List of Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMM</td>
<td>Capability Maturity Model (SEI)</td>
</tr>
<tr>
<td>COCOMO</td>
<td>Constructive Cost Model</td>
</tr>
<tr>
<td>CRU</td>
<td>Conflict/Risk/Uncertainty</td>
</tr>
<tr>
<td>CSE</td>
<td>Center for Software Engineering (USC)</td>
</tr>
<tr>
<td>ESP</td>
<td>Evolutionary Spiral Process (SPC)</td>
</tr>
<tr>
<td>IKIWINI</td>
<td>I’ll Know It When I Need It</td>
</tr>
<tr>
<td>IKIWISI</td>
<td>I’ll Know It When I See It</td>
</tr>
<tr>
<td>ISI</td>
<td>Information Sciences Institute (USC)</td>
</tr>
<tr>
<td>NGPM</td>
<td>Next Generation Process Model</td>
</tr>
<tr>
<td>NGPSS</td>
<td>Next Generation Process Support System</td>
</tr>
<tr>
<td>OCC</td>
<td>Operations Control Center</td>
</tr>
<tr>
<td>POA</td>
<td>Point of Agreement</td>
</tr>
<tr>
<td>SEE</td>
<td>Software Engineering Environment</td>
</tr>
<tr>
<td>SGS</td>
<td>Satellite Ground Station</td>
</tr>
<tr>
<td>SMART</td>
<td>SynerVision/Matisse/Articulator (HP-USC)</td>
</tr>
</tbody>
</table>
Useful Resources

  • Boehm, B.W., et. al., “Next-Generation Software Processes and
  • Garg, P.K., et. al, “The SMART Approach for the Software Process

[2] SPC Evolutionary Spiral Process Guidebook (Volume 1-3), SPC-92079-

    Large Software Systems”, TRW-TS-90-01, TRW Technology Series, Jan
    1990.


  - includes TRW Ada Process Model, IBM-Houston process model,
    several others.
  • Over, J., “STARS/SEI Process Asset Library”, Lecture Notes for Los

[6] SEI CMM Reports.
  • Paulk, M.C., et. al., “Capability Maturity Model for Software”,
    CMU/SEI-91-TR024, ADA240603.

[7] Proceedings of IEEE Process Workshops (1-8) and Conferences (1-2),
    available through IEEE Computer Society Press.

[8] DoD-STD-498 (draft), 27 May 1994, available through CODSIA or USN-
    SPAWAR.
Workshop Agenda, List of Participants and the Paper, “A Collaborative Spiral Process Model Based on Theory W” (Boehm and Bose) are only available in the hardcopy version.

Please contact USC-CSE on how to obtain a hardcopy.

Phone: 213-740-5703

WWW: “http://sunset.usc.edu/Reports.html”
Appendix:

• Workshop Agenda
• List of Participants
• Paper, “A Collaborative Spiral Process Model Based on Theory W” (Boehm and Bose)
• List of Acronyms
• Pointers to Useful Resources
NGPM theory, the Theory-W extension to the Spiral Model. The agenda shown in the beginning of the summary can be described in the following NGPM/NGPSS steps:

(a) We first used the round-robin style to acquire win conditions (identified important issues) of each stakeholder (participants).

(b) We found that to develop a taxonomy (categorization) of win conditions was greatly helpful in understanding win conditions.

(c) The acquired win conditions were then understood and merged into Points of Agreement (POA). This was mainly fulfilled by merging statements referring to the same issue and generalizing some statements.

(d) We used discussion and voting as the negotiation strategy to prioritize the win conditions and to resolve the different opinions in order to reach a consensus on the most important issues.

4. Although the validation of the process that our group utilized is challenged during the (inter-)group discussion such as: what categories would suffice for classifying issues; whether voting really can determine the most important issues, it did offer an efficient way of providing well-organized and agreeable results. “Environment” is clearly specified. Important issues are identified. The categorization offered an easy way to understand issues addressed.

5. During the discussion, it was very often that we forgot the original semantics and rationale for an issue and we have to re-examine and/or re-define it. We also found that some statements were overloaded with several different aspects; whereas, some statements just expressed the same idea. Through this process, we realized that terminology, taxonomy and glossary would be unexpectedly helpful in understanding the context and reaching a consensus.
• Architecture of environment to support different objectives (definition, analysis...)

• Rationale for environment design decisions must be visible

• Environment design needs to take into account existing assets

• Highly configurable/partitionable

3.7 Other/General

• NEED TO DEFINE PROCESS AND ENVIRONMENT CONCURRENTLY (5)

• SUPPORT MULTIPLE PROCESSES (3)

• Need to consider an organization’s process maturity when introducing an environment

• Look at (benchmark) disciplines other than software for how their environments support their processes

4 Observation

1. During the process of determining the most important issues, we had different opinions on whether it was worth while to include an issue that obviously everybody recognizes its significance. An example is that we had an intensive discussion on whether to include the following issue in our final list: support heterogeneous platforms and integration of multi-vendor tools including homegrown tools. In the first round, it got the highest votes. However, no one voted for it in the next round because someone argued that the computer society already recognized its importance. After a thorough examination, our final decision was to maintain it in the list since it was still an important issue not fulfilled and required further work although people do identify its significance.

