these slides and video are owned by the course author. The slides and videos are owned by the University of Waterloo. For further information, please contact the course author Joanne Atlee, jmatlee@uwaterloo.ca.