MASTER SYSTEMS

Traditional project management methods do not support knowledge work: Do our PM methods fall short for the 21st Century?

DR. STAN RIFKIN

MASTER SYSTEMS INC.

11160-C1 SOUTH LAKES DR. 611

RESTON, VIRGINIA 20191 USA

& +1 703 599 0650 SR @ MASTER-SYSTEMS.COM

Ver. 1.2

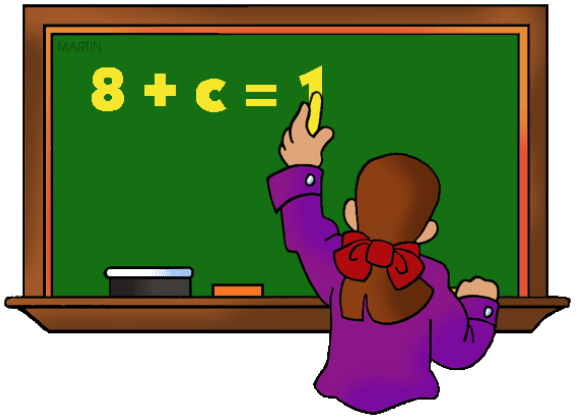
Abstract

There is little new in project management since the 1950s. Yet the subjects of PM have grown to include projects whose work products are invisible (e.g., designs, ideas, discoveries, software, novels, movies, works of art, organizational culture change).

What of traditional PM can be applied to the creation of invisible products? What else is needed?

1. Traditional PM assumes linear relationships

- **Duration = Effort required / Resources applied is built into every PM software tool, as well as most PM books and training materials.**
- **Gives rise to, "We are increasing scope a little, so we shall have to increase resource a little, too."**



Plumbers vs. programmers

- **Duration = Effort required / Resources available**

Imagine:

A good plumber can install 12 toilets in a day.

You have ten plumbers,

Whom you've given 1/2 day.

So, how many toilets will they install?

½ day = 1/12*x* / 10

x = 60

Plumbers (cont.)

Imagine:

A good programmer can write 12 thousand lines of code in a year.

You have ten such programmers.

Therefore, how many lines will the team write in 6 months?

From the previous example, sixty thousand.

BUT, from data on actual, completed software projects, the team will write only 30 thousand!

Why does it take longer for knowledge work?

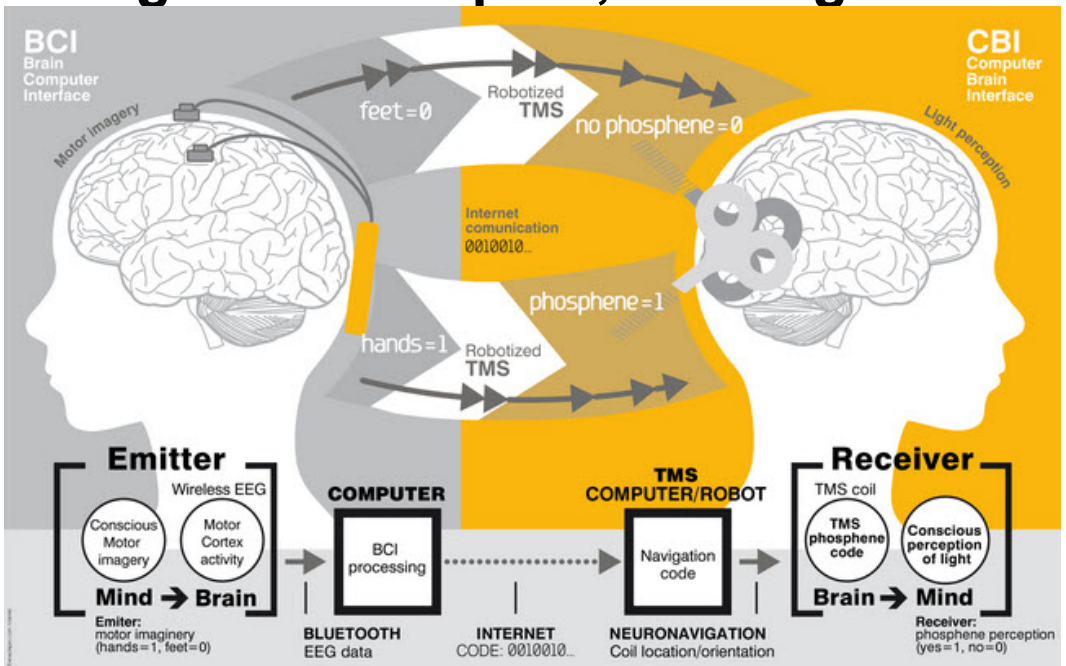
- **Communication overhead**

  As we add each person, he/she needs to communicate with all of the rest of the staff, so we get less than a full person with each addition. In fact, the percent of a full person that we get goes DOWN as team size goes UP.

There is some inherent sequentiality in knowledge work. It is incompressible at some point. What point?

Why does it take longer for knowledge work? (cont.)

There is a "ramp up" time when adding resources to knowledge work projects. And who is most impacted by the on-boarding process? It's the most talented, most productive team members as they scurry to transfer knowledge to the new project team members, even expert ones. So, those talented, productive members STOP working on the outputs, which grinds the project to a halt!



Why does it take longer for knowledge work? (cont.)

- **Bottom-up, WBS-based estimating does not account for bugs, rework, changed scope, waiting, and all of the many interdependencies. They are always there, but we don't ever seem to expect them.**

	Deliveries			Revenues per then-year list price			Boeing List Prices			
Airplane Model	2013	2012	%	2013 $ mill.	2012 $ mill.	%	Airplane Model	2013 $ mill.	2012 $ mill.	%
737-700	12	7	71.4%	912	524	74.2%	737-700	76	74.8	1.6%
737-800	347	351	-1.1%	31,404	31,274	0.4%	737-800	90.5	89.1	1.6%
737-800A (P-8)	8	9	-11.1%	724	802	-9.7%	*737-800 (P-8)*	*90.5*	*89.1*	*1.6%*
737-900ER	67	44	52.3%	6,439	4,162	54.7%	737-900ER	96.1	94.6	1.6%
777-200ER	4	3	33.3%	1,046	776	34.7%	777-200ER	261.5	258.8	1.0%
777-200LR	1	1	0.0%	296	291	1.6%	777-200LR	296	291.2	1.6%
777-300ER	79	60	31.7%	25,296	18,900	33.8%	777-300ER	320.2	315	1.7%
777F	14	19	-26.3%	4,207	5,618	-25.1%	777 Freighter	300.5	295.7	1.6%
767-300ER	9	14	-35.7%	1,672	2,559	-34.7%	767-300ER	185.8	182.8	1.6%
767-300F	12	12	0.0%	2,256	2,225	1.4%	767-300 F	188	185.4	1.4%
747-8F	19	19	0.0%	6,793	6,688	1.6%	747-8 Freighter	357.5	352	1.6%
747-8	5	12	-58.3%	1,785	4,217	-57.7%	747-8	356.9	351.4	1.6%
787-8	65	46	41.3%	13,767	9,513	44.7%	787-8	211.8	206.8	2.4%
BBJ (737-based)	5	2	150.0%	481	189	154.0%	*737-900ER*	*96.1*	*94.6*	*1.6%*
BBJ2 (737-based)	1	2	-50.0%	96	189	-49.2%	*737-900ER*	*96.1*	*94.6*	*1.6%*
	648	601	7.8%	97,172	87,928	10.5%				1.6%
				BCA reported revenues	52,981	49127	7.8%			
				BCA Services revenues (est.)	674	1384	-51.3%			
				BCA platforms revenues (est.)	52,307	47,743	9.6%			
				Estimated discount (*)	-46.2%	-45.7%				

The Blog by Javier

(*) This simplified calculation excludes impact on potential revenues linked to downpayments at ordering.

