Managed File Transfer in Enterprise Java Applications

By David Sims
Flux

I: Why Should You Care About Managed File Transfer?

In an SOA world, bulk data transfer occurs largely by way of file transfer. "Multiple studies show that around 80% of business-to-business traffic consists of files," says Jonathan Lampe of File Transfer Consulting, a vendor-neutral consultancy focused solely on file transfer issues.

File transfer remains a critical component of enterprise architectures. Enterprise Java developers are all familiar with point-to-point file transfers. However, the demands of *enterprise* file transfers require more sophistication. Managed file transfer provides the solution.

So what is ***managed*** file transfer and why should you care?

Managed file transfer is holistically concerned with your organization's critical file transfers.

1. These file transfers must succeed or, if they fail, the failed transfers must be handled in a manner consistent with your organization's and your trading partners' business needs and service level agreements.

2. Managed file transfers must be integrated into the business processes and workflows of your organization and that of your trading partners. Business value derives from orchestrating files from their origin through various steps in a workflow to their final destinations.

3. The people involved in automated file transfers are as important as the software that performs those transfers. At various points in the file transfer lifecycle, IT operations staff must be engaged to troubleshoot delayed and failed transfers. IT staff may also be engaged to provide critical business judgments that can impact file transfer workflows.

4. Metrics that provide information on the volume and timeliness of file transfers are important for infrastructure planners, IT management, and business management.

II: Why a Point-to-Point File Transfer Solution Is Inadequate

There are various Java APIs and libraries that can transfer a file from point A to point B. Generally speaking:

- They cannot provide a mechanism for error handling and enforcing service level agreements (SLAs). Sure, they provide the developer with a notification that an error occurred by way of an exception or an error code. However, they do not provide a *mechanism* for reacting to errors and enforcing SLAs. As usual, that task is left as an exercise for the developer.

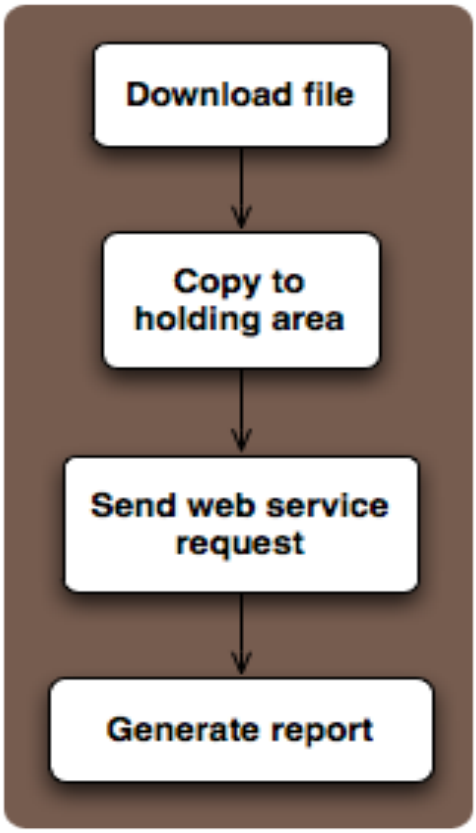
- They cannot orchestrate files through a data workflow. In real world scenarios, files must be ushered through various steps in a data workflow. Point-to-point solutions do not have this capability.

- They do not provide a management console for IT operations staff to monitor and manage file transfers. Inevitable transmission errors require intervention from operations staff.

- They do not provide metrics to assist in IT infrastructure planning or reporting for management.

"If you dig into the technical aspects of what separates a managed file transfer from a secure file transfer, you end up dealing with two core capabilities," says Lampe. "First, there are provisions to automatically retry or reroute failed transfers, and send notifications to various systems and people if transfers continue to fail. Second, there are mechanisms to ensure that each transfer meets a non-repudiation test: that is, that we can prove the identity of the sender, the content of the file, and time of every transfer made."

III: A Practical Example



Let's step through a simplified but practical example to illustrate some typical capabilities that a managed file transfer solution provides. In this example, we'll orchestrate the transfer of files that originate on a stock exchange to a central server. The file transfer workflow can be expressed as follows.

1. Download a file from the exchange at the end of the trading day.
2. Copy the file into a holding area.
3. Send a web service request to an ETL (extract, transform, load) tool, where the file will be extracted and loaded into a database.
4. Generate a report from the data and make it available to customers of the stock exchange.

Configure Files to Watch For and Transfer

First, the exchange's download file must be configured. To streamline operations and reduce configuration errors, we'll configure the file name and network location through a point-and-click management console.

