How to Define CMMI Based Processes that are Short & Usable

NDIA 2005 CMMI Technology Conference
November 14, 2005

Tim Olson, President
Quality Improvement Consultants, Inc. (QIC)
(760) 804-1405
Tim.Olson@qic-inc.com
www.qic-inc.com

“I have made this letter longer than usual because I lack the time to make it shorter”

Blaise Pascal
Tutorial Objectives

Describe common problems with process documentation.

Discuss some people aspects of using process documentation.

Discuss how to define short and usable processes and procedures.

Provide some real examples.

Provide some lessons learned of “what works” and “what doesn’t work”.

Outline

Introduction

Defining Short, Usable Processes

Process Modeling Overview

Real Examples

Some Lessons Learned

Questions and Answers
Who is QIC?

The mission of Quality Improvement Consultants, Inc. (QIC) is to help organizations to measurably:

- become “best-in-class” or “world-class” quality leaders in their respective markets (e.g., using benchmarking)
- improve quality and productivity (e.g., lower product defect rates, increased KSLOC per person month, etc.)
- reduce the cost of poor quality (e.g., rework, waste, scrap, etc.)

Why Are You Here?
Tutorial Agenda

30 Minutes  Introduction
30 Minutes  Defining Short, Usable Processes
30 Minutes  Process Modeling Overview
30 Minutes  Break
30 Minutes  Real Examples
30 Minutes  Some Lessons Learned
30 Minutes  Questions and Answers

What Will You Walk Away With?

Motivation and knowledge of key process definition principles.

A better understanding of what short and usable processes are, and why you want them.

Examples that work in the real world.

A copy of this tutorial notebook, and a list of key references on process definition from the literature.
Outline

Introduction

Defining Short, Usable Processes

Process Modeling Overview

Real Examples

Some Lessons Learned

Questions and Answers

Why Short, Usable Processes?

SHELFWARE
Common Process Problems

**Too Big:** Processes become “Big Honkin’ Binders”.

**Poor Usability:** Not “fit for use” by process users. Many processes contain mixed information types.

**Poor Design:** Process documentation usually violates good definition and writing principles.

**Lack of Pictures:** Processes need to be defined by well thought out diagrams, pictures, or models.

**Defined Sequentially:** Processes are not novels.

**Shelfware or Unused Webware:** Unused processes.

Why Short, Usable Processes?

**Reduce Size:** Short processes are precise, concise, and more usable.

**Better Usability:** Short, usable processes require defining “chunks” and labeling them for use.

**Better Designs:** Short, usable processes require good process definition and writing principles.

**Use of Pictures:** Processes need to be defined by well thought out diagrams or “process models”.

**Defined Non-Linearly:** Processes are dynamic, parallel, and concurrent.
Information Mapping®

Information Mapping® is a structured, principle-based method for developing and documenting policies, standards, processes, and procedures.

Information Mapping® is based on seven principles and classifies all information into seven map types.


### Seven Principles

- Accessible detail
- Chunking
- Consistency
- Hierarchy
- Integrated graphics
- Labeling
- Relevance

### Seven Info-Maps®

- Classification Map
- Concept Map
- Fact Map
- Principle Map
- Procedure Map
- Process Map
- Structure Map

---

Some People Aspects

**People Perform Processes:** People process information using human behaviors (e.g., cognitive psychology), and people are not always logical.

**Chunking:** Information should grouped into small, manageable units (e.g., 7 plus or minus 2 chunks).

**Labeling:** A label should be used for each “chunk” of information (people like to find things quickly).

**Hierarchy:** Small, relevant units of information should be organized into a hierarchy and labeled.

People Aspects, Continued

Relevance: All the information in one chunk should relate to one main point based on that information’s purpose and function for the user.

Level of Detail: Information should be written at a level of detail that meets the users needs, and provides accessibility to more detail if the user needs it.

Consistency: Wording should be used consistently for similar subject matters, labels, formatting, etc.

Integrated Graphics: Diagrams, tables, models, etc., should be used as an integral part of the text.


