Table of Contents

References Useful in Training

Note: to download when you cannot get to YouTube online, put “ss” in front of YouTube in the address.

Games

- Simulating Agile Execution with the Ball Point Game
- Simulating WIP Effects with Balls and a Banana

Useful Videos

A list of videos I've come across over the years that help get concepts across.

- Don Reinertsen - Second Generation Lean Product Development Flow. Talks about some of the content of his book. As usual a very “dense” pitch with a lot of information. If you want to understand impacts of things like queuing theory, lean, and why variability should be preserved for new product development (hint: Black Scholes option pricing model) then this is the video for you. See My notes for more information.
- Dan Pink’s TED Talk. Or if you want the more fun presentation then use Drive: The Surprising Truth About Motivation. Helps you understand how knowledge workers are motivated: autonomy, mastery, and purpose (and take money off the table)
- Dave Thomas - Agile is Dead. Provocative title which talks about how implementations of agile don’t always match up with the specification of agile.
- Martin Fowler - Microservices. Found this to be a useful discussion to help me talk to technical people.
- “Uncle” Bob Martin - The Principles of Clean Architecture. Great anecdotal story at the beginning, as always. Then discussion around how architecture should reflect the domain - what the application does.
- New Zealand All Blacks Haka (this is the meaning of Scrum? no ...
- Jeff Sutherland’s TedX video on doing “Twice the Work in Half the Time"
- VersionOne’s description of the Agile Manifesto
- Scrum in 7 minutes
- The Rong Way to Do Agile: Team Structure from Atlassian
- Planning Poker for Estimates from Mike Cohn
- Example Ball Point Game
- Prudential Ad Showing That We Are All Optimists When Considering the Future
- Why Cost of Delay Matters?
- Sinek’s Ted Talk - Start with Why - To understand how to communicate with people especially as you introduce something new. Bit from 1:35 to 5:15 relevant for product owners, for example, when explaining “vision”.
- High Performance Tree - Lisa Adkins on High Performance teams

- How The Brain Stores Information - TED Talk on importance of visual processing etc.
- Kenny Rubin “Essential Scrum” on Requirements and Change Management
- Kenny Rubin “Essential Scrum” on Product Backlog Refinement
- Jeff Sutherland on the Daily Scrum
- S&*% Bad Scrum Masters Say - Funny
- Henrik Kniberg on the Product Owner role - Key idea “Product Owner must say ‘no’”.
- Lyssa Atkins on Scrum in about 10 mins - Every Scrum Master should know how to explain the framework.
- Dave Allen - Teach Kids About Telling the Time - Funny video to understand how slippery the english language is to drive requirements.
- Steven Johnson - Where Good Ideas Come From - On understanding how innovation works - requires a collision of half ideas (the slow hunch) that have been fermenting in the background for a while. So idea is to provide an environment to connect. “Chance (of innovation) favors the connected mind.”
- The Backwards Bicycle - Great video to understand how its hard to unlearn what we know, that knowledge isn’t the same as understanding, to learn you have to practice, practice, practice, and that you have biases and are unaware of them. See more at Why Is Agile So Hard - The Backward Bicycle?
- Leeroy Jenkins - What happens when 1 person doesn’t consider the rest of the team. From World of Warcraft.
- Day in a Life of Mob Programming. Helpful to talk about learning and trying practices even if we don’t adopt wholesale.
- Dilbert has “the knack". Funny video about becoming an engineer.
- Rugby game. Shows structure emerging from chaos, minimal control, common goals, etc
- Introduction to DevOps
- Agile Manifesto Explained - Quick 3 minute video on the basics of the Agile Manifesto, including a bit of history.
- Are we in Control of Our Decisions: Behavioral economist Dan Ariely, the author of Predictably Irrational, uses classic visual illusions and his own counterintuitive (and sometimes shocking) research findings to show how we’re not as rational as we think when we make decisions.
- Coordination Chaos: Good video to help explain why the old way of working no longer works as the organization grows.
- High-tech Anthropology at Menlo: Video to explain how gemba helps when working on understanding customer requirements.
- Submarine Leadership: David Marquet on changing the leadership model from “leader - follower” to “leader - leader”.
- 5 Dysfunctions of a Team: Patrick Lencioni presenting the materials of the book
- Wisdom of the Crowds demonstration counting gum balls: Useful to help people understand how even uninformed people can contribute to a discussion in estimation.
- To help people understand small vs large batch processing (the ideal of one piece flow in manufacturing world) when you cannot run something like the penny game:
  - Batch of 10 vs batch of 1, simultaneously
  - Batch of 10 vs batch of 1, serially
- Dave Snowden on Organizing a Children’s Birthday Party
- The Power of Empathy - For Leadership
- The Resource Utilization Trap: Henrik Kniberg's excellent demonstration of the problem of focusing

