

# IBM Bluemix cloudantDB and Sterling Connect:Direct integration

---

---

<b>Authors:</b>	Mehmet Cambaz
<b>Version:</b>	1.0
<b>Date Created:</b>	04.10.2017
<b>Last Updated:</b>	15.10.2017

---

## Disclaimer:

All of the information is given “as-is” basis and author do not give any kind of guarantee of any sort, use it as your own risk.

### Document Revision History

Revision	Date	Description	Author
1.0	15.10.2017	First Draft	Mehmet Cambaz

### Document Approval History

Name	Signature	Date	Version Approved/Comments

### Reference Documents

Document Name	Description	Owner	Location

## Table of Contents

1. Introduction
2. Content

## 1. Introduction

This document's purpose is to give helpful information on how to integrate IBM Sterling Connect:Direct with IBM Bluemix to show capabilities of an on-premise software such as Sterling Connect:Direct could be used in Hybrid Cloud environments such as IBM Bluemix via REST APIs.

The components that are used to demonstrate a use case which is a step in a Sterling Connect:Direct process **assuming** that a file is copied to PNODE and all the JSON (JSON is lightweight data format that is used in REST APIs. For more details see: <http://www.json.org/>) transformations are done (It can be done via IBM Transformation Extender) and the content is ready to send to IBM Bluemix CloudantDB via REST API.

- **IBM Bluemix**, you need a trial (free) or paid subscription to use Bluemix capabilities. It might take couple of days for your account to be approved. Register link: <https://console.bluemix.net/registration/>
- **IBM Sterling Connect:Direct** for Windows v4.7 (latest)
- **Curl** for command line based REST API processing, could be publicly downloaded from here: <https://curl.haxx.se/> (command line tool and library for transferring data with URLs)

After downloading the relevant curl version, you can unzip and copy the "bin" folder to your virtual machine's C: folder so that you can use C:\bin as path to use curl

- **IBM Bluemix Cloudant** (Cloudant NoSQL DB provides access to a fully managed NoSQL JSON data layer that's always on. This service is compatible with CouchDB, and accessible through a simple to use HTTP interface for mobile and web application models) database, an instance could be created from: <https://console.bluemix.net/catalog/?category=data>



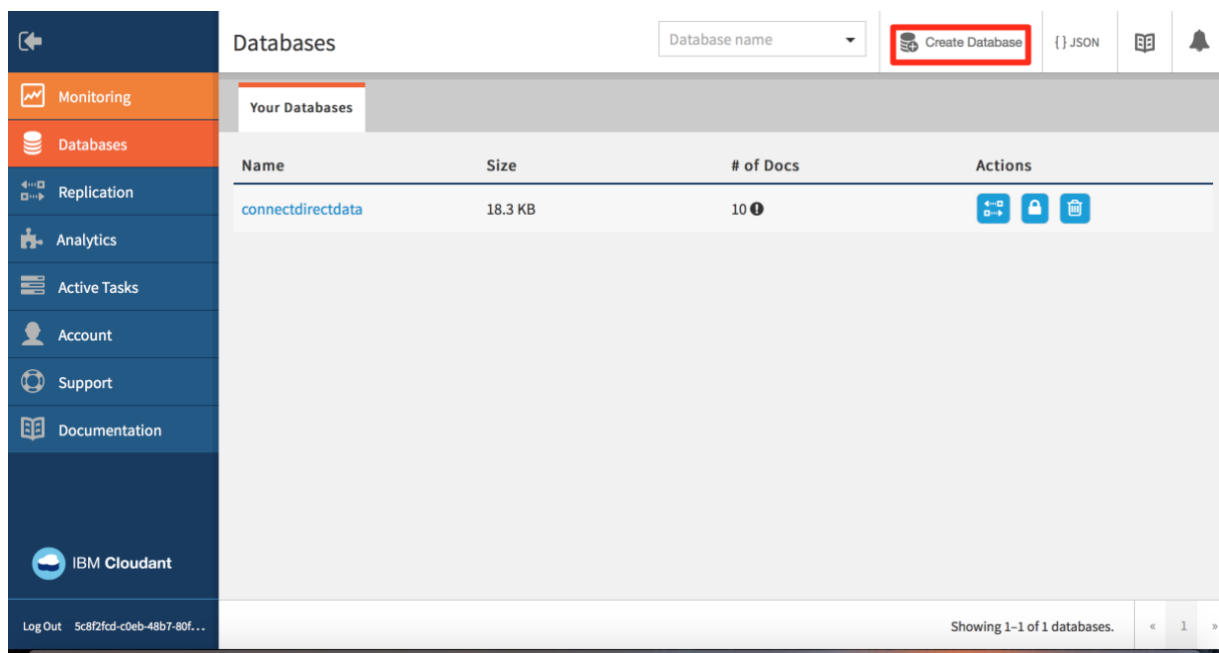
Select pricing as Lite since it is free for IBM CSPs and is enough for the use case.

