

2.

Auflage



Stefan Hofer • Henning Schwentner

Domain Storytelling

Ein Einstieg

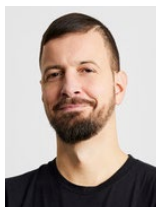


dpunkt.verlag

Domain Storytelling



Stefan Hofer hat in Österreich Software Engineering studiert und einen Dokortitel in Informatik an der Universität Hamburg erworben. Seit 2005 arbeitet er für die WPS – Workplace Solutions GmbH. Requirements Engineering und Domain-Driven Design bilden seine Themenschwerpunkte. Stefan ist auf Mastodon (@hofstef@social.wps.de) und per E-Mail (stefan@domainstorytelling.org) erreichbar.



Henning Schwentner beschäftigt sich mit Computern, seit er Anfang der 90er-Jahre einen Amiga 500 zum Geburtstag bekam. Er hatte das Glück, diese Leidenschaft zum Beruf zu machen, und arbeitet als Coder, Coach und Consultant bei WPS – Workplace Solutions. Er hilft Teams dabei, Struktur in ihre bestehende Software zu bringen oder neue Systeme mit einer nachhaltigen Architektur von Grund auf aufzubauen. Henning ist der Autor von *Domain-Driven Transformation* und LeasingNinja.io, der Übersetzer von *Domain-Driven Design kompakt* und Mitorganisator des ComoCamps. Henning ist stolzer Vater von sechs Kindern in einer ganz besonderen Patchwork-Situation. Er schreibt auf <https://hswentner.io> und liest E-Mails, die an henning@domainstorytelling.org gerichtet sind.

Copy Editing: Ursula Zimpfer, Herrenberg

Satz und Herstellung: Frank Heidt

Umschlaggestaltung: Silke Braun

Druck: www.wp-consult.eu, Mannheim

Artikel-Nr. 077.95746

2., überarbeitete Auflage 2024

Copyright © 2024 dpunkt.verlag GmbH

Wiebinger Weg 17

69123 Heidelberg

Die vorliegende Publikation ist urheberrechtlich geschützt. Alle Rechte vorbehalten. Die Verwendung der Texte und Abbildungen, auch auszugsweise, ist ohne die schriftliche Zustimmung des Verlags urheberrechtswidrig und daher strafbar. Dies gilt insbesondere für die Vervielfältigung, Übersetzung oder die Verwendung in elektronischen Systemen.

Es wird darauf hingewiesen, dass die im Buch verwendeten Soft- und Hardware-Bezeichnungen sowie Markennamen und Produktbezeichnungen der jeweiligen Firmen im Allgemeinen warenzeichen-, marken- oder patentrechtlichem Schutz unterliegen.

Alle Angaben und Programme in diesem Buch wurden mit größter Sorgfalt kontrolliert. Weder Autor noch Verlag können jedoch für Schäden haftbar gemacht werden, die in Zusammenhang mit der Verwendung dieses Buches stehen.

5 4 3 2 1

Stefan Hofer • Henning Schwentner

Domain Storytelling

2., überarbeitete Auflage 2024

Vorwort

Was wir seit 2016 als Domain Storytelling bezeichnen, hat eine mehr als 20-jährige Geschichte. Diese begann an der Universität Hamburg unter dem Namen »Kooperationsbild« und später »exemplarische Geschäftsprozessmodellierung«. Viele Menschen haben zu diesen Methoden beigetragen und wir sind ihnen dankbar dafür.

Besonders herzlich bedanken möchten wir uns bei Prof. Dr.-Ing. Heinz Züllighoven. Er hat an der Entstehung von Domain Storytelling und seinen Vorläufern mitgewirkt und uns dabei unterstützt, die erste Auflage dieser Broschüre zu schreiben.

Der Untertitel verrät es, diese Broschüre möchte *ein Einstieg* sein. Wir hoffen, dass sie Lust macht, tiefer in die Methode einzutauchen. Wenn das gelingt, sei Ihnen unser Buch *Domain Storytelling: Gemeinschaftlich, visuell und agil zu fachlich wertvoller Software* [Hofer & Schwentner 2023] als weitergehende Lektüre ans Herz gelegt. In dessen Anhang findet sich auch ein kurzer Abriss der Geschichte von Domain Storytelling.

Stefan Hofer, Henning Schwentner
Hamburg, Oktober 2024
<https://domainstorytelling.org>

1 Domain Storytelling – kurzgefasst

Domain Storytelling ist eine kollaborative Modellierungstechnik, die verdeutlicht, wie Menschen zusammenarbeiten. Ihr Hauptzweck ist, Fachwissen in Unternehmenssoftware zu überführen. Dazu kommen Menschen mit unterschiedlichem Hintergrund zusammen und lernen durch das Erzählen und Visualisieren von Geschichten voneinander.

1.1 Was nützt mir Domain Storytelling?

Um gute Unternehmenssoftware zu entwickeln, müssen die Entwicklungsteams die Fachlichkeit verstehen. Fachleute und Entwicklungsteams brauchen eine gemeinsame Sprache, um über die Anforderungen an die Software zu sprechen. Denn letzten Endes spiegelt eine Software nur das wider, was die Entwicklerinnen und Entwickler von den Anforderungen und der Domäne verstanden haben – und nicht, was jemand in einem Anforderungsdokument oder in Jira aufgeschrieben hat.

Ein Beispiel: Wir entwickelten Software auf Basis eines Lastenhefts, das Hunderte funktionale Anforderungen enthielt. Jede Anforderung war gut und ausführlich beschrieben. Jedoch blieb offen, wie die einzelnen Funktionen zusammenwirken sollten. Die Autoren des Lastenhefts hatten zwar eine Vorstellung von den zukünftigen Arbeitsabläufen, konnten sie aber nicht vermitteln. Als wir im Projekt die Arbeitsabläufe gemeinsam betrachteten, zeigte sich, dass diese nicht ausreichend durch die zu bauende Software unterstützt wurden.

Solche Probleme begegnen uns immer wieder: Das Management, die Business-Analysten und das Entwicklungsteam haben die tatsächlichen Aufgaben und Abläufe im Unternehmen nicht wirklich verstanden. Anforderungsdokumente und Prozessmodelle hin- und herschicken, löst die Probleme nicht. Und wenn die Beteiligten einmal direkt miteinander sprechen, dann reden sie aneinander vorbei.

In der Informatik galt lange die Auffassung, dass präzise und möglichst formale Anforderungsspezifikationen dieses Verständnisproblem beseitigen könnten. Doch unsere Erfahrungen in der Anwendungsentwicklung über viele Jahre zeigen, dass die Formalisierung die Kluft zwischen Geschäftsanwendungen und deren Anwendern nur noch vergrößert hat.

Dies ist wohl auch ein wesentlicher Grund, weshalb als Gegenbewegung dazu agile Softwareentwicklung, Domain-Driven Design und Behavior-Driven Development entstanden sind. Diese Ansätze betonen besonders das Feedback und die Einbeziehung von Anwenderinnen und anderen Stakeholdern. Doch auch bei diesen Vorgehensweisen braucht es Methoden, um das fachliche Verständnis zwischen den Beteiligten einfach und schnell zu fördern.

Dabei gibt es ein altbewährtes Mittel, um Wissen und Erfahrungen weiterzugeben: Geschichten erzählen! Diese grundlegende Form menschlicher Kommunikation ist tief in uns allen verwurzelt. In unserer modernen Welt mag das Geschichtenerzählen archaisch oder naiv erscheinen. Aber überlegen Sie doch einmal: Beim Mittagessen mit Kolleginnen und Kollegen oder bei Vorträgen auf Konferenzen ist das Erzählen von Geschichten ein wirksames Mittel, um Erfahrung weiterzugeben.

Die Broschüre soll Sie dazu anregen, diese Kommunikationsform auch bei der Entwicklung geschäftskritischer Software für Logistik, E-Commerce, Banken und andere Domänen zu nutzen. Storytelling ist ein Mittel, um Fachwissen aus den Köpfen der Fachleute in die Köpfe von Entwicklerinnen, Testern, Product Ownern, Produktmanagerinnen und Business-Analysten zu transportieren – zu allen, die an der Softwareentwicklung beteiligt sind.

Dabei teilen wir unsere Geschichten nicht am Lagerfeuer, sondern versammeln uns vor dem Whiteboard. Die Fachleute sind unsere Geschichtenerzähler. Wir wollen, dass sie uns echte Geschichten aus ihrem Arbeitsalltag erzählen. Keine abstrakten Verallgemeinerungen, kein hypothetisches Hätte und Könnte, sondern konkrete Beispiele dafür, was tatsächlich passiert, damit wir gemeinsam darüber sprechen können. *Domain Stories* eben.

1.2 Ihre erste Domain Story

Sicher haben Sie schon einmal am Flughafen auf den Bus zum Flieger gewartet. Ist Ihnen dabei die Frage durch den Kopf gegangen, wie eigentlich der Bustransfer vom Gate zum Flugzeug organisiert wird? Es gibt immer noch Flughäfen, bei denen ein Bustransfer viele manuelle Arbeitsschritte erfordert.

Gehen wir vom einfachsten Fall aus – für den Transfer wird nur ein Bus benötigt: Am Abflug-Gate ruft der Gate Agent (das ist die Person, die das Boarding durchführt) eine Busdisponentin an. Der Gate Agent bestellt bei ihr einen Bus und gibt die Anzahl der Passagiere, das Abflug-Gate und den Flug an. Die Busdisponentin erzeugt einen Auftrag in ihrem Bus-Dispositionssystem. Diesen Auftrag vergibt sie per Telefon an einen Busfahrer. Der Busfahrer weiß dann, wie viele Passagiere er wann von A nach B zu bringen hat. Doch damit ist die Arbeit noch nicht getan. Während des Transports protokolliert der Busfahrer die Details zum Fahrtverlauf (u.a. ob es Störungen auf dem Weg zum Flugzeug gab). All das notiert der Busfahrer per Hand. Zum Schichtende gibt der Busfahrer die gesammelten Protokolle bei der Busdisponentin ab, die dann die Details zum Fahrtverlauf im Dispositionssystem erfasst (siehe Abb. 1–1).

