PRES 15053
The Effects of Software Process Maturity on Software Development Effort
Dashboard Concept

- **Code Inspection Coverage**
- **Pass Rate**
- **Defect Distribution**
- **Km/h**
- **Quality Management Tool**
- **Budget**
- **Scope**
- **Lagging**
- **Leading**

- **Historical Data Base**

- **50% Longest Chain**
- **Buffer Consumption**

- **(scope, leading, lagging)**
Architecture and design

**Preanalysis phase**
- Business events: identify
- Process requirements and models: analyse
- Business measures: analyse
- Service category map: prepare
- Sourcing considerations
- System use case: create

**Specification phase**
- Service specification: specify
- Service contract: specify
- Solution process flow: specify
- Architecture: scope and size: specify
- QA-review of architecture: Spec phase
- User profiles and Usage scenarios: create
- User eXperience: QA-review of architecture: Spec phase
- QA-review of architecture: Spec phase
- System use case: create

**Service category map**
- Preparation

**Functionality requirements and models**
- Analyse

**Architecture**
- Scope and size

**Specification**
- Preanalysis phase
- Solution process flow
- User eXperience QA-review of architecture
- System use case
Construction

Preanalysis phase

- Proof of concept
- Establish project surroundings

Technical implementation

- Technical infrastructure
- Design
- Physical design
- Develop
- Detailed prototype
- Construct

Business data

- Analyse changes
- QA-review of physical design
- Performance
- Evaluate

Performance

- User interface and Dialogue process control
- Construct
- System test environment
- Move solution to

Specifications phase

- Operational (modules)
- Construct
- Production environment
- Prepare
- Deploy to

Development phase

- Technical infrastructure
- Establish
- Data sources
- Construct
- QA-review of physical solution
- System test environment
- Check and prepare for

Implementation phase
Test

Preanalysis phase

Specification phase

Development phase

Test strategy general prepare

Test strategy specify

Test basis assure quality

System test execute

User test execute

Acceptance test execute

Non-functional requirements test execute
‘Cheshire Puss,’ she began, … ‘Would you tell me, please, which way I ought to go from here?’
‘That depends a good deal on where you want to get to,’ said the Cat.

‘I don't much care where –’ said Alice.
’Then it doesn't matter which way you go,' said the Cat.
‘- so long as I get somewhere,' Alice added as an explanation.
‘Oh, you're sure to do that,' said the Cat, ‘if you only walk long enough.’

Tell me where you want to be and I will show (measure) you the way
Bug Database

"which way I ought to go from here"

~33000 Records
With
36 Attributes
```
Call Center – Calls Database

```

```
~45000 Records
With
22 Attributes

“which way I ought to go from here”
```
“That depends a good deal on where you want to get to,’ said the Cat.”

‘Immediate’ Level Analysis

<table>
<thead>
<tr>
<th>Version View vs Other</th>
<th>Internal Status View vs Other</th>
<th>Company View vs Other</th>
<th>Call View vs Other</th>
<th>Priority View vs Other</th>
<th>View Cross vs. LC Record</th>
<th>Count of Call per View</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version vs OnAir</td>
<td>Internal Status vs OnAir</td>
<td>Company vs Status</td>
<td>Call vs Status</td>
<td>Priority vs Environment</td>
<td>Company &amp; LC</td>
<td>Priority</td>
</tr>
<tr>
<td>Version vs Sub Module</td>
<td>Internal Status vs TargetMile</td>
<td>Company vs Internal</td>
<td>Call vs Internal Status</td>
<td>Priority vs Status</td>
<td>Priority &amp; LC</td>
<td>Company</td>
</tr>
<tr>
<td>Version vs Status</td>
<td>Internal Status vs Sub Module</td>
<td>Company vs OnAir</td>
<td>Call vs Company</td>
<td>Priority vs Internal Status</td>
<td>Type of Call &amp; LC</td>
<td>Type of Call</td>
</tr>
<tr>
<td>Version vs Internal</td>
<td>Status View vs Other</td>
<td>Company vs TargetMile</td>
<td>Call vs OnAir Module</td>
<td>Priority vs Company</td>
<td>Closed on Initial Call &amp; LC</td>
<td>Closed on Initial Call</td>
</tr>
<tr>
<td>Version vs Call</td>
<td>Status vs OnAir</td>
<td>Company vs Sub Module</td>
<td>Call vs Target Milestone</td>
<td>Priority vs OnAir Module</td>
<td>Status &amp; LC</td>
<td>Environment</td>
</tr>
<tr>
<td>Version vs Company</td>
<td>Status vs TargetMile</td>
<td>Environment vs Status</td>
<td>Call vs Environment</td>
<td>Priority vs Target Milestone</td>
<td>Internal Status &amp; LC</td>
<td>Status</td>
</tr>
<tr>
<td>Sub vs TargetMile</td>
<td>Status vs Sub Module</td>
<td>Environment vs Internal</td>
<td>Call vs Sub Module</td>
<td>Priority vs Sub Module</td>
<td>Environment &amp; LC</td>
<td>Internal Status</td>
</tr>
<tr>
<td>OnAir Module View vs Other</td>
<td>Environment vs Company</td>
<td>Priority vs Call</td>
<td>Golive Target &amp; LC</td>
<td>Golive Target</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OnAir vs TargetMile</td>
<td>Environment vs OnAir</td>
<td></td>
<td>Target Milestone &amp; LC</td>
<td>OnAir Module</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OnAir vs Sub</td>
<td>Environment vs TargetMile</td>
<td></td>
<td>Sub Module &amp; LC</td>
<td>Target Milestone</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Environment vs Sub Module</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Process Maturity

