

Explore the new features in IBM Integration Bus version 10.0.0.10

BenThompsonIBM

Published on September 11, 2017 / Updated on September 24, 2019

We're very pleased to announce the availability of [IBM Integration Bus v10.0.0.10](#). Fix pack releases, published every quarter, provide both regular maintenance for the product, and also new functional content. This blog post summarises all the latest and greatest capabilities:

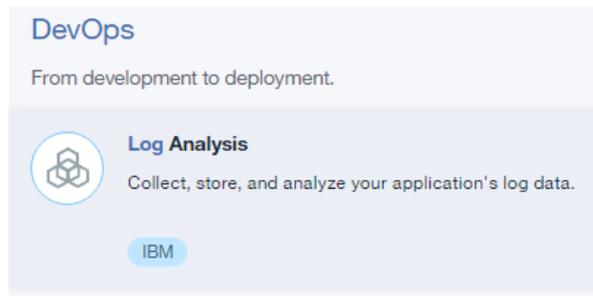


- Send IIB Logging and Statistics data to the new Bluemix Log Analysis Service
- Send Resource Statistics data to Bluemix
- Write Resource Statistics data to the filesystem
- Import of Swagger documents containing recursive references
- New Toolkit option to export a remote node's connection information as a .broker file
- New CPU benchmark processing command options

Expand the sections below to find out more!

Send IIB Logging and Statistics data to the new Bluemix Log Analysis Service

For a while now, IIB installations have been able to send operational information ([system logs](#) and [statistical performance information](#)) to a centralised cloud-hosted dashboard in the IBM Bluemix Logmet service. IBM has recently enhanced the logging services available in Bluemix and you will now find a new Log Analysis tile in the Bluemix catalog.



This tile lets you create an instance of the Log Analysis service which aggregates your application and environment logs, retains log data for as long as you need it, and provides searching capabilities enabling you to quickly detect and troubleshoot issues. You can choose the amount of logs that you can search per day. Different plans are available that you can use to search up to 500MB, 2GB, 5GB, and 10GB of logs per day. The Lite plan, which is free of charge allows searching up to the first 500MB of logs per day for the last 3 days. You can [read more about the details of this new service](#) and there is a page in the IIB Knowledge Center providing [an overview of reporting logging and statistics data to the IBM Cloud Log Analysis service](#).

Send Resource Statistics data to Bluemix

As mentioned above, previous v10 fix packs have enabled integration servers to send logging information and message flow statistics to IBM's Bluemix cloud platform. This latest fixpack v10.0.0.10 adds a new capability to also send resource statistics there too! Resource statistics are collected by an integration node to record performance and operating details of resources that are used by integration servers. As a system administrator, you can use the resource statistics to ensure that your systems are using the available resources in the most efficient manner. By monitoring systems and analyzing statistical trends, you can keep system resource usage within boundaries that you consider acceptable, and help to pre-empt situations where system resources are overstretched and might become unavailable.

[Resource statistics can be applied to multiple resource types](#) and you can specify the output formats and destinations of the data. Statistics can be emitted in XML, JSON, SMF, user trace, CSV, Log Analysis service in IBM® Bluemix™, and any combination of these output formats. The Knowledge Center has more details about [configuring an integration server to send logging and statistics data to a Log Analysis service in IBM Bluemix](#).