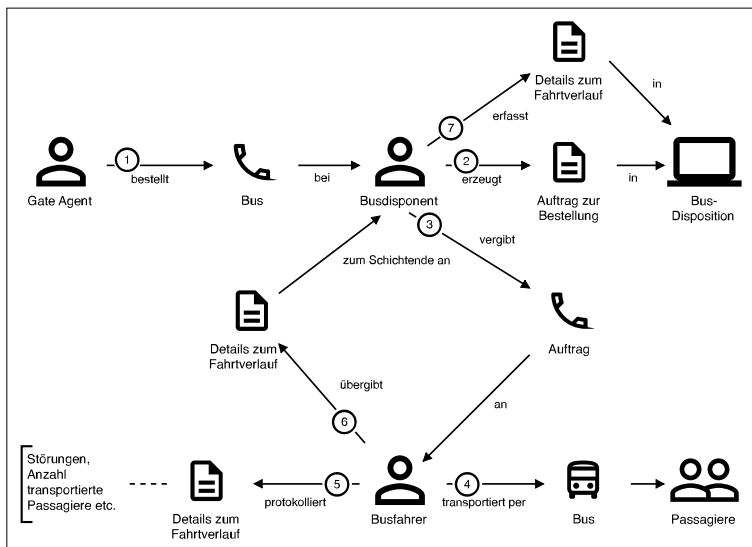


Abb.1–1 Bustransfer vom Gate zum Flugzeug organisieren

Stellen Sie sich vor, diesen Text hätten uns ein Gate Agent, eine Busdisponentin und ein Busfahrer erzählt, während wir gleichzeitig das Bild aus Abbildung 1–1 auf einem Whiteboard festhalten. Das war Ihre erste Domain Story!

1.3 Bildsprache + Workshop = Domain Storytelling

Domain Storytelling nutzt eine einfache Bildsprache mit Icons¹, Pfeilen und Text:



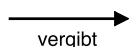
Busdisponent

- *Akteure* sind die handelnden Personen, Organisationseinheiten und IT-Systeme einer Geschichte. Menschliche Akteure benennt man üblicherweise mit ihrer Rollenbezeichnung (z.B. Busdisponentin).



Auftrag

- *Arbeitsgegenstände* repräsentieren etwas, an/mit dem ein Akteur arbeitet. Das können physische Dinge wie der »Bus« sein, Informationen oder Konzepte wie der »Auftrag«.



vergibt

- *Aktivitäten* beschreiben, was die Akteure tun, und werden durch Pfeile ausgedrückt. Die Aktivitäten sind die Verben in unseren Geschichten.



- Mit *Sequenznummern* werden die Sätze einer Domain Story in eine Reihenfolge gebracht.

Während die Fachexperten ihre Geschichte erzählen, zeichnet ein Modellierer die Geschichte auf. Damit ist Geschichtenerzählen nicht länger nur eine mündliche Aktivität. Es ist zu einer Modellierungsaktivität geworden. Während die Beteiligten diskutieren, hilft das Modell, Widersprüche und »Handlungslücken« in der Geschichte aufzudecken. Alle Beteiligten sehen, wie sich das Modell zusammen mit der Geschichte entwickelt. Das

1. Für die Beispiele in dieser Broschüre verwenden wir Googles frei verfügbare Material Icons (<https://fonts.google.com/icons/>).

macht es einfach, Fragen zu stellen und einen Beitrag zur Geschichte zu leisten. Versuchen Sie es selbst: Schauen Sie sich Abbildung 1–1 noch einmal an. Stellen sich Ihnen auch die folgenden Fragen?

- Wie entscheidet die Busdisponentin, welchem Busfahrer sie einen Auftrag gibt?
- Warum muss sie die Details zum Fahrtverlauf in das Dispositionssystem eingeben?

Domain Storytelling bringt Menschen in Workshops zusammen. Die Ziele werden vor dem oder zu Beginn des Workshops festgelegt. Die teilnehmenden Personen haben einen unterschiedlichen Hintergrund und verschiedene Aufgaben. Üblicherweise kommen sie aus Fachabteilungen und aus der IT.

Das grafische Modell ist ein wichtiges Ergebnis eines Workshops, denn es dokumentiert das gemeinsame Verständnis der Beteiligten. Mindestens genauso wichtig ist das Gespräch, das zu dem Modell geführt hat, dann dadurch lernen die Personen voneinander und verbessern ihre Zusammenarbeit. Konversation und Modell entfalten nur zusammen ihre volle Wirkung. Anders kann aus unserer Sicht kein wechselseitiges Lernen und Verstehen zustande kommen.

2 Eine Domain Story entsteht

Von einem guten Beispiel kann man mehr lernen als von einer schlechten Abstraktion. Schauen wir uns deshalb ein ganz konkretes Beispiel an.

2.1 Die Vorgeschichte

Karl betreibt ein kleines Kino für Arthouse-Filme namens Metropolis, das unter Cineasten einen hervorragenden Ruf genießt. Lokales Craft-beer und Biosnacks runden das Kinoerlebnis ab. Eines Tages trifft Karl seine Schulfreundin Anna. Als er erfährt, dass Anna seit fast zehn Jahren Apps entwickelt, hat er eine Idee.

Kinobetreiber Karl: »Meine Kundschaft mag den altmodischen Charme meines Kinos. Aber was sie nicht mag, ist meine altmodische Kinokasse. Die Kinobesucher von heute sind es nicht mehr gewohnt, die Eintrittskarten persönlich an der Kasse zu kaufen. Meine Kunden fragen mich, warum sie die Karten nicht online kaufen können. Kannst du nicht eine App für mich entwickeln?«

App-Entwicklerin Anna: »Du betreibst nur ein Kino und es gibt nur eine Handvoll Filme pro Woche, zwei oder drei Vorstellungen pro Tag. Klingt einfach.«

Karl: »Toll! Aber noch eine Kleinigkeit: Wir zeigen in unserem Programm auch internationale Filme in Fremdsprachen. Außerdem brauche ich neben dem Online-Verkauf über die App noch die Kinokasse für weniger technikaffine Kinobesucher. Und ich möchte, dass die Nutzer der App ein Jahresabonnement abschließen können.«

Anna: »Abonnements? Online- und Offline-Verkauf? Vorstellungen in Fremdsprachen? Das ist komplizierter, als ich dachte ...«

2.2 Der Workshop

Am nächsten Tag treffen sie sich in Karls Büro wieder. Sie stehen vor einem Whiteboard und Anna hält einen Marker in der Hand.

App-Entwicklerin Anna: »Gestern hast du gesagt, dass die App im Wesentlichen drei Anwendungsfälle hat. Erstens Verkauf von Standardeintrittskarten, zweitens Verkauf von speziellen Karten für fremdsprachige Filme und drittens Abschließen eines Jahresabonnements.«

Kinobetreiber Karl: »Äh, ja, das stimmt.«

Anna: »Ich würde gerne verstehen, wie das Metropolis heute funktioniert. Das wird mir helfen, eine App zu entwickeln, die deine Anforderungen erfüllt. Könntest du mir bitte erklären, wie Eintrittskarten an der Kasse verkauft werden?«

Karl: »Klar. Man verkauft die Karten und markiert den Sitzplatz im Saalplan und ...«

Anna: »Moment, Moment. Wer verkauft die Karten?«

Karl: »Ich habe zwei Studenten, die für mich arbeiten. Aber manchmal mache ich es auch selbst.«

Anna: »Okay, aber welche Rolle hast du oder die Studenten dann?«

Karl: »Ach so, bei uns heißt das Kassierer.«

Anna zeichnet ein Strichmännchen auf das Whiteboard und schreibt »Kassierer« darunter (siehe Abb. 2-1).



Abb. 2-1 Der erste Akteur

Anna: »Und wer kauft die Karten?«

Karl: »Ein Kinobesucher. Also einer ohne Abonnement.«

Anna zeichnet ein zweites Strichmännchen und benennt es »Kinobesucher«. Daneben notiert sie, dass der Kinobesucher kein Abonnement hat (siehe Abb. 2–2).

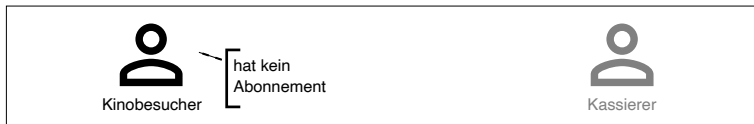


Abb. 2–2 Der zweite Akteur und die erste Notiz

Anna: »Was muss ein Kinobesucher tun, um eine Karte zu kaufen?«

Karl: »Er nennt dem Kassierer die Vorstellung, die er sehen will.«

Anna: »Ich zeichne hier eine Sprechblase als Icon für die Vorstellung, weil die beiden miteinander reden.«

Anna zeichnet weiter und nummeriert den Pfeil (siehe Abb. 2–3).

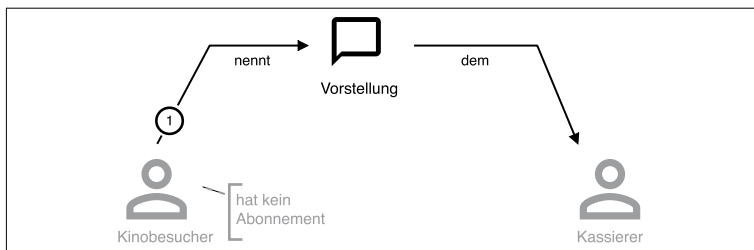


Abb. 2–3 Die erste Aktivität

Anna: »Und dann?«

Karl: »Normalerweise schlägt der Kassierer den besten verfügbaren Sitzplatz vor.«

Anna: »Wie funktioniert das?«

Karl: »Ich nehme den Saalplan für die Vorstellung und suche nach verfügbaren Plätzen. Im Saalplan kann ich sehen, welche Plätze bereits verkauft wurden und welche noch verfügbar sind.«

Anna zeichnet und erklärt die Icons.

