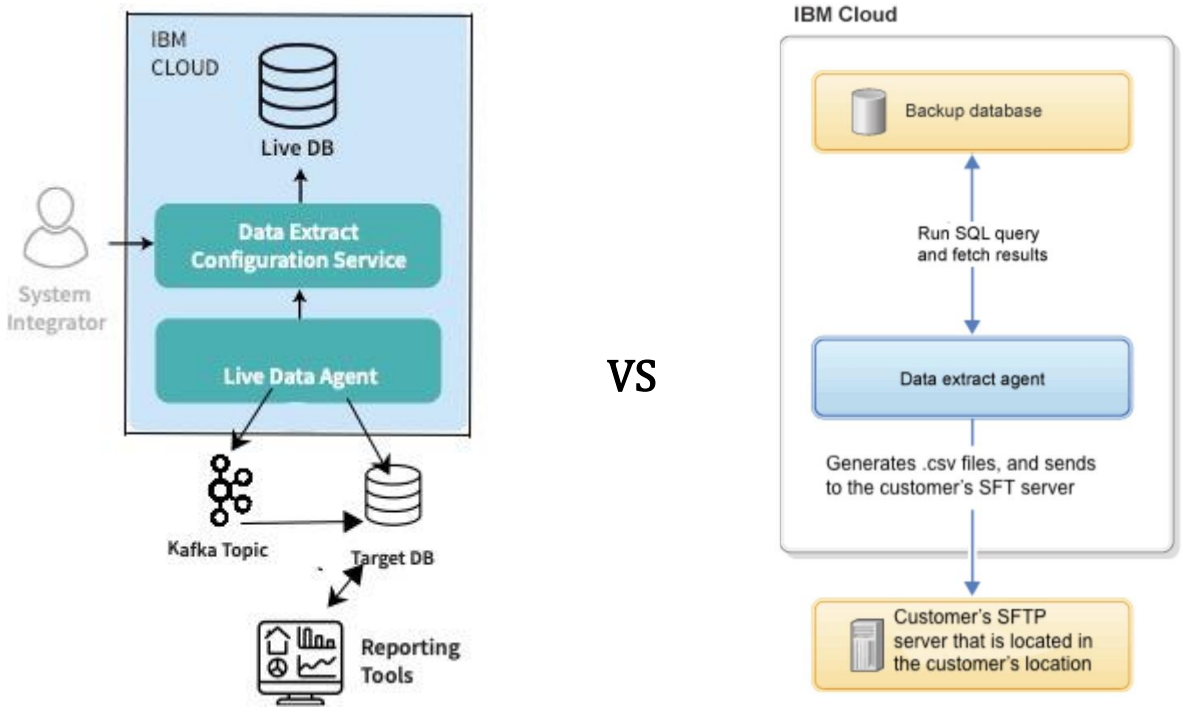
Live Data Agent™

Free your IBM OMoC production data to your Business Users

Since the release of the IBM Order Management on Cloud (OMoC) customers have been challenged to access their production data in near-real time and have faced the frustration of having to use the only solution made available by IBM that moved table-level entity data over FTP to CSV flat files on an hourly basis at best. Consuming data in this form is not only error-prone, but cumbersome and makes it extremely difficult to expose that data to their business users who need to glean important insights from that data trapped in their IBM OMoC production system.

Figure 1 - IBM Live Data Agent (Replaces Data Extract Agent for OMoC)



1. The Obstacles

In the world of IBM Order Management on Cloud (OMoC), customers are ***not*** granted direct access to their production database via JDBC or any other open standard. This creates a major obstacle for customers looking to use their enterprise reporting tools to connect to their order repository to run reports from in near real time. Although IBM provides an agent for customers that will extract data from select tables, the agent does so by creating FTP files and transferring them to the customer site leaving it up to the customer to convert those files into something their reporting tools can generally use. Furthermore, there is no way to schedule these FTP extracts any more frequently than one hour at a time, which means, at best, the data is more

than an hour old, while further, significant efforts are required to adjust the routines processing the FTP files to adapt to any changes to the data extract configurations. This is just plain difficult and customers often wonder why they can't have their own data extracted directly into their own shadow copy for use in analytics and reporting and on a near-real time basis. **Well now that's possible with Live Data Agent for IBM OMoC!**

1.1 A Better solution to Access Customer Data

It's apparent that a better solution is needed for IBM OMoC customers to access their production data in near-real time and the ideal solution would have the following benefits and capabilities.

