

# Load Plan pada Oracle Data Integrator 11g

**Yuafanda Kholfi Hartono**

yuafanda@yahoo.com

<http://ilmukomputer.org/author/yofanda/>

## **Lisensi Dokumen:**

Copyright © 2003-2013 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

Load Plan adalah objek baru yang diperkenalkan dalam Oracle Data Integrator (ODI) untuk mengatur eksekusi package dan scenario pada tingkat kerumitan yang kompleks. Load Plan menyediakan fitur untuk eksekusi scenario secara paralel, sekuensial, dan kondisional, serta pola restartabilitas dan penanganan dalam *exception*. Load Plan dapat dibuat dan dimodifikasi dalam server production.

Load Plan dapat dijalankan, dihentikan dan direstart dari command line, ODI studio, ODI console, atau di interface web service. Objek ini juga dapat dijadwalkan dengan menggunakan runtime agent scheduler ataupun eksternal scheduler. Saat Load Plan dijalankan, *Load Plan instance* akan dibuat dimana kemudian akan dieksekusi terpisah dengan jalannya *Load Plan Run*.

## **Pendahuluan**

Oracle Data Integrator adalah *middleware* yang dirilis oleh Oracle Corp. yang berfungsi dalam proses integrasi platform data yang komprehensif. Oracle Data Integrator terintegrasi dengan seluruh teknologi Oracle, termasuk Oracle Database, Exadata Database Machine, Exalogic, Big data Appliance, WebLogic Server, Business Intelligence, dan Aplikasi Oracle. ODI adalah platform integrasi data yang strategis untuk Oracle.

Oracle Data Integrator sering digunakan untuk mempopulasikan data warehouse sangat besar. Dalam kasus tersebut, sangat umum untuk memiliki ribuan tabel yang akan diisikan menggunakan ratusan scenario. Pelaksanaan scenario ini harus diatur sedemikian rupa bahwa aliran data dari sumber ke target adalah yang paling efisien. Load Plan membantu user mengatur pelaksanaan scenario dalam hierarki step, sekuensial dan paralel untuk jenis kasus tersebut.

Load Plan adalah objek baru yang diperkenalkan dalam Oracle Data Integrator untuk mengatur eksekusi package dan scenario pada tingkat kerumitan yang kompleks. Load Plan menyediakan

fitur untuk eksekusi scenario secara paralel, sekuensial, dan kondisional, restartabilitas, dan penanganan dalam *exception*. Load Plan dapat dibuat dan dimodifikasi dalam server production.

Load Plan adalah objek yang dapat dijalankan di ODI yang dapat terdiri atas step-step yang terhierarki serta dapat dijalankan secara kondisional, paralel ataupun berurutan. Objek dari hierarki yang dijalankan ini adalah scenario-scenario. Pengaturan scenario yang dibuatkan Load Plannya dapat meliputi proses eksekusi dari package, interface dan variable.

Load Plan dapat dijalankan, dihentikan dan di restart dari *command line*, dari ODI studio, ODI console, atau di interface web service. Objek ini juga dapat dijadwalkan dengan menggunakan run-time agent scheduler ataupun eksternal scheduler. Saat Load Plan dijalankan, *Load Plan instance* akan dibuat dimana kemudian akan dieksekusi terpisah dengan jalannya *Load Plan Run*.

*Step* pada Load Plan ODI, adalah komponen proses yang dapat diatur urutan eksekusinya apakah itu bersifat paralel ataupun serial terhadap step lainnya, dimana isi dari step dapat berupa variabel, scenario serta interface.