Anna: »Hier verwende ich ein Film-Icon anstelle der Sprechblase, um die Vorstellung zu symbolisieren.«

Karl: »Der Saalplan ist ein Raster. Kannst du ein Raster zeichnen?« (Siehe Abb. 2-4.)

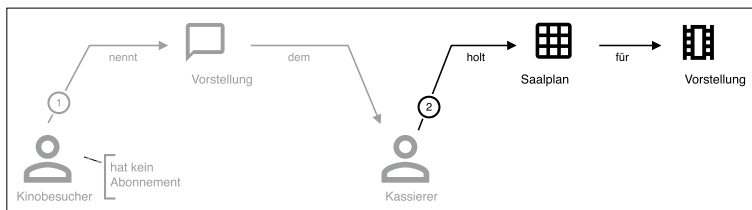


Abb. 2-4 Die zweite Aktivität

Dann liest Anna vor, was sie verstanden hat.

Anna: »Zweitens holt der Kassierer den Saalplan für die Vorstellung. Drittens: Er sucht im Saalplan nach verfügbaren Sitzplätzen. Stimmt das so?« (Siehe Abb. 2-5.)

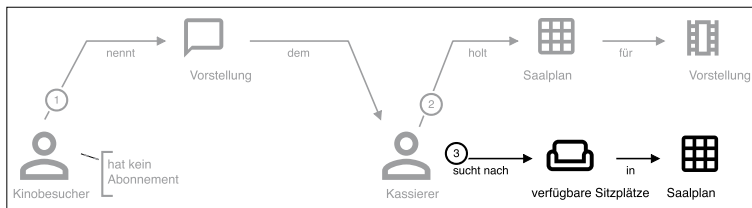


Abb. 2-5 Die dritte Aktivität

Karl nickt zustimmend.

Anna: »Und jetzt schlägt der Kassierer dem Kinobesucher die verfügbaren Plätze vor?«

Karl: »Genau.«

Anna: »Ich werde die Notiz ›hat kein Abonnement‹ etwas nach oben verschieben, um genug Platz für den vierten Satz zu haben.« (Siehe Abb. 2–6.)

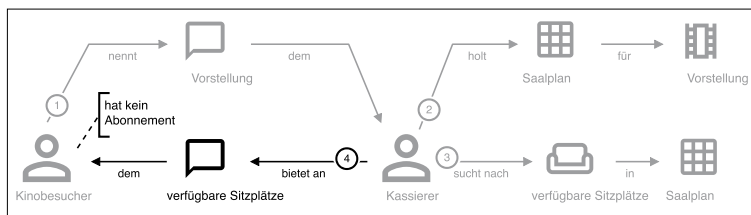


Abb. 2–6 Die vierte Aktivität

Innerhalb weniger Minuten füllt sich das Whiteboard mit einer Geschichte über einen Kinobesucher, der eine Karte an der Kasse kauft. Die Icons und Pfeile werden während des Workshops immer wieder neu angeordnet. Zum Schluss erzählt Anna die Geschichte noch einmal von vorne (siehe Abb. 2–7).

Karl: »Ja, das passt so. Nur die internationalen Filme habe ich noch vergessen.«

Anna: »Du meinst die Vorstellungen in einer Fremdsprache? Ich dachte, du verkaufst dafür spezielle Karten.«

Karl: »Nein, nein! Normalerweise zeigen wir die Filme auf Deutsch. Einige ausländische Film zeigen wir in Originalsprache. Wir verkaufen dafür keine speziellen Karten; du musst die Kinobesucher nur darauf hinweisen, in welcher Sprache der Film gezeigt wird.«

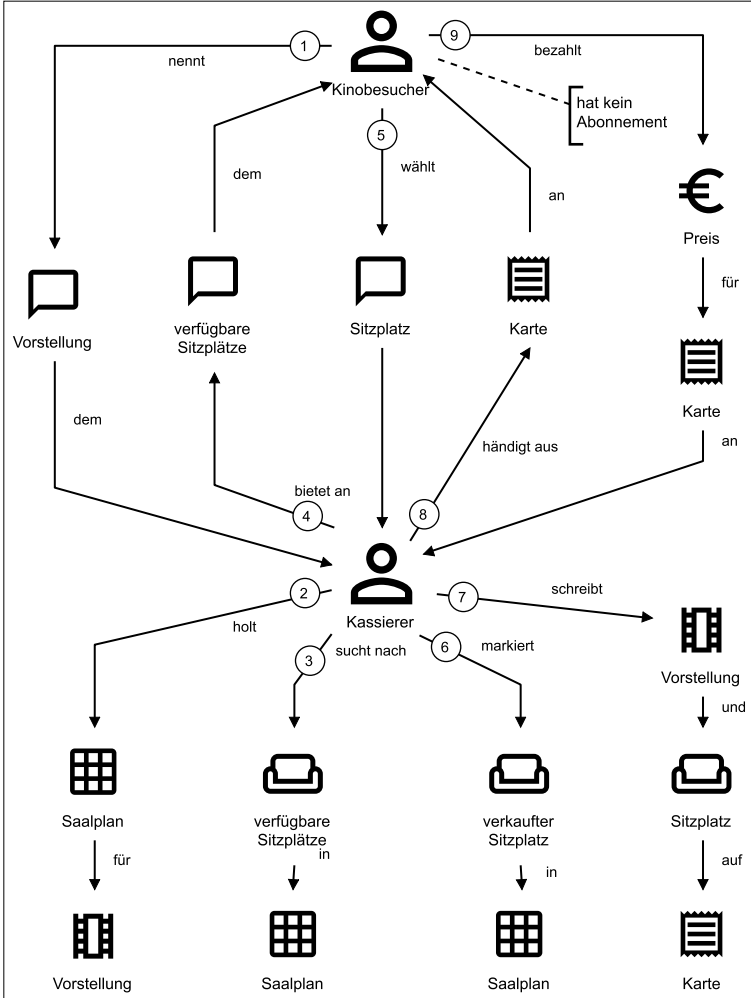


Abb. 2–7 Die ganze Geschichte

Anna: »Wann macht der Kassierer das?«

Karl: »Hier.«

Karl zeigt auf den Pfeil mit der Nummer 4 in Abbildung 2–7. Anna ergänzt den Satz »Der Kassierer bietet an verfügbare Sitzplätze dem Kinobesucher« mit einer Notiz »und erwähnt Sprache« (siehe Abb. 2–8).

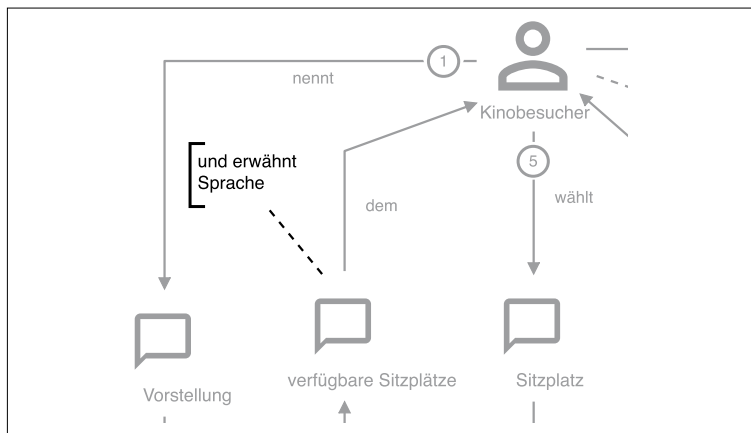


Abb. 2–8 Hinzufügen einer weiteren Notiz

Anna: »Es scheint, dass wir mit unserer kleinen ›Ins Kino gehen‹-Geschichte fertig sind. Natürlich haben wir bisher nur den bestmöglichen Fall betrachtet; ich nenne das auch den ›Happy Path‹. Später werde ich dich noch nach anderen Fällen fragen.«

Nach ein paar weiteren Domain Stories hat Anna einen guten Einblick in die Fachlichkeit des »Metropolis« bekommen. Sie versteht Begriffe wie »Saalplan«, »Vorstellung«, »Kasse«, »nach verfügbaren Plätzen suchen« und »Plätze markieren«. Sie hat ein erstes Verständnis der wichtigsten Abläufe und kann sich nun Gedanken machen, was die Einführung einer App für diese Abläufe bedeuten würde.

3 Wie funktioniert Domain Storytelling?

Früher war das Geschichtenerzählen eine rein mündliche Tätigkeit. Domain Storytelling ist eine mündliche *und* visuelle Aktivität, eine Form der *Modellierung*: Während die Fachexperten ihre Geschichte in gesprochener Sprache erzählen, zeichnet einer der Workshop-Teilnehmer – der Moderator – die Geschichte als Diagramm aus einfachen Icons, Pfeilen und Text auf. Auf diese Weise erhalten die Teilnehmenden eine visuelle Darstellung der Geschichte, die dabei hilft, Missverständnisse, Widersprüche und Handlungslücken aufzudecken. Alle Teilnehmenden sehen, wie sich das grafische Modell gleichzeitig mit der Geschichte entwickelt. Das macht es ihnen einfach, Feedback zu geben und sich einzubringen.

3.1 Szenariobasiertes Modellieren

Jede Domain Story ist eine Geschichte – erzählt von Anfang bis Ende, ohne Fallunterscheidungen (siehe Abb. 3–1). Andere Ansätze der Prozessmodellierung zielen darauf ab, alles zu beschreiben, was passieren könnte. Ein solche vollständige Erzählung aller möglichen Prozessvarianten ist nicht das Ziel von Domain Storytelling. Die Methode wurde bewusst für die Kommunikation zwischen Entwicklungsteams und Fachleuten entworfen. Andere Notationen wie UML oder BPMN zielen eher auf die Kommunikation von Entwickler zu Entwickler ab.

