



Szene-Trends nachgefragt – zum Thema: informationelle Selbstbestimmung

Open Source, Webdienste und die vielen Gesichter des Datenschutzes

Trends wie *Bring-your-own-device*, die Verschiebung des Fokus von Desktop- auf Webanwendungen, aber auch die Technologisierung der Entwicklung durch *Continuous Integration and Delivery*, *DevOps*, *SOA* und *Big Data* beeinflussen zunehmend das Arbeitsumfeld und den Instrumentenkasten von Softwareentwicklern. Was das für den Einsatz von Technologien, die Agilität und den Datenschutz im Entwickler-Alltag bedeutet, darüber hat Szene-Redakteurin Annegret Handel-Kempff (AHK) mit Johannes Jacop, Geschäftsführer des Softwarehauses und Eclipse-Mitgliedunternehmens Yatta Solutions GmbH, gesprochen.



Johannes Jacop

ist Geschäftsführer und Mitgründer der Yatta Solutions GmbH. Vor seiner Tätigkeit für Yatta war er in München in einer Wirtschaftskanzlei

tätig und absolvierte sein Studium an der Universität Bayreuth. Für Yatta ist Johannes Jacop vor allem für Geschäftsstrategie und Investor Relations, aber auch für die Compliance zuständig. Er engagiert sich für die Gründerszene in Deutschland und setzt sich für die aktive Mitgestaltung der Digitalisierung ein.

AHK: Eclipse wird immer komplexer. Der Webdienst „Yatta Profiles for Eclipse“ soll die Arbeit von Softwareentwicklern effizienter machen. Wie hat man sich das vorzustellen?

Johannes Jacop: Die Welt der Softwareentwicklung wird immer komplexer. Selbst die Bandbreite an Programmiersprachen wächst wieder. Wer hätte das zu Beginn des Siegeszugs von Java gedacht?

Um dieser Komplexität Herr zu werden, setzen Unternehmen zunehmend Webdienste ein – auch für die Softwareentwicklung. Etwa um die Zusammenarbeit in Teams zu verbessern. Beispiele hierfür sind Kollaborationsplattformen wie Jira, Online-Repositories wie GitHub und Online-Informationssysteme wie Stack Overflow.

Als Tool-Entwickler müssen wir zum einen permanent an der Integration unserer Werkzeuge arbeiten, zum anderen den Zugang von Nutzern zu Eclipse und Eclipse-basierten Tools wie unseren erleichtern. Das ist für den Erfolg von

Eclipse aus meiner Sicht entscheidend – und dafür nutzen wir auch die Cloud.

Als offene Entwicklungsumgebung oder Open-Source-Plattform liegt der Fokus von Eclipse eben nicht auf einer bestimmten Anwendung, zum Beispiel als Java IDE, sondern auf verschiedenen Eclipse-Projekten. Auf der einen Seite ist Eclipse deshalb großartig, weil die IDE für so gut wie jede Aufgabe in der Softwareentwicklung Lösungen bereithält – und dazu noch Open Source ist.

Auf der anderen Seite müssen Softwareentwickler bei Tausenden Plug-ins und mehr als 250 Open-Source-Projekten erst herausfinden und entscheiden, was sie nutzen möchten, um ihr Softwareprojekt zu verwirklichen. Ein Beispiel: Das beliebte Eclipse IDE Package for Java EE Developers besteht technisch aus über 900 Plug-ins aus zwanzig bis dreißig verschiedenen Open-Source-Projekten. Natürlich stimmen sich die Projekte und insbesondere die Project Leads zum Release untereinander ab – und gerade die sogenannten Package Maintainer leisten Unglaubliches. Sie koordinieren als Open-Source-Entwickler die Releases der Packages – und das oft ehrenamtlich. Trotzdem bleibt der Aufwand für die Endnutzer der Eclipse IDE erheblich. So etwa für Softwareentwickler im Enterprise-Umfeld. Sei es nur, dass jeder einzelne Entwickler eines Projekts in Eclipse zunächst die Sourcecode- und Task-Repositories einrichten, Projekte importieren und diverse Einstellungen der IDE – wie zum Beispiel die Code Formatter – anpassen muss. Ohne diesen Konfigurationsaufwand kommen Sie in einem Projekt nicht weit. Und das Ganze kostet eben viel Zeit und Nerven. Umso mehr, je größer die Entwickler-Teams sind. Eclipse kann sein Setup und die richtige Konfiguration nicht einfach eraten.