2. In the whole discussion, questions concerning “whether it is process-related or process-support-related” are often raised. Discussion helps us clarify the relationship between process itself and environment for support processes. We also identify the need of addressing support issues for the environment as well. That is how we came with the additional two categories: Support Desirable Processes and Environment Implementation.

3. The issue identifying and prioritizing process that we utilize is in fact an exercise of
3.4 Support Desirable Process

- SUPPORT ARCHITECTURE AS FOUNDATION OF PRODUCT-LINE SOFTWARE DEVELOPMENT
  (6)
  - Support risk management
    (2)
  - Support definition and use of win conditions

3.5 People-Oriented

- IMPROVE EFFICIENCY/EFFECTIVENESS OF PEOPLE USING ENVIRONMENT
  (6)
  - Support different roles
    (1)
  - Environment must not straight jacket you - granularity of process is a hard issue
    (1)
  - “Efficient” environment - fast response time
  - Environment must be non-intrusive enactment support
  - Easy to learn
  - Good visualization - graphical representation of information (and tabular form, etc.)
    to make it easy to understand information and friendly entry and acquisition

3.6 Environment Implementation

- SCALABLE FOR DIFFERENT SIZE PROJECTS - FOR DIFFERENT “SIZE” PROCESSES
  (4)

- SUPPORT MULTIPLE CONSISTENT VIEWS OF PROCESS AND PRODUCT DATA
  (3)

- SUPPORT HETEROGENEOUS PLATFORMS and INTEGRATION OF MULTIVENDOR TOOLS INCLUDING HOMEGROWN TOOLS

1. People in this group considered it very important and decided to add it to the list despite its order in the category
3.1 Definition and Evolution of Process

- ENVIRONMENT HELPS YOU DEFINE, TAILOR AND ENACT A PROCESS (8)
  - Generate process to build system from requirements for system - need tooling to help do it
  - Domain specific vocabulary for architectures

3.2 Execution of Process

- ENVIRONMENT MUST BE ABLE TO SUPPORT THE RICHNESS OF THE PROCESS FORMALISM (4)
  - Environment must support range of fidelity of enacting process (3)
    - Support domain specific design notations
    - Support geographically distributed and asynchronous collaboration including distributed information
    - Environment must be able to react quickly to process changes or get out of the way

3.3 Analysis of Process

- SUPPORT DATA COLLECTION FOR MONITORING AND MEASUREMENT (7)
  - Look back at historical use of process in order to be able to apply it on new project (see what it would be like to apply old process on new project through simulation, etc.) - requires process capture (1)
  - Generate schedule from process model (ala Articulator)
  - Facilitate capture of lessons learned from use of environment to support process
1 Meeting Agenda

1. defined “Environment” to support Next Generation Software Process (NGSP)

2. addressed important issues in the environment support of NGSP, with the focus on organizational processes, in a round-robin fashion

3. identified categories of issues according to the definition of Environment and other important aspects

4. assigned issues to the identified categories including
   • clarifying the semantics of each statement
   • merging statements that refer to the same issues
   • assigning finalized statements to appropriate categories
   • adjusting categories to better model the contents of finalized issues

5. voted to decide the most important issue(s) in each category

6. reviewed to resolve
   a. no consensus on the most important items within a category
   b. issues that should be pronounced regardless of its order in its category

2 Definition of Environment to Support NGSP

Automation that facilitates the definition, execution, analysis and evolution of organizational processes.

3 Issues Addressed

The issues are grouped into the following categories: Definition and Evolution of Process, Execution of Process, Analysis of Process, Support Desirable Process, People-Oriented, Environment Implementation, and Other/General. In each category, issue(s) caught the most attention is(are) shown in boldface in the following sections. Number in the parenthesis after each item tells us how many people in the group voted for it. An item without parenthesized number means that nobody voted for it.
Summary of the Environment/Specification Group Discussion

Leader: Dr. Arthur Pyster (SPC)

Participants: Ahmed Abd-Allah (USC/CSE)

Dr. Robert Balzer (USC/ISI)

Ms. Kirsten Blakemore (SPC)

Maj Ron Byrnes (US ARMY RESEARCH LAB.)

Mr. Richard Kaste (US ARMY RESEARCH LAB.)

Capt Roderick Taylor (US AIR FORCE ROME LAB.)

Mr. Doyl Tully (E-SYSTEMS, Inc.)

Scribe: Mingjune Lee (USC/CSE)

October 27-29, 1993
RDD-100, Virtual Software Factory, PCMS (ASD/SEE), Project Bridge Modeler, Articulator, Arcadia, USAF’s PDSS PMMTs etc. are new and immature.