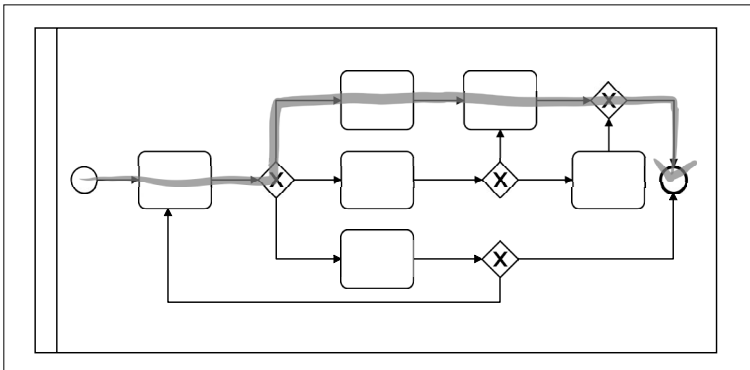


Abb. 3–1 Ein Szenario ist ein konkretes Beispiel für einen Geschäftsprozess.

Eine Domain Story ist ein konkretes, aussagekräftiges Beispiel für einen Geschäftsprozess. Aber wie kommen wir von den vielen möglichen Prozessvarianten hin zu wenigen, repräsentativen Domain Stories? Was, wenn es mehrere wichtige Alternativen gibt, die abgedeckt werden müssen? Werfen wir einen weiteren Blick auf das Kinobeispiel:

Das Kino Metropolis verkauft (a) Karten für einzelne Vorstellungen und (b) Jahresabonnements. Das bedeutet, dass die App-Entwicklerin Anna den Kinobetreiber Karl nicht einfach bitten kann, ihr eine Geschichte über einen Kinobesucher zu erzählen. Stattdessen fragt sie ihn, welcher Fall häufiger vorkommt, und er sagt ihr, dass der Verkauf von Karten für einzelne Vorstellungen 75 % seiner Einnahmen ausmacht. Deshalb bittet Anna Karl, ihr diesen Fall zu erläutern, und schreibt »Kartenverkauf« an die Tafel – den Namen für diese Domain Story.

Während sie die Geschichte aufzeichnet, werden weitere Annahmen getroffen, um ein Abdriften in andere Handlungsstränge zu vermeiden. Sie nehmen beispielsweise an, dass genügend Sitzplätze verfügbar sind, dass der Kinobesucher die angebotenen Sitzplätze annimmt und so weiter: »Angenommen, es sind noch Sitzplätze verfügbar, was tust du dann?« Anna schreibt all diese Annahmen als Notizen auf.

Wir empfehlen, zuerst mit der Modellierung des Standardfalls – des »80 %-Falls« – oder des »Happy Path« zu beginnen.

Diese Szenarien vermitteln den Zweck eines Prozesses und helfen zu verstehen, *warum* die Akteure etwas tun. Schränken Sie den Fall bewusst ein – keine Ausnahmen, keine Fehler, keine Sonderfälle. Später können Sie explizit danach fragen, was alles schiefgehen kann, und wichtige Abweichungen und Fehlerfälle als eigene Domain Stories modellieren.

Normalerweise reicht eine Handvoll Domain Stories aus, um einen komplexen Geschäftsprozess zu verstehen. Kleinere Varianten wie optionale Aktivitäten sind keine eigene Domain Story wert und können als textuelle Anmerkungen festgehalten werden.

Mehr über szenariobasiertes Modellieren finden Sie in [Hofer & Schwentner 2023] in Kapitel 3.

3.2 Die Bildsprache

Domain Stories werden mit einer einfachen Bildsprache visuell festgehalten. Akteure und Arbeitsgegenstände werden als beschriftete Icons abgebildet und Aktivitäten als beschriftete Pfeile. So entstehen »visuelle Sätze«. Für die Beschriftungen verwenden wir Begriffe aus der Domänensprache: ein Akteur »Kassierer«, ein Arbeitsgegenstand »Saalplan«, eine Aktivität »sucht nach« usw.

Bei Arbeitsgegenständen kann ein Icon Folgendes darstellen:

- den eigentlichen Arbeitsgegenstand, z.B. eine Kinokarte aus Papier,
- eine Repräsentation des Arbeitsgegenstands, z.B. eine digitale Kinokarte,
- das Medium, über das Informationen zum Arbeitsgegenstand ausgetauscht werden, z.B. ein Telefonat zur Reservierung einer Kinokarte.

Passen Sie die Icons an die Domäne an, die Sie modellieren möchten. Wenn Sie in der Logistikbranche arbeiten, wo Frachtcontainer ein wichtiger Arbeitsgegenstand sind, sollten Sie ein Container-Icon in Ihren Domain Stories verwenden. Das Gleiche gilt für Akteure – Schiffe, Autos, Smartphones usw. könnten Akteure in Ihrer Domäne sein. Wenn Sie sich Icons zusammenstellen, empfehlen wir Folgendes:

- Beschränken Sie sich auf etwa ein Dutzend Icons, da eine zu große Anzahl an Icons die Bildsprache verwässert.
- Die Icons sollen eine Bedeutung vermitteln und die Domain Story »greifbarer« machen.

Für die Beispiele in dieser Broschüre verwenden wir Googles frei verfügbare Material Icons (<https://fonts.google.com/icons/>).

Je nach Kontext können die Konzepte einer Domäne als Akteure *oder* Arbeitsgegenstände modelliert werden. In der eingangs geschilderten Geschichte vom Bustransfer am Flughafen (siehe Abb. 1–1) ist der Bus ein Arbeitsgegenstand. Der Busfahrer holt damit die Fluggäste vom Gate ab. Einen autonom fahrenden Bus hätten wir hingegen als Akteur dargestellt.

Menschliche Akteure werden in der Regel nach ihrer Rolle und nicht nach ihrem Namen benannt (»Kassierer« und nicht »Karl«). In einigen Situationen kann es sinnvoll sein, eine Ausnahme von dieser Regel zu machen.

Mit Pfeilen werden Akteure und Arbeitsgegenstände zu Sätzen kombiniert. Die Grundsyntax lautet: Subjekt – Prädikat – Objekt. Ein Satz in der Bildsprache soll sich fast wie ein natürlichsprachlicher Satz lesen. Drei häufige Satzformen sind in Abbildung 3–2 dargestellt.

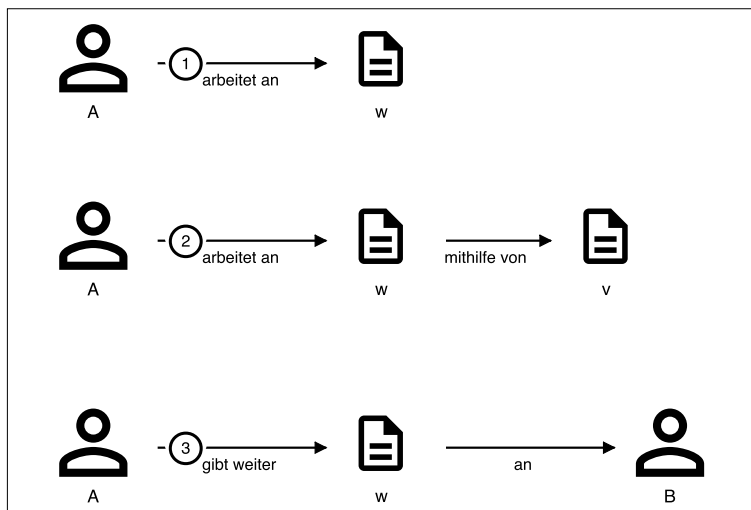


Abb. 3–2 Häufige Satzformen einer Domain Story

1. Ein Akteur A arbeitet mit einem Arbeitsgegenstand w (erstellt ihn, bearbeitet ihn, schaut etwas darin nach, ...).
2. Ein Akteur A verwendet einen Arbeitsgegenstand w, um einen anderen Arbeitsgegenstand v zu bearbeiten.
3. Ein Akteur A übergibt einen Arbeitsgegenstand w an einen anderen Akteur B, oder A tauscht mit B Informationen über w aus. Oft ist eine Präposition (z.B. *an*, *zu*, *mit*, *in*) eine passende Bezeichnung für den zweiten Pfeil.

Um eine Geschichte zu erzählen, braucht man mehr als einen Satz. Die Nummerierung der Aktivitäten drückt aus, in welcher Reihenfolge die Aktivitäten stattfinden.

Domain Stories entwickeln sich um die Akteure herum. Sie sind die Subjekte in unseren Sätzen. In einem Workshop sollten die Akteure durch teilnehmende Personen repräsentiert sein. Damit sie sich leicht mit den Akteuren der Domain Story identifizieren, sollten die Akteure im visuellen Modell gut zu erkennen sein. Wir empfehlen daher, sie etwas größer als die Arbeitsgegenstände zu zeichnen.

Mehr Informationen zu Bildsprache erhalten Sie in [Hofer & Schwentner 2023] in Kapitel 2.

3.3 Scope

Domain Storytelling ist kein »One-size-fits-all«-Ansatz. Je nach Zweck passt der Modellierer den sogenannten **Scope** einer Domain Story an:

- Die **Granularität** bestimmt die Flughöhe der Modellierung: Interessiert uns der Überblick oder die Details? Domain Stories können **grobgranular**, **feingranular** und alles dazwischen sein.
- Der **Zeitpunkt**, zu dem eine Domain Story handelt, kann im **Ist** liegen und den aktuellen Prozess beschreiben. Das **Soll** hingegen ist ein Blick in die Zukunft. Verschiedene zukünftige Zeitpunkte werden in unterschiedlichen Domain Stories modelliert.
- Bei der Modellierung von Domain Stories berücksichtigt man entweder (vorhandene oder noch zu entwickelnde) Software oder man lässt sie weg. Dieser Scope-Faktor wird **Domain Purity** genannt. Wir nennen Domain Stories ohne Softwaresysteme **pur** und solche mit Softwaresystemen **digitalisiert**.

Beispielsweise hat die Domain Story »Bustransfer« aus Kapitel 1 den Scope *grobgranular, ist, digitalisiert* und der »Kartenverkauf an der Kinokasse« aus Kapitel 2 hat den Scope *feingranular, ist, pur*.

Mehr zum Scope beschreiben wir in [Hofer & Schwentner 2023] in Kapitel 4.

