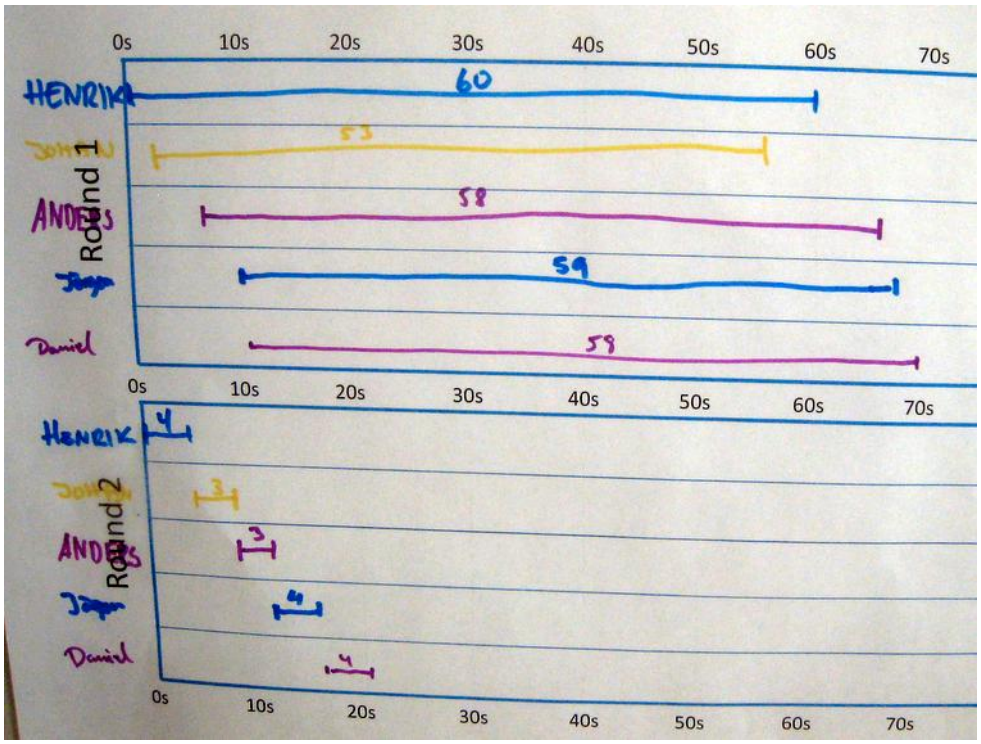
Das Multitasking-Namensspiel

oder: Wie lange dauert es einen Namen zu schreiben?

Henrik Kniberg
Version: 2011-12-07

Übersetzung: Christoph Oberle (2013-05-16) @OldAgile



Über diesen Artikel 3
- Spoiler-Warnung 3
- Zielsetzung dieses Artikels und der Übung 3

Über die Übung 4
- Herkunft des Multitasking-Namensspiels 4
- Der Name der Übung 4
- Lizenzierung 4
- Zeitaufwand und Materialbedarf 4

Zusammenfassung für Eilige 5

Detaillierte Beschreibung 5
- Schritt 1: Schätze,wie lange es dauert, einen Namen zu schreiben 5
- Schritt 2: Diskutiere über Einflussfaktoren 6
- Schritt 3: Bilde Gruppen und beschreibe die Rolle der Kunden 6
- Schritt 4: Beschreibe die Entwickler-Richtlinie für die erste Runde 8
- Schritt 5: Die erste Runde läuft 9
- Schritt 6: Ergebnis-Erfassung für die erste Runde 10
- Schritt 7: Entwickler austauschen und zweite Runde erklären 12
- Schritt 8: Die zweite Runde 13
- Schritt 9: Messwerte der zweiten Runde erfassen 14

Skalieren dieser Übung 19

Varianten 20

Moderatoren-FAQ 20

Über diesen Artikel

Spoiler-Warnung

Dieser Artikel wurde für alle diejenigen geschrieben, die als Coaches, Trainer oder wer auch immer erklären wollen, was das Problem beim Multitasking ist

Wenn du vorhast, einen meiner Kurse über Lean & Agile (oder einen eines anderen Lean/Agile-Trainers wie z.B. Mary Poppendieck) zu besuchen, dann empfehle ich dir diesen Artikel nicht weiterzulesen. Du wirst weniger Spaß an dem Spiel haben, wenn du die Details dieser Übung schon vorher kennst.

Das meine ich ernst! Höre jetzt sofort auf zu lesen, wenn du vorhast, diese Übung in den nächsten Monaten mitzuspielen.

Zielsetzung dieses Artikels und der Übung

Die Zielsetzung dieses Artikels ist es zu beschreiben, wie schlecht Multitasking ist und wie leicht wir uns zum Multitasking verleiten lassen.

In erster Linie wurde dieser Artikel für Lehrer und Moderatoren geschrieben, die wissen wollen, wie die Übung funktioniert und wie man sie erfolgreich moderiert. Jedoch kann damit vermutlich auch jeder andere Leser ein besseres Verständnis über die Problematik von Multitasking erlangen und damit hoffentlich Ansätze finden, das Problem im eigenen Arbeitsumfeld zu erkennen und zu lösen.

PS – wenn du nicht weißt, was das Wort *multitasking* bedeutet, dann ist das nicht schlimm. Lies einfach weiter! Betrachte diesen Artikel einfach als eine ausführliche Begriffsdefinition :o)

Über die Übung

Herkunft des Multitasking-Namensspiels

Als Coach bin ich immer wieder fasziniert davon, wieviel Multitasking bei den meisten meiner Kunden geschieht und wieviel unnötige Verschwendung dadurch verursacht wird. Immer wieder erkenne ich, dass die Identifikation und Reduktion aller Arten von Multitasking eine der billigsten und schnellsten Möglichkeiten darstellt, um die Produktivität jedes Teams signifikant zu erhöhen.

Über die Jahre habe ich versucht eine Übung zu finden um das Problem zu verdeutlichen. Wenn ein Bild mehr sagt als tausend Worte, dann ist eine Übung mehr wert als tausend Bilder. Ich lernte mehrere Übungen kennen, manche einfach, manche komplex, aber keine die das Problem wirklich auf einfache und wiederholbare Art und Weise auf den Punkt brachte. Deswegen borgte ich mir die besten Ideen der anderen Übungen aus und erfand meine eigene. Zu meiner Überraschung funktionierte sie beim ersten Mal schon richtig gut (einige der Teilnehmer, die von der Übung schwer beeindruckt waren, sagten sie sei der Höhepunkt des ganzen Kurses gewesen). Im Laufe der Jahre habe ich sie kontinuierlich verbessert und jetzt funktioniert sie noch besser als damals.

Diese Übung ist bei den Lean & Agile Trainings mittlerweile sehr populär geworden. Eine der Trainerinnen, die diese Übung anwenden, ist Mary Poppendieck. Normalerweise steht sie solchen Übungen eher skeptisch gegenüber und wendet sie in ihren Trainings nicht an. Aber als sie diese Übung in einem Lean-Workshop kennenlernte, den wir gemeinsam moderierten, machte sie begeistert mit und nahm sie umgehend in ihr eigenes Kurs-Material auf. Mary und andere Trainer erzählen mir immer wieder, wie stark diese Übung wirkt, insbesondere bei Management-Teams.

