**Title**:

# **Manage PostgreSQL Database and Users**

**Description**:

Workflow for managing DataBase and User Roles on PostgreSQL. This bundle has dependency on Core and Validation KIs bundles.

**Readme**

**Important**: the Knowledge Items of this bundle try to simulate Workflows seen in frameworks such as Chef, Puppet and Ansible.

For this reason, Workflow KIs seem to not follow traditional best practices of KI Creation in terms of reusability. They are, however atomic pieces of knowledge, meaning the knowledge cannot be broken down further while still maintaining the same level of functionality.

Due to the controlled nature of the Workflow approach, the Workflow KIs try to structure Tasks like steps in a sequence rather than letting AutoPilot pick the path to a solution.

In order to simulate these Workflows, the Knowledge Items are split into 3 groups:

* Workflow KIs
* Validation KIs and
* Core KIs.

This bundle intends to Manage PostgrSQL Database and User.

**Dependencies**

1. Core KIs bundle
2. Validation KIs bundle

For this bundle to run, you will need MARS nodes in AutoPilot engine. For your reference, please see sample nodes below (at the end of this readme).

**Steps to run**

1. Put the validation KIs bundle in AutoPilot engine
2. Put the core KIs bundle in AutoPilot engine
3. Put the Workflow KI according to the usecase like to add a new DB *PostgreSQLDBManagementWorkflow*.xml , to remove a DB *PostgreSQLRemoveDBWorkflow.xml,* to remove a user  *PostgreSQLRemoveUserWorkflow.xml,* to add a user  *PostgreSQLUserManagementWorkflow.xml* (all these sample KIs are attached with this bundle) in AutoPilot engine
4. Put the Issue in AutoPilot engine to trigger this KI
   * For your reference, please see sample issue below
5. This will trigger the KI and run the workflow to create/delete a DataBase and User on PostgreSQL on your target machine (as specified in the Issue)

Workflow KI in this bundle, runs below steps on the target machine (as specified in the Issue)

1. Create Database with specified name
2. Drop Database with specified name
3. Drop user which is specified in issue
4. Create user with password specified.

**Sample Issue**

Issue:

NodeID: OpexSoftware:Workflow:Application:EnterpriseInfrastructure

xmlns: http://www.arago.de/IssueSchema

IssueSubject: 'Creation of PostgreSQL database'

PostgreSQLDBManagementWorkflow:

TargetApp: EnterpriseInfrastructure

Database: TempDB

TargetMachine: zeeshan

TargetState: Created

User: postgres

Host: 192.168.1.90

Issue:

NodeID: OpexSoftware:Workflow:Application:EnterpriseInfrastructure

xmlns: http://www.arago.de/IssueSchema

IssueSubject: 'Remove of PostgreSQL database and Users'

PostgreSQLRemoveUserWorkflow:

TargetApp: EnterpriseInfrastructure

TargetMachine: zeeshan

DBUserName: PK

TargetState: Created

User: postgres

Host: 192.168.1.90

Issue:

NodeID: OpexSoftware:Workflow:Application:EnterpriseInfrastructure

xmlns: http://www.arago.de/IssueSchema

IssueSubject: 'Creation of PostgreSQL database and Users'

PostgreSQLUserManagementWorkflow:

TargetApp: EnterpriseInfrastructure

TargetMachine: zeeshan

DBUserName: PK

DBPassword: PK

TargetState: Created

User: postgres

Host: 192.168.1.90

Issue:

NodeID: OpexSoftware:Workflow:Application:EnterpriseInfrastructure

xmlns: http://www.arago.de/IssueSchema

IssueSubject: 'Remove of PostgreSQL database'

PostgreSQLRemoveDBWorkflow:

TargetApp: EnterpriseInfrastructure

Database: TempDB

TargetMachine: zeeshan

TargetState: Created

User: postgres

Host: 192.168.1.90

**Machine Node**

<Linux xmlns="http://mars-o-matic.com" ID="OpexSoftware:Workflow:Machine:zeeshan"

HasAgentType\_WatchMe="False" MachineArchitecture="x86\_64"

NodeType="Machine" MachineClass="Linux" NodeName="zeeshan">

<Dependencies>

<Node ID="OpexSoftware:Workflow:Software:PostgreSQL" />

</Dependencies>

<CustomerInformation ID="opex.com" Name="opex"/>

<Extensions>

<ssh User="zeeshan" />

<PasswordPolicy MinLength="8" MaxLength="14" CharSet="alnum" />

</Extensions>

<OSInformation Name="Ubuntu" MajorVersion="14" Vendor="Ubuntu" />

<NetworkInformation>

<InterfaceInformation>

<Interface Name="eth1" IP="192.168.1.90" />

</InterfaceInformation>

</NetworkInformation>

</Linux>

**Software Node**

<PostgreSQL xmlns="http://mars-o-matic.com" ID="OpexSoftware:Workflow:Software:PostgreSQL"

NodeType="Software" SoftwareClass="DBMS"

SoftwareSubClass="PostgreSQL" NodeName="zeeshan">

<Dependencies>

<Node ID="OpexSoftware:Workflow:Machine:zeeshan" />

<Node ID="OpexSoftware:Workflow:Resource:WebResource" />

</Dependencies>

<CustomerInformation ID="opex.com" Name="opex"/>

</PostgreSQL>

**Resource Node**

<Service xmlns="http://mars-o-matic.com" ID="OpexSoftware:Workflow:Resource:WebResource"

NodeType="Resource" ResourceClass="Service" NodeName="WebResource">

<Dependencies>

<Node ID="OpexSoftware:Workflow:Software:PostgreSQL" />

<Node ID="OpexSoftware:Workflow:Application:EnterpriseInfrastructure" />

</Dependencies>

<CustomerInformation ID="opex.com" Name="OpexSoftware" />

</Service>

**Application Node**

<EnterpriseInfrastructure xmlns="http://mars-o-matic.com" ID="OpexSoftware:Workflow:Application:EnterpriseInfrastructure"

NodeType="Application" ApplicationClass="Enterprise"

ApplicationSubClass="EnterpriseInfrastructure" NodeName="EnterpriseInfrastructure">

<Dependencies>

<Node ID="OpexSoftware:Workflow:Resource:WebResource" />

</Dependencies>

<CustomerInformation ID="opex.com" Name="OpexSoftware" />

</EnterpriseInfrastructure>