

# WS02 voor Oracle professionals

**September 2017**

**Auteur:**

Maurik-Jan Veenman  
INTEGRATIE SPECIALIST



## Inleiding

Steeds vaker zien we bij klanten meerdere opties ontstaan voor componenten in de informatievoorziening stack. Op databasegebied zie ik naast de Oracle database vaak ook Microsoft SQL Server of MySQL ingezet worden. Voor integratie wordt naast de Oracle OSB- en SOA-omgeving regelmatig WSO2 gebruikt. Dit wordt mede ingegeven doordat in overheidsland het credo gehanteerd wordt 'Open Source tenzij...'. Het Oracle-platform, met een aanzienlijke eenmalige investering voor de licenties, wordt dan ingezet voor de echte huzarenstukjes en echte bedrijfskritische functies. Voor de wat meer rechttoe rechtaan projecten wordt WSO2 ingezet waarvoor geen licenties afgenomen hoeven te worden. Ook mijn klant heeft ervoor gekozen om WSO2 in te zetten naast Oracle Service Bus. WSO2 komt nu vaak voorbij, onder andere als aanroepende partij van de Oracle OSB. Ik kijk graag eens bij een ander in de keuken en ben mij daarom in WSO2 gaan verdiepen. Met dit Whitebook wil ik een inleiding geven in de WSO2 wereld. Daarnaast geef ik voor de belangrijkste componenten een "vertaling" van termen uit de Oracle wereld naar de WSO2 wereld om de lezer vlot op gang te helpen in de WSO2 wereld.

## WSO2, wat is het?

De website van WSO2 omschrijft hun oplossing als volgt:

*WSO2 Enterprise Integrator is een krachtige, 100% open source enterprise service bus (ESB). Het wisselt berichten uit en transformeert data tussen legacy systemen, SaaS applicaties, services, and APIs, waarbij de fundamentele mogelijkheden van een geconnecte SOA architectuur worden geboden.*

**Dat belooft heel wat.**



## Wat kan het?

WSO2 kan alle acties die je zou verwachten van een Enterprise Service Bus uitvoeren. WSO2 kan daarbij ook een deel van de functionaliteit die in de SOA suite van Oracle zit. Zonder uitputtend te willen zijn, worden de volgende zaken ondersteund:

- wisselen tussen verschillende protocollen, zoals bijvoorbeeld van HL7 (ziekenhuis protocol) naar FIX (financieel protocol);
- transformeren van berichten;
- centraal opslaan en afdwingen van policies voor authenticatie, autorisatie en de druppelfunctie;
- centrale auditing en monitoring;
- schemavalidatie;
- routeren van berichten;
- legacy systemen als standaard interface representeren;
- business functionaliteiten representeren middels het orkestreren van services (hiervoor heb je bij Oracle ook de SOA omgeving nodig);
- versiebeheer;
- optreden als een betrouwbare berichtenopslag om heraanbieden mogelijk te maken en limieten afdwingen.

## WSO2 versies, installatie en bouwmethoden

### Versies

Er zijn op het moment van schrijven drie versies gangbaar van WSO2. Versie 4.8, 5.0 en 6.0. De eerste heb ik gebruikt omdat je daar nog direct in de source view kunt editen. Dit was nodig om het boek te kunnen volgen (is in 5.0 dichtgezet, en ik weet nog niet hoe je dat weer beschikbaar kunt maken). Bij mijn klant wordt 5.0 gebruikt, de laatste versie in deze 'eenvoudige' vorm. Versie 4.8 en 5.0 ontlopen elkaar nauwelijks. Er is nu ook een nieuwe 6.0 omgeving, die meer wordt gepresenteerd als Enterprise product.

