Contents

Introduction............................................................................................................................................ 6

Installing Console................................................................................................................................. 8
  System and Firewall Requirements........................................................................................................ 8
  Installing Console................................................................................................................................. 9
  Logging Into Console for the First Time............................................................................................... 10
  Upgrading Console to the Current Version.......................................................................................... 10
  Uninstalling Console............................................................................................................................. 12

Configuring Email Notifications........................................................................................................ 13
  Email Server Configuration.................................................................................................................... 13
  Configuring Notification Time Zones and Cutoff Times....................................................................... 13
  Configuring Advanced Rulesets for Email Notifications................................................................. 14
  View Email Notification Statistics....................................................................................................... 15
  Configuring Personal Email Notifications.......................................................................................... 16

Configuring Email Notification Templates.......................................................................................... 18
  Creating a New Email Notification Template...................................................................................... 18
  Editing Email Templates....................................................................................................................... 19
  Displaying Email Notification Templates for an Email Address......................................................... 21

Adding Nodes to Console.................................................................................................................... 22
  Overview: Adding Nodes to Console.................................................................................................... 22
  Adding an Administrative Account to a Windows Machine............................................................... 22
  Adding an Administrative Account to an OS X Machine..................................................................... 23
  Adding an Administrative Account to a Linux Machine..................................................................... 24
  Creating a Managed Node in Console................................................................................................... 26
  Updating a Node's Admin Credentials.................................................................................................. 26
  Adding Unmanaged Nodes..................................................................................................................... 27
  Adding Endpoints................................................................................................................................ 27
  Set a Global Docroot for the Node.......................................................................................................... 28

Managing the Localhost Node.............................................................................................................. 30
  The Localhost Node.............................................................................................................................. 30
  Adding Unmanaged Nodes................................................................................................................... 30
  Editing the User or Group on a Node..................................................................................................... 30
  Set a Docroot for a Node User or Group.............................................................................................. 31
  Adding Endpoints................................................................................................................................ 32
  Configuring Virtual Links...................................................................................................................... 33
  Scheduling Virtual Links....................................................................................................................... 33

Adding Managed Clusters to Console.................................................................................................. 34
  Adding a Managed Cluster to Console.................................................................................................. 34
Running Reports................................................................. 87
  Creating a Basic Report...................................................... 87
  Creating an Advanced Report........................................... 88
  Finalizing and Running a Report....................................... 89
  Editing Custom Variables.................................................. 90
  Creating Custom Fields.................................................... 91

Configuring SSH Keys......................................................... 94
  SSH Keys............................................................................. 94
  Storing SSH Keys on Console............................................. 94
  Transferring Files with an Endpoint Using SSH Keys........... 95

Working With SSL.............................................................. 96
  Installing a Signed SSL Certificate Provided by Authorities.. 96
  Generating a New Self-Signed SSL Certificate............... 97
  Regenerating Self-Signed SSL Certificate (Apache)............ 98

Working with Shares and Directory Services....................... 99
  Console and Shares on Same Machine............................... 99
  Configuring the Directory Service.................................... 99
  Adding Remote Users...................................................... 100
  Adding Remote Groups.................................................... 100

Working with SAML............................................................ 101
  Working with SAML.......................................................... 101
  Configuring SAML............................................................ 101
  Configuring Your Identity Provider (IdP)......................... 101
  User Accounts Being Provisioned by SAML Just-In-Time (JIT) Provisioning..................................................... 102

Backing Up Console Database............................................. 103
  Back Up Console with asctl.............................................. 103
  Backing Up Console with the Web UI................................ 103
  Restoring the Console Database....................................... 103
  Backing Up the Current Console Configuration............... 104
  Restoring the Current Console Configuration.................... 105

Managing the MySQL Database........................................... 106
  Configure MySQL Settings.............................................. 106
  Running MySQL on a Separate Machine......................... 106
  Purging Data from Console.............................................. 107
  Restoring Purged Data.................................................... 108

Troubleshooting Console................................................... 109
  Updating your Console License....................................... 109
  Restart Console Services............................................... 110
Appendix .................................................................................................................. 112

Disabling SELinux ........................................................................................................ 112
Configuring Console Defaults .................................................................................... 112
Understanding Space Watcher .................................................................................. 115
Working with Tags ....................................................................................................... 116
Configure Background Processes ............................................................................... 117
Configure the Apache HTTP Server .......................................................................... 118
asctl Command Reference ....................................................................................... 119
Advanced Search ......................................................................................................... 123
Setting Up the Console Environment ........................................................................... 125
  Setup Example #1: Monitoring Transfers with Another Organization....................... 125
  Setup Example #2: Managing Aspera Faspex Transfers ........................................... 128
  Setup Example #3: Create Groups of Different Permissions ..................................... 131
Email Template Examples ........................................................................................... 134
  Email Template Example: Creating a Simple Notification for a Successful Transfer  134
  Email Template Example: Adding Company Branding to Your Template ............... 135
Node References ......................................................................................................... 136
  Node-Level Configuration Options ........................................................................... 136
  Node Account-Level Configuration Options ............................................................ 146
Transfer References .................................................................................................... 153
  Simple Transfer Options .......................................................................................... 153
  Smart Transfer Options ......................................................................................... 155
  Watchfolder Options .............................................................................................. 157
  Specify Base for Source Path .................................................................................. 160
Report References ....................................................................................................... 161
  Reference: Basic Report Organization Options ....................................................... 161
  Reference: Built-In Fields for Custom Field Rules .................................................... 161
  Reference: Reporting Filters .................................................................................... 163
  Reference: SQL Variables for Advanced Reports ..................................................... 164
  Reference: Database Fields for Advanced Reports .................................................. 166
Advanced Report Usage Notes .................................................................................... 175
  Advanced Report Usage Notes: Avoid Duplicating Identical Records ....................... 175
  Advanced Report Usage Notes: Avoid Duplicating Redundant Records .................. 176
  Advanced Report Usage Notes: Filter on Raw Values .............................................. 179
  Advanced Report Usage Notes: Filter Strings by Using "Begins With" ....................... 179
  Advanced Report Usage Notes: Always Include a Date Filter .................................. 180
  Advanced Report Usage Notes: Always Name Your Computed or Aggregated Columns 181
  Advanced Report Usage Notes: Avoid Joining Reporting Views ................................ 181
Example Reports ......................................................................................................... 185
  Basic Report Example: Faspex User Activity ........................................................... 185
  Basic Report Example: Hot Folder Activity .............................................................. 188
  Basic Report Example: Faspex Metadata ................................................................. 189
  Advanced Report Example: Transfer Sessions with High Packet Loss ...................... 193

Technical Support ....................................................................................................... 196

Legal Notice ................................................................................................................. 197
Introduction

IBM Aspera Console is a web-based application which allows users to centrally manage, monitor and control Aspera servers and transfers. Console offers the following features:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transfer monitoring and control</td>
<td>View transfers, pause / resume / cancel, change transfer rates.</td>
</tr>
<tr>
<td>Transfer Initiation</td>
<td>Initiate and schedule transfer jobs remotely.</td>
</tr>
<tr>
<td>Node Configuration</td>
<td>An administrator can configure all nodes directly from Console such for options such as bandwidth, priority, and encryption.</td>
</tr>
<tr>
<td>Email Notification</td>
<td>Notify users of transfer events with customizable messages.</td>
</tr>
<tr>
<td>Reporting</td>
<td>Create detail and summary reports of transfer activity.</td>
</tr>
<tr>
<td>Role-based access control</td>
<td>Manage what transfers are visible and controllable to Console users with security groups.</td>
</tr>
</tbody>
</table>

Figure 1 shows the relationship between Console, managed nodes, and an unmanaged node. A managed node is defined as a node which can be configured by Console. In addition, its transfer activities can be managed from Console and are logged to the Console database. Console controls managed nodes by connecting with SSH to allow node configuration and file browsing. Console handles regular nodes and legacy nodes differently in the following ways:

- For regular nodes, Console makes REST calls to the Node API to pull transfer details into console as well as to start and control transfers.
- For legacy nodes, Console processes data that gets pushed to the database by the node. Console also communicates with legacy nodes via SOAP calls to start and control transfers and to check for cases where nodes failed to log the end of a transfer.

Unmanaged nodes are not under control of the Console and are used as external transfer endpoints. In the blue box, Node 1 and Node 2 are managed nodes. All transfer activity on these nodes can be monitored and controlled. Node 3, outside of the blue box, is an unmanaged node. Console is only aware of transfers on this node that are with a managed node (for example, Transfer 2, above).
Figure 2 illustrates the concept of nodes and endpoints. An endpoint is an individual user account on a node which can perform Aspera transfers. You can have one or multiple endpoints on a node depending on your business needs. As shown above, Node 1 contains Endpoint A (asp1@node1) and Endpoint B (asp2@node1). Transfers take place between endpoints, for example, Transfer 1 between Endpoint A and Endpoint D, and Transfer 2 between Endpoint B and Endpoint C.
## Installing Console

