Building a Performance Team around a Maturity Model

There are reasons why there is a Malcolm Baldrige Award for Quality, ISO standards and maturity models for process improvement. The maturity model shows how to build processes and procedures that provide the ability to bring in capital projects on time and within budget. In Performance, leverage the maturity models to build processes to support the ability to measure, monitor and quickly analyze and manage the monitoring results.

The Performance Team is a highly cross-functional team interacting with the business, architecture, infrastructure, development, QA and often the user community. You need to have clearly defined processes and procedures documented when approaching each one of these teams. Also, performance is the last team to signoff on a release; therefore, repeatability of testing and configuring for tuning is a must. I recommend define a starting point with documentation and mature up the maturity model as a performance team.

Maturity Models
Early in my career I was fortunate to have a management team that was Quality based – from the top down the organization was focused on Quality of Product for the customer. I received training in Capability Maturity Model Interaction (CMMI) and ISO standards and today, when I go into an organization I still leverage the tools to determine the maturity level of the organization or department.

I have noted that organizations that follow a maturity model also remove a certain level of the infectious conditions that reside inside virtually all organizations that can cause more damage than economic downturns, management upheavals, or global business shifts – Robert Herbold calls it “The Fiefdom Syndrome”. A maturity model assists in reducing the Fiefdom Syndrome by establishing cross-functional team alignment through collaborative direction.

Maturity Models provide a starting point, leverage prior experiences, get cross functional teams on the same page of understanding to provide a shared vision, provide a framework for prioritizing tasks and a way show what improvements have taken place.

An Example - CMMI Levels
The CMM consists of five different levels of increasing process maturity. This five-stage structure of the CMM is based on the various principles championed by W. Edward Deming, Philip Crosby, Walter Shewart and Joseph Juran.

They include the following levels:
1 - Initial Level
2 - Repeatable Level
3 - Defined Level
4 - Managed Level
5 - Optimising

Level 1 - Initial level
Organizations at this entry level carry out their work on an ad hoc basis. A handful of formal processes are defined properly while project management discipline is, at best, unclear. As such success and failures have little impact on future undertakings. Results therefore become unpredictable, processes are poorly controlled and the ultimate success depends on the dedicated effort of a few enterprising individuals instead of the entire organization as a whole.

Level 2 - Repeatable level
At this second level, organizations depend mainly on policies for managing a software project and measures to apply those policies are established. These measures help the organizations to repeat successfully the previously mastered tasks and avoid the repetition of failures. The major chunk of an organization's processes at this level stays institutionalized, through staff experience instead of detailed documentation procedures.

Level 3 - Defined level
The various engineering activities and the processes of management at this level is formally defined, documented and integrated. In the process of development and maintenance of software, the organization's staff follows this defined standard process. At this third level, newer methods and tools can be added, and it becomes easier to train new staff to adapt according to the requirement of the organization.

Level 4 - Managed level
At this level, organizations stress the importance of quantitatively measuring the quality of the products delivered by each process. Detailed measures of the software process and product quality are collected and used to identify and correct issues with process performances. Organizations set quantitative goals for both software products as well as processes. As part of the organization's measurement program, productivity and quality of all software process activities and its supporting activities are measured. As new sets of tools or processes are added, or adjustments are made to already existing processes, measurement data enables the organization to access the success of the adjustment as well as prevent the recurrence of defects.

Level 5 - Optimizing level
At level 5, focus is on the continuous process improvement. The organization proactively identifies strengths and weakness in process, with the aim of preventing the occurrence of defects. Here continuous improvement becomes institutionalized into the development process. Instead of merely correcting defects as they are found, the main aim at this highest level is to stall future defects and address the key to those defects by planning in advance.
Characteristics of the Maturity levels

<table>
<thead>
<tr>
<th>Maturity Level</th>
<th>Key Process Areas</th>
</tr>
</thead>
</table>
| 5              | a. Defect Prevention  
|                | b. Technology Change Management  
|                | c. Process Change Management |
| 4              | a. Software Quality Management  
|                | b. Quality Process Management |
| 3              | a. Organizational Process Focus  
|                | b. Organizational Process Definition  
|                | c. Training Program  
|                | d. Software Product Engineering  
|                | e. Integrated Software Management  
|                | f. Inter-Group Coordination  
|                | g. Peer Review |
| 2              | a. Software Requirement Management  
|                | b. Software Project Planning  
|                | c. Software Project Tracking & Oversight  
|                | d. Software Subcontract Management  
|                | e. Software Quality Assurance  
|                | f. Software Configuration Management |
No Key Performance Areas or Indicators

Percent Improvement Compared with Results at Maturity Levels

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Level 1 - 2</th>
<th>Level 2 - 3</th>
<th>Level 3 - 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduce defects</td>
<td>12%</td>
<td>40%</td>
<td>85%</td>
</tr>
<tr>
<td>Reduce cycle time</td>
<td>10%</td>
<td>38%</td>
<td>63%</td>
</tr>
<tr>
<td>Reduce cost</td>
<td>8%</td>
<td>35%</td>
<td>75%</td>
</tr>
<tr>
<td>Schedule variance</td>
<td>145%</td>
<td>24%</td>
<td>15%</td>
</tr>
</tbody>
</table>

Source: Carnegie Mellon SEI CMMI

<More to come>