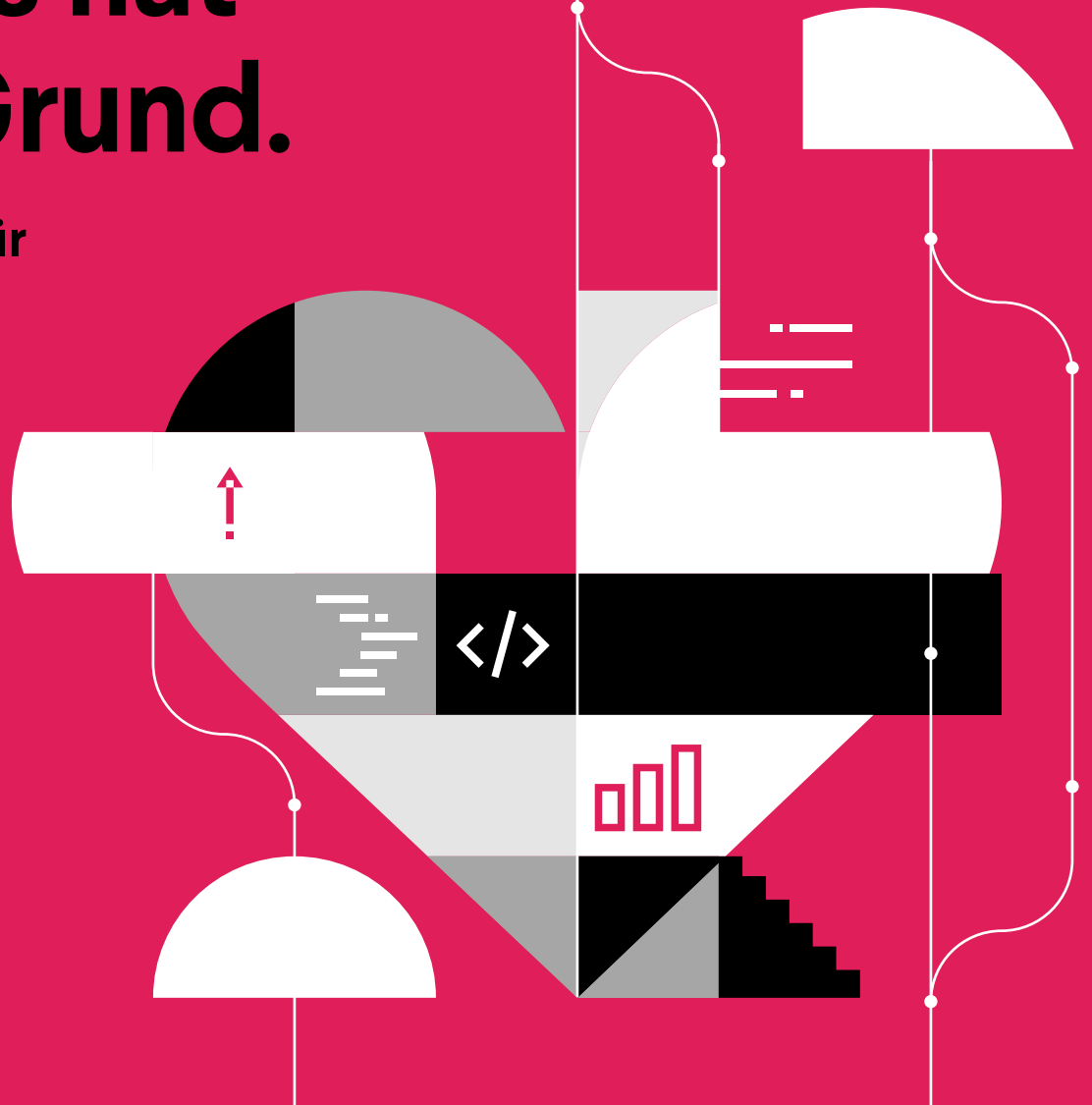


ENGINEERING

Entwickler-Teams lieben Slack – und das hat seinen guten Grund.

Die adaptive, zentrale Plattform für Zusammenarbeit in der Software-Entwicklung



Wow, das ging schnell!

Im Jahr 2009 waren wir ein kleines Team von Software-Entwicklern, die ein Massive-Multiplayer-Game namens Glitch entwickelten.

Wir fingen an, Internet Relay Chat zu benutzen (erinnerst du dich an IRC?), um alle auf demselben Stand zu halten. Im Laufe der Arbeit an dem Spiel wünschten wir uns immer wieder, dass unser IRC-Channel mehr als nur einfache Kommunikation leisten könnte. Also begannen wir, ihn zu optimieren, zu ergänzen und neue Möglichkeiten hinzuzufügen, um die Dinge schneller zu erledigen.

Nun ja, das Spiel war ein Flop, also haben wir uns auf dieses neue Zusammenarbeits-Ding konzentriert, das wir entwickelt hatten.

Das war eine gute Entscheidung. Denn letztendlich hatten wir einen hervorragenden Product-Market Fit: eine wirklich effektive Möglichkeit für Teams, zusammenzuarbeiten.

Vielleicht liegt das daran, dass wir nicht versucht haben, etwas zu „designen“. Es gab kein Ego und keine Spekulationen über einen fiktiven Benutzer. Wir waren die Nutzer.

So wurde Slack geboren.

Heute ist Slack größer geworden, als wir es uns jemals erträumt hätten – und die Einführung durch Entwickler-Teams jeder Größenordnung ist ein entscheidender Grund dafür.

