



Endlich ist GlassFish 5.1 da, alle freuen sich und sind aus dem Häuschen – oder doch nicht? Jedenfalls hat die Eclipse Foundation – halt, sagen wir es doch ganz offen – jedenfalls hat Oracle es nun doch endlich mal geschafft, die gleiche Software durch die Tür zu bringen, die sie schon vor über anderthalb Jahren veröffentlicht hat. Juhu! Da kommt doch mal Partylaune auf! Und was sollen wir nun damit? Ja, es ist keine Vaporware – die Software kann tatsächlich downgeloadet werden [1]. Dass das keiner an die große Glocke hängt und man den Link kaum findet, hat einen Grund: Es ist nämlich die falsche. Ja, richtig gelesen: Oracle hat es doch tatsächlich dank des, umschreiben wir es mal vorsichtig mit dem Wort „seltsamen“, Staging-Prozesses geschafft, die falsche Software zu veröffentlichen! „LOL“, wie man das online kommentieren würde! GlassFish 5.1 ist de-facto nicht die Menge all der ganzen Teile, die in monatelanger Kleinarbeit „staged“, geprüft, „released“ und letztendlich auf Maven Central [2] veröffentlicht wurden. Nein, es ist irgendein unbedeutender, veralteter Zwischenstand aus dem vierten Quartal 2018, dessen Download sich nicht lohnt. Wieso das passiert ist, konnte mir bislang niemand sagen, jedoch wurde mir eine Person aus dem Hause Oracle benannt (ich möchte betonen: eine), die wohl beauftragt ist, das zu fixen. Einen Termin konnte man mir nicht nennen (das ist, nach der Erfahrung der vergangenen Monate, wohl auch besser so).

Was tut eigentlich der Rest der Eclipse Foundation? Die EF selbst kämpft noch mit ihrer überlasteten Infrastruktur und streitet sich nach wie vor mit Oracle um die Nutzungsrechte der Namen „Java“, „JAX-RS“, etc. Stand der Dinge ist hier, dass Oracle es wohl nicht erlauben will und alles daher einen neuen Namen erhält. Wir „freuen“ uns schon darauf, dass es nun also bald nicht nur „Jakarta EE“ statt „Java EE“ heißt, sondern auch „Jakarta API for RESTful WebServices“ – denn das Akronym bleibt vermutlich tabu! Dass dies implizit auch bedeutet, dass es keine Java-Standards mehr sind, sondern nur noch vereinsinterne Eclipse-APIs, möchte ich mal ganz am Rande erwähnen. Mag sein, dass dies vielen egal ist, vielen aber eben auch nicht! Neben Oracle sind Red Hat, IBM, Payara und weitere Stakeholder. Zumindest namentlich. Inhaltlich habe ich nicht viel entdecken können, was diese Firmen unter dem Jakarta-Dach an echter Code Contribution geleistet hätten, also von regelmäßigem Marketing und Selbstorganisation abgesehen. Schade eigentlich, wo doch alle unbedingt mehr Einfluss auf Java EE gefordert hätten!

Also alles Mist? Nein, bei Weitem nicht! Sobald Oracle seine Handvoll involvierter Mitarbeiter dank des hoffentlich finalen GlassFish 5.1.1 entlastet hat, können diese endlich den Suspend Mode in den EE4J-Projekten beenden und sich den angestauten Pull Requests [4] widmen. Das heißt ganz konkret, dass dann beispielsweise JAX-RS 2.2 oder Jersey 2.29 endlich wieder Fahrt aufnehmen können und wir uns auf der JavaLand über eine ganze Reihe neuer Features und veröffentlichter Bug Fixes unterhalten können, die eigentlich schon fertig, aber eben nicht integriert sind. Da ist einiges in der Pipe [3], das wirklich Partylaune aufkommen lässt (etwa die Transition von angestaubten Application-Servern auf moderne Architekturen), und darauf lohnt es sich auch noch ein paar Wochen mehr zu warten! Wem es nicht schnell genug geht, der sei auch bei dieser Ausgabe aufgefordert, sich an den zahlreichen Open-Source-Projekten zu beteiligen – jeder ist herzlich willkommen!

Referenzen

- [1] <https://www.eclipse.org/downloads/download.php?file=/glassfish/glassfish-5.1.0.zip>
- [2] https://search.maven.org/search?q=g:jakarta.*
- [3] <https://github.com/eclipse-ee4j/jaxrs-api/wiki/Roadmap>
- [4] <https://github.com/eclipse-ee4j/jersey/pulls>



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.