The Improvement Kata

Annual Improvement Planning
Meets Agile
and Lives Happily Ever After

Hello, My Name is Kathy Iberle

• 30 years in technology
  • Software development
  • Quality engineering
  • Process improvement
  • Hardware and software
  • Big and small organizations

• Consulting
  • Lean process improvement
  • Agile at the edges
  • Software quality
Enough About Me...

What About You?

Agenda

• Process Improvement Methods Today
• What is the Improvement Kata?
• Scaling Up the Improvement Kata
• Adapting the Improvement Kata for Software
• Conclusions
• Q&A
Who Has Annual Improvement Plans?

Raise Your Hand!
Who is Doing Agile?

Raise Your Hand!

Plan  Collaborate  Deliver

Feedback

Notice any problems?
Different Assumptions

- Ability to identify and plan solution up front
- Need for coordination across teams and across time
- Best way to organize across teams

Annual Planning Assumes We Know the Path to Take
We Do Know – Up to a Point

Our Goal

Threshold of Knowledge

Today

And Then...

Our Goal

Today
Agile Retrospectives

Independent Changes

Coordinated Changes

Adapted from Mike Rother
Organizations Have Multiple Value Streams

**Value stream**: end-to-end delivery of value
Why Focus on the Value Stream?

How many jobs per day will get done?

15 jobs/day 17 jobs/day 10 jobs/day

Ask for changes
Development
System Test
Deployment
DONE

Good Process Improvement Method

Handles uncertainty:
Adapts to new information
Corrects course quickly
Doesn’t plan too far ahead

Scales up:
Can solve big problems
Coordinates change across entire organization
Good Process Improvement Method

**Handles uncertainty:**
- Adapts to new information
- Corrects course quickly
- Doesn’t plan too far ahead

**Scales up:**
- Can solve big problems
- Coordinates change across entire organization

**IMPROVEMENT KATA**

1. Understand the Direction or Challenge
2. Grasp the Current Condition
3. Establish the Next Target Condition
4. Iterate Toward the Target Condition

**IMPROVEMENT KATA - WHAT IS IT?**
Origins of the Improvement Kata

Dr. Mike Rother

Highly successful continuous improvement and adaptation

What & how: patterns for managing and thinking?

THE TOYOTA KATA RESEARCH

2004 - 2009

Guided by these two research questions:

1. What are the unseen managerial routines and thinking that lie behind Toyota’s success with continuous improvement and adaptation?
2. How can other companies develop similar routines and thinking in their organizations?

Mike Rother says:
If you study Toyota’s management system, you will see a common pattern of thinking and acting at all levels in the company.
What Did They Find?

Continuous improvement using:
• Go-and-See (Gemba)
• Small batches
• Cadence

AND
• Scientific Method (PDCA)
• Kata

Agile uses the same Lean principles.
Small batches: User stories
Cadence: Sprints
Go-and-See: User demos

What is a Kata?

Kata are practice routines that help us adopt new ways of acting and thinking
Counting: a Musician’s Kata

THE IMPROVEMENT KATA

1. Get the Direction or Challenge
2. Grasp the Current Condition
3. Establish your Next Target Condition
4. Conduct Experiments to get there

Adapted from Mike Rother
Start by Setting a Direction

What is our current challenge?
Six months to a year out.

Grasp the Current Condition

Science = measurements, not opinions!

How effective is our process now?
What are its results?
How does it work?
A Target Condition is a step on the way to the Challenge. It describes the future “bottom line”, not a specific solution.

Example: Early Adoption of Agile

Meet 60% of sprint commitments

Meet 80% by starting testing at midpoint

Fast, flexible, reliable

Matrioned people are yanked off

Testing Starts Late

Interruptions from outside

Adapted from Mike Rother
Without a Target Condition

• Arguing over what to change
• Process for its own sake

With a Target Condition

Which *one* obstacle are we addressing now?
A Kata: The Learner’s Storyboard

**Focus Process:** Development of new features

**Challenge:** Fast, flexible, reliable

**Next Target Condition**
Achieve by: August 30
Meet 80% commitments.

**Outcome Metrics:**
- 80% of stories finished
- Throughput not lower.

**Process Metrics:**
- Testing starts day 6 of 15.

**Target Operating Pattern:**
- Start testing when first story checked in.

**Current Condition**
Not meeting sprint commitments.

**Outcome Metrics:**
- 65% of stories finished in sprint.
- Throughput = 10 stories/sprint

**Process Metrics:**
- Testing starts day 10 of 15.

**PDCA Cycles**
**Record**

**Obstacles**
**Parking Lot**

**Outcome Metrics:**
- 80% of stories finished
- Throughput not lower.

Achieve by: August 30
Meet 80% commitments.

