ORACLE®

FUSION MIDDLEWARE
MANAGED FILE TRANSFER

Managing Files for the Hybrid Cloud

Use Cases, Challenges and Requirements

ORACLE®

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Table of Contents

Overview

Fully integrated file transfer solutions are a core requirement in the new and hybrid world of cloud and on-premise enterprise. Organizations still make heavy use of files utilizing bulk, batch, and real-time updates for application integration. As cloud Software as a Service (SaaS) is introduced in the IT landscape, file integrations simply increase in both size and volume. The general approach for on-premises and partner file integrations using Managed File Transfer (MFT) is similar to other IT infrastructure integration: Plan, Scope and Deliver incrementally. This paper discussed best practices for sharing files within your enterprise and externally for partners and cloud services.

Use Cases and Requirements

This document discusses some core use cases for getting started with Oracle MFT. For most organizations, MFT is not new but rather there is a need to put together a single enterprise wide file transfer solution that scales to partners and the cloud. Changes in security and compliance requirements are driving IT organizations to replace aging or custom script based file transfer technologies. Encryption such as PGP and Secure FTP servers are still the norm, but they must utilize existing Identity Management to define users, groups and roles for remote access. Finally, the need to integrate MFT with existing integration tooling such as Oracle SOA Suite cannot be overstated. Since MFT typically does not support document transformations and other complex orchestration use cases, it is critical for MFT to act as a file gateway when used in tandem with the existing integration platform.

On Premises: Connecting Your Internal Applications

Challenges

There are many challenges organizations face when deploying seemingly simple integrations of internal applications, one of which is file handling. The main technology challenges all file transfer products should address are full security, extensibility and global file transfer visibility with special file handling capabilities. Other challenges include easily scheduling receipt or delivery of files and configuring endpoints once for maximum reusability.

Use Case Description

The use case depicted in the picture below is a Secure FTP to File system transfer that makes use of an embedded sFTP server which processes files in real time as they are uploaded to the server. An MFT designer in a web based user interface configures file transfers to include endpoint sources, targets, filters, schedules, encryption, compression, notifications etc.
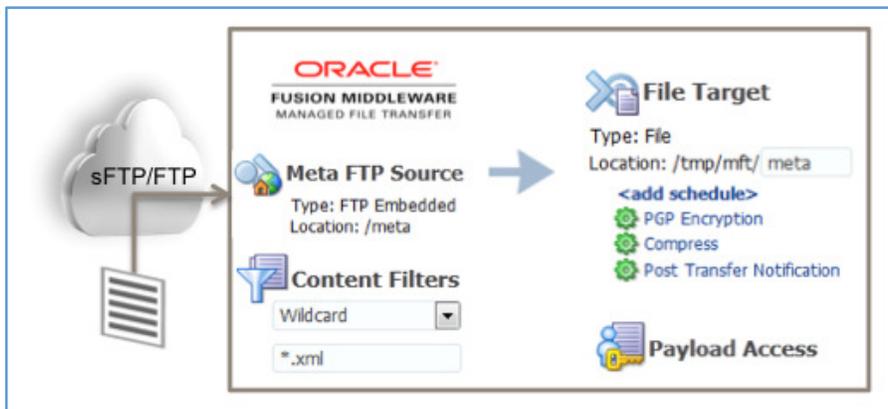
An administrator provisions the FTP user access using the standard WebLogic OPPS security layer allowing the credentials to be mapped to Oracle Identity Management backed by a database or LDAP system.
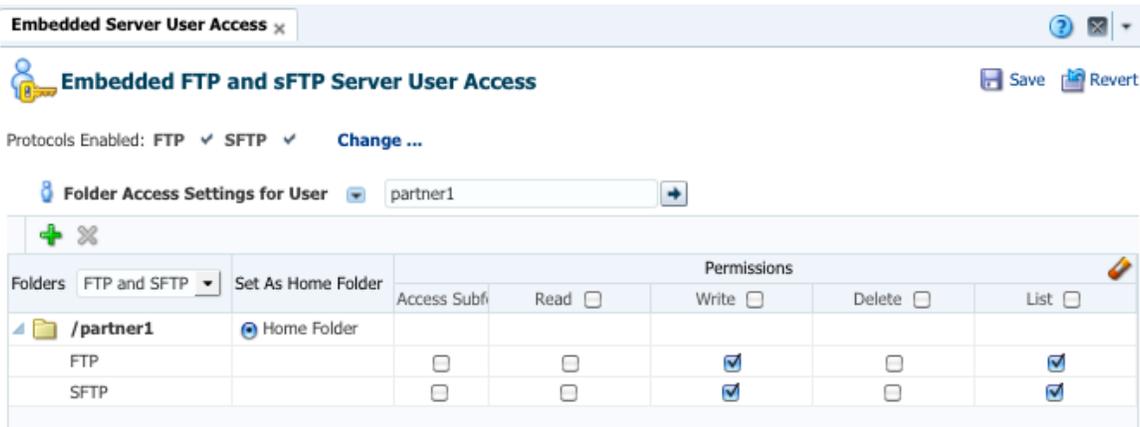