Repository adalah tempat sentral di mana sebuah agregasi data disimpan dan dipelihara dengan cara yang terorganisir, biasanya dalam sebuah media penyimpanan komputer. Istilah ini berasal dari bahasa Latin repositorium, kapal atau ruang di mana hal-hal tertentu dapat ditempatkan, dan itu bisa berarti tempat di mana hal-hal dikumpulkan.<sup>[6]</sup>

## Isi

Load Plan adalah objek eksekusi terbesar di Oracle Data Integrator. Objek ini menggunakan Scenario dalam langkah-langkah eksekusinya. Ketika suatu objek dieksekusi dalam Load Plan, maka secara otomatis akan diubah menjadi scenario. Sebagai contoh, sebuah package digunakan dalam bentuk scenario package tersebut dalam Load Plan. Perhatikan pula bahwa Load Plan tidak dapat ditambahkan ke Load Plan lainnya. Namun, dimungkinkan untuk menambahkan scenario dalam bentuk *Run Scenario step* yang dijalankan Load Plan lain menggunakan tool *OdiStartLoadPlan*.<sup>[5]</sup>

Setelah dibuat, Load Plan disimpan dalam *work repository*. Load Plan dapat diekspor kemudian diimpor ke repositori lain dan dieksekusi dalam *contexts* yang berbeda. Load Plan juga dapat dibedakan sesuai versinya. Load Plan akan terlihat dalam Designer Navigator dan Operator Navigator di akordeon Load Plan dan Scenario. Load Plan ditampilkan dalam akordeon *Load Plan Executions* di Operator Navigator.

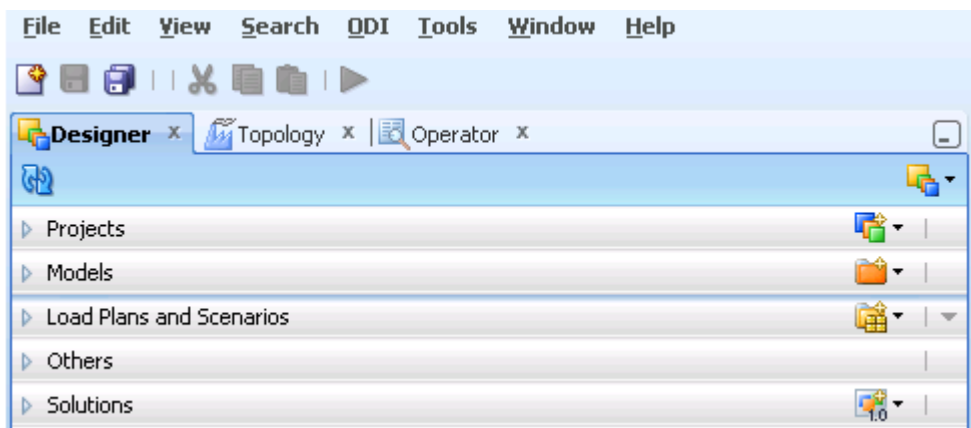
Load Plan bukan pengganti untuk package atau scenario, tetapi digunakan untuk mengatur tingkat eksekusi yang lebih tinggi (kompleks) dalam pelaksanaan package dan scenario. Tidak seperti package, Load Plan benar-benar memberikan dukungan untuk proses eksekusi secara paralel, eksekusi ulang dan penanganan *exception*. *Load Plan instances* dan *Load Plan runs* mirip dengan Sessions. Perbedaannya adalah bahwa ketika session restart, session yang ada diganti oleh eksekusi baru. Load Plan baru yang tereksekusi tidak menimpa pada Load Plan yang sedang berjalan, namun akan ditambahkan setelah Load Plan sebelumnya telah berjalan untuk *Load Plan Instance*. Perhatikan bahwa *Load Plan Instance* tidak dapat diubah pada saat run-time.<sup>[1]</sup>