Documentation Framework

- POLICIES “Laws” or “principles” that govern operations
- STANDARDS “Operational definitions” for work products
- PROCESSES “What happens over time” to build products
- PROCEDURES “How to” or step by step instructions
- TRAINING Provides the needed knowledge and skills
- TOOLS Supports and automates operations

*Slide adapted from “A Software Process Framework for the SEI Capability Maturity Model”, Olson, Timothy G., et al. CMU/SEI-94-HB-01
Key Process Questions

<table>
<thead>
<tr>
<th>Key Process Question</th>
<th>Process Element</th>
</tr>
</thead>
<tbody>
<tr>
<td>Why is the activity performed?</td>
<td>1. Purpose</td>
</tr>
<tr>
<td>Who does what activity?</td>
<td>2. Role(s)</td>
</tr>
<tr>
<td>What work products are used?</td>
<td>3. Input(s)</td>
</tr>
<tr>
<td>What work products are produced?</td>
<td>4. Output(s)</td>
</tr>
<tr>
<td>When does the activity begin?</td>
<td>5. Entry criteria</td>
</tr>
<tr>
<td>When does the activity end?</td>
<td>6. Exit criteria</td>
</tr>
<tr>
<td>Where is activity performed?</td>
<td>7. Context (e.g., hierarchy)</td>
</tr>
<tr>
<td>What activity is next?</td>
<td>8. Flow (e.g., sequence, selection)</td>
</tr>
<tr>
<td>How is the activity implemented?</td>
<td>9. Procedure</td>
</tr>
</tbody>
</table>

• Slide adapted from "A Software Process Framework for the SEI Capability Maturity Model", Olson, Timothy G., et al, CMU/SEI-94-HB-01

"Pictures" and "Words"

**Pictures**
- Pictures are worth a 1000 words, and most people are visually oriented.
- Organizations have reached high quality and process maturity (e.g., Baldrige Award, CMMI®/CMM® Maturity Level 5) with pictures and words.
- Pictures can be formalized into models.
- Pictures are necessary, but not sufficient.

**Words**
- Words fill in the needed details that pictures or models miss.
- Some people are textually oriented.
- Words are necessary, but not sufficient.
What is Expert Mode?

“Make three correct guesses consecutively and you will establish a reputation as an expert.” (Lawrence Peter)

Expert: “possessing special skill or knowledge; trained by practice; skillful or skilled.” (Websters)

“Expert Mode” is defining processes that:
• are as concise and precise as possible
• are at an appropriate level of detail for an expert
• are “chunked” so that an expert will use them
• use process models, checklists, forms, and tables
• are free from training material
• are free from beginner information
• have pointers to detailed explanations

Process Definition Modes

Beginner Mode:
• Assumes user has little or no experience
• Includes training material
• Includes process guidance and lessons learned

Intermediate Mode:
• Assumes user has some experience with process
• Includes process guidance and lessons learned

Expert Mode:
• Assumes user is very experienced
• Concise and precise as possible

All Modes should be “Chunked” and Include:
• Process models, checklists, forms, and tables
• Pointers to detailed explanations
QIC Process Measurement Framework SM

<table>
<thead>
<tr>
<th>GOALS</th>
<th>KEY QUESTIONS</th>
<th>METRICS</th>
<th>DC</th>
<th>DS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLAN</td>
<td></td>
<td>Cost, defects, effort, size, schedule, etc.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CONTROL</td>
<td></td>
<td>Cost, defects, effort, size, schedule, etc.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IMPROVE</td>
<td></td>
<td>Cost, defects, effort, size, schedule, etc.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- DC = Data Collection; DS = Data Storage

Outline

Introduction

Defining Short, Usable Processes

Process Modeling Overview

Real Examples

Some Lessons Learned

Questions and Answers
Why Model Processes?

“Understanding the process can be aided by laying out the process in a flow or logic diagram. Several types are helpful.” (Juran, Quality Control Handbook)

Informal processes are frequently misunderstood (e.g., unclear state transitions, concurrency, complexity, fuzzy pre and post conditions, etc.)

Process models facilitate communication and understanding of processes.

Process models can have automated support (e.g., tools, simulation, automatic code generation, etc.)

What is a Process Model?

Process Model:
- An abstraction of a process typically characterized by formal notations for representing roles, activities, and/or work products, and the relationships (e.g., events, transformations) among them

Types of process models:
- Descriptive (as-is): describes what is actually done
- Prescriptive (to-be): prescribes what to do (e.g., by new policies, standards, process guidelines, etc.)
- Mixed (both): most process models are a mixture of prescriptive and descriptive processes
What is a Process Model?