Figure 2: FTP User Access Permissions

When an FTP client "Put" operation is completed, the FTP server immediately sends an event to the MFT server that processes the file using content filters to determine what target endpoints will receive the file. MFT transfers can override the targets file system location, which promotes reusability of target endpoint configurations. Files are retained in the MFT temporary repository using schedules and optional actions such as compression, encryption and file delivery notifications that are configured for each transfer. IT departments require the ability to quickly find the status of files in the system. Monitoring dashboards enable this functionality by allowing users to search for files based on multiple criteria such as filename, file size etc. as shown below.
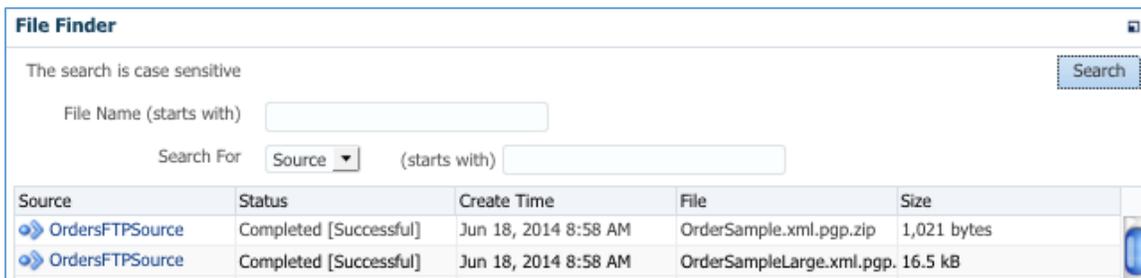

Figure 3: IT Monitoring File Finder Interface

Global visibility of file transfers requires detailed drill down status of file. Each file transfer instance has a flow report that shows a transfer flow diagram and a summary status as illustrated below.
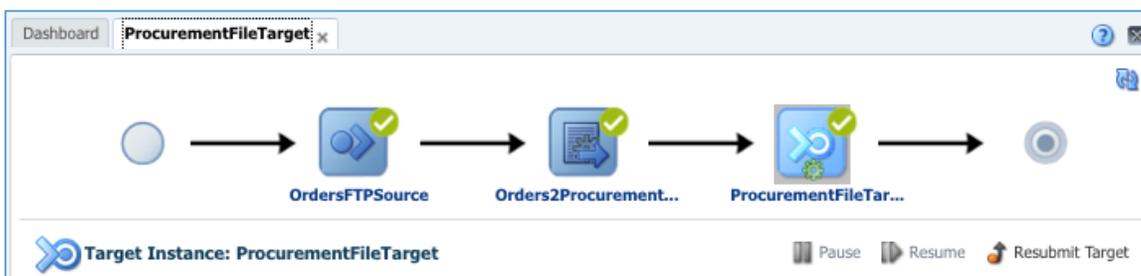
Core transfer properties such as file name, location, file size, transfer time, and processing actions such as compression and encryption are clearly viewable by clicking the source and target flow icons. Files can easily be resubmitted from all nodes by clicking the Resubmit button.

```
◢ Summary

             File Name  OrderSample8.xml
   Delivered File Name  OrderSample8.xml
             Location   /scratch/dberry/soa12/user_projects/c

          Target Name   ProcurementFileTarget
        Transfer Name   Orders2ProcurementTransfer
             Endpoint   file://localhost/tmp/mft/partner1
        Creation Time   May 20, 2014 2:17 PM
               Status   Completed [Successful]
     Bytes Transferred  2651
           Time Taken   13 milliseconds
```

