A Tutorial on

BUILDING A SOFTWARE PROCESS IMPROVEMENT INFRASTRUCTURE: MODEL & PRACTICAL LESSONS

Author: Dr Sami Zahran

1 JUSTIFICATION AND DEFINITION

1.1 Why an Infrastructure is needed?

Process institutionalisation requires a process culture and a process infrastructure. The infrastructure is necessary to enable and facilitate the software process activities and support the process-related roles and responsibilities. This Paper describes the software process infrastructure that needs to be in place in order to support the process and sustain continuous process improvement.

A software process infrastructure covers two areas:

- **Organisational and management infrastructure:** with defined roles and responsibilities that have to be in place to sponsor, manage, perform and monitor software process improvement activities.
- **Technology and tools infrastructure:** This incorporates the necessary facilities and tools for automating process activities and supporting the various process improvement roles and responsibilities.

Both of these aspects have to be covered in order to sustain an environment for software process improvement. The role of the infrastructure is to institutionalise the software process improvement behaviours into the organisation culture. This is necessary if these are to survive into the organisation after those who championed the original efforts have gone. An expressive quotation from Phil Crosby (Crosby 1980) emphasises the need for quality to be "engrained" in the organisation: "If quality isn't "engrained" in the organisation, it will never happen" (Philip Crosby)

This is equally true for process improvement. We can convert this quote into an equally true statement for continuous software process improvement to read:
"If continuous software process improvement isn't engrained in the organisation, it will never happen"

"Engraining" or "institutionalising" the process improvement behaviours into the organisation will not happen without an effective infrastructure to support the process. Investing in the

software process infrastructure is a critical factor if an organisation is serious about building a culture of continuous software process improvement. Management decision to invest in the infrastructure illustrates a commitment to software process improvement. I have seen many an organisation and many senior managers who are ready to pay lip services to the process culture. But when it comes to authorising investment for building a support infrastructure, the enthusiasm evaporates.

Building an effective support infrastructure to support and emphasise process improvement activities and behaviours is one of, if not the main, mechanisms that help in embedding and ingraining process improvement behaviours in the organisation.

An effective infrastructure should cover:

- roles and responsibilities for process ownership.
- roles and responsibilities for process training and knowledge dissemination.
- enforcement procedures to ensure adherence to the process standards.
- feedback mechanisms for the collection and analysis of feedback data on the process performance.

This Paper describes the main components of the software process infrastructure covering organizational and management as well as the technical aspects.

1.2 What is an Infrastructure?

Infrastructure is an important concept. The word "infrastructure" is defined in the Little Oxford Dictionary of Current English as the "structural foundations of a society or an enterprise; roads, bridges, sewers, etc. regarded as countries economic foundations". In the context of an organisation, infrastructure can be defined as: "the underlying framework of an organisation or system, including organisational structures, policies, standards, training facilities, and tools, that support its ongoing performance." (Paulk et al. 1995).

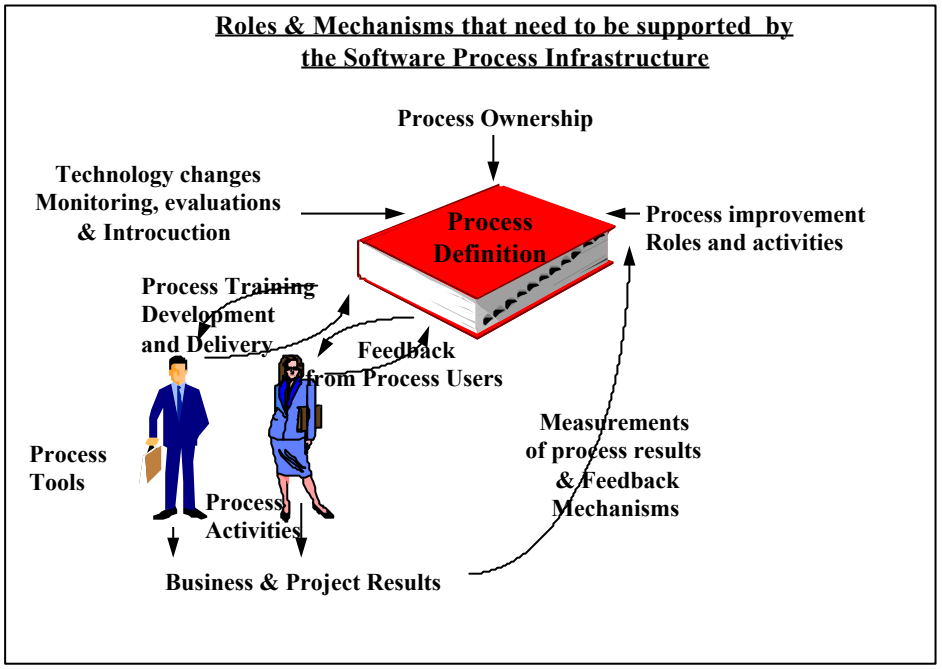
In general the role of the infrastructure is to provide support and guidance to the process activities and provide the channels and tools through which the process activities are to be performed and monitored. In the software process context, we can define the software process improvement infrastructure as follows. "software process infrastructure is the underlying framework of organizational and technical foundations that support the ongoing software process improvement activities including process definition, process modelling, process training, process monitoring, process enforcement, and ongoing feedback on the process performance".

1.3 A Model for the Software Process Infrastructure

The software process infrastructure should provide the roles and responsibilities necessary for supporting an effective process environment. Figure 1 illustrates the roles and mechanisms that need to be supported by the process infrastructure. The implementation of these roles and mechanisms should cover all the relevant organisational levels (e.g. corporate, project, team and personal processes). These roles and mechanisms should cover all the key areas for the software process. An effective infrastructure for the software process improvement environment should cover two aspects, an organisational and management roles and responsibilities, and technical environment. When designing a software process infrastructure we have to bear in mind two domains:

- **the organisational domain:** Organisational levels that the infrastructure need to support (e.g. corporate, project, team, or personal level)
- **the process domain:** what key process categories will the infrastructure support (e.g. software engineering processes, management processes, support processes, etc.)

Figure 1 Roles and Responsibilities for the Software Process Infrastructure



Some of the mechanisms illustrated in Figure 1 are static and some are dynamic:

- static components holding the process definitions, and
- dynamic feedback mechanisms and process activities.

An infrastructure that supports these activities and feedback mechanisms should provide the technical components that will support the static element of the model, and the organisational roles and responsibilities that will drive the dynamic element of the model. For the infrastructure to support effective process performance, it should provide roles and mechanisms that will lead to the process displaying the characteristics summarised in Table 1.

Table 1 Mechanisms for An Effective Process Support Infrastructure

No.	Process Characteristics	Infrastructure Mechanism
1	defined	process documentation
2	owned	process group
3	trained	training mechanisms
4	supported	process tools
5	followed	process enforcement
6	monitored	feedback mechanisms
7	continuously improving	management sponsorship

1.4 Roles and Mechanisms

The infrastructure should support roles and mechanisms to cover the following functions:

- **Process Ownership:** This is ownership, maintenance, and dissemination of the software process definition. These cover roles and responsibilities for ownership of the process documentation, plus tools to enable storage and retrieval of the process documentation and data.
- **Process Training:** This covers the development and provision of the software process training. For this the infrastructure should define roles and tools to cover: all aspects for process training such as: what, when, where, who, why. It also should cover the collection of feedback data on process training.
- **Performance Monitoring**: This covers monitoring of the performance of software process activities. For this the infrastructure should cover aspects such as: what to monitor, where to direct the feedback, how to monitor the actions, to whom do we report the actions, etc.
- **Process Enforcement:** Enforcement of the software process standards and practices, for which the infrastructure should define (and provide if necessary) the measurement tools, enforcement roles and mechanisms, collection, and the escalation actions)
- **Process Support:** Support of the ongoing process activities, for which the infrastructure should provide the latest tools and supporting technology through monitoring of the technological development and continuous alignment with business goals.
- **Technology Introduction:** Intercepting, evaluating and introducing new software engineering technologies selecting appropriate technologies and planning their introduction into the organization.

The software process infrastructure fulfil these functions, through its two domains:

- **Management and organisational infrastructure:** to provide and support the roles and responsibilities required for supporting the process.
- **Technology and tools infrastructure:** to support the automation of process activities, collection of process performance activities, retrieval of the process definitions, etc.

These two types of infrastructure should extend their support to cover the different organisational levels that may exist in the organisation (e.g. corporate, project, team, and personal level). A case study describing a process support infrastructure roles and responsibilities as well as the supporting technology is described in (Zahran 1996).

Figure 2 illustrates an example of the organisational levels that may exist in most organisations (corporate level, project level, and personal level), the type of process at that level, and the main objective of that process.

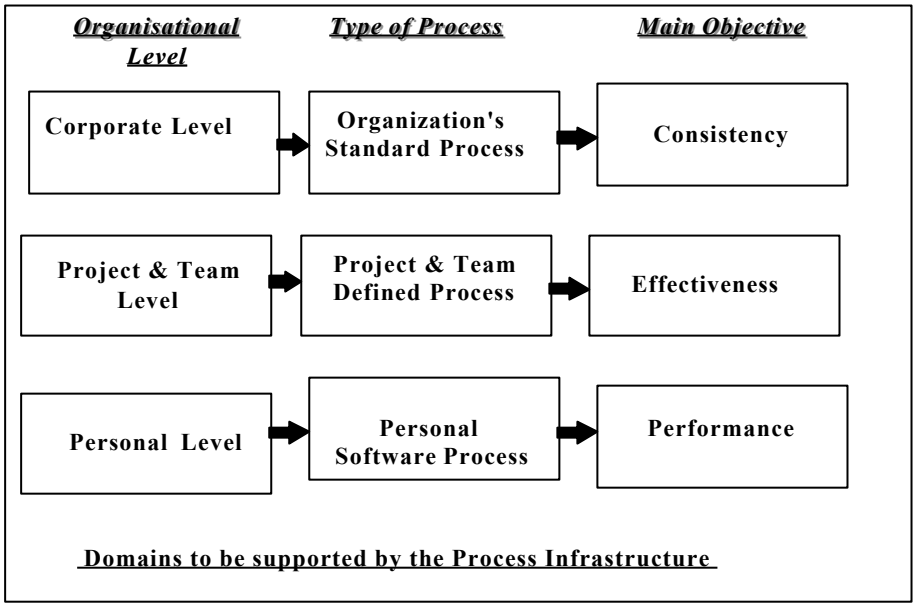


Figure 2 Organisational levels for the process infrastructure

1.5 Organisational Levels for Infrastructure Support

Both organisational and management infrastructure and technical infrastructure should enable and enhance the roles and functions across three organizational levels. The nature and focus of the infrastructure support at each of these levels could vary along the following lines:

Corporate level:

At the corporate level, the process infrastructure comprise the organisation's process assets which incorporates the "organisation's standard process". Following are some of the main objectives for the process resources at the corporate level.

- Providing support and consistency to the process activities throughout the organization.
- Providing direction and guidance to the process improvement activities throughout the whole organisation,
- Ensuring visibility of and support for the process focus at a corporate level and throughout the organization.

- Encouraging the adoption of best practices across projects, and promoting the process discipline across the organisation.