(2) Right now we do not have much operational experience on process tools.

(3) The existing product tools are far more than the existing process tools.

(4) Software Engineer Environment (SEE) and Software Process Environment (SPE) are not integrated together.

New Agenda Issues

(1) Need to Integrate three environments together, that is integrate program management and control environment, system engineering environment, and process engineering environment together.

(2) The compatibility issues need to be addressed.

   All tools have impact on the process, but few tools are “open”, they are not designed to be integrated together. Also the proprietary tools problem need to be addressed.

(3) Functionality issues need to be addressed, as the tool functionality limitations may not accommodated your ideal process.
of people.
Is it for people to check with, to execute, or for problem shooting, learning, training?

(2) The feeling is the problem is not well understood.

New Agenda Issues
(1) Process representation must be on-line, multi-user and collocated
(2) Process representation must promote CPI

4. Process Enactment and Enforcement

New Agenda Issues
(1) Process enactment and enforcement should be non-intrusive, that is collecting information behinds the scenes, no “big brother” monitoring, no dictatorship.
(2) Process enactment and enforcement should be unobtrusive, that is help without hindrance, proper level of abstraction, ability to change process.
(3) Creativity that is focused on the problem at hand
(4) Process problem identification and improvement
(5) Resource control and process visibility

5. Analysis and Simulation

New Agenda Issues
(1) Process monitoring should be goal oriented measurement
(2) Need a common framework for evaluation of methods and procedures
(3) Measurement criteria (metrics) should be normalized
(4) Process engineers should be allowed to assess impacts of potential process improvements
(5) Process engineers should be allowed to assess impacts of potential technology improvements

6. Integration of Process Tools and Environment

The Problems
(1) The existing process tools such as SynerVision, Process Weaver,
(2) How to integrate product environment with process environment?
(3) Integration should be requirement oriented.
Try to satisfy users need is the goal of integration.
(4) Tools should be built with open architecture. It should provide the common base for future integration.
(5) Tools should be able to talk with each other in all kinds of level.

2. Community of Process Users

As process exists throughout the whole life cycle of the product, the user community of the process is widely spread. In an organization the process users include the corporate management people (making corporate policy), the general/functional management people (administrating the organization), project management people (estimating product cost, scheduling, and monitoring performance), customers, subcontractors, vendors, end users, system engineers, designers, testers, quality control people, business management people, marketing people, tool providers, and supporting people etc..

New Agenda Issues

(1) How to satisfy the need of this widely spread of user community
(2) What kind of environment is need to facilitate users with different motivations who require access to the process information
(3) WIN-WIN technology must support these diverse views

3. Process Representation and Manipulation

Currently there are a couple of ways to represent process. Among them text based representation (using templates on paper or on-line), graphical based representation (some kinds of graphical notations as used in CASE), relational process models (using RDBMS), and hypermedia based representation (for example interleaf) are most often used.

In order to represent the process properly, process attributes need to be identified. Basically for a process it need attributes to address the issues like activities, people/skills, work products, description, start and end triggers (entry and exit criteria), duration of the process (parameterized formulas), tools required, architecture/platform, regulations, lessons learned, and comments from users, managers and practitioners.

The Problems

(1) What are the real goals for process representation is not clear for a lot
1. Our Environment Today

Our software environment today consists of hundreds of tools. The biggest characteristic of this environment is great disparity, heterogeneity, and non-interoperability. The tools and software environment differ from organization to organization, and from project to project. They are for different purpose, on different platforms, and have different capability. Among those tools currently in use, some of them are for basic needs, some are production enhance oriented, some provide special capability, and some are for general use.

The Problems

In the real world there are a lot of so called “home grown” tools. They are usually built as an “addition duty” (quick and dirty), most of them are not for general purpose, instead they are built to meet specific project need. In general these kind of tools are lack of tailoring ability, not adaptable across project, they hardly live through from project to project, thus they do not evolve.

Realizing the importance of tool integration, some of the vendors have begun to provide loosely integrated tools and environments, such as Sun’s SparcWork, HP’s SoftBench and SynerVision, CADRE’s Teamwork, IDE’s Software Though Pictures, etc.. The problem of these kind of tools and environments are usually vendors claim of integration do not measure-up, and currently there are very few real users of these environments.

Another major trend of integrated environment is to build Software Engineering Environment (SEE) for the system developers. One of the major problems is there is a gap between system and software people, thus the current SEEs are usually too “computer science” oriented. In the real world more than eighty percent of the engineers need tools only to design, edit, compile, and test etc., they are not interested in cumbersome process/product logic representations (for example Petri Nets, LISP, SQL etc.), they need more user friendly representations. Also project management and project tracking need be integrated with SEE as well.