Figure 5: Flow Transfer Summary

Requirements

» Role based security for file transfer repositories such as FTP or SSH Secure FTP servers

» Event based Sources for modern non polling architecture

» End to end visibility of file transfer status including detailed view and resubmission of "lost" files

» Special handling actions such as encryption, compression and notifications

» Easy to use for non technical staff

Extensibility: Use MFT's SOA Integration for Debatching of Large Files

Challenges

Large files passed as an attachment or inline in a message payload may impact performance and adversely affect the total memory footprint in a SOA environment. This is particularly true of longer running stateful or asynchronous business processes because the large file is writing to a process repository or database sometimes called a dehydration store.

Use Case Description

Partners especially in retail or finance rely heavily upon scheduled batch processing windows after business hours for updates to the their fulfillment, inventory, customer or clearinghouse applications. Messages are accumulated in a local database or file system and batched and sent to external systems for overnight processing. Files need to be protected, reliably transferred, and are typically processed in the same order they were generated. If any one item in the batch fails, then there must be logic to determine if processing of the rest of the items should continue or send notifications and stop processing until it is resolved. While MFT can often handle debatching use cases with custom java actions or use of zip files; error handling in batch file processing is a great illustration of how extending to a familiar orchestration tool such as Oracle BPEL Process Manager may be a better fit.

To accomplish this, an MFT SOA target endpoint is configured to invoke either a SOA 11g or 12c composite with a BPEL process. Delivery preferences are used to configure a reference to the file so the message does not have to include the large file in its payload. The file can optionally be sent as part of the SOA message payload by specifying a maximum size file to send inline. This gives the SOA developer the ability to reuse the same composite to process both small files and large files. MFT makes use of the embedded Oracle Enterprise Scheduler to deliver the message to the composite only during required business processing time windows.
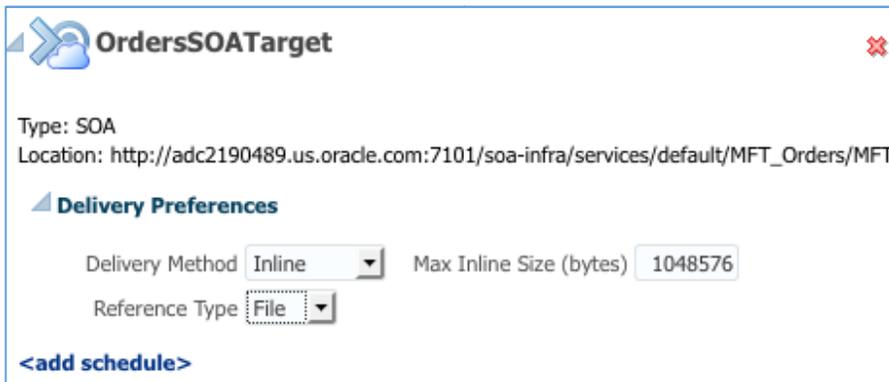
Figure 6: MFT SOA Target Definition

