

Wieviel Demokratie vertragen Prozesse?

Eine Alternative zur Prozesspolizei: Wie kann man Prozesse am Leben und lebbar halten?

Andreas Stucki, Solcept AG

Prozesse haben einen schlechten Ruf, nicht nur weil der Name an einen Gerichtsprozess erinnert. Sondern auch, weil Ihre Definition und Implementation häufig von Menschen getrieben werden, die der praktischen Anwendung organisationsbedingt fremd sind. Wenn ihre Durchsetzung dann auch noch mit Massnahmen erfolgt, die den Betroffenen keine Wahl lassen, dann ist klar: Prozesse behindern uns nur!

Wirklich? Ist das die Absicht dahinter? Kann man Prozesse auch anders definieren und leben, so dass sie zur Hilfe werden? Ich glaube schon, wir tun das seit mehr als zehn Jahren... Dazu haben wir für uns in diesen Jahren folgende sechs Fragen beantwortet:

Innovation und «Prozesse»: Ist das nicht ein Gegensatz?

Behindern uns Prozesse nicht in der Innovation, in der Kreativität? Wir glauben nicht, dass dies so ist. Denn wir bilden das Alltagsgeschäft in Prozessen ab, nicht die Innovation. Dadurch, dass es Namens- und Ablageregeln gibt, muss ich bei der Erstellung und vor allem bei der Suche nicht lange Zeit verlieren; dadurch, dass es eine Beschreibung und Vorlagen für Softwarewerkzeuge gibt, muss ich diese nicht erst erfinden usw. So schaffen Prozesse mehr Zeit und Energie für Innovationen, welche meiner Meinung nach sowieso nicht in standardisierten Prozessen abgebildet werden können.

Und dann ist hier auch noch die Frage von Kernkompetenzen (siehe z.B.[kernkompetenzen]), von Organisationswissen. Diese Kernkompetenzen können hoffentlich einiges zur Innovation beitragen. Nun ist Organisationswissen Handlungswissen, Wissen *wie* man etwas macht. Zusätzlich ist Technologie ein *Team* sport, also müssen wir alle wissen, wie wir zusammen etwas machen.

Und genau das bilden Prozesse ab: *wie* arbeiten *wir* hier, was ist unsere Arbeitsweise. Und dieses Können verbessern wir immer wieder, indem wir *als Organisation* aus Fehlern lernen, z.B. indem wir mit den Erkenntnissen aus unseren Fehlern unsere Checklisten anpassen.

Was wollen wir mit Prozessbeschreibungen überhaupt erreichen?

Was sind Prozesse eigentlich? Prozesse sind Arbeit, d.h. Wege wie wir ein Ziel erreichen. Das was im Qualitätshandbuch steht («Prozessbeschreibungen») sind Arbeitsbeschreibungen: «so tun wir das hier» und nicht die Prozesse d.h. die Arbeit selbst! Daher ist es manchmal hilfreich, «Prozess» durch «Arbeit» zu ersetzen.

Es gibt verschiedene Absichten, welche mit «Prozessen» verfolgt werden können:

Zuerst *Effizienz*, denn Arbeitsweisen, welche niedergeschrieben sind, führen dazu, dass nicht jede und jeder das Rad neu erfinden muss. Z.B. wie man ein Projekt effizient leitet, wie man eine Software-Architektur möglichst fehlerarm erstellt.

Dann *Qualität*, indem wir nach Arbeitsbeschreibungen, mit Vorlagen und Checklisten arbeiten, welche das Wissen aller bisherigen Projekte in sich vereinen.

Und auch *Konformität*, mit Anforderungen der funktionalen Sicherheit und je länger je mehr auch mit denen von Cybersicherheit: «Manage risk with quality».

Der wohl unerwartetste Aspekt ist die *Führung*. Funktionierende (nur solche!) Prozesse erlauben es, die Sachlogik der Organisation über Prozessänderungen zu beeinflussen. Und als Zusatznutzen können Prozessbeschreibungen bis in die Psychologie ausstrahlen.