Pricing Plans			Monthly prices shown are for country or region: <a href="#">United Arab Emirates</a>
PLAN	FEATURES	PRICING	
✓ Lite	<b>1 GB of data storage</b> Provisioned throughput capacity: 20 Lookups/sec 10 Writes/sec 5 Queries/sec	Free	
<hr/>			
The Lite plan provides access to the full functionality of Cloudant for development and evaluation. The plan has a set amount of provisioned throughput capacity as shown and includes a max of 1GB of encrypted data storage.			
Lite plan services are deleted after 30 days of inactivity.			

After you create the database instance itself, you could create the database you will use in this document from the Bluemix management user interface. To open the CloudantDB:



Click “Launch”



Click “Create Database” and give the database name “**connectdirectdata**”

You can see the contents of the database like this:

The screenshot shows the IBM Cloudant web interface for a database named 'connectdirectdata'. The left sidebar contains navigation links: Monitoring, Databases (selected), Replication, Analytics, Active Tasks, Account, Support, and Documentation. The main content area is divided into a left pane with 'All Documents', 'Query', 'Permissions', 'Changes', and 'Design Documents' (all with expand/collapse icons), and a right pane displaying a table of documents. The table has columns for 'id', 'key', and 'value'. It lists 10 documents, each with a unique ID and key, and a JSON value containing a 'rev' field. At the bottom, it indicates 'Showing document 1 - 10. Documents per page: 20'.

id	key	value
1cdde35e876f36efc37c2...	1cdde35e876f36efc37c2...	{"rev": "1-8fa137159845..."}
2896d55e1ca90c46102ca...	2896d55e1ca90c46102ca...	{"rev": "1-d4097ealc8d..."}
2b7c10cd9a024d4da62d...	2b7c10cd9a024d4da62d...	{"rev": "1-1df6f73b397d..."}
4a87bb7b9255dfa4dc8af...	4a87bb7b9255dfa4dc8af...	{"rev": "1-522323fb50b..."}
52af24e37ff5d3fce5e5b8...	52af24e37ff5d3fce5e5b8...	{"rev": "1-6816cd4539e3..."}
6fc0c9526f7700b1773c1...	6fc0c9526f7700b1773c1...	{"rev": "1-13429227c57e..."}
6fc0c9526f7700b1773c1...	6fc0c9526f7700b1773c1...	{"rev": "1-13c88cbc4089..."}
b8406e2f795c935363a56...	b8406e2f795c935363a56...	{"rev": "1-6a10fbcc716..."}
d14dcbe84e642b3964e3...	d14dcbe84e642b3964e3...	{"rev": "1-33e6b65a09ac..."}
d14dcbe84e642b3964e3...	d14dcbe84e642b3964e3...	{"rev": "1-dall1c8b630b..."}

To give access to your application via API key and password, click “Permissions”

The screenshot shows the 'Permissions' tab in the IBM Cloudant interface. The left sidebar is the same as the previous image. The main content area shows a table of permissions for the database 'connectdirectdata'. The table has columns for the permission type (e.g., \_admin, \_reader, \_writer, \_replicator) and a checkbox to grant or revoke the permission. The 'Unauthenticated connections' row is highlighted. Below the table, there is a section for 'Grant database permissions to:' with a text input field for 'Username or API Key' and a 'Grant Permissions' button. A large black arrow points to the 'Generate API Key' button. Below this, there is a section for 'API keys' with a 'Generate API Key' button. A text box shows the generated 'Key' and 'Password', with a note: 'Please make a note of the password. For security reasons, we will not store it for you to retrieve in the future.'

Then click “Generate API Key”, also give that key `_reader` and `_writer` access so that it can do CRUD operations to the database. It will give **key** and **password** you will use these two information to connect to this database.

