Documentation

IBM Workload Scheduler integration with SAP Data Services

Written by : **Miguel Sanders** Uniforce	**Date :** **August 1 2016**

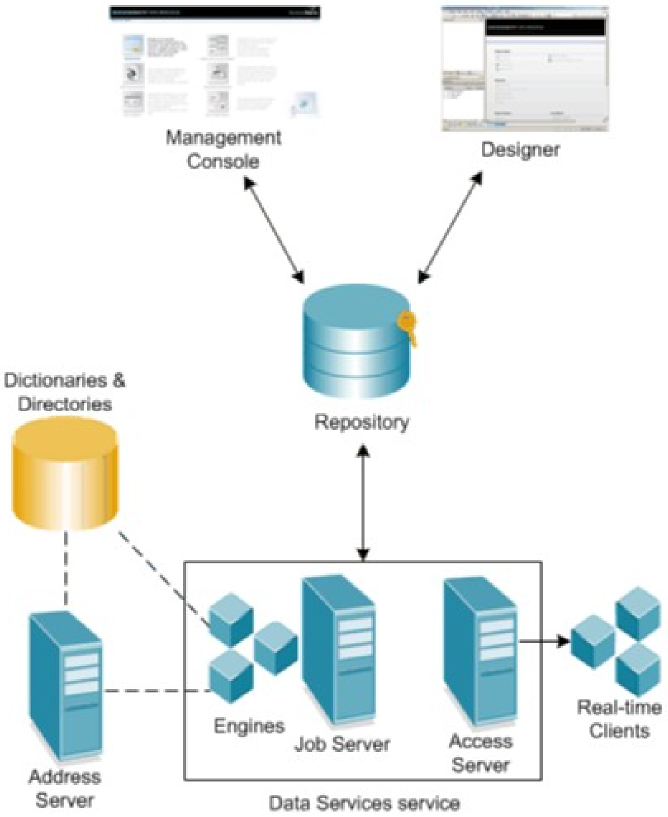
Table of Contents

CHANGE HISTORY

Version	Date of change	Change detail
1.0	August 1 2016	Initial version by M. Sanders

1. INTRODUCTION AND ARCHITECTURE

SAP Data Services (formerly known as SAP BusinessObjects Data Services) delivers a single enterprise-class solution for data integration, data quality, data profiling, and text data processing that allows you to integrate, transform, improve, and deliver trusted data to critical business processes. It enables IT organizations to lower TCO and accelerate time to value. With SAP Data Services, IT organizations can maximize operational efficiency with a single solution to improve data quality and gain access to heterogeneous sources and applications. By leveraging the IBM Workload Scheduler plug-in for SAP Data Services, you will be able to schedule these mission critical jobs from a single interface.



2. INSTALLING AND CONFIGURING THE PLUG-IN FOR SAP DATA SERVICES

The following prerequisites must be met in order to use the IBM Workload Scheduler plug-in for SAP Data Services.

- IBM Workload Scheduler 9.1 or later
- IBM Dynamic Workload Console 9.1 or later

To install and configure the IBM Workload Scheduler plug-in for SAP Data Services, perform the following steps:

- Copy *com.ibm.scheduling.agent.sapbods_<version>.jar* to the *<TWA_HOME>/TWS/applicationJobPlugIn* folder on either the Master Domain Manager or a Dynamic Domain Manager
- Copy *com.ibm.scheduling.agent.sapbods_<version>.jar* to the *<TWA_HOME>/TWS/JavaExt/eclipse/plugins* folder on the Dynamic Agent that will run the SAP Data Services jobs
- Modify *config.ini* located in *<TWA_HOME>/TWS/JavaExt/eclipse/configuration* on the Dynamic Agent that will run the SAP Data Services jobs. At the end of the line that starts with "osgi.bundles=", add the following: *",com.ibm.scheduling.agent.sapbods@4:start"*
- Restart the WebSphere Application Server of either the Master Domain Manager or the Dynamic Domain Manager.
- Restart the Dynamic Agent
- Restart the Dynamic Workload Console

3. DEFINING A SAP DATA SERVICES JOB

From the Dynamic Workload Console, you can define a SAP Data Services job as follows:

- In the Dynamic Workload Console navigation tree, expand *Administration* and select *Manage Workload Definitions*.
- Specify the name of the engine. Subsequently, the Workload Designer is displayed.
- In the Working List panel, select *New* -> *Job Definition* -> *ERP* -> *SAP Data Services*.
- On the *SAP Data Services* panel, fill in the job details
    - **CMC hostname** : Hostname of the *Central Management Console*.
    - **CMC port** : Port number of the *Central Management Console*.
    - **CMS system** : The *Central Managent Service* system name. If the CMS server is listening on the default 6400 port, then pass only the CMS system name. If the server is listening on any other port, then also pass the port number with a colon.
    - **CMS username** : The CMS username.
    - **CMS password** : Password of the CMS user.
    - **CMS authentication** : Specifies the type of authentication to use for logging on to a Data Services web service. Values include:

      - Enterprise

      - LDAP

      - Windows AD

      - SAP
    - To validate the connection, click *Test Connection*.
    - **Repository** : The name of the repository.
    - **Job name** : The name of the batch job. Please note that this batch job needs to be published as a web service. Please refer to Appendix A.
    - **Polling period** (*) : The monitoring frequency determines how often the job is monitored. The default value is 10 seconds.
    - **Timeout** (*): The monitoring time determines for how long the job is monitored. If the job hasn't finished by the end of the timeout interval, the job is marked as ABEND. The default value is 7200 seconds.
    - **Job server** : The name of the job server where the batch job will be executed.
    - **Server group** : The name of the server group.
    - **Distribution level** : You can distribute the execution of a job or a part of a job across multiple job servers within a server group to better balance resource-intensive operations. You can specify the following values on the distribution level option when you execute a job:

      - Job : A job can execute on an available job server

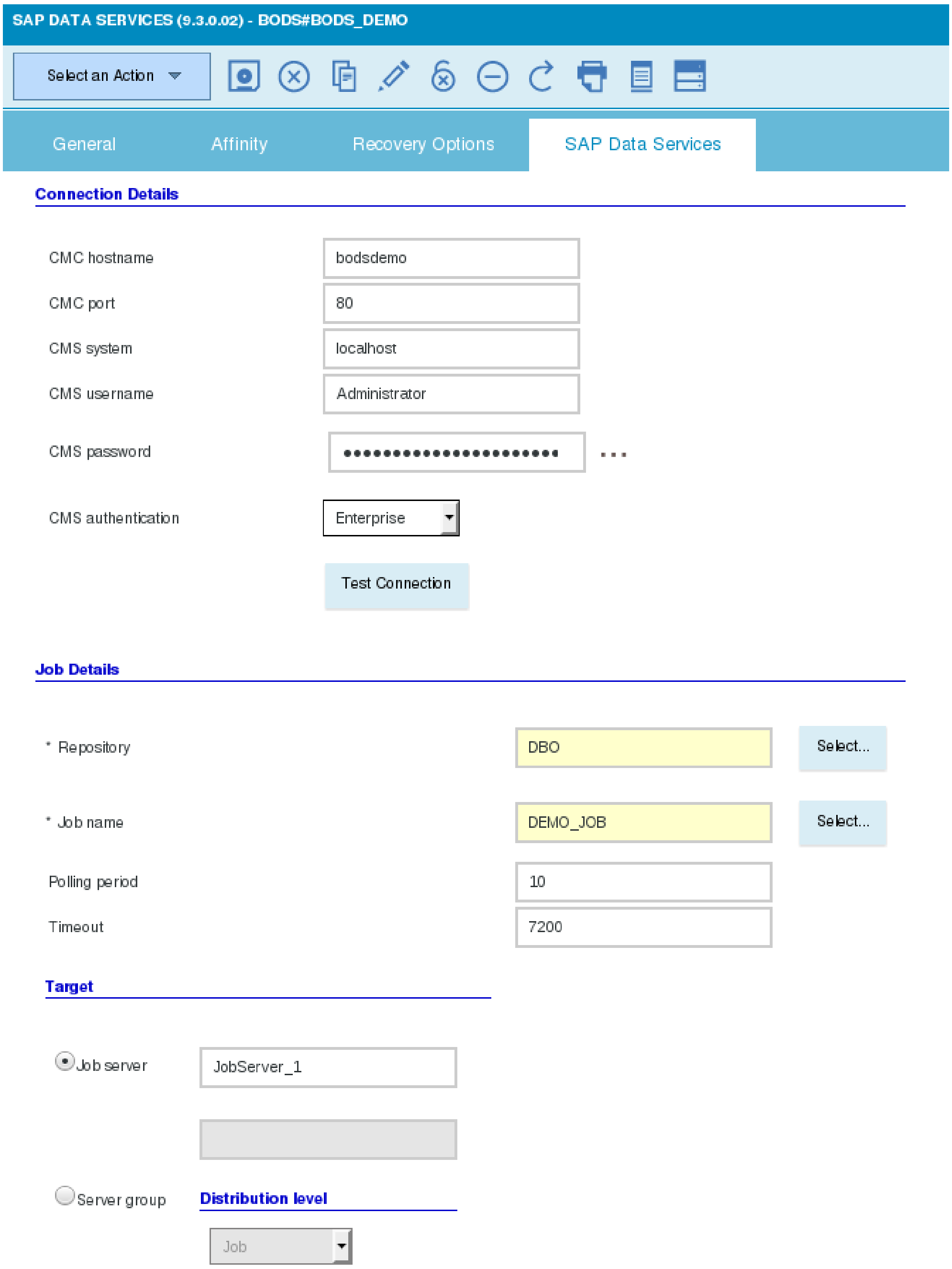
      - Data flow : Each data flow within a job can execute on an available job server.