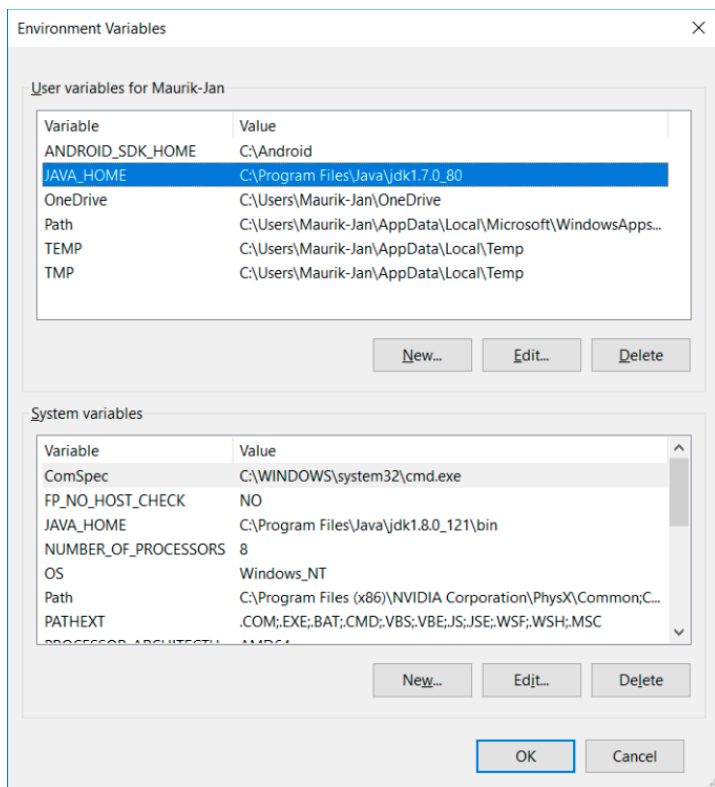
### Installatie

De installatie van WSO2 is erg simpel. Om te beginnen moet je het product downloaden van de [WSO2 website](#). Vervolgens pak je het uit en zet je het in een eigen directory, bijvoorbeeld `c:\wso2esb-4.8.0`