### System and Firewall Requirements

| Operating System                              | • Red Hat Linux 6 or 7  
|                                               | • CentOS 6 or 7  
|                                               | • SUSE 11  
| Minimum Hardware Requirement                  | • 2 GHz dual-core CPU (or better)  
|                                               | • 4 GB of RAM  
|                                               | • 2 GB of disk space  
| Data Storage Requirement                      | In terms of planning for the size growth of the database, the per file records generate 1-2KB per file transfer, and the session records generate 8-12KB per session. For some size estimates, here are a few examples:  
|                                               | • 100 sessions per day of 1000 files each, all external transfers between managed and unmanaged nodes = approx 2 MB per day db growth, 60 MB per month, 700 MB per year.  
|                                               | • 1000 sessions per day of 1 file each, all internal between managed nodes = approx 14 MB per day, 420 MB per month, 5 GB per year.  
|                                               | • 1000 sessions per day, 10,000 files each, 50% internal between managed nodes, 50% external with unmanaged node = approx 15 GB per day, 450 GB per month, 5.4 TB per year.  
| Applications                                  | • Aspera Common Components  
|                                               | • MySQL Database  
|                                               | • Apache HTTP Server  
|                                               | ✨ **Note:** If there is an existing installation, shut down the MySQL database and the Apache HTTP server during installation.  
| Node Machine                                  | • In order to work with Console, node machines must have an Aspera transfer product installed (IBM Aspera Enterprise Server, IBM Aspera Connect Server, or IBM Aspera Point-to-Point Client).  
|                                               | • To use the new Console architecture implemented in Console version 3.0 and on, the Aspera transfer products on existing node machines must be upgraded to at least version 3.4.6 and ideally to the latest version. These legacy nodes must be converted to use Node API using the built-in convert option in the Console 3.0 Node edit screens. If a node is not converted to use Node API, the legacy node will continue to report to Console using the existing mechanism. For more information on node conversion, see [Converting Legacy Nodes](#) on page 36.  
| Firewall (on the Console Machine)             | Open the following ports on the Console machine:  
|                                               | • For the Web UI, allow inbound connections for HTTP or HTTPS Web access (for example, TCP/80, TCP/443).  
|                                               | • Allow outbound connections for SSH (to be used for node administration) on a non-default, configurable TCP port (for example, TCP/33001).  
|                                               | • Allow an outbound connection for TCP/9092 to allow Console to connect with nodes through the Node API  
|                                               |  


| Firewall (on the Node Machines) | • Allow an outbound connection for TCP/40001 and an inbound connection for TCP/4406 to allow Console to connect with legacy nodes. |
| • To ensure that your server is secure, Aspera strongly recommends allowing inbound connections for SSH on TCP/33001 (or on another non-default, configurable TCP port), and disallowing inbound connections on TCP/22. If you have a legacy customer base using TCP/22, you can allow inbound connections on both ports. For details on securing your individual Aspera transfer server product, review the corresponding user manuals. |
| • Allow inbound connections for FASP transfers, which use UDP/33001 by default, although the server may also choose to run FASP transfers on another port. |
| • For current nodes and legacy nodes that have been converted to current nodes, allow an inbound connection on TCP 9092. |
| • For legacy nodes (unconverted), allow an inbound connection for Aspera Central (for example, TCP/40001). |
| • For legacy nodes (unconverted), allow an outbound connection for logging to Console on TCP/4406. |

**Note:** No servers are listening on UDP ports.

When an Aspera client initiates a transfer, the client opens an SSH session to the SSH server on the designated TCP port and negotiates the UDP port over which the data transfer will occur.

## Installing Console

**Warning:** Due to incompatible common components, do not install IBM Aspera Console and IBM Aspera Faspex on the same machine. Aspera does not support this combination.

This topic describes how to perform a fresh installation of Console. To start the installation process, log into your computer as an Administrator and follow the steps below. If you are upgrading Console rather than starting a fresh installation, please refer instead to Upgrading Console to the Current Version on page 10.

1. **Download Console installation components.**
   
   Use the credentials provided by Aspera to download both the Console installer and the common components from the Aspera website: [http://downloads.asperasoft.com/en/downloads/3](http://downloads.asperasoft.com/en/downloads/3)

2. **Run the downloaded installation components.**

   Run these commands with proper administrative permissions. Replace file names accordingly:

   
   ```
   $ rpm -Uvh aspera-common-version.rpm
   $ rpm -Uvh aspera-console-version.rpm
   ```

3. **Run the asctl setup command.**

   Run the following command:

   ```
   $ asctl console:setup
   ```

4. **Perform a streamlined setup or a detailed setup.** Press Enter to select a streamlined setup unless you need to configure non-standard ports or to configure components running on other servers. Follow the instructions to complete the setup.

5. **Optional:** Update the Aspera license from the command line.

   Console administrators can update the license in the Console web UI, see Logging Into Console for the First Time on page 10. However, if you are automating Console installation, you can use a rake command to set the license without logging into the web UI.
1. Set the license text as an environment variable.
   
   ```bash
   # export LICENSE_TEXT='<ASPERA_LICENSE> <DETAILS expiration_date=... </KEY> </ASPERA_LICENSE>'
   ```

   In this example, only part of the license text is shown. You must paste the entire license text for the license to be valid.

2. Update the Aspera license:

   ```bash
   # asctl console:rake aspera:update_license
   ```

   To access the Console interface, go to the following address with a browser: `http://server_ip_or_name:port/aspera/console`. For instructions on logging in for the first time, see *Logging Into Console for the First Time* on page 10.

---

**Logging Into Console for the First Time**

- **Note:** IBM Aspera Console requires Adobe Flash Player. If you do not have Adobe Flash Player installed, download and install Flash Player here: [http://get.adobe.com/flashplayer](http://get.adobe.com/flashplayer).

1. Access the Console interface by entering its hostname or IP address followed by `/aspera/console` in your web browser. For example, enter `https://IP_address/aspera/console`.

   Console requires a valid license key before a user can access and interact with the Console interface. An administrator must log in to paste a valid license or upload a valid license file before other users can log into Console. No other user interaction is permitted until a valid license is installed.

2. Enter the username and password and click **Login**. At this point, Console prompts you to change your password. Verify the old password. Then, enter and confirm a new password. Click **Change Password** to save your new password and login.

   - **Note:** Passwords must be at least six characters long, with at least one letter, one number, and one symbol.

3. Console directs you to the Console configuration page to update your Console license with the license Aspera provided to you. Click **Upload a license file** to upload a license file or paste the license text into the text window.

---

**Upgrading Console to the Current Version**

If an older version of IBM Aspera Console is already installed on your machine, upgrade to the newest version. Aspera supports upgrading from the following versions.

<table>
<thead>
<tr>
<th>Console and Common Versions</th>
<th>RHEL 6 (64-Bit) / CentOS 6 (64-Bit)</th>
<th>CentOS 7 (64-Bit) / RHEL 7 (64-Bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Console 3.1.2 + Common &gt;= 1.2.18</td>
<td>• Console 3.1.2 + Common &gt;= 1.2.18</td>
<td></td>
</tr>
<tr>
<td>• Console 3.1.1 + Common &gt;= 1.2.18</td>
<td>• Console 3.1.1 + Common &gt;= 1.2.18</td>
<td></td>
</tr>
<tr>
<td>• Console 3.1.0 + Common &gt;= 1.2.16</td>
<td>• Console 3.1.0 + Common &gt;= 1.2.16</td>
<td></td>
</tr>
<tr>
<td>• Console 3.0.7 + Common &gt;= 1.2.16</td>
<td>• Console 3.0.7 + Common &gt;= 1.2.16</td>
<td></td>
</tr>
<tr>
<td>• Console 3.0.6 + Common &gt;= 1.2.15</td>
<td>• Console 3.0.6 + Common &gt;= 1.2.15</td>
<td></td>
</tr>
</tbody>
</table>
If you are upgrading from a version not listed above, contact Aspera Support for further information.

**Note:** As of Console 3.0.3+, Console stores license information in the database, rather than in a file located at `/opt/aspera/console/config/`. If a license is added to the directory before the upgrade script is run, the script migrates the license information from the file into the database and deletes the file. To modify your license, see *Updating your Console License* on page 109.

1. Back up Console's database using the following `asctl` command in a Terminal:

   ```
   $ asctl -v console:backup_database
   ```

2. Stop Console and all related services.
   Run the following `asctl` command:

   ```
   $ asctl all:stop
   ```

3. Download Console installation components.
   Use the credentials provided by Aspera to download both the Console installer and the common components from the Aspera website: [http://downloads.asperasoft.com/en/downloads/3](http://downloads.asperasoft.com/en/downloads/3)

4. Install components.
   Run the following commands to install the downloaded files:

   ```
   $ rpm -Uvh aspera-common-version.rpm
   $ rpm -Uvh aspera-console-version.rpm
   ```

5. Run the following `asctl` command to upgrade Console:

   ```
   $ asctl console:upgrade
   ```

6. Optional: Upgrade legacy nodes.
Upgrade your legacy nodes (running an Aspera transfer product prior to version 3.4.6) to take advantage of Console features. For more information on converting your legacy nodes, see *Converting Legacy Nodes* on page 36.

