ONE SMALL STEP FOR TECHNOLOGY: THE ART OF THE MICRO-IMPROVEMENT

Dan Fang

Data Coordinator & Market Support Specialist

Tickets for Kids Charities

ON AN INDEX CARD:

- Write down one task you do that falls into one or more of the following criteria:
  - Makes your eyes glaze over
  - Makes you groan out loud
  - Makes you think, "I cannot believe I have to do this…again."
- How often do you find yourself doing this task?
- How much time does it take up during your day?
- (If you don't have a task in mind, you might be too tech-savvy for this presentation. Pretend you're one of your coworkers. Write down one of their tasks.)

ONE SMALL STEP FOR TECHNOLOGY: THE ART OF THE MICRO-IMPROVEMENT

Dan Fang

Data Coordinator & Market Support Specialist

Tickets for Kids Charities

SESSION OVERVIEW

- Introduction
- What is a micro-improvement?
  - What is a macro-improvement?
- Mantras of the micro-improver
- Challenges of the micro-improver
- Checklist for the micro-improver
- Example
- Discussion and Q&A

INTRODUCTION

About me:

- New to tech
- New to nonprofit
- Former life
- Tickets for Kids Charities
- "Accidental Techie"

(definition forthcoming at 2:30pm in Breakout Session III)

WHAT IS A MACRO-IMPROVEMENT?

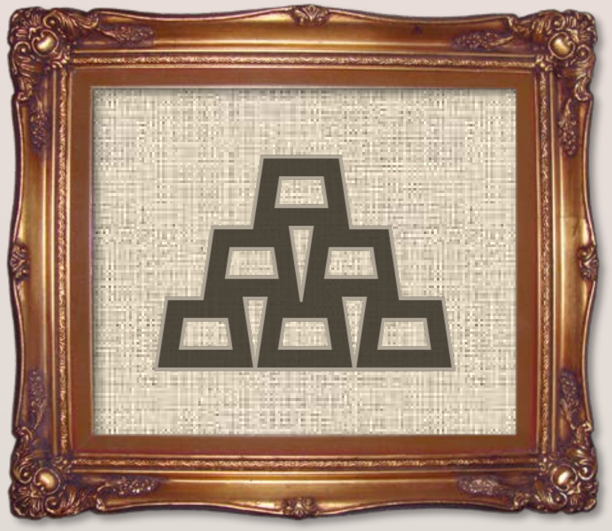
- Examples:
  - Data platform migration
  - Major changes to database structure (new entities/tables)
  - New website
  - Major changes to website structure (unveiling new features like donation portal)
- Involves a lot of planning, strategizing, and coordinating
- May involve a suspension of normal operations
- Long time and high cost

WHAT IS A MICRO-IMPROVEMENT?

- Examples:
  - Document/form management
  - Spreadsheets, reports, and other repeated manipulations therein
  - JavaScript & other cures for manual entry
- Sustainability of human resources
- Some planning/strategizing/coordinating, but not always between many people
- Usually does not involve any suspension of operations longer than a few minutes
- Easier to roll out on a trial-basis
- Small-scale change, small-scale expectations; easier to manage anxiety, too

WHAT IS A MICRO-IMPROVEMENT?

- Iterative and incremental—lessons to learn from software developers
- Build as you learn, learn as you build
- Small things add up!



MANTRAS OF THE MICRO-IMPROVER

BE SELFISH

- Begin with your own tasks
- Automate as much of your daily work as you can.
- If you don’t have additional capacity, you cannot help others
- Easier to test on and train yourself than others

MANTRAS OF THE MICRO-IMPROVER

BE FLEXIBLE

- Research-Write-Rewrite
- Have a concrete goal in mind, but don't waste time holding tightly onto it
- Do your research to figure out what it'll take (Google is your friend!); sometimes, it's just not worth it.
- You may not know what exactly you want to accomplish until after you accomplish it

MANTRAS OF THE MICRO-IMPROVER

PROGRAM FIRST, ASK LATER

…in dev

- Easier to convince others to use something if you can show a finished product.
- A placeholder is NOT a dirty word! If building small blocks saves time in the long run, it’s still worth it.

CHALLENGES OF THE MICRO-IMPROVER

TIME

- Time to learn/code/test
- Time to train. Could be training more people at multiple times

CHALLENGES OF THE MICRO-IMPROVER

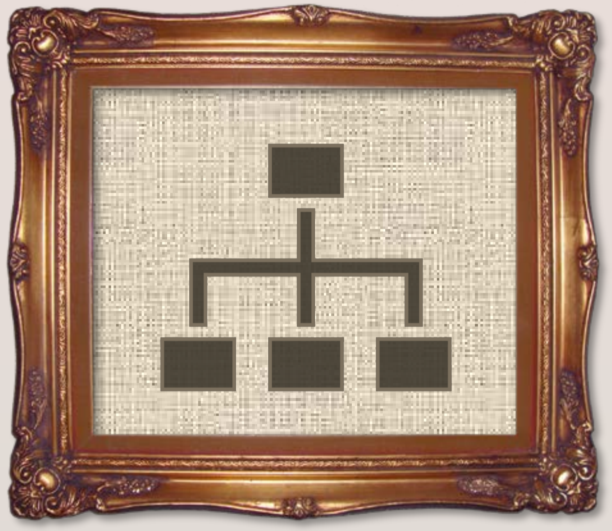
RESOURCES

- Difficult to improve efficiency if you're doing work all the time
- Difficult to train others if they're working all the time

CHALLENGES OF THE MICRO-IMPROVER

ORGANIZATIONAL STRUCTURE

- Not being allowed to deviate from plan
- Resistance to learning new ways of doing something
- Newness overload



CHECKLIST OF THINGS THE MICRO-IMPROVER CANNOT GET AROUND

- Testing!!! Testing, testing, testing. Testing.
- Documentation
- Writing manuals
- Convincing people to read your manual
- Training
- Convincing people to come to training
- Convincing people to use what you just built (easier to ignore because it's not a totally new system; can just go on as before)

Technology	Time to Implement	Time Spent on Task per Week	Time Saved per Week
Manual entry	…it's probably faster than writing by hand!	5 minutes per page to create and check for errors, up to 100 requests for different venues in one week = 8.3 hours (500 minutes)	0
Word templates from database	10 minutes to follow guide 15 minutes to create and upload template	1 minute per page to download 1 minute per page to format fields = 3 hours (200 minutes)	3 minutes per document = 5 hours (300 minutes)
Mail merge from Excel	5 minutes to read a guide	10 minutes to adjust data format and merge per venue (x 5) = 1 hour (50 minutes)	7.5 hours (450 minutes)
Python from Excel	10-20 hours to learn basic coding; 1 hour to write code; 3 hours to try and fail to connect to database	1 minute to download all necessary data	8.3 hours (499 minutes)
C# from database	5 hours to learn a different kind of code; 2 hours to write code	Immediate; push button and done	8.3 hours (500 minutes)
Total	30 hours	Around 1 minute per week to open up the program and run it	1 entire workday

DISCUSSION & QUESTIONS

- What were your tedious tasks? Can they be resolved via micro-improvements?
- What are some challenges you face in getting *to* the technological improvements?
- Other comments, questions, or sarcastic remarks?