

Wie man Open Source ohne Katerstimmung einführt und Entwickler in Verzückung bringt

Chefs von Devs #95, 4. Juni 2026

Liebe Chefinnen, liebe Chefs,

arbeiten eure Entwickler unter Linux? Genau dieses Betriebssystem wünschen sich laut Nicolas Christener die meisten Entwickler als Arbeitsumgebung. Der CEO von Adfinis muss es wissen: Das in Bern beheimatete Unternehmen optimiert Entwicklerworkflows. Aber auch bei seinen übrigen Dienstleistungen setzt das Team um Nicolas Christener überraschend konsequent auf Open Source.

Solche Produkte benötigen Unternehmen und Behörden, die sich von ausländischen Clouds lossagen und ihre Betriebsgeheimnisse wieder in die eigenen vier Wände zurückholen wollen. Oder müssen: Immer strengere rechtliche Anforderungen erzwingen sogar zunehmend eine digitale Souveränität. Viele Organisationen scheuen allerdings den Wechsel zu quelloffenen und liberal lizenzierten Lösungen. Zu groß ist die Angst, liebgewonnene Software und Workflows zu verlieren.

Diese Furcht nimmt Nicolas Christener in unserem Interview. Darin empfiehlt der CEO eine schrittweise Einführung von Open-Source-Lösungen, gibt Tipps aus seiner Praxis und verrät, wie man Entwicklungsprozesse effizienter gestaltet – unter anderem, indem man seinen Angestellten Linux-Clients spendiert.

Schritt für Schritt zu Open Source

Nicolas Christener entdeckte Ende der 1990er Jahre Linux und wechselte 2006 zur damals noch jungen [Adfinis](#). Das Unternehmen setzt bereits seit seiner Gründung vor über 25 Jahren vollständig auf Open-Source-Produkte. Zunächst als Beratungsunternehmen für Linux-Systeme gestartet, bietet Adfinis mittlerweile komplette Cloud- und Sicherheitslösungen an. Die Referenzen lesen sich dabei beeindruckend.

So haben Nicolas Christener und seine Mitarbeiter beim Pharma-Konzern Roche in der Entwicklungsabteilung Linux-Clients eingeführt, bei den Schweizerischen Bundesbahnen (SBB) ein Secrets Management umgesetzt und bei der Verwaltung des Kantons Uri ein

digitales Baubewilligungsverfahren implementiert – selbstverständlich immer mit Open-Source-Software.

Kannst du kurz deine Aufgaben bei Adfinis beschreiben?

Seit 2017 bin ich CEO bei Adfinis. Dabei leite ich einerseits den Betrieb in unserem Mutterhaus in der Schweiz und unterstütze andererseits unsere ausländischen Niederlassungen bei ihrer Entwicklung. Als CEO bin ich für die Umsetzung der Strategie zuständig. Ich sehe mich aber in erster Linie als Teil eines Teams, in dem ich dabei helfe, Kundenwünsche, Mitarbeiterbedürfnisse und die langfristige Strategie der Eigentümer für alle gewinnbringend umzusetzen und dabei einen positiven Einfluss auf das Open-Source-Ökosystem auszuüben.

Und wie bist du zu Open Source und Adfinis gekommen?

Ende der 1990er Jahre, während meiner Berufsausbildung zum Informatiker, musste ich mich mit den Tücken von Windows Me herumschlagen. Irgendwo zwischen Bluescreen, Virenbefall und fehlendem Compiler fragte ich mich, ob es keine besseren Alternativen gibt. Ein Freund zeigte mir dann Suse Linux und von da an ging es schnell: Ich stellte meinen Windows-Desktop und meinen Heimserver auf Linux um, gründete mit Freunden eine Linux-User-Group und stellte meine Arbeitsinhalte von Visual Basic auf Linux-Systems-Engineering und LAMP-Entwicklung um.

Kurz darauf wurde ich via IRC auf eine Stelle bei Adfinis aufmerksam – der Reiz, bei einem Arbeitgeber angestellt zu sein, der vollumfänglich auf Open Source setzt, klang sehr reizvoll. Ich hatte Glück, wurde eingestellt und arbeite auch 20 Jahre später immer noch bei Adfinis – ich nutze nach wie vor einen Linux-Desktop und habe mein Interesse an der Technik noch nicht verloren.

Warum setzt ihr so massiv auf Open-Source-Software? Und warum sollte ich in meinem Unternehmen überhaupt Open-Source-Komponenten verwenden?

Für mich zählen viele der Gründe, die mich damals an Open Source faszinierten, auch heute noch zu den relevanten Argumenten. Während ich mein damaliges System nur unzureichend selbst debuggen und erweitern konnte, weil es kaum Dokumentationen zur inneren Funktionsweise von Windows gab und ich mir keinen Compiler leisten konnte, hatte ich im Open-Source-Ökosystem alles zur Hand. Der Code war offen und transparent zugänglich und dank der Fülle an Entwickler-Tools konnte ich meiner Kreativität freien Lauf lassen.