How do I know I have a traditional or a knowledge work project?

The plumber test:

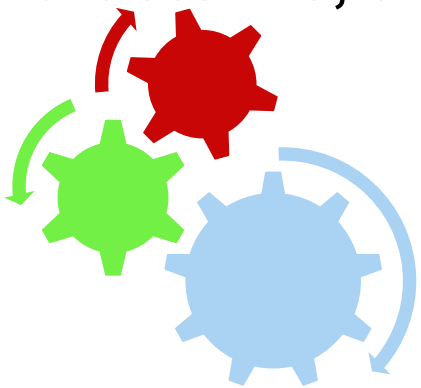
- **If I double the number of people on the project does it get done in half the time?**
  - **If yes, then you have a traditional project. Use the *PMBOK Guide*.**
  - **If doubling the staff STOPS the project then you have a knowledge work project! And read on!**

2. What if we are trying to optimize something besides completion time?

- Traditional PM is completely organized around duration.
- What if our knowledge project instead is trying to optimize:
  - Product quality (shortest schedule = lowest quality).
  - Usability (or any other –ility) (shortest schedule = lowest value of the –ility).
  - Cost (shortest schedule = highest cost).
  - Connection, amazement (e.g., marketing campaign, mission statement).
  - Creation of something really new.
  - "Good enough" outcome.

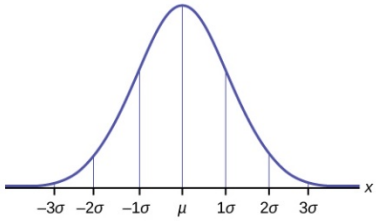
3. What if our work is iterative, can't be accomplished in a single cycle?

- Most (all?) knowledge work is iterative, because we do not know the answer beforehand – by definition. So, we start somewhere in the solution space and keep searching until we are satisfied.
- *PMBOK Guide* and the *Software Extension* both permit iterations, but give NO guidance on how to estimate the number and duration, nor what tools might be used to characterize, track, replan them.

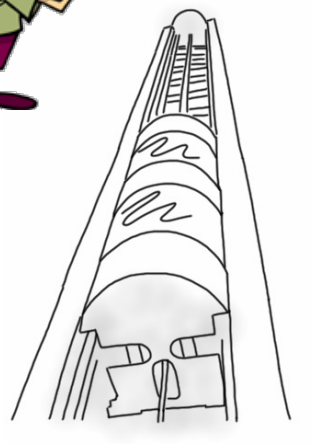


4. We don't use what we already know

- PERT values for optimistic, pessimistic and most likely durations, for example, almost never depend upon actual historical project information, but rather "aerial extraction."
- Problem with Monte Carlo simulation of project plans: the distribution of task durations is symmetric (same chance that tasks finish early as finish late).
- Dependencies may not be (just) precedences as much as they are requirements (software is not finished until it's integrated). In other words, it's not always about timing, critical path.

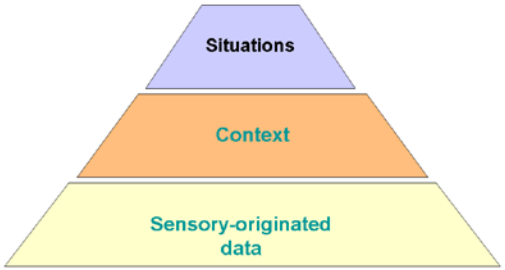


4. We don't use what we already know (cont.)

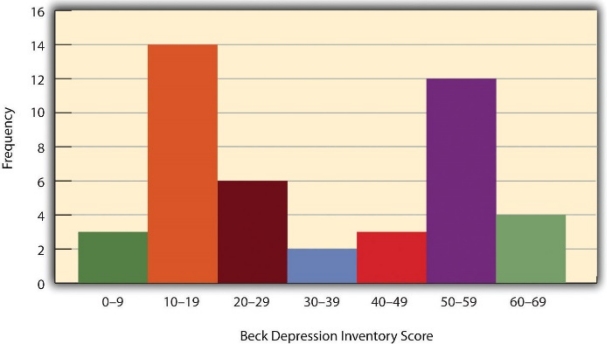


- How many schedules have a re-work task after every review? Yet, why have a review if there is not going to be any re-work? And how should the re-work effort be estimated?
- How many schedules take into account resource scarcity? What if all of the plumbers had to use just a single elevator?
- How many of us collect project execution information and use it to estimate new projects going forward? Do we collect context information along with it so that we know under what conditions to apply what we hav captured?

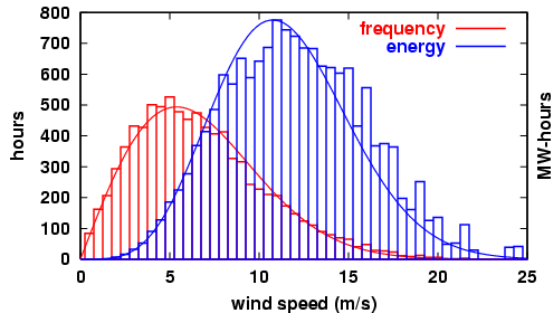
Context-Situation pyramid



4. We don't use what we already know (cont.)

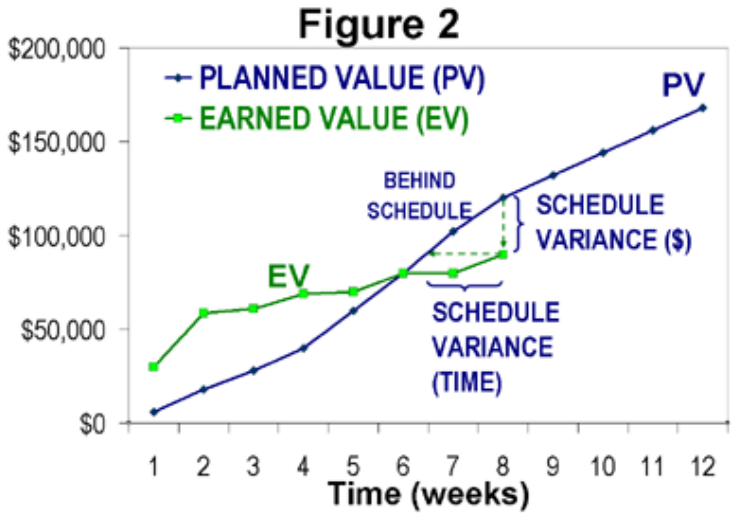


- How many of us use averages of past data for new estimates? What makes us think that averages are at all informative? What if the distribution of the data is bimodal? Same problem with statistics applied to everything else, too (e.g., likely variances).