- Sub data flow : A resource-intensive operation (such as a sort, table comparison, or table lookup) within a data flow can execute on an available job server.

- (*) These parameters can also be supplied in a common properties file *SAPBODSJobExecutor.properties* located in *<TWA_HOME>/TWS/JavaExt/cfg* on the Dynamic Agent that will run the SAP Data Services jobs. Values from the job definition (if provided) override the values in the properties file.

Additionally, the following properties are available in the properties file:

- proxyServerType : HTTP / HTTPS or SOCKS
- proxyServer : Hostname or IP address of the proxy server
- proxyServerPort : Port of the proxy server



Connection Details

CMC hostname	bodsdemo
CMC port	80
CMS system	localhost
CMS username	Administrator
CMS password	••••••••••••••••••••••
CMS authentication	Enterprise

Job Details

* Repository	DBO
* Job name	DEMO_JOB
Polling period	10
Timeout	7200

Target

● Job server	JobServer_1
○ Server group	

Distribution level

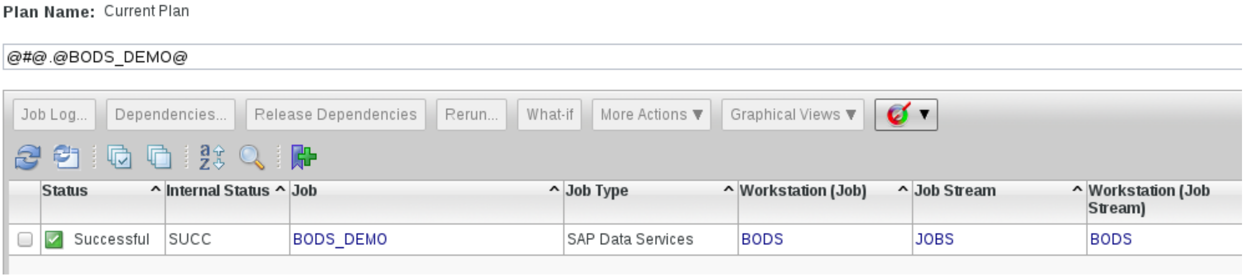
Job

Alternatively, the *composer* command line can be used to define the job. Example:

```
BODS#BODS_DEMO

 TASK

<?xml version="1.0" encoding="UTF-8"?>

<jsdl:jobDefinition xmlns:jsdl="http://www.ibm.com/xmlns/prod/scheduling/1.0/jsdl"
xmlns:jsdlsapbods="http://www.ibm.com/xmlns/prod/scheduling/1.0/jsdlsapbods" name="SAPBODS">

  <jsdl:application name="sapbods">

    <jsdlsapbods:sapbods>

      <jsdlsapbods:SAPBODSParameters>

        <jsdlsapbods:SAPBODSParms>

          <jsdlsapbods:serverInformation>

            <jsdlsapbods:hostname>bodsdemo</jsdlsapbods:hostname>

            <jsdlsapbods:port>80</jsdlsapbods:port>

            <jsdlsapbods:system>localhost</jsdlsapbods:system>

            <jsdlsapbods:username>Administrator</jsdlsapbods:username>

            <jsdlsapbods:password>{aes}JT14Thm5ANxEh12Vm9/qAE0xkk+U=</jsdlsapbods:password>

            <jsdlsapbods:authType>secEnterprise</jsdlsapbods:authType>

          </jsdlsapbods:serverInformation>

          <jsdlsapbods:jobDetails>

            <jsdlsapbods:repository>DBO</jsdlsapbods:repository>

            <jsdlsapbods:jobName>DEMO_JOB</jsdlsapbods:jobName>

            <jsdlsapbods:pollingPeriod>10</jsdlsapbods:pollingPeriod>

            <jsdlsapbods:timeout>7200</jsdlsapbods:timeout>

            <jsdlsapbods:target>

              <jsdlsapbods:jobServer>

                <jsdlsapbods:jobServerValue>JobServer_1</jsdlsapbods:jobServerValue>

              </jsdlsapbods:jobServer>

            </jsdlsapbods:target>

          </jsdlsapbods:jobDetails>

          <jsdlsapbods:executionDetails>

            <jsdlsapbods:enableAuditing/>

            <jsdlsapbods:useCollectedStatistics/>

          </jsdlsapbods:executionDetails>

          <jsdlsapbods:jobParameters>

            <jsdlsapbods:parametersValues>

              <jsdlsapbods:parametersValue key="TEST">100</jsdlsapbods:parametersValue>

            </jsdlsapbods:parametersValues>

          </jsdlsapbods:jobParameters>

        </jsdlsapbods:SAPBODSParms>

      </jsdlsapbods:SAPBODSParameters>

    </jsdlsapbods:sapbods>

  </jsdl:application>

</jsdl:jobDefinition>

 DESCRIPTION "Added by composer."

 RECOVERY STOP
```