## 2. Content

Sterling Connect:Direct can call the REST APIs CloudantDB provides via command line execution leveraging the curl libraries and executables. Sample curl usage for json POST action:

```
curl -H "Content-Type: application/json" -X POST -d  
'{"username":"xyz","password":"xyz"}' http://localhost:3000/api/login
```

blue is the json data part

red is the url part to post

When using windows OS (which is the case for the sample virtual machine I used) the escape characters should be included in \ for example:

```
curl -X POST -H "Content-Type: application/json" -d "{ \"key1\": \"value1\"  
}" http://localhost:3000/api/method
```

See more details here: <https://stackoverflow.com/questions/7172784/how-to-post-json-data-with-curl-from-terminal-commandline-to-test-spring-rest>

At our sample with CloudantDB the right script for json POST is this:

```
C:\bin\curl --user APIKEYGENERATED:APIKEYPASSWORD -X POST -H "Content-Type: application/json" -d JSONDATA https://YOURCLOUDANTINSTANCEID-bluemix.cloudant.com/connectdirectdata/
```

This command will insert the **JSONDATA** provided to the database as a record.

In order to make things more simple we will add this script to batch script file ending with .bat and add some variable as JSONDATA to get another file's content to JSONDATA.

```
for /f "delims=" %%x in (C:\bin\input.txt) do set Build=%%x  
C:\bin\curl --user APIKEYGENERATED:APIKEYPASSWORD -X POST -H "Content-Type: application/json" -d %Build% https://YOURCLOUDANTINSTANCEID -bluemix.cloudant.com/connectdirectdata/
```

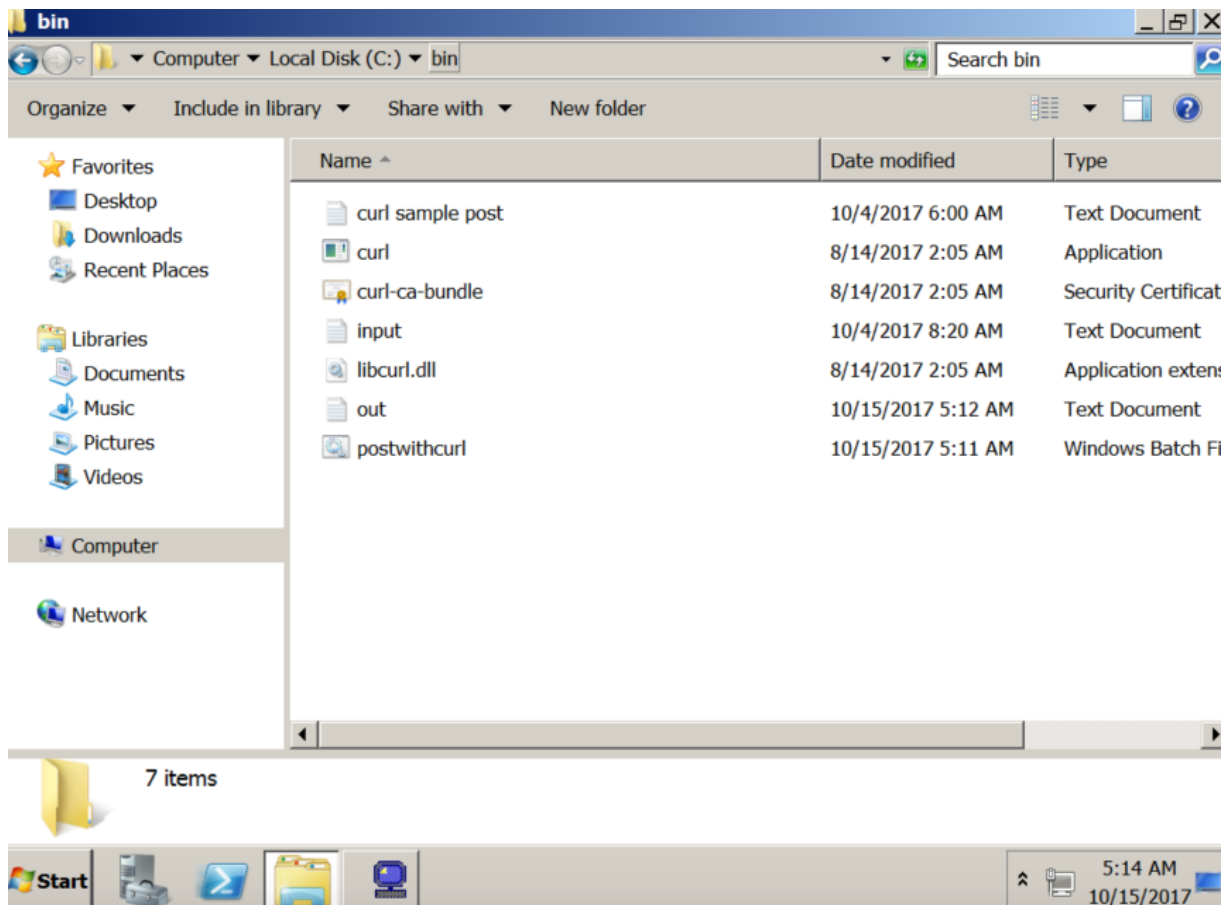
Line 1 script in **yellow** is to get the input file's content (C:\bin\input.txt in this case) to variable **Build**. This variable will be used in line 2 as **%Build%** which JSONDATA to be inserted into the database.