Diese Einsicht ist auch heute noch zentral innerhalb der Diskussion, warum Open Source wichtig ist. Offene Lösungen können auditiert und verstanden werden – die gebotene Transparenz ist in einer Welt, in der digitale Dienste immer wichtiger werden, von unschätzbarem Wert. Wer möchte schon seine Stimme einer Evoting-Maschine geben, wenn unklar ist, was genau damit passiert? Nur transparente und offene Systeme können das nötige Vertrauen für solch kritische Systeme schaffen. Natürlich sind auch die Wahlfreiheit der Komponenten und der Lieferanten große Vorteile, denn dadurch entsteht Wettbewerb, ein Gleichgewicht der Machtverhältnisse und Partnerschaften auf Augenhöhe. Wer Wahlfreiheit, Transparenz und Partnerschaften statt Abhängigkeiten sucht, sollte sich im Open-Source-Ökosystem umschauen.

Zu guter Letzt möchte ich die Innovationsgeschwindigkeit hervorheben. Durch Open Source sind wir Teil einer weltweiten Community, die dazu beiträgt, dass die Lösungen täglich verbessert werden. Es ist nicht nötig, auf einen spezifischen Hersteller zu warten, bis ein Feature auf die Roadmap kommt – die Roadmap wird durch die Community-Mitglieder definiert. Wir beobachten bei vielen Projekten eine sehr hohe Innovationsgeschwindigkeit.

Für immer mehr Unternehmen ist die digitale Souveränität ein wichtiges Thema, viele wollen ihre sensiblen Daten wieder aus den großen Clouds zurückholen. Euch müssten doch eigentlich die Türen eingernannt werden?

Wir beobachten zwar ein großes Interesse, stellen jedoch fest, dass aktuell noch wenige umfassende Entscheidungen für konkrete Migrationen getroffen werden. Viele

Unternehmen beobachten erst einmal, was andere machen und halten vorerst an den lieb gewonnenen Tools sowie dem guten Gefühl fest, zu wissen, wie diese funktionieren. Einen Wechsel weg von Office, Photoshop, Slack und Co. sehen viele Entscheider als kompliziert und als Experiment mit ungewissem Ausgang an.

Doch genau hier liegt die Chance: Diejenigen, die jetzt den ersten Schritt machen – wie wir es in Deutschland bereits bei wegweisenden Projekten wie Opendesk sehen –, sichern sich heute das Know-how, das in drei Jahren, wenn der Handlungsdruck durch regulatorische oder geopolitische Anforderungen zu hoch wird, extrem teuer und rar sein wird. Gleichzeitig wächst das allgemeine Unbehagen spürbar und die Risiken der Abhängigkeit werden kritischer beleuchtet, weshalb wir eine verstärkte Nachfrage nach Strategien für einen geordneten Ausstieg sehen.

Auch wenn bei der Migration der eigenen Infrastruktur auf digital souveräne Lösungen aktuell noch zu wenig passiert, bewegt sich beispielsweise Deutschland bereits relativ schnell und hat erkannt, dass die eigene Wirtschaft hervorragende Alternativen anbietet. Der Aufbau souveräner IT-Services wird heute durch eine breite Auswahl zuverlässiger Open-Source-Projekte erheblich erleichtert.

Ob Chat-Dienste, Dokumenten-Kollaborations- oder Planungstools: Etablierte Lösungen ermöglichen einen sicheren, unabhängigen und kontinuierlichen Betrieb für unzählige Organisationen. Mit jeder Ablösung eines bestehenden Services ergibt sich die Chance, eine Entscheidung für mehr Unabhängigkeit zu treffen, indem auf Open-Source-Software gesetzt wird. Auch die Bereitschaft des öffentlichen Sektors, in die eigene Wirtschaft zu investieren, nimmt zu, und angesichts der geopolitischen Verschiebungen ist es aus meiner Sicht nur eine Frage der Zeit, bis wir deutlich mehr reale Migrationen sehen werden.

Tun sich eurer Erfahrung nach Behörden schwerer mit einem Umstieg und warum?

Meiner Meinung nach tun sich Behörden mit einem Umstieg nicht per se schwerer als Firmen. Durch die öffentlichen Beschaffungen haben Behörden jedoch oft längere Zyklen. Während ein Unternehmen von heute auf morgen eine Entscheidung zugunsten einer neuen Lösung treffen kann, hat die öffentliche Hand meistens Verträge über fünf bis sieben Jahre abgeschlossen. Neben diesen beschaffungsrechtlichen Realitäten sehe ich den Kostendruck und die dünne Personaldecke als die größten Herausforderungen für Behörden, einen Umstieg zu vollziehen.