Project/Team level:
At the project/team level process infrastructure comprise the process assets for the project or the team. There could more than one team in the project sharing common tasks and objectives that need specific process for that team. The common project process is referred to as the "project's defined process", and a team's process could similarly be defined as the "team's defined process": Whether it is a team's process or a project's process the main objectives of the process are:

- To ensure the effectiveness of the process in terms of its impact on the project or the team's progress,
- To provide support and guidance to the project personnel to team members,
- To support the project management in monitoring the project progress by reporting the status of the project activities, and providing feedback the project learning to the organisation-level process.

Personal level: At the personal-level, the process infrastructure comprise the process assets available to support the individual software engineers perform their software engineering activities. This is referred to as the "personal software process (PSP)", a term coined by Watts Humphrey (Humphrey 1997). The main objectives of the infrastructure are the following:

- To automate mundane tasks and activities
- To monitor the personal performance of individual software engineers by reporting on the status of the personal software engineering activities of individual software engineers.

2 ORGANIZATION AND MANAGEMENT INFRASTRUCTURE

2.1 Requirements Of the Organization and Management Infrastructure

Alignment with Business Objectives
The main objective of an organisational and management infrastructure is to support the software engineering groups' effort to produce software in an efficient, systematic, and continuously improving manner. Such infrastructure could have an impact on the following: process effectiveness, product quality, software engineers productivity, accuracy of cost/time estimations, effectiveness of project planning and project tracking, management visibility of project progress, consistency of processes across the different projects, developers and top management acceptance, contribution to shifting the culture and behaviours from fire fighting to continuous improvement, contribution to the establishment of a learning organisation's culture across the software engineers and project managers. Ultimately all these factors should lead towards the achievement of the quality, cost and functionality goals of the software projects.

In order to achieve success in "(trans)planting" the process into the organisation, a number of important factors should be taken into consideration when planning and implementing

software process improvement environment. The following factors will have an impact on the process infrastructure.

- Existing organisational culture
- Existing organisational structure and hierarchy
- Current roles and responsibilities
- Potential sources of support
- Potential sources of resistance.

This is why understanding the organizational context is critical for achieving alignment between the infrastructure and the current organization.

Process Support Roles

The following observations should achieve alignment between the process organizational infrastructure and the existing organization structure:

- The software process should be supported by an organisational infrastructure that will enable the process activities run smoothly and effectively. throughout the organisation
- New and redesigned software processes must be supported by appropriate new / redefined roles, responsibilities and procedures at the different levels within the organisation
- The software process support roles and responsibilities must be embedded into the organisation's culture at all relevant organisational strata

Degree of Flexibility

Modern organisations are often complex and span across national borders. An organisation may comprise a multitude of autonomous units with varying degrees of co-ordination. The following questions should be considered when designing the process infrastructure:

- What degree of flexibility (or rigidity as the case may be!!) do we need to build into the infrastructure without loosing the consistency across the corporation. Flexibility may be necessary to respect the local characteristics and conditions of the different parts of the business.
- What degree of uniformity (or autonomy !) do you want to build across the different organisational entities without loosing consistency.

Relevance to the Projects and Benefits to the Business

A disciplined software process should help software projects in achieving their goals of quality, cost, and schedule. The process infrastructure should be designed with this primary goal in mind, that is to help projects achieve (and possibly exceed) their target goals. Helping the projects succeed, is the bridge between the creation of an effective software process and benefits to the business. Process discipline is a way of enhancing the process effectiveness. An effective process is the way for ensuring that the project achieves its quality, cost and schedule goals, thus leading to improved products and services for the business.

Creating a process infrastructure should never be treated as a theoretical exercise or an experiment performed by a small number of people in an ivory tower. Rather it should be considered as a strategic initiative by all those involved in software development and in the quality of products and services that depend on software. The business benefits expected out of improved quality of the products and services should be easy to establish. An effective way to justify investment in software process improvement is to establish a link between an effective process environment and the improved quality of the resulting products and services.

There are other potential business benefits of an effective software process. They could be just as important as the improved product quality. Examples of such benefits are:

- Reduction of projects' cost and duration. This is especially important in the area of big custom projects, where the margins are very tight and the risk factor is rather high.
- Increasing the competitiveness and effectiveness as a result of the increased predictability that an effective process could bring. This should have a positive impact on the time-to-market, value-for-money, and customer satisfaction.
- Minimising the risk through a reusable, repeatable process that has been proven to work in other projects. This provides a proof of the organization software capability, which should increase its competitiveness.
- Increased flexibility. This results in being better able to respond to the requirements of a diverse and demanding market and the variations of customer requirements.
- Availability of proof of credibility. This results from the track record of process performance data and metrics collected from past projects, and a proof of the process repeatability.

All such benefits collectively should assist in raising the company profile, and increasing its competitiveness.

2.2 Organizational And Management Enablers

Enablers for Process Change

Software process improvement actions are likely to impact the organizational structure, working policies and procedures, as well as skills, roles and responsibilities of human resources. Effectively everyone in the organisation involved with software or software engineering activities, is likely to be affected by the process organization and management infrastructure. Such involvement could take the form of one or more of the following roles:

- User of the process infrastructure.
- Performer of the process activities.
- Recipient of process training.
- Provider of feedback on process performance.
- Participant in process improvement activities.

In his book titled "Process Innovation" , Thomas Davenport (Davenport 1993) lists a number of organizational and human resource enablers of process change. Following is a list of potential change enablers for software process improvement and their main focus:

- ***Organisational Enablers:*** **Focus: Structure and Culture.**
  These include the organizational roles and responsibilities needed to support the process and process improvement. These include roles and responsibilities for management sponsorship, corporate software engineering process group and software process improvement teams.
- ***Cultural Enablers:*** **Focus: Empowerment and Open Participation in Decision Making**
  These include organisational policies and procedures. The objectives are to empower the corporate software engineering process group, empower the software process improvement teams, encourage and reward feedback on the process performance and encourage and reward novice suggestion of process improvement.

- ***Human Resource Enablers:*** **Focus: Process Competency**
  These include process training, motivation and enforcement. The objective is to get software engineering staff:

- Trained in the process.
- Motivated to follow to the process.
- Rewarded for over-achieving the process goals.
- Incentivised for their contribution to process improvement activities.

These enablers should be mapped and assigned to the following roles necessary for supporting the software process infrastructure:

- Sponsorship Roles
- Management Roles
- Co-ordination and Enforcement Roles
- Process Improvement Roles

In summary we stress the importance of considering software process improvement actions primarily as a change process. They aim at changing the habits, and behaviours of software engineers and project managers. Table 2 summarises the organizational and management enablers for software process improvement.

Table 2 Organisational and Management Enablers for Software Process Improvement

Type of Enabler	Software Process Improvement Enablers & Mechanisms
Organisational Enablers	Executive Sponsor for Software Process Improvement
	Corporate Software Process Group (SEPG)
	Software Process Improvement Teams (Software PITs)
	Project-Level Software Process Co-ordinator
Cultural Enablers	Empowerment of software process group (SEPG) and Software Process Improvement Teams
Human Resources Enablers	Career paths for members of the software process group (SEPG)
	Compensation and Rewards for participants in the Software Process Improvement Activities
	Compensation and Rewards for projects and individuals who conform with the new processes and prove its effectiveness and impact on the business results

Special Role of the SEPG

Although all the roles mentioned above are important for spreading and emphasising the process discipline, two of these roles are of paramount importance. These are the executive sponsorship and the software engineering process group (SEPG). The importance of executive sponsorship is clear enough and is necessary for the success of any major change programme. The role of the SEPG is specific to software process changes. A number of authors have addressed the SEPG roles and responsibilities (Fowler & Rifkin 1990, Dorsey & McDonald 1996). According to Fowler and Rifkin the SEPG is the focal point for process improvement. *"Composed of line practitioners who have varied skills, the group is at the centre of collaborative effort of*

everyone in the organization who is involved with software engineering improvement." (Fowler & Rifkin 1990).

More discussion of the SEPG is provided later in this Paper.Table 3 summarises the SEPG special role, objectives and tasks in a software process improvement programme. Practical lessons and experiences of the formation and performance of SEPGs by organisations who pioneered software process improvement appear the literature (Dorsey & McDonald 1996, Reed 1996).

Table 3 Objective and Tasks of the Software Engineering Process Group (SEPG)

Objective	**SEPG Task/Activity**
Leadership and Keeping the SPI Momentum:	• Obtains and maintain the support of all levels of management • Provides overall direction for the SPI programme • Ensure involvement of all software engineers and project managers
Facilitating the Change	• Works with line managers whose projects are affected by changes in software engineering practice, providing a broad perspective of the improvement effort and helping them set expectations. • Maintains collaborative working relationships with software engineers, especially to obtain, plan for, and install new practices technologies
Process Training & Consultancy	• .Arranges for any training or continuing education related to process improvements. Provides process consultation to development projects and management.
Process Assessment and Improvement Planning	• Facilitates software process assessments • Transforms the process assessment into process improvement plans • Drive and co-ordinate the process improvement plan
Monitoring Process Performance and Collecting Feedback	• Encourages, facilitates and collects feedback on process performance • Tracks, monitors, and reports on the process performance Analyses the process metrics, publish results and feed them into process improvement plans
Defining the Process	• Facilitates the creation and maintenance of process definitions, in collaboration with the software process improvement teams (Software PITs) • Maintains a process database. • Encourages the managers and engineering staff. to participate in the Software PITs activities
Co-ordinating Software Process Improvement Teams	• Co-ordinating the Software PITs activities • Securing management sponsorship for members of the Software PITs activities

2.3 Management Roles and Responsibilities

The software process organisation and management infrastructure could be defined as *"an organisational schema with defined roles and assigned responsibilities capable of handling the software process management and process improvement activities".*

To support the effective operation of such organisational schema, the following need to be defined:

- Roles and their respective responsibilities should be sufficiently documented
- Deliverables of each role
- For every deliverable, define templates and / or quality control check-lists where applicable

Following are examples of the functions that need to fulfilled by the process improvement roles and responsibilities:

- **Consolidated Allocation of Resources for Software Process Improvement Activities:** Unavailability of resources for software process improvement is one of the most serious factors that could impede the progress of software process activities. The most common excuse is that all software engineering resources are busy with project work. Such attitude could jeopardise the success of software process improvement.
- **Cost/Time Estimation of the Software Process Improvement Activities:** This is especially relevant at the time of business justification, and preparing a return on investment case for software process improvement.
- **Managing Software Process Improvement Activities as proper Projects:** Software process improvement activities should be managed in the same way as a development project. It should have the disciplines of project management, project planning, project monitoring, project metrics, and so on.
- **Quality Control and Co-ordination of Inter-departmental Activities:** Management should monitor the quality of process improvement actions. For actions shared across more than a single function, management should allocate ownership and responsibility of such interdepartmental activities. Otherwise ownership could be lost among the different departments involved.
- **Feedback on Software Process Improvement Activities:**. Possible role for management could include receiving feedback information, analysing feedback data, and planning improvement actions for the software process improvement plan itself.
- **Planning of necessary Training for the Software Process Improvement Teams and the SEPG**: Some software process improvement actions are different from software engineering activities (for example the design of new processes). This may require special training for those involved in these activities. Management has to plan for and authorise such training.
- **Motivating the Software Process Improvement Teams:** This is necessary to ensure the continuous enthusiasm and commitment to software process improvement.

