Teaching *New* Dogs *Old* Tricks

Leveraging PM-related Intellectual Capital

November 19, 2008

Dr. Carl A. Singer
Sharing Expertise

A Process Driven Approach

For IT Projects / Management

Carl A. Singer, PMP

THE BANK OF NEW YORK MELLON
Presentation Outline

• Purpose
• Background
• History
• Solution approaches

1. The METHOD  
   What YOU can do
2. PMOs, EPMOs, PM-COEIs & friends

+ some digressions and diversions
Purpose

…. Of the concepts

• To do things better
  – Shared Intellectual Capital (IC)
  – Process related improvements

Specific to the IT domain
Purpose

…. Of this presentation

Carl Singer – PM Process – Advocate

• To engage (aka, “tickle”) your curiosity

• To help you get motivated / started towards building your own solutions

This isn’t Shrinkwrap or Turn Key
Carl’s 1st theorem of project teams

All of us together are smarter than any of us alone

• THIS IS BASIS FOR MAKING PROGRESS

• I hope to share some experience and opinion – feel free to do the same
I think this is the most extraordinary collection of talent, of human knowledge, that has ever been gathered together at the White House, with the possible exception of when Thomas Jefferson dined alone.

- John F. Kennedy Remarks at a Dinner Honoring Nobel Prize Winners of the Western Hemisphere. April 29th, 1962
Background

Background of the Problem / Opportunity

• Lack of Skills
• Lack of appropriate Skills
• Lack of skills that fit a specific situation

• The inability to do the right job right
SKILL

the ability to use one's knowledge effectively and readily in execution or performance

a learned power of doing something competently: a developed aptitude or ability

Merriam-Webster on-line Collegiate Dictionary
“Skills imply the capacity for action toward some application.

As such, much of IC (Intellectual Capital) is prescriptive and attempts to supplement individual skills to achieve a solution or solve a problem.*”

Characteristics of...
Desired Characteristics of ... The SKILLED Worker

1. Works more quickly
2. Produces better results

Smells Smoke
History

History of Solution Approaches

- **IC-related Tools**
- Guilds / Trade Schools / Apprenticeships
- OJT – On the Job Training
- Mentoring
- Teaming
“Recent” History

History of Solution Approaches

• **ISDOS** — Information System Design & Optimization System
• **Boot Camp / Stay in Step**
• Requirements Guidelines & Templates

**The Method**

**PMOs**
ISDOS 1965

- Information System Design & Optimization System -- Case Tech / University of Michigan
- Dan Teichroew
- **IF** Fully Describe the System and SDLC Process – **THEN** Generate Systems Design (SODA)
- PSL / PSA – 1965 → Requirements Statement Language / Requirements Statement Analyzer
- Real Business Problems: Transshipment Problem
Problem:
Cutting Costs in Production and Shipping at The Salt Co.

The Salt Company had:
- 4 salt mines ("street salt" for winter roads)
  - 12 month operation
  - 3 shifts
  - Different costs and capacities at each mine
- 84 warehouses
  - Seasonal demand, municipal contracts, etc.
  - Multiple shipping modes (barge, rail, truck)
  - Different availabilities, costs, and capacities
4 Mines x 12 Months = 48 "sources"

84 Warehouses x 12 Months = 1008 "sinks"

- Constraints include -- you can't ship backwards in time

Yields a 48 (4 x 12) x 1008 (84 x 12) shipping tableau

Three key accomplishments

- We "tackled" the data -- gathered & understood the numbers
- We solved the mathematics, in theory, then in practice (operations research issues)
- We solved the computing issues - "Interactive computing"
Results:

- Previous year's shipping costs $26 Million
- "Optimized" Shipping Cost $14 Million
  - We saved the client
  $12 Million = 46% cost reduction
History

History of Solution Systems

• ISDOS — Information System Design & Optimization System
• Boot Camp / Stay in Step
• Requirements Guidelines & Templates

The Method
PMOs
BOOT CAMP 1990

• Bellcore
  – Growth - large influx of new employees
  – Specialized SDLC model
  – CMM – level 5
  – 4 week intense, off-site class
    • Technical Content, Leadership Message & Motivation
  – Focus on our SDLC

  – “Stay in step” program for “old dogs”
History

History of Solution Systems

• ISDOS — Information System Design & Optimization System
• Boot Camp / Stay in Step
• Requirements Guidelines & Templates