Ja, Slack wird in so ziemlich jeder Abteilung und Nische eingesetzt, doch Software-Entwicklung ist der Bereich, in dem alles seinen Anfang nahm. Und gerade von Software-Entwicklern bekommen wir nach wie vor viel positives Feedback – einer der Gründe, weshalb wir morgens lächelnd zur Arbeit kommen.

Es ist schön zu wissen, dass wir etwas schaffen können, was sich direkt auf den Arbeitsalltag von Entwicklern auswirkt.

In diesem E-Book wird anhand einiger Beispiele erklärt, wie Entwickler Slack in ihrem Arbeitsalltag nutzen.

Wir hoffen, dass es ein wenig Aufschluss darüber gibt, warum gerade Entwickler offenbar sehr begeistert von Slack sind.



Warum Slack so gut zur Software-Entwicklung passt

Alle erdenklichen nicht-technischen Teams nutzen Slack, und zwar jeden Tag. Die Lösung scheint sich organisch an die jeweilige Arbeit anzupassen.

Doch es hat den Anschein, dass Slack zur Software-Entwicklung passt wie die Faust aufs Auge. Schließlich handelt es sich dabei um eine ziemlich spezialisierte Art von Arbeit.

Software-Entwicklung ist:

- **Komplex**
Mit vielen beweglichen Teilen
- **Hochgradig iterativ**
In Form von Sprints und Zyklen
- **Gemeinschaftlich**
Mit Designern, Entwicklern, Produkt- und QA-Leuten, die in ständigem Kontakt sind
- **Verteilt**
Oft sind Leute an verschiedenen Standorten und in unterschiedlichen Zeitzonen beteiligt
- **Offen**
Teilen ist der Standardmodus
- **Zunehmend automatisiert**
Unterstützt von jeder Menge Tools und Cloud-Services

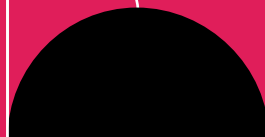
Wenn man darüber nachdenkt, entwickelt sich die Arbeit im Allgemeinen immer mehr in diese Richtung. Aber Software-Entwicklung ist ein idealer Anwendungsbereich. Sie erfordert eine Art von Zusammenarbeit, die mit E-Mails und persönlichen Meetings einfach nicht möglich ist.

Für diese Art von Tätigkeit wird eine neue Form der Zusammenarbeit benötigt.

Die adaptive, zentrale Plattform für Zusammenarbeit: etwas völlig Neues

Leute, die Slack noch nie benutzt haben, betrachten die Plattform als eine Messaging-App. Doch Slack ist sehr viel mehr als das. Tatsächlich ist Slack etwas ganz und gar Neues, das es vorher einfach nicht gegeben hat.

Wir bezeichnen Slack als eine adaptive, zentrale Plattform für Zusammenarbeit, weil sich die Lösung an die Arbeitsweise verschiedener Teams, an ihre bestehenden Software-Entscheidungen – und an Veränderungen – anpasst.



Eine adaptive Plattform für Zusammenarbeit vereint drei Dinge in einem Tool

1.

Channel-basierte Nachrichtenübermittlung

Auf diese Weise können Teams Channels erstellen, die für bestimmte Aufgaben, Projekte oder Probleme bestimmt sind. Zum Beispiel einen Channel #entw-neue-website, in dem sich alle Entwickler zusammenfinden, um an der neuen Website zu arbeiten. Oder einen Channel #triage-mobile-app, in dem Teams zusammenarbeiten, um Fehler in der mobilen App zu beseitigen.

Channels sind viel besser als Einzelnachrichten oder geschlossene E-Mail-Threads, weil sie es einfach machen, die richtigen Leute zur richtigen Zeit in die richtigen Themen einzubeziehen.

2.

Ein durchsuchbarer Wissensspeicher

Ein zentraler Ort, an dem alle Beteiligten sämtliche relevante Dokumente, Unterhaltungen und Entscheidungen finden – wie z. B. bestimmte Produktspezifikationen oder die Unterhaltung über diese neue Funktion von neulich.

E-Mail-Anhänge verschwinden in der Regel für alle aus dem Blickfeld, die nicht in der E-Mail auf CC gesetzt sind. Doch Wissen ist nur dann wertvoll, wenn es auch auffindbar ist.

3.

Eine Integrationsschicht

Ein Ort, an dem sich die Software, in der deine Mitarbeiter den Großteil ihrer Zeit verbringen, (z. B. GitHub, Jira, Jenkins und Trello), mit dem Ort verbindet, an dem die Arbeit tagtäglich diskutiert wird.

Dadurch wird der ständige Kontextwechsel minimiert, der durch die Arbeit in vielen verschiedenen Apps entsteht. Anstatt deine Leute zu zwingen, zu den Apps zu gehen, bringst du die Apps zu deinen Leuten.

Hinweis:

Indem diese drei Dinge an einem einzigen Ort zusammengeführt werden, wird das Potenzial jeder einzelnen Komponente noch erhöht. Die zentrale Plattform ist weitaus mehr als die Summe ihrer Teile.

Vorteile für Teams in der Software-Entwicklung

Die richtige Plattform für Zusammenarbeit hat einen direkten Einfluss auf die Dinge, die für jedes Entwickler-Team am wichtigsten sind: besserer Code, der schneller geliefert wird, effizientere Fehlerbehebung und eine bessere Erfahrung für Entwickler (damit deine Mitarbeiter glücklich sind – und bleiben). Jede Software, die dir bei alledem hilft, ist wahrscheinlich einen Blick wert.