Process Metrics:
- Testing starts day 10 of 15.

Iterate Toward the Target Condition

**1** Understand the Direction or Challenge
**2** Grasp the Current Condition
**3** Establish the Next Target Condition
**4** Iterate Toward the Target Condition

Let’s try science!
The Target Condition is in Unknown Territory

If we already knew how to reach the Target Condition, we would go there now.

One Experiment Each Day

Is the problem really the problem? Is the solution really the solution?

What is a fast, easy way to prove or disprove my hypothesis?
Pattern: Plan-Do-Check-Act (PDCA)

- **PLAN**
  - Testable
  - What do we expect?

- **DO**
  - Conduct the experiment
  - What happened

- **CHECK**
  - What we learned

- **EVALUATE**
  - Adjust based on what you learn

### A Kata – PDCA Cycle Record

<table>
<thead>
<tr>
<th>PDCA CYCLES RECORD</th>
<th>Target Condition:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obstacle: Unexpected network changes</td>
<td>Learner: Coach:</td>
</tr>
<tr>
<td>Date, step &amp; metric</td>
<td>What do we expect?</td>
</tr>
</tbody>
</table>

- **Plan-Do-Check-Act (PDCA) Cycle**

- **A Coaching Cycle**
  - Do a Coaching Cycle
  - Conduct the Experiment

© 2015 Iberle Consulting Group, Inc.
### The Experiment

<table>
<thead>
<tr>
<th>PDCA CYCLES RECORD</th>
<th>Target Condition:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obstacle:</td>
<td>Learner:</td>
</tr>
<tr>
<td>Date, step &amp; metric</td>
<td>What do we expect?</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

1) *What are you going to do?*

2) *What do you expect to see?*

### The Results

<table>
<thead>
<tr>
<th>PDCA CYCLES RECORD</th>
<th>Target Condition:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obstacle:</td>
<td>Learner:</td>
</tr>
<tr>
<td>Date, step &amp; metric</td>
<td>What do we expect?</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

3) *What did you actually see?*

4) *What did you learn?*
What Should We Do Next?

<table>
<thead>
<tr>
<th>PDCA CYCLES RECORD</th>
<th>Target Condition:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obstacle:</td>
<td>Learner:</td>
</tr>
<tr>
<td>Date, step &amp; metric</td>
<td>Coach:</td>
</tr>
<tr>
<td>What do we expect?</td>
<td>What happened</td>
</tr>
<tr>
<td>What we learned</td>
<td></td>
</tr>
</tbody>
</table>

Last experiment

5) What will you try next?
6) What do you expect to see?

Example: Start Testing Sooner

<table>
<thead>
<tr>
<th>PDCA CYCLES RECORD</th>
<th>Target Condition:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obstacle: Testing starts too late.</td>
<td>Learner:</td>
</tr>
<tr>
<td>Date, step &amp; metric</td>
<td>Coach:</td>
</tr>
<tr>
<td>What do we expect?</td>
<td>What happened</td>
</tr>
<tr>
<td>What we learned</td>
<td></td>
</tr>
</tbody>
</table>

Tell team to start testing at midpoint.
Magic occurs.
Test beds aren't ready.
Typical Response

<table>
<thead>
<tr>
<th>PDCA CYCLES RECORD</th>
<th>Target Condition:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Obstacle:</strong></td>
<td><strong>Learner:</strong></td>
</tr>
<tr>
<td>Date, step &amp; metric</td>
<td><strong>Coach:</strong></td>
</tr>
<tr>
<td>What do we expect?</td>
<td></td>
</tr>
<tr>
<td>What happened</td>
<td></td>
</tr>
<tr>
<td>What we learned</td>
<td></td>
</tr>
</tbody>
</table>

**Last experiment**

5) What will you try next?

