

# *Building and Implementing a Quality Index*

*Jim Hancock*

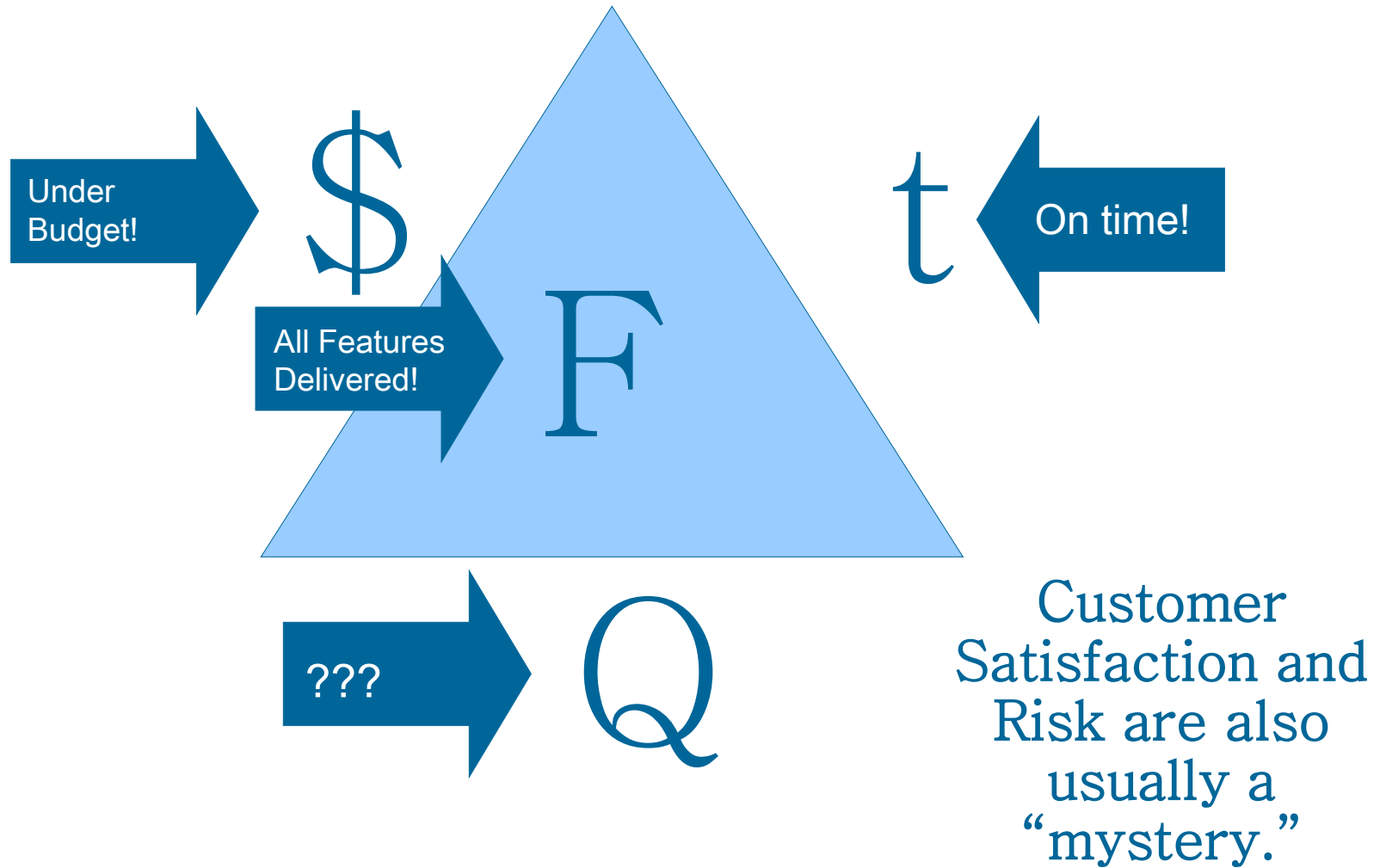
*Director of QA, IT and Customer Delivery*