„Slack ist eine dynamische Dokumentationsplattform, und alles ist durchsuchbar.“

Malika Rajvanshy, Senior Engineer, Slack

1. IDC-Studie, „The Business Value of Slack, 2017“, gesponsert von Slack

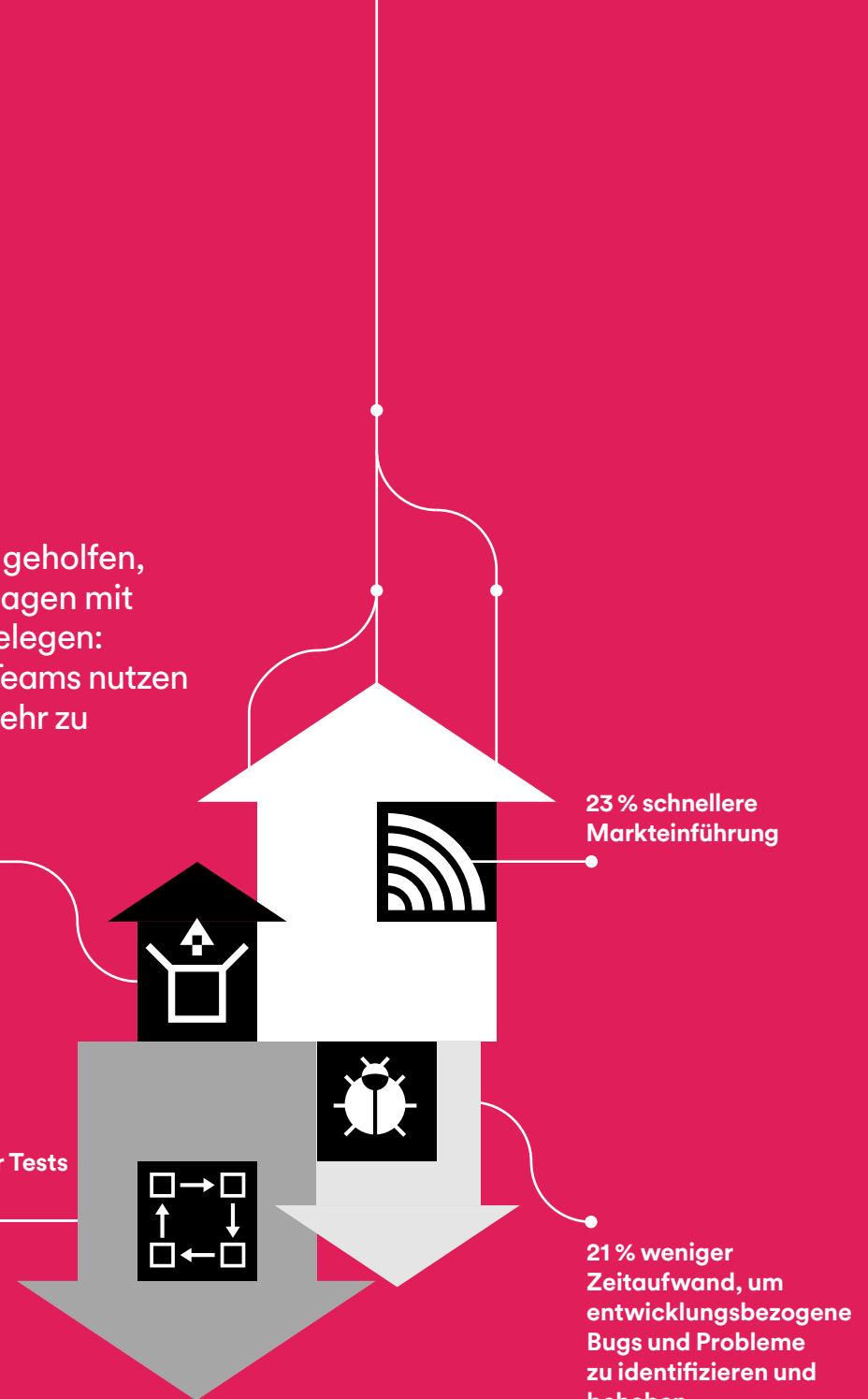
IDC hat uns geholfen, unsere Aussagen mit Zahlen zu belegen: Entwickler-Teams nutzen Slack, um mehr zu erledigen.¹