**Process** – A set of interrelated activities, which transform inputs into outputs, to achieve a given purpose

**Institutionalization** - The ingrained way of doing business that an organization follows routinely as part of its corporate culture.

**Maturity level** - Degree of process improvement across a predefined set of process areas
Utilizing Project Current Data for Better Management Decisions

Increasing Project Data Usability Real Life Case Study
Disclaimer

- We have based the presentation content on the current program raw data, therefore presentation accuracy or level details presented may impacted by it.
- In some cases we guesstimate on data or some of its segments.
Unit Improvement Objectives

- Improve communication among the different stakeholders
- Increase system interfaces management and control efficiency
- To increase insight to effort deviation for better planning
‘Quality’ Objectives

To give the program and the division ideas, how to:

- Increase product / deliverable quality
- Reduce project lifecycle duration
- Reduce project cost
- Increase resource (human) utilization
- Increase processes efficiency
- Have better control on effort distribution
Initial Effort Planning

- Specification: 36%
- Management: 35%
- Development: 29%
Current Effort Distribution
For all Project Phases

- Construction: 30%
- Testing: 14%
- Analysis: 17%
- Management: 28%
- UAT: 0%
- Design: 5%
- SPEC: 5%
- Implementation: 0%
- BA: 1%
- OA (PCON): 1%
Defects by Type

- Cancelled: 9%
- Testing: 18%
- Documents: 28%
- STD: 42%
- External Data: 3%
Clients
<table>
<thead>
<tr>
<th>All</th>
<th>LC</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>71</td>
<td>40</td>
<td>56%</td>
</tr>
<tr>
<td>693</td>
<td>575</td>
<td>83%</td>
</tr>
<tr>
<td>1670</td>
<td>1572</td>
<td>94%</td>
</tr>
<tr>
<td>43</td>
<td>24</td>
<td>56%</td>
</tr>
<tr>
<td>547</td>
<td>455</td>
<td>83%</td>
</tr>
<tr>
<td>102</td>
<td>44</td>
<td>43%</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>8%</td>
</tr>
<tr>
<td>15</td>
<td>7</td>
<td>47%</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>25%</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>100%</td>
</tr>
<tr>
<td>112</td>
<td>81</td>
<td>72%</td>
</tr>
<tr>
<td>13</td>
<td>6</td>
<td>46%</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>36</td>
<td>20</td>
<td>56%</td>
</tr>
<tr>
<td>373</td>
<td>231</td>
<td>62%</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>15</td>
<td>9</td>
<td>60%</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>57%</td>
</tr>
<tr>
<td>676</td>
<td>418</td>
<td>62%</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>100%</td>
</tr>
<tr>
<td>15</td>
<td>7</td>
<td>47%</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>80%</td>
</tr>
<tr>
<td>661</td>
<td>569</td>
<td>86%</td>
</tr>
<tr>
<td>46</td>
<td>34</td>
<td>74%</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0%</td>
</tr>
</tbody>
</table>
Defects by Test Level