New Agenda Issues

The consensus of the discussion indicated that the major problem of the current environment is lack of meaningful integration of tools. The following issues have been mentioned as the new agenda issues which needed to be address in the future research:

(1) How to integrate tools together? What kind of framework do we need?
Summary of the Environment Group Discussion

Leader: Professor Ellis Horowitz (USC/CSE)

Participants: Mr. Peter G. Clark (TASC)

Ms. Stacey Gelman (AT&T Bell Labs)

Professor Lewis Johnson (USC/ISI)

Mr. Christos Kolonis (SAIC)

Ping Luo (USC/ISI)

Mr. Ray Madachy (LITTON Data Systems)

Mr. Jock Rader (HUGHES Aircraft Co.)

Scribe: Yimin Bao (USC/CSE)

October 27-29, 1993
Continuous Collaboration

Stakeholders

Collaboration Process

Dynamic List of Congruent goals/reqs/ expectations

Continuous Evolution of User/ Acquisition needs & constraints

Needs, Constraints, Process Feedback

System Requirements, Process Requirements

Process Requirements

Continuous Evolution of Process and Product

Process Feedback, Product Feedback, Status/Estimates

Community

Acquisition Process

Supplier Organization & Project Group

Product Engineering “Catchers Mit”
• TRANSITION PLAN from AS-IS to TO-BE

6 - Development process should accommodate sporadic requirements clusters. Maybe supporting parallel incremental builds

7 - Process design guidance and acceptance criteria should be included in the NGPM

• Recognize the need for a “step-function” in process engineering that could be able to handle any type of model (waterfall, spiral, etc.)
II. Conclusion

As a conclusion, the group reached a set of recommendations for process improvement based on a “people” viewpoint.

1 - Recognize there are separate, continuous process systems (see figure):
   - Acquisition
   - Collaboration
   - Development (or Product Engineering)

2 - Model/define acquisition process
   - Rationale:
     - We know a lot about development
     - We know something about collaboration
     - We know little about acquisition
   - Approach:
     - Model “as-is” acquisition process
     - Re-engineer acquisition process to identify and eliminate inefficiencies, redundancies, workflow gaps, value-subtracting steps, etc.

3 - Model/define “as-is” development and collaboration processes
   - Rationale:
     - Enables formalization, analysis, re-engineering, simulation, customization, etc.
     - Pay-off is short-term, but paves the way towards long-term process engineering and support system

4 - Model acquisition, collaboration, development, process interfaces, interconnections, and interactions. I.e., next generation process architecture.

5 - Model/define
   - AS-IS processes
   - TO-BE processes
12 - It is easier to merge organizations that have the same process, than to merge organizations and try to impose the process adopted by one of the original organizations into the other.

13 - Trajectory and guidelines are essential for people involved in the process.

14 - The process is supposed to be corporate memory.
   - Processes as corporate assets
   - “I know I can go in there and find an answer.”
   - It cannot be so large that you cannot find what you want. It needs some indexing.

15 - Go only as far as the money being invested is still paying off.

16 - The process must make the risks, costs, motives, goals and problems visible.

17 - Collaboration
   - A joint interaction with mutual accountability and dependence on the inputs, process, and output, wherein each member makes some contribution to achieve a more global achievement objectives. That is, congruency of goals among shareholders.
   - It is not coordination, cooperation, it is more like partnership, joint venture.

18 - There is no exact answer on when collaboration has been achieved, this varies from project to project.

19 - The process must recognize that requirements are going to be changing.

20 - Barriers to collaboration
   - “No ante, no play” (Do you have something to loose if the project fails?)
   - Goal incongruency among stakeholders
   - Lack of trust
   - Lack of sufficient incentives
   - Payoffs are not visible
   - Acquisition constraints (budget lead time)
   - Uncertainty in long term direction

21 - There is a need for tools to support elicitation.
1 - How to make the process more predictable? (understandable/repeatable process)

2 - Win-win is really what should be searched for. Is it happening?

3 - How do you manage people’s personal risk?

4 - Things are starting to change because upper level management is realizing that profits are reducing.

5 - If we don’t nail the requirements down we are headed for trouble.

6 - Would like to have better forecasting. Reward people that come up with things that are good for the company, although it could hurt their own personal interests.

7 - Give the customer choices, giving him/her the option of evaluating half the way through,. I.e., collaboration along the way.

8 - Resistance to change

   • Change and improvement are not synonyms.
   • How do you get people in an organization to “buy in” a new process?
     What kind of incentives do they have?
     Technology itself is not enough, career contingencies
     What anchors are holding them back?
     Career contingencies
   • Career contingencies (learning curve effects)
     Moving to a new process decreases productivity, confidence, and certainty until competence is achieved.
     Perceived loss of power

9 - A process must not be static. It should allow feedback from the people involved, ultimately leading to a change or not.

10 - People kind of resist empowerment.