The Method

PMOs
Requirements Guidelines & Templates

• Bellcore SR-NWT-002159, December 1992
  — A Requirements Tutorial – Quality Systems & Software Requirements
• Elements of a Single Requirement
• Requirements Tools
• Structure – labeling / hierarchy / links
• Attributes
• Requirements Categories: Explicit / Conditional / Phased
• Change Management
Faster, Cheaper, Better

- On Time
- Within Budget
- Achieving Quality Goals

Quality?
- Conformance to Requirements
Better Faster Cheaper

- Project Integration Management
- Project Scope Management
- Project Time Management
- Project Cost Management
- Project Quality Management
- Project Human Resource Management
- Project Communications Management
- Project Risk Management
- Project Procurement Management
PM is a “soft science”

Quantum Mechanics
- Own Language
- Own Views
- Observations
- Theorems & Proofs

Teach / Learn T & Ps

Project Management
- Own Language
- Own Views
- Observations
- RoTs & HW

Share Expertise
Carl’s 1st Lament

- The laws of physics are *not* determined by a management committee and are *not* subject to annual budget review

The “laws” of Project Management *are* …

*Projects exist in a business context*
Sharing Expertise

• Book Learning / Formal Instruction
• Leadership / Mentoring
• OJT – On the Job Training / Simulation
• Knowledge Management / Intellectual Capital Management
Knowledge Management

*To be useful*

- Relevant knowledge readily available
- Tailored to the task at hand
- Tuned to the user’s training & experience
- Enhanced by selectable levels of detail
- Context Specific / Context Aware
- Enabled with supporting templates & guidelines
- Supported by effective navigation tools
PM - Intellectual Capital
Definitions

**Principle:** A fundamental truth, rule of conduct or law upon which others is based.

**Guideline:** A guideline is is: (1) actionable (i.e. it recommends, or recommends against, an action to be taken) and (2) authorized by consensus. Guidelines are not set in stone and should be treated with common sense and the occasional exception.

**Policy:** Governing principles that apply to the management of the business. A policy is similar to a guideline, only more official and less likely to have exceptions. One should not generally edit policy without seeking consensus first.

**Rule:** An authoritative regulation, law or established practice by which conduct, methods and procedures are controlled in alignment with stated policies and/or principles.

**Process:** A central and organized way of doing things, generally following certain policies or guidelines (e.g. the "deletion policy" tells us how the "deletion process" works).

**Doctrine:** the body of principles in a branch of knowledge or system of belief

**Methodology:** A disciplined accumulation of the above.
Sharing Smarts at IBM
The Method

• 35,000 Consultants – lots & lots of projects
• Lots of development types
  – Web based
  – Legacy + Web front end
  – Rapid Custom Development ….
• Lots of domains
  – Insurance
  – Banking
  – Manufacturing – Supply Chain ….
COMMONALITY

✓ Planning
✓ Requirements
✓ Design
✓ Testing
✓ Staffing
COMMONALITY

One size does NOT fit ALL

BUT .... 100 projects do not require 100 clean sheets of paper

Some things CAN be shared

For example,

100 projects may be serviced by perhaps 5 different approaches to gathering requirements  5 << 100
The Method 50,000 Foot View

• Engagement models
  – Piece-parts – Intellectual Capital for reuse
• Uploading & storing Intellectual Capital
• Locating & downloading Intellectual Capital
• Build Engagement Model Instances (planning)
  – MAW (Methodology Adoption Workshop)
• Project Execution support
Project Lifecycle

PLAN the work

WORK the plan

Project Preparation
- Defining
- Planning

Project Execution
- Resourcing
- Project Control
- Project Tracking
Engagement Model Lifecycle

**Authoring**
- Developing
- Publishing

**Project Preparation**
- Defining
- Planning

**Experience Harvesting**
- Intellectual Capital Management

**Project Execution**
- Resourcing
- Project Control
- Project Tracking

**Organizational / Project Context**
HOW TO

Action Steps
- Identify domains
- Find experts / leaders
- Identify best practices
- Capture the goodness
- Break components down
  - Work Products
  - Work Breakdown Structure
- Build Supporting Artifacts
- Continuous updating

Project
- Realistic Boundaries
- Approach Framework
- Plans / Resources

Scaffolding
- Information Structure
- Toolset
- Support Structure
Two Approaches to PM IC

• Task-oriented approach

• Output-oriented approach
Breaking components down

Work Breakdown Structure

• Project
  – Phase
  • Activity
    – Task
      » Subtask

Work Product(s)