The composite application will expose a web service implementing the MFT WSDL contract. It also references a File adapter to dynamically read the file reference provided by MFT and it invokes an outbound order service to create the orders from the incoming request.
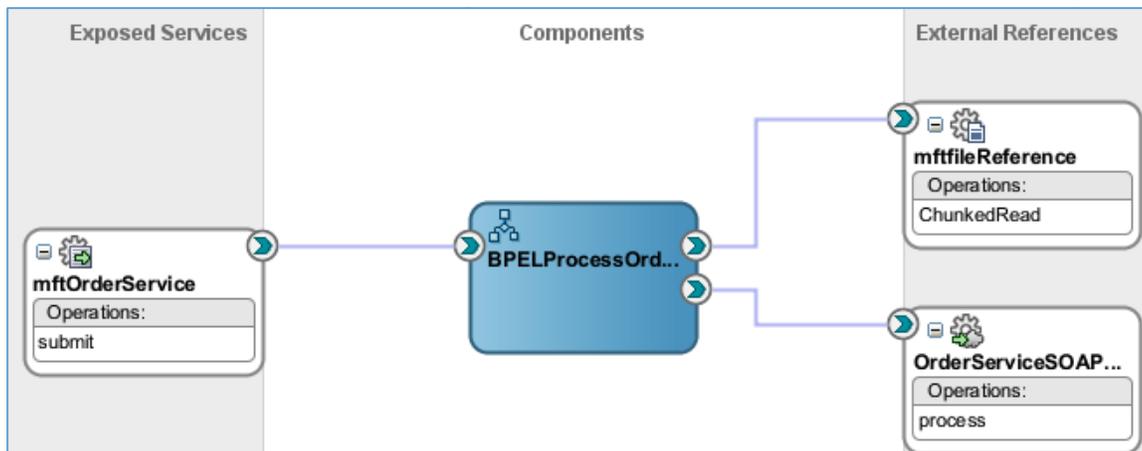

Figure 7: SOA Composite Definition

The BPEL process uses the MFT header data passed in the SOAP message to determine how and where to process the file. The MFT PayloadType attribute can be used with a BPEL 2.0 If statement to MFT support the following MFT payload types:

» FtpRefFile – reference a file from the MFT embedded FTP server
» FtpRefDir – reference to a directory in the MFT embedded FTP server
» FileRefFile – reference to a file in the shared MFT/SOA file system
» FileRefDir – reference to a directory in the shared MFT/SOA file system
» InlineBinary – inline encoded payload
» InlineXML – inline xml payload

MFT headers can be used by the BPEL process to determine whether to process the file inline (small file) or by reference (large file). The following headers information is provided by MFT in the request.

- » User who uploaded the file
- » Transferred file name and file size
- » Link URL back to the MFT console
- » The names of the configured MFT artifacts: Transfer, Source and Target