11 - The way a process is designed has a huge effect on whether people accept it or not.
Corporate memory

- Quality/Cost/Schedule Trades
- Process Discipline
  Enforcement
  Buy-in
  Training
  Resistance to change
  Perceived value
  Awareness
Enactment
Scope
Strategy
Flexibility
Tailoring

4 - Community

- Process Control
  Use of metrics
  Accountability
  Incentives/anchors
- Commercial Unique Issues
  Time to market
- Quality/Cost/Schedule Trades
- DOD Unique Issues
  Policies and standards
  Contracting process
  Collaboration
  Open communications
  Developer---acquirer
  Acquirer---user

II. Some More General Issues

This group also discussed some more general issues like:
Resistance
Managers
Developers
Buy-in
Perceived value
Perceived integrity of corporate Training
Flexibility
Resource application
Process tailoring
Decision making

• Process Control
  Motivation
  Use of metrics
  Accountability
  Incentives/anchors
  Insight
  Empowerment

3 - Organizational

• Collaboration
• Corporate Maturity
  Process predictability
• Process Technology
  Tools
  Assets
  Size
  Search time
  Guidelines

• Process Control
  Motivation
  Use of metrics
  Accountability
  Training
The approach used by this group was to first identify the several types of people involved in the process, and then discuss the several issues involved on the process, considering who is affected by them and how.

The constituencies have been determined to be at the individual, organization, project and community levels.

I. Issues Identified

The issues identified by this method were distributed as follows:

1 - Individual

- Identification of Stakeholders
  
  Identification criteria -> some stakeholders come in half-way though the process having a great effect on the final outcome.

- Resistance to change
  
  Perceived value
  Perceived loss of personal power
  Perceived loss of productivity

- Discipline

- Process control
  
  Accountability
  Empowerment
  Incentives/Anchors
  Motivation

2 - Project

- Collaboration

- Process Technology
  
  Tools
  Assets
  Search time
  Operability

- Process Discipline
  
  Enforcement
  Audit
Summary of the Process/People Group Discussion

Leader: Professor Chris Westland (USC/BA)

Participants: Mr. Greg Baker (SAIC)

Ms. Joan Bebb (TRW)

Dr. Rick Hefner (TRW)

Mr. Jim Over (SEI)

Dr. Ron Rudman (EDS)

Professor Walter Scacchi (USC/BA)

Mr. Sterling (NORTHROP Corporation)

Scribe: Cristina Gacek (USC/CSE)

October 27-29, 1993
This part of this proceeding is only available in the hardcopy version.

Please contact USC-CSE on how to obtain a hardcopy.

Phone: 213-740-5703

WWW: “http://sunset.usc.edu/Reports.html”
Summary of the Process Group Discussion

Leader: Dr. Dewayne Perry (AT&T Bell Labs)

Participants: Mr. Garry Brannum (NORTHROP Corporation)

Ms. Jean Lehmann (EDS Corp.)

Mr. Rich Lobsitz (TASC)

Mr. Ed McCamey (E-SYSTEMS, Inc)

Mr. John Shelton (LOCKHEED Corporation)

Professor Dave Wile (USC/ISI)

Scribe: Dr. Prasanta Bose (USC/CSE)

October 27-29, 1993
6. Discussion Group Results

This section presents the results of the four discussion groups:

• Process Group (led by Dewayne Perry, AT&T)
• Process/People Group (led by Chris Westland, USC)
• Environment Group (led by Ellis Horowitz, USC)
• Environment/Specification Group (led by Art Pyster, SPC)
Ron Rudman reported on some instrumented EDS experiments in collaborative software processes. This involved group performance of prioritization and action determination, with and without technology support and mid-course feedback. These provided insights which helped EDS determine operational modes for a collaboration-oriented set of tools.
Stacey Gelman cited lessons learned in the AT&T Silver Bullet Project. To achieve the objectives of interval reduction (70-75 weeks to 39 weeks) and 28% reduction in staff cost, Silver Bullet Processes were developed by a process-engineering team within Software Technology Center (STC), and Advanced Software Construction Center (ASCC) was set up to apply these processes to Operations Systems products on AT&T’s BaseWorX™ application platform. While the process-engineering team identifies technology, platforms, reusable assets, processes and support for ASCC, ASCC provides feedback through process data for STC to evaluate and improve processes. Gelman concluded that while process definition facilitates reuse with the technical foundation provided by platform, environments are also key to supporting the process and the foundation. She also pointed out another critical success factor: The Silver Bullet team was composed of new personnel who did not have to unlearn previous process habits.

James Over addressed his process evolution considerations for the NGPM/NGPSS. He pointed out that NGPM/NGPSS has captured many important principles such as Theory W and collaboration orientation. However, in order to achieve a long-term success, it is critical to NGPM/NGPSS to include capability for process management and continuous process improvement (CPI). Under the premise that the quality of a software system is governed by the quality of the process used to develop and evolve it, process management discipline and CPI help increase productivity and profit as well as customer satisfaction. Over suggested that NGPM should be engineered by a defined process and address process management principles and embedded process improvement mechanism. This implies that NGPSS should provide support for a manageable, measurable, and evolvable process.
el, Charette’s risk management approach, and SEI Capability Maturity Model (CMM). The ESP is described by five main steps: 1) understand context, 2) analyze risks, 3) plan development; 4) develop product, and 5) manage and plan. By repeating the five steps and using the knowledge gained in each cycle, it helps deliver a quality product within cost and schedule, and provide continual refinement of project plans. Blakemore concluded with near-term plans for ESP that include ESP for software intensive systems, enhanced risk support and engineering risk-reduction waterfall process.