4. MONITORING A SAP DATA SERVICES JOB

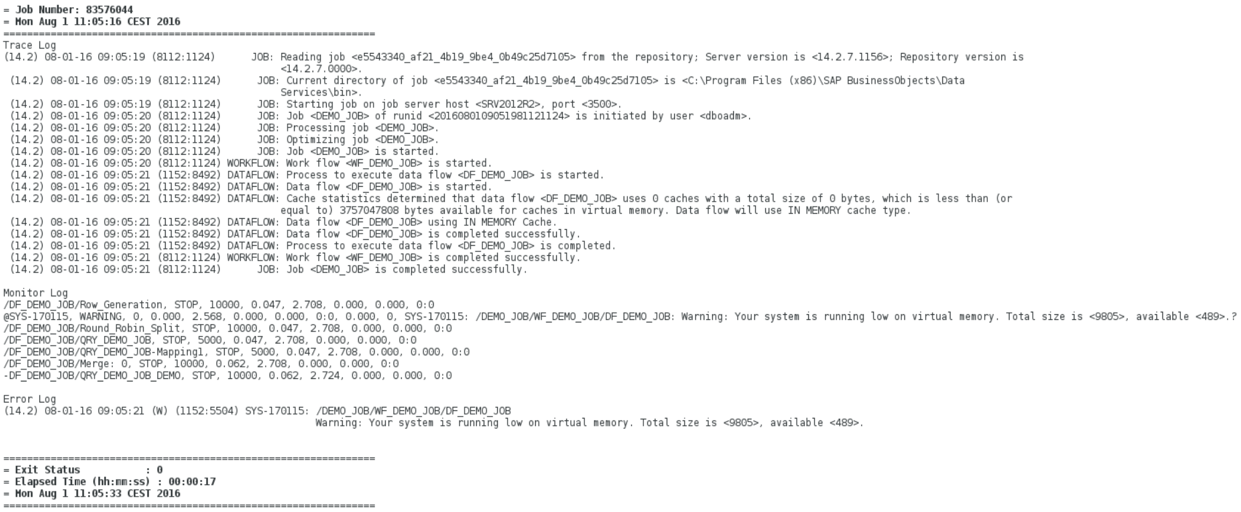
Like regular jobs, you can monitor SAP Data Services jobs by using either the Dynamic Workload Console or the *conman* command line.