We save this script to C:\bin\ as **postwithcurl.bat**

As input sample JSONDATA, we save the below content to C:\bin\ as **input.txt**

```
"{\"company\":\"bank16\", \"feedback\":\"i have no comment\"}"
```





In order to execute this batch file reading the file content of input.txt and posting the info via curl to CloudantDB as a record we will use Sterling Connect:Direct process like this:

(For more information on Connect:Direct processes:

[https://www.ibm.com/support/knowledgecenter/en/CD\\_PROC\\_LANG/com.ibm.help.cdproc/overview.doc/cdproc over what is a cd process.html](https://www.ibm.com/support/knowledgecenter/en/CD_PROC_LANG/com.ibm.help.cdproc/overview.doc/cdproc%20over%20what%20is%20a%20cd%20process.html) )

```
BMTEST1 PROCESS
```

```
    SNODE=CDW2008
```

```
    PNODEID=(Administrator,passwd0rd)
```

```
    SNODEID=(Administrator,passwd0rd)
```

```
BMJOB1 RUN JOB PNODE (DSN=Windows)
```

```
    SYSOPTS="cmd(C:\bin\postwithcurl.bat >> c:\bin\out.log 2>&1)"
```

```
PEND
```

At this Connect:Direct process we use the submit job step to execute the batch file we prepared and using some scripting to generate logs about the action.

```
BMJOB1 RUN JOB PNODE (DSN=Windows)
```

```
SYSOPTS="cmd(C:\bin\postwithcurl.bat >> c:\bin\out.log 2>~&1) "
```

