

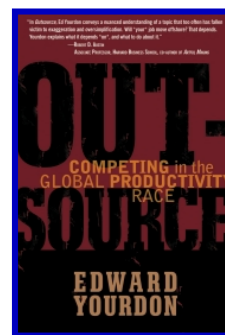
Software Litigation

Intro

- I'm not a lawyer
- Civil -> proponderance
- IT Project failures, not IP
- Both sides usually @ fault
- Both vendor, customer

Trends

- Litigation costs > coding costs
- Complex projects, demanding clients
- More outsourcing
- Lawyers with CS degrees
- 95% of cases settled before trial



History: Y2K

- Avoid litigation
- Have a process
- Document what you do

Observations

- Litigation-> 3-4 years later
- People die, email lives forever
- Vendor-client relationships are like teenage sex
- VPs quit, nerds live on
- Sometimes one party wants out
- Sometimes one party should get out

Contracts

- Contract details
- Specifications: OOA, SA, whatever...
- Acceptance test
- Warranties ("best efforts")
- Representations ("entire agreement")

Conclusions

- Trust your common sense
- Involve lawyers early
- Remember: lawsuits messy, expensive, nobody wins
- So incorporate external QA reviews
- ... and counselling, mediation

Metrics

-  Expect to see experts here
- Involve them early
- Incorporate into contract

Technology

- Usually a red herring
- But sometimes the straw that breaks camel's back

Peopeware

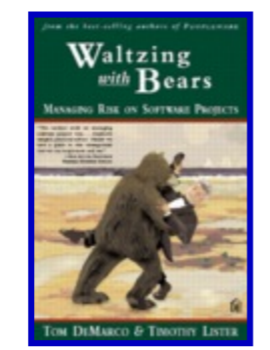
- Skills assessment
- Sufficient quantity
- Backfilling



- SEI-CMM
- ISO-9001

Processes

- Details of OO, UML, etc.
- Beware XP, RAD, "agile"
- Best practices: risk mgmt, etc.



Project Mgmt

- Timeliness of documentation
- Standards (PMBok)
- Who should have known? When?
- Breathalyzer test
- Death March projects

