LEAN IN THE SOFTWARE TEST LAB

Kathy Iberle
Iberle Consulting Group, Inc.
Pacific Northwest Software Quality Conference
October 15, 2013
Hello, My Name is Kathy Iberle

Kathy Iberle has been working with agile software development and Lean development teams for many years. She has developed unusual expertise in the challenging area where hardware and software meet.

Kathy recently retired from Hewlett-Packard after a long career as a programmer, quality engineer, and process improvement expert and is now the principal consultant and owner of the Iberle Consulting Group. Visit her website at www.kiberle.com.
INTRODUCTION

• What is Lean?
• History of Lean Methods
• How Lean Methods Work
• Batch Size
• WIP Controls
• Other Lean Tools
Lean: Three Bodies of Knowledge

Lean Science: Batch Size, WIP Limits, Pull, Flow, Cadence

Lean Management: Visual Controls, Leadership, Education

Lean Knowledge Stewardship: A3s, Five Whys, Kaizen

Alan Shalloway in Scaling Software Agility
THE SYSTEMS VIEW

• Lean Science views your development system as a machine turning ideas into saleable features.
LEAN: A NEW WAY TO SEE THINGS

See the work as discrete batches

See the organization as a set of activities, not as an org chart

PEOPLE DOING STEP 1

PEOPLE DOING STEP 2

Waiting in line
The work moves through a system

Queue: A waiting line
START OF DAY 1

WORK

QUEUE

PEOPLE DOING
STEP 1

QUEUE

PEOPLE DOING
STEP 2

DONE
END OF DAY 1

WORK → QUEUE → PEOPLE DOING STEP 1 → PEOPLE DOING STEP 2 → DONE
START OF DAY 2

WORK

QUEUE

PEOPLE DOING STEP 1

QUEUE

PEOPLE DOING STEP 2

DONE
START OF DAY 2

WORK

QUEUE

PEOPLE DOING STEP 1

QUEUE

PEOPLE DOING STEP 2

DONE
**Middle of Day 2**

- **WORK**
- **QUEUE**
- **PEOPLE DOING STEP 1**
- **QUEUE**
- **PEOPLE DOING STEP 2**
- **DONE**
END OF DAY 2

WORK

QUEUE

PEOPLE DOING STEP 1

QUEUE

PEOPLE DOING STEP 2

DONE
END OF DAY 2
START OF DAY 3

WORK

QUEUE

PEOPLE DOING STEP 1

QUEUE

PEOPLE DOING STEP 2

DONE
START OF DAY 3

WORK

QUEUE

PEOPLE DOING STEP 1

QUEUE

PEOPLE DOING STEP 2

DONE
AN AVERAGE DAY

- This system deals with:
  - *unpredictably arriving* batches of work
  - of *variable size*
  - using *resources of fixed capacity*
TELEPHONE SYSTEMS

How Many?

PHONE CALLS

<table>
<thead>
<tr>
<th>PHONE CALLS</th>
<th>QUEUE</th>
<th>Phone lines</th>
<th>QUEUE</th>
<th>Switchboard operators</th>
<th>DONE</th>
</tr>
</thead>
</table>

IBERLE CONSULTING GROUP, INC.
**Math to the Rescue!**

- Ågner Krarup Erlang applied statistical methods to solve the problem.
- He invented *queueing theory.*

Ågner Krarup Erlang
Denmark
What Erlang (and others) learned

It is possible to do more with the same capacity.

Which bank will serve more customers today?

Why?
TODAY, QUEUEING THEORY IS USED FOR:

- Telecomm systems
- Internet design
- Traffic control
- Manufacturing Lines
**So What?**

- Queueing theory lets us accurately predict the behavior of a system.
  - How many batches will get done per week - throughput
  - How long it will take to do the average batch – cycle time
  - **What to change** to get different results

- This method can be used to manage work.
  - Better predictability
  - Higher throughput
  - Less stress on people
WATCH THE WORK, NOT THE WORKER

• Split work into *batches* which deliver value.
• Make your activities and queues visible.
• Track progress of batches through your system.
• Optimize system via
  • Batch size
  • Work-in-process controls
  • Other tools
WHAT IS A BATCH?

• Batch = a set of work
  • Has a distinct start and a distinct end
  • Delivers value when finished

Why must a batch deliver value?
Because we want to maximize our output of value not our effort
WHAT VALUE DOES A TEST LAB PRODUCE?
WHAT VALUE DOES A TEST LAB PRODUCE?

Information!

Known good configurations

Likelihood that user will encounter a failure

Defect reports

Prediction of unfound defects

Prediction of project end date
Typical Batches in a Software Test Lab

- Testing batches
  - Feature 1
  - Feature 2
  - Feature 3
  - Performance test
  - Final report

- Other work
  - Improve automated test framework
  - Move equipment to building 2
A Batch Has a Start and an End

• When is a testing batch done?
  • Regression: a pre-defined suite has been run
  • Functionality: No new critical defects found in last 20 hours of testing
  • Spot-check: 10 hours of testing have been performed

• Other possible criteria for “done”
  • Customer agrees to accept batch as-is
  • Measurable attributes are within predetermined values
  • All items on checklist are “green”
  • We have used up the allocated time
**WATCH THE WORK, NOT THE WORKER**

- Split work into *batches* which deliver value.
  - Make your activities and queues visible.
  - Track progress of batches through your system.
- Optimize system via
  - Batch size
  - Work-in-process controls
  - Other tools
MAKE YOUR ACTIVITIES VISIBLE

- Show handoffs between different groups or activities.
- Show queues when there is significant waiting.