on resource utilization in bringing value to our customers
- Locating Yourself - The Key to Conscious Leadership - Are you operating above the line or below the line.
- Design Thinking - Introduction to the basic ideas. While it is presented as a “linear” process, and misses notions of divergent and convergent thinking, it is a good start.
- How to Use the Customer Empathy Map
- Eric Ries on Innovation Accounting - How do we know we are making progress when all we are doing experiments.
- The Lucky Iron Fish - Helps people understand why we need to do “gemba” (go and see) when trying to understand the requirements of the product.
    - TEDTalk on How the Lucky Fish Can Treat Anemia - Useful video that expands on the discussion above to look at a complete process of making a product “fit for purpose” including the design of the product, the implementation of the product, and the delivery of the product.
- How to Trust People You Don’t Like - Podcast dispels a number of misconceptions about trust.
- How to Write Mission Statements That Don't Suck - stop over wordsmithing your mission statement and avoid “solutions”
- Russ Ackoff on Systems Thinking - ever wondered what it means to do systems thinking? This is a short video explaining what Systems Thinking is and why you cannot just decompose the system into parts and expect improvement overall.
- 10 Reasons Estimation and Planning Fails and What to Do About It by Troy Magennis. Great pitch. Love the basics here. Main message is “a lot of bad things happen in projects which increase utilization into the 'non-linear' (when lead-time graphed over utilization - above 80% utilization leads to exponential lead time) zone and so make estimating useless and forecasting difficult.” Key learnings include:
    - If you are in the non-linear region of utilization for your project (i.e. greater than 80% utilization) then estimation will always fail.
    - Sometimes you need just enough information to decide what not to do, and perhaps toss a coin for remaining options (if they really are close enough).
    - If we only have a high utilization system and we cannot change this, then estimating using points etc doesn't make sense. For high utilization systems we need to track system level impediments to the flow.
    - Top 10 list (portfolio level): (BTW why this project won't finish on time is because of last project - and no amount of estimating is going to help)
        - Don't start on time. Eg delay in previous project
        - No team (no one to do the work).
        - Partial team. Eg on prior project. Half strength team → over utilization
        - Partial body substitution. Similar to above.
        - Missing skills sets. Variation on a theme. Are we producing useful stuff. Look busy but don't know whether we are making progress. See “capability matrix” based on level of capability - novice and learner (but will to learn), do / maintain and so able to modify (and make small additions), teacher / creator (able to do work and show others). Help actively develop your T-shaped people.
        - Over-stated parallel effectiveness. Adding teams / people to make it faster does not scale linearly. Limited by whatever serial processes you have in place. Variation on Amdahl's but applied to people. Spend 8x the effort but only 3x additional capability. Need to invest in independents and tools (eg continuous deployment)
        - Dependency and friction. Tracing through how work was actually done. How many

levels of dependency. 1.5X the Sprint length is the fastest you could move through the system (as sometimes it doesn't get delivered in the Sprint expected.) Need to understand the dependencies you have. Every dependency you can remove from your delivery stream doodles your chances of delivering on time. This could mean, for example, having merged larger teams to reduce the number of dependencies.
        - Carried over defects and debt.
        - Ship stoppers. There are always these. If you ask the question “if I double the team would this help”. Often answer is “no”. Tried to find these out early eg by surveys every week “would you ship this tomorrow …”
        - Splitting. Rate the we finish items and the rate that we arrive are never the same, because we split. We need to take into account that we are splitting items as we do work.
    - Use these “assumptions” to understand whether the project is on track. Don't really need status reports if these assumptions are not met.

Games

- Zin Obelisk Game - Fun game to experience and examine the sharing of information in team problem solving and how leadership, cooperation, and conflict issues effect team problem solving.
- Scrum Master vs Project Manager Game - Useful game which helps folks understand how the responsibilities of a traditional project manager are moved to both the Scrum Master, the Product Owner and the Team when applying agile approaches.

Also videos of games:

- Airplane game
- Ball Point Game

Other Useful Links for Training

- 12 Principles of Agile Development by Julien Henzelin
- VersionOne "State of Agile" Survey (9th Edition). You need to register to get it - my version.
- Help answer question "What does a Scrum Master do?"
- Base approach to splitting stories. Additional thinking at from Bill Wake 20 ways to split stories. These ideas are to help to get to a minimal, end-to-end solution present which usually has high value, from which you can thin then fill in the rest of the solution in subsequent iterations.
- Role of Product Management from Pragmatic Marketing framework
- The Burning Platform metaphor by Daryl Conner. Idea is to have the resolve to make the change happen. Resolve can come from current or anticipated problems or current of anticipated opportunities, not necessarily as a result of “peril” (which is “current problem” classification.
- Alistair Cockburn etc on Communication including richness chart and the implications of this

richness.