When designing the organisational schema for the infrastructure, you should define and allocate roles and assign responsibilities to the appropriate level of authority. These should be sufficiently documented, and the appropriate commitment gained from those concerned.

2.4 Organizational Model

Figure 3 illustrates a model for the process organizational and management infrastructure with its components and feedback mechanisms. The model is based on the architecture suggested in (Fowler & Rifkin 1990).The entities shown are:

- Executive sponsor
- Steering committee
- Corporate Software Engineering Process Group (SEPG)
- Process Improvement Team (Software PITs)
- Projects

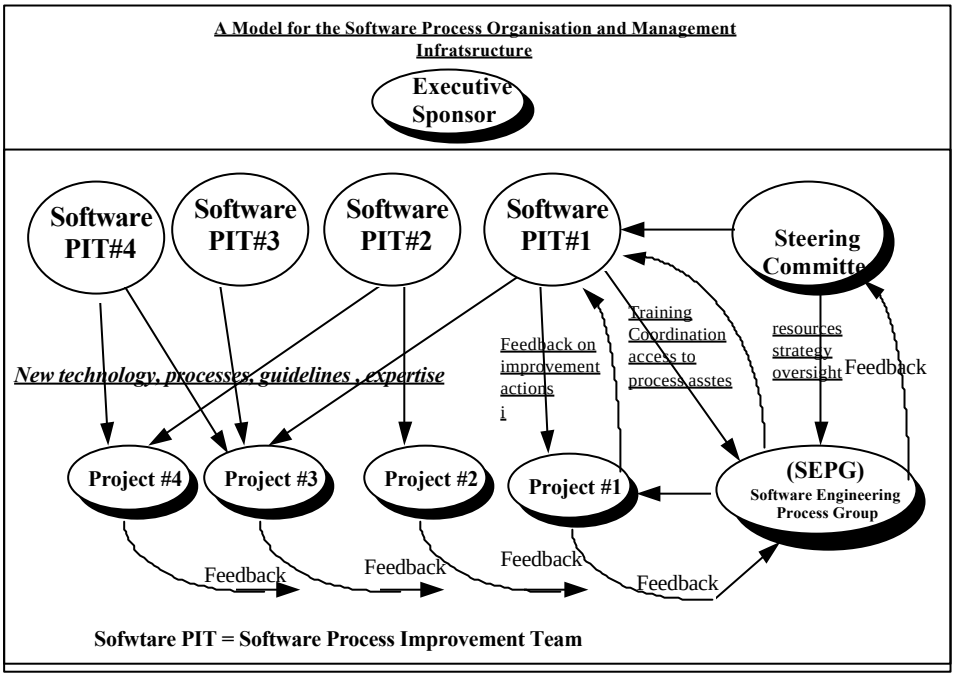


Figure 3 A Model for Software Process Organisation and Management Infrastructure

Following is a description of the roles and responsibilities specific to process improvement.

Executive Sponsor and Leader

Role:

There should be an overall sponsor of the software process improvement programme. Such a programme could be part of a bigger corporate initiative, for example a total quality management (TQM) initiative or a business process reengineering initiative. Usually the sponsor is a senior executive at a corporate level (as opposed to departmental or project level). He/she is expected to adopt a vision for the long-term business objectives of the process improvement programme, and assume leadership towards achieving these adjectives.

Responsibilities:

- Authorisation of funding for the software process improvement programme.
- Ensuring the management commitment and continuous sponsorship of the process improvement effort in the organisation.

- Ensuring the co-ordination between the software process improvement programme and related corporate programmes, for example. total quality initiatives (TQM), or business process re-engineering (BPR).
- Authorisation of any software process enforcement measures which may require changes / additions to the corporate policies and procedures.
- Acting as the driving force behind the overall programme for software process improvement and injecting enthusiasm about continuous software process improvement throughout the organisation.

Steering Committee

Role:

The steering committee (also sometimes called Process Improvement Council, or Executive Improvement Team) is a policy-making body of a group of senior and line managers who develops the organisation's overall strategy for software process strategy and monitors its progress. The SPI steering committee meets periodically to translate corporate policies into actions and align the SPI programme priorities and direction accordingly.

Responsibilities:

- Setting strategic direction for SPI activities
- Reviewing the results of the of the software process assessments
- Approving the formation of the software process improvement teams (PITs)
- Approving the operating plans of, and set priorities for the various software PITs
- Monitoring and overseeing the progress of the software PITs
- Helping obtain sponsorship and resourcing for the software PITs
- Conducting ongoing policy oversight, resource and process management
- Negotiating with higher levels of management and corporate planning organisations
- Integration and consensus building among different software PITs

Software Engineering Process Group (SEPG)

Role:

The SEPG is a corporate level group tasked with the co-ordination and support of all software process improvement teams and activities across the whole organisation. The SEPG should act the central driving force for software process improvement, and the focal point for process improvement effort across the organization. Due to its significant role in software process improvement, special attention should be given to the formation of the SEPG (Dorsey & McDonald 1996) both in terms of its organisational structure, membership, and responsibilities. Table 4 summarises some of the issues that need to be addressed when forming an SEPG.

Table 4 Typical SEPG Issues that need to addressed

No.	Aspect	Issues
1	Organisational Structure	• Inside or outside the software engineering organization ? • At what organizational (seniority) level ? • Separate department or virtual group ? • Reporting channel ? • What to report ?

		• Organisational relationships ?
2	Membership	• How many people ? • Full time or part time ? • Period of membership ? • Qualifications for membership ? • Nominations procedures ?
3	Responsibilities	• Specific tasks and functions (see list of suggested roles above? • Example areas of responsibilities: assessments, recommendations, implementation of improvement actions, process consulting, communications, training, metrics, technology introduction, etc.
4	SEPG Internal Process	• Mission statement ? • Operational procedures ? • Mechanisms and tools to assist / automate SEPG tasks ?

Responsibilities:

- Co-ordinating all process improvement activities across the organization, and obtaining and maintaining the support of all levels of management.
- Nominating and recruiting members of the software process improvement teams (Software PITs), and co-ordinating their activities. (The SEPG does not have to perform all the process improvement activities themselves, but should provide support and co-ordination to the process improvement teams)
- Acting as the "keepers" of the process. They should have the responsibility of the maintaining the process assets, encouraging / soliciting the feedback of the process, planning and driving the process improvement efforts, co-ordinating the process improvement teams, and assuring synergy across the various software process improvement teams (Software PITs)
- Assuming responsibility for developing / selecting / installing process-related methods, techniques and tools within the corporation.
- Maintaining collaborative working relationships with software engineers. This is especially important to design, plan for, and install new practices and technologies.

Software Process Improvement Teams (Software PITs)

Role:

These are teams of software engineers that give part of their time to the process improvement, focusing on a specific process (for example, requirements' management process, project planning process, project tracking process, etc.). Every key process area should be allocated a Process Improvement Team (PIT) Each team takes responsibility and ownership of the improvement efforts of that of the key process areas. Membership of each PIT would vary according to the nature and boundaries for the key process area. An example is that membership of a "Requirements Management Process Improvement Team (RM-PIT) should include representatives of the customers, representatives of the project management, representatives of the subcontractors, representatives of the whole system of which the software is a component.

Responsibilities:

- The general responsibility of a PIT is implementing the software process improvement actions for the key process area allocated to the team. Specific activities could cover a wide variety of areas such as:
- Documentation, analysis and redesign of current processes
- Redesign of the processes under consideration
- Documenting the new processes
- Evaluating, selecting methods, techniques, and tools in support of the new processes (for example configuration management techniques and tools, design methods and tools, etc.).
- Developing and conducting process training
- Liaison with the SEPG to co-ordinate with related software PITs responsible for related key process areas. An example is co-ordination between the Configuration Management PIT and the Software Quality Assurance PIT to design common change control and review procedures).
- Designing procedures for monitoring the process performance, and specifying the feedback mechanisms that should be in place to collect feedback and the tools for analysis of the feedback data.

Process Owners

Role:

A process without ownership could easily decay and turn into a "shelfware". Without explicit ownership everyone will be busy in his/her daily activities. Explicit ownership is essential if the process is to be kept alive and in alignment . The ownership role could come under the banner of the SEPG, but we believe that the process ownership is quite different from the mainly co-ordinating role of the SEPG. Also ownership could vary depending on the nature of the key process area.

Responsibilities:

- Acting as the ultimate authority for that process and leading the process improvement team for that process.
- Taking responsibility for process design for that specific key process are and for co-ordinating the process improvement team for that process.
- Assuming responsibility for ensuring that the total process is effective and efficient.
- Providing vision, strategy and leadership to the process improvement team.
- Ensuring that the process is followed, and monitoring the feedback on the process performance, and drive any resulting improvement actions.
- Anticipating business changes and their impact on the process.

2.5 Software Process Improvement Teams (Software PITs)

A software PIT should take on the responsibility of one or more of the key processes. One PIT could be assigned one key process area, or a number of inter-related key process areas. A software PIT will consist of 4 to 12 professionals, and will be responsible for designing and continuously improving its allocated key process area.

The acronym PIT was first used by James Harrington in the context of business process improvement. In his book "Business Process Improvement" (Harrington 1991) he (amusingly) describes PIT as the most appropriate acronym for the process improvement teams because in his own words:

"The pit is the centre of many of the delicious fruits we eat. It is the seed that brings about new life, new growth, and increased productivity. The PIT is also the centre of our improvement activity. ..... As

with the pit in a peach or palm, proper nurturing of the PITs will bring about new growth for your organization and increased effectiveness, efficiency, and profits." (Harrington 1991)

Following are some examples of software PITs (Figure 4):

Requirement Management PIT: To be responsible for designing, tracking, and improving the Requirements Management Key Process Area.

Project Planning & Project Tracking PIT: To be responsible for designing, tracking, and improving the Project Planning & Project Tracking Key Process Area.

Software Subcontract Management PIT: To be responsible for designing, tracking, and improving the Software Subcontract Management Key Process Area.

Software Quality Assurance PIT: To be responsible for designing, tracking, and improving the Software Quality Assurance Key Process Area.

Software Configuration Management PIT: To be responsible for designing, tracking, and improving the Software Configuration Management Key Process Area.

