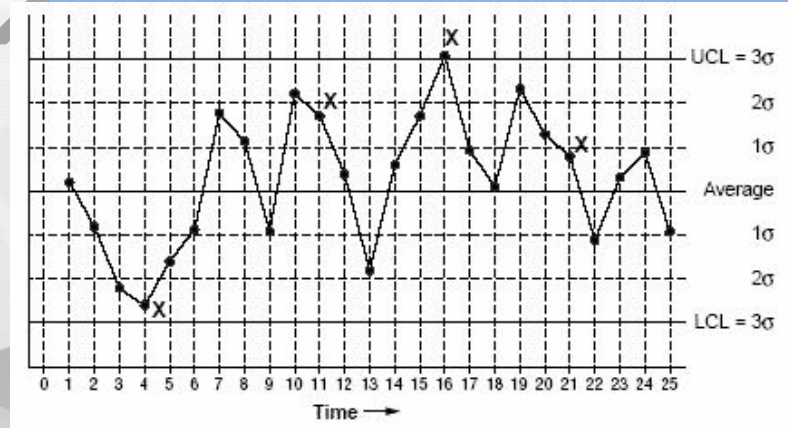
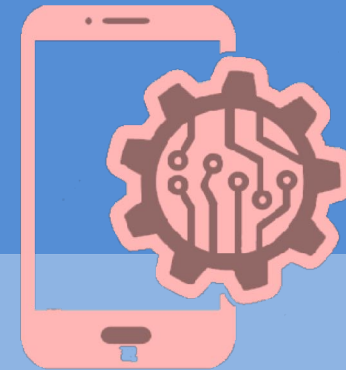


Software Process Improvement



Software Process Improvement

- Software process improvement encompasses a set of activities that will lead to a better software process and, as a consequence, higher-quality software delivered in a more timely manner. It implies many things:
- First, it implies that elements of an effective software process can be defined in more effective manner;
- Second, that an existing organizational approach to software development can be assessed against those elements;
- Third, that a meaningful strategy for improvement can be defined.