Mit Profiles kann ein einzelner Entwickler oder Projektleiter ein fertig eingerichtetes Eclipse an beliebig viele Entwickler weiter verteilen. Die Installation funktioniert mit wenigen Klicks. Updates von Plug-ins oder der Umstieg auf eine neuere Eclipse-Version führen häufig dazu, dass jeder im Team die komplette IDE neu aufsetzen muss. Profiles beinhaltet deshalb einen Update-Mechanismus: Ein Entwickler nimmt die notwendigen Anpassungen vor und verteilt dann das Update mit einem Klick. Teams werden einfach schneller produktiv.

Open-Source-Projekte profitieren ebenfalls, weil sie für die Mitarbeit ein fertig eingerichtetes SDK zur Verfügung stellen können. Das senkt Einstiegshürden für potenzielle Contributor oder Committer.

Welche Zukunft erwartet Eclipse in einer Welt von rasant wachsenden Massen an Big Data, die auf Strukturierung warten?

Mit Blick auf klassisches Big Data hat die Apache Foundation mit Projekten wie Hadoop, Hive, Cassandra und anderen hier die Nase vorn. Anders sieht es im IoT-Umfeld (Internet of Things) aus, wo mit Bosch einer der größten Player diverse Open-Source-Projekte initiiert und bei Eclipse veröffentlicht hat. Auch hier geht es darum, wie gigantische Datenmengen aus der Sensorik erfasst und verarbeitet werden.

Die Eclipse Foundation bemüht sich in Abstimmung mit dem Board of Directors und strategischen Mitgliedsunternehmen wie Oracle, IBM, Google & Co. dediziert darum, mehr Informationen über Eclipse-Nutzer zu gewinnen, um damit auch eine bessere Beziehung zu ihnen aufzubauen.

Hierzu gehören Initiativen wie das Speichern und Verteilen von Nutzerdaten auf eclipse.org, aber auch Funktionen wie das automatisierte Error Reporting, bei dem letztlich viele relevante Daten erhoben werden. Ziel ist es, eine verstärkt datengetriebene Steuerung der Eclipse-Projekte zu ermöglichen. Nutznießer sind also vor allem die strategischen Eclipse-Mitgliedsunternehmen, die mehr Informationen über ihre Nutzer erhalten. Langfristig sollen durch die gewonnenen Erkenntnisse die Eclipse-Projekte auch erfolgreicher werden.

Die Eclipse-Mitgliedsunternehmen, -Committer und -Contributor arbeiten ihrerseits vor allem an zwei Zukunftsfeldern: Bei dem einen Feld geht es darum, Technologien, wie Frameworks und Tools, für die Erhebung, Verarbeitung und potenziell auch gemeinsame Nutzung großer Datenmengen zu entwickeln und bereitzustellen. Bosch hatte ich ja hier für das IoT schon als Beispiel genannt, die zusammen mit Red Hat das neu aufgesetzte Eclipse-Projekt Hono leiten. Andere Beispiele sind Eclipse SmartHome und Eclipse Kura, aber auch Tools wie DataStax DevCenter oder die Hadoop Development Tools für Eclipse. *Letztlich wird Eclipse hier zur Plattform für die gemeinsame Entwicklung von standardisierten Big-Data-Lösungen.*

Ein anderes Feld ist die Erhebung und Verarbeitung von Daten aus der Entwicklung selbst. Das Error Reporting vermittelt durch die Menge an Fehlerdaten den Eclipse-Entwicklern neue Erkenntnisse. In eine ähnliche Richtung gehen auch automatisierte, web-gestützte Codeanalysen. Hier steht die Verbesserung der Softwaretechnik durch Big Data im Vordergrund.

