

Chapter 1

Supplementary Notes

Software Engineering As A Discipline

- Several commonly-acknowledged models
- All the models attempt to describe “how to get it done”
 - Linear
 - Iterative
 - Evolutionary
 - Parallel
- In real world, hybrids abound
 - A little of this, a little of that
 - What is the project? Who are the people? When is the deadline?
These (and more) are all factors in determining the right process model for the project, team, or company.

Some Process Tools

- Software Process Dashboard Initiative
 - <http://processdash.sourceforge.net/>
- Version One
 - <http://versionone.com/>
- Rally
 - <http://rallydev.com/>
- Target Process
 - <http://targetprocess.com/>
- Scrum for Team Process
 - <http://conchango.com/>

Fact/Fallacy Tidbit

- Fact 2
 - The best programmers outperform the worst by a factor of 28***
- Data
 - Decades-old studies led to the original performance ratio
 - Widely accepted, widely ignored!
 - Some interesting reading on the topic:
 - <http://www.devtopics.com/programmer-productivity-the-tenfinity-factor/>
 - <http://ibmsystemsmag.blogs.com/idevelop/2008/09/looking-for-way.html>
 - <http://www.joelonsoftware.com/articles/FieldGuidetoDevelopers.html>
 - <http://www.artima.com/weblogs/viewpost.jsp?thread=4738>
 - An excellent book on the topic, widely regarded as the best:
 - *Peopleware: Productive Projects and Teams* by Tom DeMarco and Timothy Lister.

*From Robert Glass, “Facts & Fallacies of Software Engineering”

Student Performance Data

- From Prof. Eisenstat at Yale, results of CS-323 course
- Courtesy of Joel Spolsky:
 - <http://www.jelonsoftware.com/articles/HighNotes.html>

Project	Avg Hrs	Min Hrs	Max Hrs	StDev Hrs
CMDLINE99	14.84	4.67	29.25	5.82
COMPRESS00	33.83	11.58	77	14.51
COMPRESS01	25.78	10	48	9.96
COMPRESS99	27.47	6.67	69.5	13.62
LEXHIST01	17.39	5.5	39.25	7.39
MAKE01	22.03	8.25	51.5	8.91
MAKE99	22.12	6.77	52.75	10.72
SHELL00	22.98	10	38.68	7.17
SHELL01	17.95	6	45	7.66
SHELL99	20.38	4.5	41.77	7.03
TAR00	12.39	4	69	10.57
TEX00	21.22	6	75	12.11
ALL PROJECTS	21.44	4	77	11.16

Improving Performance

- Process
 - Fit the process to the project and the team
 - Understanding-Planning-Developing-Testing-Documenting
 - Waterfall, Evolutionary, Agile, Hybrid, etc.: Select, know, use
- Tools
 - IDEs (Visual C++) and Frameworks (Rails)
 - Make/Build tools & scripts (Ant)
 - Automated testing tools (JUnit)
- Teams
 - Pair programming + solo programming
 - Code reviews & walk-throughs
 - Analyzing examples of quality code
- Workplace
 - Option for “Do-Not-Disturb”
 - Ergonomics overall
- An interesting discussion is here:
 - <http://www.nerdymusings.com/LPMArticle.asp?ID=11>