## Uninstalling Console

1. Prior to removal, stop the Console application and its services in a Command Prompt window using the `asctl` command.

   ```bash
   # asctl all:stop
   
   $ rpm -e aspera-console
   $ rpm -e aspera-common
   
   3. Remove the `/opt/aspera/common` and `/opt/aspera/console` directories.
Configuring Email Notifications

Email Server Configuration

IBM Aspera Console needs to connect to a Simple Mail Transfer Protocol (SMTP) server to send email notifications.

1. Go to Notifications > Email Server.
2. Enter the SMTP server information [A, B, C].
   - If your SMTP server requires login credentials, select Login required under Authentication type and enter your login credentials. Otherwise, select Open authentication.
5. In the 'From' address [F] and 'From' name [G] fields, enter the default sender email address and sender name that appear in email notifications when they receive an email notification.
6. Enter your email address and select Save settings and send test email.
   - Check your email inbox for the confirmation email titled Email settings test. If you do not receive the email, review your settings or check your spam folder.

Configuring Notification Time Zones and Cutoff Times

1. Go to Notifications > Email Notification Options.
2. Select a default time zone for email timestamps.
3. Enter a cut-off time for delivering older emails.

### Configuring Advanced Rulesets for Email Notifications

Configure advanced rulesets for automated generation of additional email notifications beyond the simple announcement of transfer events. IBM Aspera Console checks configured rulesets whenever a transfer starts, completes successfully, or errors out for the final time (the transfer runs out of retries or Console detects a transfer that was supposed to retry but never did). If the transfer matches the ruleset, Console sends an email notification to the designated recipients.

1. Go to **Notifications > Advanced Rulesets** and click **Create New Ruleset**.
2. Enter a description of the ruleset.
3. Optional: Disable the ruleset to control when the ruleset comes into effect.
4. Select a filter.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>Filter by the IP address of a node machine.</td>
</tr>
<tr>
<td>Cookie</td>
<td>Filter by information in a transfer cookie. For more information on transfer cookies, see <em>Creating a Cookie Parsing Rule</em> on page 68.</td>
</tr>
<tr>
<td>Contact</td>
<td>Filter by the contact assigned by Console. A contact can be a Console username, a Faspex user name, a SSH account, or a customized value obtained from a transfer cookie. For example, a contact can be &quot;admin console&quot;, &quot;asera ssh&quot;, or &quot;asera faspex&quot; and so on.</td>
</tr>
<tr>
<td>Failover Group Name</td>
<td>Filter by the failover group name of the node. For more information about failover groups, see <em>Configure Failover Groups</em> on page 67.</td>
</tr>
<tr>
<td>Faspex Metadata</td>
<td>Filter by metadata found in a Faspex file package.</td>
</tr>
<tr>
<td>File Path</td>
<td>Filter by the file path of the transfer.</td>
</tr>
</tbody>
</table>
5. Select the side to apply the filter.

<table>
<thead>
<tr>
<th>Side</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Either</td>
<td>Apply the filter to both sides.</td>
</tr>
<tr>
<td>Source</td>
<td>Apply the filter to the source node.</td>
</tr>
<tr>
<td>Destination</td>
<td>Apply the filter to the destination node.</td>
</tr>
<tr>
<td>Client</td>
<td>Apply the filter to the node initiating the transfer request.</td>
</tr>
<tr>
<td>Server</td>
<td>Apply the filter to the node receiving the transfer request.</td>
</tr>
</tbody>
</table>

6. Select the comparator and enter the value.

Note: Select NOT to exclude entries matching the value.

For example, set the following parameters to send an email notification every time a node with the defined IP address participates in a transfer.

<table>
<thead>
<tr>
<th>MATCH</th>
<th>SIDE</th>
<th>NOT</th>
<th>COMPARISON</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>Either</td>
<td></td>
<td>=</td>
<td>10.0.0.1</td>
</tr>
</tbody>
</table>

7. Designate email recipients.

Enter an email address and click Add. Select an email template for each transfer event.

8. Click Create.

The newly created template appears on the Advanced Rulesets page where you can disable or enable, edit, copy, and delete rules.

**View Email Notification Statistics**

You can monitor notification activity in the Session Notifications report for each transfer. To view the report, go to a transfer's Sessions Details page. Click The Statistics column contains either a link describing the type of notifications configured for that session or None if no notifications were configured. Click the link to display the Session Notifications page.
The Session Notifications page provides the following information about the transfer:

- **Session Details**: This section gives basic information about the transfer, such as its name, status, and start and stop times.
- **Configured Notifications**: This section shows which types of notification were configured for this transfer (start, success, or error) and the name of the template configured for each.
- **Email Messages Sent (or Attempted)**: This section shows which types of notification were actually sent or attempted for this transfer (start, success, or error) and the name of the template used for each. You can see more detail about a message by clicking on it to launch the Email Message Details page, which provides more detail about a message, including its content. You can also resend messages listed in this section by clicking resend. This may be useful in cases where recipients are not receiving messages due to email server or configuration issues.

### Configuring Personal Email Notifications

Individual users can manage personal email notifications from their Preferences menu.

1. Open the Preferences page and select **Email Notifications**.

2. Select email templates for notifications that are triggered by the following events: transfer start, transfer success, or transfer error.

   You can create new templates or modify existing templates by going to **Notifications > Email Templates**. For more information on how to create and modify email templates, see **Editing Email Templates** on page 19.

3. Select or clear global email notifications. By default, Console notifies you for transfers that you start when those transfers start, succeed, or fail.

4. For each specific transfer path listed, select or clear notifications for transfer path. These notifications are disabled by default.
5. Click **Update**.
Creating a New Email Notification Template

IBM Aspera Console allows you to create and modify email notification templates based on three transfer events: transfer start, transfer success, and transfer error. You can customize emails based on recipient needs by creating a new template. For example, an error notification email to an internal admin typically contains as much information as possible, while a notice to an outside party might contain a bare minimum of information. You can edit the included default templates, create and edit new templates, and change which templates are used as defaults.

1. Go to Notifications > Email Templates.
2. Click on the appropriate "Create new..." link.

To create a new template, click Create new transfer start email template (A), Create new transfer success email template (B), or Create new transfer error email template (C) depending on the situation for which you want to send an email notification.
The new template (D) appears listed under the default template.

3. Rename the template.
   Click Edit Plain Template to open the plain text editor. Enter a descriptive name of this template in the Template name field. At this point, you can edit the template. For more information on editing templates, see Editing Email Templates on page 19. Otherwise, click Save to rename the template and return to the template preview page.

   Note: To ensure that information displays correctly in the email, edit both the plain text and HTML code versions of the template.

4. Optional: Make this template the default template.
   Return to the Email Templates page by clicking the Email Templates tab. Find your renamed email template and click default.

- Email Notifications for Transfer Error

Create new transfer error email template

<table>
<thead>
<tr>
<th>NAME</th>
<th>DEFAULT</th>
<th>ACTIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Error</td>
<td>✔️</td>
<td>edit</td>
</tr>
<tr>
<td>Customer Transfer Error</td>
<td></td>
<td>edit default delete</td>
</tr>
</tbody>
</table>

**Editing Email Templates**

IBM Aspera Console allows you to create and modify email notification templates based on three transfer events: transfer start, transfer success, and transfer error. You can customize emails based on recipient needs by creating a new template. For example, an error notification email to an internal admin typically contains as much information as possible, while a notice to an outside party might contain a bare minimum of information. You can edit the included default templates, create and edit new templates, and change which templates are used as defaults.

1. Go to Notifications > Email Templates.
2. Click edit for the email template you want to configure.
   Note: To ensure that information displays correctly in the email, edit both the plain text and HTML code versions of the template.
3. Click Edit Plain Template.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Template name</td>
<td>Modify the name of the template displayed in Console.</td>
</tr>
<tr>
<td>From Name</td>
<td>Enter the name displayed as the email sender.</td>
</tr>
<tr>
<td>Reply-to Address</td>
<td>Enter the email address receiving replies from the email recipient.</td>
</tr>
<tr>
<td>Subject</td>
<td>Modify the email subject line.</td>
</tr>
<tr>
<td>Body</td>
<td>Modify, add, or remove the default text. The yellow box at the top of the</td>
</tr>
<tr>
<td></td>
<td>page lists special text strings you can use in the message body. Console</td>
</tr>
<tr>
<td></td>
<td>replaces the strings with the appropriate value in the actual email. The</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> To ensure that information displays correctly in the email, edit</td>
</tr>
<tr>
<td></td>
<td>both the plain text and HTML code versions of the template.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>available text strings differ depending on the type of template (transfer</td>
</tr>
<tr>
<td></td>
<td>start, transfer success, or transfer error)</td>
</tr>
</tbody>
</table>

For an example of how to edit the plain text version of the template, see *Email Template Example: Creating a Simple Notification for a Successful Transfer* on page 134.

4. Click **Save**.

5. Click **Edit HTML Template**.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Template name</td>
<td>Modify the name of the template displayed in Console.</td>
</tr>
<tr>
<td>From Name</td>
<td>Enter the name displayed as the email sender.</td>
</tr>
<tr>
<td>Reply-to Address</td>
<td>Enter the email address receiving replies from the email recipient.</td>
</tr>
<tr>
<td>Subject</td>
<td>Modify the email subject line.</td>
</tr>
<tr>
<td>Body</td>
<td>Modify, add, or remove the default text. The yellow box at the top of the</td>
</tr>
<tr>
<td></td>
<td>page lists special text strings you can use in the message body. Console</td>
</tr>
<tr>
<td></td>
<td>replaces the strings with the appropriate value in the actual email. The</td>
</tr>
<tr>
<td></td>
<td>available text strings differ depending on the type of template (transfer</td>
</tr>
<tr>
<td></td>
<td>start, transfer success, or transfer error)</td>
</tr>
</tbody>
</table>

For an example of how to edit the HTML code of the template, see *Email Template Example: Adding Company Branding to Your Template* on page 135.

6. Click **Save**.

7. Optional: Test the email template. Enter an email address in the field and click **Send Test Email**.

8. Optional: Make this template the default template.
   Return to the Email Templates page by clicking the **Email Templates** tab. Find your renamed email template and click **default**.

![Email Notifications for Transfer Error](image)

You can take the following actions for the new template:

- Set this template as your personal default from your **Personal Preferences** page.
- Select this template when creating a transfer.
- Select this template for an **Advanced Ruleset**.
Displaying Email Notification Templates for an Email Address

See a list of email notifications enabled for a given email address. The results show the email recipient's node endpoints, smart transfers, and user preferences. The results also list the email templates selected as the default for each transfer event (start, success, failure).

1. Go to **Notifications > Email Templates**