Create the File Workflow

To make file workflows flexible and configurable by IT resources outside the development team, we'll create the workflow through our management console.

1. This file workflow example watches a secure FTP server for the exchange's data file to appear at the end of the trading data. Often, the current date is encoded into the file name. In the following example, the date 04 July 2011 is encoded in the file name:

   ```
   nyse_volatile_stocks_04_07_2011.zip
   ```

   Our file workflow watches the FTP server for a file that matches this pattern to appear each day. When it does, the file is ready to be downloaded and copied into a holding area.

2. Once the file is in place on the local network, a Web Service call is placed to an ETL tool, instructing it to extract data from the freshly downloaded file and load it into a database.

3. When the ETL tool finishes its task, a report is generated from the data that was just delivered.

When Errors Occur, SLAs Are Violated, and Penalties Are Levied

Of course, nothing is perfect when it comes to networks and computing platforms. In our example, our organization and the exchange have agreed to an SLA of four hours from the time the stock exchange data becomes available until the report is ready and available.

Naturally, our organization wants to stay within the SLA and avoid SLA penalties, which are frequently written into SLA agreements involving financial organizations.

Our organization has decided that when a file transfer fails, it will automatically retry the transfer once an hour for up to three hours. After three hours of failed transfers, an escalation notice is sent to the IT operations staff as a warning that the SLA may be violated.

After four hours of failed transfers, a new escalation notice is sent to the second level of IT operations staff. Their task is to mitigate the penalties levied by the SLA by investigating the cause of the file transfer failures, resuming the workflow, and making the report available to the stock exchange customers.

Operations Staff Monitors File Transfer Workflows

In our example, the IT operations staff is tasked with responding to upholding the SLA after three hours of failed file transfer attempts. The SLA escalation notifications occur by email and through red flag notifications on the management console.

The pro-active operations staff also monitors file transfers and the overall health of key file workflows in order to spot an unusual problem before an automatic SLA escalation is sent as per the error handling policy.

IV: How to Choose a Managed File Transfer Solution for Enterprise Java Applications

Now that we understand better why managed file transfer is a key component in many enterprise architectures and applications, what factors should you consider in selecting a managed file transfer solution for your enterprise Java applications?

Do You Need to Embed the Solution in your Application?

Depending on the nature of your application, you may need to embed a managed file transfer solution directly in your application, much like embedding a class library. On the other hand, others prefer running the managed file transfer solution as a standalone server. The decision rests with what seems to be the best approach for your situation:

- Embedding the solution hides it from others, which simplifies the environment for others. Embedding also tends to imply that configuration changes are most easily made by the development team.
- Deploying the solution as a standalone server means it can be set up, configured, and reconfigured later by any member of the IT team.

Are the Protocols You Need Supported?

In managed file transfer solutions, files are moved using various protocols. The usual choices are Secure FTP (SFTP) or FTP-over-SSL (FTPS). However, there are other options.

- AS2 is an EDI protocol used to exchange data between trading partners. AS2 was popularized when Walmart adopted it for communicating with its suppliers.
- Connect:Direct is a file transfer product used for file transfer between mainframes and mid-range computers.

In short, make sure the managed file transfer solution you consider supports the file transfer protocols you need.

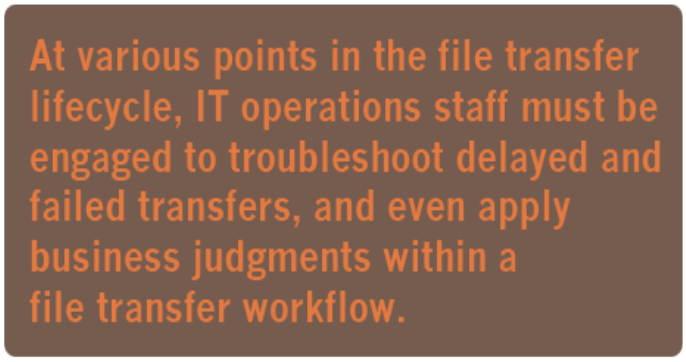
Do You Need to Orchestrate Data Workflows?

Some business situations require data workflows. A data workflow is simply a workflow orchestration consisting of multiple steps with conditional branching and looping logic to meet the needs of your business and its trading partners.

Do You Need to Handoff Files to Web Services?

Inbound files are meant to be processed. They might be processed by logic internal to your enterprise Java application. However, they just as easily could be processed by an external server or process. These days, those external servers expose their services with a Web Service.

A managed file transfer solution that includes built-in Web Services integration points reduces development complexity.



How Are Errors Handled?