Innerhalb von Eclipse wird intensiv daran gearbeitet, die IDE selbst ins

Web zu bringen. Das eröffnet neue Nutzungsmöglichkeiten – aber mit Blick auf Datenschutz und Datensicherheit auch neue Herausforderungen.

Mit welchen datenschutzrechtlichen Themen haben Sie sich vor Einführung von Yatta Profiles beschäftigt?

Um auch für uns selbst vollständig zu erfassen, welche Daten überhaupt durch unsere Software erhoben, verarbeitet, übermittelt und genutzt werden – und sei es auch nur potenziell –, haben wir einen strukturierten Prozess umgesetzt, bis zu den einzelnen Entwicklern. Unsere Privacy Policy sichert Datenhoheit und -autonomie unserer Nutzer und macht die – potenzielle – Nutzung personenbezogener Daten bei Profiles, aber auch bei unseren anderen Webdiensten, transparent.

Ein Beispiel: Ein Entwickler nutzt unseren Support, indem er sich persönlich registriert und einen generierten Error Report an uns versendet. Intern stellen wir sicher, dass wir diese Information, die ja nur für den Support gedacht war, weder für Werbung noch für andere Zwecke verwenden.

Dass man vorhandenes Wissen in Form von Informationen und Daten nicht nutzt, weil es eben nur für bestimmte Zwecke freigegeben ist – wie zum Beispiel den Support, aber nicht für Werbung – ist für Ingenieure freilich auch ein Lernprozess. Das ist auch eine Kulturfrage.

Eine Schwäche des deutschen Bundesdatenschutzgesetzes ist es indes, dass die technischen Realitäten dort schlicht nicht abgebildet sind. So stellt das Datenschutzgesetz zwar abstrakte zahlreiche Anforderungen, bietet aber wenig Orientierungshilfe für eine datenschutzfreundliche Gestaltung. Das Ziel des Gesetzes, allein möglichst wenig Daten, also Wissen, zu erheben oder zu nutzen, hilft einem Softwareentwickler einfach nicht weiter. Dabei gibt es hier auch von versierten Experten konkrete Reformvorschläge – zum Beispiel durch das Gutachten von Prof. Dr. Roßnagel und anderen im Auftrag des Bundesinnenministeriums. Nur leider mahlen die Mühlen des Gesetzgebers sehr langsam: Das Gutachten stammt aus dem Jahr 2001. Damals gab es Facebook, GitHub und Atlassian noch nicht, dafür aber noch den Netscape Navigator.

Welche Konsequenzen haben Sie aus Ihren Erfahrungen gezogen?

Tatsächlich haben wir drei wesentliche Konsequenzen daraus gezogen: Erstens haben wir nach unserer Analyse noch einmal neu evaluiert, ob die Software die Er-

hebung, Verarbeitung oder Übermittlung personenbezogener Daten wirklich in dem Umfang braucht – und auch wofür. Zum Teil haben wir nach reiflicher Überlegung auch auf Funktionen verzichtet. Beispielsweise kann man den Profiles Launcher für Eclipse auch lokal als reine Desktopanwendung verwenden; in dem Fall werden keine Setup-Informationen übermittelt. Auch verlangen wir für die Nutzung des Launchers oder Installation eines Profils keine Registrierung, sondern setzen sie erst voraus, sobald man ein Setup Dritten zur Verfügung stellen möchte. Das heißt, wir haben das Produktdesign angepasst und uns dem Gebot der Datensparsamkeit verpflichtet.

Zweitens haben wir erkannt, dass wir mit unseren Kunden und Nutzern über das Thema Datenschutz und Datensicherheit in Dialog treten müssen. Was ist ein guter Datenschutz? Wie möchten wir diesen Schutz umsetzen? Ein Projektleiter hat oft ganz andere Vorstellungen davon, was er über die Einrichtung und Nutzung der IDE seines Teams wissen möchte, als die betroffenen Entwickler. Beispielsweise scheint es sinnvoll, den Projektleiter darüber zu informieren, welche Plug-ins im Team verwendet werden. Andererseits möchten wir aber keine Rückschlüsse darüber erlauben, zu welcher Uhrzeit oder wie intensiv die IDE verwendet wird. Deshalb werden solche Daten gar nicht erst erhoben. Klar ist auch, dass es eine gute Abwägung zwischen dem technisch Möglichen und den Datenschutzinteressen der Nutzer zu finden gilt. Es gibt da keine absoluten Antworten.