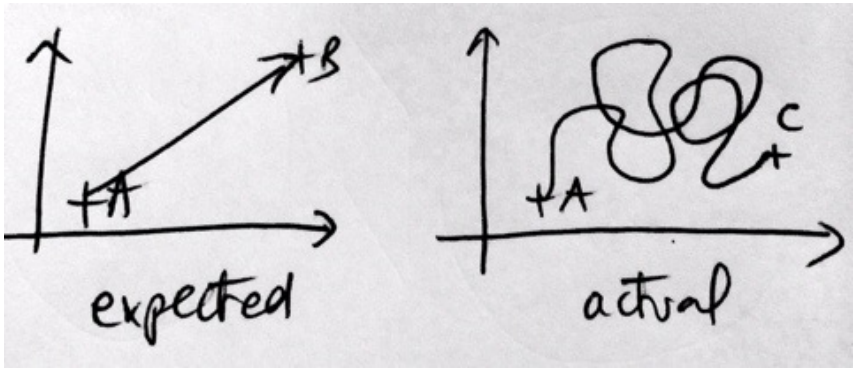
- If typical staffing for projects follows a curve, do we estimate by fitting a curve for a new project to the past ones? (Using averages?!)
- Is there more stability in macro-estimating than bottom-up? Yes, so do we use macro-estimating?

5. Existing tools mislead when work products are invisible



- **These do not work (effectively):**
  1. **Percent complete. Since the product is invisible one cannot know how complete it is – until the END!**
  2. **Earned value (EVA). This computes what is burned (duration & effort), not what is accomplished. It only works if burning = accomplishment, which it NEVER does in knowledge work.**
  3. **Work breakdown schedule. While it's handy to know the totality of the work, WBS generally do not capture it. They omit bugs, rework, changed scope, waiting.**

5. Existing tools mislead (cont.)



- **List (cont.):**
  4. **If there is a variance, what is the cause? Say, productivity. Then how will one change the schedule going forward? That is, the schedule does not shift to the right by a constant (= the slip after entering a task actual completion date). Productivity is ASSUMED in Microsoft Project & cannot be changed.**
  5. **There are no loops (iterations) in most current PM tools. Loops have to be un-wound to be represented in MS Project.**

5. Existing tools mislead (cont.)

- **List (cont.):**
  6. **If macro-estimating (top-down) creates more stable estimates, then why don't our tools support it?**

OK, some answers! R&D is filled with non-linear relationships

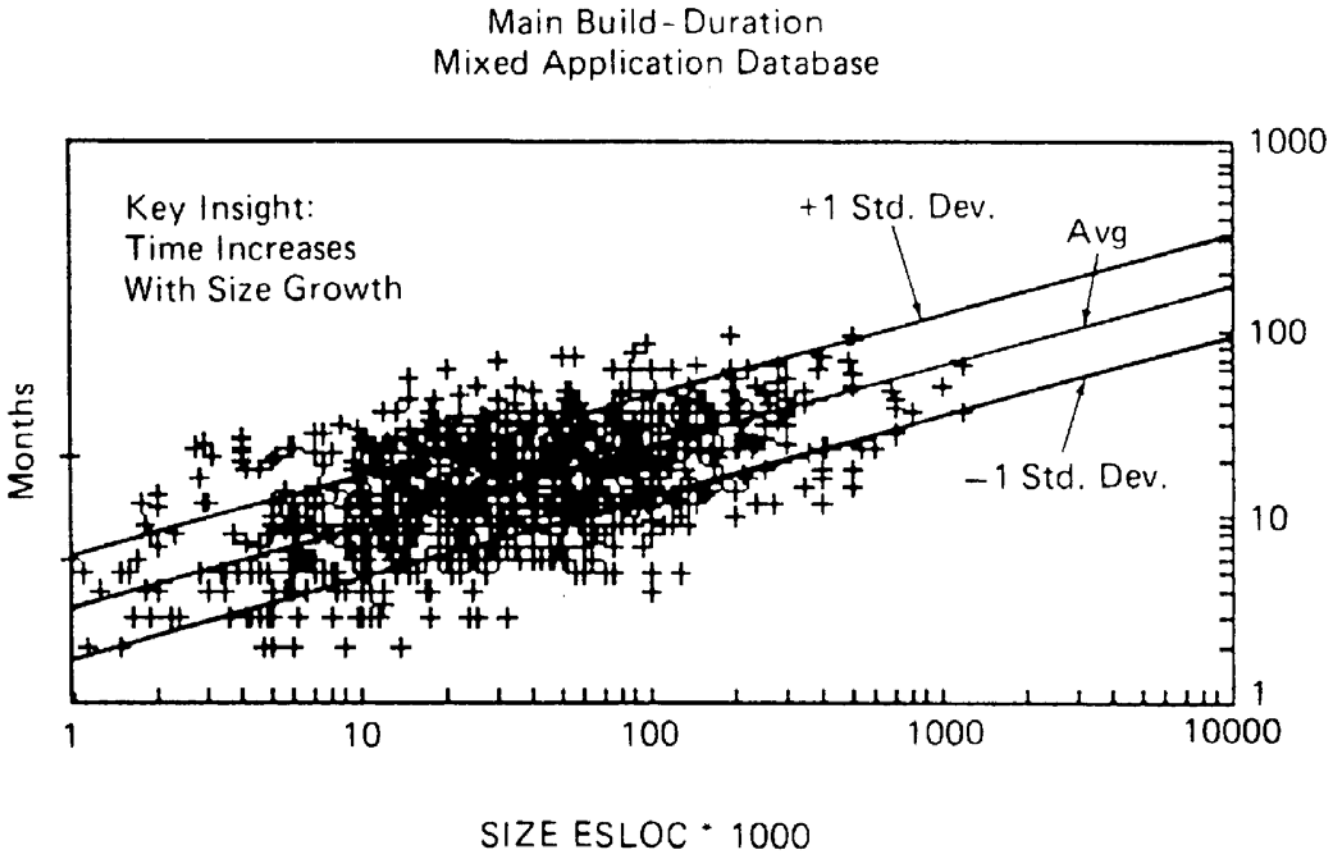


FIGURE 1.6. Trend lines on the QSM database reveal the same pattern as Nelson found: project duration increases with system growth.

Challenges *(cont.)*

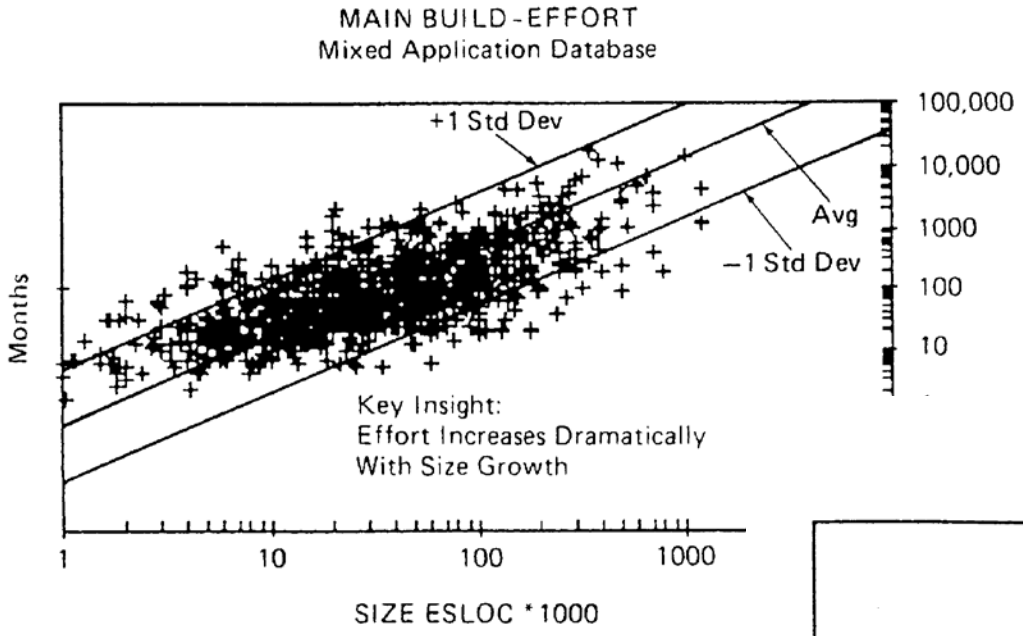


FIGURE 1.7. Effort increases far more rapidly than c growth. Note the difference in vertical scale between Figu