Joan Bebb described TRW's Ada Process Model. It is an adaptation of the spiral model which had proven to be key to the Command Center Processing and Display System-Replacement (CCPDS-R) project’s success in developing over 300K lines of Ada source code executing in a distributed environment. This Ada Process Model enables evolutionary development with potential of continuous risk management, “software first” lifecycles, accelerated availability of useful capabilities, higher quality, change freedom, and substantially reduced lifecycle cost. Universal Network Architecture Services (UNAS) is evolved from its predecessor NAS to support the Ada Process Model. As it is recognized that all large complex software systems will suffer from design breakage due to early unknowns and require a lot of rework, UNAS combines strategies such as demonstration-based design, and architectural quality improvements through megaprogramming techniques to enhance the probability of delivering a quality product before deadline. The results showed that UNAS and the Ada Process Model produced superior results to the conventional approach in reusability, adaptability, reliability, producibility and performance.
5. Affiliates’ Presentations

Walt Scacchi of USC began by presenting an Articulator-based meta-environment (SMART) for supporting next generation process. The Articulator is a process-centered environment developed by the USC Software Factory Project. SMART integrates the HP Softbench and SynerVision tools, broadcast message server (BMS), team-based database and the Articulator to support process modeling, integration, instantiation, enactment, and management. Scacchi also proposed a distributed process infrastructure (DHT) for integrating distributed software repositories and processes. DHT provides a hypertext-based model that allows transparent access to remote objects and tools.

Garry Brannum discussed the Northrop experience with the Articulator process modeling notation. He demonstrated that process modeling can facilitate human understanding and communication. In applying the Articulator notation to process modeling, he concluded that it was successful in the aspects of rapid team consensus, automated graphic presentation, consistency checking, and formal process repository but insufficient in the aspects of people’s resistance to change and formal representation to capture narrative policies and plans.

Kirsten Blakemore presented the Evolutionary Spiral Process (ESP) developed by Software Productivity Consortium (SPC). The ESP model is a framework for integrating key management and development steps necessary for successfully producing a product. In order to accommodate today’s complex software-intensive development efforts, the ESP model establishes a unique combination of risk management, project management, product life cycle, and process engineering principles including Boehm’s spiral mod-
This part of this proceeding is only available in the hardcopy version.

Please contact USC-CSE on how to obtain a hardcopy.

Phone: 213-740-5703

WWW: “http://sunset.usc.edu/Reports.html”
4. Issue Summary and Copies of Issue Papers

This section presents:

- A list of the issue papers contributed by the workshop participants
- A summary of the major issue paper points, categorized in terms of process, representation, environment, and people/groupware factors
- Copies of the contributed issue papers
This part of this proceeding is only available in the hardcopy version.

Please contact USC-CSE on how to obtain a hardcopy.

Phone: 213-740-5703

WWW: “http://sunset.usc.edu/Reports.html”
In the next set of charts, Prasanta Bose presented experience with USC/CSE’s NGPSS-0 prototype and results from the corresponding bootstrap experiment. The NGPSS-0 prototype was based on Perceptronics’ CACE/PM. Bose demonstrated hypothesis of the experiment: the adequacy of NGPSS-0 for supporting the NGPM; the generality of win conditions on NGPSS for applying to other software engineering environments (SEE’s); and the generation of product and process specific objectives, constraints and alternatives by adding Theory W to Spiral Model. In this experiment, four USC/CSE researchers played roles as user, developer, customer and system engineer to specify win conditions on NGPM/NGPSS, identify conflicts, risks, and uncertainties (CRU’s), resolve conflicts and generate points of agreement (POA’s) for the NGPSS-1. The experiment results showed that NGPSS-0 language was inadequate for supporting win conditions and CRUs; however, the experiment itself was adequate in providing insights for the NGPM and valuable enhancement for the NGPSS. On the hypothesis of applying NGPSS-0 win conditions to other SEE’s, the results demonstrated the commonality of NGPSS-0 and TRW SEE win conditions. The results also showed the product and process oriented objectives, constraints and alternatives are generated from win conditions. Bose concluded that future work on NGPM/SS would be to develop a SEE for domain specific information architecture.
3. Workshop Objectives and Context