Drittens mussten wir diese Balance zwischen Offenheit und Datenschutz auch mit Dritten einstellen. Ursprünglich war geplant, Profiles for Eclipse als Webdienst auf eclipse.org zu hosten. Das ließ sich allerdings nicht realisieren, weil die Eclipse Foundation unter US-Recht operiert und Management und Server in Nordamerika sind. Da gelten einfach andere Regeln: Die Foundation hat keine Privacy Policy, die dem deutschen Datenschutzstandard gerecht wird. In der vorhandenen Erklärung bleibt völlig unklar, welche personenbezogenen Daten von wem, wann und wie erhoben werden – zum Beispiel im Zusammenhang mit den Eclipse-Nutzerkonten –, wie diese Daten genutzt und an wen sie übermittelt werden. Der Datenschutz der Profiles-Nutzer hätte sich auf dieser Basis nicht mehr sicherstellen lassen. Für diese Situation haben wir leider auch weiterhin keine Lösung, deshalb hosten wir den Webdienst selbst – und zwar in Deutschland – oder auf Wunsch direkt bei den Enterprise-Kunden.

Damit versuchen wir die Privatsphäre und die Datenhoheit unserer Nutzer zu sichern, sowie auch Compliance und Arbeitnehmerdatenschutz für den Einsatz von Eclipse in Unternehmen und Behörden zu gewährleisten.

„Politik und Diplomatie versagen beim grenzüberschreitenden Datenverkehr“

Die „Safe-Harbour“-Klausel zum Umgang mit personenbezogenen Daten ist Geschichte. Was bedeutet das Urteil des Europäischen Gerichtshofs vom 6. Oktober 2015, demzufolge Datenschutzbeauftragte die Einhaltung des Europäischen Grundrechts auf den Schutz persönlicher Daten zusätzlich prüfen dürfen, für Ihr kostenfreies Eclipse-Produkt und für diejenigen, die damit arbeiten?

Der Webdienst zu Profiles, der sogenannte eHub, wird bis auf Weiteres in Deutschland gehostet. Wir stellen den Datenschutz, insbesondere die Compliance, den Arbeitnehmerdatenschutz und den Schutz der informationellen Selbstbestimmung unserer Nutzer nach unseren Richtlinien und auf Grundlage des deutschen Bundesdatenschutzgesetzes sicher. Für Firmen und Open-Source-Entwickler, mit denen wir zusammenarbeiten, entwickeln wir deshalb auch eine entsprechende Verpflichtungserklärung.

Trotzdem sind viele unserer Kunden selbst natürlich auch auf den grenzüberschreitenden Datenverkehr angewiesen: international operierende Unternehmen auf die Nutzung von Rechenzentren, die schon aus Gründen der Redundanz nicht nur in Deutschland und Europa stehen, weltweit verteilte Entwicklungsteams. Das ist gerade im Open-Source-Umfeld doch vielfach Realität. Da braucht es eine vernünftige Lösung der Europäischen und US-amerikanischen Gesetzgeber. Ich empfinde es als Zumutung, dass Politik und Diplomatie so lange versagen, bis am Ende Gerichte entscheiden müssen. Das muss besser gehen.

Bald wird kein größeres Unternehmen mehr ohne Software-Ingenieure auskommen. – Wird nach Ihrer Einschätzung die Freiheit beim Entwickeln von Software künftig stärker durch Datenschutz oder durch Sicherheitsanforderungen begrenzt?

Durch beides. Verfügbarkeit und Sicherheit von Daten stehen in einem Spannungsverhältnis. Beides muss gut austariert werden – und man darf das eine dem anderen nicht opfern.

Text: Annegret Handel-Kempff (AHK)

Foto: privat