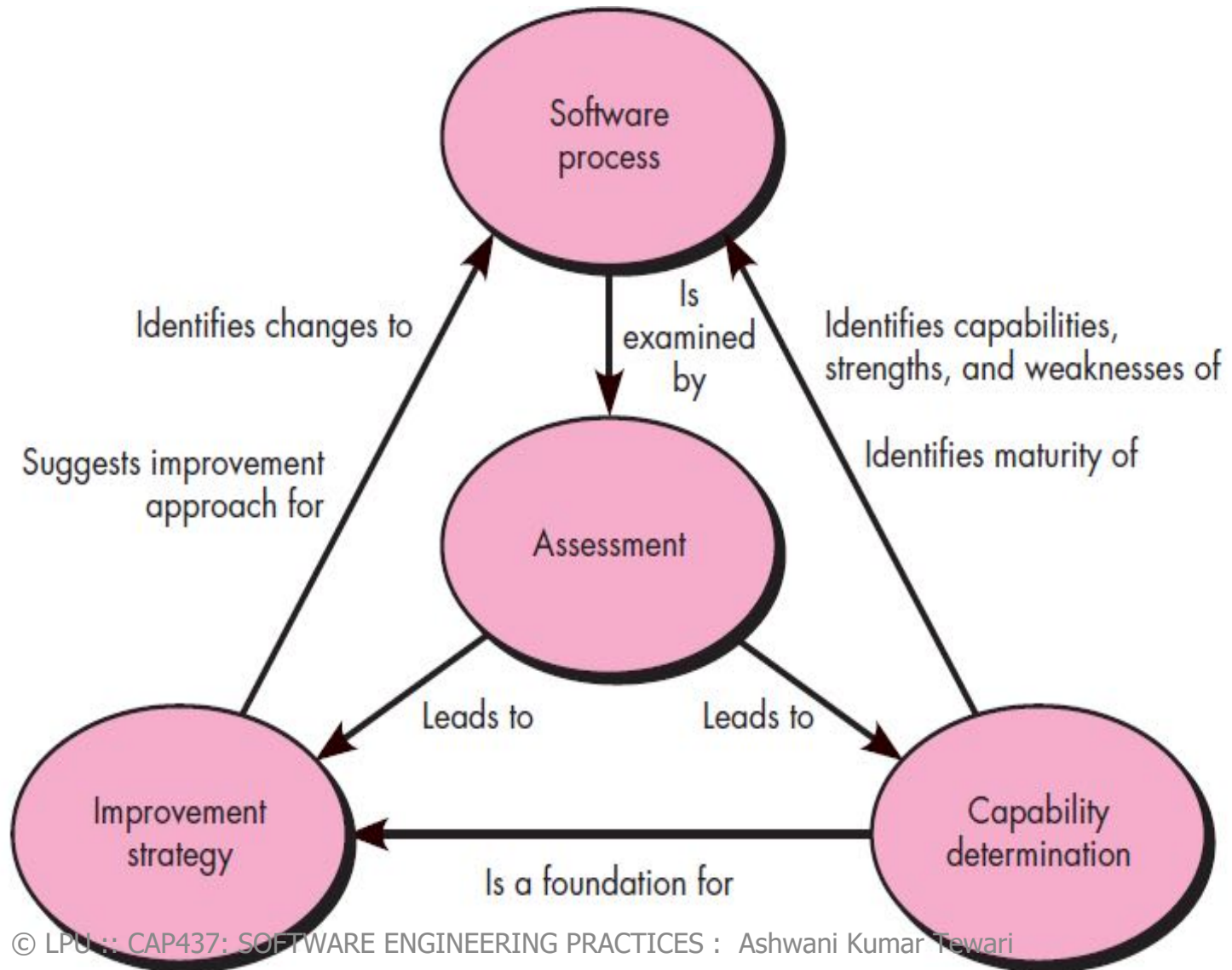
Software Process Improvement

- Because SPI is not free, it must deliver a return on investment. The effort and time that is required to implement an SPI strategy must pay for itself in some measurable way.
- To do this, the results of improved process and practice must lead to:
 1. a reduction in software “problems” that cost time and money.
 2. It must reduce the number of defects that are delivered to end users,
 3. reduce the amount of rework due to quality problems,
 4. reduce the costs associated with software maintenance and support,
 5. reduce the indirect costs that occur when software is delivered late.

Approaches to SPI

- An *SPI framework* defines
 - (1) a set of characteristics that must be present if an effective software process is to be achieved
 - (2) A method for assessing whether those characteristics are present.
 - (3) A mechanism for summarizing the results of any assessment.
 - (4) A strategy for assisting a software organization in implementing those process characteristics that have been found to be weak or missing.
- An SPI framework assesses the “maturity” of an organization’s software process and provides a qualitative indication of a maturity level.

SPI framework.



SPI support constituencies:

1. Quality certifiers. Process improvement efforts championed by this group focus on the following relationship:

$$\text{Quality}(\textit{Process}) \Rightarrow \text{Quality}(\textit{Product})$$

- Their approach is to emphasize assessment methods and to examine a well-defined set of characteristics that allow them to determine whether the process exhibits quality.
- They are most likely to adopt a process framework such as the CMM, SPICE, TickIT, or Bootstrap.

2. Formalists. This group wants to understand (and when possible, optimize) process workflow.

3. Tool advocates. This group insists on a tool-assisted approach to SPI that models workflow and other process characteristics in a manner that can be analyzed for improvement.

SPI support constituencies:

4. **Practitioners.** This constituency uses a pragmatic approach, “emphasizing mainstream project-, quality- and product management, applying project level planning and metrics, but with little formal process modeling or enactment support.
5. **Reformers.** The goal of this group is organizational change that might lead to a better software process. They tend to focus more on human issues and emphasize measures of human capability and structure.
6. **Ideologists.** This group focuses on the suitability of a particular process model for a specific application domain or organizational structure. Rather than typical software process models (e.g., iterative models), ideologists would have a greater interest in a process that would, say, support reuse or reengineering.

THE SPI PROCESS

- The Software Engineering Institute has developed IDEAL—“an organizational improvement model that serves as a roadmap for initiating, planning, and implementing improvement actions”.
- IDEAL is representative of many process models for SPI, defining five distinct activities—initiating, diagnosing, establishing, acting, and learning
- SPI originally proposed applying a commonsense philosophy that requires an organization to
 - (1) look in the mirror,
 - (2) then get smarter so it can make intelligent choices,
 - (3) select the process model that best meets its needs,
 - (4) instantiate the model into its operating environment and its culture, and
 - (5) evaluate what has been done.

1. Assessment and Gap Analysis

- The intent of assessment is to uncover both strengths and weaknesses in the way your organization applies the existing software process
- Assessment examines a wide range of actions and tasks that will lead to a high quality process.
- For example, regardless of the process model that is chosen, the software organization must establish generic mechanisms such as:
 1. defined approaches for customer communication;
 2. established methods for representing user requirements;
 3. defining a project management framework that includes scoping, estimation,
 4. scheduling,
 5. project tracking;
 6. risk analysis methods;
 7. change management procedures;
 8. quality assurance and control activities including reviews; and many others.