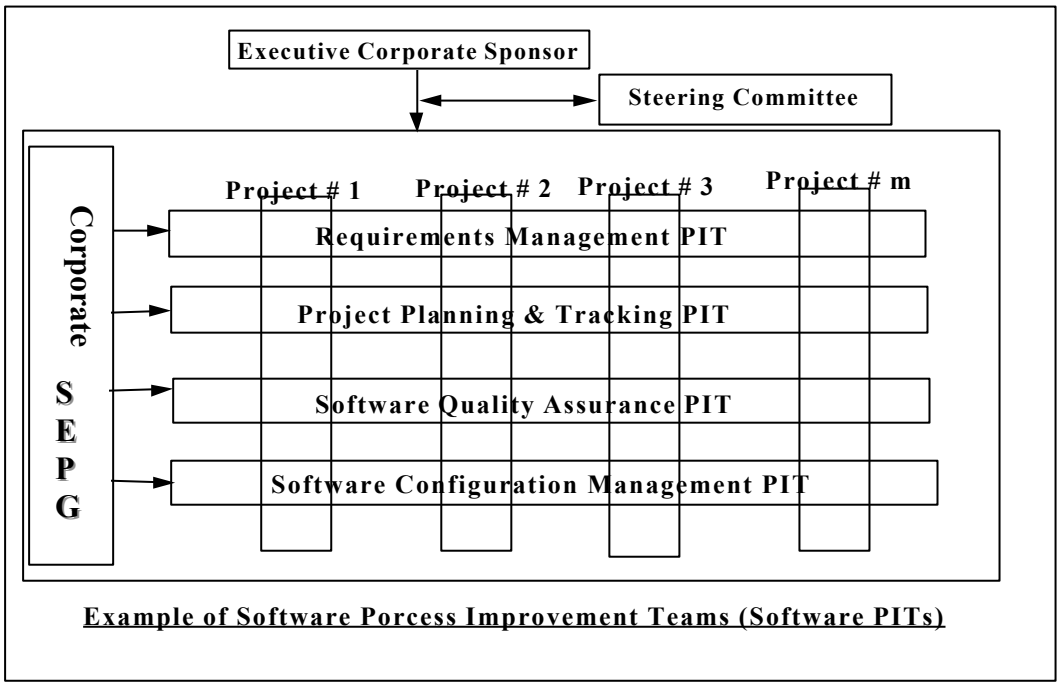


Figure 4 Examples of Software Process Improvement Teams (Software PITs)

Every PIT will have a team leader who could also take the role of the process owner for the key process area allocated to the team. Co-ordination of the different Software PITs is the responsibility of the SEPG.

The SEPG will be responsible for the overall corporate software process improvement programme. The programme should be divided into discrete process improvement projects, that are assigned to the software PITs and other groups if necessary. Membership of the SEPG should be on a permanent basis, as opposed to membership of the Software PITs that could be on a "rotating basis" and part-time basis. This will give the opportunity for as many software engineers as possible to become involved in process improvement activities.

On-going Role of the SEPG and the Software PITs

Both the SEPG and the Software PITs should not be just transient phenomena in the life of the organisation. Although they may not have permanent membership, they should remain in place as organisational entities for as long as software development is critical for the business. Table 5 summarises the on-going responsibilities. It is a modified version of the responsibilities proposed in (Fowler & Rifkin 1990).

Table 5 Responsibilities of the Process Organizational & Management Infrastructure Entities

Infrastructure Entity	Level	On-going Responsibilities
Executive Sponsor	Strategic	• Policy, • Funding
Steering Committee	Strategic & Tactical	• Strategy, • Oversight, • Feedback to Sponsor
Process Group (SEPG)	Strategic, Tactical & Operational	• Maintaining action plan, • Disseminating reports, • Co-ordinating process improvement activities
Software PITs	Tactical & Operational	• Documenting, Analysing, and Redesign of Current Processes • Tracking and evaluating Process Performance. • Acquiring new methods technologies, and tools to support the process.
Projects	Tactical & Operational	• Piloting new processes and technologies, • Provide feedback on the process effectiveness

2.6 Leadership Role and Executive Sponsorship

Establishing a software process organisation and management infrastructure will involve many parts of the organisation that cross the organisational boundaries. Local management sponsorship may not be enough to push and enable SPI actions across organisational boundaries. Sponsorship in its traditional way of being limited to authorisation of budgets, may not be enough. Additional to this software process improvement requires leadership and commitment. This is evident in a number of well-publicised early success stories of software process improvement. Two of the most well publicised case studies are the software process improvement at Hughes Aircraft and at Raytheon (Humphrey et al. 1991, Dion 1993). Leadership is more than management.

Leadership has the following ingredients (Bennis 1989, Kennedy 1996, Humphrey 1997):

- **Guiding Vision and Establishing Direction:** The leader has a clear idea of what he wants to do and the strength to persist in the face of setbacks, even failures.
- **Passion:** A leader of software process improvement should have a passion about the case of SPI. He should love what he/she does and love doing it ...
- **Integrity and Self-discipline:** Integrity means self-knowledge and maturity. An SPI leader will be promoting messages on process maturity, discipline and continuous improvement. If he/she does not exhibit these attributes, the messages coming out of him will be no more than "vapour ware". Soon the staff around him/her will realise that it is no more than "lip service", and the faith in the management commitment to process improvement will soon disappear.
- **Taking Risks:** The leader is willing to take risks, experiment and try new things. The software process improvement involves change, and every change carries with some risks. Leaders take risks. They doe not worry about failures, but embraces errors, knowing that they will learn from them.
- **Motivating, Inspiring and Aligning People:** A leader of software process improvement should keep people moving in the right direction despite major political, bureaucratic and resource barriers to change.

As in the case of any change programme, leadership is essential for the success of software process improvement. In spite of being very important, leadership is not tangible. An organisational schema will not usually has a box labelled leadership. Leadership could materialise at more than one level: strategic, tactical, or operational. The ideal scenario is that leadership is evident at the top of the organisation. It will then disseminate to other levels in the organisation.

The majority of failures of change programmes could be attributed to absence of leadership, and lack of management commitment. While management skills and resources are abundant, true leadership is rare and hard to get. It is worthwhile spending a moment to compare the attributes and behaviours of managers versus those of leaders.

Table 6 contrasts a selection of attributes and behaviours of managers versus those of leaders.

Table 6 Managers Versus Leaders

The Manager	The Leader
The manager administers	The leader innovates (leading the way to a vision for software process improvement effort in the organisation)
The manager focuses on systems and structure, and stresses the status quo (stress current processes)	The leader focuses on people, has his eye on the horizon, and challenges the status quo (leading the way to changing the current status of the software processes towards a continuously improving processes)
The manager relies on control (of software engineering activities)	The leader inspires trust (into the software process improvement teams)

The manager has a short term view (the current projects)	The leader has a long-range perspective (the future project and future business direction)
The manager asks how and when (to apply current processes)	The leader asks what and why (what needs improvement and why it will add value to the business)
The manager focuses on control and problem solving (Fire fighting)	The leader energise people to overcome problems (Defect prevention)
The manager does the right thing (by stressing the current processes)	The leader does the right thing (by starting a software process improvement programme to move the organisation from a fire fighting culture to a defect prevention culture)
The manager follows	The leader sets the direction

It is important to emphasise that we are not trying to devalue the important role of management at the expense of leadership. Leadership; management sponsorship and staff buy-in are all critical factors that determine the success or failure of a software process improvement. The importance of management sponsorship is only equally paralleled with the importance of the staff buy-in. These two aspects are strongly inter-dependent:

- Having a management sponsorship in place should succeed in achieving the following. Passing the management message on software process improvement to software engineers and project managers and ensuring their buy-in and commitment. Without such a sponsorship, the software process improvement is likely to fail.
- In the same way, securing executive sponsorship is essential for getting the staff buy-in and gaining their commitment. As evidence has shown, it is very easy for a project manager to stop his staff participation in software process improvement activities at the first sight of a fire fighting battle.

In summary the following three factors are all important and inter-linked.

- Leaderships provide vision and direction,
- Management provides control and focus, and
- Staff commitment enables all these to happen.

The absence of any of these factors could put the software process improvement in jeopardy:

- **In the absence of leadership**, software process improvement activities are likely to be considered by project managers as an overhead, and by software engineers as a secondary non-core activity. If this is the case, the forces of resistance to change will prevail due to the inertia of the current organisation, and the software process improvement programme will gradually disappear in the background.
- **In the absence of management commitment**, software engineers are not likely to be released from their project work for participating in the activities of the software process improvement teams. The time and effort spent by software engineers in process improvement will not be valued as much as the time spent in the project activities. This attitude will ultimately dissuade the software practitioners from participation in the software PITs activities.

- **In the absence of staff buy-in**, there will be resistance to implement new process changes. The new processes could be ignored (under one execs or another). Staff participation in the software process improvement teams is likely to dwindle..

2.7 Need for Organisational Fit

The exact definition of the roles suggested above will vary according to the specific situation within the organization. The introduction of these roles into the organization could occur gradually and in steps. The risks of introducing these roles prematurely are non-trivial. They could jeopardise the probability of success for software process improvement activities.

Software process improvement experiences now increasing in popularity. The number of practical lessons and case studies published in the literature is on the increase. One of the shared lessons is the need to carefully consider soft factors. These include factors such as resistance to change and cultural drivers (Meyrs 1996, Dorsey & McDonald 1996, Jones et al. 1996). Taking these factors into consideration, and being sensitive to you organization's internal rhythm you should be able to define an effective organisational and management infrastructure to support the software process improvement.

Matching the Organisational Context

A process organisational and management infrastructure covers the roles and responsibilities for establishing, promoting, monitoring, and enforcing the process discipline. It should match and be aligned with the corporate culture and corporate organization. Any misalignment could lead to the creation of an organisational stress. Misalignment with corporate cultural drivers could lead to resistance to software process improvement. Cultural drivers manifests themselves as patterns of behaviours that seem to be driven by pre-existing mental models that become "engrained" within the organisational fabric. A recent paper (Jones et al. 1996) summarises experiences in introducing software process improvement in Defence organizations, and highlights the potential conflict between new insights and deeply-held internal images of how the world works. These shape the cultural drivers which can be, quite often, the enemies of "change agents".

On the other hand, alignment between process roles and responsibilities and the organisational structure could facilitate "acceptance of the process by managers and software engineers. into the organisation. As an example, a multi-national corporation would require a corporate-level process roles and responsibilities, a country-level process roles and responsibilities, project and team level process roles and responsibilities, as well as personal level process roles and responsibilities.

For the software process to be effective, it needs to be introduced effectively and uniformly into the organisation's texture in a consistent manner.

3 PROCESS TECHNICAL INFRASTRUCTURE

3.1 An Architecture for the Process Technical Infrastructure

Technical Infrastructure

The technical infrastructure should provide support to the software process activities at all organisational levels (corporate level, project and team level, and individual level). It enables the establishment and management of the organisation's process assets. A software process technical infrastructure includes the technical facilities, computing platforms, and tools that support the software engineering process group and process improvement teams in performing their process-related activities. Generally the process technical infrastructure covers two areas (Figure 5):

- Process Support Tools
- Organisation's Software Process Assets

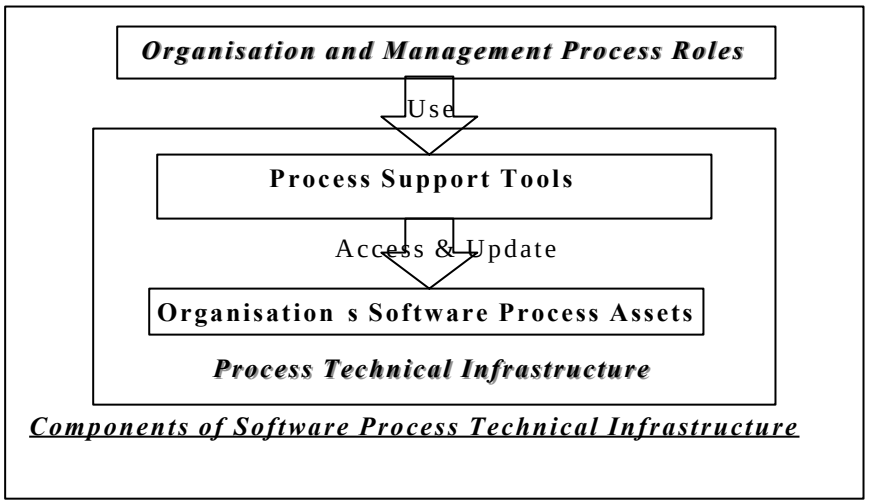


Figure 5 Components of Software Process Technical Infrastructure

The infrastructure should provide support at the following organizational levels:

- **Corporate level** The technical infrastructure should support process-related functions at the corporate level. Infrastructure users at corporate level include the executive sponsor, the steering committee, and the SEPG.
- **Project and Team level**: The technical infrastructure should support process-related functions at the project and team level. Infrastructure users at the project and team level include project managers, the project controller, team leaders and the software engineers working on the project.
- **Personal level**: The technical infrastructure should support the process-related activities at the personal level. Software engineers will use the infrastructure facilities at the personal level for improving and monitoring their personal software process (PSP).

