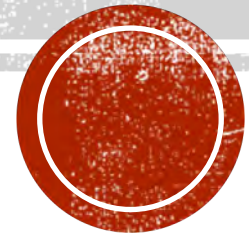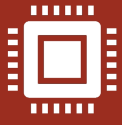# PROCESS IMPROVEMENT

Software Engineering

Dr. Raj Singh

The SPI strategy transforms the existing approach to software development into something that is more focused, repeatable, and reliable.

Elements of an effective software process can be defined in an effective manner.

An existing organizational approach to software development can be assessed against those elements.

A meaningful strategy for improvement can be defined.

SOFTWARE PROCESS IMPROVEMENT (SPI)

An SPI framework assesses the "maturity" of an organization's software process and provides a qualitative indication of a maturity level.

A set of characteristics that must be present if an effective software process is to be achieved

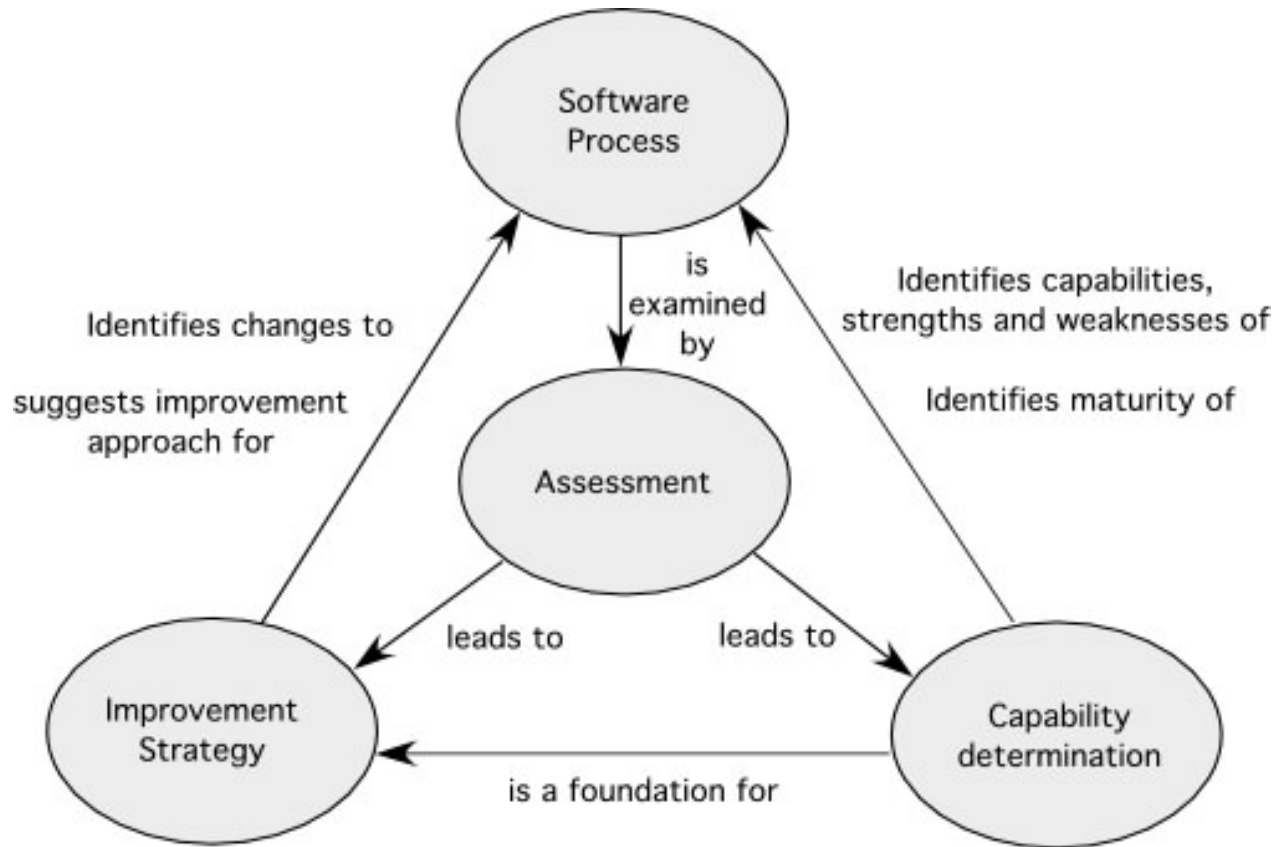A method for assessing whether those characteristics are present.

A mechanism for summarizing the results of any assessment.

A strategy for assisting a software organization in implementing those process characteristics that have been found to be weak or missing.

SPI FRAMEWORK

3

ELEMENTS OF A SPI FRAMEWORK

SPI CONSTITUENCIES

Quality Certifiers
- Based on quantifiable metrics for process and product.

Formalists
- Sticking to the standards and guidelines. Is it working?