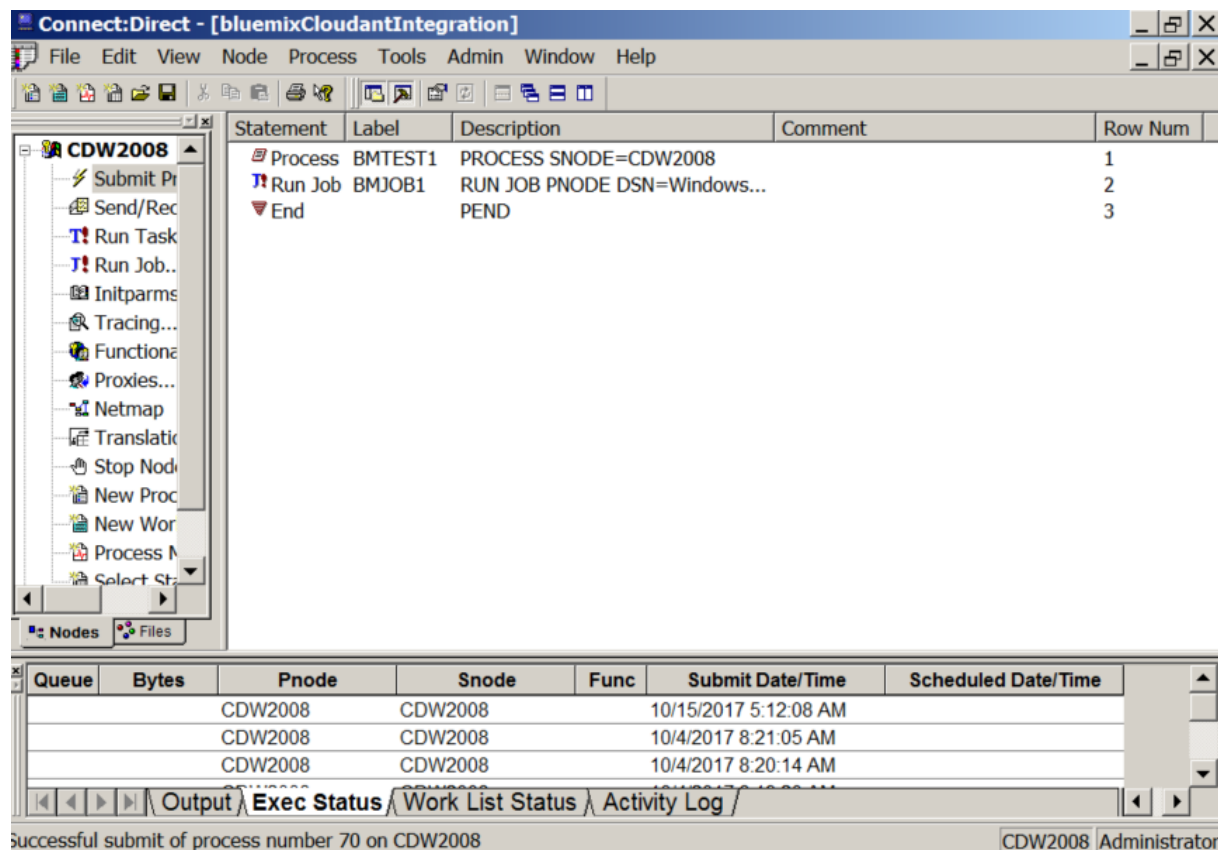
DSN=Windows means that this job is a Windows job execution

SYSOPTS="cmd()" means that it will run windows command line which is cmd.exe

At the windows command line, it will execute the command below:

```
C:\bin\postwithcurl.bat >> c:\bin\out.log 2>~&1
```

Which means executing the batch file and appends the outputs to C:\bin\out.log file



you can right click and click the "Edit/View Text" to see the Connect:Direct process source code

**Connect:Direct - [bluemixCloudantIntegration]**

Statement	Label	Description	Comment	Row Num
Process	BMTEST1	PROCESS SNODE=CDW2008		1
Run Job	BMJOB1	RUN JOB PNODE DSN=Windows...		2
End		PEND		3

Queue	Bytes	Pnode	Snode	Func	Submit Date/Time	Scheduled Date/Time
CDW2008		CDW2008	CDW2008		10/15/2017 5:12:08 AM	
CDW2008		CDW2008	CDW2008		10/4/2017 8:21:05 AM	
CDW2008		CDW2008	CDW2008		10/4/2017 8:20:14 AM	