Wir sehen jedoch, dass sich wichtige Parameter ändern. So hat beispielsweise die Schweiz mit dem EMBAG (Bundesgesetz über den Einsatz elektronischer Mittel zur Erfüllung von Behördenaufgaben) ein Gesetz erlassen, das staatliche Organisationen dazu zwingt, Softwareentwicklungen, die mit Steuergeldern finanziert wurden, als Open Source zu veröffentlichen. Auch Deutschland fördert mit Plattformen wie Opencode oder Lösungen wie Opendesk die Nutzung und Veröffentlichung von offenem Code. Auch in anderen europäischen Ländern gibt es solche Bestrebungen, beispielsweise DINUM und SILL in Frankreich oder CAD in Italien. Es tut sich also so einiges.

Zudem beobachten wir, dass immer mehr öffentliche Verwaltungen eigene Communities bilden und Code veröffentlichen. In der Schweiz sind wir beispielsweise bei Inosca stark involviert, einer Community von Kantonen, die die Digitalisierung der Baugesuche als Open-Source-Lösung vorantreibt. In Deutschland sind wir stark bei Opencode und Opendesk involviert. Meine Einschätzung ist, dass Behörden schon heute sehr viel mehr Open Source einsetzen als allgemein bekannt ist.

Aus eurer Erfahrung: Was sind die größten Stolperfallen beim Umstieg auf Open Source? Überschätzen beispielsweise Unternehmen die Möglichkeiten von Open Source?

Es gibt definitiv Stolperfallen. Es zeigt sich, dass eine enge Unterstützung durch erfahrene Personen der beste Wegbegleiter ist, um erfolgreich ans Ziel zu kommen.

Wir müssen beispielsweise früh in Projekten erklären, dass Open-Source-Lösungen kein 1:1-Ersatz und keinesfalls kostenfrei sind. Uns ist wichtig, die Geschäftsmodelle der Hersteller zu unterstützen und die Lösungen nachhaltig zu finanzieren. Daher koppeln wir viele unserer Leistungen daran, dass Organisationen Subscriptions beim Hersteller

beziehen. So wird aus einem vermeintlich kostenlosen Produkt ohne Unterstützung ein Produkt, bei dem man auch auf Support-Leistungen zählen darf. Leider gibt es immer noch viele Organisationen, die mit dem Gefühl an eine Migration herangehen, dass die Software kostenlos ist und die Community den nötigen Support bietet. Gelingt es uns, diese Einstellung zu korrigieren, ist oft schon ein großer Schritt getan.

Ein weiterer wichtiger Punkt ist die Akzeptanz der Benutzer. Schulungen und eine schrittweise Einführung sind der Schlüssel zum Erfolg. Auf der letzten Meile müssen dann noch fehlende Funktionen identifiziert und entsprechende Entwicklungsaufträge erteilt werden, damit diese implementiert werden. Wenn man in offenen und ehrlichen Gesprächen den Boden dafür richtig vorbereitet, stehen die Chancen gut, dass man am Schluss ein erfolgreiches Projekt hat, bei dem die Lösung um einige Funktionen verbessert wurde, die der nächste Kunde dann nicht mehr bemängeln wird. So nähern sich die OSS-Lösungen Schritt für Schritt einem sehr hohen Qualitätsstandard und die Stolperfallen werden immer kleiner.

Häufig herrscht eine große Skepsis unter den Mitarbeitern gegenüber einem Umstieg auf Open Source, weil die gewohnten Programme und Workflows wegfallen. Wie kann ich die Mitarbeiter in meinem Unternehmen mitnehmen?

Uns erscheint es wichtig, Projekte nie mit einer Big-Bang-Umstellung umzusetzen. Wir verfolgen einen schrittweisen Ansatz. Zunächst können die Tools ersetzt werden, die wenig Diskussionen auslösen, und die Personen migriert werden, die offener für einen Wechsel sind und ihren Kollegen später auch bei Fragen helfen können.

Wir haben bei Kunden auch schon Diskussionsforen implementiert und betreut, in denen Mitarbeiter Fragen stellen konnten. Es gab dann eine interne Community, die diesen Personen geholfen hat. Die dabei entstehende Dynamik ist sehr spannend und für viele Mitarbeiter bereichernd. Auch IT ist ein People-Business – wenn wir uns nicht die Zeit für die Menschen und ihre Probleme nehmen und die fördern, die anderen helfen, werden wir keine Kultur schaffen, in der ein Wechsel Positives bewirkt.