For example, to send resource statistics to the Log Analysis service use the following commands (with fix pack 10 the combination of r,o,a,u,p flags on the mqsichangebluemixreporting command will default to selecting the new Log Analysis service ingestion point):

```
mqsicreatebroker TESTNODE
mqsistart TESTNODE
mqsicreateexecutiongroup TESTNODE -e default
mqsichangebluemixreporting TESTNODE -e default -l active -r us-south -o bthomps@uk.ibm.com -a
IIBSpace -u bthomps@uk.ibm.com -p password
mqsichangeresourcestats TESTNODE -e default -c active
mqsichangeproperties TESTNODE -e default -o ComIbmResourceStatsManager -n bluemixReporting -v active
mqsistop TESTNODE
mqsistart TESTNODE
```

Alternatively if you are not yet using the new Log Analysis service and just want to use the old Bluemix Logmet service (the technique below of using the x flag on the mqsichangebluemixreporting command will cause IIB to use the older Bluemix Logmet service ingestion point):

```
mqsicreatebroker TESTNODE
mqsistart TESTNODE
mqsicreateexecutiongroup TESTNODE -e default
curl -k -XPOST -d "user=
bthomps@uk.ibm.com&passwd=password&space=IIBSpace&organization=bthomps@uk.ibm.com"
https://logmet.ng.bluemix.net/login
```

... take the values for logging_token and space_id from the reply and use them for parameters i and t in the next command below ...

```
mqsichangebluemixreporting TESTNODE -e default -l active -x logs.opvis.bluemix.net -i da7a0a08-30bc-
4184-8906-60d535ee5f28 -t wnZVATokYBV2
mqsichangeresourcestats TESTNODE -e default -c active
mqsichangeproperties TESTNODE -e default -o ComIbmResourceStatsManager -n bluemixReporting -v active
mqsistop TESTNODE
mqsistart TESTNODE
```

Here is a screenshot from Bluemix showing that the IIB resource statistics data has been sent in using a JSON payload:

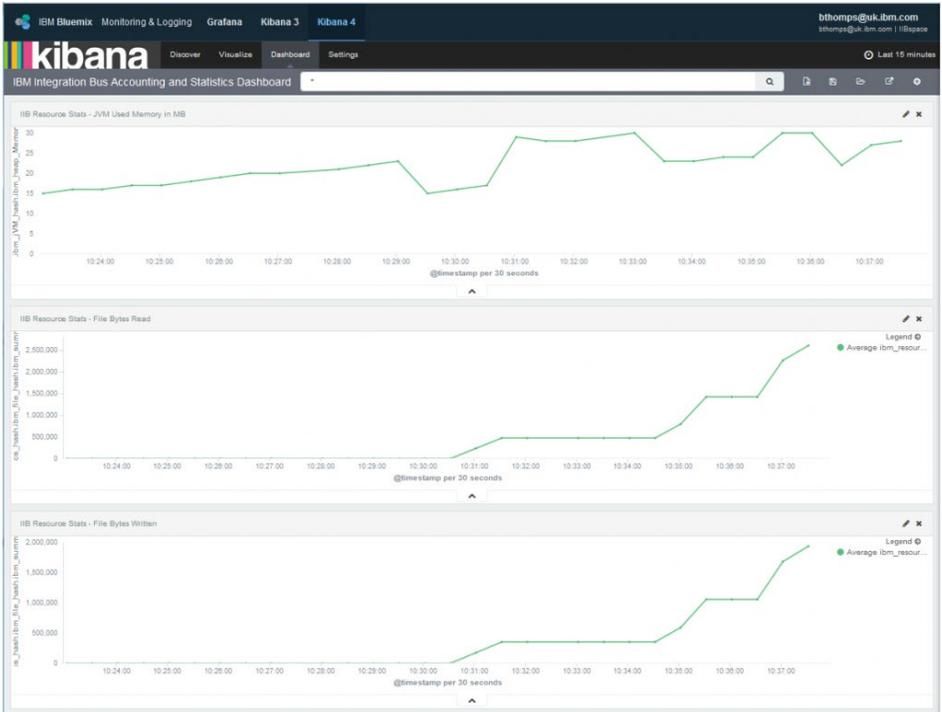
```

Kibana 3  Kibana 4
Time  _source
September 11th 2017, 09:19:27.906  message:  @version: 10.0.0.10 @timestamp: September 11th 2017,
bm_resourceStatistics_hash.ibm_collectionStartTime: 09:18:54 ibm_
11 ibm_resourceStatistics_hash.ibm_endTime: 09:19:28 ibm_resour
fault ibm_resourceStatistics_hash.ibm_brokerName: TESTNODE ibm_
istics hash.ibm JVM hash.ibm summary hash.ibm usedMemoryInMB int

Table  JSON
1  {
2  "_index": "logstash-2017.09.11",
3  "_type": "logs",
4  "_id": "AVShBaQ1U-xF-Cv138Vx",
5  "_score": null,
6  "_source": {
7  "message": "",
8  "@version": "10.0.0.10",
9  "@timestamp": "2017-09-11T08:19:27.906Z",
10 "host": "IBM425-R9E9V8K",
11 "ALCH_TENANT_ID": "da9a8a08-30bc-4184-8896-60d535ee5f28",
12 "ibm_resourceStatistics_hash": {
13   "ibm_collectionStartDate": "2017-09-11",
14   "ibm_collectionStartTime": "09:18:54",
15   "ibm_startDate": "2017-09-11",
16   "ibm_startTime": "09:18:54",
17   "ibm_endDate": "2017-09-11",
18   "ibm_endTime": "09:19:28",
19   "ibm_timezone": "Europe/London",
20   "ibm_type": "iib_resourcestats",
21   "ibm_module": "TESTNODE.default",
22   "ibm_brokerName": "TESTNODE",
23   "ibm_serverName": "default",
24   "ibm_JVM_hash": {
25     "ibm_summary_hash": {
26       "ibm_initialMemoryInMB_int": 32,
27       "ibm_usedMemoryInMB_int": 47,
28       "ibm_committedMemoryInMB_int": 318,
29       "ibm_maxMemoryInMB_int": -1,
30       "ibm_cumulativeGCTimeInSeconds_int": 7,
31       "ibm_cumulativeNumberOfGCCollections_int": 27
32     },
33     "ibm_heap_Memory_hash": {
34       "ibm_initialMemoryInMB_int": 32,
35       "ibm_usedMemoryInMB_int": 17,
36       "ibm_committedMemoryInMB_int": 34,
37       "ibm_maxMemoryInMB_int": 256
38     },
39     "ibm_non-Heap_Memory_hash": {
40       "ibm_initialMemoryInMB_int": 0