1. The solution is a turn-key replacement of the IBM Data Extract Agent that currently uses cumbersome FTP to transfer table row/column data from the **Backup** IBM OMS system to a remote FTP file system with an agent that moves the data from the **Production** instance over a *Kafka* Topic. From that Kafka Topic, it can be easily loaded into a database of the customer's choosing (DBaaS or On-Site) using a Client-Side Kafka Consumer application (provided with the solution) that can automatically move the data on the Kafka Topic directly into a client-side shadow database. That database, in-turn will be directly accessible to your enterprise reporting tools via JDBC with not restriction on your capabilities.

2. The solution utilizes all the same configuration tables and API's that are already available to set up the IBM Data Extract Agent for FTP. This means customers can use existing IBM API's to configure what is fed to the Live Data Agent. For the most part, the documentation for that IBM Data Extract agent should still apply, disregarding the FTP-centricity aspects since FTP is not used. You can find the IBM Knowledge Center documentation for these API's here.
here:https://www.ibm.com/support/knowledgecenter/en/SSGTJF/com.ibm.help.omcloud.administer.doc/tools/c_omc_dataextract.html

3. The solution utilizes Transactional Kafka technology (Kafka is the latest in Scalable Messaging Technology and is used by IBM internally and by many others) to move the data from the IBM OMoC Production DB to a Kafka Topic. The data is delivered to the queue in Comma Delimited Format and the delivery is very fast, very reliable, and very scalable. Using Kafka Transactions, the clients can request only the committed records from the topic and can communicate to the Kafka Partition which records have been consumed. This virtually eliminates any issues with data loss and ensures only what was intended to be consumed to be consumed. It also allows for multiple consumers to access the same topic maintaining their own unique offsets into the topic by consumer group. That means Marketing can get a copy of the data in one local DB and Sales in another.

4. The solution allows customers to "**Override**" the part of the ***Live Data Agent*** that moves the data to **Kafka** in CSV format, so the customer can instead, decide to move it over their own ESB technology The default implementation of this override moves it over Kafka in CSV format and/or directly into a target database you can configure via **IBM OM properties**.

5. Since the data is typically going to be targeted for a Database vs FTP files, the solution allows the customer to configure table groups and sequence the extracted tables so they

can, for example, send the YFS_ORDER_HEADER records before YFS_ORDER_LINE records which have a one-to-many relationship and leverage foreign keys to tie the headers to the lines. This grouping should not require any change to the existing YFS_DATA_EXTR_CFG table to accomplish this.

6. The solution comes with a Client Side Server application that can move the data from the Kafka Topic to a **target** database (DBaaS or Local) via JDBC. That client application only reads the committed records from the Live Data Agent topic. It also has the ability to create the corresponding table schema on the destination database using the YFS_DATA_EXTR_CFG configurations stored and to re-create any of these tables as new columns are added or others dropped.

7. The solution extends IBM's DataExtract's "**First Run**" capabilities that is used to synch back any number of days up to the current date/time. This solution allow customers to manually or automatically trigger a **Reset** to reload the data to a given start date dictated by the table's **FirsRunExtractInDays** configuration setting. As part of this process you can configure what tables, if any, should be deleted, dropped, created, or dropped and created, or left as is in preparation for getting a new full data synch.

8. The solution provides a "**FetchLimit**" criteria parameter that restricts how much data from a table to extract in any one trigger of the agent. This allows you to throttle the agent so as to minimize the impact on performance, especially when peaks occur.

9. The solution supports a "**MinAtRestSeconds**" criteria parameter that requires any record coming from the source database be at rest for a given number of seconds before it can be extracted. This allows you to give the data a chance to settle in the database before it's extracted and is especially useful in the case of extracting order data that may go through a bunch of transactions before coming to a useful and restful status.

1.2 Conclusion

The Live Data Agent lays the groundwork for IBM OMoC customers to realize the full potential of their order management data that is today, ***trapped*** inside the four walls of the IBM Cloud data centers. Using this revolutionary Data Extract tool that is purpose built for IBM OMoC using all the standard IBM OM Agent framework, customers can finally get at their critical transaction data, in near real-time, quickly, efficiently, and with little to no impact on their production system. Once the data is extracted into the customer-owned databases, they're free to leverage the proprietary tools, or any of their other enterprise reporting frameworks to build the real-time reports..

Finally, if you're an IBM OMoC customer leveraging the IBM Cognos on Cloud solution, you now have a much more flexible alternative that will allow you to use your own internal enterprise reporting tools. Your reports will be more real-time and you can reduce the resources needed to care and feed this less than optimal reporting solution.