- Systemtest: 8%
- Unit-test (Operation): 4%
- Accept-test: 7%
- Brugertest: 81%
Let’s Try Some Mix and Match

We Will Demonstrate How Relationships Between Measures Can Benefit the Organization for Better Planning and Management
Some guesstimations on cost effectiveness

- If an average developer day cost is ~7000 units
- The total project effort was 10022 day (100%)
- The testing phase was 1453 day (14.5%)
- Defect that are the result of documentation are 69% of all defects

- If we will assume the to correct 69% of all defects will take around 40% of the testing duration;
- means that:
  - that will be 581 day
  - With the overall cost of 4068400 units

- However
  - Adding 30 review days in the static tests
  - and another 80 days of code inspection
  - will end with the cost of 770000 units

- And still we have saved at least 9401000 units (1343 days)
- Means that we were able to reduce 13.04% of the project time
What Organizational Processes we have touch

- Tailoring
- Scope and Size
- Status meetings
- Static Tests
- Testing (planning and execution) all phases
- Lesson learned
- Process Improvement
CMMI Effecting PA’s

• Project Planning
• Project Monitor and Control
• Measurement and Analysis
• Validation
• Verification
• Requirements Development
• Technical Solution
• Product Integration
• Organizational Process Focus
Practical Improvements Suggestions

- Requirements Development
  - Writing
  - Verifying
  - Validating

- Effort Distribution
  - Overhead planning
  - Estimation models
  - Project control
  - Lessons learned

- Verification
  - Planning
  - Guidelines for conducting
  - Checklist
  - Results analysis
  - Efficient communication
  - Lessons learned and root causes
# Activity Work Breakdown Structure

<table>
<thead>
<tr>
<th>PHASE</th>
<th>PERCENT OF EFFORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements Evaluation Phase</td>
<td>8%</td>
</tr>
<tr>
<td>Project Planning Phase</td>
<td>3%</td>
</tr>
<tr>
<td>Analysis Phase</td>
<td>10%</td>
</tr>
<tr>
<td>Design Phase</td>
<td>20%</td>
</tr>
<tr>
<td>Construction Phase</td>
<td>32%</td>
</tr>
<tr>
<td>Test Phase</td>
<td>23%</td>
</tr>
<tr>
<td>Implementation Phase</td>
<td>1%</td>
</tr>
<tr>
<td>Customer Support Phase</td>
<td>2.5%</td>
</tr>
<tr>
<td>Completion Phase</td>
<td>.5%</td>
</tr>
</tbody>
</table>

## Characteristic

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Level</th>
<th>Weightage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product complexity</td>
<td>High</td>
<td>1.15</td>
</tr>
<tr>
<td>Main storage constraints</td>
<td>High</td>
<td>1.06</td>
</tr>
<tr>
<td>Applications experience</td>
<td>Low</td>
<td>1.13</td>
</tr>
<tr>
<td>Programmer capability</td>
<td>Low</td>
<td>1.17</td>
</tr>
<tr>
<td>All other characteristic</td>
<td>Nominal</td>
<td>1.00</td>
</tr>
</tbody>
</table>

**Effort Adjustment Factor**

\[1.15 \times 1.06 \times 1.13 \times 1.17 \times 1.00 = 1.61\]

## Activity Work Breakdown Structure

<table>
<thead>
<tr>
<th>Activity</th>
<th>Small Project</th>
<th>Medium Project</th>
<th>Large Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Documentation</td>
<td>10</td>
<td>05</td>
<td>03</td>
</tr>
<tr>
<td>Project Management</td>
<td>25</td>
<td>15</td>
<td>10</td>
</tr>
<tr>
<td>Quality Assurance</td>
<td>15</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>User Training</td>
<td>10</td>
<td>07</td>
<td>02</td>
</tr>
<tr>
<td>Acceptance Testing</td>
<td>10</td>
<td>05</td>
<td>05</td>
</tr>
<tr>
<td>Performance Tuning</td>
<td>05</td>
<td>08</td>
<td>10</td>
</tr>
<tr>
<td><strong>Totals (%age)</strong></td>
<td><strong>75</strong></td>
<td><strong>50</strong></td>
<td><strong>40</strong></td>
</tr>
</tbody>
</table>
IMPERATIVE OCCURRENCE

shall 0
must 46
is required to 0
are applicable 0
are to 0
responsible for 0
will 18
should 3
TOTAL 67

