



# Tool Time mit SUSE

Im Gespräch mit iX erzählen Andreas Jaeger und Dirk Müller, welche Tools in den global verteilten Entwicklungsteams von SUSE zum Einsatz kommen – und warum es doch nicht ganz ohne persönliche Treffen geht.

Von Jonas Volkert

**iX: Beginnen wir mit der Kommunikation: Welche Tools setzen die Entwickler bei SUSE ein, um sich intern abzusprechen?**

**Dirk Müller (DM):** In der Entwicklerkommunikation verwenden wir bei SUSE verschiedene Kommunikationstools, viele davon sind offen für die Communitys: E-Mail und Mailinglisten, interne und externe IRCs und unterschiedliche Slack-Instanzen. Seit Ende letzten Jahres nutzen wir zusätzlich SUSE-intern Slack für bereichsübergreifende Kommunikation.

Der größte Teil der Abstimmungskommunikation findet in Issue-Trackern statt, für verschiedene Bereiche, übergreifend und in den meisten Fällen offen für die Communitys. Dabei folgen wir den Bedürfnissen und Ansprüchen der mit uns verbundenen Communitys, externen Partnern und Open-Source-Projekten und arbeiten wo möglich mit den vom jeweiligen Partner bevorzugten Tools.

**Andreas Jaeger (AJ):** Allgemein nutzen wir für unsere Meetings Microsoft Teams. Ad hoc treffe ich mich mit Dirk auch oft über einen eigenen Huddle in Slack. Als Entwicklerteams sind wir sehr flexibel und nutzen zum Beispiel für Diskussionen mit externen Communitys die openSUSE-Jitsi-Infrastruktur.

**iX: Stellen offene Kommunikationswege nicht auch eine Gefahr aus Security-Sicht dar?**

**DM:** Für alle Arbeiten und die Kommunikation zu Produkten, die zum Beispiel Common-Criteria-zertifiziert sind oder anderen Regularierungen unterliegen, werden nur dedizierte, isolierte Kommunikationssysteme verwendet. Offene Kommunikationskanäle werden nur für die nicht regulierte Arbeit verwendet.

**AJ:** SUSE hat umfangreiche Zertifizierungen durchlaufen. Entsprechend gelten für diese Security-Zertifizierung Anforderungen, die bis ins kleinste Detail festlegen, welche Tools wir wofür nutzen dürfen und welche nicht. Bei Se-

curity-Vulnerabilities zum Beispiel ist klar definiert, was auf welchem Weg besprochen werden darf.

**iX: Noch einmal zur Kommunikation über Slack. Sie sagten, die Instanz würden Sie erst seit letztem Jahr nutzen. Welche Entwicklungen gab es da?**

**DM:** IRC und E-Mail nutzen wir fast seit dem SUSE-Gründungstag – am 2. September sind das 30 Jahre, die unser Unternehmen besteht. In den letzten Jahren kam noch eine Rocket.Chat-Instanz dazu. Das war eine on Premises gehostete Instanz, die hauptsächlich vom Engineering eingesetzt wurde. Im Rahmen eines Konsolidierungsprojektes nutzen wir unternehmensweit Slack. Parallel dazu gibt es speziell für das Engineering wie bereits erwähnt weitere Kommunikationswerkzeuge. Vor SUSEs Eigenständigkeit haben wir natürlich auch die Tools des damaligen Mutterkonzerns genutzt und dann gibt es noch persönliche Präferenzen.



**Andreas Jaeger arbeitet bei SUSE als Senior Produktmanager an SUSE-Produkten, die SAP-Kunden helfen. Als Open-Source-Enthusiast hat er an Projekten wie der GNU C Library, dem Port von Linux auf x86-64, openSUSE, OpenStack und Kubernetes mitgearbeitet.**

**iX: Und jetzt genießt das Unternehmen die Freiheit der Toolwahl?**

**DM:** Das Thema Tools wird heiß diskutiert. Viele Gruppen haben eigene Präferenzen und Gründe, warum sie genau dieses bestimmte Tool nutzen möchten. Mit SUSEs Eigenständigkeit (siehe Kasten „Firmengeschichte von SUSE“) konnten wir im Rahmen eines IT-Projektes den natürlichen Wildwuchs konsolidieren und ein standardisiertes Tool-Set-up bauen. Die erste Phase des Projekts ist abgeschlossen und wir hosten unsere Tools eigenständig, haben die Log-in-Systeme harmonisiert und die Auswahl der Tools konsolidiert.