In the initial set of charts below, Barry Boehm overviewed the objectives of the workshop and its relation to the USC/CSE research program. This research program, depicted by his chart of the USC/CSE Research Roadmap, concurrently addresses three major software front-end issues: (1) collaborative next-generation process models (NGPM) and associated support systems (NGPSS) (2) software architecture that is currently focused on the domain of operations control center (OCC), and (3) cost modeling that reflects the emerging software approaches, processes and architectures of the next decade. This workshop is focused on NGPM/SS, with the objectives of assessing new directions in software life-cycle process, providing guidelines for affiliate activities, and stimulating USC-CSE/Affiliate collaboration in the software process area. Boehm’s charts showed the inter-relationship among CSE’s three research directions. The software architecture research is expected to provide better understanding of candidate application areas for NGPM/SS, such as Satellite Ground Station (SGS) and, more generally, OCC. The cost modeling research is expected to help NGPM/SS evaluate objectives and offer alternatives subject to economical constraints.
Notes: The Unexamined Collaboration Process is Unlikely to Improve

A final valuable insight from the workshop addresses the importance of instrumenting collaborative process support environments to provide the observational basis for improvement of both processes and support environments.

Our response to this and other workshop insights is included on the next page, which summarizes the major NGPM/SS-related needs we heard Affiliates express, and the new items we have added to our research agenda as a result. Some of these have already been implemented (e.g., COCOMO negotiation support in our February 1994 NGPSS-1 demonstration). Others are high-priority summer 1994 projects, such as instrumentation, spiral model coordination, visualization support, entry condition elaboration, and process-architecture integration. We have also included a recent paper on NGPSS-1 related process in the Appendix.
The Unexamined Collaboration Process Is Unlikely to Improve

- Collaboration support is poorly understood
- Need to characterize existing collaboration processes
- Need to instrument collaborative support systems
- Drive instrumentation by collaboration hypotheses
  - Effect on cycle time reduction
  - Fewer win-lose outcomes
  - Improved downstream collaboration
Notes: Use of Product/Process Duality

One of the advantages of considering software processes as software is that one can use this software product/process duality to identify candidate process solutions that have been developed for analogous product problems. Thus, one can consider the approaches developed for enabling interoperability of heterogeneous software products (open interfaces, information hiding, wrappers, mediators, etc.) as good candidates for enabling interoperability of heterogeneous software processes.
Example Use of Product/Process Duality:
Enabling interoperability of heterogeneous processes, using analogous product techniques

• Process architectures with open interfaces
  - Waterfall: documents; Spiral: risk resolution

• Information hiding: encapsulate sources of change
  - Product representation techniques/SDD; version proliferation/CM

• Process element wrappers and mediators
  - Incremental product acceptance process

• Process evolution planning
  - Context monitoring
  - Change anticipation and execution
Notes: No Process Is An Island

A significant discussion group conclusion was that a project’s software process needs to be synchronized with a number of interacting concurrent processes involving the various stakeholders. A related position paper, “No Process is an Island,” for a software process evolution workshop, is appended as a source of further elaboration of this topic.
No Process Is An Island

• Project processes need to synchronize with other stakeholder processes
  - Product line, personnel, management, procurement
• Lack of synchronization subverts process investments
  - User colocation with developer: SAC 465L
  - Process flexibility and progress payments: 2167A
• Synchronization needs to deal with process heterogeneity
  - Process analog of Conway's Law: Software process structure reflects corporate process structure
Notes: Collaboration Process Preconditions

The workshop also convinced us that a lot of work needs to be done before stakeholders sit down at a set of NGPSS workstations to define and resolve win conditions. First, the process will not succeed if the wrong people are involved, e.g., if a user representative is an unempowered researcher, an unaccountable gadfly, an unrepresentative policy wonk, etc. Second, the process will not succeed if the system engineers and stakeholders have not done a good deal of homework in defining and internalizing knowledge about the existing system, its environment, candidate solutions, etc., and in learning how to use a collaborative process support system.
Collaboration Process Preconditions

• Appropriate staffing of stakeholder representatives, system engineering function
  - Stakeholder representatives: empowered, accountable, representative, collaborative, knowledgeable
  - System engineers: understanding of stakeholder domains, collaboration management ability, analytic skills

• System engineer/stakeholder pre-work
  - Characterize existing system, related concerns
  - Identify, discuss candidate solutions
  - Clarify collaboration process objectives, approach
  - Hold stakeholder workshop
    >> Context-setting, teambuilding, support system training
Notes: “Stakeholders” Aren’t Homogeneous

Based on some initial observations, we were attempting to characterize some expectable stakeholder patterns (e.g., developers tend to be technology and implementation-oriented; customers tend to be risk-averse; users tend to be change-averse). The workshop and its participants gave us a better understanding of the wide range of stakeholder representatives who can become involved as representatives of their constituencies. Thus, we are now considering the characterizations above to be suggestive, but not to be strong assumptions. Further process implications are indicated; these are elaborated on the next page.
“Stakeholders” Aren’t Homogeneous

- May be some general stakeholder archetypes
  - Implementation-orientation, risk-aversion, change-aversion

- But don't assume collaborators represent homogeneous constituency
  - People represent themselves, projects, organizations, communities to varying degrees

- Collaboration process implications
  - Getting the right people involved
  - Empowerment and accountability
  - Incentives to collaborate
  - Teambuilding
Notes: Stakeholder Role Variations

We found the stakeholder model valuable in helping to clarify the different process contexts of our various Affiliates. The government and aerospace Affiliates operate in a “development contract” market sector, in which the customer developer, user, and maintainer come from four different organizations (e.g., Air Force Aeronautical Systems Center, Lockheed, Air Combat Command, and an Air Logistics Center for F-22 software).