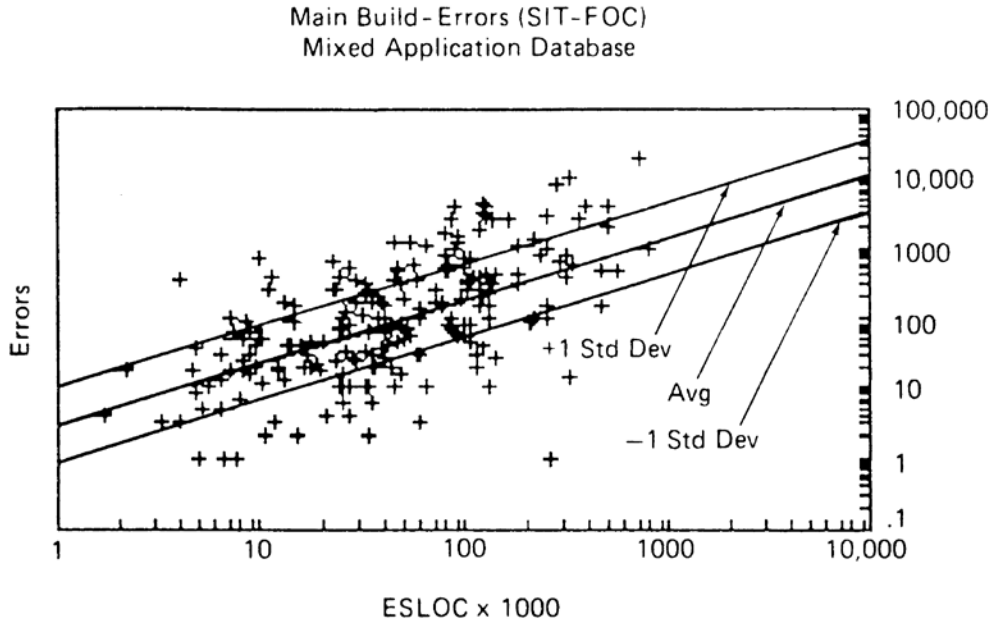
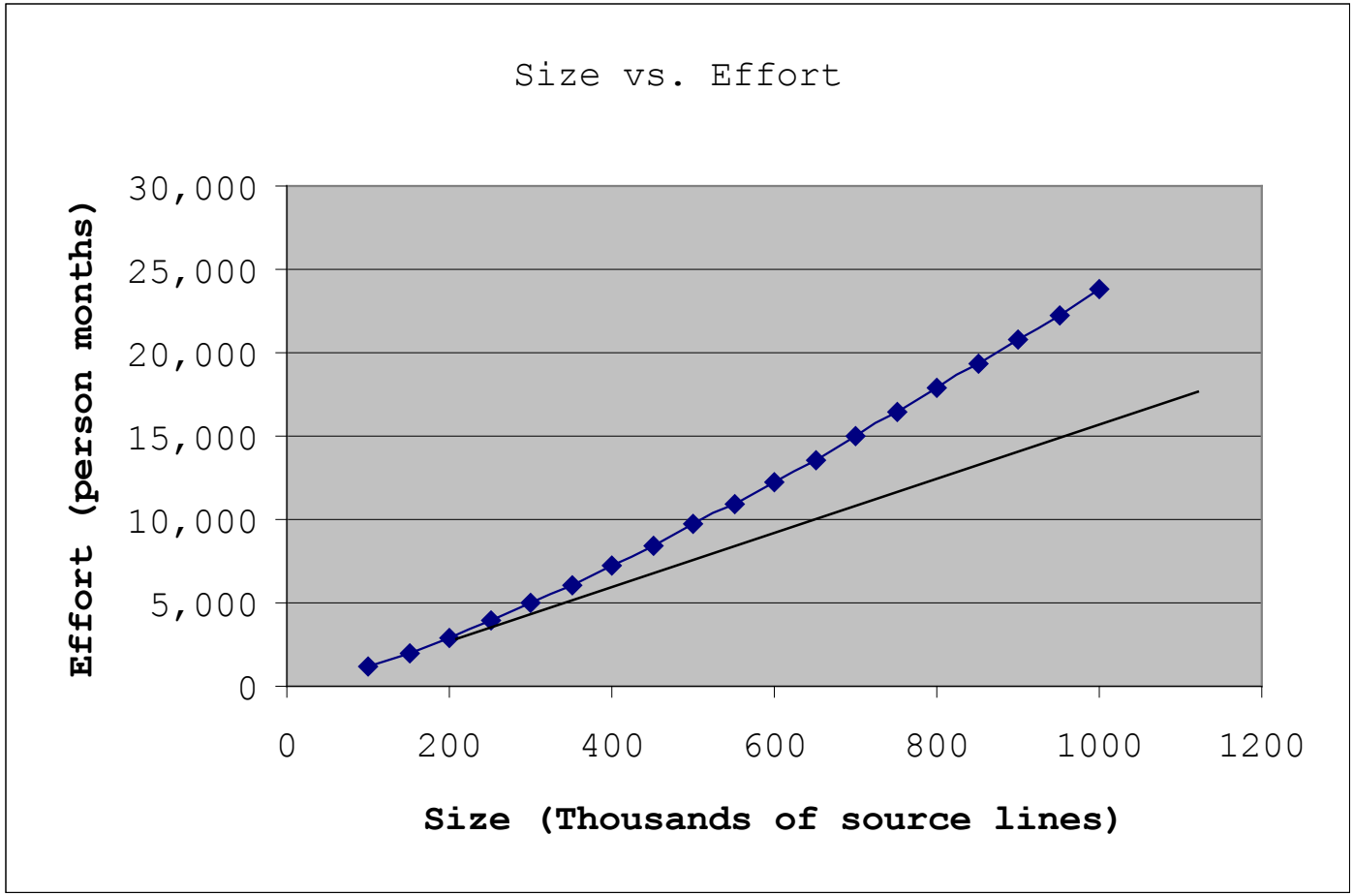


FIGURE 1.15. The number of errors detected between system integration testing and full operational capability increases rapidly with system size.