**Perbedaan antara Load Plan dan Package**

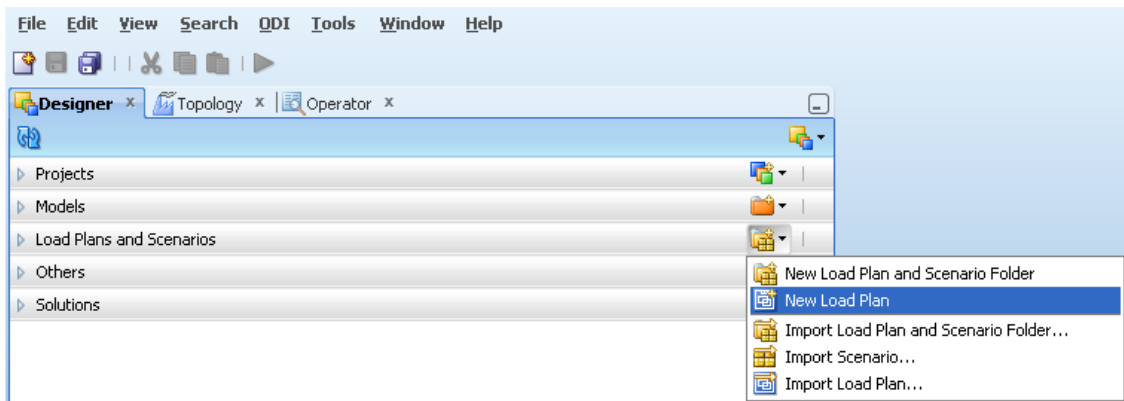
Kapabilitas	Load Plan	Packages	Keterangan
Edition	Design-Time dan Run-Time	Design-Time. Packages dikompilasi ke dalam scenario-scenario pada run-time	Jika production membutuhkan modifikasi pada alur eksekusi, akan lebih baik untuk menggunakan Load Plan
Starting / Monitoring	UI, Command-Line, Web Services, Scheduling	UI, Command-Line, Web Services, Scheduling	Fitur pada kedua objek ini setara
Transaksi	Setiap step Load Plan mengandung transaksi dari Load Plan tersebut	Step-step pada Package dapat membagi transaksinya	jika workflow membutuhkan transaksi yang memunculkan beberapa step yang saling berseberangan , gunakan package
Paralelisme	Ya. Gunakan Parallel step. Eksekusi paralel mudah untuk diikuti dalam tab Operator.	Ya. Dengan menghidupkan scenario lainnya. Eksekusi paralel sulit untuk diikuti dalam tab Operator.	Ketika dalam kondisi sangat dibutuhkan untuk eksekusi paralel step, gunakan pererably Load Plan.
Restartabilitas	Ya. Status dari eksekusi sebelumnya akan tetap ada.	Ya. Status dari eksekusi sebelumnya akan ditimpa. Transaksi database tidak berlanjut, ketika terjadi restart maka seluruh package sering dibutuhkan.	Dikarenakan sifat transaksinya dan fakta bahwa eksekusinya tertimpa oleh eksekusi baru, package sering terestart sebagai unit kecil dari sebuah project. Load plan menghasilkan fleksibilitas yang lebih baik dalam hal restartabilitas.
Percabangan / Loops	Percabangan (Case / When dapat dilakukan). Loops tidak dapat dilakukan.	Percabangan dan Looping dapat dilakukan	Apabila dalam suatu workflow perlu dilakukan looping, gunakan package