Commercial Affiliates with in-house products such as AT&T, EDS, and HP perform all four roles within the same company, and generally within the same product line division. If these products are offered as stand-alone software products, then the software product consumers and their management become significant stakeholders. This disaggregation of stakeholder roles has helped us to understand the interplay of win conditions, and to avoid oversimplifying the process of win condition reconciliation.
# Stakeholder Role Variations

<table>
<thead>
<tr>
<th>Market Sector</th>
<th>Customer</th>
<th>Developer</th>
<th>User</th>
<th>Maintainer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commercial Product</td>
<td>Vendor, Consumer Mgmt.</td>
<td>Vendor</td>
<td>Consumer</td>
<td>Vendor</td>
</tr>
</tbody>
</table>
Notes: New Understandings

The following charts show our new understandings about the nature of collaborative, multi-stakeholder software processes at the end of the workshop that we did not understand so well at the beginning.

- Stakeholder Role Variations
- “Stakeholders” Aren’t Homogeneous
- Collaboration Process Preconditions
- No Process Is An island
- Example Use of Product/Process Duality
- The Unexamined Collaboration Process Is Unlikely to Improve
2. Highlights: New Understandings

- Stakeholder roles vary by market sector
- “Stakeholders” aren't homogeneous
  - Resulting collaboration process preconditions
- No process is an island
- Product/process duality provides helpful insights
- The unexamined collaboration process is unlikely to improve
EDS work on collaborative technology and group processes.

Section 6 presents the results of the four parallel Discussion Group deliberations on Process Context, Processes and People, Process Environments, and Process Environment/Specification. An Appendix provides the Workshop Agenda, List of Participants and a set of Pointers to Useful Resources.
Knowledge Summary of the Focused Workshop:
Next Generation Software Processes and Their Support

Mingjune Lee, Barry Boehm, Editors
USC Center for Software Engineering
October 27-29, 1993

1. Background and Overview

The USC Center for Software Engineering’s Focused Workshop on “Next Generation Software Processes and Their Support” was held on October 27-29, 1993, as part of USC-CSE’s Affiliates’ Program. Forty-two attendees exchanged ideas, presented research topics, and discussed issues in groups during the exposition, including top research staff from CSE’s 19 Affiliate organizations spanning across academia, government and industry.

Focused Workshop Process

6/93 -- NGSP paper provided to USC-CSE Affiliates
9/93 -- Affiliate parties provide issue papers
10/93 -- USC-CSE provide issue summary
10/27-29/93 -- 2.5 day workshop
  10/27 -- Process presentations by AT&T, EDS, Northrop, SEI, SPC, TRW and USC
  -- Issue comments by all parties
  10/28-29 -- Breakout groups: processes, process/people, environment, environment/specifications
5/94 -- Knowledge Summary

Knowledge Summary

The next section presents our perceived highlights of the workshop, in terms of new understandings and new directions for the USC-CSE software process research program. Section 3 presents the workshop objectives and context, in terms of USC-CSE presentations at the workshop and the original issue paper provided to the participants. Section 4 presents a summary of the resulting issue papers we received in response from the participants, followed by copies of their contributed issue papers.

Section 5 provides copies of the Affiliate presentation charts from the first day of the workshop. These include a pair of presentations on the USC/Hewlett-Packard SMART and Northrop’s use of it; the SPC Evolutionary Spiral Process; the TRW Ada Process Model; the AT&T Silver Bullet Process; the SEI Process Management approach; and the
TABLE OF CONTENTS

1. Background and Overview

2. Highlights: New Understandings

3. Workshop Objectives and Context

4. Issue Summary and Copies of Issue Papers

5. Affiliates’ Presentations

6. Discussion Group Results
   Process
   Process/People
   Environment
   Environment/Specifications Group

Appendix
   Workshop Agenda
   List of Participants
   Recent Paper,
      “A Collaborative Spiral Process Model Based on Theory W”
      (Boehm and Bose)
   Pointers to Useful Resources
   List of Acronyms
KNOWLEDGE SUMMARY

USC Center for Software Engineering

Focused Workshop on

Next Generation Software Processes and Their Support

Mingjune Lee and Barry Boehm, Editors
USC-ISI (Marina del Rey, CA)
October 27-29, 1993