For Help, press F1

CDW2008 Administrator

**Connect:Direct - [bluemixCloudantIntegration]**

```

/*BEGIN_REQUESTER_COMMENTS
$PNODE$="CDW2008" $PNODE_OS$="Windows"
$SNODE$="CDW2008" $SNODE_OS$="Windows"
$OPTIONS$="WDOS"
END_REQUESTER_COMMENTS*/

BMTEST1 PROCESS
  SNODE=CDW2008
  PNODEID=(Administrator,passwd)
  SNODEID=(Administrator,passwd)

BMJOB1 RUN JOB PNODE (DSN=Windows)
  SYSOPTS="cmd(C:\bin\postwithcurl.bat >> c:\bin\out.log 2>&1)"

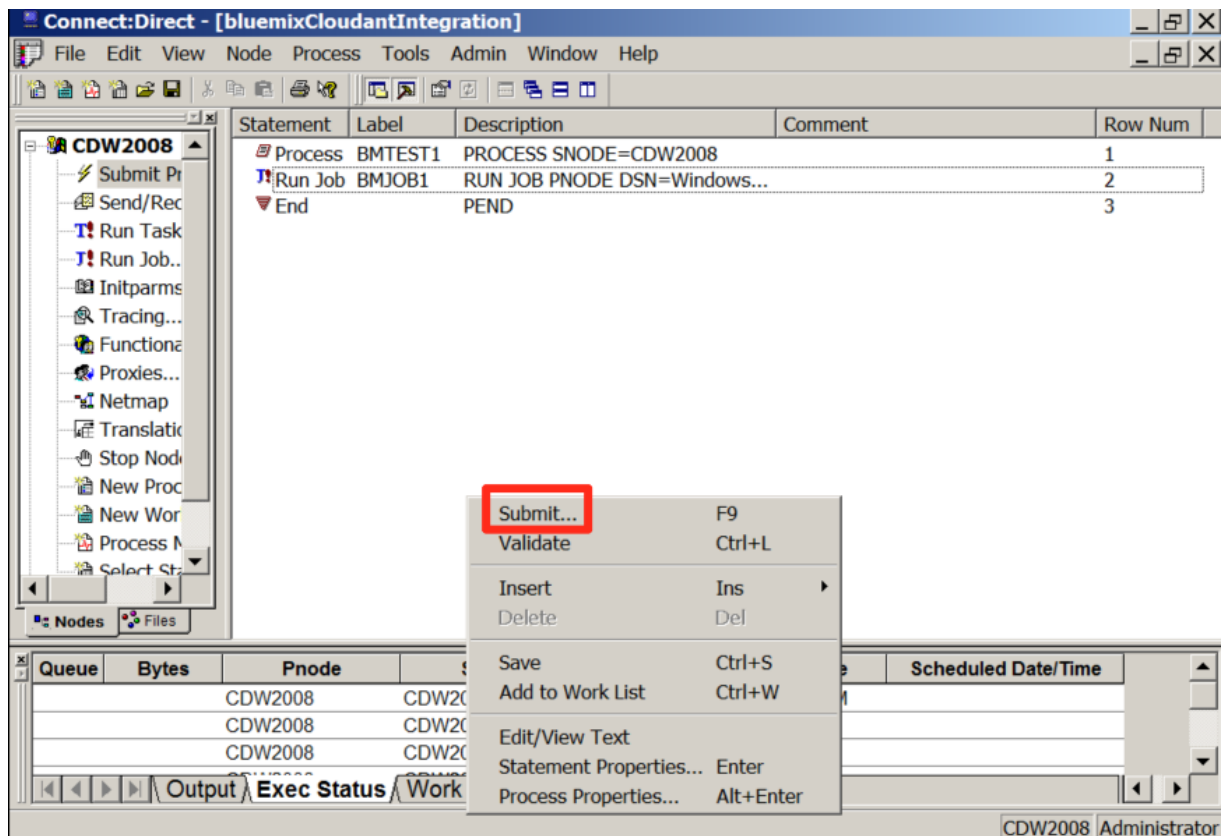
PEND
  