```

Once you have statistics being pushed to Bluemix, you can configure Kibana charts to graphically represent what is happening to IIB, such as the example below showing resource statistics such as JVM used heap memory, bytes read from files and bytes written to files, all plotted against time:



Write Resource Statistics data to the filesystem

Fix pack 10 also introduces the possibility of writing resource statistics directly to the filesystem. When this option is enabled each resource manager will write a separate file containing it's own resource stats. There are size limits in place for the size of each file and once these limits are reached a new file is be created. There is also a limit to the number of files that can be written for each resource manager and when this limit is reached the resource manager will write separate log files in a circular fashion.

By default resource stats files will be written to a subdirectory of the IIB workpath: `MQSI_WORKPATH/common/resourceStats`. To specify an alternative location you can use the options `-n filePath -v write/to/this/directory`. If you choose to do this then the referenced directory must already exist. The filename for each resource manager's files will be in the form `ResourceStats_<IntegrationNodeName>_<IntegrationServerName>_<ResourceManagerName>.txt`.

The example commands below show how to enable writing resource statistics to a file for an integration node named TESTNODE and its server named default:

```
mqsichangeresourcestats TESTNODE -e default -c active
```

```
mqsichangeproperties TESTNODE -e default -o ComIbmResourceStatsManager -n fileReporting -v active
```

You can modify the default properties used when logging resource statistics to files by altering the following attributes on the `ComIbmResourceStatsManager`:

`numberOfFiles` – default 4 – The number of files to roll data across

`sizeOfFiles` – default 25 – The maximum size in Mb of a single log file

`filePath` – default "" – The name of a directory where resource statistics files should be written.

