



Das Ende der Finsternis

Markus Karg, Java User Group Pforzheim

Nachdem die Anwenderschaft, neudeutsch „Community“, monatelang lautstark an der offensichtlichen Untätigkeit von Oracle herumgemeckert hat, ging es plötzlich ganz schnell: Java EE wurde an die Eclipse Foundation übergeben. Doch was bedeutet das im Detail? Ist Java EE nun tot und die JCP aufgelöst? Ein bewusst provokativer Kommentar aus Anwendersicht.

Plötzlich war es da: das lange geforderte Öffnen von Test Compatibility Kits (TCK), Reference Implementations (RI) und Specifications (Specs). Was auch immer der letztendliche Auslöser war, sei es die monatelange Nörgelei oder einfach nur der Wunsch, sich einer Altlast zu entledigen, der Community soll es recht sein: Java EE ist nun endlich und vollständig Open Source. Oder doch nicht? Naja, es kommt darauf an, wen man fragt! Oracle sieht das so, die Community nicht unbedingt.

Trau, schau wem ...

Die Eclipse Foundation gibt sich zumindest schon mal demonstrativ euphorisch: Mike Milinkovic, Executive Directory eben jenes Industrie-Clubs, gab jüngst in der Keynote seiner Hausmesse EclipseCon Europe stolz die Parole aus, seine Organisation sei nun „der Hort der Innovation“, da neben IBM OpenJ9 nun auch EE4J (also das bisherige

Java EE) dort weiterentwickelt werde. Aber werden sie das denn wirklich? Fakt ist: Bisläng tut sich rein gar nichts, sieht man vom allgegenwärtigen Auf-den-Busch-Klopfen einmal ab.

Zumindest tut sich nicht mehr als vorher. So schnell wird sich das auch nicht ändern. Auf die Frage, wo Oracle EE4J in einem Jahr stünde, will sich Oracle kaum mehr abringen lassen als ein „Aller Code ist von der Rechtsabteilung geprüft, auf die Server der Eclipse Foundation kopiert und besteht als das Java EE 8 TCK.“ Moment mal ... Java 8? Wurde denn nicht behauptet, durch den Move zur Eclipse Foundation gehe dank der breiten Community-Beteiligung nun alles viel flotter? Schade. Also ein weiteres Jahr Stillstand.

Dafür kann sich die Community endlich beteiligen. Zum Beispiel mit

Contributions. Ach, halt, nein, doch nicht. Denn erstens sind die Committer nach einem bislang nicht veröffentlichten Schlüssel handselektiert und die meisten stammen weiterhin von Oracle. Zweitens, während des Moves können keine Contributions von Dritten angenommen werden, wie auch, sonst kommt die Rechtsabteilung beim Code-Review ins Trudeln. Drittens, was „später“ (also nach besagtem Wartejahr) überhaupt von Dritten angenommen wird, entscheiden Personen, die noch gar nicht festgelegt sind, also gibt es dazu bis auf Weiteres keine Aussage – der Prozess ist noch gar nicht definiert. Das muss nichts Negatives bedeuten, kann es aber.

Apropos gewählt. Deren Wahl wird, wen hätte es gewundert, keineswegs demokratisch erfolgen (wie etwa beim Oracle-gelenkten JCP), sondern die Funktionsträger bei EE4J werden nach Gutdünken durch das Project Management Committee (PMC) bestimmt, ganz im Open-Source-Sinne eines „Benevolent Dictator“, der nun mal aller Benevolenz zum Trotz eben doch nur ein Diktator ist. Ist es das, was die Community wollte: einen anderen Diktator? Und übrigens, klar, das PMC (initial besetzt durch Oracle, IBM, Red Hat, Payara, Tomitribe und Ivar Grimstad) wird natürlich auch nach Gutdünken von der Eclipse Foundation dominiert. Dort bestimmt, man mag es ja kaum sagen, ausschließlich die Industrie: Ein Stimmrecht ist an die Zahlung so erheblicher Euro-Summen gekoppelt, dass sich wohl kein Anwender erdreisten wird, ein Stimmrecht kaufen zu wollen – sei es ein ambitionierter Freelancer, ein JUG-Mitglied oder ein wissenschaftlicher Mitarbeiter einer Hochschule. Im Schwabenland würden wir sagen: „Das hat ein Gschmäcke“ – außer es ist eben Open Source, da wird

so etwas dann plötzlich als normal angesehen. In der Summe ist kein echter Vorteil gegenüber dem bisherigen Stewart auszumachen, im Gegenteil, die JCP war zumindest demokratisch und angesichts der kürzlichen Oracle-Wahlschlappe zum Thema „Jigsaw“ bewiesenermaßen auch nicht zahlos. Also alles nix, oder?