Inevitably, file transfers fail. Reacting to those errors in a way that is appropriate for your enterprise and trading partners is required, especially when costly SLA penalties may apply.

Because development rarely knows ahead of time how errors need to be handled, IT staff needs to be able to design and update appropriate error handling responses.

- Are errors highlighted on the management and operations console so they are easily spotted by the operations staff?
- Are errors logged to the file system or to SNMP traps?
- Are notifications sent only when the severity of a failure reaches a heightened state? No one wants to be awakened when a file transfer fails once in a while but is subsequently transferred successfully later.

- Are the error handling logic and SLA escalation capabilities sufficiently expressive to meet the needs of your business and trading partners without having to resort to custom code?

What Kind of File Transfer Retry Logic is Supported?

Failed file transfers are typically retried automatically before handing control over to the operations team. A managed file transfer solution that allows your retry policies to be configured easily through a user interface dramatically speeds up development, configuration, and deployment time.

An example file retry policy is to wait 30 minutes between failed file transfers before retrying, with a hard failure and notification to operations staff after three hours of repeated failures.

Error Notifications and SLA Escalations

How are success and error notifications sent? What options are there for sending to different audiences at different priority levels?

Make sure that your managed file transfer solution supports the kinds and levels of error notifications and SLA escalations that your business and its trading partners require. For example, do you need email, SMS, or SNMP notifications? Can you design error handlers that mirror your SLA policy?

Promotions and Migrations

Ideally, file transfer definitions and workflows, once defined and tested, should move up the chain from development to QA to production without suffering copying errors.

How are those file transfer definitions transferred from development to QA to production?

Security Credentials Assigned by Operations Staff

Frequently, development or an IT group may create your file transfer definitions. However, those people are often not privy to the passwords and security credentials required for running those file transfers in production.

Does your solution under consideration allow passwords and other security credentials to be set by operations staff and not the IT staff who create the file transfer definitions and workflows?

How Is Operations Staff Supported Through a Monitoring Interface?

As noted at the beginning of this article, the people involved in automated file transfers are as important as the software that performs those transfers. At various points in the file transfer lifecycle, IT operations staff must be engaged to troubleshoot delayed and failed transfers, and even apply business judgments within a file transfer workflow.

What kind of operations console will best support these needs?

Look for a solution that contains the minimum security requirements that your organization requires.

- *No Security Approach:* In rare situations, using an operations console with no security is acceptable. In these atypical situations, everyone is fully trusted to not touch what they are not permitted to touch or view what they are not permitted to view.

- *All Or Nothing Approach:* Security is applied and users are broken down into only two groups: administrators and users. Administrators can do anything while users from performing certain functions. Users cannot be further differentiated.
- The problem with this approach is that in the real world, there are different classes of users. For example, some operators work with Asian stock exchanges, some operators work on just North or South American stock exchanges, and finally some operators work only on European or African exchanges.
- It's often necessary to assign different privileges to different users as not all users are assigned the same tasks.
- *Segmented Security Approach:* Security is applied using a granular or fine-grained approach. Certain users are permitted to perform only certain operations. These privileges are configured and maintained by an administrator, with possible integration into your LDAP or Active Directory server.

- For example, the European operations staff is permitted to work on European file transfer workflows but not allowed to work with file transfers from any other continent.
- Furthermore, some European operations staff are permitted to restart failed file transfers while other staff are allowed to merely view the status of the European file transfers. Some European operations staff may be prevented from viewing the status of certain file transfers altogether.

Metrics

Does the solution provide information on the volume and timeliness of file transfers in the form of metrics? These metrics can be useful for infrastructure planners, IT management, and business management.

V: Summary

Managed file transfer provides crucial business value above and beyond traditional point-to-point file transfer class libraries and server software.

1. Errors with transferring files must be handled in accordance with the practices of the enterprise and its trading partners. Escalations must occur when service level agreements are in danger of being violated or have, in fact, been violated.
2. Workflow orchestration is needed to usher a file through its lifecycle as it flows from its origin to its destination, traversing across multiple networks and through different software applications such as reporting software and ETL (extract, transform, load) tools.
3. A graphical management and operations console is required to engage the operations team who are crucial to the smooth execution of file transfer workflows, for resolving failed or delayed file transfers, and for possibly applying business judgments within file transfer workflows.
4. Finally, metrics provided IT management, infrastructure planners, and business management with information to spot emerging trends, troublesome areas, and insight in your enterprise and its file transfer activities.

About the Author

David Sims is president of Flux, a leading workload automation and managed file transfer solution.