**iX: Kommen wir nun zu den Entwicklungsteams und der Projektarbeit. Welche Planungstools sind da bei SUSE im Einsatz?**

**AJ:** Entsprechend unserem Community- und Partneransatz nutzen wir verschiedene Tools, in erster Linie Jira und Confluence, einige Teams nutzen Trello. Zusätzlich existieren die Wiki-Seiten im openSUSE-Wiki. Was noch, Dirk?

**DM:** Wir nutzen zum Beispiel das Whiteboarding-Tool Lucidspark, etwa zur Organisation von Sticky Notes. Das ist gut fürs Brainstorming oder die kollaborative Ideenfindung. Für konkrete technische Tasks, Taskarbeiten oder Bugtracker, nutzen wir zum Beispiel eine Bugzilla-Instanz, die communityoffen ist. Fürs Feature-Tracking, etwa für Informationen, die für externe Partner interessant sind, verwenden wir eine eigene Jira-Instanz, die on Premises gehostet wird, um so die Kollaboration auf einer gemeinsamen Plattform zu ermöglichen. Open-Source-Projekte verwenden natürlich die jeweiligen vom Projekt gehosteten Tracking-Tools. Für interne Tools verwenden wir häufig eine selbst gehostete GitLab-Instanz. openSUSE nutzt ebenfalls eine eigene Infrastruktur, verwendet aber auch Issue-Tracker in GitHub. Ähnlich ist es bei den Rancher-Projekten, große Teile der Projekt-Issues laufen extern in GitHub.

**iX:** *Das klingt, als sei das alles sehr organisch gewachsen. Ist es teamabhängig, welche Tools sich etablieren?*

**AJ:** Zum einen ist es teamabhängig, zum anderen aber auch historisch. Wie gesagt, dies ist auch eine Frage, mit welchen Community-Projekten wir zusammenarbeiten. Jira ist ein gutes Beispiel: Wir hatten ein eigenes selbst geschriebenes Tool fürs Feature-Tracking namens Fate, das wir durch Jira ersetzt haben, weil es einfach flexibler ist.

**iX:** *Das funktioniert aber nur mit Aufgaben, die ausschließlich innerhalb des Teams erfüllt werden müssen, weil sonst die Synchronisation schwierig ist, oder?*

**AJ:** Das ist richtig. Wir arbeiten bei SUSE an verschiedenen Produkten und Projekten, mit unterschiedlicher Koordination und Abhängigkeiten. Ein Rancher-Produkt-Manager ist weitgehend unabhängig von der SUSE-Linux-Enterprise-Entwicklung, wohingegen SUSE Linux Enterprise Server und SUSE Linux Enterprise High Availability viele Gemeinsamkeiten haben. Bei den Produkten der SLE-Family ist es daher verpflichtend, unabhängig davon wie die Teams arbeiten, dass sie Jira-Issues annehmen und mit ihnen als Input umgehen. Ein SLES-Release wird komplett in Jira geplant.

**iX:** *Wie harmonisieren Sie ganz konkret die Ergebnisse der Programmierarbeit untereinander, im Team?*

**DM:** Das hängt von der Produktlinie ab. Bei der SLE-Produktlinie gibt es eine klare Policy, die Factory-first-Policy, die extern dokumentiert ist. Das heißt, relevante größere Änderungen müssen zuerst in openSUSE Tumbleweed eingepflegt werden. Die Integration erfolgt also zuerst in openSUSE und dort läuft das mit dem Tooling und der Zusammenarbeit, die wir uns für die openSUSE-Community wünschen: mit einem externen Feature-Tracker, einem externen Wiki und oft auch einem Bugzilla. Vermittelt werden die Sachen im Open Build Service. Dort gibt es einen CI-artigen Workflow für das Staging. Sie durchlaufen dann verschiedene Review- und Tes-



**Dirk Müller arbeitet als Distinguished Engineer bei SUSE an den SUSE-Linux-Enterprise- und openSUSE-Distributionen. Mit über 25 Jahren im Open-Source-Umfeld war er an Projekten wie KDE und OpenStack beteiligt.**

ting-Phasen. Änderungen werden nur dann integriert, wenn alle den Daumen hoch gegeben haben, sowohl der automatische Test als auch ein händischer, persönlicher Review. Wenn dieser Schritt durchlaufen ist, dann können größere Änderungen auch in die SLE-Codebasis zurückportiert werden. Dieser Prozess, über verschiedene Phasen des Hardnings und des Nachschleifens von Änderungen und schließlich die Überführung in ein SUSE-Produkt, gilt immer.