Wichtig ist auch, die Dateiformate früh umzustellen und im Parallelbetrieb zu prüfen, wo es zu Kompatibilitätsproblemen kommt. So können frühzeitig Support-Cases eröffnet und der Hersteller im Rahmen der Subscription-Leistungen gebeten werden, diese Probleme zu lösen. Solchen Migrationen sollte man durchaus Zeit geben und den Mindset in der Firma wachsen lassen. Eine Umstellung sollte als Marathon und nicht als Sprint betrachtet werden. Viele proprietäre Produkte sind in Organisationen schon zehn, fünfzehn oder gar zwanzig Jahre im Einsatz – da sollte man keine Erwartungen haben, die innerhalb von drei Monaten erfüllt werden können. Hilfreich ist, dass heute viele Prozesse primär im Browser laufen, so dass der Unterschied des Betriebssystems und Browsers heute weniger ins Gewicht fällt als noch vor ein paar Jahren.



Nicolas Christener (Bild: Privat)

Kannst du ein paar Standard-Open-Source-Komponenten empfehlen, die sich in der Praxis bewährt haben? Ihr setzt ja beispielsweise auf Linux-Systeme von Suse und Red Hat sowie Gitlab, Openbao und die Hashicorp-Produkte. Warum fiel die Wahl auf genau diese Komponenten?

Wir haben uns vor fast zehn Jahren für Mattermost als unsere Chat-Lösung entschieden; diesen Entscheid haben wir nie bereut. Dabei haben sich verschiedene Tools, die wir auch direkt in Mattermost integrieren, immer wieder geändert und neue Integrationen sind dazugekommen. Am Beispiel von Mattermost zeigt sich, wie offene Lösungen die nötige Flexibilität bieten, immer wieder Anpassungen an neue Bedürfnisse vornehmen zu

können.

Ein anderes Beispiel ist Univention UCS, das wir seit rund 20 Jahren im Einsatz haben. Dank der neu dazugekommenen SSO-Funktionen und den Möglichkeiten, auch mit proprietären Plattformen interagieren zu können, hat UCS bei uns alle technischen Schritte und das Organisationswachstum sehr gut begleitet.

Auch Gitlab setzen wir intensiv ein. Sowohl die technischen Teams als auch alle Management-Levels setzen Gitlab für die Planung und Umsetzung ein. Viele der internen Prozesse knüpfen an Gitlab an, und auch mit unseren Kunden arbeiten wir intensiv mit Gitlab.

Last but not least setzt der größte Teil unserer Mitarbeitenden Fedora und Gnome auf dem Arbeitsplatz-Rechner ein. Auch damit sind wir sehr zufrieden.

Auf der Kundenseite haben sich insbesondere Ansible und Opentofu für Infrastructure as Code durchgesetzt. Dieses Duo ist für uns unschlagbar gut. Für Container-Workloads setzen wir seit sehr vielen Jahren stark auf Kubernetes, Argo CD und Prometheus. Mit Openbao arbeiten wir zudem auch als Co-Maintainer an einer transparenten, einfachen und zuverlässigen Lösung für Secrets Management – auch damit sind wir bei den Kunden sehr erfolgreich.

Diese Beispiele zeigen für mich, dass man in Organisationen sowohl auf der Enduser-Seite mit Open Source problemlos ans Ziel kommt und dass gerade im technischen Bereich ein großes Potenzial für den Einsatz von offenen Lösungen vorhanden ist.

Wie schwierig ist die Einbindung von Open-Source-Software und Linux-Systemen in die bestehende Infrastruktur? Kannst du hier Tipps für die Integration geben? Was sind die typischen Stolperfallen in heterogenen Umgebungen?

In Organisationen mit einem hohen Anteil proprietärer Lösungen ist ein schrittweises Vorgehen essenziell, denn letztlich ist die Integration kein rein technisches Problem, sondern ein Management-Problem. Wer alles auf einmal will, scheitert – wer hingegen modular denkt, gewinnt.

Um die Sicherheit zu haben, dass der laufende Betrieb nicht gefährdet wird, empfehlen wir beispielsweise, zunächst lediglich die Dateiformate von OOXML auf ODF umzustellen. Man sammelt so über einige Monate wertvolle Erfahrungen und kann gezielt Interoperabilitätsprobleme mit spezialisierten Partnern wie Collabora lösen, bevor die eigentliche Softwareumstellung erfolgt.

Im nächsten Schritt muss geprüft werden, wie Daten über offene Schnittstellen exportiert werden können, wobei ein Parallelbetrieb mit kontinuierlicher Synchronisierung sicherstellt, dass alle Daten sauber auf der neuen Plattform ankommen. So lassen sich etwa E-Mails, Kalender und Adressbücher schrittweise in eine offene Groupware wie Open-Xchange übertragen.

Auch der Wechsel einer IAM-Lösung über eine Anbindung von UCS an bestehende Domains oder die Migration von Datenbanken zu Postgres kann stufenweise erfolgen.

Wenn es einen universell gültigen Tipp gibt, dann diesen: Man muss das Problem in Teilprobleme stückeln und schrittweise vorgehen, um die Disruption kleinzuhalten.