Size vs. effort (linear scale)

KSLOC	Effort
100	1,194
150	2,023
200	2,941
250	3,930
300	4,982
350	6,087
400	7,241
450	8,439
500	9,678
550	10,955
600	12,266
650	13,612
700	14,988
750	16,395
800	17,829
850	19,291
900	20,780
950	22,293
1000	23,830



R&D Staffing profile

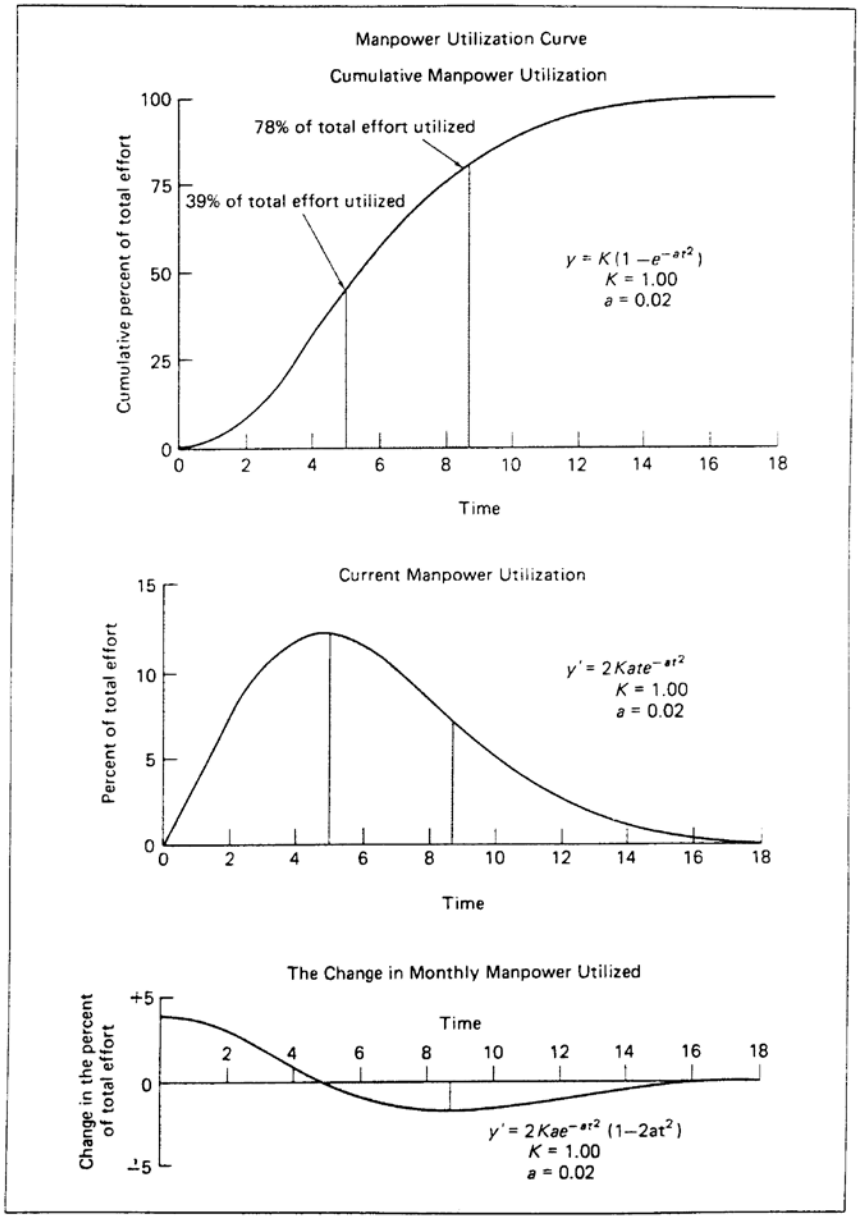
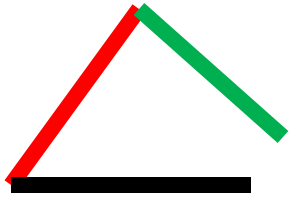


FIGURE 3.3. The Rayleigh curves chart cumulative manpower, current manpower, and change in the rate against time.

6. Traditional PM does not (need to) ask "Is this project possible?"

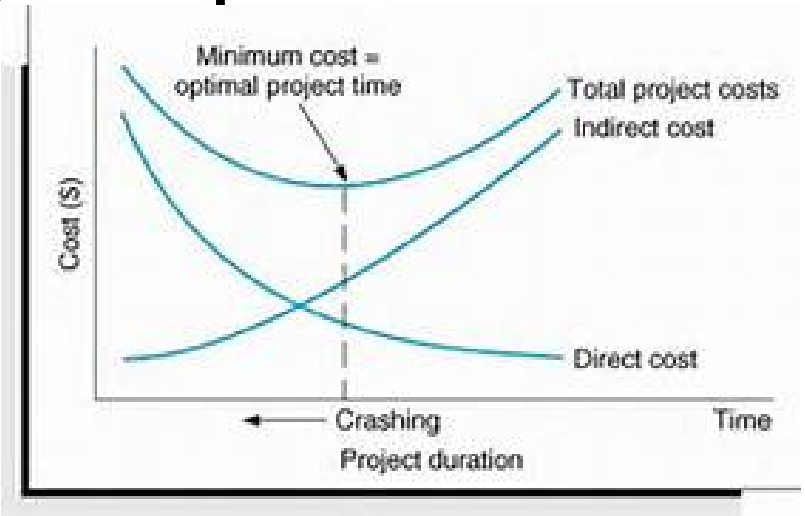
- We all know the project triangle: cost (effort), duration, and scope. But how do you know if the triangle closes? That is, if you select any two, how do you COMPUTE the third?



- For R&D there is a minimum project duration & effort! Schedules cannot be compressed without bound. There is an impossible region.
- That is, duration and effort are independent! In R&D it depends upon how one loads the effort over the duration.

6. Traditional PM does not ask "Is this project possible?" (cont.)

- It is worth noting that the shortest duration project has the least scope, highest cost & greatest number of delivered errors among alternatives plans. Always! It's a matter of arithmetic, not will.
- In R&D projects our PM job is to offer decision-makers alternatives, showing the impacts of trade-offs.



There is a minimum
(and everything is non-linear)