Having an effective and flexible technical process infrastructure is essential for achieving the process effectiveness. Ownership of the process definitions and use of the infrastructure elements should be specified at all these levels, along the line discussed in the previous sections. Managing the infrastructure itself could be part of the management of the total IT

infrastructure of the corporation. What we discuss in this section is the functionality required by such infrastructure in order to effectively support the software process activities.

Requirements of the Process Technical Infrastructure

The software process roles and responsibilities should be supported by a set of tools to enable the performance of the process activities effectively. Figure 6 highlights the types of tools required to support the process activities.

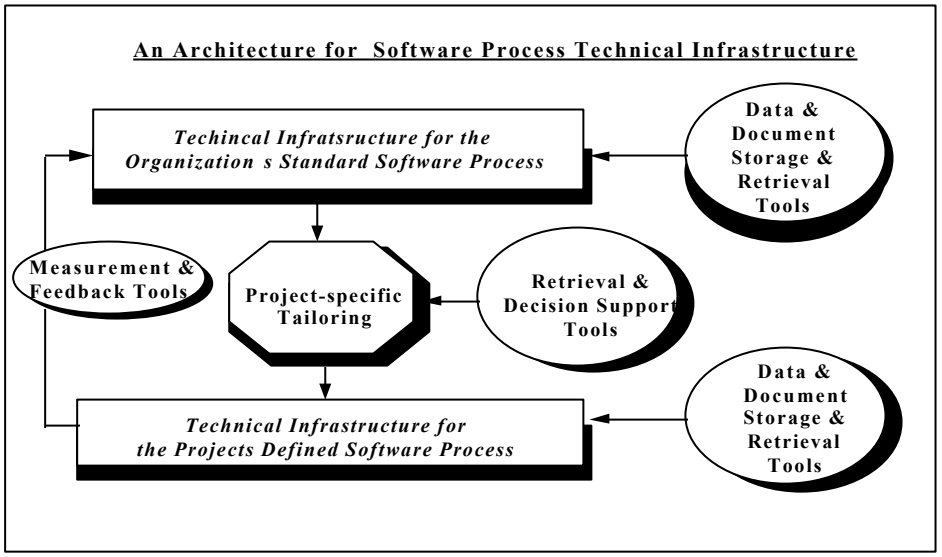


Figure 6 Process-support Technical Infrastructure

The technology infrastructure should provide functions to support the corporate level, the project level, and the personal level as summarised in Table 7.

Table 7 Functions of Process Support Tools

No	Type of Tool	Functions
1	**Data and Document Storage & Retrieval Tools**	***Corporate Level*** - To assist the software engineering process group (SEPG) store and modify the organisation's standard process models, definitions, and measurement data. *Project Level* - To enable project managers and the project's software process controllers retrieve the organisation's standard process models, definitions and performance data. - To enable project managers submit feedback data on the performance of the project's defined software process - To enable project managers and the project's software process controllers to store and modify and retrieve the project's defined software process models, definitions, and performance data. *Personal Level* - To enable software engineers retrieve the project's defined process

		- To enable software engineers store, modify, and retrieve the personal software process models, definitions, and performance data - To enable software engineers submit feedback data on the performance of the personal software process
2	**Retrieval and Decision Support Tools**	*Corporate Level* - To assist the SEPG retrieve, analyse, and evaluate information on the project's defined software process *Project Level* - To assist the project managers access the organisation's standard process definitions, models and data plus the tailoring guidelines. - To assist the project managers simulate various tailoring scenarios for tailoring the organisation's standard process to match the specific requirements of the project. *Personal Level* - To assist software engineers retrieve, analyse, and evaluate information on their personal software process performance
3	**Process Modelling & Simulation Tools**	*Corporate Level* - To enable the SEPG to model and simulate the organisation's standard process, store, retrieve and update these models *Project Level* To enable the project managers and the project process controller to model and simulate the project's defined process, store, retrieve and update these models *Personal Level* To enable software engineers to model and simulate the personal software process (PSP), store, retrieve and update these models

The process technical support infrastructure at the corporate level should be flexible enough to allow individual projects to build their own technical process support environment. A project specific processes should match the project's special features or specific requirements. Having an effective technical support infrastructure for the process is essential for achieving the effectiveness of the process. The process technical infrastructure should cover all the components of the organisation's process assets.

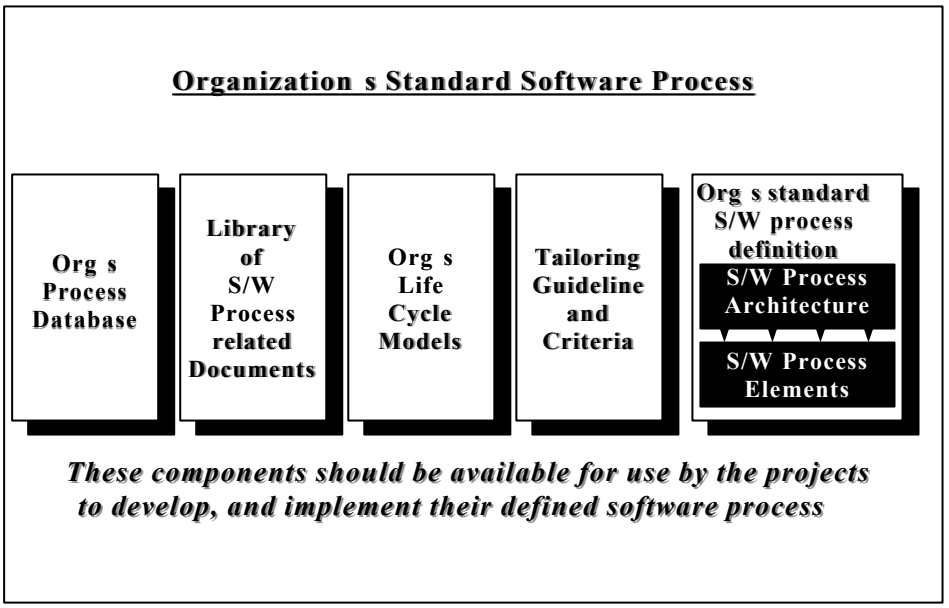
3.2 The Organisation's Standard Software Process Assets

An Architecture of the Organisation's Software Process Assets

The elements of the process definitions and documentation are collectively known as the organisation's software process assets. These are defined as follows: "Software process assets are a collection of entities maintained by an organisation for use by projects in developing, tailoring, maintaining, and implementing their software processes (Paulk et al. 1995)"

Figures 7 and 8 illustrate the main components of the organisation's standard software process and software process assets. Such architecture is described as a conceptual framework for the software process capability (Paulk et al. 1995).

Figure 7 An Architecture for the Organisation's Standard Software Process



The actual instances of these assets should cover all the key process areas for the software process. They should cover for example the Requirements Management Process, the Project Planning Process, the Project Tracking Process, the Software Quality Assurance Process and the Configuration Management Process.

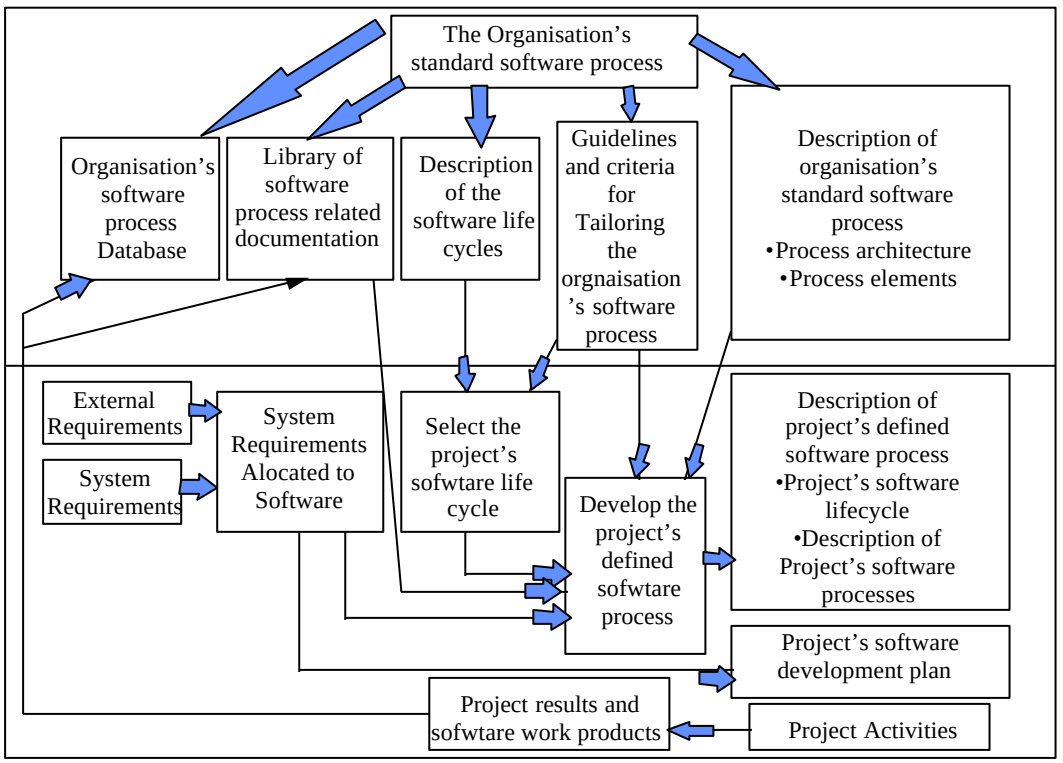


Figure 8 Organization's Software Process Assets

(Organisation's and Project's Levels)

The organisation's software process assets may appear in different formats and use a variety of media (for example electronic data and information stored in a database, physical documents or document images, or standard templates). Collectively these assets represent the corporate memory that contains the collective project experiences. It can be used as an evidence and an indicator of the organisation's software capability. The corporate software engineering process group assumes the overall responsibility of the organisation's software process assets. The organisation's software process assets comprise the following components:

- **The Organisation's Standard Software Process:**
  This covers definitions and descriptions of the software process architecture and the software process elements. A process modelling tool may be used to generate and store the process descriptions in textual and/or diagrammatic forms, and to make them available for retrieval, in a variety of formats, by the software engineers and managers.
- **Approved Software Life Cycles:**.
  This covers descriptions of the life cycles approved by the organisation for use by the projects. This may take the form of a list and descriptions of these life cycles, plus (a reference or pointer to) the detailed technical documentation and available training for these life cycles.
- **Tailoring Guidelines:**
  These specify guidelines and criteria for tailoring the organisation's standard process. Such tailoring is necessary to make the organization's standard process usable and applicable to a specific project. The tailoring guidelines could be either general across the board, or specific to a key process area. For example the general guidelines may cover: how to select a life cycle for the project out of the approved life cycles, what project characteristics influence the choice of the life cycle, what life cycle is compatible with specific standards, etc. An example of the specific guidelines is the guidelines for tailoring the corporate definition of the configuration management process in order to generate a project-specific configuration management plan. Such guidelines may vary from one key process area to another, and they could be attached to the description of the key process area concerned.