5 % höhere Gesamtleistung

23 % schnellere Markteinführung

27 % weniger Zeitaufwand für Tests und Iterationen

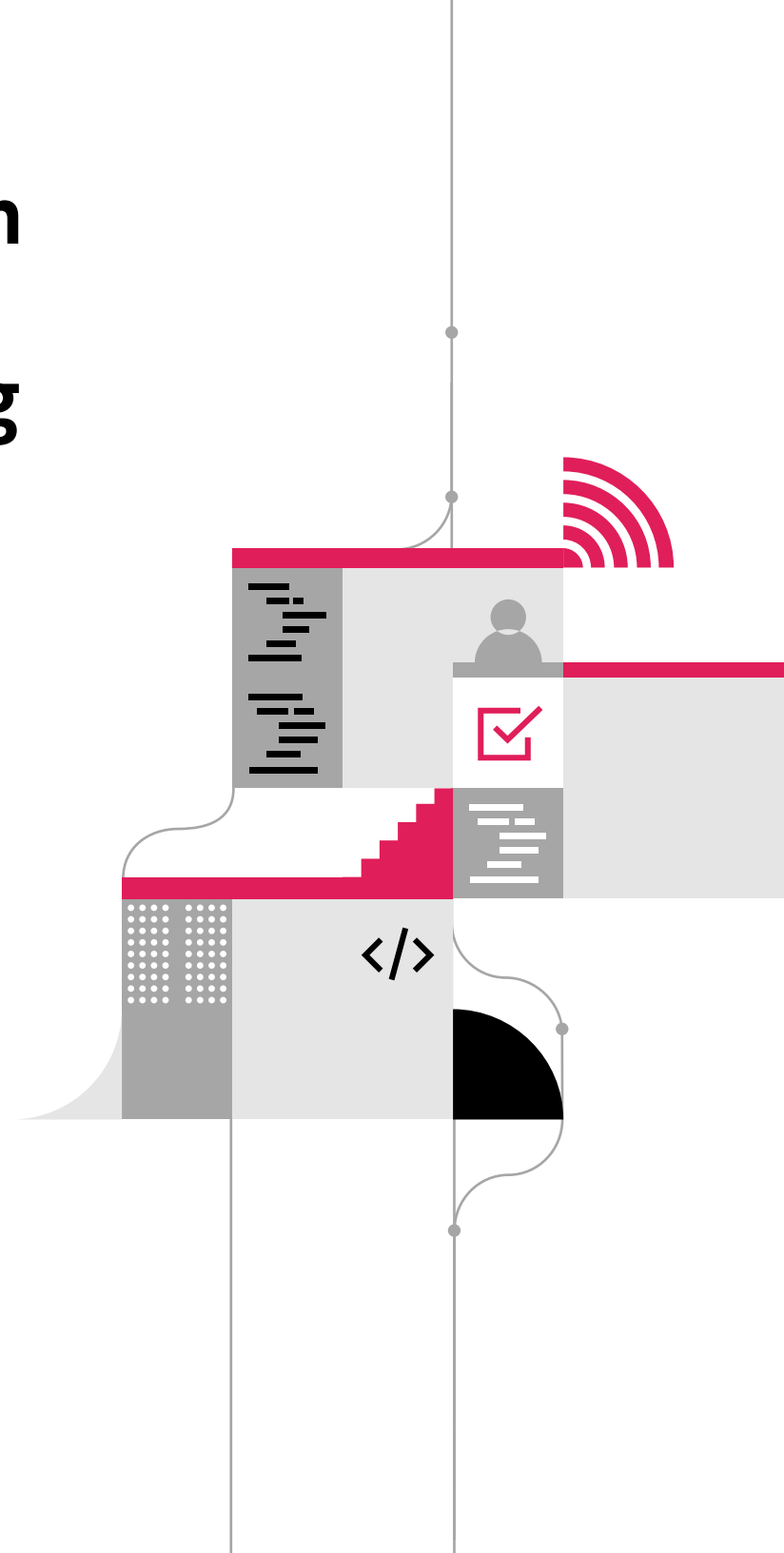
21 % weniger Zeitaufwand, um entwicklungsbezogene Bugs und Probleme zu identifizieren und beheben



So optimiert Slack den gesamten Prozess der Software-Entwicklung

Wir sind zweifelsohne einige der erfahrensten Nutzer von Slack für die Software-Entwicklung – alle unsere Entwickler-Teams leben quasi in der App. Doch jeden Tag hören wir nach wie vor von neuen Anwendungsbereichen und interessanten Apps und Integrationen, die Software-Teams nutzen.

Gehen wir ein paar von ihnen durch, organisiert nach Phasen im Software-Entwicklungszyklus.



Planen

Slack hilft Produkt-Managern, Designern und Entwicklern, sich darüber einig zu werden, was sie entwickeln und warum.

- **Den gesamten Prozess mit einem einzigen Channel für ein neues Produkt oder eine neue Funktion starten**

Er könnte einen Namen wie #funktion-neu-app haben.

Jetzt gibt es einen einzigen Ort, um das Projekt auszuarbeiten, Funktionsanforderungen zu sammeln, Alternativen zu diskutieren und grundlegende Entscheidungen bezüglich Funktionalität und UX zu treffen.

- **Wenn Dokumente hier geteilt werden, ist alles stets auffindbar**

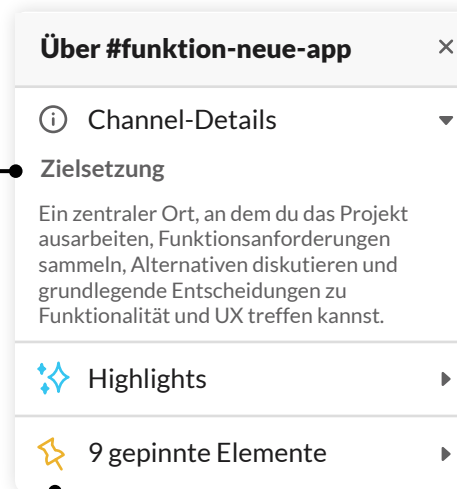
Für alle Beteiligten und eventuelle Neuankömmlinge.

Slack lässt sich wunderbar mit Google Docs integrieren, sodass alle Dokumente nur einen Klick entfernt sind.

- **Du hast eine Frage? Stelle sie einfach im Channel**

Starte eine Diskussion und erhalte eine Lösung, die alle sehen können. Jetzt ist das Ganze dauerhaft festgehalten.

