
Managed File Transfer in Java Applications

Finding the Right Fit

Reading Time: 7 minutes

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 generally 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?

Managed file transfer is holistically concerned with your enterprise'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 engage to troubleshoot delayed and failed transfers. IT staff may also engage 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

Why a Point-to-Point File Transfer Solution Is Insufficient

There are various Java APIs, open source projects, and commercially available 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."

A Sample Use Case

Just for context, 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 from a customer's enterprise (e.g., a corporate client of our service). Against these files we run analysis and reporting processes, and return the results to the customer. The files could be payments, invoices, or even a data feed for us to perform analytics against. The file transfer workflow can be expressed as follows.

1. Download a file from the customer at the end of the business 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 return the results to the customer.

To accomplish the above, the solution must first allow the customer's download file to be configured. This normally requires the network address, network port, security credentials, and protocols be configured.

Next, the file's processing workflow needs to be defined. In this example the solution watches a secure FTP server for the customer's data file to appear at the end of the business day. The file is selected by searching for the current date being encoded into the file name (e.g., client_data_feed_04_07_2011.zip). The 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 from the customer and copied into a holding area on the local network.

Once the file is downloaded onto 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. When the ETL tool finishes its task, a report is generated from the data that was just delivered. The report is then transferred back to the customer.

Beyond encoding the file transfer and processing, the workflow also defines the service level agreement of the file transfer. In this case, our organization and the corporate customer have agreed to an SLA of four hours from the time the customer data becomes available until the report is ready and returned to the customer.

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.

Finally, the solution requires a means for operations staff to monitor these file transfers and their associated workflows as well as assure that SLA's are being adhered to. In this case 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 a management console.

The operations staff also proactively 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.

While this example is simplistic, it covers many of the salient points that must be addressed in supporting managed file transfers in enterprise Java environments.

Finding the Right Fit

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?

1. 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. Or you may need both. 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.
- Sometimes the requirement crosses both spectrums. In this case, a standalone server that exposes a robust REST API allows applications to interact with the file transfer solution programmatically, while allowing IT operations staff to interact with the server via its user interface.

2. Are the Protocols You Need Supported?

In managed file transfer solutions, files are moved using various protocols. The usual suspects 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 (also known as NDM) is a file transfer product used for file transfer between mainframes and mid-range computers.

- New, high throughput file transfer solutions are available today. For those enterprises requiring these new protocols, having a managed file transfer solution that can be extended to accommodate new and evolving protocols may be the best solution.

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

3. Do You Need to Orchestrate File Workflows?

Some business situations require file workflows. A file 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.

4. Are File Workflows Created Via Scripts, or Graphically, or Both?

In those situations requiring file workflows, the creation and maintenance of these workflows becomes a key consideration in their flexibility and ease of understanding. There are pros and cons associated with scripts and graphical representations. Ideally the solution can handle both approaches.

5. Where are File Workflows Stored and Maintained?

A central, database-managed store of file workflows simplifies and extends the degree of control and management that can be performed. Keeping workflows in text files is adequate where the number of workflows is small, but as more and more workflows are developed the management overhead becomes untenable. A central store of workflows provides greater oversight and reuse, while providing better control over workflow promotion and testing.

6. Do You Need to Schedule Transfers?

Oftentimes the initiation of a file transfer requires more than the simple detection of a file in a folder. The existence of the file, (and sometimes the non-existence of the file) must be checked for periodically based on a schedule. This schedule may be as simple as every day at 5:00 pm, or may be more involved, such as the last Monday of every month except December. Having a capable scheduler that also addresses business calendars is often a requirement for such solutions.

7. Do You Need to Handoff Files, or Specific File Content, to Web Services?

Inbound files generally require some degree of processing be performed on them. This processing may involve logic internal to your enterprise Java application, or the processing may be delegated to an external server, or external process, or possibly even a network appliance. Such external servers generally expose their services with a web service.

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

8. 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?

9. 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.

10. 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?

11. How Will You Promote and Deploy Workflows?

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. The actual workflows and configurations should be exportable and capable to be placed into a version control system for audit and configuration management.

Also consider how file transfer definitions transferred from development to QA to production? Is there a straightforward and integrated promotion mechanism, or is the effort highly labor-intensive and manual?

12. How Will You Assign Security Credentials?

Frequently, development or an IT group may create 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 or security staff and not the IT staff who create the file transfer definitions and workflows?

13. Use the Vendor's Monitoring Interface or Create Your Own?

The central control point for many solutions is the provided operations console or monitoring interface. In many cases this is sufficient, but in many situations, particularly with embedded solutions, your organization needs a specialized or unique interface.

Does the solution provide an API and example code (preferably web service enabled) that allows you to create your own customized 'dashboards' to meet your organization's unique requirements?

14. 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 following capabilities:

- *Workflow and File Transfer Control:* Support for starting, pausing, restarting, recovering, canceling, expediting, and prioritizing workflows and file transfers is essential.
- *Logs, Reports, and Workflow Run History:* Being able to research events and previous runs reduces the need and security risk of granting IT Operations staff access to the file system to review logs.
- *Segmented Security Support:* Security is applied using a granular or fine-grained approach. Certain users are permitted to perform only certain operations. The operations console should be configurable, without programming, to provide users the capabilities that they can access without showing capabilities they are not authorized to access. Integration into the organization's LDAP or Active Directory server is highly desirable.

15. 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.

16. Support and Extensibility

Whatever solution you select today, realize your commitment may be for a very long term. Numerous file transfer solutions are deployed and run fundamentally unchanged for many years – sometimes decades. Not only will your team need to understand and use the solution, but successors to your team will also have the same learning and support efforts ahead of them. The vendor's long-term commitment to their solution is key to your not being orphaned on an abandoned platform.

Beyond support is the issue of extensibility. Can you take control and extend the solution to add new features, new protocols, and new kinds of work? Considering such extensibility features allows one to contemplate features and functions beyond what is on the vendor's roadmap.

In Summary

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

- 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.
- 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, analytics engines, and ETL (extract, transform, load) tools.
- 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.
- Security features exist to restrict access to file transfer operations whether such

operations are done via a web console, command line, or code. Support for audit and logging, and data-at-rest and data-in-transit security concerns is also provided.

- Finally, metrics provide 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 Flux

The Flux software platform orchestrates file transfers and batch processing workflows for banking and finance. First released in 2000, Flux has grown into a financial platform that the largest US, UK, and Canadian banks and financial services organizations rely on daily for their mission critical financial systems.

Contact Flux

+1 720-438-4304
sales@flux.ly www.flux.ly