3.4 Der Workshop

Zu den Teilnehmenden des Workshops gehören Fachleute (oft aus mehreren Abteilungen), IT-Expertinnen und ein Moderator. Es ist wichtig und leider auch schwierig, dass die »richtigen« Personen teilnehmen. Achten Sie auch darauf, dass Sie echte Experten einladen – und nicht »Stellvertreter«, die das Tagesgeschäft nur vom Hörensagen kennen. Selten ist eine einzelne Person in der Lage, eine ganze Domain Story umfassend zu erzählen, denn in der Regel sind nicht triviale Geschäftsprozesse kooperativ und entsprechend gibt es verschiedene Sichtweisen auf einen Prozess. Das erfordert, dass mehrere Workshop-Beteiligte zur Geschichte beitragen und sich Satz für Satz auf eine Geschichte einigen. Jede Domain Story verkörpert das gemeinsame Verständnis ihrer Erzähler.

Damit ein Workshop erfolgreich wird, muss die Domain Story für alle Beteiligten jederzeit sichtbar sein. Das bestimmt die Wahl des Tools, mit dem Sie das Bild zeichnen (siehe Kapitel 4 »Modellierungswerkzeuge«). Ein Whiteboard mit handgemalten Icons ist für kleine Gruppen geeignet. Wenn Sie ein Softwaretool nutzen, können Sie über einen Projektor oder ein großes Smartboard den Bildschirm des Moderators für alle im Raum sichtbar machen (siehe Abb. 3–3).

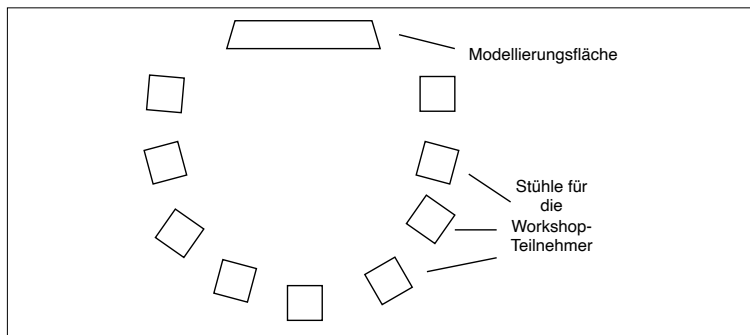


Abb. 3–3 Anordnung im Workshop-Raum: Die Domain Story muss jederzeit für alle sichtbar sein.

Während die Teilnehmer ihre Geschichte erzählen, zeichnet der Moderator sie grafisch auf – Satz für Satz – und bestimmt so das Tempo des Geschichtenerzählens. Während ein neuer Satz festgehalten wird, sollte der Moderator diesen laut mitsprechen, um sicherzustellen, dass die im Bild verwendeten Begriffe tatsächlich aus der Domänensprache stammen. Die Beteiligten sehen und hören, was aufgezeichnet wird, und geben sofort Feedback.

Sobald die Diskussion zu einer Domain Story abgeschlossen ist, können die Beteiligten besprechen, welche alternativen Geschichten relevant sind. Lassen Sie die Teilnehmer entscheiden, was kleine Varianten sind, die im Bild als Kommentar aufgenommen werden, und was eine eigene Domain Story verdient. Als Moderator oder Moderatorin können Sie diese Diskussion mit Fragen anregen wie:

- Sprechen wir von der gleichen Aufgabe, die manchmal unterschiedlich gelöst wird? Oder reden wir über eine andere Aufgabe?
- Was wäre anders, wenn ...?
- Habe ich recht, dass der einzige Unterschied darin besteht, dass ...?

Falls erforderlich, modellieren Sie eine neue Domain Story. Vielleicht planen Sie einen Folgeworkshop, um einen detaillierteren Blick auf Teile der Geschichte zu werfen oder um sich mit wichtigen Variationen auseinanderzusetzen. Wenn alles gut gelaufen ist, haben Sie ein gemeinsames Verständnis über einen relevanten Teil der Domäne aufgebaut. Allerdings wird es selten zu 100 % gelingen, die Wahrnehmungen mehrerer Menschen aufeinander abzustimmen. Achten Sie darauf, nicht zu Verallgemeinerungen oder abstrakten Begriffen zu springen, um widersprüchliche Ansichten auszublenden. Verwenden Sie stattdessen Anmerkungen, um nicht gelöste Unterschiede sichtbar zu machen.

Nach Abschluss des Workshops wird das Ergebnis festgehalten. Machen Sie z.B. ein Foto vom Whiteboard. Wenn Sie das Bild mit einem Werkzeug gezeichnet haben, lohnt es sich, das Modell »aufzuräumen«, um die Anordnung der Icons und Pfeile übersichtlicher zu gestalten. Verteilen Sie dann das Bild. In erster Linie ist das Bild für die Personen gedacht, die die Geschichte erzählen, während sie diese erzählen. Später

kann das Bild als Gedächtnisstütze für die Teilnehmenden des Workshops dienen. Weitere Artefakte und Dokumente des Softwareentwicklungsprozesses werden um die Domain Stories herum entstehen, beispielsweise verschriftlichte Anforderungen oder Mock-ups. Sie können das Gespräch über die Domäne auch mit anderen Modellierungstechniken fortsetzen.

In [Hofer & Schwentner 2023] beschreiben wir das Workshop-Format detailliert in Kapitel 6 und das Verhältnis zu anderen Modellierungstechniken in Kapitel 7.

4 Modellierungswerkzeuge

Domain Stories können mit verschiedenen Werkzeugen modelliert werden. Wir stellen einige von ihnen vor.

4.1 Analoge Werkzeuge

Wir haben Erfahrung gesammelt bei der Modellierung mit ...

- Stift und Papier,
- Klebezetteln auf Flipchart oder Papierrolle,
- Klebezetteln auf dem Whiteboard,
- dem »Whiteboard-Kit«
- und Whiteboards ohne weitere Hilfsmittel.

Analoge Werkzeuge geben allen Teilnehmer »Schreibzugriff« auf das Modell und ermöglichen damit gemeinschaftliche Modellierung. Das ist bei digitalen Tools mit Maus und Tastatur nicht immer der Fall.

Der offensichtlichste Nachteil der papierbasierten Modellierung ist, dass die Modelle schwer zu korrigieren und umzugestalten sind. Der Moderator braucht eine gewisse Erfahrung, um das beim Layout des Modells im Voraus zu berücksichtigen. So sollten Akteure, die im Mittelpunkt der Geschichte stehen, auch in der Mitte platziert werden. Damit ist genügend Platz für viele ein- und ausgehende Pfeile vorhanden. Die Verwendung von Klebezetteln für Akteure und Arbeitsgegenstände erleichtert das Leben erheblich. Wenn Sie Klebezettel auf einem Whiteboard verwenden und die Pfeile direkt auf dem Whiteboard zeichnen, haben Sie ein einfach zu bedienendes Werkzeug. Als Konvention hat sich bei uns herausgebildet, Akteure auf längliche, senkrecht geklebte Stickies und Arbeitsgegenstände auf quadratische Stickies in derselben Farbe zu zeichnen (siehe Abb. 4-1).

Mit Klebezetteln müssen Sie die Icons immer noch von Hand zeichnen. Viele Moderatoren neigen deshalb dazu, wenige Icons zu zeichnen und Akteure oder Arbeitsgegenstände als Text zu beschreiben. Wenn die Textform für den gegebenen Zweck funktioniert, ist das in Ordnung.

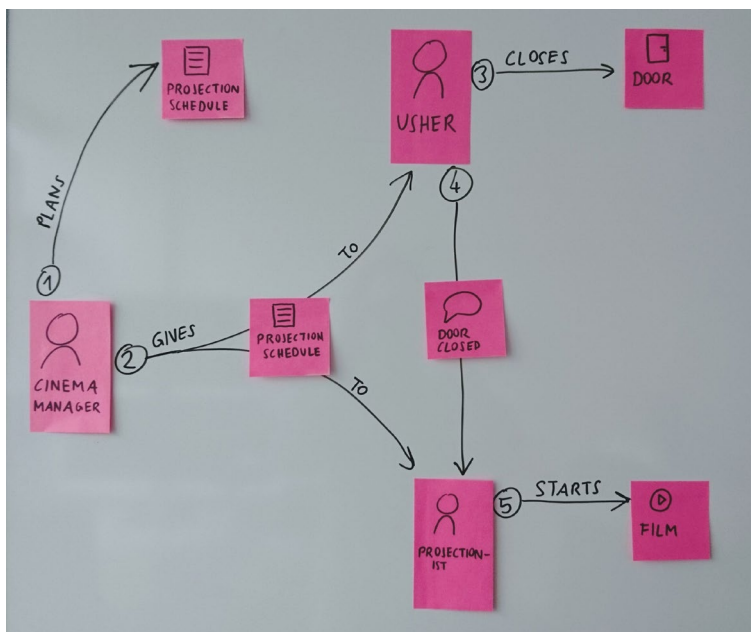


Abb. 4-1 Domain Story mit Klebezetteln

Aber wir denken, dass die Ausdrucksmöglichkeiten und die Verständlichkeit von Bildern so nicht voll genutzt werden. Die Diskussion im Team darüber, wie ein Arbeitsgegenstand dargestellt werden soll, deckt oft verborgene Annahmen auf. Icons zu zeichnen, erleichtert solche Diskussionen und ist daher sinnvoll.

Als Alternative zum Zeichnen der Icons mit der Hand können Sie sich ein Whiteboard-Kit anfertigen. Dazu drucken Sie Ihre Icons auf dickeres Papier, laminieren sie und kleben kleine flache Magnete auf die Rückseite. Die so entstehenden Karten haften auf dem Whiteboard. Wenn Sie die Karten mit einem Whiteboard-Marker beschriften, sind sie wiederverwendbar. Sie finden eine Vorlage für ein Whiteboard-Kit auf www.domainstorytelling.org und ein Beispiel für seine Verwendung in Abbildung 4-2.