```

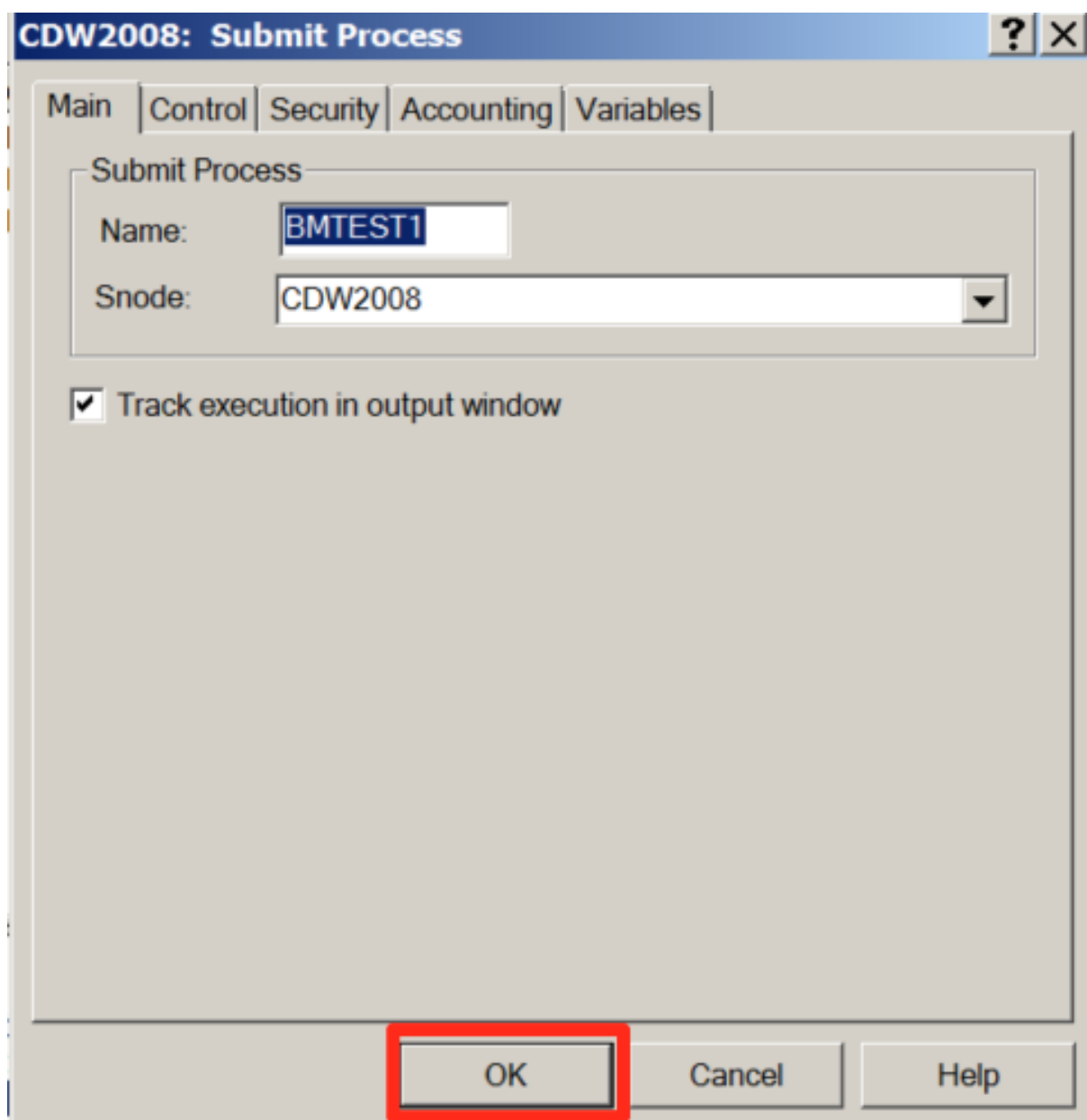
Queue	Bytes	Pnode	Snode	Func	Submit Date/Time	Scheduled Date/Time
CDW2008		CDW2008	CDW2008		10/15/2017 5:12:08 AM	
CDW2008		CDW2008	CDW2008		10/4/2017 8:21:05 AM	
CDW2008		CDW2008	CDW2008		10/4/2017 8:20:14 AM	

Ln 1, Col 1

OVR CDW2008 Administrator

We will submit this process via right click to process and click the "Submit Process" menu item





The image shows a Windows-style dialog box titled "CDW2008: Submit Process". It has a blue title bar with a question mark icon and a close button (X). Below the title bar are five tabs: "Main", "Control", "Security", "Accounting", and "Variables". The "Main" tab is selected. Inside the dialog, there is a section titled "Submit Process" which contains two input fields: "Name:" with the text "BMTEST1" and "Snode:" with a dropdown menu showing "CDW2008". Below these fields is a checkbox labeled "Track execution in output window" which is checked. At the bottom of the dialog are three buttons: "OK", "Cancel", and "Help". The "OK" button is highlighted with a red rectangular border.

In order to see what was the outcome of the process you can right click the submitted process and click "View Details":

**Connect:Direct - [bluemixCloudantIntegration]**

File Edit View Node Process Tools Admin Window Help

CDW2008

- Submit Pr
- Send/Rec
- Run Task
- Run Job..
- Initparms
- Tracing...
- Functiona
- Proxies...
- Netmap
- Translati
- Stop Node
- New Proc
- New Wor
- Process M
- Select St

Statement	Label	Description	Comment	Row Num
Process	BMTEST1	PROCESS SNODE=CDW2008		1
Run Job	BMJOB1	RUN JOB PNODE DSN=Windows...		2
End		PEND		3

View Details...  
Remove Row  
Clear Exec Status  
Create Monitor  
Allow Docking  
Hide  
Float In Main Window

Queue	Bytes	Pnode	Submit Date/Time	Scheduled Date/Time
		CDW2008	10/15/2017 5:12:08 AM	
		CDW2008	10/4/2017 8:21:05 AM	
		CDW2008	10/4/2017 8:20:14 AM	