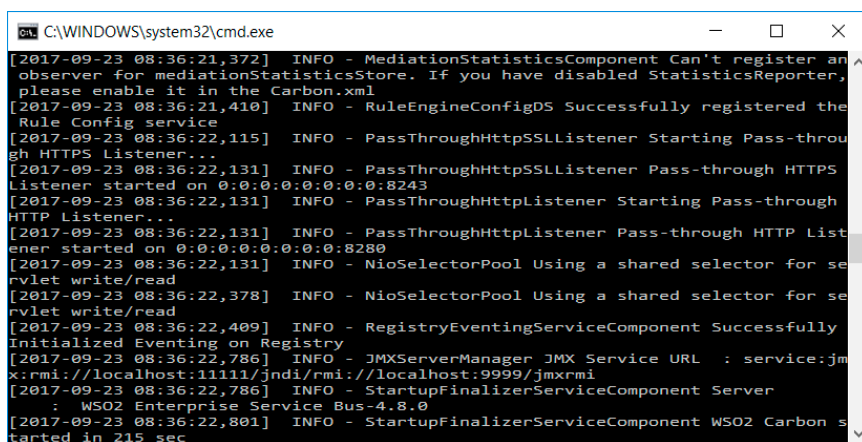




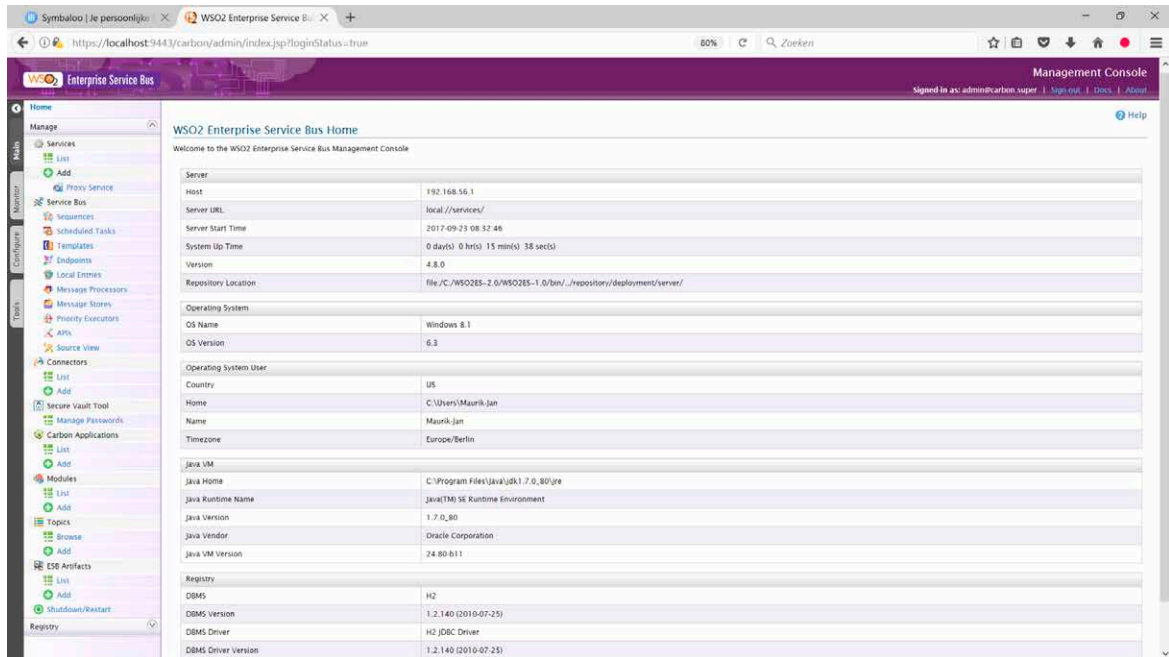
Als je nog geen Java Development Toolkit op je laptop/pc hebt, dan download en installeer je dat (voor WSO2 4.8 heb je JDK 1.7 nodig, 1.8 is te nieuw). Vervolgens stel je in Windows bij advanced system options je pad in naar de directory waar je de JDK hebt geïnstalleerd:



Daarna ga je naar de WSO2 bin directory (in mijn geval C:\wso2esb-4.8.0\wso2esb-4.8.0\bin) en dubbelklik je op wso2server.bat (of wso2server.sh als je op Linux werkt). Er opent dan een command schermpje, en na een tijdje zie je dit:



Dan draait WSO2. Je ziet, het gaat wat eenvoudiger dan een WebLogic installatie  
Als je dan naar deze URL gaat: <https://localhost:9443> en aanlogt met admin/admin, dan opent het management console:

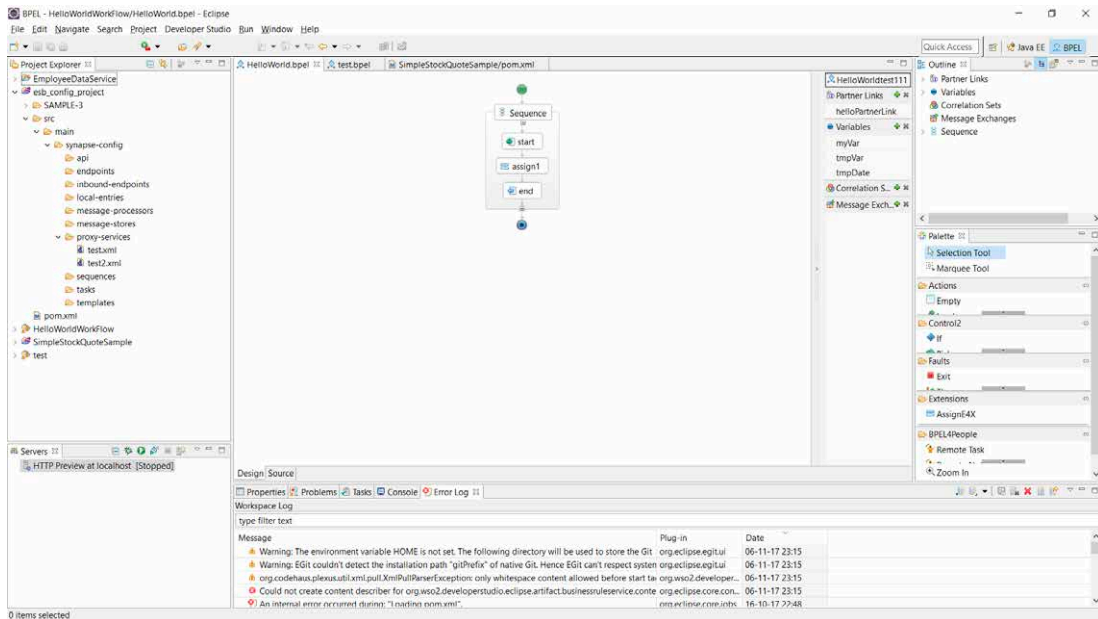


## Bouwen

Je kunt op drie manieren nieuwe interfaces bouwen:

- Je kunt in de WSO2 console bij Home>Main>Services>Add>Proxy Service een nieuwe proxy toevoegen middels een wizard.
- Je kunt in de WSO2 console bij Home>Main>Service Bus>Source View direct in de code een proxy toevoegen (dit lijkt mij wel wat onoverzichtelijk worden als er al vele services zijn, want hierin zit dus alle code die in de service bus draait). Een WSO2 ontwikkelaar bij mijn klant gaf aan dat dit zeer ongebruikelijk is. Het boek wat ik doorgenomen heb gebruikt deze methode om snel te laten zien hoe verschillende oplossingen voor ESB vraagstukken in elkaar zitten, maar in het algemeen wordt onderstaande derde methode gebruikt:
- De derde methode is door gebruik te maken van de WSO2 Developer Studio. Dat is een oude bekende, want dat is Eclipse met een aantal plugins, net zoals wij Eclipse OEPE hebben. Die kun je overigens [hier downloaden](#). In de documentatie las ik dat men de Luna variant aanraadt, een reden wordt niet gegeven. Hieronder een voorbeeldje van hoe een Hello World projectje er uit ziet in deze variant van Eclipse:





Dat ziet er voor Eclipse kenners heel vertrouwd uit. Links zie je de projectstructuur met het configuratieproject, een SRC- (source), main-, en synapse-config component. Daaronder hangen dan de onderdelen van een interface. In het midden zie je het onderhanden component. Wanneer je nieuwe artefacten wilt toevoegen selecteer of sleep je deze uit het palet aan de rechterkant van het scherm, net als je bij de OSB Eclipse gewend bent.

## Hoe heten de elementen in WS02?

Deze alinea beschrijft hoe de belangrijkste bouwcomponenten die wij uit de Oracle OSB wereld kennen genoemd worden in de WS02 wereld.

Je hebt in Developer Studio ook een Workspace, deze heet in dit geval Developer Set. Daarnaast configureer je ook een configuration project en daarin een project voordat je een proxy kunt gaan bouwen.

Een Proxy heet gewoon hetzelfde: Proxy.

Een stage kennen ze niet bij WS02. Bij de OSB wordt het ook eigenlijk alleen gebruikt om proxy componenten te groeperen.

Het kleinste component wat je in een proxy kunt gebruiken heeft bij de OSB steeds verschillende namen: replace of assign, if-then else e.d.. Bij WS02 heet zo'n 'atomair' component altijd 'mediator', die heb je wel in verschillende smaken. Je hebt bijvoorbeeld filter-, send- en switch mediators.





Een Assign component heet bij WSO2 ook Assign.

Een Validate component heet in WSO2 ook Validate.

