



De langverwachte release van Jakarta EE

Een overzicht van de release en toekomstplannen

Inmiddels is het zo'n twee jaar geleden dat Oracle afstand deed als eigenaar van het Java EE platform. Vanwege de trage evolutie van dit platform onder de hoede van Oracle, werd besloten om het geheel te doneren aan de open source community. Het doel was om Java EE sneller te laten evolueren met ontwikkelingen op de softwaremarkt. Oracle heeft ervoor gekozen om, op het logo en de merknaam na, alle content, code en rechten over te dragen aan de Eclipse Foundation, kortweg Eclipse. Deze overdracht bleek een hele klus te zijn. Zoveel werk zelfs, dat de migratie van code en content zo'n anderhalf jaar heeft geduurd. Dit was een samenwerking tussen Eclipse en Oracle. Developers en juristen aan beide zijden hebben zo'n 50 projecten gemigreerd.

Parallel aan deze migratie liep het kiezen van een nieuwe merknaam en logo voor het platform, omdat de Java EE merknaam en het logo eigendom blijven van Oracle. Al snel was de merknaam Jakarta EE met de vrolijk gekleurde zeilboot als bijbehorend logo gekozen.

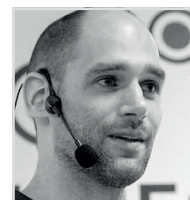
Release van Jakarta EE 8

In het resterende half jaar na de overdracht van de code en content is men binnen de Eclipse community bezig geweest om alle code en content te voorzien van het Jakarta EE merk. Dit heeft ertoe geleid dat op 10 september 2019 een officiële release van het Jakarta EE platform heeft plaatsgevonden. Wat betekent dit? Wat houdt deze initiële release precies in en wat zijn de toekomstplannen?

Daar wil ik je graag een overzicht van geven.

Compatibiliteit met Java EE

Het initiële versienummer Jakarta EE 8 is in lijn met Java EE 8, de laatste versie van het platform onder Oracle. Dit nummer kenmerkt daarmee ook meteen de compatibiliteit van Jakarta EE met zijn voorganger. Dit is een belofte die Eclipse is nagekomen. Op specificatie niveau heeft geen enkele wijziging plaatsgevonden. Het aanpassen van de specificaties is het doel van toekomstige versies van Jakarta EE. Wat bij de migratie naar Eclipse wel veranderd is, zijn de implementaties van de specificaties. Deze zijn opgeschoond om aan Eclipse's kwaliteitseisen en projectstructuur te voldoen. Deze zijn al terug te vinden in GlassFish 5.1. Dit is een applica-



Edwin Derks is Software Architect en CodeSmith bij Ordina JTech en heeft naast Java een passie voor cloud-driven development en serverless architecture.

tieserver ook geërfd van Oracle en nu tevens een Eclipse project. In de nabije toekomst zijn deze implementaties tevens terug te vinden in andere applicatieservers die zich Jakarta EE 8 compatible noemen.

Java EE → EE4J

Dit project (Enterprise Edition for Java) is het technische top-level project binnen Eclipse waaronder ongeveer vijftig projecten vallen die de specificaties of gerelateerde projecten bevatten. Binnen Oracle droeg dit project natuurlijk de naam Java EE die, zoals eerder gezegd, niet overgenomen werd. Jakarta EE is dan ook de merknaam waaronder dit project als een platform 'vermarkt' wordt door Eclipse.

Specificatie- en projectnamen

Voor developers die ervaring hebben met Java EE zijn termen als EJB, CDI, Servlet, JSP en JPA niet meer weg te denken. Toch zullen deze sinds de Jakarta EE 8 release voor sommige specificaties niet meer van toepassing zijn, omdat deze om verschillende redenen van naam veranderd zijn. Soms omdat de naam conflicteerde met rechten van Oracle, soms omdat men er gewoonweg een logischere naam voor verzonnen heeft.

Concreet betekent dit dat projecten binnen Eclipse de naam "Eclipse Project for <<specificatie>>" droegen tijdens de migratie. Omwille van Oracle, dan wel Eclipse zelf, zijn deze projectnamen aangepast naar een nieuw formaat. Een voorbeeld hiervan is het project "Eclipse Project for JAX-RS". Deze specificatie vinden we nu terug in Jakarta EE 8 als "Jakarta RESTful Web Services". Op de Jakarta EE website (referentie 4) kun je een overzicht vinden van alle specificaties die onder Jakarta EE 8 vallen. Hierdoor zijn we bepaalde ingeburgerde namen kwijt, maar er zit wel een positieve kant aan. We hebben nu de basis voor het Jakarta EE platform opgefrist met logischere en meer consistente naamgeving voor de specificaties waar het uit bestaat.

