Software processes

Differences between processes

1. Flow of activities , action and tasks and interdependences
2. Degree to which tasks are defined
3. Degree of quality assurance
4. Degree to which work products are identified and required
5. Rigor and detail with which process is described

Differences between processes

6. Manner of project tracking and control
7. Degree of stakeholder involvement
8. Degree team organization and roles are prescribed
9. Level of autonomy of team

Compare this differences with process attributes?

Coping with change

- Change is inevitable

Types of changes

1. Business environment
2. Platform changes
3. New customer requirements

Change adds to costs as work need to be redone

Two ways of reducing costs of rework

1. Change avoidance – process has activities for anticipating change before it occurs - prototyping
2. Process designed so change can be accommodated at relatively low costs – incremental development

Compare this with approaches

- OS Deadlock avoidance and prevention
- Handing of errors by networks

1. Incremental development to get feed from customer. Changes can be made in next increment or small part of current increment is modified.

- Refactoring improving structure and organization of system , supports tolerance for change

Prototype problems

Prototype may not be used in same way as final system

Reasons not to deliver prototype as final system

1. May be impossible to fine tune prototype to meet non functional requirement
2. Prototype not documented
3. Changes to prototype likely to degrade its structure
4. Organization quality standards usually relaxed for prototype

Software process improvement

Processes need to improved to produce better quality software with tight deadlines

Two approaches to process improvement:

1. Process maturity approach – improving process, and project management and introducing good software engineering approaches in organization. Multiple levels

Software process improvement

2. Agile approach focuses on:

- Iterative development and reduction of overheads.
- Rapid delivery and response to changes.
- Focuses on code being delivered and minimize formality and documentation

Software process improvement

Deming introduced idea of statistical quality control which measures product defects and relates them to process. Analyzes the process and modifies so chances of introducing errors is reduced and defection detection is improved

Software process improvement

Products like software, other intellectual product like books and films where quality depends on design quality depends on four factors:

1. Development technology
2. Process quality
3. People quality
4. Cost , time and schedule

Inadequate processes or poor training can result in failure

- Problems of large projects are project management , communication and integration caused by multiple systems developed by multiple teams
- For small teams people quality is more important than development process and development technology is important.
- Inadequate budget, unrealistic delivery schedule the product quality will be affected

Process attributes

- Understandability – how easy it is easy to understand
- Standardization extent its based on generic standards and extend its used in different parts of organization
- Visibility does process produce clear results that are externally visible

Process attributes

- Measurability does process include data or activities that allow product or process characteristic to be measured
- Supportability extent process is supported by tools
- Acceptability – is process accepted by engineers developing the software product

Process attributes

- Reliability is process designed so errors are trapped avoided or trapped so that they don't result on product errors
- Robustness – can process continue inspire of errors

Process attributes

- Maintainability – can process change to organization requirements and identified process improvements
- How quickly can process of delivering system from given specification be completed

How do these compare with method modularity criteria