Welche Prinzipien liegen einer Prozesslandschaft zugrunde, die diese Ziele erreicht?

Das wichtigste Grundprinzip ist das *Vertrauen*, das Überlassen der Kontrolle über die Arbeitsweise an die Arbeitenden (Intent-based-Leadership, z.B. [ibl]). Daraus folgt, dass für die Pflege der Prozess-Assets die Pfadfinderregel gilt: wer einen Fehler oder eine Ineffizienz findet, der korrigiert diese sofort und selbst oder stösst selbst die Korrektur an (siehe unten: PI).

Im täglichen Gebrauch ist jeder für die Prozesseinhaltung selbst verantwortlich. Für Informationen, auch über meine Arbeitsweise gilt das Holprinzip: ich bin dafür verantwortlich, dass ich alle Informationen für die Erreichung meiner Ziele habe.

Und sehr wichtig gegen Prozess Over-Engineering und zu komplizierte Prozessbeschreibungen ist der Grundsatz, dass die Arbeitsbeschreibungen nur *80% der Arbeitsfälle* abdecken müssen: Arbeitsbeschreibungen sind keine Software, die jeden Corner-Case abdecken muss! Spezialfälle, die unter die restlichen 20% fallen, werden vom Team im Projekt selbst entschieden.

Welche Implementation unterstützt diese Prinzipien?

Wir nutzen ein einfaches und flexibles Beschreibungsformat: *ETVX* (Entry-Task-Verification-Exit).

In *E (Eingang)* wird der Trigger für den Prozess definiert plus die Rollen und Werkzeuge (inkl. Vorlagen) für diese Arbeitsbeschreibung. Auch sind die Eingangsartefakte für diese Arbeitsschritte sowie deren Qualitätsanforderungen angegeben.

T (Aufgaben) und *V (Verifikation)* beschreiben was die Rollen wie tun sollen und wie das Resultat verifiziert wird, durch z.B. Reviews.

X (Ausgang) enthält die Artefakte, die von der Arbeit generiert werden und deren Qualitätsmerkmale (z.B. wie formell und durch wen deren Reviews durchgeführt werden). Zusätzlich sind hier die nachfolgenden Prozesse definiert, welche die in

diesem Schritt produzierten Artefakte konsumieren.

Bevor wir eine Prozessbeschreibungen erstellen, schreiben wir die Absicht dieses Prozesses auf. Diese steht dann am Anfang jeder Prozessbeschreibung als grundsätzliche Orientierung. Sie besteht aus der Policy, einer kurzen Leitlinie, was die Prozessbeschreibung erreichen soll und der Rationale, einer Beschreibung, was das Ziel des Prozesses ist.

Wir nutzen eine textbasierte Beschreibung, keine graphischen Werkzeuge. Der Grund dafür ist, dass es einfacher ist, Texte zu ändern, zu warten oder auch einmal miteinander zu vergleichen. Diese textuellen Prozessbeschreibungen und die dazugehörigen Vorlagen sind in einem Wiki abgelegt.

Das hat den Vorteil, dass sie einfach von jedem zu ändern sind. Nun höre ich schon den Aufschrei: Da kann ja jeder etwas löschen, das ihm zu mühsam ist! Das stimmt. Nur gibt es im Wiki eine History, welche jede Änderung nachvollziehbar macht. Und noch wichtiger ist, dass so etwas in unserer nun zwölfjährigen Geschichte nie vorgekommen ist...

Wie stellen wir sicher, dass die Prozessbeschreibungen ohne eine zentrale Qualitätssicherung eingehalten werden?

Dies ist eine wichtige Frage, die auch jeder neue Auditor stellt. Die Antwort ist einfach, sie nennt sich Process Impact Monitoring (PIM) Prozess. Dabei findet eine time-boxed Peer-Review mit einem Kollegen statt. In dieser wird eine Prozessbeschreibung, bzw. deren Ausführung genau angeschaut.