110% of 11.75 = 13

14.5/9 = 1.6

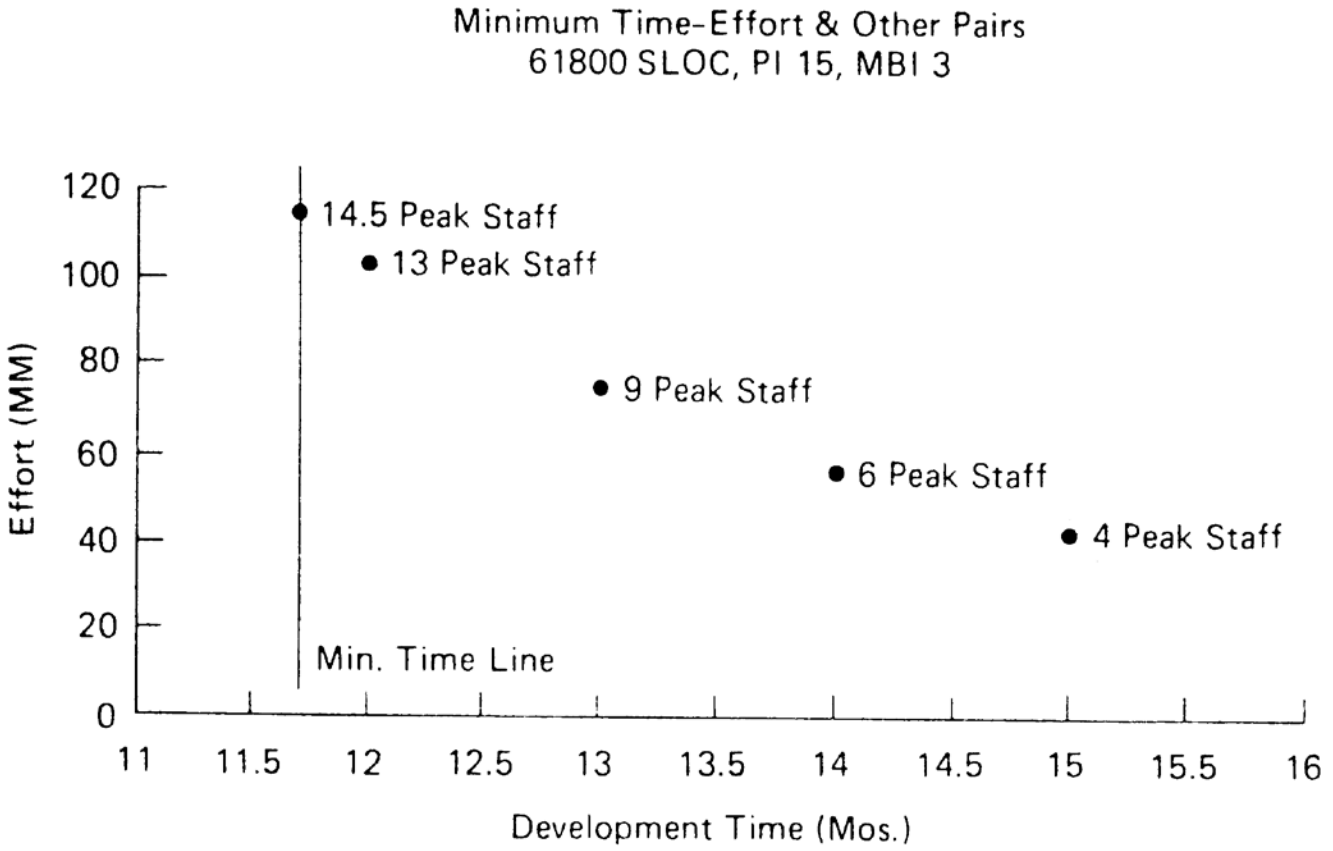


FIGURE 6.5. In this case, extending the planned development time from 11.7 months to 15 months would reduce the effort by a factor of more than two. If management can plan for about three additional months beyond the minimum time, it can accomplish the project for 63 percent less effort. Note: MBI = 3 at the minimum time because that is the only place MBI 3 is relevant.

7. Traditional PM does not address optimality wrt constraints

- **Is there – or has there ever been – a project of any type that was not constrained?**
- **What is the traditional PM advice on how to deal with constraints?**
  1. **Work around them**
  2. **Negotiate**
  3. **Compromise**
  4. **Work harder, schedule overtime**
  5. **Hope for the best**

Take the example of software product quality

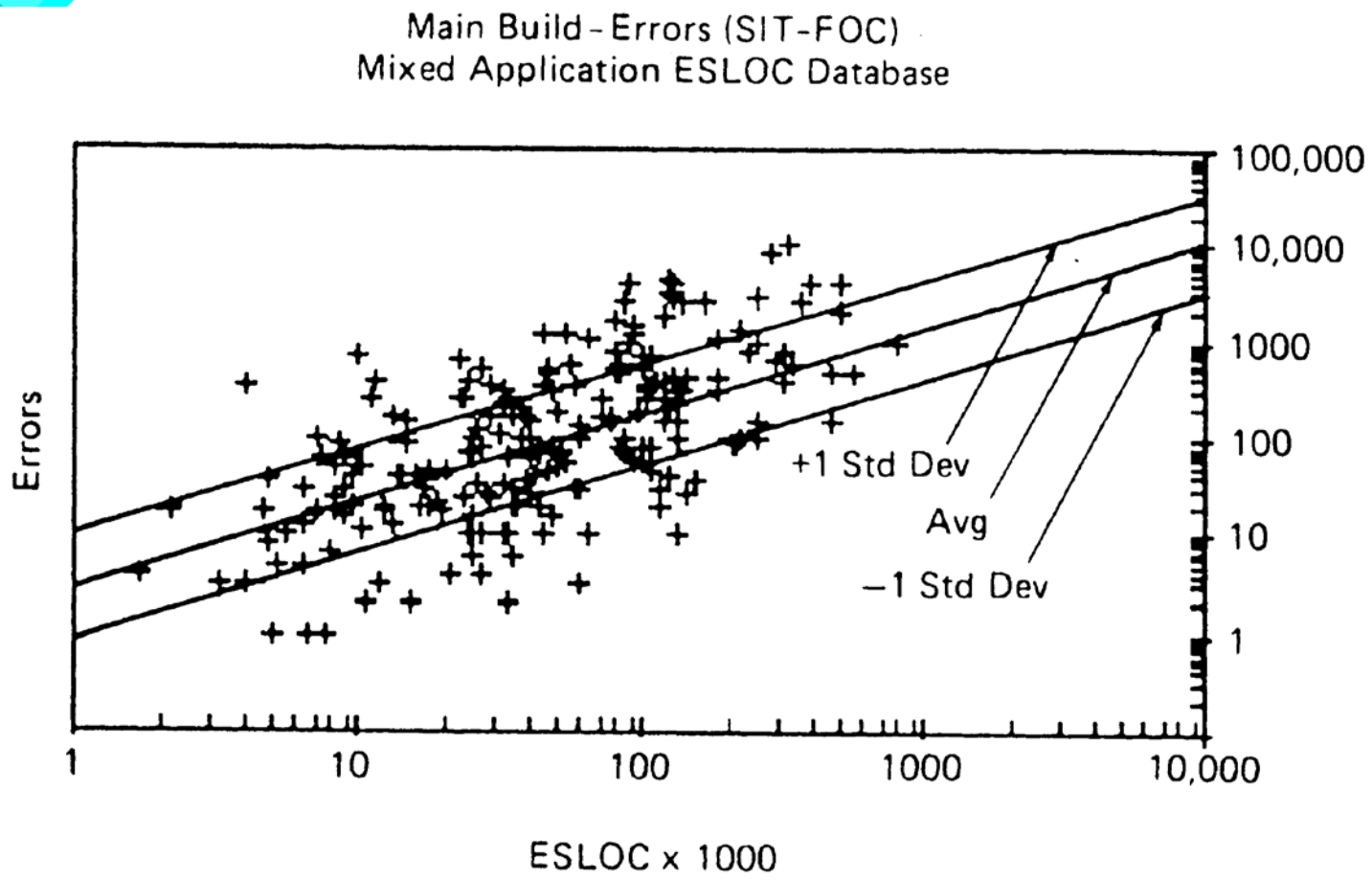


FIGURE 8.1. The crosses locate the number of project errors in relation to size in SLOC. At each size there is considerable variation in the number of errors.

Quality (cont.)