Der Name der Übung

Die Übung heißt ”Das Multitasking-Namensspiel”, weil es genau das ist. Jedoch nenne ich es in den Trainings oft ”Wie lange dauert es einen Namen zu schreiben”, um nicht die Überraschung zu verderben. Außerdem ist der zweite Name deshalb cool, weil ich die Übung damit eröffne diese einfache Frage zu stellen :o)

Lizenzierung

Benutze diese Übung wie du willst, aber ich würde es begrüßen, wenn du es erwähnen oder irgendwo angeben würdest, dass die Übung von mir kommt.

Die offizielle URL ist http://www.crisp.se/henrik.kniberg/multitasking-name-game

Formal ist die Übung unter der Creative Commons Lizenz (Creative Commons Attribution-ShareAlike 3.0 Unported License) lizensiert.

Zeitaufwand und Materialbedarf

Die eigentliche Übung dauert ungefähr 20 Minuten (unabhängig von der Größe der Gruppen). Dazu kommen noch einmal 20 Minuten für die Nachbereitung. Das einzige benötigte Material sind Stifte & Papier & eine Stoppuhr. Idealerweise dicke Marker (Whiteboard-oder Permanent-Marker), Karteikarten und eine große Stoppuhr, die an Wand projiziert wird. Auf Computer und Beamer kann

aber auch ganz verzichtet werden. Die Übung kann genausogut auch einfach mit einem Flipchart moderiert werden.

Zusammenfassung für Eilige

1. Schreibe 5 Namen, alle gleichzeitig. Messe, wie lang es für jeden Namen dauert, und wie lange insgesamt.
2. Dann schreibe 5 Namen, einen nach dem anderen. Messe wieder die Dauer.
3. Staune über die Kosten des Multitasking.

Detaillierte Beschreibung

Schritt 1: Schätze wie lange es dauert einen Namen zu schreiben

Ich überspringe die einleitenden Worte und beginne damit, die einfache Frage zu stellen: ”Wie lange dauert es, einen Namen zu schreiben?”

Das führt normalerweise zu einiger Verwirrung weil die Leute überlegen, was ich mit der Frage meinen könnte. Ich halte eine Karteikarte hoch ”Zum Beispiel so.”

Zunächst wollen die Leute keine Meinung äußern, versuchen sich hinter ”das kommt darauf an” zu verstecken. Ich bringe sie dazu, eine Antwort zu geben, indem ich Dinge sage, wie: ”Ihr könnt das wirklich nicht schätzen? Dauert es länger als ein Jahr? Oder weniger als eine Sekunde?” Schließlich fangen sie an, mir Schätzungen zuzurufen, normalerweise liegen die so um die 4 Sekunden.

Dann frage ich, wie lange es dauert 5 Namen zu schreiben. Das Ergebnis liegt normalerweise bei ungefähr 20 Sekunden (die erste Schätzung x 5)

Diese Zahlen schreibe ich auf das Flipchart.

Wie lange dauert es, einen Namen zu schreiben?

	Schätzung
1 Name:	4 Sek.
5 Namen:	20 Sek.

Schritt 2: Diskutiere über Einflussfaktoren

Danach frage ich "Welche Faktoren beeinflussen diese Zeit? Wenn ihr sagt, 'das kommt drauf an', worauf kommt es da an?". Die Übungsgruppe beginnt, Punkte zu nennen und ich schreibe sie auf. Üblicherweise Punkte wie:

Welche Faktoren beeinfussen diese Zeit?

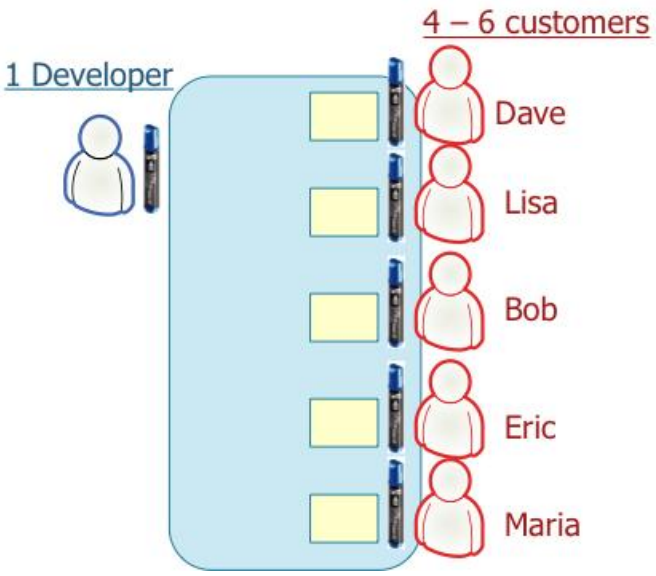
- *Länge des Namens*
- *Komplexität*
- *Handwerkszeug*
- *Erwartete Qualität*
- *Schreibkenntnisse*
- *... und so weiter ...*

Das ist eine Falle! In der überaus größten Mehrzahl aller Fälle wird niemand "Multitasking" als einen der Faktoren nenne, die die benötigte Zeit beeinflussen. Deshalb erwähne ich normalerweise den auch den Namen der Übung nicht, ich möchte dabei zeigen, dass wir dazu tendieren den Einfluss von Multitasking zu vergessen.

Danach sage ich "OK, lasst uns die Wahrheit herausfinden".

Schritt 3: Bilde Gruppen und beschreibe die Rolle der Kunden

Ich bitte die Teilnehmer sich in Gruppen zu 5 bis 7 Leuten aufzuteilen. In den Gruppen ist jeweils eine Person Entwickler (Developer) und alle anderen sind Kunden (Customers).



Die einzige Fähigkeit, die der Entwickler haben muss ist es, Buchstaben auf ein Blatt Papier schreiben zu können. Das Werkzeug, das er dazu braucht, ist ein dicker Marker.

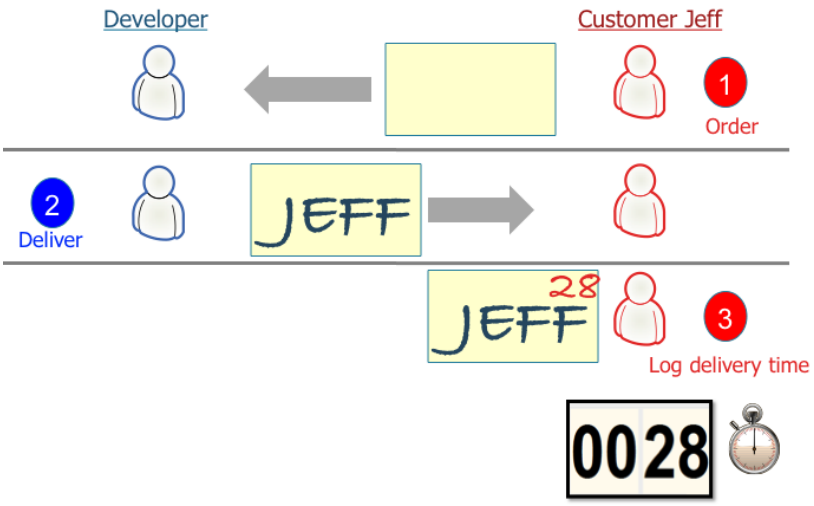
Das erzähle ich den Kunden:

Jeder von euch hat eine leere Karteikarte. Alles was ihr wollt ist dass euer Name auf der Karte steht. Das Problem dabei ist, dass ihr nicht wisst, wie man Buchstaben schreibt – deshalb braucht ihr den Entwickler!

Außerdem willst du, der Kunde, deinen Namen so schnell wie möglich geliefert bekommen. Es ist deine Aufgabe die benötigte Zeit zu erfassen. Wenn deine Namenskarte fertig ist, dann schreibst du drauf, wie lange es gedauert hat (ja, du kannst Ziffern schreiben). Um dir dabei zu helfen, werde ich (der Moderator) eine große Stoppuhr mit dem Beamer anzeigen.