Tool Advocates
- Which tools worked effectively? Should we try something new?
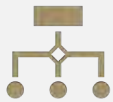
Practitioners
- What process worked effectively? Should we try something else?

Reformers
- Change is required to be more effective.

Provides an overall indication of the "process maturity" exhibited by a software organization.

An indication of the quality of the software process.

The degree to which practitioner's understand and apply the process.

The general state of software engineering practice.

MATURITY MODELS

IS SPI FOR EVERYONE?

✓ Can a small company initiate SPI activities and do it successfully? — Yes

₿ Small organizations are more informal. They apply fewer standard practices and tend to be self-organizing. — What financial benefit does it provide? I am not big enough to take the risk.
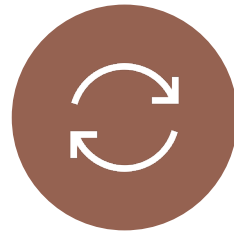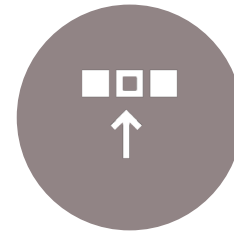
THE SPI PROCESS

ASSESSMENT & GAP ANALYSIS

EDUCATION & TRAINING

SELECTION & JUSTIFICATION

INSTALLATION & MIGRATION

EVALUATION

ASSESSMENT & GAP ANALYSIS

Assessment examines a wide range of actions and tasks that will lead to a high quality process.

Consistency — Are important activities, actions and tasks applied consistently across all software projects and by all software teams?

Sophistication. — Are management and technical actions performed with a level of sophistication that implies a thorough understanding of best practice?

Acceptance — Is the software process and software engineering practice widely accepted by management and technical staff?

Commitment — Has management committed the resources required to achieve consistency, sophistication and acceptance?

Gap analysis is the difference between local application and best practice represents a "gap" that offers opportunities for improvement.

EDUCATION AND TRAINING

Generic Concepts & Methods

- Directed toward both managers and practitioners, stresses both process and practice.
- Apply the software process effectively and to make rational decisions about improvements to the process.

Specific Technology & Tools

- Directed primarily toward practitioners, stresses technologies and tools that have been adopted for local use.

Business Communication & Quality

- Directed toward all stakeholders, focuses on "soft" topics that help enable better communication among stakeholders and foster a greater quality focus.

SELECTION AND JUSTIFICATION

Choose the process model that best fits your organization, its stakeholders, and the software that you build

Decide on the set of framework activities that will be applied to produce the quality work.

Checkpoints that will enable your team to assess progress

Develop a work breakdown for each framework activity, defining the task set that would be applied.

Once a choice is made, time and money must be allocated and expenditures should be justified.

INSTALLATION/MIGRATION

SPR is concerned with identification, application, and refinement of new ways to improve and transform software processes.

Three different process models are considered

the existing ("as-is") process	a transitional ("here-to-there") process	the target ("to be") process

Assess the degree to which changes have been instantiated and adopted.

The degree to which such changes result in better software quality or other tangible process benefits.

The overall status of the process and the organizational culture as SPI activities proceed.

Check the practioner's pulse after installation of process changes.

EVALUATION

13

- budget and cost
- content and deliverables culture
- maintenance of SPI deliverables
- process stakeholders
- schedule
- environment and process
- SPI project management

RISK FACTORS FOR SPI

CRITICAL SUCCESS FACTORS

	Commitment and Support	Get everyone on same page
	Staff Involvement	Get people excited about the change
	Process Integration and Understanding	Do we understand why change is needed?
	Strategy	Do we have a strategy for the change?

OTHER SPI FRAMEWORKS

SPICE — an international initiative to support the International Standard ISO/IEC 15504 for (Software) Process Assessment [ISO08]

Bootstrap — a SPI framework for small and medium sized organizations that conforms to SPICE [Boo06]

PSP and TSP — Individual and team specific SPI frameworks ([Hum05a], [Hum05b]), more rigorous approach to software development coupled with measurement

TickIT — an auditing method [Tic05] that assesses an organization compliance to ISO Standard 9001:2000

SPI RETURN ON INVESTMENT

How do we know that we'll achieve a reasonable return for the money we're spending?

- ROI = [S (benefits) – S (costs)] / S (costs)] 3 100%

Where,

- benefits: the cost savings associated quality, less rework, reduced effort associated with changes, and the income that accrues from shorter time-to-market.
- costs: both direct and indirect costs associated with greater emphasis on quality control and change management.

SPI TRENDS

Future SPI frameworks must become significantly more agile

Rather than an organizational focus (that can take years to complete successfully), contemporary SPI efforts should focus on the project level

To achieve meaningful results (even at the project level) in a short time frame, complex framework models may give way to simpler models.

Rather than dozens of key practices and hundreds of supplementary practices, an agile SPI framework should emphasize only a few pivotal practices

REFERENCE

- Roger Pressman, Software Engineering: A Practitioner's Approach, 8th edition, McGraw Hill, ISBN 0078022126