Technology Compatibility Kits

Het is voor veel developers niet bekend dat er voor het certificeren van implementaties zogenaamde Technology Compatibility Kits (TCK) gebruikt worden. Eén van de redenen hiervoor is dat deze voorheen werden beheerd door Oracle en een black box waren

Jakarta EE 8 Deliverables (per Specificatie)

Specificatie Document	Document waarin de details van de specificatie staan. Waar is deze voor en hoe gebruik je hem.
API Release (Binary JAR, JavaDocs en broncode)	De specificatie artifacts die je kunt gebruiken om Jakarta EE applicaties mee te bouwen. Deze zul je terug vinden in o.a. Maven en applicatieservers.
TCK	De TCK die je kunt gebruiken om te verifiëren of een bepaalde implementatie compatibel is met een desbetreffende specificatie.

voor vendors om hun implementaties mee te testen. Gelukkig zijn deze TCKs onder Eclipse als open source projecten opgezet (referentie 5). Daardoor heb je nu zowel als vendor als individu inzicht in de inhoud van deze tests. En kun je deze naar eigen inzicht toepassen. Als een implementatie bewezen compatible is door een TCK, kan deze een betreffende certificering krijgen.

Deliverables

Nu deze wijzigingen achter de rug zijn, konden de deliverables Jakarta EE 8 gedefinieerd en gerealiseerd worden. Deze zijn later uiteraard terug te vinden in applicatieservers die Jakarta EE 8 compliant zijn.

Eclipse GlassFish en IBM OpenLiberty zijn de eerste applicatieservers die Jakarta EE 8 compatible gecertificeerd zijn. Deze certificatie is uitgegeven nadat de TCK tests met de vereiste resultaten waren uitgevoerd op deze applicatieservers. Hiermee zijn Eclipse GlassFish en IBM OpenLiberty helemaal klaar om door te evolueren met Jakarta EE, waar natuurlijk naar alle waarschijnlijkheid de applicatieservers van andere vendors spoedig zullen volgen. De eerste concrete stappen gaan zijn dat Java EE 8 specificaties en implementaties die de applicatieservers nu implementeren, worden vervangen met hun Jakarta EE 8 variant.

Marketing Offensief

Eclipse zet veel in op het gebied van marketing om Jakarta EE onder de aandacht te brengen. Daar is noodzaak voor en Eclipse is zich hier maar al te goed van bewust. Het is geen geheim dat de perceptie bij veel developers leeft dat Java EE zijn beste tijd gehad

DE OVERDRACHT VAN JAVA EE NAAR JAKARTA EE BLEEK EEN HELE KLUS TE ZIJN. ZO-VEEL WERK ZELFS, DAT DE MIGRATIE VAN CODE EN CONTENT ZO'N ANDERHALF JAAR HEEFT GEDUURD

heeft en ondanks de transitie naar Jakarta EE niet kan tippen aan soortgelijke platforms, zoals Spring. Eén van de redenen is de trage evolutie van het Java EE platform onder de hoede van Oracle. Daarnaast heeft Oracle weinig gedaan aan de marketing van Java EE en was dit platform ook niet in beeld bij veel developers wegens het ontbreken van talks hierover op conferenties.

Volop inzetten op marketing voor Jakarta EE is de strategie van Eclipse om dit platform weer onder de aandacht te krijgen, zodat bedrijven en developers enterprise software gaan bouwen.

EE 9 beschikbaar te zijn. Dit is JNoSQL (referentie 6), een specificatie voor het communiceren met NoSQL databases in je Jakarta EE applicatie. Super gaaf!

EFSP

Er is een nieuw proces binnen Eclipse geïntroduceerd voor het adopteren van nieuwe specificaties: het Eclipse Foundation Specification Process (EFSP). De community die JNoSQL bouwt, heeft zich aangedragen als proefkonijn voor het ijken van dat proces. Naar verwachting zullen meerdere nieuwe specificaties worden toegevoegd die hetzelfde proces hebben gevolgd. Er wordt zelfs



Een onderdeel van dit marketingoffensief is de ceremonie die Eclipse had opgezet voor de officiële release van Jakarta EE. Deze was vormgegeven als een online event genaamd JakartaOne (referentie 2).

Toekomstvisie

In de tijd dat alle voorbereidingen voor het releasen van Jakarta EE 8 in volle gang waren, heeft men uiteraard nagedacht over de vervolgstappen. Eclipse wil de evolutie van het platform in gang zetten door het wijzigen, en eventueel zelfs toevoegen, van specificaties. Jakarta EE 9 is de eerstvolgende release van het platform. Wanneer deze versie gereleased wordt, is helaas nog niet bekend. Ook de officiële release cycle van Jakarta EE moet nog bepaald worden. Dit kan gebeuren op basis van een releaseschema van twee jaar of op basis van incrementele releases wanneer nieuwe features of wijzigingen zich aandienen.