The BPEL process uses the If statement to configure both an inline and a Sync Chucked Read operation to dynamically retrieve the file based on the incoming payload MFT. A sample BPEL process flow is shown below.
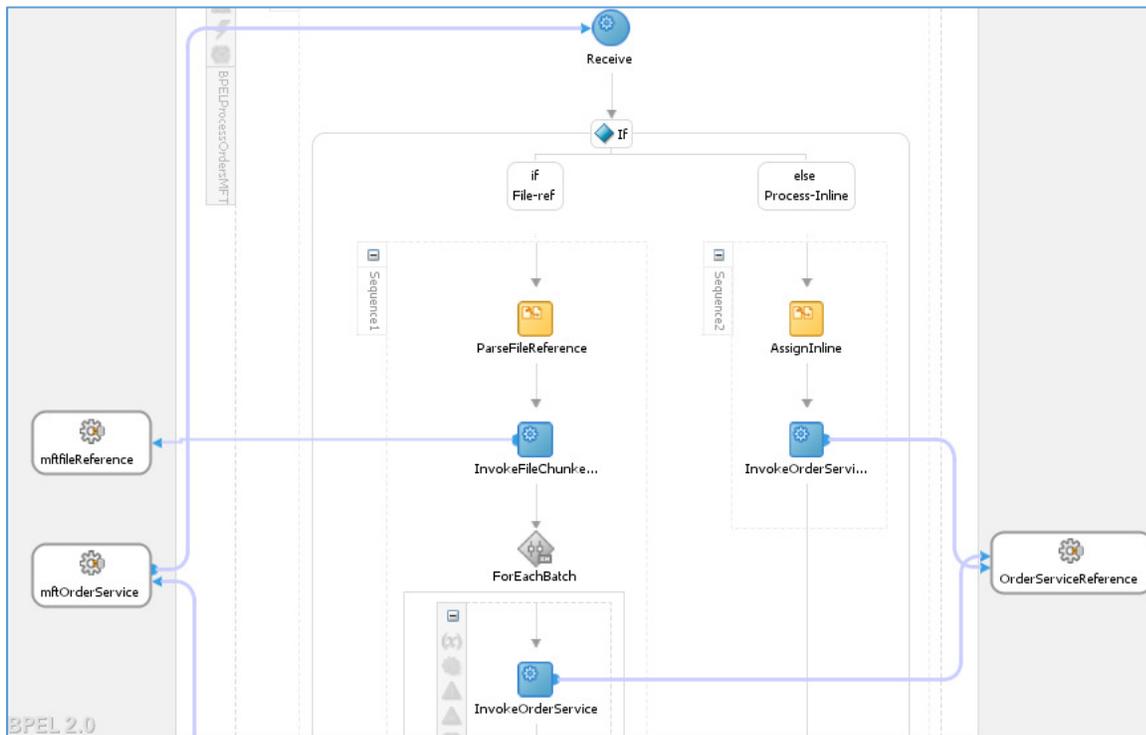

Figure 8: Example MFT Orders BPEL Process Flow

To promote global end to end visibility of file transfers, MFT makes use of the SOA global flow id to cross link consoles. This allows SOA admins or helpdesk personnel working in SOA Enterprise Manager to easily research file history for possible resubmit of lost or corrupted documents.

Figure 9: MFT and SOA EM Monitor Console Integration

Requirements

» Tight integration with MFT and orchestration products such as SOA Suite BPEL Processor

» MFT file processing capabilities with built in large file processing architecture including pass by reference, zip files and extensibility

» Built in MFT scheduler for deferring batch processing based on business schedules

» Cross linking of runtime consoles to enable deep dive global visibility

» Mature orchestration layer for dynamically opening and iteratively processing of de-batching large files

Partners: Extending On Premises Applications to B2B Partners

Challenges

Extending file transfers to partners can be a challenge for many reasons. Each partner may have different file formats, business calendars and schedules, transfer notification channels, batch processing needs, service levels and contractual agreements. To fully satisfy this myriad of requirements requires a full integration solution beyond the core MFT product that incudes a B2B solution and integration orchestration layer such as Oracle B2B and BPEL Process Manager.

Use Case Description

Consider a partner who uses a B2B message exchange such as Oracle B2B to submit EDI documents for order processing and requires, according to the terms of their partner agreement, both guaranteed delivery of service and validation of origin or nonrepudiation.
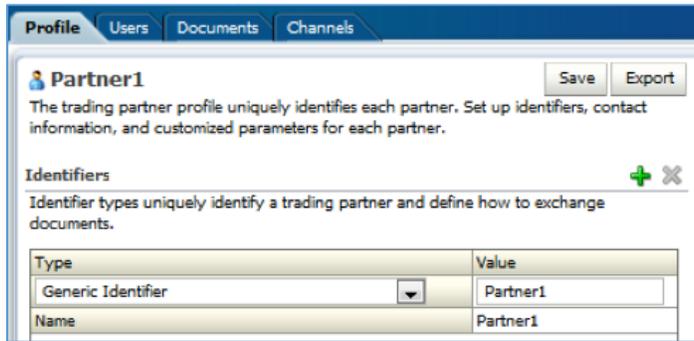
Figure 10: B2B Partner Agreement