NUMBERING STRUCTURE

DEPTH OCCURRENCE

1 2201 1 49
2 81 2 2
3 55 3 14
4 54 4 2
5 0 5 0
6 19 6 0
7 0 7 0
8 0 8 0
9 0 9 0
TOTAL 2410 TOTAL 67

DIRECTIVE OCCURRENCE

q. 0
e. 1
For example 0
Figure 0
Table 0
Note: 0
TOTAL 1

WEAK PHRASE OCCURRENCE

adequate 0
at this time 0
be able to 0
be capable of 0
capable of 0
capable to 0
determined 0
required 0
not required 0
normal 0
provided that 0
likely 0
easy to 0
TOTAL 0

CONTINUANCE OCCURRENCE

believe: 0
as follows: 0
following: 0
listed: 0
in particular: 0
support: 0
and 2
TOTAL 2

OPTION PHRASES OCCURRENCE

can 8
may 16
Optionally 0
TOTAL 24

INCOMPLETE OCCURRENCE

TBD 0
TBS 0
TBE 0
TBC 0
TBR 0
not defined 0
not determined 0
but not limited to 0
as a minimum 0
TOTAL 0
Practical Improvements Suggestions

• Validation
  • Planning
  • Guidelines for conducting
  • Checklist
  • Results analysis
  • Efficient communication
  • Lessons learned and root causes

• Measurements
  • Definition with direct line to business objectives
  • Measurements structures, content and context
  • Guidelines for collecting and ‘work with’
  • Checklist
  • Results analysis
  • Efficient communication
  • Lessons learned and root causes
### Control Measures

<table>
<thead>
<tr>
<th>Computed Metric Name</th>
<th>Alias</th>
<th>Objective of Computed Metric</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACWP</td>
<td>Actual Cost of Work Performed</td>
<td>Identify the actual labor hours spent on the project to date.</td>
</tr>
<tr>
<td>BAC</td>
<td>Budget at Completion</td>
<td>Identify the project’s budget.</td>
</tr>
<tr>
<td>BCWP</td>
<td>Budgeted Cost of Work Performed</td>
<td>Identify budgeted labor hours associated with the work that has been completed.</td>
</tr>
</tbody>
</table>

### Performance Measures

<table>
<thead>
<tr>
<th>Goal</th>
<th>Question</th>
<th>Metric</th>
<th>Definition</th>
<th>Source</th>
<th>frequency (dev)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Improve productivity</td>
<td>How efficient are tests?</td>
<td>Testing efficiency</td>
<td>Defects detected through testing / hour of testing</td>
<td>DTS</td>
<td>Monthly</td>
</tr>
<tr>
<td></td>
<td>How efficient are reviews?</td>
<td>Review efficiency</td>
<td>Defects detected through reviews / hour of review</td>
<td>DTS</td>
<td>Monthly</td>
</tr>
<tr>
<td></td>
<td>What is the productivity in fixed price projects?</td>
<td>Productivity</td>
<td>(Actual size of the product delivered to the customer / Actual effort spent to complete the project) in each technology platform</td>
<td>PINS (add size field)</td>
<td>End of the project</td>
</tr>
<tr>
<td></td>
<td>How effective is best practices sharing?</td>
<td>KR artifact usage index</td>
<td>KR artifacts used / project</td>
<td>KR</td>
<td>Monthly</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KR artifact contribution index</td>
<td>KR artifact added / project</td>
<td>KR</td>
<td>Monthly</td>
</tr>
</tbody>
</table>
Practical Improvements
Suggestions

Development & Interfaces Integration
- Improve content of guidelines in the different technical document to build more strong and clear descriptions
- Peer reviews
- ‘Internal’ documentation

Quality Assurance and Process Improvements
- Identify process goals and targets with direct line to business objectives
- Plan to process evaluation; including:
  - Guidelines for conducting
  - Checklist
  - Results analysis
  - Efficient communication
  - Lessons learned and root causes
Questions ?
Contact

Kobi Vider
K.V.P Consulting

Kobi.Vider@hotmail.com
KobiVP@aol.com

Phone: +972522946676