Plan Name: Current Plan

@#@.@BODS_DEMO@

	Status	Internal Status	Job	Job Type	Workstation (Job)	Job Stream	Workstation (Job Stream)
□	✓ Successful	SUCC	BODS_DEMO	SAP Data Services	BODS	JOBS	BODS



```
= Job Number: 83576044
= Mon Aug 1 11:05:16 CEST 2016
==================================================================
Trace Log
(14.2) 08-01-16 09:05:19 (8112:1124)      JOB: Reading job <e5543340_af21_4b19_9be4_0b49c25d7105> from the repository; Server version is <14.2.7.1156>; Repository version is
                                                <14.2.7.0000>.
 (14.2) 08-01-16 09:05:19 (8112:1124)      JOB: Current directory of job <e5543340_af21_4b19_9be4_0b49c25d7105> is <C:\Program Files (x86)\SAP BusinessObjects\Data
                                                Services\bin>.
 (14.2) 08-01-16 09:05:19 (8112:1124)      JOB: Starting job on job server host <SRV2012R2>, port <3500>.
 (14.2) 08-01-16 09:05:20 (8112:1124)      JOB: Job <DEMO_JOB> of runid <20160801090519811211124> is initiated by user <dboadm>.
 (14.2) 08-01-16 09:05:20 (8112:1124)      JOB: Processing job <DEMO_JOB>.
 (14.2) 08-01-16 09:05:20 (8112:1124)      JOB: Optimizing job <DEMO_JOB>.
 (14.2) 08-01-16 09:05:20 (8112:1124)      JOB: Job <DEMO_JOB> is started.
 (14.2) 08-01-16 09:05:20 (8112:1124) WORKFLOW: Work flow <WF_DEMO_JOB> is started.
 (14.2) 08-01-16 09:05:21 (1152:8492) DATAFLOW: Process to execute data flow <DF_DEMO_JOB> is started.
 (14.2) 08-01-16 09:05:21 (1152:8492) DATAFLOW: Data flow <DF_DEMO_JOB> is started.
 (14.2) 08-01-16 09:05:21 (1152:8492) DATAFLOW: Cache statistics determined that data flow <DF_DEMO_JOB> uses 0 caches with a total size of 0 bytes, which is less than (or
                                                equal to) 3757047808 bytes available for caches in virtual memory. Data flow will use IN MEMORY cache type.
 (14.2) 08-01-16 09:05:21 (1152:8492) DATAFLOW: Data flow <DF_DEMO_JOB> using IN MEMORY Cache.
 (14.2) 08-01-16 09:05:21 (1152:8492) DATAFLOW: Data flow <DF_DEMO_JOB> is completed successfully.
 (14.2) 08-01-16 09:05:21 (1152:8492) DATAFLOW: Process to execute data flow <DF_DEMO_JOB> is completed.
 (14.2) 08-01-16 09:05:21 (8112:1124) WORKFLOW: Work flow <WF_DEMO_JOB> is completed successfully.
 (14.2) 08-01-16 09:05:21 (8112:1124)      JOB: Job <DEMO_JOB> is completed successfully.

Monitor Log
/DF_DEMO_JOB/Row_Generation, STOP, 10000, 0.047, 2.708, 0.000, 0.000, 0:0
@SYS-170115, WARNING, 0, 0.000, 2.568, 0.000, 0.000, 0:0, 0.000, 0, SYS-170115: /DEMO_JOB/WF_DEMO_JOB/DF_DEMO_JOB: Warning: Your system is running low on virtual memory. Total size is <9805>, available <489>.?
/DF_DEMO_JOB/Round_Robin_Split, STOP, 10000, 0.047, 2.708, 0.000, 0.000, 0:0
/DF_DEMO_JOB/QRY_DEMO_JOB, STOP, 5000, 0.047, 2.708, 0.000, 0.000, 0:0
/DF_DEMO_JOB/QRY_DEMO_JOB-Mapping1, STOP, 5000, 0.047, 2.708, 0.000, 0.000, 0:0
/DF_DEMO_JOB/Merge: 0, STOP, 10000, 0.062, 2.708, 0.000, 0.000, 0:0
-DF_DEMO_JOB/QRY_DEMO_JOB_DEMO, STOP, 10000, 0.062, 2.724, 0.000, 0.000, 0:0

Error Log
(14.2) 08-01-16 09:05:21 (W) (1152:5504) SYS-170115: /DEMO_JOB/WF_DEMO_JOB/DF_DEMO_JOB
                                                     Warning: Your system is running low on virtual memory. Total size is <9805>, available <489>.


==================================================================
= Exit Status               : 0
= Elapsed Time (hh:mm:ss) : 00:00:17
= Mon Aug 1 11:05:33 CEST 2016
==================================================================
```

5. APPENDIX A : PUBLISH A SAP DATA SERVICES JOB AS A WEB SERVICE

In order to schedule a SAP Data Services batch job, you will need to publish this batch job as a web service in the *Central Management Console*.

Perform the following steps :

- Logon to the *Central Management Console.*
- Select *Administrator.*
- Select the *Web Services Configuration* tab and select *Add Batch Job ...*from the drop-down list and click *Apply.*
- Select the batch job that you want to publish as a web service and finally click *Add.*