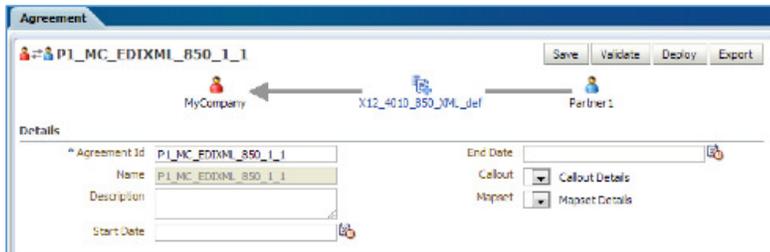The B2B system receives the document and translates it from the EDI format to standard XML.

Figure 11: B2B EDI Definition

In this use case, the B2B system uses MFT and its embedded Secure FTP server as a file virtualization layer so the B2B partner endpoint channels can be configured separate from the B2B partner definition. Additionally, an MFT Secure FTP proxy service residing in the green zone can be used to ensure multiple levels of security between the external and internal systems.
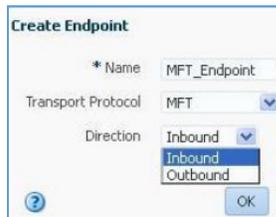

Figure 12: B2B MFT Listener Channel Definition

An MFT Endpoint is created to handle file transfer requests to and from B2B. MFT enforces enterprise wide security encryption policy to ensure safe delivery of the EDI document.
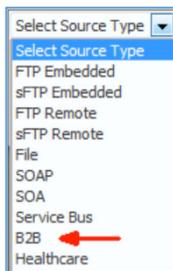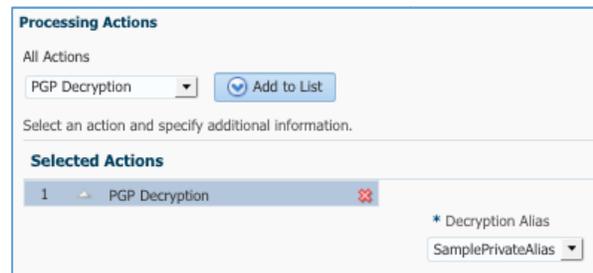

Figure 13: MFT Creates B2B Endpoint

Figure 14: MFT Decryption Option is selected

Requirements

» Leverage MFT Secure FTP servers to virtualize file endpoints for B2B messages

» Utilize MFT built-in PGP encryption and Decryption to implement organizational document security policies

Cloud: Extending On Premises Applications to Cloud

Challenges

Cloud Software as a Service or SaaS systems such as RightNow, Oracle Fusion, NetSuite or SalesForce all provide file based interfaces but they may have limitations such as file size or use non standard protocols to connect. Many use HTTP, which is easier for IT departments to open ports through the firewall. sFTP and FTP are still widely used especially for large files and support for restarting or resuming interrupted transfers. As discussed previously in the B2B use cases, file formats vary from application to application and module to module which is a challenge beyond the built in functionality of the typical enterprise application. The same is true for debatching large files and error handling. Enterprise applications do provide a thin integration layer but are often limited to API's or export/import incremental changes with no real ability to integrate directly with other applications. This is where a well-integrated suite with transfer and orchestration tools will be required.