Education and Training

- A key element of any SPI strategy is education and training for practitioners, technical managers and more senior managers who have direct contact with the software organization. Three types of education and training should be conducted:
- **Generic concepts and methods.** Directed toward both managers and practitioners, this category stresses both process and practice. The intent is to provide professionals with the intellectual tools they need to apply the software process effectively and to make rational decisions about improvements to the process.
- **Specific technology and tools.** Directed primarily toward practitioners, this category stresses technologies and tools that have been adopted for local use.
- **Business communication and quality-related topics.** Directed toward all stakeholders, this category focuses on “soft” topics that help enable better communication among stakeholders and foster a greater quality focus.

Selection and Justification

In selection and justification process characteristics and specific software engineering methods and tools are chosen to populate the software process.

- First, you should choose the process model that best fits your organization
- Next, develop a work breakdown for each framework activity (e.g., modeling), defining the task set that would be applied for a typical project.
- You should also consider the software engineering methods that can be applied to achieve these tasks. In reality, selection can be a rocky road. It is often difficult to achieve consensus among different constituencies.
- Once a choice is made, time and money must be expended to instantiate it within an organization, and these resource expenditures should be justified

Installation/Migration

- In some cases, an entirely new process is recommended for an organization. Framework activities, software engineering actions, and individual work tasks must be defined and installed as part of a new software engineering culture.
- Installation and migration are actually *software process redesign* (SPR) activities
- When a formal approach to SPR is initiated, three different process models are considered:
 - (1) the existing (“as is”) process,
 - (2) a transitional (“here to there”) process,
 - (3) the target (“to be”) process.
- If the target process is significantly different from the existing process, the only rational approach to installation is an incremental strategy in which the transitional process is implemented in steps.

Evaluation

- *Evaluation* occurs throughout SPI. The evaluation activity assesses the degree to which changes have been instantiated and adopted, the degree to which such changes result in better software quality
- Both qualitative factors and quantitative metrics are considered during the evaluation activity.
- From a qualitative point of view, past management and practitioner attitudes about the software process can be compared to attitudes polled after installation of process changes.
- Quantitative metrics are collected from projects that have used the transitional or “to be” process and compared with similar metrics that were collected for projects that were conducted under the “as is” process.

Risk Management for SPI

Among the most common risks are:

1. a lack of management support
 2. cultural resistance by technical staff,
 3. a poorly planned SPI strategy,
 4. an overly formal approach to SPI,
 5. selection of an inappropriate process,
 6. a lack of buy-in by key stakeholders,
 7. an inadequate budget,
 8. a lack of staff training,
 9. organizational instability etc.
- A software organization should manage risk at three key points in the SPI process:
 1. prior to the initiation of the SPI road map, during the execution of SPI activities and during the evaluation activity that follows the instantiation of some process characteristic.

- In general, the following categories can be identified for SPI risk factors:

1. budget and cost,
2. content and deliverables,
3. culture,
4. maintenance of SPI deliverables,
5. mission and goals,
6. organizational management,
7. organizational stability,
8. process stakeholders, schedule for SPI development,
9. SPI development environment,
10. SPI development process,
11. SPI project management,
12. SPI staff.

- Within each category, a number of generic risk factors can be identified.
- Using the risk factors and generic attributes as a guide, risk exposure is computed in the following manner:

- $\text{Exposure} = (\text{risk probability}) \times (\text{estimated loss})$

Critical Success Factors

- The top five CSFs are presented here:

1. Management commitment and support: SPI will succeed only if management is actively involved. Senior business managers should recognize the importance of software to their company and be active sponsors of the SPI effort. Technical managers should be heavily involved in the development of the local SPI strategy. It is not feasible without investing time, money, and effort”. Management commitment and support are essential to sustain that investment.

2. Staff involvement: If SPI efforts are to succeed, improvement must be organic—sponsored by technical managers and senior technologists, and adopted by local practitioners.

Critical Success Factors

3. Process integration and understanding: S/W process must be characterized in a manner that is integrated with other business processes and requirements. To accomplish this, those responsible for the SPI effort must have an intimate knowledge and understanding of other business processes.

4. A customized SPI strategy: SPI road map must be adapted to the local environment—team culture, product mix, and local strengths and weaknesses must all be considered.

5. Solid management of the SPI project: SPI is a project like any other. It involves coordination, scheduling, parallel tasks, deliverables, adaptation (when risks become realities), politics, budget control, and much more.