Die Überprüfung basiert auf einer Checkliste für jeden Prozess und beinhaltet eine Review von Artefakten (Dokumente, Sourcen, Schemas...) und ein Interview mit dem Kollegen. Wir machen das alle Quartale, jeweils ein Mitarbeiter reviewt den anderen über ein paar Prozesse, der andere den einen über andere Prozesse.

PIM hat drei Effekte, primär die *Überprüfung* der Prozesseinhaltung, sekundär und ebenso wichtig einen *Lerneffekt* (ich muss mich mit der Arbeitsbeschreibung befassen, damit ich deren Einhaltung beurteilen kann) und tertiär eine *Prozessverbesserung*. Denn wenn der Prozess nicht eingehalten wurde, liegt es dann immer am Prozess? Oder ist die Prozessbeschreibung so nicht lebbar und muss angepasst werden, wie im nächsten Kapitel beschrieben?

Wie halten wir die Prozessbeschreibungen lebbar und implementieren eine dauernde Verbesserung, die allen Betroffenen etwas bringt?

Sobald jemand sieht, dass eine Prozessbeschreibung ineffizient oder nicht lebbar ist, dann kann er den Performance Improvement (PI) Prozess anstossen. In einem ersten Schritt wird in einem Ticket festgehalten, was das Problem ist, wenn möglich sogar gerade ein Vorschlag zur Lösung.

Im PI-Meeting, das je nach Bedarf ungefähr alle zwei Monate stattfindet, werden diese Tickets dann von allen Interessierten diskutiert. Es gibt eine Agenda mit

allgemeinen und spezifischen (Projektleitung, Software, Hardware) Themen, grundsätzlich können alle an jedem Ticket mitdiskutieren und vor allem können *alle an allem mitentscheiden*. Für grössere Organisationen wäre eine solche «Vollversammlung» wohl nicht praktikabel, dann könnte man mit einem Delegiertensystem arbeiten.

Für die Änderungen gibt es Impact-Kriterien: Steigt der Wert für den Kunden? Können wir mehr mit weniger tun? Wird die Firmenkultur verbessert? Werden Mura (Ungleichmässigkeit), Muri (Überlast) und Muda (Verlust) minimiert? Und eine Abschätzung der Amortisation: zahlt sich der Änderungsaufwand innert zwei Jahren zurück?

Direkt während der Sitzung wird definitiv entschieden Es gibt nur zwei Entscheide: *kill* oder *accelerate*, wir machen es nicht oder jemand übernimmt die Verantwortung, die Änderung zu implementieren. Der Prozessverantwortliche und der Geschäftsführer hätten ein Vetorecht, v.a. damit die normenspezifischen Prozesse konform und damit die Prozesse mit der Firmenstrategie aligniert bleiben.

Dies führt dazu, dass unsere Prozesslandschaft nie fertig ist, sondern eigentlich immer nur den «aktuellen Stand des Irrtums» definiert. Und wir denken, das ist gut so...

Es klingt verwegen, aber aus meiner Sicht ist dies einer der wenigen Wege, um Prozesse und vor allem deren Anpassung an die Aufgaben so zu gestalten, dass sie ein Aktivposten werden und nicht einfach eine Bürde, die man auch noch erledigen muss.

Was ist Ihre Meinung, lassen sie es mich wissen: a.stucki@solcept.ch.

Referenzen

[kernkompetenzen] Solcept: "Q&A: Kernkompetenzen", 2022, (download: 2022-09-29), <https://www.solcept.ch/de/qa/auswahlkriterien/kernkompetenzen/>

[ibl] A. Stucki: "Von roter und blauer Arbeit", 2021, ESE Kongress 2021

Autor



Andreas Stucki, Dipl. El.-Ing. ETHZ, Geschäftsführer von Solcept AG, einer Ingenieur-Boutique für Produkt-Entwicklung und Beratung für komplexe, kritische und vernetzte embedded Systeme. Er hat eine 30-jährige Erfahrung mit embedded Soft- und Hardware Projekten im industriellen und kommerziellen Umfeld, von Forschung bis funktionale Sicherheit.

Kontakt: a.stucki@solcept.ch, www.solcept.ch