3) What did you actually see?

4) What did you learn?

READ WHAT YOU SEE

JUMPING TO CONCLUSIONS
Our brain automatically fills in the blanks
WHY IT FEELS ODD

You have to consciously think about it and be more deliberate

Unconscious thinking takes care of routine decisions & conserves brain energy. It’s fast and instinctive.

Deliberate thinking is slow and intentional. It costs much more energy and attention.
Another Kata: Reflecting with a Coach

**Reflect on the Last Step Taken**

Because you don’t actually know what the result of a step will be!

1) What did you plan as your Last Step?
2) What did you Expect?
3) What Actually Happened?
4) What did you Learn?

---

**Rapid PDCA Cycles**

<table>
<thead>
<tr>
<th>Date, step &amp; metric</th>
<th>What do we expect?</th>
<th>What happened</th>
<th>What we learned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write process &amp; give to Team B.</td>
<td>Team B follows the process &amp; that helps.</td>
<td>Team B didn’t like the new process.</td>
<td>New process causes problems for Team B.</td>
</tr>
<tr>
<td>Ask Team B about the problems.</td>
<td>Do a Coaching Cycle</td>
<td>Conduct the Experiment</td>
<td></td>
</tr>
</tbody>
</table>

---

**Learning Leads to a New Path**

**Target Condition:** Fewer failed installs

**Obstacle:** Unexpected network changes

Learner: Coach:

<table>
<thead>
<tr>
<th>Date, step &amp; metric</th>
<th>Plan</th>
<th>Do</th>
<th>Check</th>
<th>Act</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write process &amp; give to Team B.</td>
<td>Team B follows the process &amp; that helps.</td>
<td>Team B didn’t like the new process.</td>
<td>New process causes problems for Team B.</td>
<td></td>
</tr>
</tbody>
</table>
### Example: Start Testing Sooner

**PDCA CYCLES RECORD**

<table>
<thead>
<tr>
<th>Obstacle</th>
<th>Learner:</th>
<th>Coach:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Testing starts too late.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date, step &amp; metric</th>
<th>What do we expect?</th>
<th>What happened</th>
<th>What we learned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tell team to start testing at midpoint.</td>
<td>Magic occurs.</td>
<td>Test beds aren’t ready.</td>
<td>Testers don’t know what to prepare for.</td>
</tr>
<tr>
<td>Tell dev to include test beds in acceptance tests.</td>
<td>Tests that specify test beds.</td>
<td>Some test beds ready, not all.</td>
<td>Testers didn’t understand the short-hand.</td>
</tr>
</tbody>
</table>

### Yes, It’s a Lab Notebook

**PDCA CYCLES RECORD**

<table>
<thead>
<tr>
<th>Obstacle:</th>
<th>Learner:</th>
<th>Coach:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date, step &amp; metric</th>
<th>What do we expect?</th>
<th>What happened</th>
<th>What we learned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experiment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hypothesis</td>
<td>Results</td>
<td>Analysis</td>
<td>Conclusions</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© 2015 Iberle Consulting Group, Inc.
A Single Team Can Use the Improvement Kata

1. Understand the Direction or Challenge
2. Grasp the Current Condition
3. Establish the Next Target Condition
4. Iterate Toward the Target Condition

Process-Level PDCA
1-2 days

Target Conditions
Monthly or every sprint

SCALING UP
We Need Cross-Team Alignment

We Want to Optimize the Value Stream

Value stream: end-to-end delivery of value

How many jobs per day will get done?
Typical IT Department

Track work from initial idea through delivery to customer.