**iX:** *Wie wird die Zusammenarbeit dabei koordiniert?*

**DM:** Der Open Build Service hat eigene Kollaborationsmöglichkeiten mit einem integrierten Submit- und Review-System. Das sorgt dafür, dass automatisierte Checks auch an der gleichen Stelle ihre Fehler oder Anmerkungen positionieren. So ist alles gemeinsam sichtbar und auch für Externe verfügbar. Die Entwicklung muss eben möglichst offen sein. Es ist häufig so, dass in dieser Phase Feedback von interessierten Enthusiasten und/oder Partnern kommt, das entscheidend ist für die weitere Featureentwicklung.

Das kann bei anderen Produktlinien anders sein: SUSE hat im Oktober letzten Jahres Neuvector gekauft und schon im Januar dieses Jahres zu Open Source gemacht. Die Rancher-Produkte sind von Anfang an sehr stark auf

die Tools der Cloud-native-Entwicklung ausgelegt. Die Projekt-Repositorys sind öffentlich auf GitHub und der größte Teil wird öffentlich über Issues und Pull Requests getrackt.

**AJ:** Open-Source-Anbieter wie SUSE haben eine andere Art der Entwicklung als ein Softwareunternehmen, das seine Entwicklung oder große Teile davon nicht öffentlich macht. Teilweise arbeiten wir in Open-Source-Projekten, in denen wir komplett im Rahmen der Projekte entwickeln, so etwa bei Rancher. Anders die Entwicklung von SUSE Linux Enterprise, wo wir zum einen Komponenten haben wie YaST, die wir in GitHub entwickeln und die offen für die Community sind. An anderen Komponenten wie dem Linux-Kernel, der C-Library, dem Compiler arbeiten wir mit. Der Linux-Kernel ist ein gutes Beispiel: Wir arbeiten upstream und schicken die Changes an die Upstream-Maintainer. Es kann aber auch sein, dass ein bestimmter Fix wichtig ist für einen Hardware-Bugfix bei einem Kunden, den submitten wir direkt mit in unsere Codebasis.

**DM:** SUSE hat eine öffentlich dokumentierte Open-Source-Policy. Im Wesentlichen beschreibt diese, wie die Entwicklerteams in Open-Source-Projekten arbeiten.

**iX:** *SUSE-Entwickler sitzen weltweit verteilt. Beobachten Sie bei der Präferenz der eingesetzten Tools Unterschiede zwischen den einzelnen geografischen Arbeitsorten?*

**DM:** Das ist eine interessante Frage. Ich glaube schon, dass es geografisch zuordenbare Vorlieben für gewisse Tools gibt. Vorlieben bezüglich des persönlichen Datenschutzes sind zum Beispiel unterschiedlich stark ausgeprägt. Die eigentliche Toolwahl hängt meines Erachtens aber weniger von der Region ab, sondern von der Rolle und den Anforderungen des Teams und der Mitarbeiter:innen. Entwickler:innen haben andere Präferenzen bezüglich des Toolings als Vertriebsmitarbeiter:innen oder Mitarbeiter:innen, die nicht im Technikbereich arbeiten. Das ist einer der Gründe, warum wir eine Kombination unterschiedlicher Tools nutzen. Der zweite Grund ist die Offenheit und Anpassungsfähigkeit, weil wir mit vielen Communityprojekten und unterschiedlichen Partnern und Kund:innen zusammenarbeiten. Das hat natürlich auch zur Folge, dass wir oft deren Werkzeuge nutzen.

**iX:** *Und wie sieht das mit dem Kontakt nach und dem Einfluss von außen konkret aus?*

**AJ:** Die Communitys spielen eine große Rolle: Die OpenStack-Kommunikation nutzt IRC und E-Mail, die Kubernetes-Community ver-

## TRACT

- ▶ SUSE-Entwicklungsteams arbeiten in verschiedenen Teilen der Welt. Im Interview erklären Andreas Jaeger und Dirk Müller, die beide bereits lange im Unternehmen tätig sind, wie sich die Kommunikation über Tools im Laufe der Zeit verändert hat.
- ▶ Geprägt ist die Toolauswahl vom Teamkontext. Bei der Zusammenarbeit mit Open-Source-Projekten richten sich die Mitarbeiter nach dem Projektstandard, bei sicherheitsrelevanten Aufgaben kommen hingegen nur dedizierte, isolierte Kommunikationssysteme zum Einsatz.
- ▶ Für Chats nutzt SUSE seit letztem Jahr eine Slack-Instanz, Video-Calls laufen über Microsoft Teams. Dabei herrscht eine „Everybody dials in“-Policy: So gewährleisten die Teams gleiche Partizipationschancen für alle.

wendet Slack – Slack und Meetings, kaum E-Mail. Die Open-Source-Communitys sind ganz unterschiedlich und bauen auf ihren Erfahrungen auf.