110% of 15.75 = 17.325

600/400 = 1.5

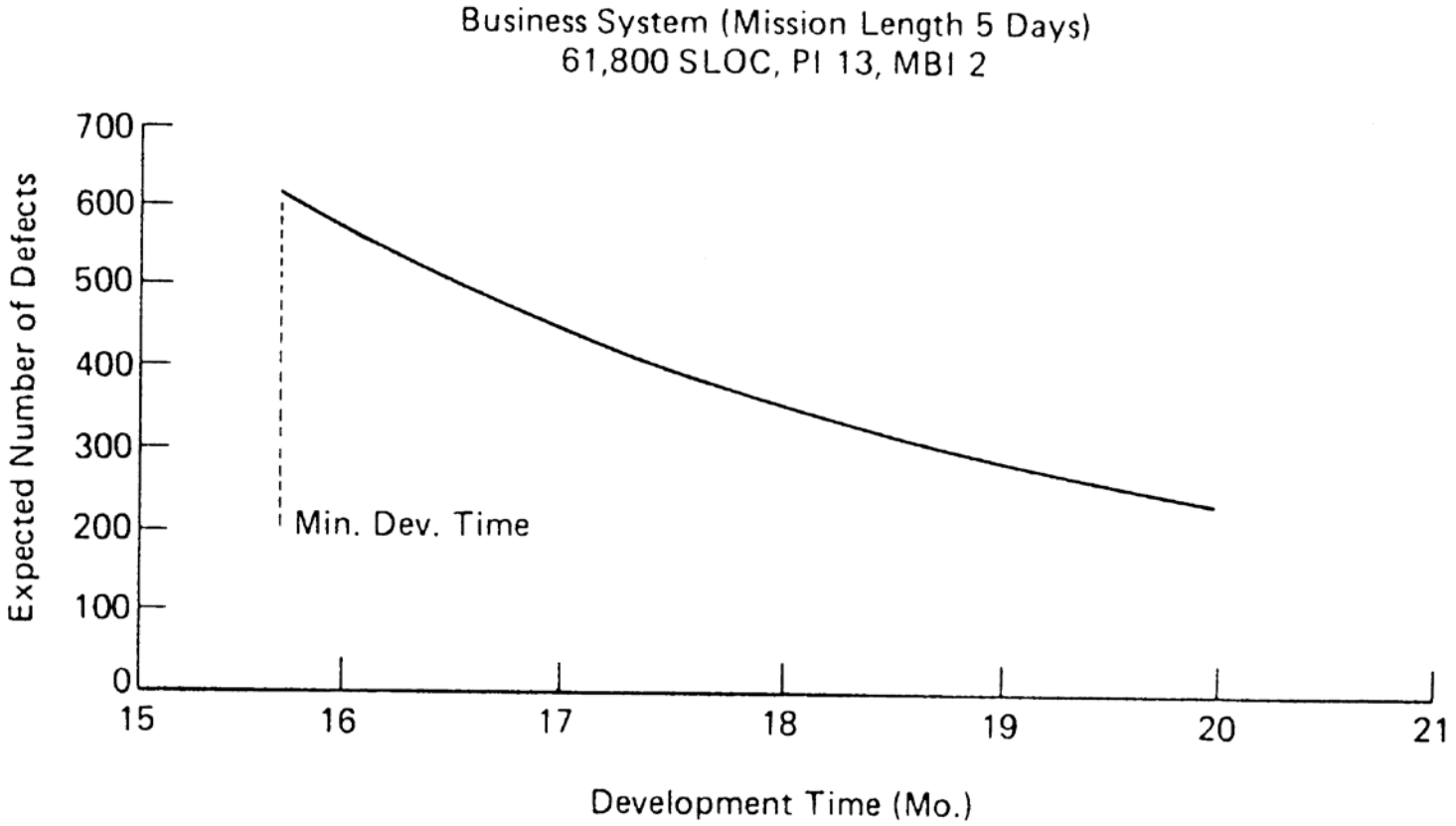


FIGURE 8.7. The expected number of defects is plotted against alternative planned development times. Extending the planned development time by 4.33 months—a 28-percent longer development period—reduces the number of defects by a factor of 2.7. This Cobol business system of 61,800 SLOC was calculated at a productivity index of 13, manpower buildup index of 2.

8. Traditional PM compresses schedules without limit

- **The usual (PMI) way of estimating duration is by taking the work-hours required and dividing by the number of people available to do the work.**
- **In traditional PM this is the link between effort and duration. One need only "pour" the effort into the dates available, governed by the sequences of tasks.**

R&D: Compression factor (Manpower build-up index)

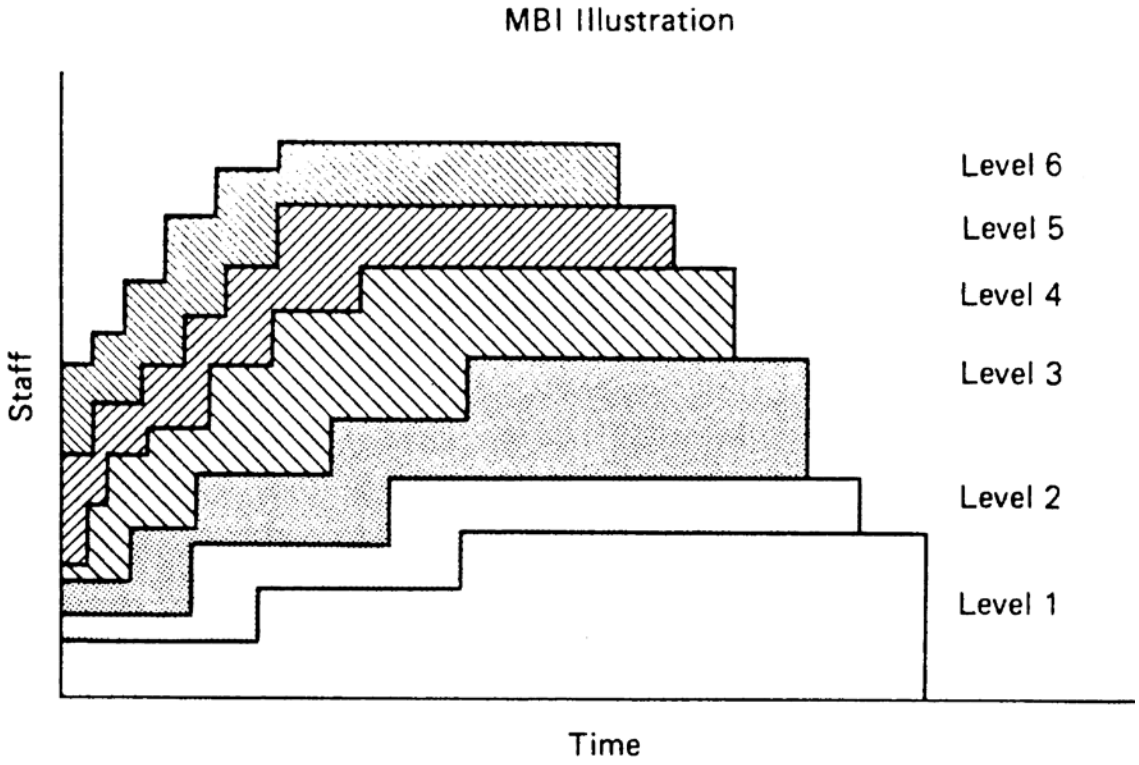


FIGURE 3.5. The Level 1 Manpower Buildup Index indicates that the buildup is slow and takes longer. As the index numbers increase, the buildup becomes steeper and more rapid.