- **The Organisation's Process Database:**
  As an essential component of the process infrastructure the process database holds the process definitions and the process performance measurements. The organisation's standard process database (also referred to as the corporate process database) holds all the process-related data. It holds the organisation's standard process descriptions. Another example of its contents is the measurements' data collected on the performance of the projects' processes, the final products, and the overall organisation's process. The measurements' data should cover the product quality, project performance, and process performance. The measurement data could be analysed and used for a number of reasons including understanding, evaluation, control, and prediction

The SEI definition of the organisation's process database is as follows:

"The process database is a central collection of files containing all critical information on the process and product aspects of software development and maintenance. It is maintained by the process group.

Using simple definitions, the measurement portion of the database should contain a few indicative, useful measures of the products and processes. In addition to a quantitative database, the process group should maintain a file of process improvement lessons learned (Fowler et al. 1990)"

- **Library of Software Process-related Documents:** This is sometime called process assets' library (PAL). This is a library where all process-related documents from the completed software projects are stored. These documents represent the process experiences of the projects, and together with the measurements data stored in the organization's process database, they indicate the process capability of the organisation.

- **Contents of the Process Assets Library**
  John Baumert describes some practical lessons gained while establishing and populating a corporate-wide process asset library (Baumert 1996). The contents of process asst library are summarised in Table 8. These assets should be "owned" and maintained by the SEPG. It should be available for use by the projects to develop and implement their project-defined software process. They collectively represent the corporate process memory.

Table 8 Example Contents of the Process Asset Library

No.	Process Asset Library Subject Area	Example Contents
1	Process Definition Assets	• Process Definitions • Process Models • Process Methods • Process Products • Process Standards • Process Roles • Process Policies • Selection and Tailoring Criteria
2	Process Implementation and Support Assets	• Product Templates • Samples • Training Resources • Tool Information
3	Evaluation Support Assets	• Industry Reference Model Descriptions (e.g., CMM, ISO 9000) • Evaluation Criteria • Cross Reference Mappings between the organization's process and industry reference models

4	Historical Project and Service Data	• Project / Service Characteristics • Project / Service Metrics • Lessons Learned • Process Profiles
5	Process Improvement Data	• Appraisal and Audit (and Assessment) Results • Process Improvement Plans • Action Plans • Status Reports • Pilot Assets
6	Process Assets Library Data	• Problem Reports • Enhancement Requests • Feedback from Process Users

Attributes of the Organisation's Standard Software Process Assets

The organisation's standard software process should be used extensively by all project managers as source for process information and tailoring guidelines that will help them define their project's defined software process. It should save them time and effort since the standard process descriptions provided will help projects "jump start" the definitions of their project-specific processes. For example the effort required for tailoring a standard configuration management process is much less than the effort required for creating a new document from scratch. The organisation's process database plays another important role. It should be the primary mechanism for ensuring consistency of the processes used by the various project's across the organisation.

Table 9 lists some of the essential attributes that should be considered when designing the organisation's standard process database.

Table 9 Attributes for the Organisation's Standard Process

Infrastructure Component	Attributes
Process Definition & Documentation	scope, format, media, coverage, availability, dissemination mechanisms, friendliness, ownership, how up-to-date, change control, feedback, relevance, flexibility, security.-
Life Cycles	relevance, completeness, availability,

	flexibility, scalability, customisability, coverage.
Customisation Guidelines	coverage, flexibility, clarity, enforceability, authority, ownership, up-to-date (ness),
Library of Process Documents (feedback from projects)	coverage, relevance, comparability, conformance to standards, scalability ownership, security
Process Measurements (Database)	coverage & relevance, accuracy, dependability, conformance to standards, usability, dissemination, format,

Use of the Organisation's Standard Software Process Database

The organisation's standard process database is effectively the corporate memory for process experiences and learning. It is the main input to any process improvement plans or activities. When analysed, the information stored in the process database should enable its users perform the following functions:

- Show the trends in product quality. These could be indicators of what improvements or otherwise have occurred to the quality of the software products. The statistical trends could take the form of terms of number of defects, bugs, etc.)
- Show the trends in project performance (e.g. the variance between the original and actual estimates of time, cost and quality for the projects completed over a period of time)
- Show the trends in process performance. Examples of process performance indicators are the productivity per software engineers, productivity per project, deviations from the original estimates, effectiveness of specific processes. Every process should have a number of key performance indicators for measuring its performance. For example the key performance indicator for the requirements management process could be the number of changes in the original requirements and the ratio of accepted versus rejected changes.
- Point to the weak spots that need attention (for example candidate areas for improvement). This could be the result of analysing the figures and trends for the key process areas. A record of the analysis of these trends could present the proof of the organisation's software

capability. This in turn could lead to the organisation's winning competitive bidding for critical contracts that require evidence of the organisation's software capability.

Access Control

There is a need to control the access to the information on the corporate process database. Some of the questions that could be asked when choosing a corporate strategy for the use of the process database are:

- To whom do we want to disseminate the process model?
- Whom do we want to allow access to the process performance data, or process performance trends?
- How do we control access to the process performance data, and to the project's process documentation?
- What is the policy for restricting access to specific data in the process database? who sets this policy? and who enforces this policy?.

Use of Measurement Data

The measurement data stored in the organisation's standard database can be used in a number of ways (Humphrey 1969):

- Understanding: To learn about a particular item or process.
- Evaluation: To determine if a process, product, or activity meet acceptance criteria.
- Control: To set limits and goals for process activities.
- Prediction: To develop rate and trend parameters for project planning.

3.3 Maintaining the Organisation's Standard Software Process Database

The process database should be owned and maintained by the process group. The definition of the contents of the process database should be a collective and collaborative effort among the following:

- **The Corporate Process Group (SEPG):**
  - To define the process definitions (structure and levels of detail) that should be recorded into the process database.
- **Software Process Improvement Teams (Software PITs):**
  - To define the metrics to be collected on the process performance and product quality.
- **Steering Committee:**
  - To determine policies on what the database will contain, and how it will be used. For example who are the authorised users, and what are the access constraints.)
- **Project Managers:**
  - To submit feedback on the performance of their project's defined process and on the resulting quality of the product.
  - To submit project document related to the software process that can be of value to future projects.

Role of the Public Process Models and Standards

Public process models and standards, such as the CMM, ISO 9001, ISO/IEC 15504, and MIL-STD-498, can be used as role models and guidelines when designing your organisation's

standards process. Usually such models and standards have associated with them a defined set of tailoring guidelines as depicted in Figure 9.

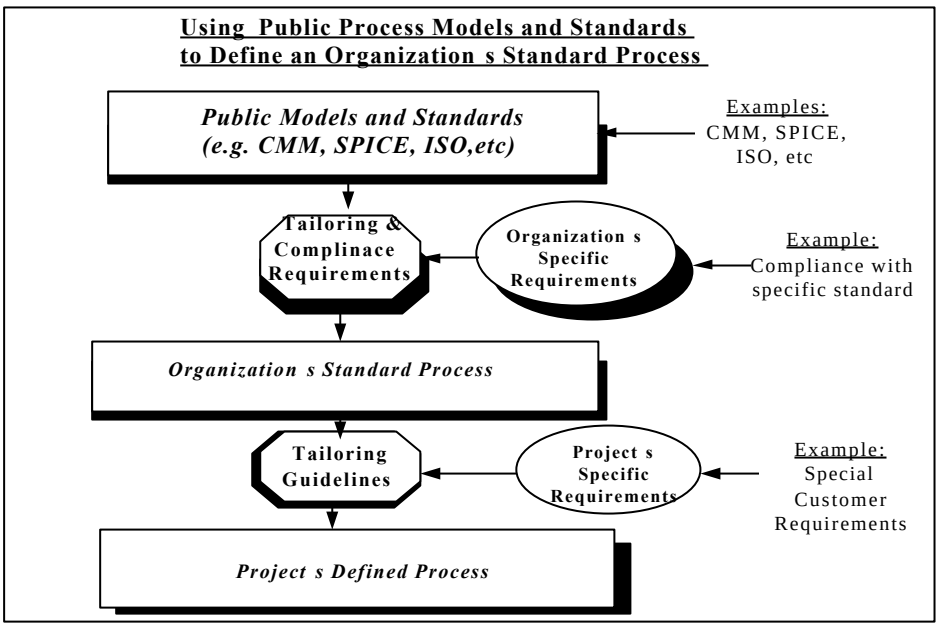


Figure 9 Using Public Process Models and Standards to Develop the Organization's Standard Process

In some cases, an organization may be required to illustrate compliance with specific standard. Such a case will define which standard should be used as the basis for defining the organization's standard process. A similar tailoring process takes place when project managers specify the project's defined process.

3.4 Process Support Tools

Functions of Process Support Tools

The architecture described above represents a conceptual model for an organisation's process assets. It needs physical implementation to make it happen. The physical implementation of the infrastructure usually takes the form of technology and tools that should provide the following:

- Facilities to enable the collection, storage and retrieval of data in the process database. For example a database management system and typical facilities usually associated with a DBMS. These could cover access control, physical distribution, integrity checks, or more sophisticated process modelling tools that enable the storage and presentation of the process architecture and process elements.
- Facilities to enable the distribution and presentation of the process database contents for authorised users. This could be assisted by existing communication network plus the database management system and its associated software facilities.
- Facilities to enable the selection, retrieval and summary of particular classes of data stored in the process database. These could take the form of management reporting, data analysis, decision support systems and management support systems.

- Facilities to enable the storage/archiving and retrieval of the library of software process documentation. Examples of these facilities include electronic document management and imaging systems.

Types of Process Support Tools

These are tools that will provide support the process groups and process users in fulfilling their roles and responsibilities. Figure 10 illustrates possible types of tools. It is likely that some of these tools could already be part of the overall IT infrastructure serving other parts of the organisation.