**iX: Ein wichtiger Teil der Kommunikation ist auch der persönliche Kontakt. Wie sieht die Homeoffice-Policy bei SUSE aus?**

**AJ:** SUSE agiert weltweit, so arbeiteten schon vor der Pandemie viele Mitarbeiter:innen im Homeoffice. Am Standort Nürnberg arbeitete die große Zahl der Mitarbeiter:innen im Büro. Schon vor der Pandemie hat SUSE eine Agile-Work-Policy entwickelt, die den Mitarbeiter:innen flexibles Arbeiten ermöglicht. Sie können wählen zwischen reinem Homeoffice, einer Mischung aus Homeoffice und Büro sowie Büro. Natürlich gibt es weiterhin Funktionen, wo das nicht möglich ist, etwa am Empfang oder vor Ort bei Kunden und Partnern. Rund die Hälfte der Mitarbeiter:innen arbeitet von zu Hause.

**iX: Besonders zwischenmenschlich gibt es aber ja durchaus Vorteile persönlicher Treffen. Gibt es Termine, an denen sich alle dann doch wieder einfinden müssen?**

**AJ:** Pandemiebedingt hatte ich letztes Jahr nur ein oder zwei Treffen, bei denen zumindest die Teammitglieder aus Deutschland physisch im Office waren. Kolleg:innen aus anderen Ländern mussten per Videokonferenz teilnehmen. Wir veranstalten je nach Produkt oder Team Workshops mit Entwickler:innen, Produktmanagement, teilweise Marketing und Vertrieb, diskutieren und arbeiten ein festes Programm ab. Da gibt es dann auch viel Raum zum Kennenlernen.

**DM:** Pro Projektgruppe oder -linie gibt es mindestens einmal im Jahr einen Kick-off-Workshop von 3 bis 5 Tagen, um die Zusammenarbeit für das nächste Projektrelease zu maximieren. Dazu gehört oft auch Retrospektive: Was hat gut funktioniert, was nicht so gut, wo gab es Feedback, wie ist die Sicht von Kunden und Partnern, welche Anforderungen und Trends gibt uns der Markt vor? Durch die Pandemie hat sich viel auf virtuelle Meetings verlagert. Auch wenn die Resultate der virtuellen Meetings gut sind, persönlich finde ich virtuelle Treffen anstrengender als In-Person-Meetings. Andererseits gibt es mittlerweile viele Tools für Whiteboarding oder Echtzeitkollaboration, die die Zusammenarbeit sehr produktiv gestalten. Jeder kann schreiben und lesen an der Stelle, die relevant ist. Diese Freiheit ist auch ein Produktivitätsgewinn.

**iX: Ein Abschied also von persönlichen Zusammenkünften?**

## Firmengeschichte von SUSE

Im Spätsommer 1992 gründeten Roland Dyroff, Thomas Fehr, Burchard Steinbild und Hubert Mantel in Fürth die Firma S.u.S.E – das noch etwas umständlicher geschriebene Akronym steht damals wie heute für „Software und System-Entwicklung“. Initial als Vertrieb einer deutschen Variante des Softlanding Linux System (SLS) gestartet, stieg das Unternehmen 1994 auf Slackware um und veröffentlichte mit S.u.S.E. 1.0 die „Deutsche Slackware, LST, DLD“. Die erste eigene Linux-Distribution veröffentlichte S.u.S.E. 1996.

Den ersten Schritt in Richtung globale Ausrichtung unternahm S.u.S.E 1997 mit der Eröffnung eines US-Büros, im Folgejahr erschien mit S.u.S.E. Business Linux die erste Distribution speziell für Unternehmen. Im selben Jahr folgte der Umzug ins benachbarte Nürnberg, in dem auch die heutige SUSE wieder ihren Unternehmenssitz hat. Innerhalb Deutschlands entstanden Standorte in München und später in weiteren Städten, was zusammen mit der internationalen Expansion den Grundstein für die heute verteilte Softwareentwicklung des Unternehmens legte.