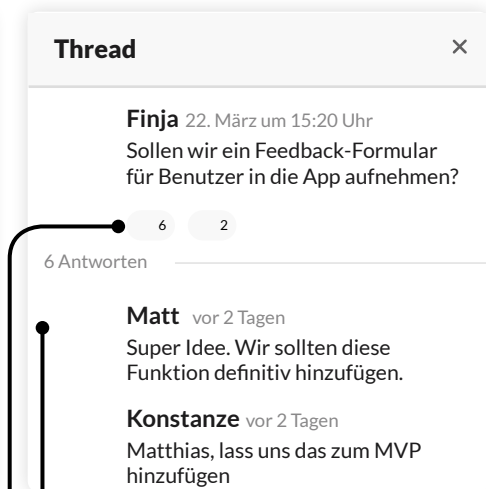
In einem #funktion_[name]-Channel spielt sich alles ab



Alle wesentlichen Dokumente oben im Channel pinnen

Ideal für das Onboarding!

Starte eine Diskussion über eine vorgeschlagene neue Funktion



Ein Thread entwickelt sich

Emojis sind wie Wahlstimmen

Code

Slack hilft Entwicklern, die vielen beweglichen Teile einer großen Code-Basis aufeinander abzustimmen, was die Entwicklung beschleunigt und die Qualität verbessert.

Wenn es an der Zeit ist, mit dem Programmieren zu beginnen, sorgt Slack dafür, dass das gesamte Team am selben Strang zieht:

- **Ein Channel #entw-produkt-name beherbergt alles**
Einschließlich der täglichen Arbeit in den Bereichen Entwicklung und QA; Pull-Anfragen, Code-Merges, Design-Revisionen, täglicher Standup-Meetings, Diskussionen usw.
- **Eine zentrale Plattform für die Code-Prüfung**
Slack unterstützt jeden Prozess, den Entwickler zum Verzweigen, Zusammenführen, Prüfen und Veröffentlichen von Code verwenden – unabhängig davon, ob die Entwicklung anhand von Versionszweigen, Funktionszweigen oder eines zusammengeführten Masters erfolgt.

Git-Integrationen (mit GitHub, Bitbucket oder dem Repository deiner Wahl) stellen sämtliche Änderungshinweise direkt in Slack zur Verfügung.

Git-Integrationen halten alle auf dem neuesten Stand.

The screenshot shows a Slack channel interface for '# deploys'. At the top, there are icons for a star, 146 members, 0 reactions, and a link to 'deploy web channel'. A search bar is on the right. The main content shows a notification from 'Checkpoint' at 10:11 AM. The notification includes a warning icon, a pull request icon, and text: 'aferrick has opened a pull request: PR #142930 in slack/webapp'. Below this, there is a link to 'PR #142930' with a video icon, followed by the text: 'Video unfurling for Usertesting.com behind a FF 7 commits on branch usertesting_unfurls'. Below that, there is a notification from 'Mark Grenier' at 10:15 AM, with a link to '@Leland Foster' and text: 'follow up on sonic apps list PR https://checkpoint.tinyspeck.com/142937'. Below this, there is a link to 'PR-142937: sonic-apps-list-unified-2' with the text: 'Tests Running · 3 commits by Alex · View PR in Github | Checkpoint'. At the bottom of the notification, there are two icons: a person icon with '1' and a checkmark icon with '1'.

Eröffne eine Pull-Anfrage und du wirst in Slack angepingt, wann immer jemand diese kommentiert

■ **Eine neue Art von Standup-Meeting**

Standup-Meetings sind ein wichtiger Teil der agilen Entwicklung, doch sie müssen nicht zwangsläufig von Angesicht zu Angesicht stattfinden. Entwickler-Teams benutzen Slack für Standup-Meetings – egal ob jeden Morgen oder jede Woche – und halten nur dann persönliche Meetings ab, wenn diese sinnvoll sind (für viele Entwickler ist das beste Meeting ein abgesagtes).

Integrationen mit Software wie Standuply übertragen zusammenfassende Berichte automatisch in Slack, damit deine Teams Ziele und Aufgaben teilen, Geschäftskennzahlen verfolgen, Meeting-Notizen posten sowie den Fortschritt und die Zufriedenheit des Teams im Blick behalten können.

Wiederverwendung von Code fördern

Die Wiederverwendung von Code ist ein Kernprinzip effizienter Entwickler-Teams, doch es stellt eine Herausforderung dar, wenn Hunderte von Entwicklern an vielen verschiedenen Produkten beteiligt sind. Bevor sie neuen Code schreiben, können deine Entwickler sämtliche Slack-Channels durchsuchen, um zu sehen, ob jemand anderes bereits etwas Ähnliches erstellt hat. Nächster Schritt – in den richtigen Channels fragen: „Hat schon jemand einen Terminplaner erstellt?“ Hör auf, das Rad neu zu programmieren.

Code mithilfe von Snippets erstellen und teilen

Mit Snippets ist es kinderleicht, Code, Konfigurations- und Log-Dateien direkt in Slack zu teilen. Kollegen können sie herunterladen, die unbearbeitete Datei anzeigen und Kommentare hinterlassen.

Slack in Aktion

Erweiterbarkeit als Kernstück



Slack ist eine zentrale Plattform für Zusammenarbeit. Das ist es, was diese Lösung am besten kann. Slack versucht nicht, die Arbeit der Software zu übernehmen, die deine Teams bereits nutzt – wie z. B. Trello, GitHub oder Jenkins.