Stuff to test

NOT STARTED

WRITING TESTS

RUNNING TESTS

DONE

"In Process"
**Visibility: A Visual Planning Board**

- Each sticky note represents a batch.
- Each column represents a state or queue.
WATCH THE WORK, NOT THE WORKER

✓ Split work into *batches* which deliver value.
✓ Make your activities and queues visible.
  • Track progress of batches through your system.
  • Optimize system via
    • Batch size
    • Work-in-process controls
    • Other tools
TRACK PROGRESS OVER TIME

• Each week, count how many batches are
• Not Started
• In Process
• Done

• Graph as a Cumulative Flow Diagram (CFD)
WHAT YOU CAN SEE ON THE CFD

Arrival Rate (Demand)

Departure Rate (Throughput)
EXTRAPOLATION OF PROGRESS
THE “HOCKEY STICK” EFFECT

Sliver Cumulative Flow Diagram

Number of Requirements

Release Date

Date

IBERLE CONSULTING GROUP, INC.
**Watch the Work, Not the Worker**

- Split work into *batches* which deliver value.
- Make your activities and queues visible.
- Track progress of batches through your system.
  - Optimize system via
    - Batch size
    - Work-in-process controls
    - Other tools
A PREDICTABLE SYSTEM CAN BE OPTIMIZED

- By eliminating *bottlenecks*
- By analyzing each step for *waste*
VALUE-STREAM MAPPING

OPTIMIZE A VARIABLE SYSTEM

Monitor and adjust in real-time

Things to watch:

• Batch size
• Work-in-process
• Cadence
• Feedback channels
**Batch Size**

- Batch = set of things moving in lockstep through your process

Diagram:
- One large batch
- Batch 1
- Batch 2
- Two smaller batches
## SOFTWARE TEST – A BATCH

<table>
<thead>
<tr>
<th>Step</th>
<th>Expected Results</th>
<th>Actual Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Log in</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2) Find account</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3) Enter deposit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4) Enter withdrawal</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5) ….</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6) …</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7) ….</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Teardown

**File Defect Reports**

This could be one long test OR two shorter tests

What are the advantages of one large test?
When this test is done, I can choose to run any of these three tests.
**Small Batches → Predictability**

- At the start of the program, uncertainty in end date is always high.
- The only way to reduce uncertainty is to **finish** a batch which is an **actual end product**. Then you know how long that one took.

---

**One Large Batch**

![Graph showing uncertainty in end date for one large batch](image1)

**Lots of Small Batches**

![Graph showing uncertainty in end date for many small batches](image2)
WHAT’S THE BEST SIZE FOR A SOFTWARE TEST?

Transaction Cost:
• Setup:
  • Enter test as “started”
  • Set up account and user
• Teardown:
  • Delete account and user
  • Enter total time spent on test, who ran it, etc.

Holding Cost:
• Delay in value: Don’t get defect reports until entire test is over.
• Lack of flexibility: Can’t decide to start a different test.
• Other cost of delay: More likely that staffer is interrupted. Test becomes inactive but test station still in use.

<table>
<thead>
<tr>
<th>Step</th>
<th>Expected Results</th>
<th>Actual Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Log in</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2) Find account</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3) Enter deposit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4) Enter withdrawal</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

File Defect Reports
TRANSACTION COST VS. HOLDING COST

"SWEET SPOT"
WORK IN PROCESS

• Time and money is spent on maintaining lists and status of partially finished work.
• Partially finished work can become obsolete mid-stream.
• Workers spend time switching between tasks.
WIP CAUSES MULTI-TASKING

Psychology studies show:
- Task switching for easy tasks: 10 seconds and up
- Task switching during knowledge work: up to 10 minutes

This cost is paid every time you switch between jobs.

Task-switching is a huge hidden cost in “knowledge” work.

Cohn, Mike; Agile Estimating and Planning; 2006, p.15;
Original data from Clark and Wheelwright’s Managing New Product and Process Development: Text and Cases; 1993
WORK IN PROCESS (WIP)
WHAT HAPPENS WHEN WIP GROWS?

Doing more jobs concurrently increases cycle time.
A WIP LIMIT CONTROLS WIP

Not Started

Writing Tests

Executing Tests

Done

WIP Limit = 8

One gate at the start.

Don’t start new things until there is room.
Cadence is a powerful tool

- Meetings or actions on a regular schedule
- Reduces overhead, reduces decision points
SLOW CADENCE CAN CAUSE HIGH WIP

Cumulative flow diagram of yearly initiatives

- NOT STARTED
- WIP
- DONE
OTHER TOOLS OF LEAN SCIENCE

- Cadence
- Clean, well-defined feedback channels
- "Standup" meetings
- "Pull" workflow via Kanban
Lean Science in a Nutshell

- Split work into *batches* which deliver value.
- Make your activities and queues visible.
- Track progress of batches through your system.
- Optimize system via
  - Batch size
  - Work-in-process controls
  - Other tools
REFERENCES

• *Principles of Product Development Flow;* Reinertsen, Donald G; 2009;
• *Kanban: Successful Evolutionary Change for Your Technology Business;* Anderson, David; 2010
• Iberle Consulting Group resource page: http://www.kiberle.com/links.html
  • “Visual Planning Boards for Small Groups”
  • “More Techniques for Visual Planning Boards”
  • “Visualizing Workflow in Your Organization”
  • “Cadence: Increase Efficiency by Using a Rhythm”
  • “Finding the Best Frequency for a Recurring Activity”
Questions?