Een Replace heet in WSO2 Payload Factory Mediator. Deze werkt met xpath expressies. Een If-then-else structuur kan in WSO2 in twee varianten voorkomen, als Switch Mediator (voor uitgaand verkeer), en als Filter Mediator (voor het routeren van terugkomende berichten).

Een Callout actie heet in WSO2 Send Mediator (wordt binnen de Switch Mediator gebruikt). Als je een beslistabel wilt maken voor het ophalen van configuratiewaarden, dan heet dat een Property Mediator.

Een Split Join is bij WSO2 een combinatie van de clone mediator en de Aggregate Mediator. De Clone Mediator verdeelt de order over meerdere parallele processen, en de Aggregate Mediator voegt ze aan het einde weer bij elkaar voor het antwoord.

Als je uit een proces wilt springen gebruik je de Drop Mediator.

Adapters worden bij WSO2 Transports genoemd. Deze types worden standaard ondersteund: HTTP, HTTPS, POP, IMAP, SMTP, JMS, AMQP, FIX, TCP, UDP, FTP, FTPS, SFTP, CIFS, MLLP, and SMS.

Een database bevragen doe je met een DBLookup Mediator, zonder aparte Adapter/Transport dus.


Raise heet bij WSO2 throw, je kunt in WSO2 ook een rethrow doen.

Je hebt ook de exit component, die dient om de flow te verlaten op het moment dat de fout is afgehandeld.

De Event Mediator kan gebruikt worden voor een publish / subscribe configuratie. Als dan het 'event' optreedt dat er een relevant bericht voor een bepaalde subscriber is, dan wordt dat naar die partij gestuurd.

BPEL activiteiten, zoals het orchestreren van services wordt in WSO2 Service chaining genoemd.





Met bovenstaande componenten kun je alles maken wat je in een Oracle Service Bus ook kunt maken. Naast de genoemde componenten bestaan er nog veel meer voor specifieke toepassingen. Hier staan alle beschikbare Mediators beschreven.

### Hulp bij Testen van Services

Voor het testen van Services gebruiken wij vaak een SoapUI omgeving waarin we een mock service maken voor zaken die buiten ons project aangeroepen moeten worden. WSO2 heeft daarvoor zijn eigen 'Simple axis2Server'. Je laadt je eigen proces service in door hem in de subdirectory C:\wso2esb-4.8.0\wso2esb-4.8.0\samples\axis2Server\repository\services te plaatsen (een .aar file, waarvan je voorbeelden kunt vinden op internet). Je start de axis2Service door in directory C:\wso2esb-4.8.0\wso2esb-4.8.0\samples\axis2Server te dubbelklikken op axis2server.bat. Er opent een command window, waarin je na een tijdje ziet dat de service draait. Om te checken of hij benaderbaar is, roep je deze URL aan: <http://localhost:9000/services/<serviceaam>?wsdl> waarbij je <servicenaam> moet vervangen voor de naam van de service die je zojuist hebt ingeladen. Handig, want je bent hierdoor niet afhankelijk van webservices, en je testtool zit al in je omgeving.

## Conclusie

Ik vind WSO2 een gemakkelijk op te zetten en snel werkende Service Bus. Veel zaken zijn al standaard aanwezig (bijvoorbeeld de simple axis server en de adapter configuratie). Er worden zelfs veel van de zaken die we kennen uit de SOA omgeving van Oracle geboden. Het kan wat onoverzichtelijk worden als je heel veel projecten hebt, omdat de projectenlaag mist, maar als je de zaak goed opzet in Developer Studio heb je daar niet zo'n last van. Het is zeker een fijn en uitgebreid product om mee te werken.

## Bronnen

Enterprise Integration with WSO2 e-book van Packt publishing (<https://www.packtpub.com/application-development/enterprise-integration-wso2-esb>)  
WSO2 website (<http://wso2.com/>)