**Membuat Load Plan**

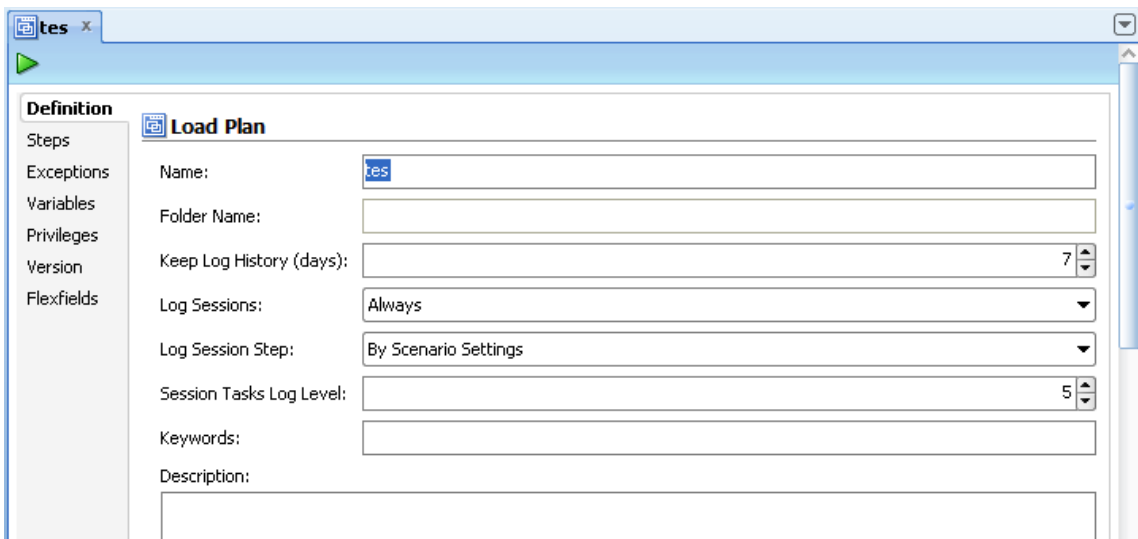
ODI Load Plan terletak pada tab Designer dan Operator Navigator. ODI Load Plan tersedia untuk edisi repositori development dan production, dan dapat diatur ke dalam folder scenario.



Pilih *Load Plans and Scenarios*, "Klik kanan" dan pilih "New Load Plan". [2]



Kemudian tentukan "Nama" untuk Load Plan yang akan dibuat.





Pada tab Steps, kita dapat menambah step-step yang merupakan turunan dari root step. Seperti yang terlihat pada gambar di bawah, kita dapat menentukan jenis step yang akan kita tambahkan, apakah bersifat *Parallel step*, *Serial step*, *Run Scenario step* ataupun *Case-when-else step*.


Pada tab steps juga terdapat tombol-tombol yang akan memudahkan Anda dalam proses editing maupun penyusunan Load Plan, antara lain :





*Search*, Pencarian step dalam tabel *Steps Hierarchy*.

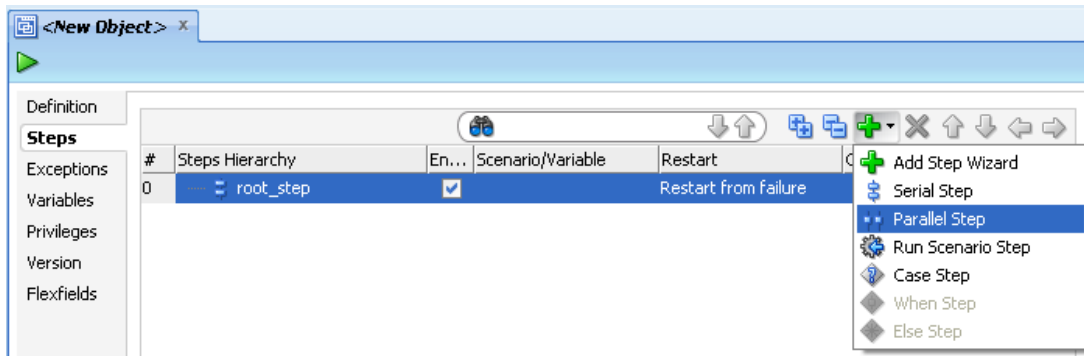
 *Expand All*, memperlihatkan semua kumpulan step di tabel *Steps Hierarchy*.

 *Collapse All*, menciutkan semua kumpulan step di tabel *Steps Hierarchy*.







 *Add Step*, Membuka menu *Add Step*. Anda dapat memilih wizard *Add Step* atau *Quick Step tool* untuk menambahkan step.

 *Remove Step*, Menghapus step yang dipilih dan semua *child step* di bawahnya.