Inputs / Prerequisites

Output

Work Product(s)
OUTPUT

Work Products

Work Breakdown Structure

• Project
  – Phase
  • Activity
    – Task
    » Subtask
Work Product Description (WPD)

Unique ID: APP 1234

Version 1.1, May 2007

1 Description

2 Purpose

2.1 The primary purpose of the XXX is to provide

2.2 Reasons for Not Needing This Work Product

3 Notation

4 Example

5 Development Approach

6 Validation and Verification

7 Advice and Guidance

8 References

9 Estimating Considerations

10 Revision History

<table>
<thead>
<tr>
<th>Date of this release:</th>
<th>Date of next revision:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revision Number</td>
<td>Revision Date</td>
</tr>
<tr>
<td>Version 1.0</td>
<td>01/01/2007</td>
</tr>
<tr>
<td>Version 1.1</td>
<td>05/03/2007</td>
</tr>
</tbody>
</table>
Work Product

Work Product

TEMPLATE

Work Product

Instance

Technique

Papers

Guidelines
Leveraging the Method
The Software Maintenance Example

• Why / How / What — *drinking from a fire hose*
• Benefits
Leveraging the Method
The Software Maintenance Example

• Many Software Maintenance Engagements
• Some real experts, but much uncertainty
• I located and gathered the experts
  – Scotland, England, Australia & USA

See Leveraging a Worldwide Project Team

• Key was reuse – How different is SW Maintenance from Software Development?
Custom Application Development is generally 'clean sheet'.

Maintenance and Enhancement start off of a baseline.

- No Impact on Business, Organization or Architecture
- No New Requirements
- Temporary Only
- Follow with Enhancement or Corrective Maintenance

Like an AD Release

Major Enhancement

Perfective Maintenance

Emergency Fix

36

27

92

Custom AD

See Enhancement
Four AMO Engagement Models

Major Enhancement
(max ~92 work products)

Same as Custom AD Multiple Release

Enhancement
(max ~36 work products)

No impact on business, organisation and architecture domains

Corrective Maintenance
(max ~27 work products)

No new requirements

Emergency Fix
(max ~6 work products)

Leads to Corrective Maintenance or Enhancement
Some features

• New engagement family
• Consistent with existing work product based architecture
• New work products identified:
  • Baseline Analysis
  • Impact Analysis
  • Impact Analysis (Emergency Fix)
  • Emergency Fix Design
  • Problem Resolution Report
• New supporting technique papers identified:
  • Working with Use Cases in an Application Development / Maintenance Environment
  • Tailoring Engagement Models in an Application Development / Maintenance Environment
  • Integration Testing Planning
  • Construction and Testing
  • Progression from Emergency Fix to Enhancements or Corrective Maintenance

*Only 5 WPs to maintain, to learn....*
Leveraging the Method
The Software Maintenance Example

• Benefits *Opened up new line of business*
  – *Based on successful artifacts ➔ less uncertainty*
  – *Little new building*
  – *Relatively short time to market*
  – *Low cost to develop this LOB*
  – *Less to build and less to maintain*
  – *Minimal retraining – great knowledge transfer*
Implementing Process

• How to leverage the Engagement Models
The Method Adoption Workshop (MAW)

- The MAW is the keystone of method deployment, and requires preparation, education, leadership and follow-on mentoring
- The MAW is undertaken during Solution Design to position the Proposal and SOW utilizing the Method
- MAWs are undertaken at major engagement checkpoints to accomplish a re-orientation

Objectives
- Understand the project
- Tailor the method to project estimation, selecting techniques, tasks and tools
- Define team organization and roles
- Assess risks
- Assess team skills

Results
- Tailored Method
- Project Plan
- Risk Assessment
- Resource Plan for the Team
- Log of method exclusions and extensions
- Identified project work products and deliverables
The MAW is the primary deployment activity to the field practitioners

- A Method Adoption Workshop (MAW) is used to bring the project and the method together
- A MAW is usually facilitated by a trained Methods Exponent
- The length and content of a MAW are adapted to the needs of the project
  - They are discussed and agreed beforehand.
  - (Typical length 2-4 days, it may be spread over >1 workshop)
- A MAW is
  - Part consulting assignment,
  - Part education, and
  - Part project workshop
- A MAW is NOT a standard course
During tailoring the general engagement model is fitted to the needs of an individual project.

- No Engagement Model can fit everything
- A specific, ready-to-use model can only be applied to very specific uses
- Therefore we use a general engagement model which is tailored to the needs of the individual project

Fitting a pre-cut suit is much cheaper than creating a suit from scratch.