Jede Organisation kann den Grad proprietärer Lösungen so substanziell verringern; 80 Prozent sind oft ohne großes Risiko möglich, und selbst wenn am Ende 20 Prozent Restbestand bleiben, ist der Gewinn an Unabhängigkeit und die eigene Position bei künftigen Verhandlungen bereits um ein Vielfaches besser.

Ihr stellt Entwicklerteams maßgeschneiderte Open-Source-Tools zur Verfügung. Welche sind das vor allem?

Die letzten Jahre waren vor allem geprägt von einer hohen Nachfrage nach Kubernetes-Plattformen und dem Bedürfnis, Code schneller von der Idee in die Produktion zu bringen. Wir haben daher viele Projekte umgesetzt, in denen CI/CD-Pipelines – vor allem Gitlab mit Kubernetes – kombiniert wurden.

Das stetige Wachstum an neuen Systemen wurde mit Ansible und Terraform (heute vor allem OpenTofu) gemeistert. Ebenfalls kamen Keycloak und UCS oft zum Einsatz, um die große Menge an neuen Tools mit einer zentralen Benutzerverwaltung und SSO handeln zu können.

Um die Anforderungen an Security und Compliance zu adressieren, sahen wir auch ein starkes Interesse an IBM Vault und Openbao.

Was vielfach unter dem Radar passiert, ist die Einführung von Linux Desktop Clients für Entwickler, Ingenieure und Wissenschaftler – da durften wir zum Beispiel Roche begleiten und einen international verfügbaren Linux-Desktop für genau diese Benutzergruppe entwickeln.

Ihr optimiert auch Entwickler-Workflows. Was sind denn eurer Erfahrung nach typische Flaschenhälse, was „bremst“ denn derzeit am meisten? Und wie lassen sich diese Bremsen lösen?

Auch heute sehen wir noch immer viele Betriebe mit einem Silo-Ansatz. Ein Team – oder einzelne Personen – kümmert sich um das Netzwerk, ein anderes um Storage, ein drittes um Computing und ein weiteres um Datenbanken etc. Die fehlende Automatisierung führt zu fehlendem Self-Service und die Entwickler finden sich im Ticket-Ping-Pong wieder.

In Organisationen mit einem hohen Grad an Automatisierung und individueller Autonomie sind Tools wie Gitlab so mit den Infrastruktur-Komponenten integriert, dass Ideen viel schneller ausprobiert und umgesetzt werden können.

Optimale Workflows sind dabei nicht eine Frage des Geldes, sondern eine Frage des Willens, bestehende Silos und Denkmuster aufzubrechen. Offene Lösungen mit gut dokumentierten Schnittstellen helfen auch hier zu schnelleren Lösungen. Linux mit Ansible bietet hier klare Vorteile gegenüber anderen Systemen – das müssten sich einige Systemadministratoren eingestehen und entsprechend den Weg freimachen, so dass Teams Linux einsetzen und von der Flexibilität profitieren können.

Wenn man dann auch noch Lösungen mit komplexen Lizenzmodellen eliminiert und so die "Architecture by License"-Sünde loswird, hat man die relevanten technischen Flaschenhälse eliminiert. Wir sehen nämlich oft, dass Organisationen Lösungen so bauen, wie die Lizenz es erlaubt, und nicht so, wie es für das Business am besten wäre. Open Source befreit die Architektur von diesen Fesseln.

Als übergreifender Faktor bleibt für mich die Investition in die Skills der Mitarbeiter ein zentraler Punkt in Bezug auf das Ausschöpfen des Potenzials respektive das Eliminieren von Bremsen. Hier sehe ich auch bei uns noch Schwächen; wir gehen zu oft davon aus, dass Mitarbeiter sich selbst weiterbilden. Wenn man als Organisation aber nicht die nötigen Freiräume schafft, passiert die Entwicklung unstrukturiert und oft zur falschen Zeit. Das Resultat ist, dass Mitarbeiter dann, wenn die Komplexität steigt, nicht das nötige Fachwissen haben und Flaschenhälse entstehen.

Kannst du anhand von ein, zwei Beispielen aus deiner Praxis beschreiben, mit welchen konkreten Maßnahmen und (Software-)Komponenten ihr die Produktivität in einem Projekt steigern konntet?

Das größte Produktivitätswachstum haben wir in den vergangenen Jahren wohl durch den Fokus auf Gitlab Pipelines erzielt. Wir haben CI/CD Pipelines, Openbao und Funktionen eines Self-Service-Portals so kombiniert, dass möglichst viele manuelle Schritte eliminiert werden. Gerade mit der Integration von automatisierten Tests und der Möglichkeit, Secrets in den Pipelines direkt zu nutzen, konnten wir Personen, die in spezifischen Bereichen weniger Expertise haben, so befähigen, dass auch sie schnell und effektiv Änderungen an einem Projekt einbringen können.