Here is an example output file:

```
BrokerName, BrokerUUID, ExecutionGroupLabel, ExecutionGroupUUID, CollectionTimestamp,
StartTimestamp, EndTimeStamp, summary_InitialMemoryInMB, summary_UsedMemoryInMB, summary_CommittedMemoryInMB,
summary_MaxMemoryInMB, summary_CumulativeGCTimeInSeconds, summary_CumulativeNumberOfGCCollections,
Heap Memory_InitialMemoryInMB, Heap Memory_UsedMemoryInMB, Heap Memory_CommittedMemoryInMB,
Heap Memory_MaxMemoryInMB, Non-Heap Memory_InitialMemoryInMB, Non-Heap Memory_UsedMemoryInMB,
Non-Heap Memory_CommittedMemoryInMB, Non-Heap Memory_MaxMemoryInMB,
Garbage Collection - Copy_CumulativeGCTimeInSeconds, Garbage Collection - Copy_CumulativeNumberOfGCCollections,
Garbage Collection - MarkSweepCompact_CumulativeGCTimeInSeconds,
Garbage Collection - MarkSweepCompact_CumulativeNumberOfGCCollections
```

```
TESTNODE_100010, 4f1b8328-dd03-41cb-acd4-8154401ddf83, default, 944b7f61-6da9-49c0-b9fb-7f961b0388cd,
2017-09-10 23:47:01.000,2017-09-10 23:48:43.000,2017-09-10 23:49:03.000,
32,53,317,-1,0,24,32,25,35,256,0,28,282,-1,0,24,0,0
```

```
TESTNODE_100010, 4f1b8328-dd03-41cb-acd4-8154401ddf83, default, 944b7f61-6da9-49c0-b9fb-7f961b0388cd,
2017-09-10 23:47:01.000,2017-09-10 23:49:03.000,2017-09-10 23:49:23.000,
32,53,317,-1,0,24,32,25,35,256,0,28,282,-1,0,24,0,0
```

```
TESTNODE_100010, 4f1b8328-dd03-41cb-acd4-8154401ddf83, default, 944b7f61-6da9-49c0-b9fb-7f961b0388cd,
2017-09-10 23:47:01.000,2017-09-10 23:49:23.000,2017-09-10 23:49:43.000,
32,53,317,-1,0,24,32,25,35,256,0,28,282,-1,0,24,0,0
```

Import of Swagger documents containing recursive references

The IBM Integration Bus Toolkit lets you easily import or create from scratch Swagger documents which provide metadata describing RESTful service interfaces with which IIB can interact. REST support is offered both in to and out of IIB:

An IIB REST API artifact exposes a REST interface which can be invoked by other applications

An IIB REST Request node can be dragged and dropped in to a message flow to invoke an external REST API.

There are some restrictions on the format of Swagger documents that can be used when creating REST APIs, which are documented [here](#). One of these restrictions is that the JSON schema "\$ref" keyword can only be used in conjunction with internal file references. Until fix pack 10, recursive self references were rejected by the Toolkit. The reason for this restriction has historically been that the Graphical Data Mapping editor cannot be used to dynamically discover the depth of recursion. The restriction on recursive self reference imports has now been lifted in order to help those users wishing to import Swagger documents containing recursive references, but who have no intention of

using graphical maps, or who are happy to use mappings only with a Move transform from input to output.