 Panah, gunakan panah untuk memindahkan step yang dipilih untuk posisi yang diperlukan.



Ketika akan menambahkan step pada root step, Anda akan melihat ikon-ikon step yang akan memudahkan Anda dalam menentukan step mana yang harus ditambahkan dalam Load Plan yang sedang Anda buat.

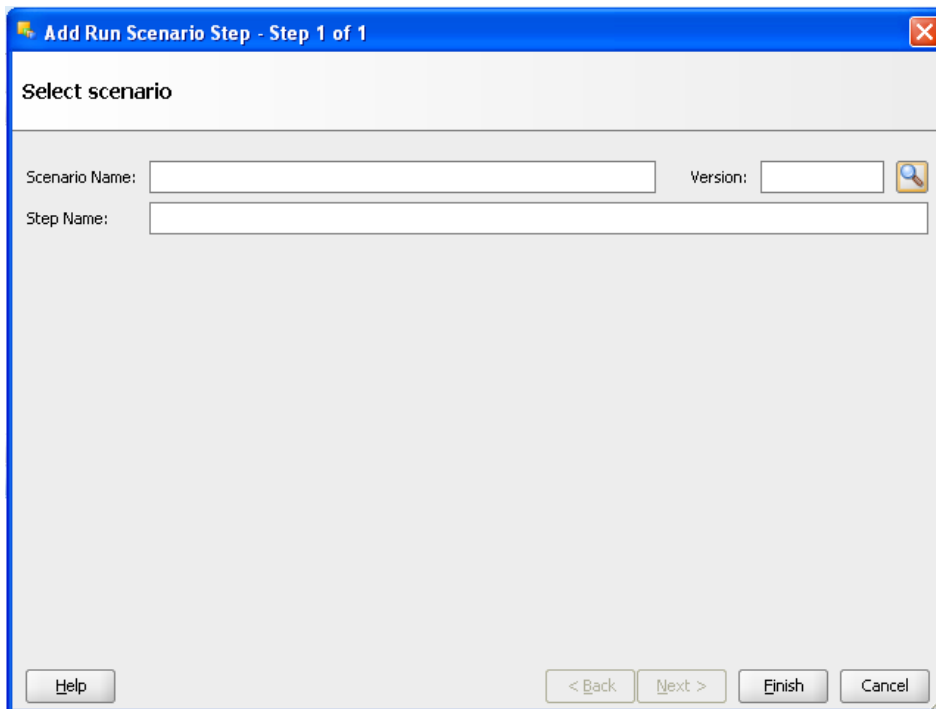
-  *Serial step icon*, menambahkan *Serial step* setelah dilakukan klik pada ikon tersebut. Menggunakan nilai default. Anda dapat mengubah nilai-nilai dalam tabel *Steps Hierarchy* atau di *Property Inspector*.
-  *Parallel step icon*, Menambahkan *Parallel step* setelah dilakukan klik pada ikon tersebut. Menggunakan nilai default. Anda dapat mengubah nilai-nilai dalam tabel *Steps Hierarchy* atau di *Property Inspector*.
-  *Run Scenario step icon*, Menambahkan *Run Scenario step* setelah dilakukan klik pada ikon tersebut.
-  *Case step icon*, Menambahkan *Case step* setelah dilakukan klik pada ikon tersebut.
-  *When step icon*, Menambahkan *When step* setelah dilakukan klik pada ikon tersebut.
-  *Else step icon*, Menambahkan *Else step* setelah dilakukan klik pada ikon tersebut.