Dazu kommt die Eliminierung von Silos. Fast alle Repositories sind intern für alle Mitarbeiter zugänglich – so findet Kollaboration über Team-Grenzen hinweg statt. Gitlab funktioniert als zentrale Schnittstelle für fast alle Bereiche. Sei es Infrastructure-as-Code-Anpassungen, Task-Management, Code-Entwicklung, Pipeline Runs, Prozess-Management etc. Wir sehen diese transparente Herangehensweise, in der vieles für alle

in einem Tool zugänglich ist, als unseren Inner-Source-Weg. Also die konsequente Umsetzung der Open-Source-Prinzipien in den eigenen vier Wänden.

Durch den klaren Fokus auf Kollaboration und Zugang zu Informationen der anderen Teams haben wir bis heute mehr Produktivität gewonnen, als durch den Einsatz von AI-Lösungen.

Entwickler fordern explizit Linux-Clients – ist das tatsächlich eure Erfahrung? Warum bekommen sie die in vielen Unternehmen nicht? Und lässt sich damit alleine schon die Produktivität steigern?

Da unsere eigenen Mitarbeiter primär Linux nutzen, ist ein Produktivitätsvergleich für uns schwierig. Bei unseren Kunden beobachten wir jedoch eine zunehmende Nutzung von WSL sowie den Wunsch der Entwickler nach Linux.

Linux wird oft als System gesehen, das einem nicht im Weg steht, klare Fehlermeldungen ausgibt, wodurch das Debugging einfacher ist, und das sich sehr einfach automatisieren lässt. Entwickler haben mit Linux oft alles zur Hand, was sie benötigen. Compiler, Debugger, Datenbanken, Runtimes, Browser, Container, VMs etc. – alles ist mit einem simplen Kommando installiert und nutzbar.

Andere Systeme haben in diesem Bereich immer noch Defizite, wobei WSL, Powershell und Winget bei Windows eine gewisse Verbesserung gebracht haben. Den jahrelangen Rückstand an Entwicklerfreundlichkeit wird aber auch Microsoft nicht mit einem Silver Bullet lösen können.

Dass viele Unternehmen ungern eine zweite Flotte an Betriebssystemen verwalten, ist verständlich. Gerade wenn alle Compliance-Anforderungen für Windows bereits umgesetzt wurden, ist der Reiz, den ganzen Aufwand noch für ein zweites System zu betreiben, verständlicherweise gering.

Windows hat jedoch an vielen Stellen seine einstige Vormachtstellung verloren: Bei Server-Systemen sinkt der Marktanteil kontinuierlich, bei mobilen Geräten ist Windows kaum vertreten und selbst die letzten Bastionen wie Gaming oder Office geraten durch Proton und neue Marktteilnehmer wie Google zunehmend unter Druck. Unternehmen sollten anerkennen, dass die Dominanz von Windows vorbei ist und wir in einer IT-Welt leben, in der die Endgeräte wieder viel heterogener sind als noch vor zehn Jahren.

Da es schwierig ist, Entwickler zu finden, sollte man bei dieser Personengruppe nicht aus Bequemlichkeit auf Windows pochen, sondern den ersten Schritt machen und Linux einführen. Es ist gut möglich, dass dann weitere Personengruppen in der Firma dazukommen und der Windows-Anteil plötzlich viel kleiner ist, als man gedacht hätte.

Bietet Open Source sogar einen Wettbewerbsvorteil? Etwa, weil sie talentierte Entwickler anzieht?

Der wohl größte Wettbewerbsvorteil ist die Reduzierung des Vendor-Lock-ins. Durch den konsequenten Einsatz offener Technologien sind wir heute nur noch von sehr wenigen

Herstellern abhängig. Wir können unsere Bezugsquellen und Produkte selbst wählen und haben bei Preisverhandlungen oft mehrere Optionen zur Hand. Diese Freiheit ist ein sehr hoher Wettbewerbsvorteil, der sich auch monetär niederschlägt.

Bei unseren eigenen Mitarbeitern sehen wir klar, dass das Open-Source-Mindset der Adfinis ein wichtiger Grund ist, warum sie uns als Arbeitgeber gewählt haben. Viele Mitarbeiter haben auch bereits Erfahrung in der Zusammenarbeit mit Open-Source-Projekten, daher ist es für sie oft einfacher, Probleme zu analysieren und zu lösen. Diese Erfahrungen haben sie oft während der Ausbildung, dem Studium oder in ihrer Freizeit gesammelt und darauf basierend suchen sich viele Talente einen Arbeitgeber mit einer klaren Open-Source-Strategie. Für uns ist der Fachkräftemangel daher wohl weniger ein Problem als für andere Firmen.