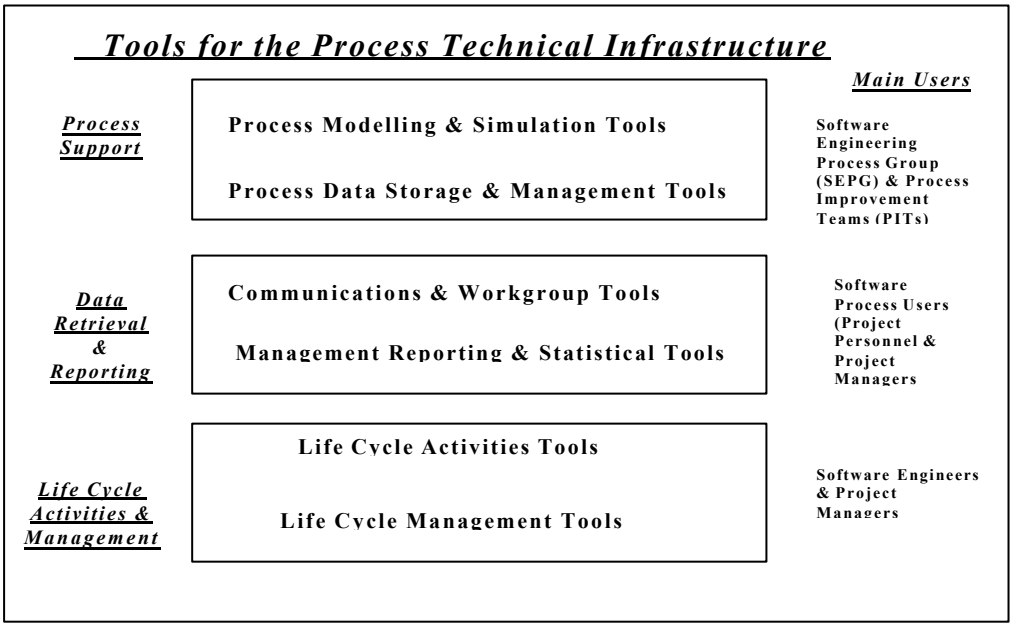


Figure 10 Process Support Tools

Types of process support tools include:

1) Tools for storing and managing process definitions and process data:

- **Process Modelling and Simulation Tools:** to enable the storage and retrieval of graphical presentation of the process architecture and process elements.
- **Process Data Storage and Management Tools:** These could be database management systems: for storing and managing the process definitions and process data.

2) Tools for retrieval and distribution of the process definitions and process data:

- **Communications and Workgroup Tools:** for managing access and distribution of the process database contents.
- **Management Reporting and Statistical Tools:** for presenting summaries and trends of the information stored in the corporate process database, and assist in further analysis and decision making.

3) Tools to support the software activities and process management activities:

- **Life Cycle Activities Tools:** These are tools that support life cycle activities. They are usually dedicated to supporting one or more phases of the life cycle. They should be integrated with other life cycle tools. There are some integration tools that act as middleware that passes information from one tool to another. Examples of these tools are:

  Requirements Management Tools

Systems Analysis Tools
Systems Design Tools
Code Generation Tools
Testing Tools
Integration Tools

- **Life Cycle Management Tools:** These are tools that support the management activities of the software development life cycle. Examples of these tools are:

  Project Planning and Tracking Tools
  Configuration Management Tools
  Contract financial management including project personnel time recording, tracking and optimising resource utilisation.
  Skill selection to assist project managers select the right resources that are viable and have the skills that match the project requirements.

3.5 Process Enforcement Mechanisms

Process enforcement is the mechanism that prevents the process from being ignored, and prevents it from turning into an ineffective "shelfware". Assume that you had a team of disciplined professionals in one project that followed the process. Without enforcement there is no guarantee the discipline across all the projects, or across generations of workers on the same project. An interesting view is to look at organisations as a set of permanent processes while employees are transient phenomena that pass through these processes to serve and enable them.

Enforcement procedures and mechanisms will ensure the effectiveness of the process. Process enforcement is particularly important at the initial stages of introducing a new process or changes to a new process. A new process or changes in current process mainly involve changing people's behaviours. This can prove difficult for some individuals especially at the initial stages of introducing such change, hence the need for enforcement mechanisms in order to ensure that the new process is followed.
The type and extent of the enforcement mechanism vary according to the nature of the new process or the new changes in the process. In some cases, the effective enforcement could take the form of organisational procedures and management practices to monitor compliance with the process. In some other cases, the enforcement could be applied through automated tools. (Example: enforcing the traffic laws used to be mainly through traffic police, but in some places speed cameras are used to monitor compliance with the law.)

Automated tools on their own are not enough. Rather they have to be part of management procedures that will define the activities for monitoring the feedback resulting from the enforcement tools. A process enforcement tool can also be used to capture and record violations of the process, and can collect statistics that will that will show the trends of conformance with the process. Process Assessors/Auditors can refer to such statistics as an evidence of the process discipline or otherwise.

4 AN EXAMPLE OF A SOFTWARE PROCESS SUPPORT INFRASTRUCTURE

4.1 The Example: Configuration Management (CM) Process

Let us take the Configuration Management (CM) Process as an example process area. The scenario assumes the existence of a number of software process improvement teams (software PITs), one of which is a team which focuses on defining and improving the configuration management process across all projects within the organization. What are the components of the infrastructure that need to be in place to support such CM PIT, and to support CM activities within the projects.. The CM support infrastructure at the three levels of corporate, project and individual levels should cover:

Organisational and Management Infrastructure:

- This covers the roles and responsibilities for sponsoring, managing, co-ordinating , defining and improving the CM processes followed across the organisation and within the projects.

Technical Infrastructure:

- This covers the CM tools, CM libraries, communication and documentation facilities used by the CM team, plus
- Any other technical facility that support the CM process team and their activities.

4.2 Organisational and Management Infrastructure for the CM Process

For each of the organisational levels (e.g. Corporate, Project, and personal), CM process-related procedures for the following roles should be defined and responsibilities allocated. Some of these roles are relevant only to the Corporate and Project levels.

- **Sponsorship Role:** A senior executive should ensure continued sponsorship of the CM process team.
- **Management Role.** A CM Process team leader should be appointed to manage the activities of the CM process team.
- **Co-ordination Role:** The CM process team should allocate responsibility for co-ordinating

Process Improvement Role: The CM process team should devise, plan and implement process improvement actions to improve the effectiveness of the CM process as it used by the projects and individual software engineers.

Table 10 summarises the possible roles and responsibilities at the different levels.

Table 10 Possible roles for CM Process-support

Organisational Level *Process-support Role*	**Corporate Level**	**Project Level**	**Personal Level**
Sponsorship	Corporate Quality Manager	Programme Director	Project Manager
Management	Corporate CM Steering Committee	Project Control Board	Project CM Co-ordinator
Co-ordination	Corporate Software Engineering Process Group	Project CM Co-ordinator	Project CM Co-ordinator

Process Improvement	Corporate CM Process Improvement Team	Project Software Engineers	Project Software Engineers

The functions for the roles at the different levels could be defined as follows:

- **At corporate level**: the following functions need to be fulfilled (by the corporate software engineering process group "SEPG")

    - **Defining:** Corporate Guidelines and Procedures for CM,
    - **Allocating:** the responsibility of these guidelines to a corporate resource or function to own these guidelines (usually this will be part of the corporate software engineering process group (SEPG),
    - **Organising:** CM training and awareness events across the organisation,
    - **Drawing up:** corporate guidelines and recommendation for CM tool selection and acquisition
    - **Encouraging and collecting:** feedback and metrics on the usability and effectiveness of the corporate CM guidelines
    - **Analysing:** the feedback and monitor the effectiveness of the corporate CM guidelines
    - **Planning and implementing:** CM process improvement actions, based on the feedback from the users of the corporate CM guidelines and based on monitoring the changes in CM standards and technology

- **At the project level:** the following functions need to be fulfilled (by the project manager or the process-support roles for the project)

    - **Defining and documenting:** the project CM plan
    - **Allocating the responsibility:** of CM activities to a CM project resource or function
    - **Organising:** CM training and awareness events for members of the project.
    - **Selecting and acquiring:** a CM tool for the project (according to the corporate guidelines and recommendation for CM tool selection and acquisition)
    - **Documenting and analysing:** feedback data from project members on the effectiveness of the project's CM process
    - **Planning and implementing:** CM process improvement actions, based on the feedback from the users of the corporate CM guidelines and based on monitoring the changes in CM standards and technology
    - **Providing:** feedback and metrics on the usability and effectiveness of the project's defined CM process and the corporate CM guidelines to the corporate process group (SEPG).

- **At the personal level:** the following functions need to be fulfilled:

    - **Defining:** a personal CM plan

- to receive training on the CM tool selected for the project
- **Following:** and adhere to the project's CM process and change procedures
- **Collecting:** metrics and feedback information on the effectiveness of both the CM tool and procedures, and submit feedback to the project manager and the corporate process group
- **Analysing:** personal feedback data and take actions for improvements as necessary.

Table 11 summarises the components of the process infrastructure to process a configuration management process at the different organisational level.

Table 11 Components of the Process Technical Infrastructure to support a Configuration Management Process

Component	Corporate Level	Project Level	Personal Level
Process Definition	Corporate Guidelines & Procedures on CM	Project Configuration Management (CM) Plan	Personal CM Guidelines
Process Ownership	CM Process Team (Part of the Software Engineering Process Group (SEPG))	Project Manager or a Project CM Responsible (e.g. a dedicated CM Team for the project)	Individual Software Engineer
Process Training	Training / Awareness of Project Managers and Others in the Corporate CM Process and in Tailoring Guidelines	Training of Project Personnel in the Project's Configuration Management Plan, Standards, Procedures, and Tool	Training of Individual Software Engineers in Configuration Management Procedures and Tool
Process Tools	Guidelines to Project Managers on the Selection and Acquisition of Configuration Management Tools	Selection and Acquisition of a Configuration Management Tool for the Project	Use of the Available Configuration Management Tool
Feedback from Process Users	CM Team (Part of the SEPG) seeks Feedback from Project Managers and Project Personnel on the Usability of the Corporate Guidelines on Configuration Management	Project Manager seeks Feedback from Project Personnel on the Effectiveness of the Project's Configuration Management Plans, Procedures, and Tools, and feeds the feedback to the Corporate SEPG	Software Engineers collects and provide Feedback on the Effectiveness and usability of the Project's Configuration Management Plan, Procedures, and Tools, and feeds the feedback to the Project Manager or directly to the Corporate SEPG
Measurements of the Process Results	Corporate CM Team (part of the Corporate SEPG) collect	Project Managers collect measurements on the Effectiveness of the Project's	Individual Software Engineers collect metrics on the

	measurements on the Effectiveness of the Organisation's Standards and Guidelines on Configuration Management and records them in the Organisation's Process Database	Configuration Management Plan, Procedure, and Tool, and feeds them back to Corporate CM Process Team (Part of the Corporate SEPG)	effectiveness and performance of the personal CM process and feed measurements back to the Project Manager or directly to the Corporate CM Team (Part of the SEPG)
Process Improvement Roles and Responsibilities	Corporate CM Team (Part of the SEPG) analyse the feedback from Project Managers and Software Engineers, and studies advances in CM tools and Standards, and tune the Corporate CM guidelines to improve them	Project Manager (or Project's CM responsible) analyses the feedback from Project Personnel on modify the Project's CM Plan and Procedures to improve their effectiveness, and/or upgrade the Project's CM Tool to improve its usability, or refresh CM training for the Project Personnel	Software Engineers analyse the feedback on the usability of the CM tools and Procedures, and provide suggestions for improvement or refresh his/her own CM training

5 MAKING THE INFRASTRUCTURE EFFECTIVE

5.1 Coverage of all Organisational Levels

In order to ensure the effectiveness of the software process infrastructure at all levels, it is useful to start with a list of the required functions and attributes for each level. These attributes should reflect the organisation's overall business requirements and should be taken into consideration when specifying the infrastructure components at each levels. To ensure that all such attributes has been covered, the list of attributes could take a tabular format similar to the one shown in Table 12.