Stattdessen vereint Slack einfach alle diese verschiedenen Apps und bringt die relevanten Informationen aus den Apps in die Channels, in denen die Arbeit besprochen wird (und bietet die Möglichkeit, direkt von Slack aus Aktionen in diesen Anwendungen auszulösen).

Diese Integrationen helfen Entwicklern, das zu machen, was sie lieben: *Systeme schaffen, die einfach funktionieren.*

Die in diesem E-Book präsentierten Beispiele sind nichts weiter als das: Beispiele. Es gibt so viele Möglichkeiten, Slack zu nutzen, wie es Software-Teams gibt, die Slack verwenden.

Bots: Apps werden Gesprächig

Ein Bot ist eine Art von Slack-App, die darauf ausgelegt ist, mit Benutzern in Form einer Unterhaltung zu interagieren. Sie sind genau wie normale Apps. Sie können auf dasselbe Spektrum an APIs zugreifen und all die Dinge tun, die eine Slack-App kann. Aber wenn du einen Bot für deine Slack-App entwickelst, gibst du dieser App ein Gesicht, einen Namen und eine Persönlichkeit und ermutigst Benutzer, mit ihr zu *reden*.

Dein Bot kann Direktnachrichten senden, er kann von Benutzern erwähnt werden, er kann Nachrichten posten oder Dateien hochladen, und er kann in Channels eingeladen (oder aus ihnen entfernt) werden.

„Wann immer ich eine Slack-Integration gesehen habe, habe ich sie aktiviert. Das hat einen gewaltigen Mehrwert geschaffen und uns so viele Schritte in unserem Prozess erspart.“

Thomas Lawless, Senior Software Engineer, IBM

Test

Das Testen ist fest mit dem modernen Entwicklungs- und Bereitstellungsprozess verflochten. Slack unterstützt einen dynamischen, kollaborativen und transparenten Ansatz für das Testen.

Durch die kontinuierliche Integration wird deine Test-Suite für jeden Merge-Vorgang mit jedem neuen Code-Block ausgeführt. Slack optimiert den Prozess in vielerlei Hinsicht, im Großen wie im Kleinen:

- **Ein Channel #testen-funktion koordiniert die QA**
Lass das QA-Team in einem offenen Forum mit Entwicklern zusammenarbeiten.

- **Jira-Integration automatisiert Test-Workflows**
Erfasse Issues in Slack und hole sie automatisch in die Pipeline. Sende anpassbare Benachrichtigungen von Jira in deine Channels. Weise Issues innerhalb kürzester Zeit Leuten zu und sei dir sicher, dass sie dort aufgezeichnet werden, wo sie hingehören.

Manche Teams benutzen Slack, um Änderungsanfragen automatisch in einen neuen Channel zu verschieben und gleichzeitig Trello oder Asana zu aktualisieren.

- **Einen Channel für jeden Client einrichten**
Mit speziellen Test-Channels für iOS, Android und Web.

Mit Jira integrieren, um den Projektfortschritt festzuhalten

The screenshot shows a Slack channel named '#testing' with 146 members. A message from Louise Forestier asks about a 'block_action' payload. David Brichau replies that he set a limit of 75 characters, which was incorrect. A search bar shows commands matching '/jira', with a highlighted command '/jira @assignee [project key] [summary] Creates a task in jira'. A sidebar on the right shows a menu with options like 'Create issue... JIRABot', 'Push to Zapier...', and 'Create a poll Polly'.

Kollegen bringen ihre Ideen ein

Jira-Aufgaben in Slack erstellen

Von Slack aus ein Jira-Ticket eröffnen, um einen Fehler zu melden

Slack in Aktion

Mit Jenkins arbeiten



Viele Teams nutzen Jenkins als ihren Server für kontinuierliche Integration. Es dauerte nicht lange, bis sie neue Möglichkeiten entwickelt hatten, um Jenkins mit Slack zu integrieren und sämtliche Arten von Routine-Entwicklungsaufgaben zu automatisieren.

Ein Beispiel: Die benutzerdefinierte Slack-Integration eines Software-Teams erstellt einen Jenkins-Server, auf dem eine große Test-Suite läuft, wann immer ein Entwickler eine Pull-Anfrage eröffnet.

Wenn die Tests durchgeführt wurden, erscheinen die Benachrichtigungen in den richtigen Slack-Channels. Wenn der Code den Test nicht besteht, wird eine Benachrichtigung an den Entwickler gesendet.

Pipeline-Rückverfolgbarkeit

Mit der Durchführung von Code-Prüfungen, Tests und Bereitstellungen in Slack wird ein aktuelles Register der gesamten Pipeline von Anfang an erstellt und gepflegt. Wenn du auf ein Problem stößt, kann das Team die Tests und Merges überprüfen und einen Rollback auf eine frühere, bessere Version vornehmen, bis der Code korrigiert ist.

Wenn Code-Prüfungen durchgeführt oder Pull-Anfragen überprüft werden, finden Kommentare und Diskussionen auf GitHub statt. Aber jeder Kommentar wird auch in Slack gepostet und benachrichtigt den Inhaber der jeweiligen Pull-Anfrage. Die Aufgabe wird also auf GitHub durchgeführt und dann als Kopie in Slack gespeichert.

Veröffentlichung

Slack hilft dabei, Code in die Produktion zu bringen, indem die Lösung die Workflows und Benachrichtigungen automatisiert.

