Success Cards
Celebrating Success in Small Increments
By Kathy Iberle and Adam Light

Key Points

- **Success cards make progress more visible.**

- **Success cards demonstrate how learning leads to successful process changes.**

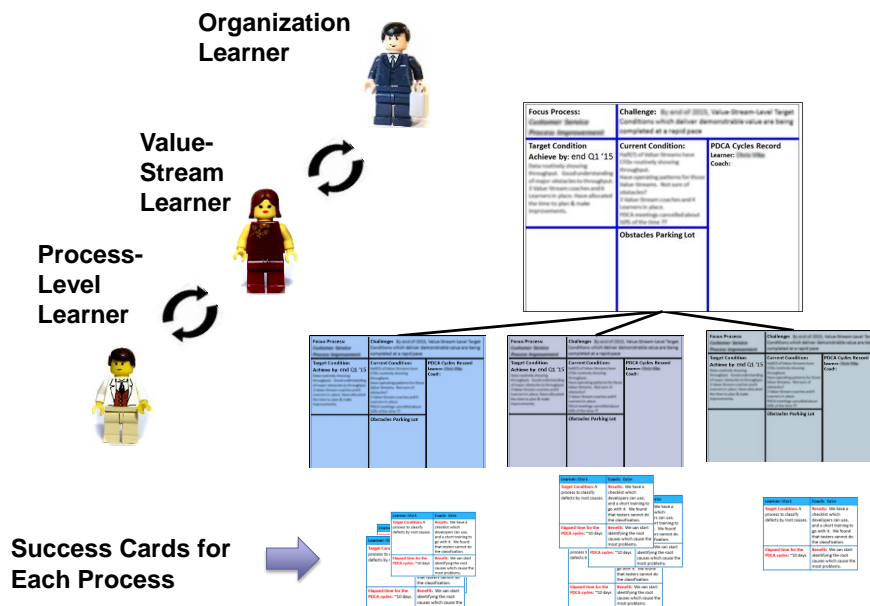- **One success card for each successful process change.**

WHAT IS A SUCCESS CARD?

Success Cards help Learners and teams recognize and celebrate progress.

For each successful process change, a success card is created on a 4x6 or 5x8 card and posted under the Learner's Storyboard.

Because change happens in such small increments, a physical record makes the steady progress more visible. The Success Card format requires the Learner to explain what they learned and how that led to process change.

Learner:	Coach:
Target Condition: *<Process-level target condition>*	**Results:** We learned …. *<describe the information that led to the process change>* so we changed … *<describe briefly what was changed.>*
Elapsed time for the PDCA cycles: *<approximate elapsed time to identify the needed change and implement it>*	**Benefit:** *<How does this change benefit the organization? Use metrics if possible.>*

Organization Learner

Value-Stream Learner

Process-Level Learner

Success Cards for Each Process

SAMPLE SUCCESS CARDS

Our experiments with Success Cards took place in IT departments. Most process-level Target Conditions required multiple "Successes" to achieve the goal, as shown above.

Learner:	Coach:
Target Condition: Size of Zebra System Test Queue is 50% of current size.	**Results:** We learned that some stories can be tested concurrently on the same test bed without interfering with each other, so we changed our process for letting stories into System Test to allow stories to enter concurrently with prior approval from Dev.
Elapsed time for the PDCA cycles: ~10 days	**Benefit:** Some stories move into System Test without waiting as long, reducing average cycle time for an update. Average # stories in System Test Queue dropped by 25%.

Learner:	Coach:
Target Condition: Understand when and where defects are most often inserted into our code.	**Results:** We learned that no one keeps track of when our defects are inserted, so we now ask developers to tag a defect with probable source when they fix it. We learned that we need to agree on definitions for the probable sources.
Elapsed time for the PDCA cycles: ~15 days	**Benefit:** We are starting to get data on what types of errors happen most often, which will help us reduce errors and thus cycle time.

Read the full paper on our adventures with the Improvement Kata at www.kiberle.com.

IBERLE
CONSULTING GROUP, INC.

360.852.1676
www.kiberle.com
kiberle@kiberle.com

SoTech
advisors

503.522.1499
www.sotechadvisors.com
info@sotechadvisors.com