Nieuwe specificaties

Voor zover bekend staat er in ieder geval één nieuwe specificatie klaar om toegevoegd te worden aan het platform en in release Jakarta

overwogen of MicroProfile hetzelfde proces kan gaan hanteren voor nieuwe specificaties. Of dit in de toekomst gaat gebeuren, zullen we meemaken.

JESP

Hoewel het EFSP een generiek proces is om specificaties binnen Eclipse te adopteren, heeft Jakarta EE zijn eigen specificatie proces: het Jakarta EE Specification Process (JESP). Dit proces neemt het EFSP als basis en evolueert met eventuele wijzigingen daarop mee. JESP bepaalt o.a. afwijkende perioden om bepaalde beslissingen in het proces te maken. Eén van de eerste deliverables van dit proces is dat JNoSQL als specificatie door Jakarta EE wordt geadopteerd.

Oracle's Afscheidscadeautje

In principe is de weg naar de eerste officiële release van Jakarta EE verlopen volgens het programmaboekje dat Eclipse daarvoor had opgesteld. In principe, want in mei 2019 kwam één eis van Oracle uit onverwachte hoek. Deze luidde dat naast "Java", ook de naam "javax" niet gebruikt mag worden in Jakarta EE. Na



JAKARTA EE

intensief onderhandelen heeft Eclipse het voor elkaar gekregen om nog wel de Java EE 8 compatibele specificaties en implementaties as-is te gebruiken met deze namen. Echter, zodra een wijziging plaats gaat vinden aan een betreffende specificatie of implementatie, dan is de eis dat alle verwijzingen naar “Java” en “javax” uit de betreffende code en documentatie worden verwijderd. Wordt dit niet gedaan, dan zullen de rechten van Oracle geschonden worden en kan dit bedrijf gerechtelijke stappen ondernemen.

Binnen Eclipse en de community is hierover veel ophef ontstaan. Voornamelijk omdat backwards compatibility hoog in het vaandel staat van het Jakarta EE platform. Deze visie staat door deze eis op losse schroeven. Er is door Eclipse en de community veel gediscussieerd over hoe deze situatie op te lossen, met als resultaat twee concrete oplossingsrichtingen. Hier zijn de voor- en nadelen geïdentificeerd en uiteraard is daarin meegenomen wat voor impact beiden hebben op de toekomstplannen van Jakarta EE.

Big Bang

Optie één is om in alle Jakarta EE specificaties en implementaties de referenties naar “Java” en “javax” te verwijderen in één “big bang” release van het platform. Een voordeel van deze aanpak is dat de conflictsituatie hiermee in één stap verleden tijd is. Het grote nadeel is dat Jakarta EE een nieuwe basis krijgt die in zijn geheel niet meer compatibel is met Java EE 8, tenzij het platform zowel de Java EE 8 als Jakarta EE specificaties en implementaties gaat bevatten.

Incremental

Vanzelfsprekend is optie twee om specificaties en implementaties alleen aan te passen

zodra daadwerkelijk wijzigingen plaats zouden vinden. Het grote voordeel hiervan is dat backwards compatibiliteit beter en langer ondersteund zou blijven. Nadelig is dat daarmee bij elke wijziging van het platform gecheckt moet worden of geen rechtenscheidingen plaatsvinden.

Hoe loopt dit af?

De discussie over dit onderwerp wordt nog steeds voortgezet. Er zal in de nabije toekomst een beslissing genomen worden, zodra alle voordelen, nadelen en eventuele alternatieven besproken zijn. Hoe dan ook, daarna is Jakarta EE verlost van referenties naar Java en valt daarmee mooi in lijn met alle andere Java gebaseerde enterprise frameworks.

Community

Nu Jakarta EE gereleased en compleet open source is, is Java EE daadwerkelijk ten gronde gegaan. Het platform kan eindelijk verder evolueren onder een nieuwe eigenaar en community. Wil je hier zelf een bijdrage aan leveren, dan kun je o.a. pull requests aanbieden in Github, talks geven of blogs schrijven (referentie 3). Hierdoor krijgt Jakarta EE de wind in de zeilen waardoor verdere evolutie van Jakarta EE in gang wordt gezet. ■

**HET PLATFORM
KAN EINDELIJK
VERDER
EVOLUEREN
ONDER EEN
NIEUWE
EIGENAAR EN
COMMUNITY**

REFERENTIES

1. <https://jakarta.ee>
2. <https://jakartaone.org/>
3. <https://jakartablogs.ee/>
4. <https://jakarta.ee/specifications/>
5. <https://github.com/eclipse-ee4j/jakartaee-tck>
6. <http://www.jnosql.org/>