### Step-step pada Load Plan

Seperti yang telah dijelaskan pada bagian pembuatan Load Plan, terdapat beberapa tipe step yang bisa dijalankan pada Load Plan. Step-step tersebut antara lain *Parallel step*, *Serial step*, *Run Scenario step* dan *Case-when-else step*. Berikut karakteristik masing-masing step pada Load Plan:

Type	Description	Possible Child Steps
Serial Step	Mendefinisikan sebuah eksekusi child stepsnya. Child steps akan diperintahkan dan dieksekusi hanya saat step sebelumnya telah selesai. Root step adalah sebuah Serial step.	<ul style="list-style-type: none"> <li>* Serial step</li> <li>* Parallel step</li> <li>* Run Scenario step</li> <li>* Case step</li> </ul>
Parallel Step	Mendefinisikan sebuah eksekusi paralel child stepsnya. Child steps dimulai segera sesuai dengan prioritasnya masing-masing.	<ul style="list-style-type: none"> <li>* Serial step</li> <li>* Parallel step</li> <li>* Run Scenario step</li> <li>* Case step</li> </ul>
Run Scenario Step	Melaksanakan eksekusi skenario.	Jenis step ini tidak dapat memiliki Child steps.
Case Step When Step Else Steps	Kombinasi dari step-step ini memungkinkan kondisi percabangan proses eksekusi berdasarkan nilai dari variabel. Catatan: Jika Anda memiliki beberapa When step pada sebuah Case step, hanya when step yang pertama diaktifkan yang memenuhi kondisi ini untuk dieksekusi. Jika tidak ada When step memenuhi kondisi atau Case step tidak mengandung When step, maka Else steplah yang akan dieksekusi.	Untuk Case Step: <ul style="list-style-type: none"> <li>* When step</li> <li>* Else step</li> </ul> Untuk When step: <ul style="list-style-type: none"> <li>* Serial step</li> <li>* Parallel step</li> <li>* Run Scenario step</li> <li>* Case step</li> </ul> Untuk Else step: <ul style="list-style-type: none"> <li>* Serial step</li> <li>* Parallel step</li> <li>* Run Scenario step</li> <li>* Case step</li> </ul>
Exception Step	Mendefinisikan sekelompok step-step yang dieksekusi ketika eksepsi ditemui pada step terkait dari Step Hierarchy. Eksepsi step yang Sama dapat dilampirkan ke beberapa step dalam Step Hierarchy.	<ul style="list-style-type: none"> <li>* Serial step</li> <li>* Parallel step</li> <li>* Run Scenario step</li> <li>* Case step</li> </ul>

Dari penjelasan step-step yang ada pada Load Plan, *Run Scenario step* adalah jenis step dimana Anda dapat memanggil scenario apa yang akan Anda eksekusi melalui Load Plan yang anda buat. Jadi ketika menambahkan *Parallel step* ataupun *Serial step*, jika Anda belum melakukan klik dan drag scenario yang akan dieksekusi, Anda belum diminta menyebutkan scenario yang akan Anda jalankan, baru ketika urutan step telah tersusun, Anda dapat menambahkan *Run Scenario step* dan menyebutkan scenario-scenario mana saja yang ingin Anda eksekusi, dengan menekan tombol “Lookup Scenario” (gambar kaca pembesar) pada wizard di bawah ini



Anda dapat mengatur properti berikut untuk *Run Scenario step* di *Property Inspector*:

- *Priority* : Prioritas untuk step ini diperlukan ketika scenario harus dieksekusi secara paralel. Integer rentang nilai dimulai dari 0 sampai 100 ( 100 menjadi prioritas tertinggi ) . Default nilai adalah 0. Prioritas *Run Scenario step* dievaluasi antara semua *runnable* scenario dalam Load Plan yang sedang tereksekusi . *Run Scenario step* dengan prioritas tertinggi dijalankan paling awal .
- *Context* : *Contexts* yang digunakan untuk eksekusi step. Default *contexts* adalah *context* Load Plan yang didefinisikan dalam Start Dialog Load Plan ketika mengeksekusi Load Plan. Perhatikan bahwa jika Anda hanya menentukan *Contexts* dan tidak ada nilai *logical agent* , step tersebut dimulai pada *physical agent* yang sama pada Load Plan yang tereksekusi, tetapi dalam *contexts* yang telah ditentukan.
- *Logical Agent* : *logical agent* digunakan untuk eksekusi step. Secara default , *logical agent*, adalah yang didefinisikan di Start Dialog Load Plan ketika mengeksekusi Load Plan, dan digunakan dalam tiap eksekusi. Perhatikan bahwa jika Anda menetapkan hanya *logical agent* dan bukan *contexts*, langkah tersebut dimulai dengan *physical agent* yang sesuai dengan *logical agent* tertentu diselesaikan dalam *contexts* tertentu ketika memulai Load Plan. Jika nilai *logical agent* tidak ditentukan, step tersebut dimulai pada *physical agent* yang sama saat Load Plan dieksekusi( apakah *contexts* ditentukan untuk step atau tidak ) .