Und es ward Licht

Naja, ganz so schwarz malen sollte man es dann doch nicht. Letztendlich hat die Eclipse Foundation in den vergangenen Jahren hervorragende Arbeit geleistet. Mit der kostenlosen und offenen Eclipse-IDE hat die Stiftung Meilensteine gesetzt und liefert eine der nach wie vor beliebtesten Werkzeuge der Branche. Auch das Lizenzmodell ist sehr liberal. Im Unterschied zum bisherigen Status quo verleiht sich die Eclipse Foundation nämlich nicht das Intellectual Property (IP) jeder Contribution ein, sondern belässt diese (ganz in Einklang mit dem deutschen Urheberrecht) beim Autor. Sie erhält lediglich (ebenfalls im Einklang mit dem deutschen Urheberrecht) ein nicht-exklusives, unbeschränktes und unterlizenzierbares Verwertungsrecht.

Das ist wesentlich besser als die bisherige Übertragung des Urheberrechts an Oracle. Darüber hinaus soll es, so der Plan, grundsätzlich jedem möglich und sehr viel einfacher werden, durch Zeichnen eines Vertrags mit der Eclipse Foundation, gefolgt vom Stellen von Pull Requests auf GitHub, Contributor zu werden. Wenn das so kommt, ist auch dies für die Community ein Vorteil, da man bislang Papier und Fax benötigte und nicht alle Projekte Pull Requests überhaupt akzeptiert hatten. Wer viele Pull Requests stellt, kann dann



```
if (
    Du.skills.contains ( "Java" ) &&
    Du.location.matches (
        "(Berlin | Hamburg | München | Köln | Frankfurt |
            Düsseldorf | Dortmund | Leipzig | Bonn | .+ )" )
    ) {
    Du.checkout ( "https://www.nextlevel.de/java-jobs" );
}
```



auf Ermessen der Projektleitung zum Committer erhoben werden. Das wäre ein echtes Novum bei einigen Projekten sowie ein großer Schritt in Richtung Community-Einbindung und würde sicherlich viele Verbesserungen in den Code bringen, die bislang an der abschottenden Oracle-Politik scheiterten.

Die Hoffnung stirbt zuletzt

Insofern könnte der Move zur Eclipse Foundation nun wirklich die leidigen Themen „Community-Beteiligung“ und „Innovationsgeschwindigkeit“ voranbringen – wenn das Wörtchen „wenn“ nicht wäre. Denn auf Nachfrage Ende Oktober konnten Eclipse Foundation, PMC und auch Oracle nur wenig Licht in ein paar wichtige Fragen bringen. Doch diese sind es, die letztendlich über den Erfolg des Unterfangens entscheiden. Denn je nach Antwort bleibt sonst von der Aktion kaum mehr übrig als der Tausch des Titels, des Besitzers und des Regelwerks – wie beispielsweise bei EclipseLink, das als Oracle TopLink startete, später als EclipseLink eine Reinkarnation unter Eclipse-Leitung erlebte, um dann ebenso in Untätigkeit vor sich hin zu dümpeln wie seine Java-EE-Geschwisterprojekte, die den Schritt in Richtung Eclipse gerade erst vollziehen. Die gestellten Fragen und ihre Antworten im Einzelnen stehen im Kasten.

Fazit

Abwarten und Tee trinken! Erst wenn Oracle, PMC und Eclipse Foundation ihre Hausaufgaben komplett gemacht haben, ist eine finale Bewertung möglich. Hoffen wir, dass die in Aussicht gestellte Teilhabe der Community in der besagten Form kommen wird und dass sie schneller kommt, als angesagt. Denn nichts wäre schlimmer für Java EE als weitere Jahre der Untätigkeit – egal unter welcher Flagge. Für mich als ISV-Entwicklungsleiter, der auf Standards und Produktunabhängigkeit angewiesen ist, sowie als Expert-Group-Mitglied (JSR 370) ist die aktuelle Situation fatal: Ich bin weiterhin in meiner Arbeit gehemmt, und zwar für mindestens ein weiteres Jahr. Was danach kommt, steht in den Sternen. Das kann gut oder schlecht sein. Ich muss akzeptieren, wie es kommt, und bereite mich auf zwei Szenarien vor: Die Sache geht schief, dann werden wir die Idee eines Industriestandards beerdigen können und wieder zu produktabhängigen APIs wechseln müssen; oder die Sache klappt im Gegensatz zu anderen Oracle-Open-Source-Aktionen der letzten Jahre – das glaube ich allerdings einfach (noch) nicht. Mal sehen ...