Der Bestellprozess sieht also so aus:

1. sende deine Karte an den Entwickler, sag ihm deinen Namen.
2. Warte darauf, dass der Entwickler deinen Namen schreibt und ausliefert.
3. Schau nach der Zeit und schreibe die Lieferzeit auf die Karte.



Wenn die Lieferung einen Fehler hat (z.B. einen Schreibfehler) dann nimm die Zeit noch nicht. Stattdessen sende die Karte zurück zum Entwickler damit er sie korrigiert. Die Lieferung wird erst dann als fertig betrachtet, wenn sie korrekt ist.

Ihr könnt mit dem Entwickler sprechen und Fragen beantworten. Ihr könnt nur keine Buchstaben schreiben.

Namensschilder, Visitenkarten etc. sollten natürlich für die Dauer der Übung weggeräumt werden.

Schritt 4: Beschreibe die Entwickler-Richtlinie für die erste Runde

Das sage ich den Entwicklern:

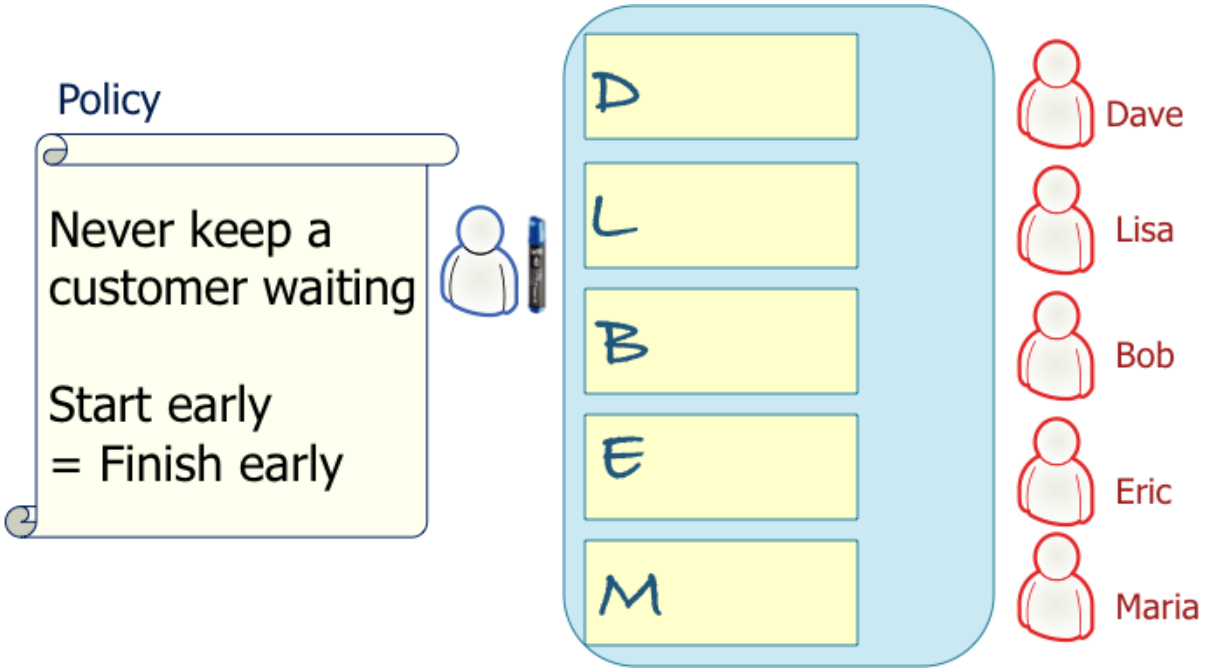
Eure Aufgabe als Entwickler ist es, einfach die Kunden-Namen aufzuschreiben. Aber, dabei müsst ihr die Unternehmens-Richtlinien befolgen!

Unsere Unternehmens-Richtlinie ist *Ein Kunde darf niemals warten müssen,* weil das ein schlechtes Geschäftsgebaren wäre. Wir könnten einen Kunden verlieren, der warten muss. Darüber hinaus glauben wir: *Umso früher man mit etwas anfängst, umso früher ist man auch damit fertig.* Stimmt's?

> *An dieser Stelle halte ich inne und prüfe sorgfältig, ob irgendjemand dieser Aussage wiederspricht. Normalerweise macht das keiner, weil es offensichtlich klingt. Eines der Ziele der Übung ist es zu zeigen wie falsch diese Aussage ist.*

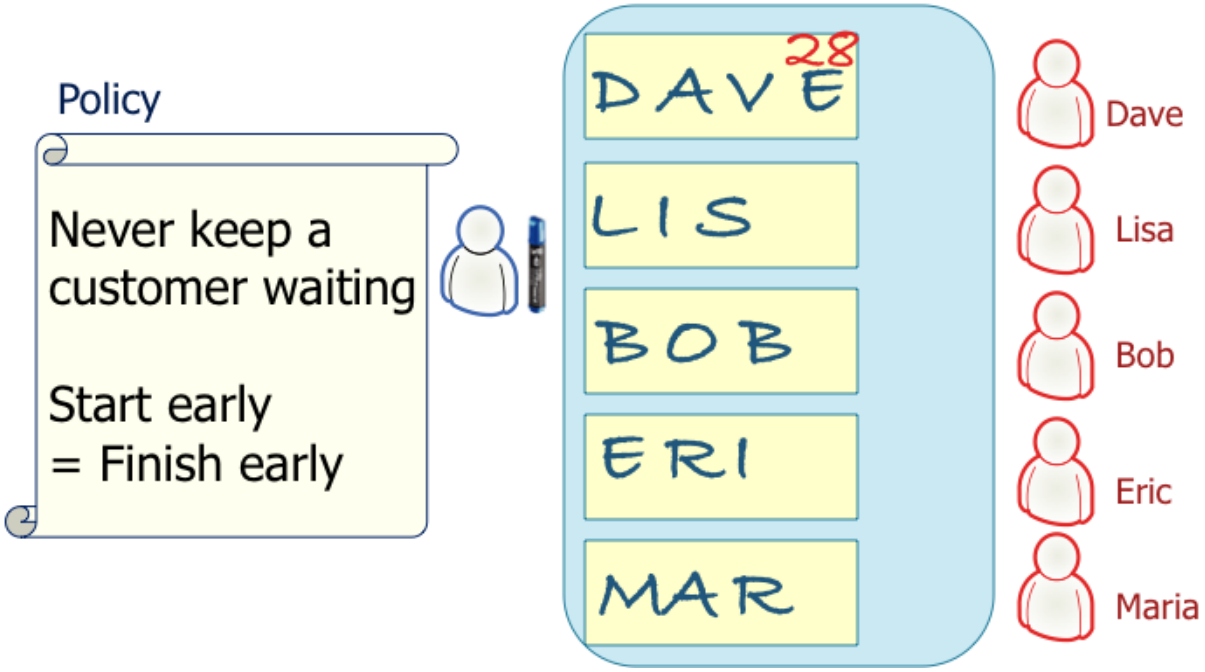
Um diese Unternehmens-Richtlinie einzuhalten, müsst ihr als Entwickler alle Projekte gleichzeitig bearbeiten!

Wenn ich die Stopuhr starte geben euch alle Kunden gleichzeitig ihre leeren Karten und sagen euch ihren Namen. Ihr schreibst dann den ersten Buchstaben des ersten Kunden, dann den ersten Buchstaben des zweiten Kunden, usw.