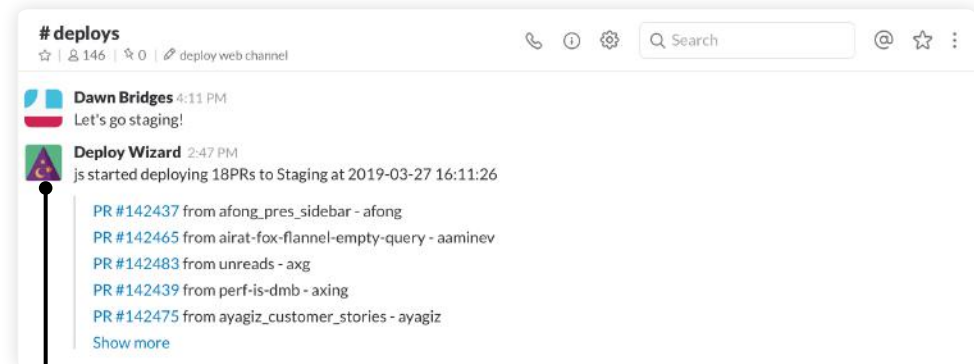
Eine kontinuierliche Lieferung erfordert generell viele kleine Code-Veröffentlichungen, die mit hoher Frequenz implementiert werden. Slack hilft Entwickler-Teams, bestimmte Aspekte davon zu optimieren.

Ein Beispiel: Eines unserer eigenen Software-Teams hat eine App namens Deploy Wizard programmiert, die sich in Arbeitsabläufe integrieren lässt und den Status des Codes im Channel kommuniziert. Sie beginnt mit einer „Kanarienvogel“-Phase (eine winzige Veröffentlichung, um plötzliche Ausfälle zu erkennen) und weitet diese dann schrittweise auf 10, 25, 75 und 100 % der Benutzerbasis aus.

Deploy Wizard pingt im Laufe der Bereitstellung die richtigen Entwickler und Channels in Slack an. Das Ganze wird von den jeweils zuständigen Bereitstellungsleitern (speziell ausgebildete Entwickler, die in Dreistundenschichten arbeiten) gemanagt.

Wenn Entwickler ihren Code in der Staging-Umgebung testen wollen, geben sie das in ihrer Merge-Anfrage an. Die Bereitstellung wird in der Staging-Phase angehalten, bis ein Entwickler im Bereitstellungs-Channel meldet, dass er oder sie den Code getestet hat.

Eine Bereitstellung geht zur vollen Produktion über



Der Deploy Wizard automatisiert Benachrichtigungen: Alles läuft reibungslos!

Direkt von Slack aus bereitstellen

Einige Entwickler-Teams benutzen Slash-Befehle (wie z. B. `/deploy_produktnamen_staging`), um eine Bereitstellung direkt über Slack auszulösen. Automatisierte Nachrichten informieren, wenn die Bereitstellung erfolgreich war, mit einem Link, um sie zu überprüfen (oder einem Button, um sie in die Produktion weiterzuleiten).

Operative Abläufe

Entwickler-Teams nutzen Slack, um Fehler-Tickets zu testen, sich gemeinsam Issues zu widmen und Fehler zu beseitigen.

- **Alle Issues fließen durch den Channel #triage-produktname**
 Einschließlich Berichte vom Kunden-Support (manuell oder über Integrationen mit Tools wie Zendesk).
- **Integrationen bringen alle Hinweise an einem Ort zusammen**
 Anstatt zu erwarten, dass Entwickler E-Mails überwachen oder in Dashboards vorbeischaun, wird Slack zum zentralen Ort, an dem alle Hinweise die am besten geeigneten Leute erreichen, die dann reagieren können.

Die Aggregation von PagerDuty-Ereignissen oder Asana-Tickets und das Posten von diesen in den richtigen Channels verkürzt die Zeit bis zur Klärung von Vorfällen und schafft einen Triage-Trail. Team-Mitglieder können gemeinsam daran arbeiten, Vorfälle direkt über Slack auszulösen, einzusehen, zu bestätigen und zu beheben.

Ebenso kann Slack alle Web-, Transaktions-, Server- und mobilen Hinweise von New Relic für eine schnelle Reaktion in einen Slack-Channel abrufen. Alle, die neugierig auf den Vorfall sind, können einfach im Channel vorbeischaun und mehr erfahren. Dies reduziert die Anzahl der Manager, die die Reaktions-Teams für ständige Updates unterbrechen. Es ist alles da.

PagerDuty-Ereignisse erreichen schnell die richtigen Leute

Jaime Cunningham 7:04 PM
 /report-incident
 PagerDuty APP
 Triggered at #448: COE-INC73254 down
 Assigned: Chad Thomas
 Service: Derivative Service - PROD - Sev1
 Acknowledged by @Jaime | Today at 7:04 PM
 Acknowledge Resolve


Aktions-Buttons beschleunigen die Klärung direkt aus Slack heraus.


Emojis und Reacjis helfen bei der Fehlersuche und lösen Workflows aus.

Reacjis sind eine effiziente Möglichkeit, die Reaktionen von Team-Mitgliedern zu erfassen – aber sie sind auch eine Möglichkeit, automatisierte Arbeitsabläufe auszulösen.

Eine App sammelt diese, damit sie aggregiert, markiert und umgesetzt werden können. Alle offenen Issues (Augen-Emoji, aber kein Häkchen) werden in PagerDuty angezeigt.

Bei Slack benutzen wir:


 = „Ich sehe mir das an“


 = „Gelöst“ (für Triage) oder „Genehmigt“ (für Pull-Anfragen)

Und wir haben einen Bot entwickelt, der diese in einem speziellen Channel sammelt und über sie berichtet.