Markus Karg

markus@headcrashing.eu

Markus Karg ist Entwicklungsleiter eines mittelständischen Softwarehauses sowie Autor, Konferenzsprecher und Consultant. JAX-RS hat der Sprecher der Java User Group Goldstadt von Anfang an mitgestaltet, zunächst als freier Contributor, seit JAX-RS 2.0 als Mitglied der Expert Groups JSR 339 und JSR 370.

Wer standardisiert zukünftig Java EE?

Die Eclipse Foundation ist keine Standardisierungsorganisation. Das kann möglicherweise weiter der JCP bleiben, es gibt jedoch keine spruchreife Entscheidung. Es ist noch nicht einmal klar, wer das überhaupt zu entscheiden hat. Der JCP arbeitet bislang wie gewohnt weiter und hält die üblichen Wahlen ab, zeigt sich aber in seinen Meetings entsprechend verunsichert. Man könnte den JCP auch einfach reformieren, daran scheint Oracle allerdings kein Interesse zu haben. Die Eclipse Foundation möchte im Prinzip einfach nur Code produzieren und zeigt sich etwas überrascht davon, dass die Community so stark an Standards interessiert ist.

Wie und wo sollen die bisherigen JSR Expert Groups weiter an Standards arbeiten?

Oracle findet es am sinnvollsten, wenn die Expert Groups einfach weitermachen wie bisher, was natürlich nicht funktioniert, da dies voraussetzen würde, dass Oracle nun aktiv in allen JSRs mitarbeitet – das war exakt das Hauptproblem der vergangenen Monate und ist es noch immer.

Wie können sich User Groups (wie iJUG oder JUGs) beteiligen?

Jeder Contributor muss einen Vertrag mit der Eclipse Foundation (also nicht mehr mit Oracle) unterzeichnen (Contributor Agreement). Der Arbeitgeber muss der Mitarbeit zustimmen, wie auch schon beim JCP. Tut er das nicht, schaut man vom Spielfeldrand aus zu. Contributions sind per GitHub als Pull Requests zu stellen. Das ist einerseits sehr willkommen für Code, geht jedoch nicht bei Diskussionen um Spezifikationen: Dieses Modell richtet sich rein auf die Programmierung von Produkten, nicht auf das Finden von Hersteller-übergreifenden Industriestandards.

Wie ist die künftige Beziehung OpenJDK/EE4J?

OpenJDK (also Java SE 9) und Java EE 8 teilen sich derzeit einige Packages. Diese sind in Java 9 bereits als „Deprecated for Removal“ markiert und werden aus Java SE verschwinden. Sehr wahrscheinlich werden sie durch EE4J übernommen – aber eben nur wahrscheinlich. Solange Oracle die Kontrolle über Java SE und Java EE hatte, war das kein Problem. Doch was passiert, wenn Oracle die Packages streicht, und die Eclipse Foundation sie nicht weiter pflegt?

Wie werden die „javax.*“-Packages zukünftig genannt?

Sie werden nicht umbenannt, auch nicht bei neuen Versionen. Nur ganz neue, bislang nicht existierende APIs werden neue Packages mit anderem Präfix erhalten.

Wird es bald Maintenance-Releases geben?

Auf keinen Fall im Laufe des kommenden Jahres, denn Oracle ist voll mit dem Move beschäftigt und kann daher viele neue Ideen erstmal weder selbst umsetzen noch als externe Contributions annehmen.

Bleiben zukünftige Major-Releases von EE4J abwärtskompatibel?

Das erste EE4J-Release wird abwärtskompatibel zu Java EE 8 sein. Bei zukünftigen Releases behält sich das PMC vor, das WORA-Prinzip (Write Once, Run Anywhere) zu brechen, da man einige unliebsame Technologien wie Common Object Request Broker Architecture (CORBA) nicht mehr für notwendig erachtet. Hier gibt es starken Widerstand in der Community, die Bestandscode weiterbetreiben können muss, ohne auf Produktupdates der Runtime beziehungsweise des Application-Servers zu verzichten. Hier zeigt sich, wie Wunsch und Wirklichkeit auseinanderdriften: Die Eclipse Foundation interessiert nur Innovation; die Community interessiert hingegen, bestehende Anwendungen möglichst lange pflegen zu können – denn ohne WORA braucht man kein Java.