Key Process Questions

<table>
<thead>
<tr>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>Why is the process performed?</td>
</tr>
<tr>
<td>Who is involved in the process?</td>
</tr>
<tr>
<td>What work products are used?</td>
</tr>
<tr>
<td>What work products are produced?</td>
</tr>
<tr>
<td>When does the process begin?</td>
</tr>
<tr>
<td>When does the process end?</td>
</tr>
<tr>
<td>What activities are performed?</td>
</tr>
<tr>
<td>What roles are responsible for performing which activity?</td>
</tr>
<tr>
<td>How is the process implemented?</td>
</tr>
<tr>
<td>What activity is next?</td>
</tr>
<tr>
<td>Where is process performed?</td>
</tr>
</tbody>
</table>

“\( M \) models \( A \) if \( M \) answers questions about \( A \)” (Doug Ross)

Developing Process Models

The Process Modeling Process:

- **Input(s):** Process requirements, process expert knowledge, documentation, etc.
- **Output(s):** A process model that meets requirements
Process Modeling: As-Is vs. To-Be

- Slide adapted from "Defining Software Processes" Training, SEI

Process Building Blocks

An analysis of process representations and process models suggest three fundamental process building blocks:

**Work Products (results):** any product, service, or result (e.g., inputs, outputs, etc.)

**Activities (actions or tasks):** the actions taken to create or produce work products, services, or results.

**Roles (actors or agents):** the process performers that carry out activities to produce or create work products, services, or results.
Structured Analysis and Design Technique (SADT) is a graphical systems modeling language developed at Softech/MIT by Doug Ross in early 1970's.

Used extensively to document all manner of systems including manufacturing processes.

Architecture includes two major components:
- activities
- things

Some features are:
- decomposition of components is supported
- extensions add data and behavioral modeling
- automated support is available
SADT Components

Activities are represented by boxes:
• 3-7 boxes per diagram
• diagrams are hierarchical

Things are represented by arrows
• inputs: things transformed by the activity
• outputs: things into which inputs are transformed
• mechanism: how activities are realized
• control: constraints on an activity

SADT/IDEF Model Example

Example from SEI “Defining Software Processes” Training
ETVX Overview

Developed at IBM in the mid 1980's.

Process architecture includes four components:
- Entry criteria
- Task
- Validation
- Exit criteria

ETVX Template

Purpose:

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Entry</th>
<th>Tasks (Major)</th>
<th>Exit</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Roles:

Verification
Role/Flow Process Model

Appendix A: Customer Request Process

Purpose: To address customer requests effectively and efficiently.

Inputs/Entity Criteria
1. Customer request received (i.e., phone call, voicemail, email, fax, inter-office mail, walk-thru)
2. Remedy ticket created

7.1.1 Request to ITSC
7.1.2 Create Remedy Ticket (ITSC (T1))
7.1.3 Triage Request
7.1.4 Resolve Request?
7.1.5 T2 Work the Request
7.1.6 T1 Resolve Request (i.e., documented)
7.1.7 T2 Resolve Request?

Purpose: To address customer requests effectively and efficiently.

Examples of control flow:
- Sequence
- Selection
- Iteration
- Concurrency

Examples of data flow:
- Input
- Function
- Output

Example of timing flow:
- State Chart
Outline

Introduction

Defining Short, Usable Processes

Process Modeling Overview

Real Examples

Some Lessons Learned

Questions and Answers

Real Examples

Process Diagrams

Process Models

When/Then Tables

Checklists

Step/Action Tables

Forms

Process Guides
World-Class Quality

Inspection Process Model

- Work Product
  - Planning Stage
  - Preparation Stage
  - Meeting Stage
  - Rework Stage
  - Follow-Up Stage

Product Disposition:
- "A" - Accept
- "C" - Conditional
- "R" - Reinspect

Inspection Forms:
1. Meeting Notice
2. Defect List
3. Defect Summary
4. Summary Report


Planning Stage

Purpose: Organize and plan resources for inspection

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Entry</th>
<th>Task</th>
<th>eXit</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Final draft of work product&lt;br&gt;• Supporting materials for work product&lt;br&gt;• Inspection historical data</td>
<td>• Work product is completed (Final Draft)&lt;br&gt;• Work product meets entry criteria</td>
<td>1. Verify entry criteria&lt;br&gt;2. Select inspection team&lt;br&gt;3. Need an overview?&lt;br&gt;4. Schedule inspection&lt;br&gt;5. Complete and distribute inspection work package</td>
<td>• Inspection Team&lt;br&gt;Selected&lt;br&gt;&lt;Overview Conducted&gt; AND Inspection Scheduled&lt;br&gt;AND Inspection Work Pack. Distributed</td>
<td>• Inspection work package</td>
</tr>
</tbody>
</table>