Es zeigt sich, dass die Interaktion mit der Community ein wichtiger Erfolgsfaktor von Open Source ist. Durch den kollaborativen Mindset kommt man in der Open-Source-Szene viel einfacher an Informationen und Hilfestellungen – das hilft auch bei Kundenprojekten; die Mitarbeiter sind es gewohnt, Infos zu teilen, einander zu helfen und wissen, wo sie suchen müssen.

Auch LLMs profitieren vom offenen Zugang zu Dokumentationen, Diskussionen und Code und liefern sehr gute Antworten bei Problemstellungen zu Open-Source-Lösungen. Auch dadurch entstehen aktuell Wettbewerbsvorteile, weil man mit der Unterstützung von LLMs mit Open-Source-Lösungen noch schneller ans Ziel kommt.

Adfinis legt einen großen Fokus auf die Sicherheit. Wird die eurer Erfahrung nach in vielen Unternehmen im Bereich der Entwicklung vernachlässigt oder gibt es eine hohe Sensibilität für das Thema?

Pauschale Aussagen sind schwierig, doch wir stellen fest, dass die Compliance-Anforderungen und rechtliche Rahmenbedingungen wie CRA dazu führen, dass Unternehmen das Thema Sicherheit heute generell viel höher gewichten. So gibt es heute beispielsweise Ausbildungen zum Thema CISO, auf Konferenzen wird über CRA referiert und in Verträgen werden Verbindlichkeiten gefordert. Zudem werden Lösungen im Bereich statische Code-Analysen, Secrets-Management oder Pentesting vermehrt nachgefragt.

Auch Open-Source-Lösungen nutzen Googles Angebote wie OSS-Fuzz etc. Die Sensibilisierung hat stattgefunden und viele Bereiche sind heute besser gegen Cyber-Risiken gewappnet als noch vor wenigen Jahren. Basierend auf meinen Beobachtungen würde ich behaupten, dass sich Entwickler generell viele Gedanken zur Sicherheit ihres Codes machen und auch verstanden haben, dass das Thema ein integraler Bestandteil des Entwicklungsprozesses sein muss. Betrachtet man jedoch, wie hoch das Thema Cybersicherheit außerhalb der technischen Teams priorisiert wird, stelle ich fest, dass der Handlungsbedarf noch lange nicht ausgeschöpft ist. Der Faktor Mensch bleibt in allen Sicherheitsbetrachtungen ein Risiko.

Ihr arbeitet aktiv mit Open-Source-Projekten zusammen. Welche sind das?

Seit dem Lizenzwechsel von Hashicorp sind wir bei Openbao sehr aktiv. Wir wollen unseren Kunden auch weiterhin eine offene Secrets-Management-Lösung bieten und investieren daher erheblich in die Entwicklung von Openbao. Wir glauben fest daran, dass Secrets-Management für alle offen zugänglich und langfristig nachhaltig sein muss.

Ebenfalls sehr aktiv sind wir im Bereich der Form- und Workflow-Lösung Caluma. Auch hier steht für uns im Zentrum, eine Community zu schaffen und insbesondere Projekte im öffentlichen Sektor umzusetzen, die dem Prinzip "Public Money? Public Code!" entsprechen. Mit der Inosca-Community haben wir mittlerweile sechs Schweizer Kantone zusammengebracht, die im Bereich eBau (Baubewilligungsprozess) eine Open-Source-Lösung auf Basis von Caluma vorantreiben.

Auch im Bereich Ember.js entwickeln wir verschiedene Module und unterstützen das Projekt finanziell. Wir haben beispielsweise auch das Projekt "Linux man pages" finanziell unterstützt, treten als Sponsor für Konferenzen auf, hosten Server für die Document Foundation und Openstreetmap oder unterstützen das Debian LTS-Projekt.

Zudem sind wir in diversen Non-Profit-Organisationen wie zum Beispiel CH Open,

Parldigi, OSBA, DOSBA, FSFE oder APELL aktiv oder unterstützen diese.

Wir sind also auf vielen Ebenen aktiv, um das Open-Source-Ökosystem zu stärken und es langfristig weiterzuentwickeln.

Was sind denn die Vorteile dieser Zusammenarbeit? Würdest du anderen Organisationen und Unternehmen ebenfalls die Zusammenarbeit mit Foss-Projekten empfehlen?

Wir beobachten, dass unsere Mitarbeiter durch OSS-Contributions eine hohe Identifikation mit unserem Unternehmenszweck entwickeln. Sie erkennen, dass die Weiterentwicklung öffentlicher, digitaler Güter eine sinnstiftende Wirkung hat. Das wiederum erhöht die Verbundenheit mit dem Unternehmen und macht unsere Mitarbeiter glücklich. Durch die Mitarbeit an entsprechenden Open-Source-Projekten generieren wir bei den Kunden Vertrauen, dass wir die nötige Expertise haben, um sie bei komplexen Fragen auf Expertenniveau zu beraten.