- James Coplien on the Borland Quattro Pro development which some say was the most productive ever recorded. To quote the abstract “The project assimilated requirements, completed design and implementation of 1 million lines of code, and completed testing in 31 months. Coding was done by no more than eight people at a time, which means that individual coding productivity was higher than 1000 lines of code per staff-week. The project capitalized on its small size by centering development activities around daily meetings where architecture, design, and interface issues were socialized. Quality assurance and project management roles were central to the development sociology, in contrast to the developer-centric software production most often observed in our studies of AT&T telecommunications software. Analyses of the development process are “off the charts” relative to most other processes we have studied.”
- Discussion on roles beyond PO, SM, and team basics
- Advice on Running Scrum-of-Scrums meeting from Mike Cohn.
- What Google Learned From Its Quest to Build the Perfect Team by Charles Duhigg (New York Times article). Article which points out how little team composition (in terms of the A team, for example) really points to an effective team and that instead the “conversational turn-taking” and “average social sensitivity” on teams (supporting “psychological safety”) is what predicts team performance best. Other factors were also recognized as important - like making sure teams had clear goals and creating a culture of dependability. To support the “safety” aspect “What Project Aristotle has taught people within Google is that no one wants to put on a 'work face' when they get to the office. No one wants to leave part of their personality and inner life at home. But to be fully present at work, to feel 'psychologically safe,' we must know that we can be free enough, sometimes, to share the things that scare us without fear of recriminations. We must be able to talk about what is messy or sad, to have hard conversations with colleagues who are driving us crazy. We can’t be focused just on efficiency. Rather, when we start the morning by collaborating with a team of engineers and then send emails to our marketing colleagues and then jump on a conference call, we want to know that those people really hear us. We want to know that work is more than just labor.” “Sakaguchi’s experiences underscore a core lesson of Google’s research into teamwork: By adopting the data-driven approach of Silicon Valley, Project Aristotle has encouraged emotional conversations and discussions of norms among people who might otherwise be uncomfortable talking about how they feel”. And “It’s easier to talk about our feelings when we can point to a number.” My version what_google_learned_from_its_quest_to_build_the_perfect_team_-_the_new_york_times.pdf
- Adaptive Leadership Accelerating Enterprise Agility - Jim Highsmith. Talks about the way we need to bring the agile mindset to the enterprise.
- Work Characteristics and Work Performance of Knoweldge Workers: What Goes Hand in Hand? - Research paper on knowledge work and importance of various characteristics of the work on the performance of knowledge workers. From abstract “The aim of the paper was to investigate the interplay among a wide range of work characteristics and knowledge workers’ performance outcomes. Specifically, we examined the nature of relationships between various task-, knowledge- and social characteristics of work design and both task and contextual performance. Using an adapted Work Design Questionnaire and applying PLS-SEM modelling technique, we analysed cross-sectional and cross-occupational sample of 512 Croatian knowledge workers from 48 organizations. Our findings confirmed the existence and importance of interaction between work characteristics and work outcomes. However, the results suggest that only knowledge characteristics of work design exhibit a significant effect on both distinct dimensions of work behaviour, while task and social characteristics showed different effects on task and contextual performance, respectively.”

- Create your own project cartoon - remember that picture - what customer asked for, what project management understood, etc. Here you can create your own.
- Article - Sliding Toward Success and related tool Project Success Sliders Tool - Mike Cohn's approach to working trade-offs between the various constraints on a project.
- Wake up in the morning game - Simulation or game to help people understand story mapping.
- This American Life - NUMMI and how America is slowly learning the lean lesson from Toyota.
- Software Development Performance Index. My copy whitepaper-sdpispecifications-v1.pdf. This is work that came originally from Larry Maccherone while working with Rally. Ideas is to use performance, quality (release ability), predictability, and responsiveness. For example:
    - Release ability - if team worked on nothing else but close out bugs how long
    - Responsiveness - how long to get in a bug
    - Performance - thought put - closed work items
    - Predictability - how consistent is our delivery (versatility)
- Discussion of remote vs co-location: Martin Fowler with a nuanced discussion on the trade-offs of remote vs co-located people.
- The Impact of Agile Quantified - A set of data looking at all types of metrics from a dataset of thousands of Teams

SAFe Training Videos

Leading SAFe

- Formula 1 Pit Stops - investing in transaction cost - wheels designed (architected) to come off easily, jack lifts entire car, etc.
- Greatness by David Marquet
- Calculating WSJF
- SAFe at Travelport
- What is DevOps?

SAFe for Teams

- Penny Large Batch
- Penny Small Batch
- Penny Small Batch Overview
- Principle 4 Building Incrementally
- Designing Your Team’s Kanban System

SAFe SM

- Scrum Master Stories: Madison Fisher on Vimeo - A day in the life of an SM
- Scrum Master Stories: Sam Ervin - high performing team
- Scrum Master Tips: Conflict Resolution
- TravelPort - The Power of PI Planning
- PI Planning: A Quick Overview
- PI Planning with Distributed Teams
- Effective Iteration Planning Meeting
- Effective Daily Standup
- Effective Backlog Refinement
- How to Run an Effective Iteration Review
- Effective Iteration Retrospective
- What is DevOps
- I&A Problem Solving and Root Cause Analysis

SAFe PO/PM

- The Program Board
- PO and Iteration Planning
- PO and the Daily Standup
- PO and Backlog Refinement
- PO and the Iteration Review
- PO and the Iteration Retrospective
- What is DevOps
- Continuous Delivery Pipeline
- I&A Retrospective and Problem Solving Overview

Web, Reference, Videos, Training