Zum Teilen von Bug-Meldungen benutzen wir außerdem:

 = Dringend!

 = Eine Frage oder ein nicht dringendes Problem

 = Feedback teilen, kein unmittelbarer Handlungsbedarf

Ein automatisierter Channel #entscheidungen

Manche Teams benutzen das Richterhammer-Emoji, um anzuzeigen, ob eine Entscheidung getroffen wurde. Ein Bot leitet dann alle diese Entscheidungen in einen Channel #entscheidungen weiter, wo das Management den Entscheidungsfluss einsehen kann – und die Team-Mitglieder mühelos suchen können.

Slack in Aktion

Der Personalaspekt



Software-Entwickler sind gefragt. Wenn du deine Talente halten möchtest, musst du ihnen die bestmögliche Mitarbeitererfahrung bieten.

Die richtige Software für Zusammenarbeit kann dabei eine große Rolle spielen: Sie trägt dazu bei, Reibungsverluste zu reduzieren, Transparenz zu schaffen, Routineaufgaben zu automatisieren und Team-übergreifend zu arbeiten.

Rede mit einem beliebigen Team für Software-Entwicklung, das Slack benutzt.

Frag nach, wie sie Channels, Apps und Integrationen verwenden.

Frag anschließend, was sie ohne Slack machen würden.

Slack in Aktion

Neue Entwickler onboarden



Zwei neue Entwickler kommen ins Team. Wie bringst du sie auf den aktuellen Stand?

Die alte Methode: Jede Menge Onboarding-Meetings und eine Reihe von weitergeleiteten E-Mail-Threads, aus denen sie vielleicht schlau werden können. Viel Glück dabei!

Die neue Methode: Lade sie in den Channel #entw-neu-produkt ein, um die gepinnten Beiträge zu lesen, wie z. B.:

- Die Produktspezifikationen
- Die technischen Spezifikationen
- Die Designs

(Wenn es sich dabei um Google Docs-, DropBox- oder OneDrive-Dokumente handelt, werden sie immer auf dem neuesten Stand gehalten).

Sie können außerdem alle bisherigen Unterhaltungen und Entscheidungen nachlesen und sehen, welche Personen beteiligt waren.

So sollte das Onboarding eines neuen Entwicklers laufen.

So nutzen Software-Entwickler Slack

Das war es auch schon mit unserer kurzen Tour dazu, wie Slack Software-Teams dabei unterstützt, ihre Arbeit zu optimieren, zu automatisieren und zu beschleunigen.

Wir hoffen, dass wir die wichtigsten Punkte vermitteln konnten:

- **Das ist etwas völlig Neues**
Eine adaptive, zentrale Plattform für Zusammenarbeit, die Entwicklern hilft, auf neue Art zu arbeiten. Slack ist weitaus mehr als eine Messaging-App.
- **Slack ist super flexibel**
Mit Workspaces, Channels, Apps und Integrationen, die ihre individuelle Arbeitsweise widerspiegeln, können sich deine Teams die App „ganz zu eigen machen“.

- **Slack hilft dir, mehr aus deiner bestehenden Software herauszuholen**
Von GitHub und Bitbucket bis hin zu Jenkins, Jira, PagerDuty, New Relic, Zendesk ... Was auch immer deine Entwickler oder deine Produkt-, QA- und Support-Leute benutzen – sie werden diese Tools effizienter nutzen, indem sie ihre Arbeit in Slack zusammenführen.
- **Slack schafft einen Mehrwert in jeder Phase des Entwicklungszyklus**
Von der Planung über Entwicklung, Testen, Betrieb bis hin zur Bereitstellung und Fehlerbehebung.
- **Software-Entwickler lieben Slack**
Und das bedeutet, dass sie die Lösung annehmen und ihre Nutzung der App im Laufe der Zeit ausweiten werden (was wiederum den Mehrwert für das Unternehmen steigert).

Wenn du mehr sehen möchtest, vereinbare einen Termin für eine Demo – oder bitte einen unserer Entwickler, dir unsere eigene Slack-Instanz zu zeigen. Wir sind stolz darauf.

„Wir haben etwas, was wir als ‚End-to-End-Pipeline‘ bezeichnen, die mit dem Quellcode beginnt und sich bis zur *Produktimplementierung* durchzieht. Und jetzt haben wir Slack in alle wichtigen Meilensteinen für diesen Prozess integriert.“

Thomas Lawless, Senior Software Engineer, IBM

Mehr Infos

[So hilft Slack Entwicklern, Fehler zu beseitigen](#)

Erfahre, wie Entwickler-Teams Vorfälle und Ausfälle mit Slack besser managen

[Das App-Verzeichnis von Slack](#)

Schau dich um

Oder vielleicht sollten wir reden.

Vereinbare einen Termin mit unserem Team.

LOS GEHT'S

Über Slack

Slack ist eine Ebene des Technologie-Stacks für Unternehmen, die Menschen, Daten und Anwendungen zusammenführt – ein zentraler Ort, an dem Menschen effektiv zusammenarbeiten, wichtige Informationen finden und auf Hunderttausende von kritischen Anwendungen und Diensten zugreifen können, um ihre beste Arbeit zu leisten.

Von globalen Fortune-100-Unternehmen bis hin zu kleinen Läden wird Slack von Unternehmen und Teams aller Art genutzt, um die richtigen Personen mit den richtigen Informationen zusammenzubringen.