Durch die Kollaboration erweitern wir zudem unser Netzwerk und gewinnen auch immer wieder Neukunden. Klar, wir enablen zu einem gewissen Grad auch unsere Konkurrenz aber das Fachwissen steckt oft nicht im Code, sondern im Verständnis für den Use-Case des Kunden. Für mich ist klar, dass diese Zusammenarbeit auf vielen Ebenen gewinnbringend ist und es daher definitiv Sinn ergibt, dieses Vorgehen zu kopieren.

Wie sieht diese Zusammenarbeit in eurem Fall genau aus? Steuert ihr beispielsweise auch Quellcode bei?

Ja, wir steuern schon länger auch Upstream Code zu verschiedenen Projekten bei, intensivieren aber unsere Bemühungen aktuell, weil wir auch unser eigenes Portfolio an Lösungen entwickeln und so unabhängiger von anderen Herstellern werden wollen. Wir bezahlen in gewissen Projekten Core-Contributor, zum Beispiel bei Openbao, wo wir auch den Vorsitz im Technical Steering Committee zur Verfügung stellen, und lassen Mitarbeiter generell ohne große Hürden an Upstream-Projekten contributen.

In Bereichen, in denen wir nicht selber über die nötige Expertise verfügen, bezahlen wir auch regelmäßig andere Firmen für die Entwicklung von Funktionen; wir haben zum Beispiel die DeepL- und die Languagetool-Integration in Libreoffice finanziert und zusammen mit einem Kunden sowie Collabora die Portierung von Libreoffice für das iPad umgesetzt.

Wie reagiert die Community darauf, dass sich ein kommerzielles Unternehmen einbringen möchte? Muss man etwas bei der Zusammenarbeit beachten?

Wir erleben die Zusammenarbeit generell als wertschätzend. Unser Ziel ist es, einen Beitrag zu leisten und das Projekt zu unterstützen, nicht unser eigenes Ding durchzuziehen. Ein Fork ist nicht unser Ziel, sondern die Kollaboration mit dem bestehenden Projekt. Daher setzen wir uns mit den Gepflogenheiten auseinander und bringen uns so ein, dass wir die Grundsätze des Projekts nicht verletzen.

Ich habe beispielsweise selbst ein paar einfache Commits an Libreoffice getätigt und davor längere Zeit die Mailinglisten und Commit-Messages gelesen, um zu verstehen, wie das Projekt funktioniert. Meine Erfahrung ist, dass Mühe und guter Wille in vielen Projekten erkannt und geschätzt werden – egal, ob man als Privatperson oder von einer Firmenadresse einen Commit einreicht.

Hast du abschließend ein paar Tipps und Ratschläge für Unternehmen und Organisationen, die mit Open Source liebäugeln? Worauf müssen sich die Firmen und Mitarbeiter einstellen?

Wie bei vielen anderen Dingen auch, empfehle ich, in kleinen Schritten zu starten. Das gilt sowohl für den Einsatz von Open-Source-Lösungen als auch für das Veröffentlichen von Code. Übung macht bekanntlich den Meister und am Anfang passieren immer Fehler. Man wählt die falsche Lizenz, hat noch nicht die Ressourcen, um eine Community richtig zu managen, oder muss sich auf User einstellen, die nur fordern, aber nichts beitragen. Für all diese Dinge muss man Strategien entwickeln und schlicht daran wachsen. Das

funktioniert am besten, wenn man sich nicht zu viel auf einmal vornimmt.

Beim Einsatz von Open-Source-Lösungen ist es genau dasselbe. Man sollte darauf achten, dass der Betrieb normal weiterläuft und Schritt für Schritt kleinere Migrationen durchführen. Und last but not least sollte man den kollaborativen Geist der Community nutzen. Viele Personen, Projekte und Firmen helfen tatsächlich sehr gerne – man sollte sich trauen, auch einfach mal nach Hilfe zu fragen.

Nicolas, vielen Dank für das Gespräch und die vielen guten Tipps!

Lasst von euch hören

Falls ihr Fragen oder Anregungen zum Newsletter habt – chefsvondevs@golem.de ist die richtige Adresse dafür. Die nächste Ausgabe kommt in zwei Wochen.

Bis dahin, alles Gute,
Tim Schürmann

Mach dich noch schlauer!

Golem unterstützt dich dabei als unkomplizierter und verlässlicher Partner mit viel Leidenschaft. Wir versprechen: sachlich und präzise recherchierte Fakten, verantwortungsbewusste Selektion von Informationen, klare Orientierung, eine starke Haltung und immer ein offenes Ohr für dich.

Impressum:
Golem Media GmbH
Wilhelmine-Gemberg-Weg 5-7
10179 Berlin