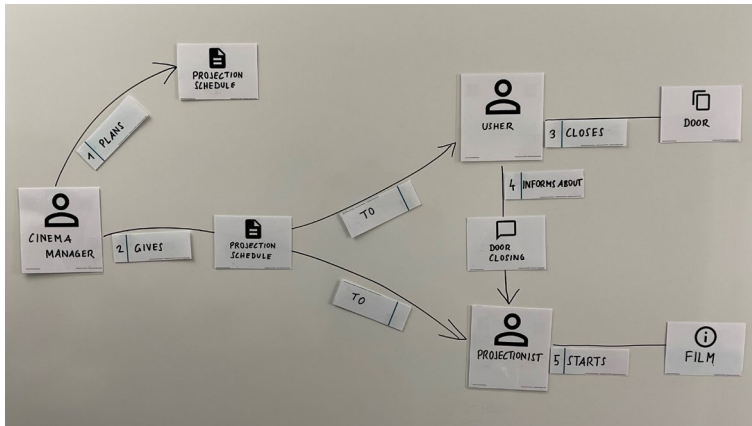


Abb. 4-2 Domain Story mit Whiteboard-Kit

4.2 Digitale Werkzeuge

Softwarewerkzeuge fallen in die folgenden Kategorien:

- Digitales händisches Zeichnen (z. B. mit Tablet und digitalem Stift)
- Universell einsetzbare Zeichenwerkzeuge (wie Gliffy, yEd und diagrams.net)
- Virtuelle, kollaborative Whiteboards (Miro, Mural, Conceptboard, ...)
- Spezialisierte Modellierungswerkzeuge, wie unser *Egon.io – The Domain Story Modeler*

Die meisten Softwarewerkzeuge sind für einen einzelnen Benutzer konzipiert, der ein Eingabegerät (Maus, Tastatur, Stift usw.) verwendet. Da gemeinsames Modellieren mit solchen Werkzeugen nicht möglich ist, muss eine Person die Rolle des *Modellierers* übernehmen und die Zeichenarbeit erledigen. Ausnahmen sind virtuelle Whiteboards und digitale Tafeln. Wie bei analogen Werkzeugen sollte das Modell für alle Teilnehmenden sichtbar sein.

Unserer Erfahrung nach gibt es kein Zeichenwerkzeug und kein virtuelles Whiteboard, das sich bei der Modellierung von Domain Stories besonders auszeichnet. Unsere Empfehlung beschränkt sich daher darauf, ein zu der jeweiligen Domäne passendes Icon-Set zu verwenden.

Unserer Meinung nach muss ein Werkzeug, das Domain Storytelling *wirklich* unterstützt, Folgendes leisten:

- Die syntaktischen Regeln von Domain Stories durchsetzen. Wir wollen ein Werkzeug zum *Modellieren*, nicht zum *Zeichnen*.
- Sätze automatisch nummerieren und dabei die Möglichkeit bieten, die Reihenfolge der Sätze zu ändern.
- Beim *Geschichtenerzählen* helfen, indem die Sätze der Reihe nach animiert werden.

Wir haben in unserer Firma WPS ein solches Modellierungswerkzeug entwickelt: *Egon.io – The Domain Story Modeler* (siehe Abb. 4–3). Egon.io erfüllt die oben genannten Anforderungen. Falls Sie sich gefragt haben, womit wir die Domain Stories für diese Broschüre modelliert haben, kennen Sie jetzt die Antwort.

Egon.io ist Open Source und kann einfach online verwendet werden: <https://egon.io>. In Kapitel 5 in [Hofer & Schwentner 2023] geben wir Tipps für die Auswahl von Werkzeugen.

5 Einsatzzwecke

Bisher haben wir den Nutzen von Domain Storytelling anhand einiger Beispiele gezeigt. Nun gehen wir näher auf die unterschiedlichen Einsatzzwecke ein.

5.1 Fachsprache lernen

Um nützliche Unternehmenssoftware zu entwickeln, muss man zuerst die Domäne des Unternehmens verstehen. Mit Domain Storytelling lässt sich Wissen über die Domäne aufbauen. Die Sprache der Fachleute zu lernen, ist dabei die wichtigste Aufgabe. Sie ist der Schlüssel zu effektiven Gesprächen über Geschäftsprozesse und Anforderungen an die Software.

Domain Storytelling kann in den folgenden Situationen hilfreich sein:

- Man ist neu in einer Domäne (z.B. als externer Entwickler) und muss sich Wissen über die Domäne aneignen.
- Man möchte Fachexperten aus verschiedenen Abteilungen zusammenbringen, um Abteilungsgrenzen zu überwinden und Annahmen zu hinterfragen.
- Die Software, an der man arbeitet, verwendet keine echten Begriffe aus der Domäne und das soll sich ändern.
- Man arbeitet in einer Organisation, in der es keine echte Fachsprache gibt, und man möchte, dass eine solche entsteht.

Gemeinsame Workshops bilden eine sehr gute Grundlage, um Fragen zu stellen, Erläuterungen von Fachbegriffen nachzufragen und bei Hospitationen mehr über die tatsächlichen Abläufe zu erfahren. Oft bauen wir parallel zu den Domain Stories ein Glossar der Fachbegriffe auf.

5.2 Mit Anforderungen arbeiten

Der Sprung vom Domänenwissen hin zum Code ist in der Regel eine zu große Herausforderung, und diese Kluft muss mit Anforderungen überbrückt werden. Funktionale Anforderungen können aus Domain Stories abgeleitet werden.

In unserem Kinobeispiel stellen Karl und Anna in einem Bild dar, wie der Verkaufsprozess an der Abendkasse in Zukunft ablaufen soll. Sie stellen dabei fest, dass die Kassierer einen digitalen Saalplan brauchen, der auch die per App verkauften Sitzplätze anzeigt.

Aus den Domain Stories lassen sich leicht funktionale Anforderungen ableiten. Ein Beispiel: Möchte eine Besucherin unseres Kinos vor Ort Karten kaufen, sucht der Kassierer nach dem passenden Saalplan für die Vorstellung und bietet die freien Plätze an. Dazu lässt sich folgende User Story für ein System formulieren: »Als Kassierer möchte ich nach verfügbaren Plätzen suchen, damit ich sie dem Besucher anbieten kann.« (Siehe Abb. 5–1.)

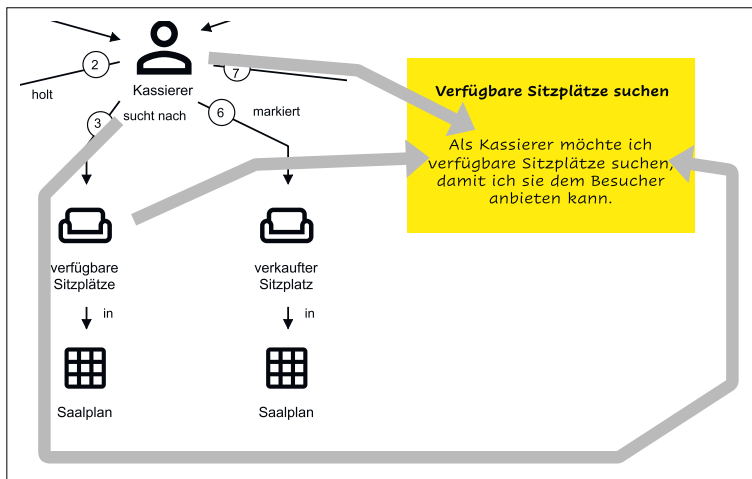


Abb. 5–1 Von der Domain Story zur User Story

Domain Storytelling ist ein hilfreiches Werkzeug für alle, die mit Anforderungen arbeiten, sei es als Product Owner, Produktmanagerin, Business-Analyst, Requirements Engineer oder als cross-funktionales Team, das selbst Anforderungsanalyse betreibt. Sie alle können damit ...

- herausfinden, was von einem Stück Software erwartet wird, und
- den Kontext von Anforderungen verstehen.

5.3 Organisatorische Veränderung unterstützen

Das Ziel eines neuen Softwaresystems ist es in der Regel, die Arbeit einfacher, schneller und effizienter (oder kurz: besser) zu machen. Dieses Ziel wird nicht dadurch erreicht, indem schlechte manuelle Prozesse digitalisiert werden. Ebenso wenig wird aus einem Haufen Anforderungen auf magische Weise ein reibungslos funktionierender Geschäftsprozess. Um gute Unternehmenssoftware zu entwickeln, muss man mehr tun, als nur den aktuellen Zustand in Software zu implementieren. Man muss auch die zukünftige Arbeitsweise entwerfen. Domain Stories helfen dabei und veranschaulichen, wie die neue Software die Arbeitsweise der Menschen verändern wird:

- Entwurf der zukünftigen Arbeitsweise einer Organisation
- Optimierung von Geschäftsprozessen
- Diskussion über und Erleichterungen von Veränderungen in Geschäftsprozessen
- Inbetriebnahme einer neuen Software
- Einführung einer neuen Version einer bestehenden Software

Domain Storytelling schafft die Möglichkeit, die von Veränderungen Betroffenen schon früh in das Projekt einzubeziehen und mit ihnen die neue Arbeitssituation durchzusprechen.

5.4 »Make or Buy«-Entscheidungen und Auswahl von Standardsoftware

Nicht jede Software ist eine Individualanfertigung. Für viele Domänen gibt es Standardsoftware. Domain Stories können bei der Entscheidung helfen, ob ein neues Softwaresystem entwickelt oder Standardsoftware beschafft werden soll. Wenn Standardsoftware infrage kommt, muss zwischen Produkten mehrerer Anbieter entschieden werden. Auch hier können Domain Stories dabei helfen, eine Auswahl zu treffen.

5.5 Domain Storytelling in Ihrer Werkzeugkiste

Wer nur einen Hammer als Werkzeug hat, sieht die Welt voller Nägel. In Entwicklungsprojekten kommen wir selten mit nur einem Modellierungsansatz aus. Wir kombinieren Domain Storytelling daher häufig mit anderen Modellierungsmethoden wie Event Storming (www.eventstorming.com).