Wenn der erste Buchstabe jedes Kunden geschrieben ist, wendet ihr euch wieder dem ersten Kunden zu und beginnt damit, den zweiten Buchstaben jedes Kunden zu schreiben, usw.

Immer wenn ein Name fertig ist, liefert ihr diese Karte an den Kunden, damit er/sie die Lieferzeit auf die Karte schreiben kann.



So, das ist der erste Durchgang. Ich vergewissere mich sorgfältig, dass alle die Anweisungen verstanden haben, dass jeder einen Stift hat und dass jeder Kunde eine leere Karteikarte bekommen hat.

Dann starte ich die Uhr.

Schritt 5: Die erste Runde läuft

Jetzt beobachte ich die Entwickler, um zu überprüfen , dass sie der Unternehmens-Richtlinie folgen (das ist normalerweise klein Problem, wenn ich die Richtlinie gut erklärt habe).

Die Teilnehmer haben großen Spaß an dieser Übung, da jeder erkennen kann, wie schlecht es funktioniert und die meisten Teilnehmer die Situation aus ihren eigenen Projekten kennen.

Es gibt viel Kommunikation und Konfusion, da der Entwickler die Kunden häufig nach ihren Namen fragen muss und in einigen Fällen auch nach der Schreibweise.

Manchmal steuern die Kunden den Entwickler indem sie ihm die einzelnen Buchstaben vorsagen. Das ist O.K., ich erlaube auch das, da es so gut wie keinen Einfluss auf das Ergebnis hat.

Sobald die Kunden ihre fertige Karte vom Entwickler zurückbekommen, prüfe ich (und erinnere ich sie daran), dass sie die Lieferzeit darauf schreiben.

Wenn alle aufgehört haben zu schreiben, stoppe ich die Uhr.

Schritt 6: Ergebnis-Erfassung für die erste Runde

Ich fordere jeden Tisch auf, den Median der Lieferzeit zu ermitteln ("wenn ihr die Karten nach Lieferzeit sortiert, was steht dann auf der mittleren Karte?"). Ich notiere den Median von allen Antworten der Tische (also den Median der Mediane...). Typischerweise liegt er so bei 50 Sekunden.

Dann frage ich jeden Tisch nach der Gesamt-Lieferzeit, also wie lange es gedauert hat, alle 5 Namen auszuliefern. Und ich schreibe wieder den Median aller dieser Zeiten auf. Typischerweise so um die 60 Sekunden.

	Schätzung	gemessen
1 Name:	4 Sek.	50 Sek.
5 Namen:	20 Sek.	60 Sek.

Jetzt ist es Zeit für eine lustige kleine Beschimpfung. "Ihr habt geschätzt, dass es 4 Sekunden dauern würde, einen Namen zu liefern, es hat aber mehr als 12 mal so lange gedauert! Ich weiß ja, dass manche Projektmanager jede Schätzung heimlich mit PI multiplizieren, aber in diesem Fall würde uns ja nicht mal 2 PI in die Nähe der wahren Lieferzeit bringen!"

Und dann kommt die Killer-Frage:

"So, dann helft mir mal bitte: Welcher dieser Einflussfaktoren hat denn diese unglaubliche Verzögerung verursacht?" (und ich zeige zum Flipchart)

Welche Faktoren beeinflussen diese Zeit?

- *Länge des Namens*
- *Komplexität*
- *Handwerkszeug*
- *Erwartete Qualität*
- *Schreibkenntnisse*
- *... und so weiter ...*

"Habt ihr alle unglaublich lange Namen?"

"War die Schreibweise sehr komplex?"

"Haben die Stifte nicht funktioniert?"

"Habt ihr wunderschöne kalligraphische Buchstaben gemalt, oder habt ihr sie in Stein graviert?"

"Hatten wir einfach sauschlechte Entwickler?"

Nach einigem Gelächter und Diskussion wird das Offensichtliche festgestellt – das Problem hatte nichts mit den beschriebenen Einflussfaktoren zu tun. Auf der Liste fehlt der wichtigste Einflussfaktor – Multitasking! Ich schreibe ihn dazu – deutlich hervorgehoben!

Wie lange dauert es, einen Namen zu schreiben?

	Schätzung	gemessen
1 Name:	4 Sek.	50 Sek.
5 Namen:	20 Sek.	60 Sek.

Welche Faktoren beeinflussen diese Zeit?

- Länge des Namens
- Komplexität
- Handwerkszeug **Multitasking!**
- Erwartete Qualität
- Schreibkenntnisse
- … und so weiter …

Die anderen Einflussfaktoren verblassen dagegen.

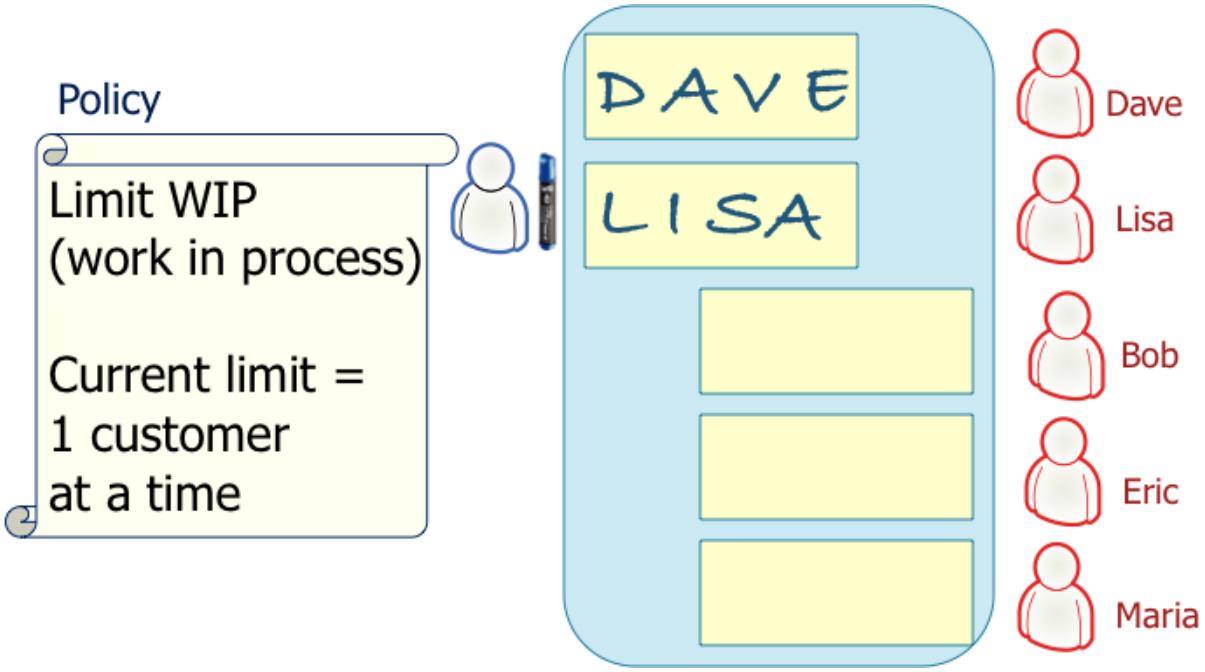
Das war das einfachste vorstellbare Projekt – einfach nur Namen auf ein Blatt Papier schreiben! Aber trotzdem hat uns das Multitasking die Lieferzeit um 1250% aufgebläht. Und nicht nur das, es hat auch unsere Produktivität auf ein Drittel von dem reduziert, was wir erwartet hatten. Wir glaubten es würde 20 Sek. dauern um 5 Namen zu schreiben, aber es hat 60 Sekunden gedauert!