Objek – objek pada ODI yang dapat digunakan dalam step Load Plan antara lain<sup>[5]</sup> :

- Scenario
- Interface
- Variable

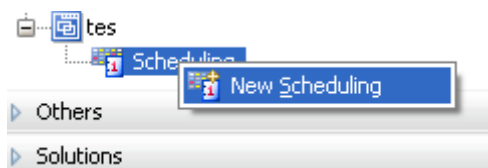
Objek-objek di atas dapat diseret dari simpul Project di sebelah kiri langsung ke Load Plan untuk membuat Step baru. Namun, setiap kali sebuah objek ditambahkan ke Load Plan, apa yang ditambahkan adalah scenario yang dihasilkan dari objek tersebut. Jika Anda menambahkan sebuah interface atau variabel, ODI secara otomatis akan menghasilkan scenario dan menambahkan scenario tersebut dalam Load Plan. Ini berarti bahwa jika Anda melakukan perubahan ke objek setelah ditambahkan ke Load Plan, Anda harus melakukan regenerate scenario, atau membuat versi baru dari scenario dan mengedit nomor versi dalam Load Plan. Salah satu cara untuk membatasi jumlah suntingan adalah dengan menggunakan "-1" untuk nomor versi, sehingga Load Plan selalu menggunakan versi terbaru dari scenario.

### Penjadwalan (Scheduling) pada Load Plan

Anda dapat menjadwalkan eksekusi Load Plan menggunakan *Oracle Data Integrator built-in scheduler* atau *scheduler eksternal*.

Untuk menjadwalkan scenario atau Rencana Beban dari Oracle Data Integrator Studio.

1. Klik kanan node Scheduling di bawah Load Plan di Designer atau Operator Navigator.
2. Pilih New Scheduling. Editor Penjadwalan akan ditampilkan.



3. Pada tab Definition di scheduling editor tentukan parameter-parameter berikut:



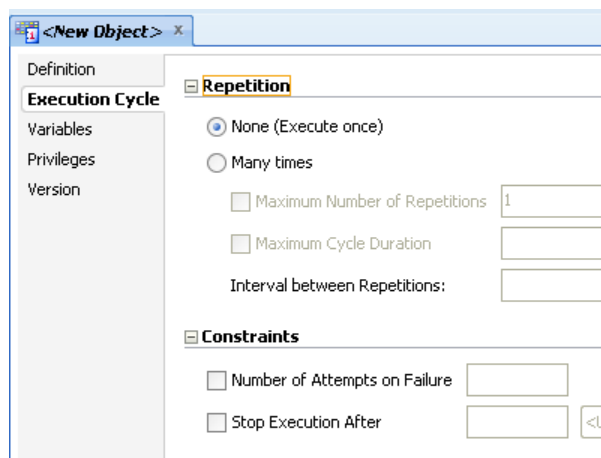
- *Contexts* di mana Load Plan dimulai.
- Agen Agen mengeksekusi scenario atau Rencana Load.
- Log Tingkat Tingkat penebangan informasi untuk mempertahankan.