Im Kontext der Anforderungsermittlung können aus Domain Stories grobe *User Stories* abgeleitet werden. Diese lassen sich in einer *User Story Map* (von Jeff Patton) organisieren – und die Domain Story liefert das »Rückgrat« der User Story Map. Alternativ zu User Stories werden Anforderungen häufig als *Use Cases* (von Ivar Jacobson, Varianten von Alistair Cockburn und Martin Fowler) geschrieben. Mit Domain Storytelling können die Haupt- und Alternativszenarien eines Use Case erarbeitet werden.

Domain Storytelling mit anderen Modellierungsmethoden kombinieren ist das Thema von Kapitel 7 in [Hofer & Schwentner 2023]. Die Kapitel 9–15 des Buches stellen die Einsatzzwecke im Detail vor.

6 Domain-Driven Design

Auf einige Einsatzzwecke möchten wir näher eingehen. Sie spielen eine Rolle im *Domain-Driven Design* (DDD), einem erfolgreichen Ansatz in der Softwareentwicklung. Als Einstieg empfehlen wir das Buch *Domain-Driven Design kompakt* von Vaughn Vernon [Vernon 2017].

Für DDD ist es zentral, die Fachlichkeit (Domäne) zu verstehen und in der Software abzubilden. Gemeinschaftliches Modellieren (engl. *collaborative modeling*) hilft allen Beteiligten, ein geteiltes Verständnis der Domäne zu entwickeln. Domain Storytelling hat sich in den letzten Jahren als Technik des Collaborative Modeling etabliert.

6.1 Grenzen finden

Viele Domänen sind zu groß, um als Ganzes verstanden und modelliert zu werden. In solchen Fällen muss eine Domäne in überschaubare Einheiten aufgeteilt werden – im DDD wird dies etwas vereinfacht als *strategisches Design* bezeichnet. Das kann in folgenden Situationen hilfreich sein:

- Sie ringen mit einem schlecht strukturierten Monolithen und möchten diesen modularisieren oder zerschneiden.
- Sie möchten Microservices oder Self-Contained Systems entwickeln.
- Sie möchten Domain-Driven Design (DDD) anwenden und dabei Bounded Contexts schneiden.
- Ein Entwicklungsteam ist zu groß, um effizient zu arbeiten, und Sie möchten es in mehrere Teams aufteilen.
- Sie haben bereits mehr als ein Entwicklungsteam und möchten die Arbeit der Teams besser organisieren.

Domain Storytelling unterstützt einen wichtigen Schritt in diesem Prozess: das Finden der Grenzen zwischen Subdomänen.

Dazu eignen sich überblicksartige Domain Stories, die die wichtigsten Geschäftsprozesse und ihre Varianten grobgranular darstellen. In den Domain Stories werden Aktivitäten grafisch gruppiert, um die verschiede-

nen Subdomänen sichtbar zu machen. Wir greifen hier wieder auf unser Kino-Beispiel zurück (siehe Abb. 6–1).

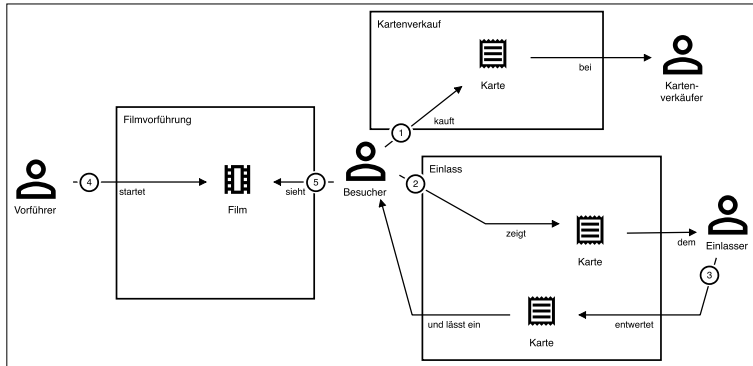


Abb. 6–1 Grobgranuläre Domain Story mit Subdomänen

Das Wissen zum Thema »Grenzen finden« wird in Kapitel 10 in [Hofer & Schwentner 2023] vertieft.

6.2 Modeling in Code

Für uns beschränkt sich die Modellierung nicht auf das Zeichnen von Diagrammen. Um Software zu entwickeln, muss man irgendwann vom Modellieren mit Klebezetteln und Diagrammen zum Modellieren in Programmiersprachen übergehen. Dabei sollten Begriffe aus der Domäne direkt im Code verwendet werden.

Sehen wir uns eine Subdomäne der Kino-Domäne genauer an. Betrachten wir noch einmal die detaillierte Story der Subdomäne »Kartenverkauf« (siehe Abb. 6–2).

Die Interaktion zwischen Besucher und Kassierer soll auch weiter außerhalb des Softwaresystems »analog« erfolgen. Die Schritte 3 und 6 sind jedoch ideale Kandidaten für eine Automatisierung. Das resultierende Domänenmodell sieht dann wie in Abbildung 6–3 aus.

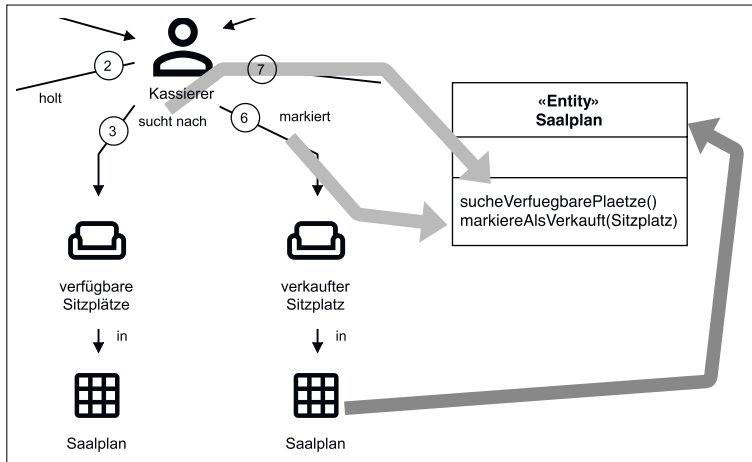


Abb. 6–3 Von der Domain Story zum Domänenmodell

Aus der Domain Story entsteht hier ein Klassendiagramm. Wir können auch direkt Code (hier in Java) schreiben. Während wir das Domänenmodell (idealerweise testgetrieben) implementieren und darüber nachdenken, wird uns klar, dass die Domain Story noch Details auslässt. Die weitere Diskussion mit den Fachexperten führt zu Abbildung 6–4.

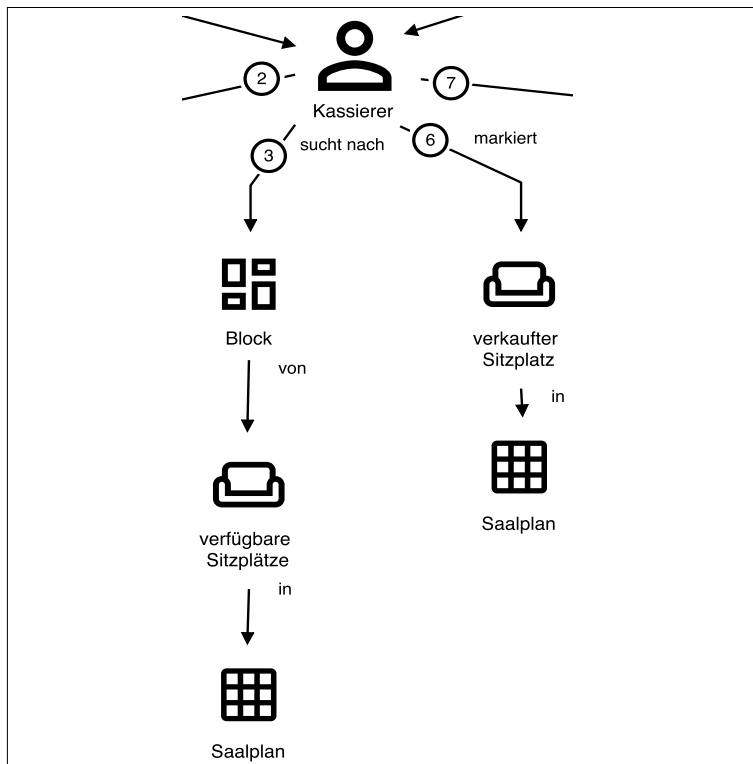


Abb. 6-4 Weiterentwickelte Domain Story »Kartenverkauf«

Als Code ergibt sich nun:

```
import org.jmolecules.ddd.annotation.Entity;
@Entity
public class Saalplan {
    public Block sucheVerfuegbarePlaetze
        (Anzahl anzahl) { /*...*/ }
    public void markiereAlsVerkauft(Block block) { /*...*/ }
}
```

Wir fügen also noch die Klassen **Block** und **Sitzplatz** in unser Modell ein. Damit verfeinert sich auch der Saalplan (siehe Abb. 6–5).

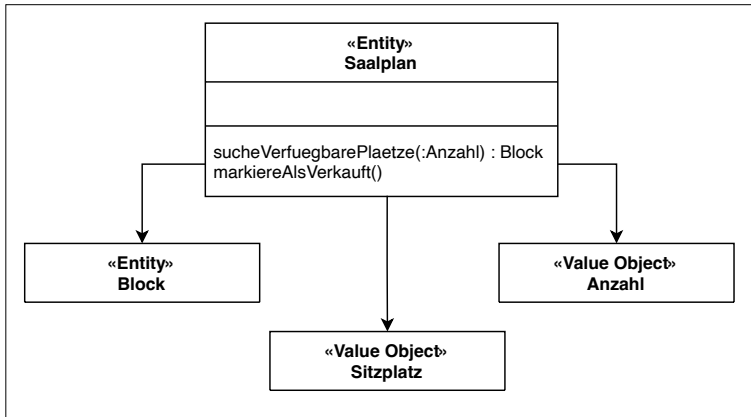


Abb. 6–5 Erweitertes Domänenmodell

Dieses Domänenmodell implementieren wir im Domain Layer unserer Anwendung.

»Modeling in Code« ist das Thema von Kapitel 12 in [Hofer & Schwentner 2023]. Dort zeigen wir u.a. auch, wie Domain Storytelling mit funktionaler Programmierung zusammenspielt.