Zeit für die zweite Runde.

Schritt 7: Entwickler austauschen und zweite Runde erklären

Ich bitte alle Entwickler aufzustehen und zum nächsten Tisch zu gehen, damit jeder Tisch einen neuen, unverbrauchten Entwickler bekommt. Den Entwicklern erzähle ich, dass sie jetzt in einer neuen Firma arbeiten, die eine komplett andere Unternehmens-Richtlinie haben.

Die Unternehms-Richtlinie dieser neuen Firma ist *Limit WIP* (Work in Process). Und das aktuelle WIP-Limit liegt bei 1.

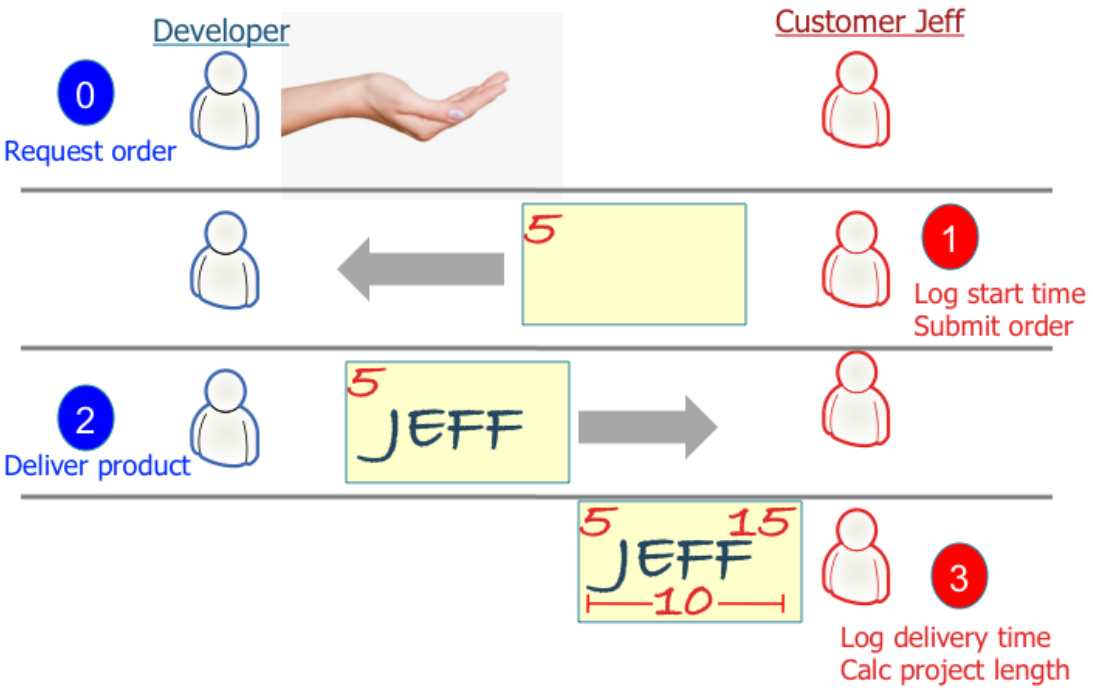


Das bedeutet, dass der Entwickler zu jeder Zeit immer nur *für einen Kunden* arbeiten darf. Das heißt, er fängt nicht an LISA zu schreiben bevor mit DAVE fertig ist. Und die arme Maria muss am längsten warten, mit ihrem Projekt wird solange nicht begonnen, bis alle anderen Aufträge erledigt sind.

Eine wichtige Auswirkung dieser Änderung ist: Der Entwickler hat jetzt die Kontrolle über seinen Arbeitsvorrat! Jetzt entscheidet der Entwickler, wann er mit einem Auftrag beginnt (er nimmt sich die Arbeit selbst), nicht mehr der Kunde (der sie dem Entwickler bisher "aufgedrückt" hat).

Das heißt, wir haben jetzt ein *Pull*-System statt eines *Push*-Systems.

Dadurch bekommen wir einen neuen Arbeitsschritt am Anfang der Auftragsbearbeitung:



Der Kunde darf dem Entwickler seine Karte nicht mehr einfach geben. Jetzt entscheidet der Entwickler, wann er bereit ist, mit dem nächsten Projekt anzufangen. Er streckt dann seine Hand zu einem Kunden aus und bittet ihn um seine Karte und den Auftrag.

Und weil jetzt nicht mehr alle Projekte zur gleichen Zeit starten, müssen alle Kunden die Anfangszeit und die Endzeit ihres Auftrags notieren, damit sie die Lieferzeit als Differenz dieser beiden Zeiten berechnen können. Der Kunde erfasst also zwei Messwerte: ”Wann habe ich meine Lieferung erhalten?” und außerdem ”Wie lange hat der Entwickler an meiner Lieferung gearbeitet?”

OK, es kann losgehen! Ich versichere mich noch einmal, dass jeder die neuen Anweisungen verstanden hat, sorge dafür, dass die Kunden neue Karten haben (und dass die alten Karten beiseite gelegt worden sind).

Dann starte ich die Zeit!

Schritt 8: Die zweite Runde

Beobachte, was passiert und genieße es!

Schritt 9: Messwerte der zweiten Runde erfassen

Nachdem alle aufgehört haben zu schreiben, stoppe ich die Uhr und sammle die Statistiken auf die gleiche Art wie in der ersten Runde. Das heißt, den Median um ein Projekt fertig zu bekommen (von Startzeit bis Endzeit) und den Median um alle 5 Aufträge zu erledigen.

Der Unterschied ist dramatisch. Typischerweise sieht es ungefähr so aus:

	Schätzung	Runde 1 (multitasking)	Runde 2 (focusing)
1 Name:	4 sek.	50 sek.	5 sek.
5 Namen:	20 sek.	60 sek	30 sek.