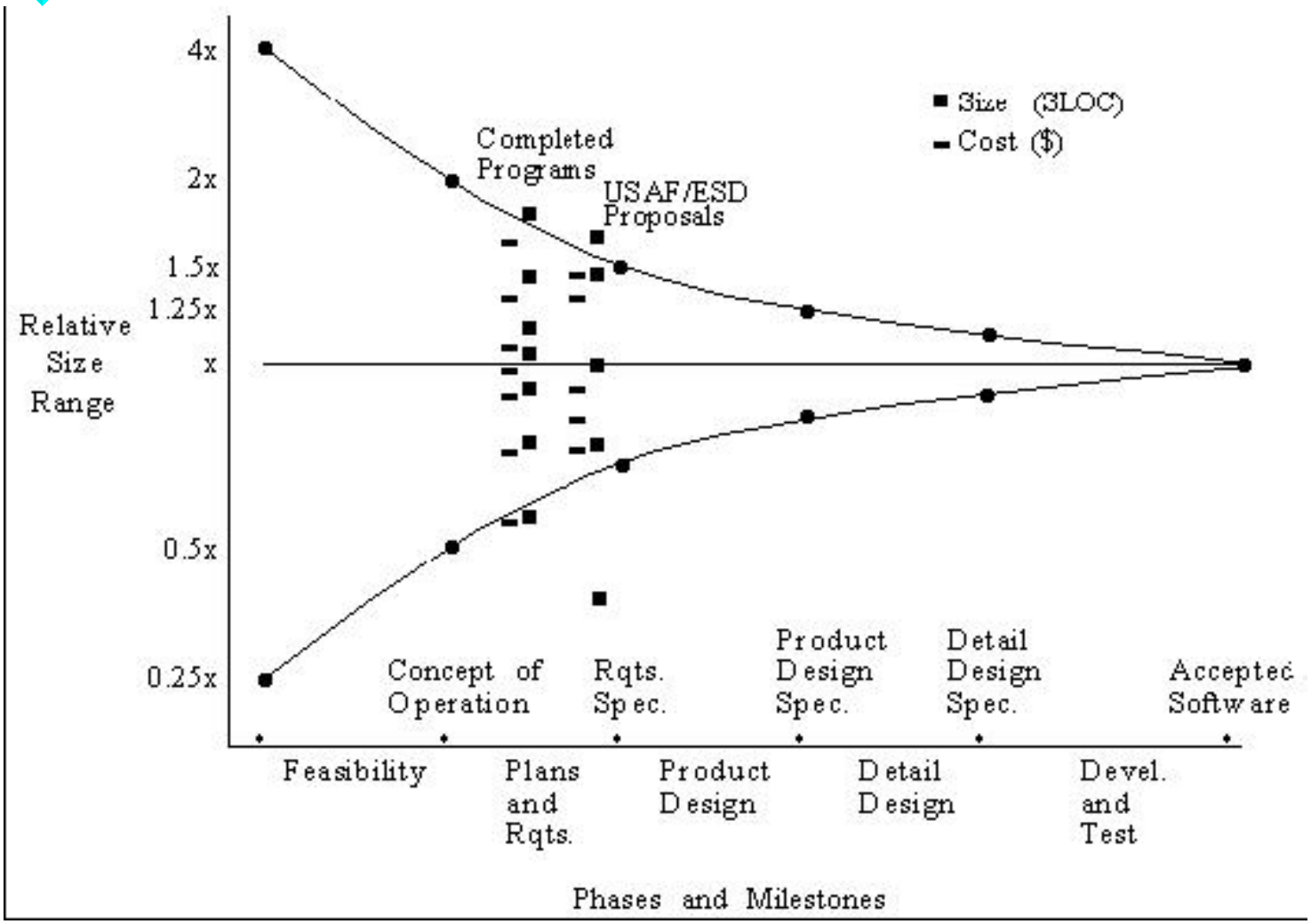
TABLE 3.2. The decision to speed up the staffing of a project has a small effect on the development time but a major effect on the effort.

MBI	Dev. Time (Months)	Effort (Manmonths)	Cost	PI
1	16	55	$458,800	11
2	14	80	666,700	11
3	13	120	1,000,000	11
4	12	180	1,500,000	11
5	11	235	1,958,000	11

9. Traditional PM asserts that the purpose of a project is to produce its promised result

- **What is the purpose of a project? What is the purpose of an organization?**
  - **To reduce equivocality = learn something**
    - Reduce uncertainty – need facts
    - Reduce ambiguity – need sensemaking
- **How many project plans show the reduction of equivocality? In fact, how is such reduction achieved? How much does it cost in resources (duration & effort) to reduce equivocality by *x*%?**

The cone of uncertainty



The original spiral: Address riskiest items first

10. Traditional PM is at odds with lots of new software engineering methods

- **Scrum bans/forbids project managers! <u>All</u> forms of traditional planning are banned – in favor of greater team accountability for its solution.**
- **Agile methods generally plan only for the current iteration, usually a matter of a week, sometimes a month. And "plan" takes on new meaning in this context!**
- **Personal software process and team software process are planning-centric. They are not very popular!**
- **Systems-of-systems (e.g., future defense systems) cannot use traditional project management because they are too inter-related, inter-dependent.**
- **Who is Microsoft's biggest competitor? Is it Google? (Neither plan very much!) No, it's free & open source software, none of which is planned in the traditional sense.**

11. There are so many simple questions we cannot answer about PM for R&D

a. If we could augment the project by an additional person, what role should he/she have? QA, deputy program manager, creator, … ?? How would we know which role(s) are the bottleneck(s)?
b. If we could augment the duration by, say, 10%, what would we spend it on? Additional scope, quality, architecture (flexibility & future hooks), … ??
c. We can speed up project by utilizing parallelism. How do we discover it? How do we trade-off with the added coordination effort/duration?
d. We all agree that preventing errors is less costly than finding them. Where is that in the project plan? In a WBS?

12. There is a special place in h#$@ for PM of organizational culture change

- **The basic assumption that we can be changed is due to a belief that we – and our organizations -- are machines (Theory X). That's a strong point for traditional PM.**
- **But what if we are not so much machines as we are, er, humans? Can our behavior, our culture, be estimated? Yes, you might say, within limits. Broad limits as we have seen. BUT, we are also interpreters. We (individually & collectively) react one way on one day and another way on another day.**
- **Better to use other tools for managing culture & technological (beyond the scope of this talk).**

So: What should R&D PMs do?

- **Use tools specifically designed to estimate R&D. Appreciate the non-linear relationships among scope, duration, effort (cost), and quality.**
- **Perform a risk analysis of the project. Include the problem space, solution space, and project/program space. Drive the choice of life cycle from it.**
- **Try to achieve parallelism in the work.**
- **Account for the social aspects of R&D. View the enterprise as information processing (i.e., communicating).**
- **Generally: Do NOT use milestones and effort burn. They usually have nothing to do with what is being achieved, except as symptoms … in the rearview mirror.**
- **Examine processes to see where they can be streamlined. This is best done as a whole, not one little segment at a time. Process improvement can speed up everything!**

Conclusion

- Project management for knowledge work is different than for traditional PM.
- We lack most everything to be an effective PM of knowledge work.
- Keep your eyes open for solutions, some of which are not new (e.g., statistics).

A few references

The diagrams come from *Measures for Excellence*, by L. Putnam & W. Myers. Prentice-Hall, 1997. Larry Putnam has a company, Quantitative Software Management (qsm.com), that licenses PM tools for software development that can be calibrated for R&D work more generally.

For guidance on how to select such tools, see "How to select software project macro-estimation tools," by Stan Rifkin in *IT Metrics Strategies*, vol. VI, no. 9, September 2000, pages 13-16. Available at Master-Systems.com

There is a (free) tool for estimating systems engineering projects: http://cosysmo.mit.edu/

The idea that organizations are machines is one of several ways in which they can be characterized. See *Images of Organization*, Updated Ed., by Gareth Morgan. Sage, 2006.

The problem of using traditional project management for enterprise transformation (i.e., culture change) is examined in "Raising questions: How long does it take, how much does it cost, and what will we have when we are done? What we do not know about enterprise transformation," by Stan Rifkin in *Journal of Enterprise Transformation* (2011), vol. 1, issue 1, pages 34-47. Available at Master-Systems.com