*JimH@geospiza.com*

*July 14, 2010*



# The “Quality” Problem (PMP Constraints)



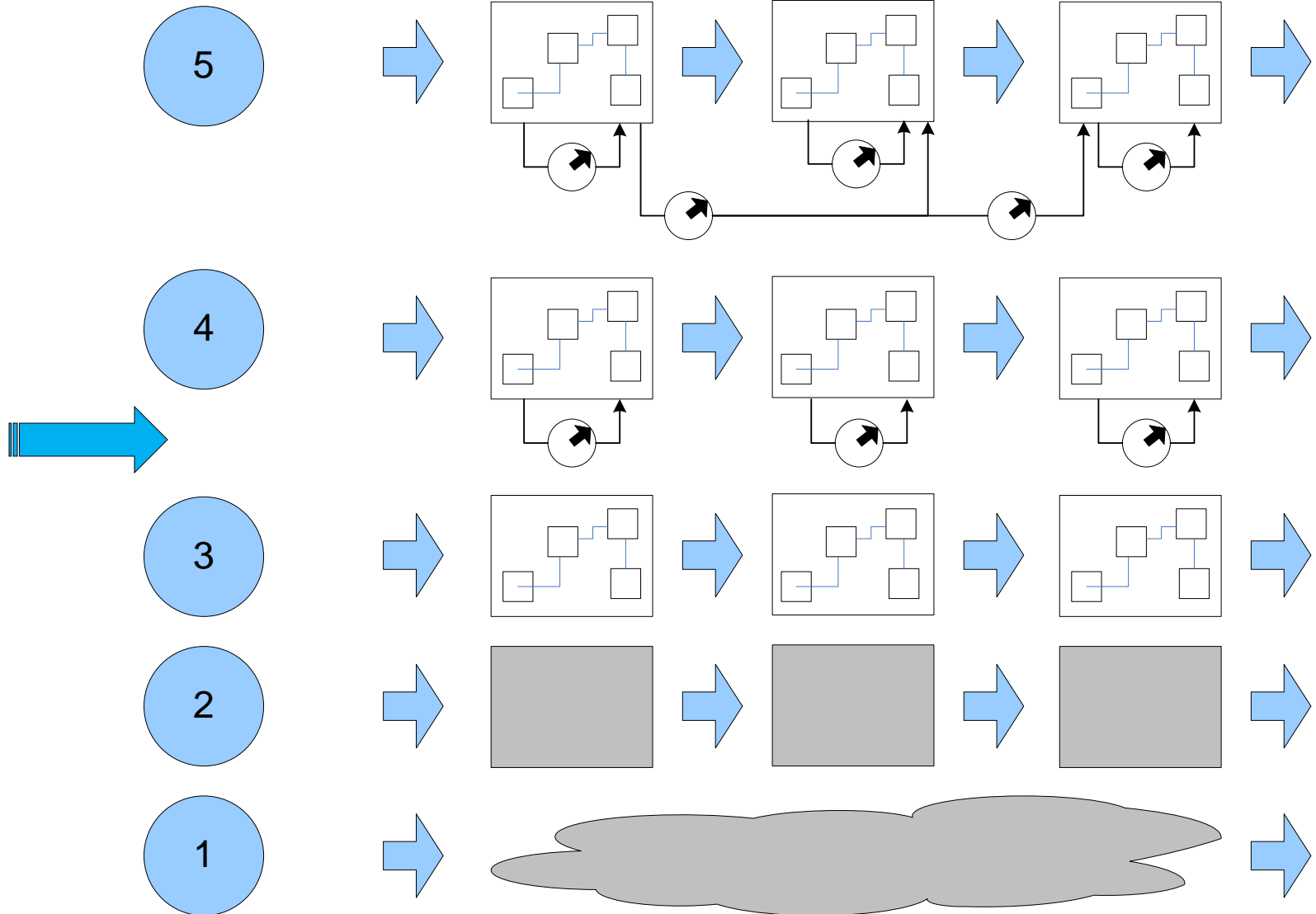
## The QI should be:

- ❖ Non-obtrusive, SDLC Agnostic
- ❖ Accurate, Repeatable, Objective
- ❖ Consistent, yet Flexible
- ❖ Holistic
- ❖ Inexpensive

Also remember -

What gets measured  
gets done!

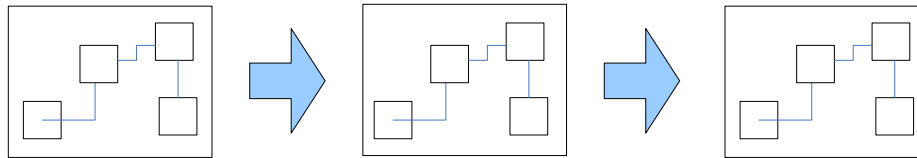
# CMM Levels



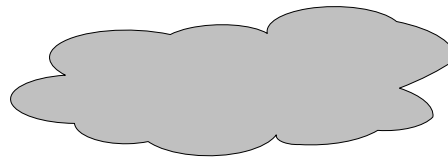
# Where to Begin?