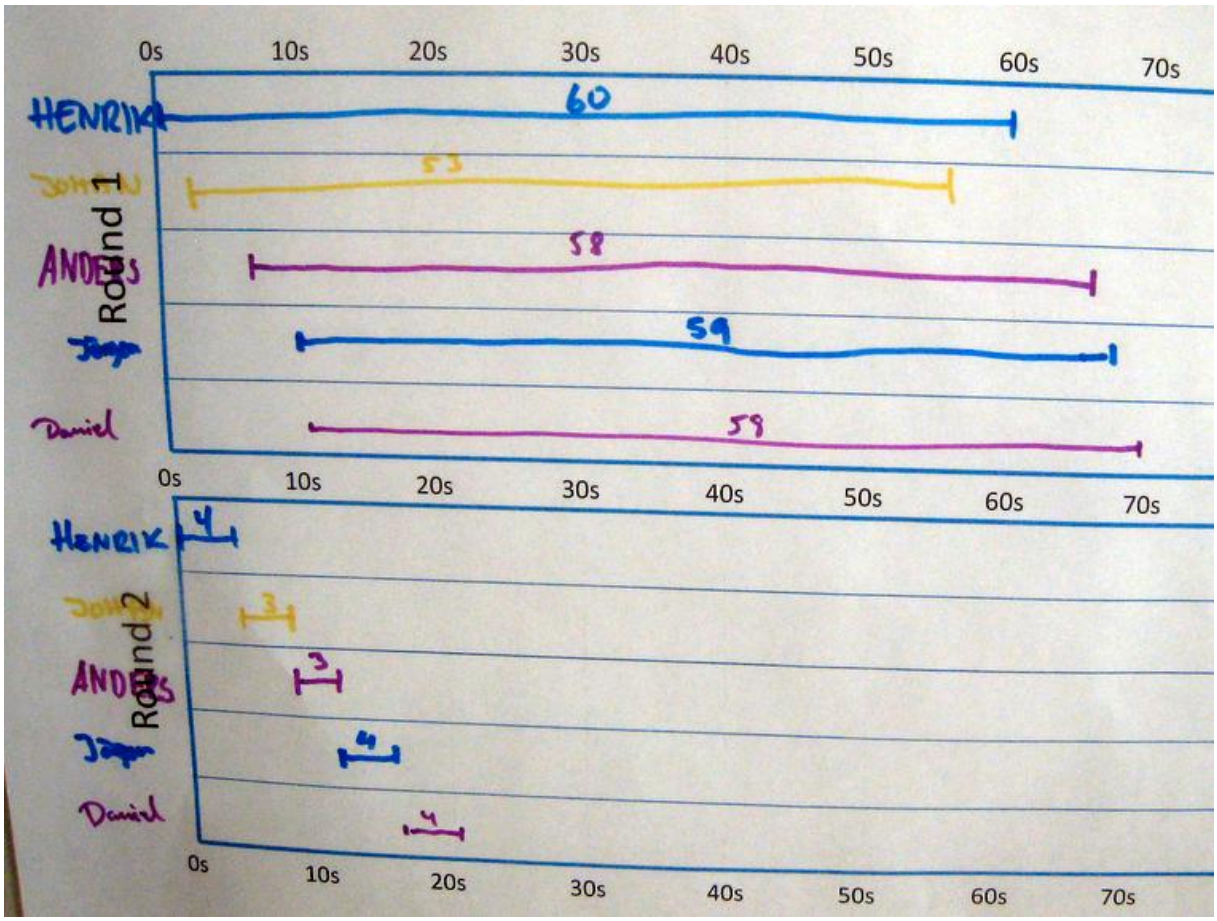
Ich zeige auf die Ergebnisse und frage nach dem Unterschied zwischen den beiden Runden.

In diesem Beispiel konnten wir jeden Auftrag zehn mal so schnell erledigen (von 50 sek. nach 5 sek.). Und da sich die Gesamtdauer für alle Aufträge halbiert hat (von 60 sek. nach 30 sek.) haben wir außerdem die Produktivität verdoppelt. In den 30 sek., die wir in Runde 2 eingespart haben, hätten wir 5 weitere Aufträge erledigen können, z.B. fünf neue Kunden, oder einen neuen Durchgang mit den fünf vorhandenen Kunden (z.B. könnten wir auch noch den Nachnamen schreiben).

So schlecht ist also Multitasking, sogar bei einer so einfachen Aufgabe wie Namen auf eine Karte zu schreiben. Und jetzt stellt euch vor, wir hätten etwas komplexeres getan, wie z.B. Software zu erstellen, wo das Wechseln zwischen einzelnen Aufgaben *viel* schwieriger ist! Diese Wechsel verursachen weitere Verzögerungen, Qualitätsprobleme und Stress. Ein echtes Projekt wird wohl eher in Wochen statt in Sekunden gemessen, aber die Verhältnisse sind oft ähnlich.

Wenn wir uns die ersten Schätzungen anschauen, dann können wir erkennen, dass die Ergebnisse von Runde 2 ziemlich genau dem entsprechen, was wir ursprünglich gedacht haben

Ich zeige dann noch dieses Bild, das ein sehr typisches Ergebnis darstellt. Wenn die Zeit reicht, dann lasse ich jeden Tisch sein eigenes Bild aufzeichnen.



Das Bild ist eine schöne Visualisierung für die Kosten von Multitasking. Ich frage die Teilnehmer dann, ob ihre eigenen Ergebnisse mit dem gezeigten Bild übereinstimmen, und normalerweise ist das der Fall.

An dieser Stelle moderiere ich eine Diskussion. Typische Fragen, die ich stelle und typische Antworten, die ich bekomme (oder selbst anbiete) sind:

Wie habt ihr euch als Entwickler und Kunde in diesen beiden Runden gefühlt?

- Runde 1 war für beide stressig und verwirrend, Runde 2 war ruhig und fokussiert.
- Als Kunde in der Runde 2 habe ich mich gut gefühlt, als ich mit dem Entwickler eine intensive Diskussion hatte, während er für mich arbeitete, und danach fertig war. In Runde 1 kam der Entwickler immer wieder zu mir und wiederholte Fragen wie ”Wie heißt du noch mal? Schreibt man das mit ph oder mit f?”

Wie hat es sich in Runde 2 angefühlt, der letzte Kunde zu sein, verglichen mit Runde 1?

- Es war in Ordnung, warten zu müssen, weil ich sehen konnte, wie schnell die anderen Projekte fertig wurden und ich konnte abschätzen, wann ich drankommen würde. In Runde 1 wurde mein Auftrag schnell begonnen, aber ich hatte keine Ahnung wann er fertig werden würde.

Was hältst du von der Aussage 'Wenn wir früher damit anfangen, werden wir auch früher fertig'?

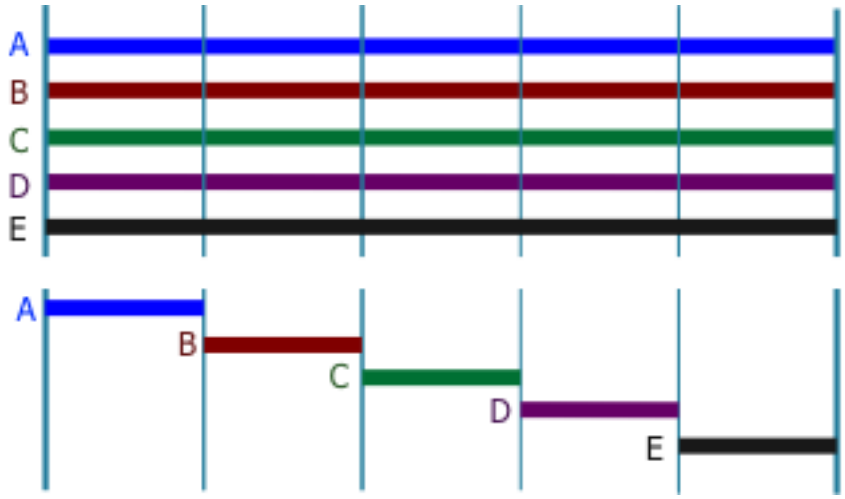
- Quatsch! Diese Übung zeigt das das Gegenteil richtig ist. In Runde 2 wurde jedes einzelne Projekt (außer dem ersten) später angefangen und trotzdem war jedes Projekt früher fertig. Die Projekte in Runde 2 wurden *deshalb* früher fertig, *weil* sie später begonnen wurden.

Wie beinflusst das die Releaseplanung? Z.B. was wissen wir nach 10 sek. in Runde 1 und in Runde 2?

- In Runde 1 haben wir nach 10 sek. keine Ahnung wann irgendetwas fertig wird. In Runde 2 haben wir nach 10 sek. fast schon zwei Projekte fertig und ausgeliefert, deshalb können wir zuverlässig abschätzen, wann die nächsten Projekte fertig werden.
- In Runde 1 ist die Schätzen eigentlich eher ein Ratespiel, selbst wenn alle anderen Einflussfaktoren (Länge des Namens, etc.) im Voraus bekannt sind, weil wir nie wissen, ob vielleicht ein weiteres Projekt in die Schlacht geworfen wird.
- In Runde 2 könnten wir ziemlich verlässlich schätzen (verglichen mit Runde 1) selbst wenn die folgenden Namen lang und kompliziert wären.