Pre-cut suit on the tailor's shelves.  The tailor fits the suit to your needs.  Your perfect suit.
Individual projects tailor the Engagement Models to fit their specific engagement

- The different Engagement Models are starting points for tailoring the method for a specific engagement
- Choose the closest Engagement Model and tailor from there
- Some engagements may also require components from *other* Engagement Models
Knowledge Management

• Relevant knowledge readily available
• Tailored to the task at hand
• Tuned to the user’s training & experience
• Context Specific / Context Aware
• Enhanced by selectable levels of detail
• Enabled with supporting templates & guidelines
• Supported by effective navigation tools
Navigation Tools

- Secure
- Easy to use
- Provide selection assistance and caveats
- Provide links to related material
- Assemble good starting point
What Can You Do In This Domain?

- Spend lots and lots of money
- Study your projects – identify useful artifacts
- Are they different because they need to be? Or did they just wind up that way?
- Choose some “best of breed” and decide where they may or may not apply
- Create an orderly “filing” system

If all you do in your first pass is find redundancies – you’re already a winner!
What You CANNOT Do?

• Buy a turnkey solution
• There’s no “EASY” button, This requires heavy lifting
• Make progress overnight
• Keep managing projects as an AD HOC activity
• Insanity is defined as doing the same thing over and over again and expecting different results

Albert Einstein
Building an Effective Project Management Office
What’s a PMO?
(multiple definitions)

1. a line organization that manages a single (complex) project

2. a line organization that manages or monitors a portfolio of projects and their resources

3. a staff organization that builds and maintains processes, standards and other project-related intellectual capital used in planning and executing all projects, yet directly manages no projects.

4. = 2 + 3
PMO

• **Managing Project Operations** – Managing the normal planning, execution and support of (one or many) ongoing projects throughout the entire project lifecycle. These functions will include:

• **Project Planning**

• **Project Execution** – managing the day-to-day project lifecycle from some defined initiation point through to a defined completion or hand-off point.

• **Single Project Selective Support** – Planned participation in various critical project support functions.
PMO

- **Multiple / cross-project support** – (Also called “Project Portfolio Management.” or “Program Management”)
- Since there may be multiple concurrent projects, there may be a need to perform functions that span or involve trade-offs among multiple projects or a portfolio or projects. Here is a typical list of such functions.
  - **Resource allocation** – allocating resources among the suite of projects.
  - **Status tracking** – tracking status of all active projects. This may include building a Status Reporting Framework.
  - **Parachuting** – (unplanned) providing emergency resources or special.
What’s a PMO?
(multiple definitions)

1. a line organization that manages a single (complex) project

2. a line organization that manages or monitors a portfolio of projects and their resources

3. a *staff* organization that builds and maintains processes, standards and other project-related intellectual capital used in planning and executing all projects, yet directly manages no projects.

\[ 3 = \text{PM-COE} \]
Manage Portfolio of Programs:
Form Projects, Do Projects, Deploy

Program Management

Project Management

Program Management

Form Projects
Select & Prioritize & Schedule
Share Resources

Deployment
Select & Stage On-going Support

Project D
Project G
Project H
Project E
Project C
Project B
Project A

Portfolio Management
Why have a PM-COE?

Motivation: The expected results from having a Project Management Center of Excellence are two-fold:

1. There’s the general improvement of project performance and
2. There’s the specific avoidance of the disastrous project, one where cost, timeliness, quality and / or customer satisfaction are not achieved.
Do’s and Don’t’s

• Do not let your PM-COE get isolated from the day-to-day issues that impact your project managers – encourage candid communications and cross-pollination.
• Do not let your PM-COE get so caught up in the day-to-day turmoil that it loses its vision and becomes simply a Project resource pool. (Measure how much time PM-COE actors are spending (fighting fires?) on specific projects versus other activities.)
• Integrate independent Project Quality Assurance (PQA) into all project plans and all projects.
• Reach out – do not be a passive “service window” but a proactive agent.
• Make sure that your PM-COE attacks issues, not people.
• Do not measure your PM-COE based on utilization or other operational metrics – this will lead to bad choices by all. What gets measured gets managed. What gets mismeasured gets mismanaged.
• Constantly seek feedback, both good and bad. Maintain avenues of communication with all project-related organizations.
• Don’t be a free resource – The savvy project managers will suck the marrow out of any free resource.
• Rotate a few experienced Project Managers in and out of the PM-COE every few years – the PM-COE should not become a sheltered workshop for tired project managers.
• Avoid NIH (Not Invented Here) at all costs. Together we are all smarter than any of us alone. The “field” has insights and solutions that are waiting to be harvested.
PMO-like things to do