Use Case Description

The use case is similar to the partner use case described above except your organization is using a SaaS application and consumer connecting to the hosted SaaS cloud service to integrate your on premises application such as Oracle Business Suite or Oracle PeopleSoft CRM. Simply put, it is the job of the application to automate data export and leave the heavy lifting to the integration layer. An MFT product deployed on premise can receive batch updates, secure and pass them on to the cloud service provider. When MFT is deployed to both the consumer and provider, it is very easy for MFT to compress and encrypt on the consumer side then uncompress and decrypt on the provider side before the updates are delivered to the cloud service application. Many cloud services such as RightNow or Fusion applications support sFTP/FTP and with a lightweight Secure FTP proxy server in the DMZ or Green zone, all of the integration products can be safely deployed inside a private zone DMZ.
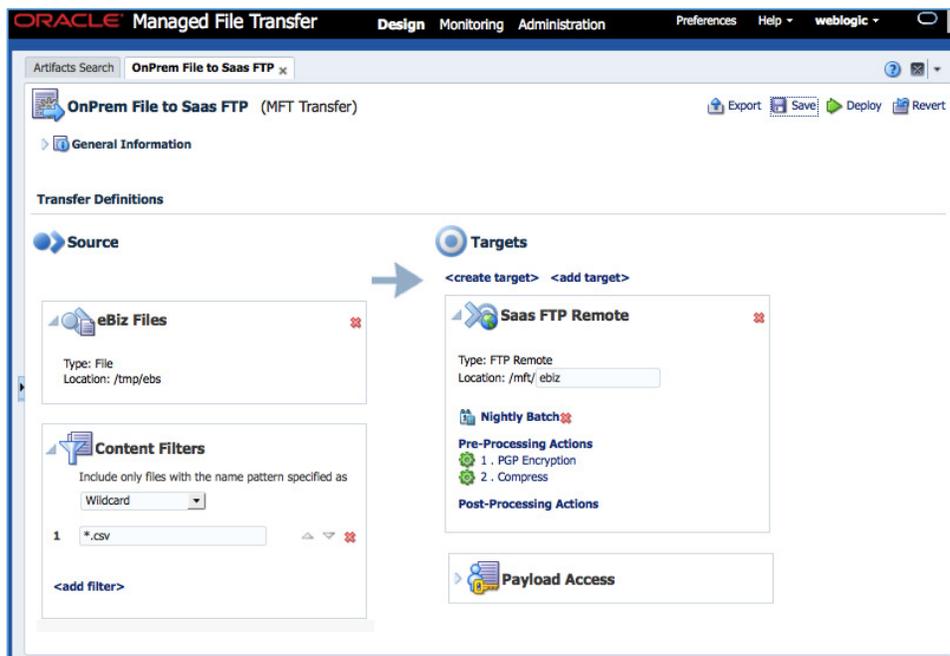

Figure 15: Oracle MFT On Premise to Cloud SaaS Transfer Example

If the cloud services do not support sFTP then support for cloud specific protocols may be required. Examples would be the Oracle Public Cloud(OPC) storage service or the Amazon S3 storage service. This allows the SaaS application to utilize the local storage service for delivery of larger batch files.

Requirements

» Endpoint protocol support for sFTP or a cloud storage service
» Proxy support to allow deployment of MFT and integration processes inside the firewall DMZ
» Compression, encryption and the ability to handle debatching of large files
» Schedule nightly batch updates

Conclusion

File processing has matured a lot in the recent years. Even though real-time processing is dominant in many integration patterns, files are still the bread and butter for many industries while integrating back-end enterprise systems. Cloud integrations are expected to increase the demand for file-based integrations even more. Built- in functionality for security, encryption, and global visibility have long been core MFT requirements. But as this paper has illustrated, an easy to use MFT solution that integrates with existing technologies, is an invaluable tool for solving complex problems such as partner EDI transactions, processing very large files or securing the enterprise.

Author: Dave Berry, Vikas Anand, Mala Ramakrishnan

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

ORACLE®

CONNECT WITH US

blogs.oracle.com/soa

facebook.com/oraclesoa

twitter.com/oraclesoa

oracle.com/soa

Hardware and Software, Engineered to Work Together

Oracle is committed to developing practices and products that help protect the environment