Table 12 An example of a matrix listing of the functions/attributes of the process infrastructure at the various organisational levels

Infrastructure Function/Attribute	Corporate-Level	Project-Level	Personal-Level
Scope of the Process to be Defined			
Boundaries and Coverage			
Accessibility and Usability			
Process Support Functions			
Consistency with other Levels			
Feedback and Metrics Collection Mechanisms			

Automation of Process Activities			
Automation of Process Management Activities			
Supporting Tools and Technology			

5.2 Process Infrastructure and the Learning Organisation

The architecture described so far in support of the software process, also facilitates the creation of a "learning organisation" culture amongst software engineers. This is mainly because the process infrastructure supports the following learning mechanisms at both the corporate level and the project level:

Repository of Experiences

The organisation process components represents a repository of the feedback from the different projects on their process experiences. This is in effect a repository of knowledge that can be shared across the projects, hence the process learning of one project are made available for future projects.

Corporate Memory

The organisation process database together with the library of process related documents represent the corporate memory which stores the lessons and experiences and makes them available for future projects.

Learning Mechanism

The main learning mechanism is the feedback or project's experience and how much will project managers and project personnel will make use of this feedback to learn the relevant lessons and increase the chances of success for their projects.

5.3 Measuring the Effectiveness of the Process Infrastructure

In business environment, effectiveness is measured through the impact on the business results. As discussed in Part 1, the effectiveness of the process is ultimately measured through monitoring its impact on the business results. The effectiveness of the process infrastructure is measured through monitoring its impact on the process effectiveness. The following questions and answers could provide indicators of how effective the process infrastructure is:

- **Question:** Does the process infrastructure enable or obstruct the process activities (software engineering and project management activities) ?
- **Answer:** An example: Sometimes the use of a tool could be so complicated that manually performing the task is more effective. The tool turns from an enabler to a "disabler" of the process activities. I can think of some of the old style time recording tools that were used in some projects, and used to waste valuable time and effort of software engineers. They hated it, did not believe in the value of the data they entered. They never received a feedback on how this data were used.)
- **Question:** Is the method of retrieving and presenting process-related data to users and managers user-friendly and reliable or is it complicated and unreliable ?

- **Answer:** An example: Some data retrieval tools are inflexible and out of date, are rigid and not responsive to user needs, are text based and do not support graphical interfaces. In other cases tools operate only in batch mode rather than on-line mode. (example a configuration management tool that does not support on-line queries on the status of configuration items).

5.4 Attributes of an Effective Process Infrastructure

i) Attributes of an Effective Organisational & Management Infrastructure

An effective process technical infrastructure should be completely aligned with the requirements of the process it supports and its users. It should provide the following:

- Support of the roles and activities at the corresponding level of the organisation (e.g. corporate, multinational, national, divisional, project, personal)
- Adaptability to changes in the business structure (e.g. to cope with mergers between functional groups, departments, or companies)
- Support of all the process roles and responsibilities shown in the model for effective software process environment (examples roles include: process owners, project managers, business managers, process users, software engineers, process groups)
- Explicit assignment of the high level sponsorship and leadership role of the process efforts in the organisation
- Flexibility to allow for sharing of roles and responsibilities across functions (for example project personnel should be made available to participate in the process improvement activities)

ii) Attributes of an Effective Process Technical Infrastructure

An effective process technical infrastructure should satisfy the requirements of the process it supports and its users. It should provide the following:

- Support the storage and retrieval of the organisation's process definitions and data.
- Support of process flexibility (e.g. process changes to accommodate new methods, new techniques, or new technologies)
- Support of the communication and feedback mechanisms shown in the model proposed for effective software process environment (examples include: feedback from, and communicating process data to process users, project managers, business managers, etc.)
- Coverage across the organisation's physical distribution (e.g. multi-location, trans-national, global, etc.)
- Flexibility and ability to adapt to major any changes in the organisation's business strategy or geographical distribution

6 SUMMARY

The need for infrastructure to support the process

The process cannot be sustained or improved without an effective support infrastructure - imagine trying to enforce the traffic laws without having a working traffic lights, or without good roads, or without up-to-date read signs.

A process without supporting infrastructure is a "vapour ware"

In the absence of a supporting infrastructure the following symptoms will be noticeable: roles not defined, ownership is vague, tools are non-existent, no follow-up procedures, no feedback mechanisms etc.. Would you call this an effective process ?.

The process infrastructure should cover organisational and technical aspects

The process support infrastructure should cover organisational roles and responsibilities, as well as the tools and technical resources dedicated for supporting the process users, process owners and process "guardians".

The Infrastructure should extend its support to all the organisational levels

The process support infrastructure should support the various organisational levels in the organisation. It should support the corporate software engineering process group (SEPG), the project managers, and the project-level process-support resources as well as the software engineers:

The Infrastructure should allow flexibility

The process support infrastructure should be flexible to:

- cover different sizes of projects
- cover different project requirements
- adapt to varying customer needs
- adapt to varying standards requirements)

Need for Organisational and Management Infrastructure

- Process discipline can not be sustained without assigned and dedicated roles and responsibilities.
- Without a process support infrastructure, software process improvement could be no more than just a lip service by management.
- Without committed resources, everyone will be busy doing their day-to-day activities with no one looking after the process.

Need for Executive Sponsorship

- Without executive sponsorship and organisational commitment, process improvement can not survive organisational changes.
- Without continuing executive sponsorship, software process improvement resources are likely to be reassigned to project work at the first sign of resource shortage for the projects.
- Starting software process improvement without management sponsorship is often a false start. It will be very likely shelved at the next business reorganisation or downsizing. False starts can damage the morale of the software engineering staff.

Need for a Dedicated Software Engineering Process Group (SEPG)

- SEPG should be staffed by full time professionals and they should mainly take a role of co-ordinators versus doing the work?. (They should not try to be jack of all trades.) They should co-ordinate the activities of the software PITs.
- At what level does the SEPG operate? Generally the SEPG should operate at the corporate level versus divisional level. This will depend on the number of organisational levels in the organisation.

Need for Software Process Improvement Teams (Software PITs)

- Software engineering practitioners, working within development projects, should form the bulk of the software process improvement teams (Software PITs). There are several benefits that can be gained. Example benefits are getting their buy-in into software process improvement and to get their specialised knowledge and experience through participation in process design and specification activities.
- The selection of members of the various Software PITs depends on the nature of the process assigned to the team concerned. For example requirements' management PIT should have representation for the customer, the project manager, the legal and contracts department, etc.

Software Process Change Enablers

- There are a number of enablers for software process change in an organisation. The main types of enablers are: organisational enablers, cultural enablers, and human resources enablers.
- There are risks to the software process improvement if the process change enablers are not in place.
- The availability and effectiveness of these enablers will ensure those software process improvement behaviours and practices are "engrained" into the texture of the organisation. Continuous process improvement will be the normal way of doing business.

Requirements of a Technical Infrastructure to support the Software Process

- Without an adequate technical infrastructure, a process is not likely to achieve the necessary levels of effectiveness and efficiency.
- The process technical infrastructure should satisfy the requirements of the SEPG, the project managers, the software engineers, and the process improvement teams
- The process technical infrastructure should match the requirements of all the organizational levels associated with the process, the corporate SEPG, the project managers, the software process improvement teams and the individual software engineers.

Need for Tools to Support the Organisation's Software Process Assets

- The technical infrastructure is necessary for facilitating the task of the SEPG and enabling other process-related activities.
- The organisation's software process assets are the primary mechanism that will assure consistency of the processes used by the projects across the organisation.
- The organisation's standard process database represents the corporate process memory. It will have the process lessons and experiences of the software projects across the organisation, that will be used for improving the process.
- Tools are not substitute of the organisation and management infrastructure, which defines how the tools will be used, and by whom, and for what purpose.

Process Support Tools

Possible types of process support tools include:

- Tools for Process Modelling and Simulation,
- Tools for Storing Process Documentation,
- Tools for the Dissemination of the Process Definition,
- Tools for Measuring and Reporting the Process Results,
- Tools for Monitoring and Enforcing the Process,

These are in addition to Tools to support software engineering activities, such as configuration management tools, software quality assurance tools, etc.

REFERENCES

(Baumert 1996) John Baumert, Experiences Developing and Deploying a Corporate-wide Process Asset Library, in Proceedings of the SEPG 1996.

(Bennis 1989) Warren Bennis, On Becoming a Leader, qouted in Carol Kennedy's book "Managing with Gurus, Century Business Books, 1996.

(Crosby 1980) Philip Crosby, Quality is Free: The art of making quality certain, McGrow Hill, 1980.

(Davenport 1993) Thomas Davenport, Process Innovation: Reengineering Work through Information Technology, Harvard Business School Press, 1993.

(Dion 1993) R. Dion, Process Improvement and the Corporate Balance Sheet, IEEE Software, July 1993, pp. 28-35.

(Dorsey & McDonald 1996) Terri Dorsey & Diane McDonald, Structured for Success: A Software Engineering Process Group (SEPG) Model, Proceedings of the SEPG Conference, Atlantic City, New Jersey, May 20-23, 1996.

(Fowler & Rifkin1990) Priscilla Fowler and Stan Rifkin, Software Engineering Process Group Guide, Technical Report CMU/SEI-90-TR-24, Software Engineering Institute, Carnegie Mellon University, September 1990.

(Harrington 1991) H. James Harrington, Business Process Improvement, McGraw-Hill, 1991

(Humphrey 1997) Watts Humphrey, Introduction to the Personal Software Process, Addison Wesley, 1997.

(Humphrey 1995) Watts Humphrey, A Discipline for Software Engineering, Addison Wesley, 1995.

(Humphrey et al 1991) Watts Humphrey, Terry Snyder and Ronald Willis, Software Process Improvement at Hughes Aircraft, IEEE Software, 1991.

(Jones et al 1996) Lawrence Jones; Mark Kusanic; and Mike Ginn, Implementing the IDEAL Model in a Less Than Ideal World, in Proceedings SEPG 1996

(Kennedy 1996) Carol Kennedy; Managing with the Gurus, Century Business Book, 1996

(Myers 1996) Charles Myers, Ingredients of a Successful Improvement Effort, In Proceedings SEPG Conference, Atlantic City, May 20-23 1996.

(Paulk et al 1995) Mark Paulk, Charles Weber, Bill Curtis and Mary Beth Chrissis, 1995. The Capability Maturity Model - Guidelines for Improving the Software Process, Addison Wesley, 1995.

(Reed 1996) Russell Reed, Planning Software Improvement After Assessment, Proceedings of the SEPG Conference, Atlantic City, New Jersey, May 20-23, 1996

(Zahran 1996) Sami Zahran, Establishing a Systems Integration Process, Proceedings of the International Conference on Software Quality Management "SQM'96", Cambridge, UK, 1996.
(Also published in the SQM Journal, December 1996.

(Zahran 1998) Sami Zahran, Software Process Improvement: Practical Guidelines for Business Success, Published by Addison-Wesley, January 1998.

Acknowledgement:

This material includes extracts from Sami Zahran's Book titled:

"Software Process Improvement: Practical Guidelines for Business Success",

Published by Addison Wesley Longman in January 1998, ISBN 0-201-17782-X.