THE IK PATTERN IS USED AT ALL LEVELS

A Target Condition at one level is the Direction for the next level

Organization Level

Understand the Direction or Challenge (from level above)

(Leader’s vision and strategy objectives)

Value Stream Level

Grasp the Current Condition

Establish the Next Target Condition

Process Level

Iterate Toward the Target Condition

PLANNING EXECUTING

Adapted from Mike Rother
Storyboard Hierarchy

Annual Plan → Challenge

Challenge Storyboard

Value-Stream Storyboards

Process-level Storyboards

Scale Up Using the Same Pattern

1) Single Team
   - Value-Stream Target Conditions
     - Monthly to Quarterly – Mgmt
   - Process-Level PDCA
     - 1-2 days

2) Multiple Teams
   - Challenge
     - Yearly – Upper Mgmt

3) Whole Organization
THE IMPROVEMENT KATA ADAPTED FOR SOFTWARE AND IT

Why Adapt the Improvement Kata?

- Effective in manufacturing and health care
- Hypothesis:
  - Lean theory predicts the Improvement Kata should be highly effective in IT and software development
  - Similarity to agile should make it easy to learn and use

- What did we try?
- What actually happened?
- What did we learn?
Our Experiences

- **Current condition**
  - IT departments: complex customer-facing and internal web-based applications, legacy code
  - Department or organization with multiple software development teams
  - Some agile teams, some not
  - Ambitious annual improvement goals
  - Little or no formal improvement planning method (other than team-level retrospectives)

---

**Start by Setting a Direction**

1. Understand the Direction or Challenge
2. Grasp the Current Condition
3. Establish the Next Target Condition
4. Iterate Toward the Target Condition

---

Use the organization’s existing annual planning goal(s).

If multiple goals, check for a common cause.
Grasp the Current Condition

Collect existing metrics relevant to the goal. Look for bottom-line results not details. May need to create new metrics.

Science = measurements, not opinions!

Some Popular Outcome Metrics

Cycle Time = average time for work item from start to finish.

Throughput = average number of work items completed per week.
Current Condition:
Manufacturing vs. Software

<table>
<thead>
<tr>
<th>Usual Improvement Kata</th>
<th>What We Did</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detailed value-stream</td>
<td>Rough value-stream map</td>
</tr>
<tr>
<td>map</td>
<td></td>
</tr>
<tr>
<td>Metrics: throughput,</td>
<td>Metrics: throughput, cycle</td>
</tr>
<tr>
<td>cycle time, takt time,</td>
<td>time, quality e.g. # defects</td>
</tr>
<tr>
<td>pitch</td>
<td></td>
</tr>
</tbody>
</table>

What We Learned:
Process map alone can cause useful change. Best to just get one or two metrics at this point. There are two conflicting definitions of cycle time.

Establish a Target Condition

A Target Condition is a step on the way to the Challenge. It describes the future “bottom line”, not a specific solution.
### Target Condition: Manufacturing vs. Software

<table>
<thead>
<tr>
<th>Usual Improvement Kata</th>
<th>What We Did</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hand-written storyboards on the walls.</td>
<td>Some hand-written, some electronic.</td>
</tr>
<tr>
<td>Not much on how to identify a good Target Condition.</td>
<td>Brainstorm impediments to challenge &amp; pick one to solve.</td>
</tr>
<tr>
<td>Value-stream measurements already exist.</td>
<td>First Target Condition may be measuring the value stream.</td>
</tr>
</tbody>
</table>

**What We Learned:**
Storyboard forces a concise, thoughtful goal. Distinction between goal and solution is tricky for many.

---

### Iterate Toward the Target Condition

1. **1** Understand the Direction or Challenge
2. **2** Grasp the Current Condition
3. **3** Establish the Next Target Condition
4. **4** Iterate Toward the Target Condition

**Planning Phase**

**Executing Phase**

Let’s try science!
What Did We Do?

• Observed and/or coached over a year’s worth of PDCA Cycles

What Did We See?