1. Develop, facilitate and provide shared **intellectual capital** to support projects.
2. Build **peer support** network for informal, ad hoc, peer-to-peer communication.
3. Provide generic **project lifecycle reference guides** and **how-to’s**.
4. Provide related **training and mentoring**.
5. Provide support for **RFP** response **Estimation & Risk**
6. Provide **project initiation support** – project kick-off **support MAW**
7. Provide **project execution support** to include tools

*Greatest opportunity / risk when project is leaving the starting gate*
PMO-like things to do

8. Provide the quality assurance / quality audit function. (This is not testing!)

9. Oversee touch points, gates and other project decision points.

10. Build / install project “dashboard” facility to monitor project performance against plan.

11. Provide Program / Portfolio management capabilities

12. Provide archive / repository for project artifacts (for completed Projects)

13. In conjunction with Technology Services provide on-request technical support. (conduit)

14. Provide project deployment support – on planned and / or emergency basis. (conduit)
Carl's Top Ten List

You know the project is in trouble when:

10. You come in Monday morning and find several wastebaskets filled with empty Pizza Boxes (or little packets of Soy Sauce are everywhere)

9. You see 4 or more people in a cubicle huddled around a workstation and they're not singing "Happy Birthday"

8. NO one can tell you who's intellectually in charge --- Or there is no one who is in charge, intellectually.

7. Coding is ahead of design, Design is ahead of specs, Testing keeps muttering about requirements

6. The customer is distancing himself / herself from the project

5. Everyone is in catch-up mode.

4. You cannot map function to requirements

3. The Tuesday Wall Street Journal keeps disappearing.

2. There is one indispensable person (maybe two) who seems to hold the key to all knowledge and goodness -- and you haven't fired him/her.

1. David Letterman has asked you to audition for "stupid pet tricks"
Seven Characteristics of Highly Successful Projects

1. A positive relationship with an active, intelligent client
2. Strong project management
3. Clear requirements, well managed
4. Ruthless change management
5. Pervasive process focus
6. Effective controls and communication
7. Technical leadership and excellence

The critical next step, an honest self analysis of projects. An analysis based on these seven criteria is something that a PM-COE should consider as part of its continuing process improvement.
1 - A positive relationship with an active, intelligent client

- Stay in your sweet spot
- Win-win contract & realistic project plan
- Effective escalation procedures
- Build / maintain positive relationship
- Seek clients with a history of success
- Client decisionmakers who are actively engaged
2 - Strong Project Management
(roles)

• Project Manager
• Technical Manager
• Schedule Manager
• Resource Manager
• Contract Manager
• Configuration Manager

• Change Manager
• Build Manager
• Data Manager
• Requirements Manager
• Documentation Manager
• Customer Relationship Manager
  (Stakeholder Relationship Manager)

Technical leadership should not have to make resource / business decisions
3 – Clear Requirements, Well Managed

• Clear, unambiguous, well-documented Requirements in a database
• Requirements review board
• If pre-existing requirements – review to assess their quality
4 – Ruthless Change Management

- Change Manager / Change Review Board
- Baseline
- **Scope creep kills projects!**
- Any / all changes via change control process
- Changes cost money
- **Churn thwarts project success**
- Not all changes reflected in billing
5 – Pervasive Process Focus

• All processes are well documented & in place
• Process steps produce auditable outputs
• No process shortcuts
• Accurate measurement is key
• Do not abandon process in times of trouble
6 – Effective Controls & Communications

- Seize the initiative
- **Bad News does not get better with age**
- Build & maintain relationships & communications with all stakeholders
- Stay on message
- Maintain current project status
- Haste makes waste & leads to poor decisions
7 – Technical Leadership & Excellence

• You need a technical lead who is not the PM
• Technical excellence requires stable products
• Technical excellence requires stable platforms
• There is risk when relying on technology that is not fully in place and stable
• Internal escalation process for technical issues
Seven Characteristics of Highly Successful Projects

1. A positive relationship with an active, intelligent client
2. Strong project management
3. Clear requirements, well managed
4. Ruthless change management
5. Pervasive process focus
6. Effective controls and communication
7. Technical leadership and excellence
Thank you!