Measurements:
- Preparation rate
- Inspection rate
- Defect density

Roles: Moderator; Author

### Overview Decision Table

<table>
<thead>
<tr>
<th>When...</th>
<th>Then...</th>
</tr>
</thead>
<tbody>
<tr>
<td>• The work product is large, or complex, or interrelates extensively</td>
<td>Schedule an Overview Meeting</td>
</tr>
<tr>
<td>with other work products.</td>
<td></td>
</tr>
<tr>
<td>• The work product represents a use of technology that is new or</td>
<td></td>
</tr>
<tr>
<td>infrequently used by the other inspectors.</td>
<td></td>
</tr>
<tr>
<td>• The work product is critical to the project, and sets direction for</td>
<td></td>
</tr>
<tr>
<td>subsequent work.</td>
<td></td>
</tr>
<tr>
<td>• There is a division of technical responsibility among the</td>
<td></td>
</tr>
<tr>
<td>inspectors, with a sharing of the role of reader because of</td>
<td></td>
</tr>
<tr>
<td>differences in inspector expertise.</td>
<td></td>
</tr>
<tr>
<td>• The inspectors are already familiar with the work product</td>
<td>Skip the Overview Stage, and proceed to the</td>
</tr>
<tr>
<td></td>
<td>Preparation Stage</td>
</tr>
</tbody>
</table>


---

### Example Checklist Questions

**Example Design Checklist:**

- **Completeness:** Does the design implement the requirements?

- **Traceability:** Does the design trace back to the requirements?

- **Interfaces:** Are the external interfaces correct? Are the interfaces between design components correct?
Forms

Forms are procedures (i.e., “how to”, step by step information) to support and implement processes.

Example Inspection Forms:
- Meeting Notice
- Defect List
- Defect Summary
- Meeting Summary

Forms should be:
- for one (or more) people to complete a repeatable activity
- hard-copy, automated, or both.
- instrumented in order to collect data.

Example Defect List Form

Software Inspection Defect List

<table>
<thead>
<tr>
<th>Project</th>
<th>Meeting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>Phase</td>
<td>Release</td>
</tr>
<tr>
<td>Component</td>
<td>Document</td>
</tr>
<tr>
<td>Preparation</td>
<td>Inspector</td>
</tr>
<tr>
<td>Time</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Location</th>
<th>Defect Description</th>
<th>Defect</th>
<th>Doc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Page</td>
<td>Section</td>
<td>Line</td>
<td>Type</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Copyright © 1990-1998 by Quality Improvement Consultants (QIC) & World-Class Quality (WQC)
Example Step/Action Table

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Inspector tracks his or her time while examining work product for defects.</td>
</tr>
<tr>
<td>2</td>
<td>Inspector uses the appropriate defect checklist to examine the work product for defects for the estimated time on the inspection meeting notice.</td>
</tr>
<tr>
<td>3</td>
<td>Inspector records defects on the Defect List according to the defect classification scheme.</td>
</tr>
<tr>
<td>4</td>
<td>Continue steps 1-3 until work product is completely inspected (record prep. time).</td>
</tr>
</tbody>
</table>


Used with Permission and Licensed by Quality Improvement Consultants, Inc. (QIC)
Outline

Introduction

Defining Short, Usable Processes

Process Modeling Overview

Real Examples

Some Lessons Learned

Questions and Answers

Some Lessons Learned - 1

The Documentation Framework helps organize your process documents.

Model based process guides are much more organized, precise, concise, and defect free.

Use a proven process to define your processes.

Applying these best-in-class process definition principles greatly enhances usability, and keeps your process documentation short.

Documentation “Modes” such as beginner, intermediate, and expert are useful.
Some Lessons Learned - 2

Process models will drive the structure of a process guide.

Information Mapping® is a helpful approach for writing procedures, and documenting process guides.

Make sure to address the people issues of process documentation.

Use “expert mode” to keep your documents short (e.g., process models, checklists, tables, etc).

No matter what you do, some people will always hate your processes and procedures.

Outline

Introduction

Defining Short, Usable Processes

Process Modeling Overview

Real Examples

Some Lessons Learned

Questions and Answers
References