Significant, sustained improvement in

• Ability to discover effective solutions
• Ability to implement solutions
• Ability to work across teams on solutions

PDCA Cycles:
Manufacturing vs. Software

<table>
<thead>
<tr>
<th>Usual Improvement Kata</th>
<th>What We Did</th>
</tr>
</thead>
<tbody>
<tr>
<td>Daily coaching meetings</td>
<td>Coaching 2x to 3x/week</td>
</tr>
<tr>
<td></td>
<td>Each step around 30 minutes.</td>
</tr>
<tr>
<td></td>
<td>Meet at a regular time.</td>
</tr>
</tbody>
</table>

What We Learned:
Less than 2x/week \(\rightarrow\) loss of momentum
Steps bigger than 30 min \(\rightarrow\) disruption to regular work
PDCA created much faster learning than usual.
What We Learned: Coach’s Role is Critical

- Most people don’t see their own assumptions
- With practice, skills improve rapidly
- Coach doesn’t need to be expert in field
- Peer coaching can work
- To be a good Coach, first be a Learner

Scaling Up: What We Did
Success Card

- One for each successful process change
- Index card → concise summary.
- Makes progress visible.

<table>
<thead>
<tr>
<th>Learner: Mark</th>
<th>Coach: Katie</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Target Condition:</strong> A process to classify defects by root causes.</td>
<td><strong>Results:</strong> We have a checklist which developers can use, and a short training to go with it. We found that testers cannot do the classification.</td>
</tr>
<tr>
<td><strong>Elapsed time for the PDCA cycles:</strong> ~10 days</td>
<td><strong>Benefit:</strong> We can start identifying the root causes which cause the most problems.</td>
</tr>
</tbody>
</table>

Scaling Up: Manufacturing vs. Software

<table>
<thead>
<tr>
<th>Usual Improvement Kata</th>
<th>What We Did</th>
</tr>
</thead>
<tbody>
<tr>
<td>Had value-stream map.</td>
<td>Created value stream map.</td>
</tr>
<tr>
<td>Had basic metrics in place.</td>
<td>Throughput, cycle time, post-release defects</td>
</tr>
</tbody>
</table>

**What We Learned:**
Seeing the value stream changes your perspective. Cross-team coordination requires active management involvement. Value stream is complicated by matrix management.
CONCLUSIONS

Good Process Improvement Method

**Handles uncertainty:**
- Adapts to new information
- Corrects course quickly
- Doesn’t plan too far ahead

**Scales up:**
- Can solve big problems
- Coordinates change across entire organization

**IMPROVEMENT KATA**

1. Understand the Direction or Challenge
2. Grasp the Current Condition
3. Establish the Next Target Condition
4. Iterate Toward the Target Condition

© 2015 Iberle Consulting Group, Inc.
### PDCA is Very Effective

<table>
<thead>
<tr>
<th>Obstacle:</th>
<th>Date, step &amp; metric</th>
<th>What do we expect?</th>
<th>Coach:</th>
<th>Learner:</th>
<th>What happened</th>
<th>What we learned</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>X</strong></td>
<td><strong>X</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1) **What are you going to do?**

2) **What do you expect to see?**

**Coach:**
- Do a Coaching Cycle
- Conduct the Experiment

**Obstacle:**
- Target Condition: **Plan**
- PDCA - Do-Check-Act (PDCA) Cycle

**1) What are you going to do?**

**2) What do you expect to see?**

**PDCA is Very Effective**

**Decomposing by Value Stream is Effective**

- **Understand the Direction or Challenge (from level above)**
- **Grasp the Current Condition**
- **Establish the Next Target Condition**
- **Iterate Toward the Target Condition**

**Organization Level**

**Value Stream Level**

**Process Level**

---

Adapted from Mike Rother
How Could You Use the Improvement Kata?

a. Adopt PDCA with a partner
b. Adopt within a team to turbo-charge team retrospectives
c. Adopt across a value-stream, operating unit, or company to focus improvement

Annual Improvement and Agile Live Happily Ever After
Learn More

• Mike Rother: *Toyota Kata* and [http://www.personal.umich.edu/~mrother/The_Improvement_Kata.html](http://www.personal.umich.edu/~mrother/The_Improvement_Kata.html)
• Hakan Forss: Improvement Kata in software: [https://hakanforss.wordpress.com/tag/toyota-kata/](https://hakanforss.wordpress.com/tag/toyota-kata/)
• PNSQC 2015: Kathy Iberle & Adam Light

**Workshop on Wednesday 8:00pm-5:00pm**
3) True Progress Becomes Visible

Development Team sees this. System Test Team sees this. Which one is correct?

CFD Shows the Entire Value Stream