7 Die nächsten Schritte

Domain Storytelling ist eine einfache Technik. Um sie aber in größeren Gruppen sicher anwenden zu können, benötigen Sie als Moderatorin oder Moderator etwas Erfahrung. Diese können Sie sich in unseren Trainings, durch die Lektüre unseres Buches und durch selbstständiges Üben aneignen.

7.1 Domain Storytelling üben

Wir empfehlen, zunächst klein anzufangen. Probieren Sie Domain Storytelling für sich aus, anhand von Übungsbeispielen mit Stift und Papier oder online mit Egon.io.

Wenn Sie die Notation und das szenariobasierte Modellieren einigermaßen verinnerlicht haben, schnappen Sie sich jemand, der die Rolle des Fachexperten oder der Fachexpertin übernimmt. Lassen Sie sich einen Arbeitsablauf aus dem Alltag dieser Person erzählen. Modellieren Sie die Geschichte visuell und probieren Sie aus, wie Domain Storytelling im Dialog funktioniert. Danach sind Sie bereit für Workshops mit mehreren Personen.

7.2 Literatur

[Domain Storytelling Website] <https://domainstorytelling.org>

Die offizielle Webseite zum Thema.

[Hofer & Schwentner 2023] Hofer, Stefan; Schwentner, Henning: *Domain Storytelling: Gemeinschaftlich, visuell und agil zu fachlich wertvoller Software*. dpunkt-verlag, Heidelberg, 2023.

Unser Buch beschreibt die Methodik umfassend. Wir geben darin komplexere Beispiele und erläutern die Einsatzzwecke ausführlicher.

[Lilienthal & Schwentner 2023] Lilienthal, Carola; Schwentner, Henning: *Domain-Driven Transformation: Monolithen und Microservices zukunftsfähig machen*. dpunkt.verlag, Heidelberg, 2023.

Carola Lilienthal und Henning Schwentner haben ihre Erfahrungen mit DDD, Domain Storytelling und langlebigen Softwarearchitekturen in diesem Buch verarbeitet, das Ihnen bei der Modularisierung monolithischer Softwaresysteme helfen wird.

[Vernon 2017] Vernon, Vaughn: *Domain-Driven Design kompakt*. dpunkt.verlag, Heidelberg, 2017.

Domain Storytelling wird oft im Zusammenhang mit Domain-Driven Design (DDD) verwendet. Als Einführung in DDD empfehlen wir Ihnen Vaughn Vernons Buch.

7.3 Schulung und Beratung

Wir bieten offene und Inhouse-Schulungen an (<https://domainstorytelling.org/training>). Gerne unterstützen wir Sie als Moderatoren bei Ihren Workshops. Neben Domain Storytelling bringen wir noch einige andere Methoden in unserem Werkzeugkoffer mit, beispielsweise Event Storming, User Story Mapping und Example Mapping. Wir sind nicht dogmatisch, sondern wählen als Berater immer die Methode, die Ihr Ziel am besten unterstützt.

Kontaktieren Sie uns gerne (stefan@domainstorytelling.org und henning@domainstorytelling.org) oder sprechen Sie uns auf Fachkonferenzen und Meetups an. Wir sind in den DDD-, Requirements- und Software-Architektur-Communitys aktiv.

Anforderungsermittlung kann man trainieren!

Gute Business-Software muss auf die Fachlichkeit und die Aufgaben der Anwender zugeschnitten sein. Deswegen funktioniert Softwareentwicklung nicht im „stillen Kämmerlein“, sondern nur in enger Zusammenarbeit mit den Fachexperten. Permanente Rückkopplung ist die Basis guter Anforderungen.

Anforderungsermittlung ist aus unserer Sicht ein ständiger Lern- und Kommunikationsprozess. Hierzu braucht es zeitnahe Rückkopplung – durch direkten Dialog, durch gemeinsames Modellieren und mit Hilfe von aussagekräftigen Dokumenten. Dazu setzen wir verschiedene Methoden und Techniken ein, die wir in unserem Werkzeugkoffer zusammengefasst haben.

Workshop-Formate

Domain Storytelling
Event Storming
User Story Mapping
Example Mapping
Design Studio

Leichtgewichtige Anforderungsdokumentation

Mock-Ups
Prototyping
Domain Stories
User Stories
Use Cases

Zu BPMN sowie Anforderungsermittlung bieten wir individuell zugeschnittene Trainings an. Event Storming und Domain Storytelling vermitteln wir außerdem in unserer standardisierten Schulung DDD Konkret (iSAQB® Advanced Modul).

Mehr Infos unter **wps.de/anforderungsermittlung**

Eine gemeinsame Sprache entwickeln: Domain-Driven Design konkret

iSAQB® Modul DDD

Profitieren Sie von echtem Praxiswissen!

Damit Ihre Software die Anwender*innen optimal unterstützt, brauchen Sie ein gutes und strukturiertes Verständnis der Fachdomäne. Das DDD-Modul zeigt Ihnen einen Weg, auf dem Sie die Fachdomäne systematisch erfassen und passgenau in Ihrer Software umsetzen können.



**Im Preis
enthalten**



www.wps.de/DDD



2024, 316 Seiten
komplett in Farbe, Broschur
€ 36,90 (D)
ISBN 978-3-86490-729-6

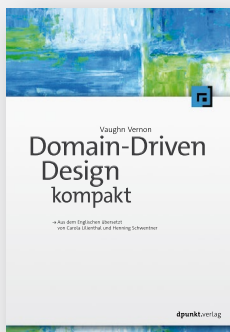
Carola Lilienthal

Langlebige Software-Architekturen

Technische Schulden analysieren, begrenzen und abbauen

4., überarbeitete und erweiterte Auflage

Das Standardwerk zur Softwarearchitektur: Die Autorin beschreibt, wie langlebige Softwarearchitekturen entworfen, umgesetzt und erhalten werden können. Mit zahlreichen gut nachvollziehbaren Beispielen aus realen Softwaresystemen erläutert sie gute und schlechte Lösungen für typische Probleme. Der vierten Auflage wurde ein eigenes Kapitel zum Modularity Maturity Index (MMI) hinzugefügt.



2017, 158 Seiten
komplett in Farbe, Broschur
€ 29,90 (D)
ISBN 978-3-86490-439-4

Vaughn Vernon

Domain-Driven Design kompakt

Aus dem Englischen übersetzt von
Carola Lilienthal und Henning Schwentner

Dieses Buch bietet einen kompakten Einstieg in die wesentlichen DDD-Konzepte, wie Ubiquitous Language, Bounded Contexts, Aggregates, Entities und Subdomänen. Nach der Lektüre sind Sie in der Lage, in Projekten eine gemeinsame Sprache für Fachanwender und Entwickler auch über Teamgrenzen hinweg zu finden.



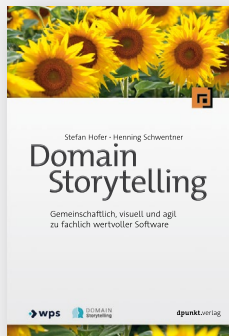
dpunkt.verlag
www.dpunkt.de

Stefan Hofer · Henning Schwentner

Domain Storytelling

Gemeinschaftlich, visuell und agil zu fachlich wertvoller Software

Fachleute aus der Domäne und Entwicklungsteams machen Fachwissen greifbar, indem sie Geschäftsprozesse als fachliche Geschichten erzählen und visualisieren. Domain Storytelling ermöglicht es allen Beteiligten, die relevanten Rollen, Aufgaben und Arbeitsgegenstände zu verstehen. Die Autoren erläutern die szenariobasierte Modellierung, das Workshop-Format und die Kombination mit anderen Modellierungsmethoden.



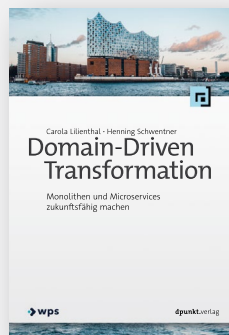
2023, 254 Seiten
komplett in Farbe, Broschur
€ 34,90 (D)
ISBN 978-3-86490-958-0

Carola Lilienthal · Henning Schwentner

Domain-Driven Transformation

Monolithen und Microservices zukunftsfähig machen

Dieses Buch veranschaulicht eine Methode, um sowohl auf strategischer als auch auf technischer und teamorganisatorischer Ebene architektonisch erodierte Softwaresysteme zu transformieren. Dabei spannt das Autorenteam den Bogen von der Analyse der fachlichen Prozesse und der Zerlegung in Bounded Contexts bis hin zu Domain-Driven Refactorings und deren Umsetzung in agilen Teams.



2023, 312 Seiten
komplett in Farbe, Broschur
€ 34,90 (D)
ISBN 978-3-86490-884-2



dpunkt.verlag
www.dpunkt.de



Stefan Hofer • Henning Schwentner

Domain Storytelling

Um gute Business-Software zu entwickeln, müssen Entwicklungsteams das »Business« verstehen. Dazu brauchen sie eine gemeinsame Sprache mit den Fachleuten, in der sie über die Anforderungen an die Software sprechen. Denn letztlich spiegelt eine Software nur das wider, was die Entwicklerinnen und Entwickler von den Anforderungen und der Fachlichkeit verstanden haben, und nicht, was jemand in einem Anforderungsdokument aufgeschrieben hat.

Diese Broschüre zeigt, wie Sie mit einfachen Mitteln, dem Domain Storytelling, fachlich stimmige Anwendungssoftware entwickeln können. Domain Storytelling hilft, das Fachwissen aus den Köpfen der Anwenderinnen und Anwender in die Köpfe von Entwicklungsteams, Product Ownern, Produktmanagerinnen und Business-Analysten zu transportieren.

Domain Storytelling bringt die Beteiligten in Workshops zusammen, um sich über Aufgaben und Prozesse abzustimmen. Das Ergebnis wird visuell mit einer einfachen Bildsprache modelliert.

Die Broschüre zeigt anhand verständlicher Beispiele, wie Domain Stories entstehen und fasst die langjährige praktische Erfahrung der Autoren zusammen – unter anderem im Kontext von Domain-Driven Design.