Was würde passieren, wenn der Entwickler perfekt darin wäre, zwischen den Aufträgen zu wechseln?

Die 30 sek. Produktivitätsverlust kamen vom "Task switching" (der Zeit, die durch das Wechseln zwischen den Aufträgen "verloren geht"). Was aber, wenn "Task switching" kostenlos wäre, wenn also der Entwickler zwischen den Aufträgen hin- und herwechseln könnte, ohne dass er Zeit verliert? In diesem Fall würde der Unterschied zwischen den beiden Runden ungefähr so aussehen:



Die Gesamtzeit, um alle Projekte zu erledigen ist gleich – es gibt in diesem Szenario also keinen Produktivitätsverlust. Aber jedes einzelne Projekt dauert mit Task switching immer noch fünf mal so lange wie ohne! Das liegt an Little's Law. Little´s Law sagt im Wesentlichen: wenn man x Sachen gleichzeitig tut, dann dauert jede Sache x mal so lang.

Deshalb sollte man sich nicht bemühen gute Multitasker zu finden. Stattdessen sollte man Leute einstellen, die Multitasking hassen und sich lieber darauf konzentrieren, eine Umgebung zu schaffen, die die Notwendigkeit von Multitasking vermindert – z.B. durch WIP-Limit.

Wie beeinflusst das die Produkt-Qualität?

- Wenn der Entwickler die Kundenwünsche missversteht (z.B. wenn er einen Namen falsch schreibt), dann würde das in Runde 1 50 sek. dauern, das herauszufinden, und nur 5 sek. in Runde 2. Und außerdem führt die höhere Produktivität in Runde 2 dazu, dass wir Zeit haben, den Namen iterativ zu verbessern, falls nötig. Das Multitasking hat demnach einen dramatischen Einfluss auf die Produkt-Qualität.

Kommt euch diese Multitasking-Problem bekannt vor? Wer macht gerade diese Erfahrung? Wer hat das schon mal erlebt?

So gut wie jeder hebt hier seine Hand ...

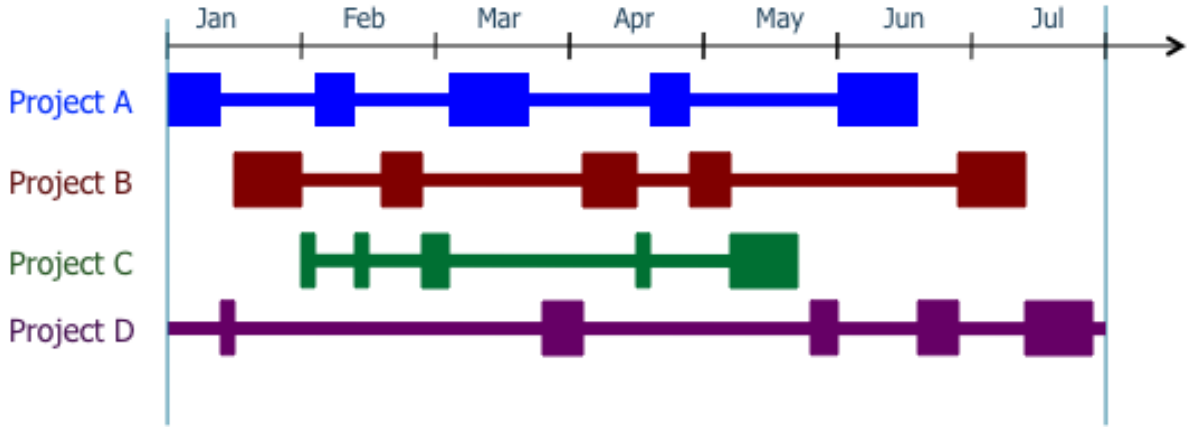
Was verleitet uns dazu so vorzugehen? Warum ist das Problem so weit verbreitet?

- Wir sind bestrebt, unsere Kunden zufriedenzustellen, und es ist einfacher zu einem neuen Kunden ”ja” zu sagen, als ihn zu bitten, ein wenig zu warten.
- Wenn wir in Runde 2 am Auftrag des Kunden Nr.1 arbeiten und Kunde Nr.3 erscheint und fragt, ob wir sein Projekt übernehmen können, dann fühlt es sich komisch an, wenn wir ihm sagen: ”Wir fangen mit deinem Projekt später an, damit es schneller fertig wird”. Obwohl es die Wahrheit ist.
- Wir konzentrieren uns zu sehr darauf, Dinge anzufangen, statt sie zu beenden. Beispielsweise indem wir Verkaufs-Boni bei die Vertragsunterzeichnung neuer Kunden vergeben, statt bei die Lieferung der Ware.
- Manchmal versuchen wir Kunden zu ”binden”, indem wir ihr Projekt früh beginnen, um das Risiko zu verkleinern, dass uns der Kunde abspringt. Aber das ist ein Verlust-Verlust-Versprechen. Es mag kurzfristig funktioneren, aber auf lange Sicht funktioniert diese Strategie nicht, wenn es intelligente Mitbewerber im Markt gibt. Der Markt wird früher oder später erkennen, dass die Projekte der Firma A 10 mal so lange brauchen und doppelt so teuer sind, als die von Firma B.

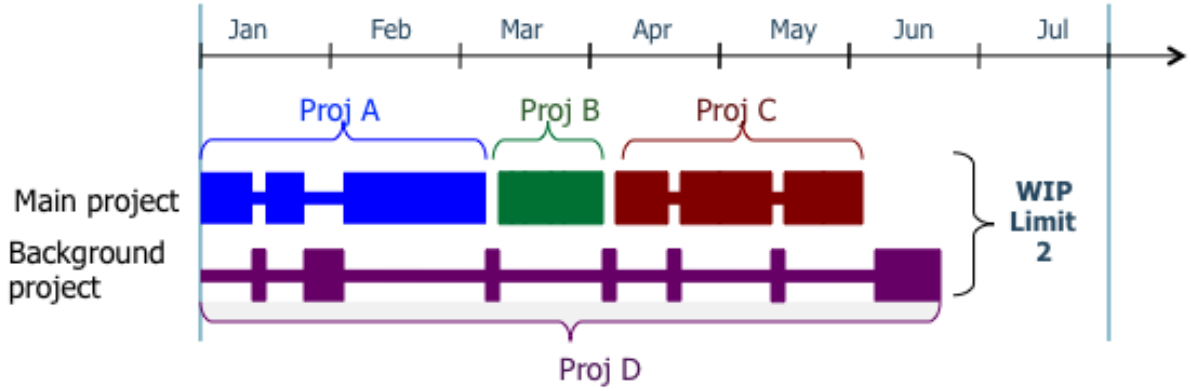
Wann ist Multitasking eine gute Idee? Wer profitiert in Runde 1?

- Multitasking ist so gut wie immer eine schlechte Idee. Niemand profitiert in Runde 1.
- Begrenztes Multitasking kann in Situationen nützlich sein, wenn Projekt A blockiert ist, weil wir auf etwas warten müssen, also arbeiten wir in der Zwischenzeit an Projekt B. Wir sollten dieses Vorgehen jedoch auf z.B. 2 parallele Projekte begrenzen. Dadurch erhöhen wir den Druck, die Hindernisse zu beseitigen, statt mit immer neuen Projekten zu beginnen.
- Ein wichtiges Lernziel ist es zu erkennen, dass obwohl WIP-Limits (Work in Process Limits) wichtig sind, das Limit nicht unbedingt 1 sein muss. Jedes Limit ist besser als kein Limit!