Sein hohes Wachstumstempo versuchte SuSE zum Ende des Millenniums durch Investoren zu finanzieren: Intel, der Investmentfond Apax und später auch IBM stiegen ein. 2000 folgte mit SuSE 7.0 die Unterscheidung zwischen einer Personal und

einer Professional Edition, Ende des Jahres erschien mit SuSE Linux Enterprise Server for S/390 das erste Linux für Mainframes. Ende 2001 erblickte der SuSE Linux Enterprise Server (SLES) das Licht der Welt, der bis heute ein zentraler Teil des professionellen Produktkatalogs des Unternehmens ist.

Das Platzen der Dotcom-Blase traf auch SuSE und führte zur 2004 abgeschlossenen Übernahme durch das US-Tech-Unternehmen Novell. Zwei Jahre später ging mit openSUSE ein freies Projekt an den Start, das bis heute zur Basisentwicklung dient und als Testplattform für die SLES-Familie fungiert. Aus dem openSUSE-Projekt entstanden eigene Linux-Distributionen, unter anderem openSUSE Tumbleweed.

Nachdem 2011 Novell von Attachmate übernommen wurde, gründete man im selben Jahr SUSE wieder als eigene Geschäftseinheit aus. Dabei blieb es auch, als Micro Focus drei Jahre später Attachmate aufkaufte. 2018 verkaufte Micro Focus SUSE letztlich an die Investoren EQT Partners. Seitdem folgten eigene Übernahmen im Containerumfeld: 2020 Rancher Labs, Ende 2021 NeuVector. Im selben Jahr ging SUSE an die Börse. Nach wie vor liegt der Firmensitz in Nürnberg – inzwischen arbeiten aber weltweit laut Firmenangaben mehr als 2000 Menschen in über 30 Ländern bei SUSE.



**DM:** Ich denke, wir kehren wieder mehr zu In-Person-Meetings zurück. Unsere Herausforderung ist nicht so sehr der Ort, sondern die Zeitzone. Unsere Teams sind verteilt in Asien und Pazifik oder an der US-Westküste. Einzelne Kolleg:innen haben sich sogar ihren eigenen Wochenrhythmus gesetzt: An einigen Wochentagen arbeiten sie angelehnt an Asia-Pacific-Zeit, den Rest der Woche nach US-Zeit. Das ist auf Dauer aber nicht tragfähig angesichts Familie und anderer Verpflichtungen. Bei einem In-Person-Treffen haben wir den Vorteil, dass wir acht Stunden am Tag zusammen arbeiten und eben nicht nur zwei. Daher werden wir uns, sobald es die Situation erlaubt, auch wieder als Team treffen. Das bedeutet eine vierfache Vergrößerung der Bandbreite.

**iX: Und wie gelingt bei den virtuellen Konferenzen die Einbindung derer, die nicht vor Ort sind, die bei Teammeetings nur zugeschaltet sind?**

**DM:** Wir haben eine „Everybody dials in“-Policy. Auch wenn Mitarbeiter:innen sich persönlich treffen könnten, weil sie zufälligerweise im gleichen Büro sitzen, wählt sich jeder

ein. Damit haben alle Teilnehmer:innen die gleichen Bedingungen und erhalten die gleichen Informationen.

**iX: Wenn die Tendenz jetzt wieder verstärkt zu Präsenzzusammenkünften geht: Wo finden die dann statt? Und wie häufig?**

**AJ:** Für die Softwareentwicklung haben wir normalerweise ein physisches Planungsmeeting im Jahr pro Team. Unsere Meetings finden oft in Nürnberg statt, wo auch sehr viele Entwickler:innen sind, ich hatte aber auch schon Teammeetings in Kanada.

**DM:** Wir nutzen auch die Tage vor oder nach Community-Events oder wichtige Messen, an denen viele Kolleg:innen teilnehmen. Zur Häufigkeit gibt es keine allgemeine Regel. Projektarbeit benötigt oft Treffen im Rahmen der vom Projekt vorgegebenen Zeitlinien.

**iX: Man nutzt quasi die Chance, wenn alle an einem Fleck sind?**

**DM:** Genau, auch um Reisekosten, Ressourcen und persönlichen Aufwand zu optimieren. (jvo@ix.de)