New Toolkit option to export a remote node's connection information as a .broker file

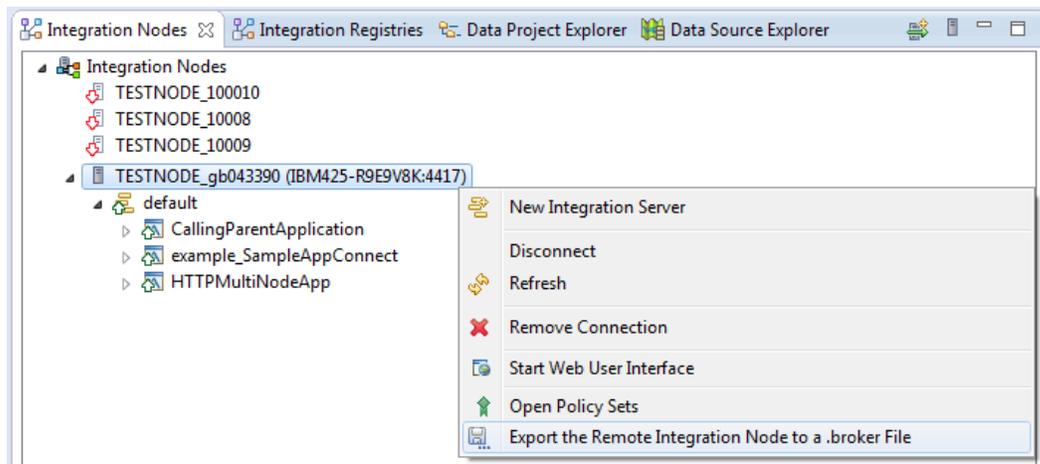
You can connect the Integration Bus Toolkit to nodes defined on the same local machine or nodes defined on remote machines. This is particularly important for large enterprise environments where you may have IIB footprints spanning multiple independent environments e.g. for development, functional test, performance test, quality assurance and production. When connecting to a remote integration node, you can define a connection file, commonly given a “.broker” extension, which holds the following basic connection properties:

Hostname

Port

Integration node name

These connection files can also be used in conjunction with about 20 different IIB commands, [as documented here](#). From fix pack 10, to make it easier to create a connection file of this type, you can now right click an integration node in your toolkit, and use the new menu option “Export the Remote Integration Node to a .broker File, as shown in the screenshot below:



New CPU benchmark processing command options

After installing IIB and creating a node, users will often turn to the `mqsicvp` command which provides a simple method of performing a quick verification test on an integration node to check aspects of its configuration, such as verifying the ODBC connections which it can make. Fix pack 10 extends this command further to enable it to perform some benchmark processing. The benchmark tests the machine's CPU performance by calculating a recursive, arithmetic sequence on one or more threads. For the duration of the benchmarking process, each thread that is used typically consumes an entire logical CPU on the system. Consequently, the optimal time to run the command is during a period of scheduled downtime, such as just after configuring a new environment for example, so that the CPU usage does not impact running production workloads. For example to run a benchmark that includes two threads, five iterations, and calculates 40 steps in the sequence, you would use this command:

```
mqsicvp TESTNODE -b benchmark -o javarecursion,threads=2,iterations=5,sequence=40
```

The output from the command will be similar in format to the following example:

```
mqsicvp TESTNODE -b benchmark -o javarecursion,threads=2,iterations=5,sequence=40
Running 2 threads
ThreadId:23 - Calculating sequence to number 40 5 times
ThreadId:24 - Calculating sequence to number 40 5 times
Took 3 seconds to run
Total CPU time: 7014ms, for 10 calculations

Rating Value:285714
```

Average Core Value:142857
BIP8071I: Successful command completion.

TAGS IBM-INTEGRATION-BUS, IIB, IIBV10, WHATS-NEW, FIX PACK V10.0.0.10

BenThompsonIBM

2 comments on "Explore the new features in IBM Integration Bus version 10.0.0.10"

Raj September 11, 2017

Hi Ben,

Do you have the release date for fixpack 10?

Ta

[Reply \(Edit\)](#)

BenThompsonIBM September 11, 2017

Hi Raj,

Yes – Fix pack 10 was released on Friday last week ... It is available now: <http://www-01.ibm.com/support/docview.wss?uid=swg24043943>

Cheers,

Ben

[Reply \(Edit\)](#)