There are 2 places to start:

- Convert existing documented process and metrics to a Quality Index:



- Use brainstorming and/or informal interviews to discover your undocumented processes:



# Steps to Create the QI

1. Determine Relevant Phases or Milestones
2. Brainstorm “QI Requirements”, such as:
  - ❖ Deliverables
  - ❖ Activities
  - ❖ Metrics
  - ❖ End-States
3. Assign Requirements to Phases
4. Determine Owners and Sign-Off
5. Assign Weights to Requirements
6. Determine Scoring and Escalation

# The End Result (for One Milestone)!

#	Document Requirements	Requirement	Responsible (Does)	Accountable (Approves)	Wt.	Score
1	Y	System <b>Performance</b> requirements met.	QA	QA Lead	10	0
2	Y	System <b>Reliability, Stress and Scalability</b> requirements met.	QA	QA Lead	10	1
3	Y	All <b>automated regression tests</b> run on latest build.	QA	QA Lead	25	2
4		<b>Release Candidate Distributable</b> is available to test.	IT	QA Lead	7	3
5		<b>QA environments</b> have been available throughout release testing.	IT	QA Lead	3	4
6		Test <b>upgrades</b> on standard platform(s) including a new OS installation. Ensure potential production data impacts are tested.	QA	QA Lead	15	3
7	Y	All <b>Defects assigned to Release</b> (Priority A, B and C) have been tested.	DEV / QA	QA Lead	10	4
8	Y	All planned <b>manual test cases</b> have been executed against release candidate (bugs may exist). This can be a small subset of the total.	QA	QA Lead	5	2
9	Y	All <b>Severity 1 Defects</b> retested (all 3.x releases).	QA	QA Lead	15	4
10	Y	All <b>Severity 2 Defects</b> retested (all 3.x releases).	QA	QA Lead	10	4
11	Y	<b>Change Log</b> document created and updated.	DOC	QA Lead /	6	4
12	Y	<b>Known issues</b> document created and updated.	DOC	QA Lead /	6	4
13	Y	<b>No new Severity 1 or 2 Defects</b> discovered in the past 3 working days.	DEV / QA	QA Lead	10	3
14	Y	No <b>Severity 1 Defects</b> (includes Priority D and E). Includes everything EXCEPT Closed.	DEV / QA	QA Lead	20	4
15	Y	No <b>Severity 2 Defects</b> (includes Priority D and E). Includes everything EXCEPT Closed.	DEV / QA	QA Lead	10	2
16	Y	No <b>Severity 3 Defects</b> (includes Priority D and E). Includes everything EXCEPT Closed.	DEV / QA	QA Lead	5	1
17	Y	No <b>Severity 4 Defects</b> (includes Priority D and E). Includes everything EXCEPT Closed.	DEV / QA	QA Lead	3	2
18		Test deployment of Production Release Candidate ...incl. component installs, database scripts, etc. (plus rollback). [execute <b>Mock Deploy</b> on Staging]	Dep Lead	QA	20	4
19	Y	All <b>new Release Candidates (i.e. last minute defect fixes)</b> are reviewed and approved by Dev, QA and PDM.	IT	Dev / QA Lead / PDM	10	4
<b>R2 - Release Candidate</b>					<b>200</b>	<b>75%</b>

# Results: In Addition to Higher Quality

## Increase Value:

- ❖ Customer Satisfaction
- ❖ Internal Efficiency
- ❖ Corporate Brand Value
- ❖ Regulatory Compliance

## Reduce Risk:

- ❖ Customer Support Impact
- ❖ Potential Legal Exposure

# Bonus Results: Changes Behavior

- ❖ Allows deployment decisions based on known risk and quality
- ❖ Forces consistent processes, roles, etc.
- ❖ Eliminates last minute crisis/code changes
- ❖ Stops creation of “mystery products” (i.e. lack of requirements, design, etc.)
- ❖ Stops “gatekeeper mentality” (QA or others)
- ❖ Makes us pro-active versus reactive
- ❖ Places the focus on the QI, not the individual  
...improves team dynamics
- ❖ When high score = failure, allows Incremental Improvement