Hier ist ein Beispiel für ein Team, das gleichzeitig an vier Projekten arbeitet. Die dicken Bereiche zeigen, wann an jedem Projekt tatsächlich gearbeitet wurde.



Hier sieht man, was passsiert, wenn wir das WIP-Limit auf 2 Projekte setzen, ein "Haupt-Projekt" und ein "Hintergrunds-Projekt", an dem wir nur arbeiten, wenn das Haupt-Projekt blockiert ist. Im Haupt-Projekt gibt es einen Release-Plan und Commitments, im Hintergrunds-Projekt nicht.



Was kann man tun, um das Problem in seinem Umfeld zu bekämpfen?

Man kann einiges dagegen tun, unabhängig von seiner Rolle im Unternehmen.

- Das Problem visualisieren: Messwerte von durchgeführten Projekten sammeln und das obige Diagramm zeichnen.
- Multitasking messen: Bitte die Kollegen aufzuschreiben, in wie vielen Projekten sie mitarbeiten. Oder einen Zettel auf den Tisch kleben und jedesmal einen Strich machen, wenn man zu einem Task-Wechsel gezwungen wird. Diese Werte für das ganze Team sammeln und diskutieren, wie man sie reduzieren kann.
- Spiele das Multitasking-Namensspiel mit Kollegen, Managern, Kunden, usw.

Und nicht vergessen: der erste und allerwichtigste Schritt ist es, dass die Menschen erkennen, dass Multitasking ein Problem ist. Nur dann kann es effektiv bekämpft werden. Ich hoffe das dieser Artkel dabei hilft!

Skalieren dieser Übung

Diese Übung funktioniert am besten in einer kleinen Gruppe (ungefähr 10 bis 20 Leute), die an kleinen Tischen sitzen, da dieses Setting gute Diskussionen und ein gutes Debriefing ermöglicht. Jedoch funktioniert sie auch mit Gruppen von Hunderten von Teilnehmern. Wenn die Teilnehmer keine kleinen Gruppen um Tische bilden können (weil keine Tische zur Verfügung stehen oder weil es zu voll ist), dann kann man die Übung auf der Bühne machen.

Lade z.B. zwei Sechser-Gruppen ein, auf die Bühne zu kommen. Stelle jede Gruppe vor ein Flipchart. Statt Karten zu verwenden, um die Namen aufzuschreiben, schreibt der Entwickler auf das Flipchart und der Kunde schreibt die Liferzeit daneben, wenn der Auftrag fertig ist. Alles andere läuft so wie oben beschrieben.

Es geht auch, wenn man nur eine Gruppe einlädt, wenn man nur ein Flipchart zur Verfügung hat. Der einzige Nachteil dabei ist, das man die Entwickler nicht so einfach austauschen kann.

Varianten

Diese Übung kann natürlich auch erweitert oder angepasst werden um andere Punkte zu zeigen. Zum Beispiel: was passiert, wenn der Kunde sich mitten im Projekt umentscheidet? Oder er sich (nach der Lieferung) entscheidet, dass er eigentlich alles in Großbuchstaben, oder in grüner Schrift, haben will? Was wäre, wenn wir Cashflow-Aspekte einbeziehen, wie Bezahlung bei Lieferung, oder Return on Investment?

Ich habe noch nicht viel mit diesen Ideen experimentiert, bisher habe ich mich stark auf die Multitasking-Aspekte dieser Übung konzentriert.

Moderatoren-FAQ

Wäre es bei der initialen Schätzung nicht besser, ein Schätz-Intervall (4-6 sek.) aufzuschreiben, statt nur eine Zahl?

Eigentlich schon. Schätz-Intervalle sind besser, da sie das Unsicherheits-Level deutlich machen. Aber alle diese zusätzlichen Zahlen komplizieren die Übung indem sie das Flipchart schwerer lesbar machen. Es handelt sich hierbei um eine absichtliche Vereinfachung.

Warum schreibst du beim Sammeln der Ergebnisse nur den Median auf?

Früher habe ich alle Ergebnisse aufgeschrieben, aber das hat nicht gut funktioniert. Oft haben 10% der Teilnehmer ein ziemlich komisches Ergebnis, weil sie z.B. einen extrem kurzen Namen haben, oder weil jemandem der Stift heruntergefallen ist, oder weil der Entwickler sich verschrieben hat und den Namen noch einmal schreiben musste, odr weil der Kunde die falsche Zeit notiert hatte. Die maßgebliche Zahl ist der Median.

Ich habe auch erkannt, dass jedes zusätzliche Detail auf dem Flipchart die Aussage verwässert, weil die Teilnehmer sich dann mehr darauf konzentrieren, die Zahlen verstehen zu wollen, statt sich darauf zu konzentrieren, was sie gelernt haben. Deshalb halte ich es einfach.

Ich verwende den Median statt des Mittelwerts, weil er schneller zu ermitteln ist und man dafür keinen Taschenrechner braucht. Der Median von [12,20,24,25,50] ist 24. Der Mittelwert ist ... mhm ... was weiß ich.

Warum Namen? Warum nicht Obst, oder Farben, oder was anderes?

Namen sind praktisch weil jeder Kunde einen hat (ich muss sie nicht zuteilen) und manchmal gibt es nicht-offensichtliche Anforderungen (spezielle Schreibweisen usw.) die zu schönen Entwickler-Kunden-Konversationen und zu möglichen Fehlern führen können. Außerdem fühlt es sich mit Namen ”echter” an, die Teilnehmer können sich mit ”Ich brauche jemanden, der meinen Namen für mich schreibt” besser identifizieren, als mit ”Ich brauche jemanden, der mir diese Farbe aufschreibt”.

Und als positiven Nebeneffekt lernen die Teilnehmer gegenseitig ihre Namen :o)

Hat die Übung schon mal versagt?

Nicht dass ich mich erinnern könnte. Ab und zu habe ich einen Teilnehmer, darauf besteht, dass die Runde 1 eine bessere Verkaufsstrategie darstellt, und dass es eine gute Idee ist, dn Kunden so zu

binden. Das führt aber normalerweise zu starken Protesten der anderen Teilnehmer und einigen sehr interessanten Diskussionen über kurzfristiges und langfristiges Denken, dehalb würde ich das nicht versagen nennen.

Ab und zu versteht eine Gruppe die Anweisungen nicht und hält sich in Runde 1 nicht an die Unternehmens-Richtlinie, was ihre Statistik durcheinanderbringt. Aber dann verwende ich einfach die Statistiken der anderen Gruppen. Oder ich wiederhole die Runde.

Manchmal gibt es einen Entwickler der ein extrem guter Multitasker ist, das kann die Ergebnisse ein bischen verwässern. Das war aber bisher noch kein größeres Problem.

Manchmal wird ein Kunde übermäßig enthusiastisch fängt an ”schwieriger Kunde” zu spielen. Er ändert die Anforderungen und stellt pedantische Qualitätsanforderungen. Wenn ich das merke, fordere ich ihn freundlich auf, mitzuspielen.

Manchmal beschwert sich jemand, dass die Übung zu stark vereinfacht ist und dass dieser Multitasking-Effekt in ”echten” Projekten nicht auftritt. Die anderen Teilnehmer werden normalerweise starke Gegenargumente dazu liefern, indem sie auf ihre eigenen Erfahrungen verwisen. Es macht Spaß, das zu beobachten.

Am Schluss erinnere ich die Teilnehmer, dass das nur eine Übung ist, sie ist per Definition künstlich. Es bleibt jedem Einzelnen überlassen, zu entscheiden, was man daraus lernen will und wie es sich in den eigenen Kontext übertragen lässt.