Parameter Status menentukan aktivasi jadwal

- *Active*, penjadwalan akan aktif ketika agen-restart atau ketika penjadwalan agen fisik diperbarui.
- *Inactive*, jadwal ini tidak aktif dan tidak akan berjalan.
- *Active for the period*, Jadwal aktif untuk jangka waktu hanya akan berjalan dalam periode ini diberikan.

Parameter Eksekusi menentukan frekuensi eksekusi untuk setiap siklus eksekusi. Parameter ini terdiri dari pilihan *On Startup*, *Simple*, *Hourly*, *Daily*, *Weekly*, *Monthly (day of the month)*, *Monthly (week day)* dan *Yearly*.

4. Pada tab *Execution Cycle*, Anda dapat menentukan parameter untuk modus mengulang penjadwalan Load Plan.



5. Pada tab Variabel, batalkan Nilai terbaru untuk variabel yang Anda ingin memberikan Nilai a. Hanya variabel yang digunakan dalam scenario atau Rencana Load dan ditandai sebagai parameter untuk scenario ini atau Rencana Beban muncul dalam tab ini.

6. Dari menu *File*, klik *Save*.

## Penutup

Load Plan merupakan sebuah solusi yang diberikan Oracle Data Integrator untuk penanganan proses eksekusi ETL (*extract, load, and transform*) dengan tata urutan yang kompleks. Dengan adanya Load Plan, seorang DBA dapat meringkas scenario-scenario menjadi satu kesatuan tema Load Plan dimana didalamnya dapat diatur proses restartabilitas, penanganan exception, serta tata urutan masing-masing prosesnya. Load Plan bukan pengganti untuk package atau scenario, tetapi digunakan untuk mengatur tingkat eksekusi yang lebih tinggi (kompleks) dalam pelaksanaan package dan scenario. Tidak seperti package, Load Plan benar-benar memberikan dukungan untuk proses eksekusi secara paralel, eksekusi ulang dan penanganan *exception*.

## Referensi

[1] Oracle, 2010, "*Working with Load Plans*"

[http://docs.oracle.com/cd/E28280\\_01/integrate.1111/e12643/loadplans.htm](http://docs.oracle.com/cd/E28280_01/integrate.1111/e12643/loadplans.htm)

diakses tanggal 23 Agustus 2013

[2] *Patrapalle, Varun Reddy. 2013, "Oracle Data Integrator (ODI): Executing A Load Plan"*



<http://www.kpipartners.com/blog/bid/148191/Executing-An-ODI-Load-Plan>  
diakses tanggal 23 Agustus 2013

[3] Rittman, Mark. 2011, “*ODI 11g New Mapping and Interface features Part-2- Load Plans*”  
<http://www.rittmanmead.com/2011/06/odi-11g-new-mapping-and-interface-features-part-2-load-plans/>  
diakses tanggal 5 September 2013

[4] Balkar, Karan. 2013, “*Tutorial #50: Using Load Plans in Oracle Data Integrator*”  
<http://karanbalkar.com/2013/08/tutorial-50-using-load-plans-in-oracle-data-integrator/>  
diakses tanggal 5 September 2013

[5] Hecksel, David. 2012, “*Getting Started with Oracle Data Integrator 11g*”, Packt Publishing  
<http://www.google.com/search?tbo=p&tbm=bks&q=isbn:184968068X>  
diakses tanggal 7 Oktober 2013

[6] Margaret Rouse, “*Repository*”, 2005  
<http://searchoracle.techtarget.com/definition/repository>  
diakses tanggal 8 Oktober 2013

## Biografi Penulis



**Yuafanda Kholfi Hartono.** Menyelesaikan S1 di Universitas Indonesia. Sebelumnya menamatkan pendidikan di Sekolah Tinggi Akuntansi Negara, minat dan ketertarikan pada bidang IT membuat pada akhirnya ditempatkan pada Direktorat Jenderal Bea dan Cukai Kementerian Keuangan sebagai Database Administrator (DBA).