Output Exec Status Work List Status Activity Log

**Process Execution Statistics**

Process Name: BMTEST1 Number: 70 Submit Node: CDW2008

	Log Date/Time	RecID	Rec Cat	CCode	Msg ID	
3	10/15/2017 5:12:08 AM	PSTR	CAPR	0	LSMG200I	Process number 70 (name BMTES
4	10/15/2017 5:12:08 AM	PSTR	CAPR	0	LSMG200I	Process number 70 (name BMTES
5	10/15/2017 5:12:08 AM	LSST	CAPR	0		
6	10/15/2017 5:12:08 AM	PSSED	CAPR	0	LSMG400I	A user RUNJOB step completed su
7	10/15/2017 5:12:08 AM	RJED	CAPR	0	LSMG400I	A user RUNJOB step completed su
8	10/15/2017 5:12:08 AM	PRED	CAPR	0	LSMG252I	A user process has completed suc
9	10/15/2017 5:12:08 AM	PRED	CAPR	0	LSMG252I	A user process has completed suc

Details:

Attribute	Value
Message ID	LCCA013I
Message Text	The submit of the process succeeded.
Message Data	
Process Name	BMTEST1
Process Number	70
Completion Code	0
Feedback	0
Log Date/Time	10/15/2017 5:12:08 AM

Cancel Help

It is important check if all the steps completed successfully.

Now we can check the application logs we had scripted in out.log file content:

```
C:\Program Files (x86)\Sterling Commerce\Connect Direct v4.7.0\Server\Program>for
/F "delims=" %x in (C:\bin\input.txt) do set Build=%x

C:\Program Files (x86)\Sterling Commerce\Connect Direct v4.7.0\Server\Program>set
Build="{\"company\": \"bank16\", \"feedback\": \"i have no comment\"}"

C:\Program Files (x86)\Sterling Commerce\Connect Direct
v4.7.0\Server\Program>C:\bin\curl --user APIKEY:APIKEYPASSWORD -X POST -H "Content-
Type: application/json" -d "{\"company\": \"bank16\", \"feedback\": \"i have no
comment\"}" https://YOURCLOUDANTINSTANCEID-bluemix.cloudant.com/connectdirectdata/
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
0 0 0 0 0 0 0 0 --:--:-- --:--:-- --:--:-- 0100
146 100 95 100 51 95 51 0:00:01 --:--:-- 0:00:01 166100 146
100 95 100 51 95 51 0:00:01 --:--:-- 0:00:01
166{"ok":true,"id":"61690d7761073fd5d788abb459dcb14d","rev":"1-
8fa137159845b97eff131ced1f2f8c22"}
```

As you can see the script read the input.txt file content which was:

```
"{\"company\": \"bank16\", \"feedback\": \"i have no comment\"}"
```

and done a REST API post to Cloudant DB and the REST API response was successful, it returned the record item ID:

```
{"ok":true,"id":"61690d7761073fd5d788abb459dcb14d","rev":"1-
8fa137159845b97eff131ced1f2f8c22"}
```

When we check it out the relevant record from the CloudantDB the record is there:

The screenshot shows the IBM Cloudant web interface. On the left is a navigation sidebar with options: Monitoring, Databases (selected), Replication, Analytics, Active Tasks, Account, Support, and Documentation. The main area displays the details for a database named 'connectdirectdata' and a specific record with ID '61690d7761073fd5d788abb459dcb14d'. The record is shown in JSON format: 

```
{
  "_id": "61690d7761073fd5d788abb459dcb14d",
  "_rev": "1-8fa137159845b97eff131ced1f2f8c22",
  "company": "bank16",
  "feedback": "i have no comment"
}
```

. Above the JSON, there are buttons for 'Save Changes', 'Cancel', 'Upload Attachment', 'Clone Document', and 'Delete'. The top right corner shows a JSON icon and a notification bell. The bottom of the sidebar shows the IBM Cloudant logo and a 'Log Out' button with a session ID.

In summary; this exercise shows that IBM Sterling Connect:Direct could be used to integrate with public, hybrid or private cloud environments easily and it is possible to send file content/record rows/items etc. to databases/application with REST APIs. This also proves that it could be used in cognitive solutions.