2. In the Search field, enter the email address (full or partial) and click **Search**.

In the example below, the results display all pre-configured email notification templates related to the email address "jdean".

![Configured Email Notifications Matching 'aspera_user'](image)
Adding Nodes to Console

Overview: Adding Nodes to Console

In Console, the term *node* is defined as a computer that has an Aspera transfer product installed and is enabled to make transfers. A node can be a managed node (typically a local system) or an unmanaged node.

- Managed Node: Console can initiate transfers with this node, monitor this node's activity, and configure settings for this node.
- Unmanaged Node: Console can only initiate transfers with this node. Console cannot monitor or configure this node.

You can also add an Aspera Transfer Cluster to Console. A transfer cluster is a provisioned cluster of Aspera servers in the cloud that autoscales with traffic and use. Console can monitor transfers running on managed clusters using access key authentication for each node. You can also run reports on managed clusters. For more information, see Adding Managed Clusters to Console on page 34.

Adding Managed Nodes

To add a managed node to Console, you must first configure the node machine. The following overview walks through the process of adding a managed node:

1. Add an admin account on the node machine.
2. Set up the Aspera Node API user on the node machine.
3. Create a new managed node in Console.
4. Authorize access to the node.
5. Set up permissions for the node.

Adding Unmanaged Nodes

To add an unmanaged node to Console, you do not need to have access to the node machine. The following overview walks through the process of adding an unmanaged node:

1. Create a new unmanaged node in Console.
2. Authorize access to the node.
3. Set up permissions for the node.

Adding an Administrative Account to a Windows Machine

The node machine that you add to IBM Aspera Console as a managed node must have a properly configured administrative account. The following instructions assume that you are logged into the node machine you want to add to Console.

**Important:** These instructions require that your Aspera service account (svcAspera, by default) be set up as a transfer user on the server. If the node's transfer product was installed by upgrading from a previous installation of IBM Aspera Enterprise Server or IBM Aspera Connect Server, a transfer user corresponding to the service account is created automatically. However, if it was a "clean" install (not an upgrade from a previous installation), only the service account is created, not the corresponding transfer user. In this case, create the transfer user manually using the GUI. For more information on creating the transfer user, see the IBM Aspera Enterprise Server Admin Guide.

1. Configure the node machine's firewall as described in System and Firewall Requirements on page 8.
2. Configure the user's docroot settings.
   Click the **Docroot** tab. Select **Override** for Absolute Path. Leave the field blank to allow the user to override default docroots under the node- or group-level setting.
3. **Add a node user associated with the svcAspera system user.**

   Console authenticates to the node machine using a Node API username and password. The following command creates a Node API user and password and associates it with the system user you created.

   ```shell
   > cd C:\Program Files (x86)\Aspera\Enterprise Server\bin
   > asnodeadmin.exe -a -u node_api_username -p node_api_passwd -x svcAspera --acl-set impersonation
   ```

   Adding, modifying, or deleting a node-user triggers automatic reloading of the user database and the node's configuration and license files. For more information on the Node API, see your transfer server's administrator guide.

   🔄 **Note:** If the transfer server on your node is running a transfer product before 3.5.5, the node is not recent enough to support setting the "impersonation" ACL. You must upgrade the node to 3.5.5+ or obtain a patch from Aspera Support.

4. **Verify that you correctly added the node user.**

   ```shell
   > asnodeadmin.exe -l
   ```

   The output should look like the following:

<table>
<thead>
<tr>
<th>user</th>
<th>system/transfer user</th>
<th>acls</th>
</tr>
</thead>
<tbody>
<tr>
<td>node_user</td>
<td>svcAspera</td>
<td>[impersonation]</td>
</tr>
</tbody>
</table>

5. **Restart the Aspera Node API and Aspera Central services to load changed settings.**

   Go to **Control Panel > Administrative Tools > Services**, right-click **Aspera NodeD** and select **Restart**. Do the same for **Aspera Central**.

Now that your node machine is configured for use with Console, open the Console interface and create a new managed node following the instructions in *Creating a Managed Node in Console* on page 26.

---

**Adding an Administrative Account to an OS X Machine**

IBM Aspera Console supports the following versions of Mac OS X:

- 10.7 (Lion)
- 10.8 (Mountain Lion)
- 10.9 (Mavericks)
- 10.10 (Yosemite)

The node machine that you add to IBM Aspera Console as a managed node must have a properly configured administrative account. The following instructions assume that you are logged into the node machine you want to add to Console.

⚠️ **Important:** These instructions require that your Aspera service account (svcAspera, by default) be set up as a transfer user on the server. If the node's transfer product was installed by upgrading from a previous installation of IBM Aspera Enterprise Server or IBM Aspera Connect Server, a transfer user corresponding to the service account is created automatically. However, if it was a "clean" install (not an upgrade from a previous installation), only the service account is created, not the corresponding transfer user. In this case, create the transfer user manually using the GUI. For more information on creating the transfer user, see the *IBM Aspera Enterprise Server Admin Guide*.

1. **Configure the node machine's firewall as described in System and Firewall Requirements on page 8.**

2. **Create an administrative account on the OS X node machine for use with Console.**

   Go to **System Preferences > Users & Groups**. Click the lock button and enter your admin credentials to make changes. Click the add button. Select **Administrative** from the New Account drop-down menu. Name the account...
"console_user". Select **Use separate password**, then enter and confirm a password for the account. Click **Create User**.

3. Enable the administrative account as a root user.

   Go to **System Preferences > Users & Groups**. Click the lock button and enter your admin credentials to make changes. Click **Login Options** (bottom of left panel). Then, click the **Edit** or **Join** button next to Network Account Server. Click **Open Directory Utility**. In the Directory Utility window, click the lock button and enter an administrator account and password to make changes. From the menu bar, select **Edit > Enable Root User**. Enter a password in the **Password** and **Verify** fields, and click **OK**.

   **Note:** If your node runs El Capitan (OS X 10.11), you must also create the file `/var/root/.ssh/environment` with the following content:

   ```
   PATH=/bin:/usr/bin:/usr/sbin:/sbin:/usr/local/sbin:/usr/local/bin:/opt/pkgconfig/bin:/Library/Aspera/bin:/Library/Aspera/sbin
   ```

4. Add a node user associated with the system user.

   Console authenticates to the node machine using a Node API username and password. The following command creates a Node API user and password and associates it with the system user you created.

   ```
   $ sudo /Library/Aspera/bin/asnodeadmin -a -u node_api_username -p node_api_passwd -x svcAspera --acl-set impersonation
   ```

   Adding, modifying, or deleting a node-user triggers automatic reloading of the user database and the node's configuration and license files. For more information on the Node API, see your transfer server's administrator guide.

   **Note:** If the transfer server on your node is running a transfer product before 3.5.5, the node is not recent enough to support setting the "impersonation" ACL. You must upgrade the node to 3.5.5+ or obtain a patch from Aspera Support.

5. Verify that you correctly added the node user.

   Run the following command:

   ```
   $ sudo /Library/Aspera/bin/asnodeadmin -l
   ```

   The output should look like the following:

<table>
<thead>
<tr>
<th>user</th>
<th>system/transfer user</th>
<th>acls</th>
</tr>
</thead>
<tbody>
<tr>
<td>node_user</td>
<td>console</td>
<td>[impersonation]</td>
</tr>
</tbody>
</table>

6. Restart the Aspera Node API and Aspera Central services to load changed settings.

   ```
   $ sudo launchctl stop com.aspera.asperacentral
   $ sudo launchctl start com.aspera.asperacentral
   $ sudo launchctl stop com.aspera.asperanoded
   $ sudo launchctl start com.aspera.asperanoded
   ```

   Now that your node machine is configured for use with Console, open the Console interface and create a new managed node following the instructions in *Creating a Managed Node in Console* on page 26.

---

**Adding an Administrative Account to a Linux Machine**

IBM Aspera Console supports Red Hat, Solaris, and FreeBSD managed nodes.

The node machine that you add to IBM Aspera Console as a managed node must have a properly configured administrative account. The following instructions assume that you are logged into the node machine you want to add to Console.
**Important:** These instructions require that your Aspera service account (svcAspera, by default) be set up as a transfer user on the server. If the node's transfer product was installed by upgrading from a previous installation of IBM Aspera Enterprise Server or IBM Aspera Connect Server, a transfer user corresponding to the service account is created automatically. However, if it was a "clean" install (not an upgrade from a previous installation), only the service account is created, not the corresponding transfer user. In this case, create the transfer user manually using the GUI. For more information on creating the transfer user, see the IBM Aspera Enterprise Server Admin Guide.

1. Configure the node machine's firewall as described in *System and Firewall Requirements* on page 8.

2. Create an administrative user account on the Linux node machine.

   ```shell
   # useradd console_username
   ```

   Give the user account a password.

   ```shell
   # passwd console_username
   ```

   **Note:** This user is meant to be used for reporting and administrative purposes only. It cannot start transfers.

3. Add the user to `aspera.conf`. Execute the following command:

   ```shell
   /opt/aspera/bin/asconfigurator -x "set_user_data;user_name,console_username;absolute,/
   ```

4. Give the user permission to write to the file `aspera.conf`.

   ```shell
   $ chown console_username:console_username /opt/aspera/etc/aspera.conf
   ```

5. Add a node user associated with the system user.

   Console authenticates to the node machine using a Node API username and password. The following command creates a Node API user and password and associates it with the system user you created.

   ```shell
   # /opt/aspera/bin/asnodeadmin -a -u node_api_username -p node_api_passwd -x console_username --acl-set impersonation
   ```

   Adding, modifying, or deleting a node-user triggers automatic reloading of the user database and the node's configuration and license files. For more information on the Node API, see your transfer server's administrator guide.

   **Note:** If the transfer server on your node is running a transfer product before 3.5.5, the node is not recent enough to support setting the "impersonation" ACL. You must upgrade the node to 3.5.5+ or obtain a patch from Aspera Support.

6. Verify that you correctly added the node user.

   Run the following command:

   ```shell
   # /opt/aspera/bin/asnodeadmin -l
   ```

   For example, if your `node_api_username` is `node_user` and your `console_user` is `console`, then the output should look like the following:

<table>
<thead>
<tr>
<th>user</th>
<th>system/transfer user</th>
<th>acls</th>
</tr>
</thead>
<tbody>
<tr>
<td>node_user</td>
<td>console</td>
<td>[impersonation]</td>
</tr>
</tbody>
</table>

7. Restart the Aspera Node API and Aspera Central services to load changed settings.

   ```shell
   # service asperanoded restart
   # service asperacentral restart
   ```
Now that your node machine is configured for use with Console, open the Console interface and create a new managed node following the instructions in Creating a Managed Node in Console on page 26.

Creating a Managed Node in Console

It is best practice to keep all your nodes up to date with the latest version of IBM Aspera Enterprise Server, IBM Aspera Connect Server, or IBM Aspera Point-to-Point Client.

To verify your node machine's product version, run the following command on your node machine:

```
$ ascp -A
```

If you have an older version, you can download the latest from http://asperasoft.com/downloads.

1. Go to Nodes and click New Managed Node.
2. Enter your managed node's IP Address, Name, SSH Port, and Node API Port.

   **Important:** When adding the Console server itself as a managed node, don't enter 127.0.0.1 as the IP address unless it is the only node you are planning to add to the console. Even then it is not recommended. Do this only if your network or firewall configuration interferes with Console communicating with its own external address.

4. Add the node to a failover group.
   Select Enable failover and load balancing for Console-initiated transfers on this node. Add the node to an existing group or select enter new name from the Failover Group Name drop-down menu to create a new group. If you select enter new name, enter a new failover group name in the prompt. For more information on failover groups, see Configure Failover Groups on page 67.
5. Optional: Create the three default Console groups. Select Create default Console groups.
   The default transfer Console groups have the following permissions:
   - **Transfer Admin:** Users in this group can manage transfers on this node. This includes initiating, canceling, and deleting transfers.
   - **Transfer Initiator:** Users in this group can only initiate transfers.
   - **Transfer Monitor:** Users in this group can only monitor transfers.

   For more information on Console groups, see Creating Console Groups on page 43
6. When finished, click Create.

   Console redirects you to the Credentials page. For more information on configuring admin credentials, see Updating a Node's Admin Credentials on page 26.

Updating a Node's Admin Credentials

If you have not already configured an administrative account on your node machine for use with IBM Aspera Console, see the following instructions:

- Windows node: Adding an Administrative Account to a Windows Machine on page 22.
- OS X node: Adding an Administrative Account to an OS X Machine on page 23.

IBM Aspera Console connects to a managed node through SSH for file browsing and node configuration. You must update Console with the node's SSH credentials before Console can access the node.

If Console automatically redirected you to the Admin Credentials page as part of the process of adding a new managed node, skip the first step.

1. Go to Nodes. Click the edit link for the managed node, and click the Credentials tab.
Adding Nodes to Console

2. Select **Edit Credentials**.
3. If you want to configure the node from Console, enter the node machine's SSH login credentials.

   Enter the administrative account username and password to allow Console to connect to the node machine. You can authenticate the account in one of two ways:

   - **Password authentication**: Enter the account password.
   - **Public key authentication**: Select **Use SSH Key** and select your uploaded key. To use public key authentication, you must have your SSH private key configured in Console. For instructions, on how to configure SSH keys in Console, see [SSH Keys](#) on page 94.

4. Enter the node machine's Node API credentials.
5. Click **Update**.
6. Click **Test Credentials** to make sure Console has a working connection to the node.

   If successful, the message "Successfully connected to node via SSH and Node API" appears in green at the top of the page.

A connection is established between Console and your managed node. To edit or remove a node, go to **Nodes** for a list of managed nodes and click **edit** or **delete** for the designated node.

Adding Unmanaged Nodes

It is best practice to keep all your nodes up to date with the latest version of IBM Aspera Enterprise Server, IBM Aspera Connect Server, or IBM Aspera Point-to-Point Client. Verify the machine's product version with the administrator of the node.

1. Go to **Nodes**. Click **List Unmanaged Nodes**.
2. Click **New Unmanaged Node**.
3. Enter the node's **Address** (IP or domain name), **Name**, and **SSH Port** number.
4. Select the **SSH Encryption** method.
5. Click **Create** when finished.
6. To verify that your new node has been created, select **List Unmanaged Nodes** and look for your unmanaged node in the table.

A connection should be established between Console and your unmanaged node. To edit or remove a node, go to **Nodes** and click **List Managed Nodes** for a list of managed nodes. Click **edit** or **delete** for the designated node.

Adding Endpoints

An **endpoint** is a combination of a login account (an account on the node that has been configured for FASP file transfers) and a node address. For example, the endpoint **userA@nodeA** is User A's login account on Node A. Endpoints enable a user to create a transfer without entering login credentials. Sharing an endpoint with a user without login credentials allows that user to send or receive files without compromising security.

Wildcards can also be used within endpoints. For example, the wildcarded endpoint **@nodeA** (interpreted as "all logins at node A") is automatically created when adding the node (in this case, **nodeA**) to Console. Using endpoints with wildcards, you can monitor all transfers on a given node as well as prompt for a specific login account when a user initiates a transfer on the node.

User-provided credentials are stored in the user's **Saved Endpoints** under the **Preferences** tab.

**Tip:** To use domain names as transfer endpoints, create an unmanaged node using a domain name, then add an Endpoint to this unmanaged node.

1. Create a new endpoint.
   - Go to **Nodes**. Click the **edit** link for the designated node. Next, click the **Endpoints** tab and click **Add Endpoint**.
2. Enter the following information:
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a descriptive name for this endpoint.</td>
</tr>
<tr>
<td>Login</td>
<td>Enter the login name of the node machine user.</td>
</tr>
<tr>
<td>Password / SSH Key</td>
<td>Authenticate the account with either a password or a saved public key. To authenticate by password, enter the account password. Otherwise, check Use SSH Key and select your uploaded key to authenticate with public key. To use public key authentication, you must have your SSH private key configured in Console. For instructions, on how to configure SSH keys in Console, see SSH Keys on page 94. <strong>Important:</strong> When using SSH key authentication, make sure that the key file on the node is not a shared key. On the node computer, the key file should be a &quot;private&quot; key in the specified user account.</td>
</tr>
<tr>
<td>Email address</td>
<td>Enter an email address to receive notifications of the transfer activity on this endpoint. This option accepts multiple email addresses.</td>
</tr>
</tbody>
</table>

3. When finished, click Create.
   The created endpoint appears in the node's maintenance page under the Endpoints tab. If Password Saved is selected, the endpoint contains password information or an SSH key.

4. Verify your endpoint connection works.
   In the list of endpoints, find this endpoint and click test. On the following page, click Test Connecting to Host. If successful, a confirmation message appears in green at the top of the page.

The endpoint is now configured and permitted users are not required to enter a password to use this endpoint. To edit or remove an endpoint, click edit or delete.

### Set a Global Docroot for the Node

A document root, or docroot, is the area of a machine that a system user has permission to access. Setting docroots are important for maintaining security by keeping unqualified users from accessing confidential information.

1. Go to Nodes and click edit for the node.
2. Go to Accounts and click edit for the user or group you want to configure.
3. Expand the Docroot configuration section and click Browse. Choose the file directory you want to set as the docroot.

   The docroot is a security feature that allows you to restrict the area asperawatchfolderd can access. If you need to access the entire file system, you can set the docroot path as / or leave it empty. The directory you choose is configured in the aspera.conf configuration file on the transfer node.

   For example, if you configure the docroot path to be / for the user root, configuring the docroot adds the following configuration to the <aaa> section of aspera.conf:

   ```xml
   <aaa>
     <realms>
       <realm>
         <users>
           <user>
             <name>root</name>
             <file_system>
               <access>
                 <paths>
                   <path>
                     <absolute>/</absolute>
                   </path>
                 </paths>
               </file_system>
             </access>
           </user>
         </users>
       </realm>
     </realms>
   </aaa>
   ```
4. Click **Save changes**.
Managing the Localhost Node

The Localhost Node

IBM Aspera Console On Demand is automatically configured with a localhost node with two pre-configured accounts for your use: xfer and xfer2. Go to Nodes and the 127.0.0.1 node appears on the list. Click edit and click Accounts to view the two accounts. Both are assigned a document root on the server in /mnt/ephemeral/data/$username.

Configuring Accounts for Transfers with a Standard Aspera Client

The xfer and xfer2 accounts are designed as transfer accounts to be used with the Shares web application, and therefore have been configured to require a valid token. If you want to transfer with the xfer or xfer2 account using a standard Aspera client, disable token authorization. Go to the Accounts tab on the localhost node, click edit for the account. Expand the Authorizations section and clear Override for Incoming Transfers and Outgoing Transfers.

You can also create new transfer users on this node. To create a new transfer account, see Editing the User or Group on a Node on page 30.

Adding Unmanaged Nodes

It is best practice to keep all your nodes up to date with the latest version of IBM Aspera Enterprise Server, IBM Aspera Connect Server, or IBM Aspera Point-to-Point Client. Verify the machine's product version with the administrator of the node.

1. Go to Nodes. Click List Unmanaged Nodes.
2. Click New Unmanaged Node.
3. Enter the node's Address (IP or domain name), Name, and SSH Port number.
4. Select the SSH Encryption method.
5. Click Create when finished.
6. To verify that your new node has been created, select List Unmanaged Nodes and look for your unmanaged node in the table.

A connection should be established between Console and your unmanaged node. To edit or remove a node, go to Nodes and click List Managed Nodes for a list of managed nodes. Click edit or delete for the designated node.

Editing the User or Group on a Node

IBM Aspera Console can configure user and group account settings for managed nodes that have valid admin credentials saved in Console. For more information on updating admin credentials, see Updating a Node's Admin Credentials on page 26.

1. Go to Nodes and click edit for the node you want to edit. Click Accounts.
2. Make sure that the group or user you want to configure has already been created and is available on the node machine. Console automatically detects new groups and users and lists them under the node's Accounts tab, but if the group or user is not listed, click Add Group or Add User.
3. Depending on whether you want to configure a node user or a node group, select Users or Groups.
4. Select the edit link for the user or group account you want to edit.
5. Configure the user or group account's transfer options. For more detailed information on these options, see Node Account-Level Configuration Options on page 146.
Set a Docroot for a Node User or Group

A document root, or docroot, is the area of a machine that a system user has permission to access. Setting docroots are important for maintaining security by keeping unqualified users from accessing confidential information. To set a docroot for a node user or group, you must have already added them into Console. For more information about adding node users or groups to Console, see Editing the User or Group on a Node on page 30.

1. Go to Nodes and click edit for the node.
2. Go to Accounts and click edit for the user or group you want to configure.
3. Expand the Docroot configuration section and click Browse. Choose the file directory you want to set as the docroot.

The docroot is a security feature that allows you to restrict the area asperawatchfolderd can access. If you need to access the entire file system, you can set the docroot path as / or leave it empty. The directory you choose is configured in the aspera.conf configuration file on the transfer node.

For example, if you configure the docroot path to be / for the user root, configuring the docroot adds the following configuration to the <aaa> section of aspera.conf:

```xml
<aaa>
  <realms>
    <realm>
      <users>
        <user>
          <name>root</name>
          <file_system>
            <access>
              <paths>
                <path>
                  <absolute>/</absolute>
                </path>
              </paths>
            </access>
          </file_system>
        </user>
      </users>
    </realm>
  </realms>
</aaa>
```
4. Click **Save changes**.

---

## Adding Endpoints

An endpoint is a combination of a login account (an account on the node that has been configured for FASP file transfers) and a node address. For example, the endpoint **userA@nodeA** is User A's login account on Node A. Endpoints enable a user to create a transfer without entering login credentials. Sharing an endpoint with a user without login credentials allows that user to send or receive files without compromising security.

Wildcards can also be used within endpoints. For example, the wildcarded endpoint **@nodeA** (interpreted as "all logins at node A") is automatically created when adding the node (in this case, **nodeA**) to Console. Using endpoints with wildcards, you can monitor all transfers on a given node as well as prompt for a specific login account when a user initiates a transfer on the node.

User-provided credentials are stored in the user's **Saved Endpoints** under the **Preferences** tab.

#### Tip: To use domain names as transfer endpoints, create an unmanaged node using a domain name, then add an Endpoint to this unmanaged node.

1. Create a new endpoint.
   - Go to **Nodes**. Click the **edit** link for the designated node. Next, click the **Endpoints** tab and click **Add Endpoint**.
2. Enter the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a descriptive name for this endpoint.</td>
</tr>
<tr>
<td>Login</td>
<td>Enter the login name of the node machine user.</td>
</tr>
<tr>
<td>Password / SSH Key</td>
<td>Authenticate the account with either a password or a saved public key. To authenticate by password, enter the account password. Otherwise, check <strong>Use SSH Key</strong> and select your uploaded key to authenticate with public key. To use public key authentication, you must have your SSH private key configured in Console. For instructions, on how to configure SSH keys in Console, see <strong>SSH Keys</strong> on page 94. <strong>Important:</strong> When using SSH key authentication, make sure that the key file on the node is not a shared key. On the node computer, the key file should be a &quot;private&quot; key in the specified user account.</td>
</tr>
<tr>
<td>Email address</td>
<td>Enter an email address to receive notifications of the transfer activity on this endpoint. This option accepts multiple email addresses.</td>
</tr>
</tbody>
</table>

3. When finished, click **Create**.
   - The created endpoint appears in the node's maintenance page under the Endpoints tab. If **Password Saved** is selected, the endpoint contains password information or an SSH key.
4. Verify your endpoint connection works.
   - In the list of endpoints, find this endpoint and click **test**. On the following page, click **Test Connecting to Host**. If successful, a confirmation message appears in green at the top of the page.

The endpoint is now configured and permitted users are not required to enter a password to use this endpoint. To edit or remove an endpoint, click **edit** or **delete**.
Configuring Virtual Links

Configure Virtual Links (Vlink) on a node to create a "virtual" bandwidth cap for the node. Transfer sessions assigned to the same Vlink take up equal shares of the capped bandwidth.

1. Go to **Nodes**, find the desired node, and click **edit**. Click **Vlinks > New Vlink**.
2. Enter a number for the **Vlink ID** and name the Vlink. Sessions assigned with the same ID share the same bandwidth cap.
3. Select **True** to activate the Vlink.
4. Enter a value for the capacity. When applying this Vlink to a transfer, the transfer's bandwidth will be restricted by this value.
5. Click **Create**.

After creating a new Vlink, you have the option of configuring the Vlink to run on a schedule by clicking **Edit Time Varying Schedule** and then **New Schedule**. For more information on scheduling Vlinks, see *Scheduling Virtual Links* on page 33.

Scheduling Virtual Links

After creating a new virtual link, you have the option of configuring the Vlink to run on a schedule.

1. Go to **Nodes**, find the desired node, and click **edit**. Click **Vlinks**, find the desired Vlink, and click **edit**.
2. Click **Edit Time Varying Schedule** and click **New Schedule**.

Configure the following options.

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>On the following days</td>
<td>Select the days or set of days for which the bandwidth rate cap is enforced.</td>
</tr>
<tr>
<td>From the following time</td>
<td>Enter a time to start the bandwidth rate cap.</td>
</tr>
<tr>
<td>To the following time</td>
<td>Enter a time to stop the bandwidth rate cap.</td>
</tr>
<tr>
<td>Set the rate to</td>
<td>Enter a value for the scheduled virtual bandwidth cap. When applying this Vlink to a transfer, the transfer's bandwidth will be restricted by this value based on the configured schedule.</td>
</tr>
</tbody>
</table>

**Note:** Overlapping time schedules are not supported. If there are overlapping schedules, they are not accurately reflected in the Vlinks chart, and precedence is indeterminate.

3. Click **Update**.
Adding Managed Clusters to Console

Adding a Managed Cluster to Console

A transfer cluster is a provisioned cluster of Aspera servers in the cloud that autoscales with traffic and use. Console can monitor transfers running on managed clusters using access key authentication for each node. You can also run reports on managed clusters. For information on configuring the Aspera Transfer Cluster, see the IBM Aspera Transfer Cluster Manager Admin Guide.

To add a cluster to Console, follow the instructions below.

1. Go to Nodes and click the New Managed Cluster button.
2. Enter the cluster's domain name address in the Domain Name field.
3. Name the cluster in Console.
4. Configure the cluster API port. Port 443 is the default port for communicating with Aspera clusters.
5. If you want to use HTTPS to connect to node, select Use HTTPS to connect to node. Aspera recommends using this feature for security reasons.
6. If you want Console to verify the SSL certificate of cluster nodes, select Require signed SSL certificate.
7. Click Create.

Console redirects you to the Credentials page to add access keys for authentication. For more information about adding access keys, see Adding Access Keys to a Managed Cluster on page 34.

Adding Access Keys to a Managed Cluster

Aspera Transfer Clusters use access keys to authenticate to other Aspera products. Each transfer cluster can have multiple access keys. For more information about access keys and access key management, see the entries in the IBM Aspera Transfer Cluster Manager Admin Guide: Managing a Cluster section.

Follow the instructions below to add access keys to a managed cluster.

1. Go to Nodes. Click the edit link for the managed cluster and click the Credentials tab.
2. Click the Add Access Key button.
3. Give the access key a name to differentiate it from other keys.
4. Enter the ID and Secret associated with the key.
5. Click Update.
   Console redirects you to the list of Access Keys for this cluster. Verify that your new access key is in the list.
6. Test your new access key by selecting test.
Managing Nodes

Editing the User or Group on a Node

IBM Aspera Console can configure user and group account settings for managed nodes that have valid admin credentials saved in Console. For more information on updating admin credentials, see Updating a Node's Admin Credentials on page 26.

1. Go to Nodes and click edit for the node you want to edit. Click Accounts.
2. Make sure that the group or user you want to configure has already been created and is available on the node machine. Console automatically detects new groups and users and lists them under the node's Accounts tab, but if the group or user is not listed, click Add Group or Add User.
3. Depending on whether you want to configure a node user or a node group, select Users or Groups.
4. Select the edit link for the user or group account you want to edit.
5. Configure the user or group account's transfer options. For more detailed information on these options, see Node Account-Level Configuration Options on page 146.

6. When you are finished, click Save changes.

Set a Docroot for a Node User or Group

A document root, or docroot, is the area of a machine that a system user has permission to access. Setting docroots are important for maintaining security by keeping unqualified users from accessing confidential information. To set a docroot for a node user or group, you must have already added them into Console. For more information about adding node users or groups to Console, see Editing the User or Group on a Node on page 30.

1. Go to Nodes and click edit for the node.
2. Go to Accounts and click edit for the user or group you want to configure.
3. Expand the Docroot configuration section and click **Browse**. Choose the file directory you want to set as the docroot.

The docroot is a security feature that allows you to restrict the area asperawatchfolderd can access. If you need to access the entire file system, you can set the docroot path as / or leave it empty. The directory you choose is configured in the aspera.conf configuration file on the transfer node.

For example, if you configure the docroot path to be / for the user root, configuring the docroot adds the following configuration to the <aaa> section of aspera.conf:

```xml
<aaa>
  <realms>
    <realm>
      <users>
        <user>
          <name>root</name>
          <file_system>
            <access>
              <paths>
                <path>
                  <absolute>/</absolute>
                </path>
              </paths>
            </access>
          </file_system>
        </user>
      </users>
    </realm>
  </realms>
</aaa>
```

4. Click **Save changes**.

## Converting Legacy Nodes

A node running an Aspera transfer product with a version prior to 3.4.6 is considered a legacy node and continues to report to Console using the legacy mechanism. To take full advantage of Console's architecture, upgrade the Aspera transfer products on legacy nodes to the latest version and convert legacy nodes to use the Node API in the Console 3.0+ Node Maintenance page.

**Important:** Once you convert from a legacy node to a regular node, you cannot revert back to a legacy node.

1. On the node machine, upgrade your Aspera transfer product to its latest version. To check the version of the running transfer product, run the following command:

   ```
   # ascp -A
   ```

2. Configure an administrative account and API user on your node machine.
   - Windows Node: See *Adding an Administrative Account to a Windows Machine* on page 22.
   - OS X Node: See *Adding an Administrative Account to an OS X Machine* on page 23.

3. From the Console menu, select **Nodes**. Select the **edit** link for a node that Console shows is a "Legacy Node".

4. Click **Node API** and enter your Node API credentials for the node machine.
   
   For more information on updating a node's admin credentials, see *Updating a Node's Admin Credentials* on page 26.

5. Click **Convert** and then click **Convert to use Node API**. When prompted by the browser to confirm conversion, click **OK**.
Note: If you do not see Convert to use Node API, make sure you have correctly configured your Node API credentials on the machine and entered them into Console. Edit the node in Console, click Credentials, and click Test Credentials.

6. Select Open all to expand all available configuration options. Clear any overridden values in the Database and Transfer Server sections.
7. Select Save changes.
8. Restart the Aspera Central service on the node.
   - Linux:
     ```
     # service asperacentral restart
     ```
   - OS X:
     ```
     $ sudo launchctl stop com.aspera.asperacentral
     $ sudo launchctl start com.aspera.asperacentral
     ```
Console now displays your legacy node as a regular node in the node list.

### Configuring Virtual Links

Configure Virtual Links (Vlink) on a node to create a "virtual" bandwidth cap for the node. Transfer sessions assigned to the same Vlink take up equal shares of the capped bandwidth.

1. Go to Nodes, find the desired node, and click edit. Click Vlinks > New Vlink.
2. Enter a number for the Vlink ID and name the Vlink. Sessions assigned with the same ID share the same bandwidth cap.
3. Select True to activate the Vlink.
4. Enter a value for the capacity. When applying this Vlink to a transfer, the transfer's bandwidth will be restricted by this value.
5. Click Create.

After creating a new Vlink, you have the option of configuring the Vlink to run on a schedule by clicking Edit Time Varying Schedule and then New Schedule. For more information on scheduling Vlinks, see Scheduling Virtual Links on page 33.

### Scheduling Virtual Links

After creating a new virtual link, you have the option of configuring the Vlink to run on a schedule.

1. Go to Nodes, find the desired node, and click edit. Click Vlinks, find the desired Vlink, and click edit.
2. Click Edit Time Varying Schedule and click New Schedule.

Configure the following options.

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>On the following days</td>
<td>Select the days or set of days for which the bandwidth rate cap is enforced.</td>
</tr>
<tr>
<td>From the following time</td>
<td>Enter a time to start the bandwidth rate cap.</td>
</tr>
<tr>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>To the following time</td>
<td>Enter a time to stop the bandwidth rate cap.</td>
</tr>
<tr>
<td>Set the rate to</td>
<td>Enter a value for the scheduled virtual bandwidth cap. When applying this Vlink to a transfer, the transfer's bandwidth will be restricted by this value based on the configured schedule.</td>
</tr>
</tbody>
</table>

⚠️ **Note:** Overlapping time schedules are not supported. If there are overlapping schedules, they are not accurately reflected in the Vlinks chart, and precedence is indeterminate.

3. Click **Update**.
Enabling S3 Storage Using Console

IBM Aspera Console can use S3 storage for a node transfer user by specifying the storage in the user docroot. Use this user to transfer files to and from your S3 storage. The steps below assume the following:

- You have purchased and booted up your Aspera On Demand product.
- You have created an S3 bucket.
- You have permissions to create IAM roles or change the policies of your IAM.
- You know how to SSH as root to your Aspera On Demand instance.

1. In Console, select a node and edit its transfer user from the Accounts tab.
2. Expand Docroot, click Override, and paste the S3 docroot for that user using the following syntax:

   S3://access_id:secret_key@s3.amazonaws.com/my_bucket/my_path

   Use URL encoding for special characters in your S3 Access ID and secret key. For example, encode a slash character (\/) by replacing it with %2F and encode a plus character (+) by replacing it with %2B.

   Click on the Save Changes button.

   For more information about setting a user’s docroot, see Editing the User or Group on a Node on page 30.

3. Restart the Aspera NodeD service on the node.

   SSH into the node and run the following command:

   # ssh -i identity_file -p 33001 ec2-user@ec2_host_ip
   # service asperanoded restart

   Configure advanced S3 storage settings and test your configuration.

4. Optional: Enable advanced S3 storage settings.

   - Enable Reduced Redundancy Storage (RRS): Append the following to the docroot:

     ?storage-class=REDUCED_REDUNDANCY
For example, enter:

```
S3://access_id:secret_key@s3.amazonaws.com/my_bucket/my_path?storage-class=REDUCED_REDUNDANCY
```

- Enable S3 Server Side Encryption (SSE). Append the following to the docroot:

```
?server-side-encryption=AES256
```

For example, enter:

```
S3://access_id:secret_key@s3.amazonaws.com/my_bucket/my_path?server-side-encryption=AES256
```

5. Test your configuration. Perform a test transfer using an Aspera client to the account configured with the S3 docroot. For information on starting a transfer, see Starting a Simple Transfer on page 56.

---

**Enabling SoftLayer Storage Using Console**

IBM Aspera Console can use SoftLayer storage for a node transfer user. Use this user to transfer files to and from your SoftLayer storage.

⚠️ **Caution:** When transferring files larger than 64 MB to SoftLayer storage, an .aspera-segment directory is created at the destination. Do not move this directory or modify any files in it. Doing so may cause corruption or loss of data.

1. Go to **Nodes** and click the **edit** button for the node.
2. Go to **Accounts** and click **edit** for the account to configure with SoftLayer access.

   🔄 **Note:** You can also create a new account by clicking on the **Add User** button. For information on how to add a new account, see Editing the User or Group on a Node on page 30.

3. Enter the SoftLayer docroot.

   Expand **Docroot**, click **Override**, and paste the SoftLayer docroot for that user using the following syntax:

   ```
   swift://username:apikey@Object Storage URI/bucket_name?
   aspera.swift.endpoint.auth-path=%2Fauth%2Fv1.0
   ```

   Use URL encoding for special characters. For example, encode the colon (:) by replacing it with %3A.

4. Click on the **Save Changes** button.
5. Restart **asperanoded** on the node.
   SSH into the node and run the following command:
   
   ```
   # service asperanoded restart
   ```

6. Test your configuration. Perform a test transfer using an Aspera client to the account configured with the SoftLayer object storage docroot. For information on starting a transfer, see *Starting a Simple Transfer* on page 56.
Managing User Accounts

Accounts and Permissions

Definition of Terms

- **User**: A user is a Console login account with customizable access permissions.
- **Group**: A group defines the transfer permissions of all its users.
- **Transfer Path**: A transfer path consists of two endpoints, the transfer direction (one-way or two-way), and a set of permissions that authorize starting transfers, monitoring transfers, and enabling email notifications.

Overview

Console uses a combination of groups, transfer paths, and user accounts to manage user permissions. A user that belongs to a group inherits permissions defined within the groups it belongs to. A group's permissions are defined by its transfer paths. If you have a non-admin user and you want them to be able to see certain transfers, you need to add them to a group. This group must have one or more transfer paths that specify the kinds of transfers that members of the group are allowed to see or control.

Each group can contain one or more transfer paths. In the figure below, Group 1 contains two transfer paths, #1 and #2. A Console user inherits transfer permissions from all of the groups he or she belongs to. For example, Console User 2 belongs to both Group 1 and Group 2, and has the permissions to use Transfer Paths #1, #2, and #3.

Tip: Console administrators are able to view and control all transfers. They automatically inherit permissions of any existing Console groups. They can add, edit, and delete any nodes, Console users, and Console groups.

Default Console Groups

When adding a new node, you have the option of creating three default groups associated with that node.

<table>
<thead>
<tr>
<th>Group name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transfer Administrators</td>
<td>The users in this group can monitor, control, and set up email notifications of all transfers on the node. They can start simple and smart transfers between this node and any node, and share smart transfer templates with other users.</td>
</tr>
<tr>
<td>Transfer Initiators</td>
<td>The users in this group can start simple and smart transfers between this node and any node.</td>
</tr>
<tr>
<td>Transfer Monitors</td>
<td>The users in this group can monitor and set up email notifications of all transfers on this node.</td>
</tr>
</tbody>
</table>
Creating Console Groups and Users

For instructions on creating a new Console group, see Creating Console Groups on page 43.

For instructions on creating a new Console user, see Creating Console Users on page 44.

Creating Console Groups

Console uses a combination of groups, transfer paths, and user accounts to manage user permissions. A user that belongs to a group inherits permissions defined within the groups it belongs to. A group’s permissions are defined by its transfer paths. If you have a non-admin user and you want them to be able to see certain transfers, you need to add them to a group. This group must have one or more transfer paths that specify the kinds of transfers that members of the group are allowed to see or control.

Tip: Console administrators are able to view and control all transfers. They automatically inherit permissions of any existing Console groups. They can add, edit, and delete any nodes, Console users, and Console groups.

Important: You must first manually add a group to the node OS before you can add it in Console.

1. Go to Accounts > Groups and click New Group.
2. Enter the group name and a brief description. When finished, click Create. You are redirected to a page that allows you to configure the group.
3. Click Add Transfer Path.
   A transfer path determines a user’s permissions to create, initiate, and monitor transfers from one endpoint to another. A transfer path consists of two endpoints, the transfer direction (one-way or two-way), and a set of permissions that authorize starting transfers, monitoring transfers, and enabling email notifications.
4. Select the endpoints for the transfer path.
   For a unidirectional transfer path, set Endpoint 1 as your source endpoint and Endpoint 2 as your destination endpoint. Order does not matter for a bidirectional transfer path. If you specify a node user in an endpoint, users in the group are limited to monitoring only transfers on the node machine that involve the specified node user. An example of such a transfer is a transfer initiated using the specified node user's credentials. Selecting "Any" grants users transfer path permissions to all nodes.

   Important: When you select Any as an endpoint and permit users to start simple or smart transfers, users can enter arbitrary addresses for file transfers.
5. Choose the direction of the transfer path.
   A transfer path can be unidirectional or bidirectional.
   • Unidirectional (to): Console users can create, initiate, and monitor transfers initiated from Endpoint 1 to Endpoint 2 (depending on the transfer path permissions) but not the other way around.
   • Bidirectional (to/from): Console users can create, initiate, and monitor all transfers (depending on the transfer path permissions) between Endpoint 1 or Endpoint 2.
6. Choose the permissions you want to give users in the group.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Simple Transfers</td>
<td>Users can start a simple transfer.</td>
</tr>
<tr>
<td>Start Smart Transfers</td>
<td>Users can start smart transfers.</td>
</tr>
<tr>
<td>Create Smart Transfers</td>
<td>Users can to create a smart transfer template.</td>
</tr>
<tr>
<td>Share Smart Transfers</td>
<td>Users can to share smart transfer templates with other users.</td>
</tr>
<tr>
<td>Control Transfers started by others</td>
<td>Users can control other users' transfers. For example, they can stop, pause, and set the rate of a transfer, and so on.</td>
</tr>
</tbody>
</table>
### Item Description

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>View Transfers started by others</td>
<td>Users can view other users' transfers on the same transfer paths.</td>
</tr>
<tr>
<td>Opt-in to email notifications</td>
<td>Users can enable email notifications for this transfer path.</td>
</tr>
</tbody>
</table>

7. Optional: Enter a description for this transfer path.

8. When finished, click **Create**.
   The Editing Group Details screen displays the new transfer path in the Transfer Paths list. To modify or remove the transfer path, click **edit** or **delete**, respectively.

9. Add users to the group.
   Select a Console user from the members drop-down and click **Add**.

   **Tip:** Alternatively, you can assign group members through user management. See *Creating Console Users* on page 44.

10. Click **Update**.

### Creating Console Users

Console user is a Console login account with customizable access permissions. Except for administrator accounts, Console user permissions are managed through group assignment. A Console user inherits permissions from its groups.

**Note:** Console users are not directly related to the login account to a node.

1. Go to **Accounts > Users** and click **New User**.

2. Enter a login username, the user's first and last name and an email address. Set the user's time zone.

   **Important:** All activity on the Console is dated according to the user's time zone.

3. Optional: Select **Set password** to create a password for the user account. If you do not set a password, Console generates a temporary password for the account and emails the password to the user.

   **Tip:** You can change password requirements in the Console Password Options section. Go to **Configuration > Defaults** For more information on password requirements, see *Configuring Console Defaults* on page 112.

4. Optional: Disable user login by clearing **Active (allow user to login)**.

   If you wish to finish setting permissions for the user account before allowing the user to log in, disable the account by clearing **Active (allow user to login)**. To re-enable the account, return to these settings and select **Active (allow user to login)**. User login is enabled by default.

5. Optional: Disable reporting features for the user by clearing **Reports Allowed**.

6. When finished, click **Create**.

   The system sends an account creation notification email to the designated email with the account's username and password. If you do not set a password, Console generates a temporary password for the account and include that in the email.

The following step is only applicable when creating non-admin users. All admin users have full permissions to all groups and transfer paths. After creating a non-admin user, Console redirects you to the user permissions page.

7. Assign the user to Console groups.

   Assign the user to groups with the desired transfer-path permissions. To assign the user to a group, select a group from the drop-down menu and click **Add**. You can review the Console user's transfer permissions in a table listing all transfer paths accessible by this user.
Once the Console user account is created, users can log in to Console with the proper account credentials. To deactivate this account or make other changes to it, go to **Accounts > Users**. Locate the account you want to change in the list of all Console users.

- To deactivate or reactivate the account, change it to a Console administrator, or modify any of the basic account information, click **edit**.
- To modify transfer permissions and group membership, click **permissions**.
- To remove a Console user from the system, click **delete**.
Monitoring Console

The Console Dashboard

The Dashboard provides a quick overview of all transfer activities and the statuses of nodes for which you have monitoring permissions. It gives continuous updates and helps identify transfer and node problems.

Go to Dashboard. The Dashboard contains the following six panels:

**Current Transfers**

Current Transfers lists up to ten ongoing transfers on all monitored nodes. To view all active transfers, click the **Current Transfers** header.

<table>
<thead>
<tr>
<th>NAME</th>
<th>CONTACT</th>
<th>ETA</th>
<th>STATUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLES to Fedora</td>
<td>admin (console)</td>
<td>10:31am</td>
<td>35%</td>
</tr>
<tr>
<td>100MB</td>
<td>root (ssh)</td>
<td>10:31am</td>
<td>35%</td>
</tr>
<tr>
<td>NAB Demo Transfers</td>
<td>user01 (console)</td>
<td>10:38am</td>
<td>63%</td>
</tr>
<tr>
<td>Test Transfer</td>
<td>admin (console)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Scheduled Transfers**

Scheduled Transfers lists up to ten scheduled transfers on all monitored nodes. To view all scheduled transfers, click the **Scheduled Transfers** header.

<table>
<thead>
<tr>
<th>NAME</th>
<th>CONTACT</th>
<th>SCHEDULED START</th>
</tr>
</thead>
<tbody>
<tr>
<td>From New York to London</td>
<td>admin (console)</td>
<td>10:00am 7-Jul (r)</td>
</tr>
<tr>
<td>Weekly Transfer</td>
<td>admin (console)</td>
<td>10:29am 12-Jun (r)</td>
</tr>
</tbody>
</table>

**Recent Transfers**

Recent Transfers lists up to ten recent transfers on all managed nodes. To view all recent transfers, click the **Recent Transfers** header.
Recent Transfers

<table>
<thead>
<tr>
<th>NAME</th>
<th>CONTACT</th>
<th>ENDED</th>
<th>TRANSFERRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Transfer</td>
<td>admin (console)</td>
<td>10:35am</td>
<td>32.9 MB</td>
</tr>
<tr>
<td>NAB Demo Transfers</td>
<td>user01 (console)</td>
<td>10:34am</td>
<td>8 MB</td>
</tr>
<tr>
<td>SLES to Fedora</td>
<td>admin (console)</td>
<td>10:32am</td>
<td>10.7 MB</td>
</tr>
<tr>
<td>100MB</td>
<td>root (ssh)</td>
<td>10:30am</td>
<td>100 MB</td>
</tr>
</tbody>
</table>

Problem Transfers

Problem Transfers lists up to ten transfers with errors on all managed nodes. To view all transfers with errors, click the **Problem Transfers** header.

Problem Transfers

<table>
<thead>
<tr>
<th>TRANSFER</th>
<th>CONTACT</th>
<th>TIME</th>
<th>STATUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLES to Fedora #3</td>
<td>admin (console)</td>
<td>10:42am</td>
<td>User aborted session</td>
</tr>
<tr>
<td>test transfer 3</td>
<td>admin (console)</td>
<td>10:42am</td>
<td>Cancelled while waiting in queue</td>
</tr>
<tr>
<td>From New York to London</td>
<td>admin (console)</td>
<td>10:38am</td>
<td>User aborted session</td>
</tr>
</tbody>
</table>

Map

The map shows the status of all your monitored nodes and shows the transfers between them. If a node fails, the icon becomes red in the map, and the node and the problem are listed in the table below the map.

Nodes are not automatically added to maps. They must be configured. For more information, see *Configuring the Map* on page 52.
**Map**

Note: You can choose to hide or display the map and bandwidth chart by clicking the blue arrow (/left) next to the map.

**Bandwidth**

The Bandwidth chart shows bandwidth usage of your monitored nodes. If you select one or more nodes on the map, the chart shows the cumulative bandwidth of the selected nodes.

Note: You can choose to hide or display the map and bandwidth chart by clicking the blue arrow (left) next to the map.
The **Activity Overview**

The Activity Overview page lists all transfers on all managed nodes. View the Activity Overview page by going to **Activity**. You can narrow down the list with the filter and advance into a transfer's session detail page. The Activity Overview screen displays the following information:

<table>
<thead>
<tr>
<th><strong>Item</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>The transfer's name.</td>
</tr>
<tr>
<td>DETAILS</td>
<td>The transfer initiator, source, and destination.</td>
</tr>
<tr>
<td>START</td>
<td>This transfer's start time.</td>
</tr>
<tr>
<td>END</td>
<td>The estimated time of arrival, or the transfer completion time.</td>
</tr>
<tr>
<td>STATUS</td>
<td>Current status of this transfer.</td>
</tr>
<tr>
<td>AVG RATE</td>
<td>The transfer rate of the active transfer, or the average rate of a past transfer.</td>
</tr>
<tr>
<td>ACTIONS</td>
<td>Show all available actions. For example, <strong>pause</strong> and <strong>cancel</strong> for a running transfer or <strong>rerun</strong> for a past transfer.</td>
</tr>
</tbody>
</table>

The Current panel lists all currently active transfers, including running and queued transfers. The Past panel shows previous transfers, including those that were completed, canceled, or those that generated errors.

The filter options on the top can be used to narrow down the list.

<table>
<thead>
<tr>
<th><strong>Item</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>History</td>
<td>Select the time frame to display the started transfers.</td>
</tr>
<tr>
<td>Scheduled</td>
<td>Select the time frame to display the scheduled transfers.</td>
</tr>
<tr>
<td>Status</td>
<td>Select a specific transfer status to display.</td>
</tr>
<tr>
<td>Search</td>
<td>Search for keywords in transfer sessions.</td>
</tr>
</tbody>
</table>

You can also perform an advanced search by clicking on the **advanced** link. For more information on searching, see **Search for a Transfer** on page 53.
Transfer Details

Overview
Details about a particular transfer can be accessed by clicking on a transfer shown in listings of past, current, and scheduled transfers. These lists can be found in three locations:

- The Activity Overview page
- The Console Dashboard
- The Managed Node Detail page (the specific node from Nodes in the Console menu)

Ongoing Transfers
For an ongoing transfer, the Session Detail page provides the transfer monitor that displays current transfer status. You can control the transfer through the options shown at the top of the graph.

Important: The failed files counter may count "directories" if the network failed at some point or the user cancelled the transfer.

Finished or Failed Transfers
For a finished or failed transfer, the Session Detail page provides detailed information about the transfer's state, endpoints, and statistics.
The Session Files panel lists all files being transferred in this session. Click on a file to review its information. You can use the search box to show only specific files or groups of files.

**Note:** When searching for files, "*" is not a wildcard. Any string you enter is treated as a "search within". In other words, the string "foo" will match "123foo", "foo456", and "123foo456".

Console also lets you monitor notification information that includes messages about transfer starts, successes, errors, and what notification templates were used under the Statistics column. Next to Notifications is a link describing some combination of **start**, **success**, and **error** depending on what notifications were configured for the transfer, or **None** if no notifications were configured. Select the link to see the Session Notifications page. For more information, see **View Email Notification Statistics** on page 15.

**Multiple-Session Transfer**
A multiple-session transfer is a smart transfer with more than one destination. In the Activity Overview page, clicking on a multiple-session transfer reveals all sessions in the transfer. To drill down to the particulars of each session, click the **Session Detail** button to open its Session Detail page.

**Monitoring Nodes**
You can monitor the node status and manage the transfers on a node. Navigating to **Nodes** from the Console menu will bring you to the list of managed nodes. To view a list of unmanaged nodes, click the **List Unmanaged Nodes** button. To monitor a node, click on the node.

**Monitor Transfers on a Node**
On the Node Detail page, the transfer chart shows all inbound and outbound transfers on this node. To control a transfer session, select a session from the graph, and use the control options above the graph to control it.
The table lists all sessions on this node. Use **Pause** and **Cancel** to control an ongoing session.

### Configuring the Map

You can configure Console to display the locations of your nodes on the dashboard map.

1. Go to **Configuration > Map**.
2. Select or upload a map image for use on the Console dashboard.
   - Upload a new map image: Click **Upload Map File**. Upload the file and then click **select**. For best results, Aspera strongly recommends using an image with a ratio of 16:9 (for example, 800 x 450).
   - Select existing map image: Choose one of two default map images or any previously uploaded image as the dashboard map by clicking the **select** link.

   **Note:** To delete a map image you have uploaded, click the **delete** link.

3. Configure node to show on map.
   - Edit your node and click the **Map** tab. Select **Show on Map**. Click and drag the green icon to its proper location on the map.

The configured nodes appear on the map on the Dashboard. Ongoing transfers between nodes are represented by a line between the nodes.
Access Logs

Once you have created accounts for Console users, you can monitor their activity from the Accounts > Access Log tabs. The User Access Log displays user logins and logouts, concurrent logins, and session timeouts.

### User Access Log

<table>
<thead>
<tr>
<th>ACTION</th>
<th>DESCRIPTION</th>
<th>AT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login</td>
<td>User (2, admin 1) from 10.0.200.82</td>
<td>3:22pm</td>
</tr>
<tr>
<td>Login</td>
<td>User (2, admin 1) from 10.0.200.82</td>
<td>1:53pm</td>
</tr>
<tr>
<td>Login</td>
<td>User (2, admin 1) from 10.0.200.82</td>
<td>1:47pm</td>
</tr>
<tr>
<td>Logout</td>
<td>User (2, admin 1) from 10.0.200.82</td>
<td>1:47pm</td>
</tr>
<tr>
<td>Login</td>
<td>User (3, user 1) from 10.0.200.82</td>
<td>1:23pm</td>
</tr>
<tr>
<td>Login</td>
<td>User (2, admin 1) from 10.0.200.82</td>
<td>12:54pm</td>
</tr>
</tbody>
</table>

Search for a Transfer
You can search for a transfer from any page in IBM Aspera Console by using the search bar in the top right corner of the page. If you want to refine your search, you can access the Advanced Search dialog by selecting the blue drop-down arrow next to the search bar.

Console will search all transfers within the last 24 hours for transfers that match the search criteria.

For more information about the advanced search form, see *Advanced Search* on page 123.
Monitoring Sync Jobs

Enabling Async Server Node Reporting

By default, Console does not report transfers with a server that are associated with IBM Aspera Sync jobs or the Sync job information. If the server is a Console managed node and is running IBM Aspera Enterprise Server or IBM Aspera Connect Server version 3.1.5 or later, then it can be configured to report transfers and Sync jobs.

1. Enable server activity logging for Sync jobs.
   Run the following asconfigurator command to enable activity logging for Sync jobs:

   ```
   # asconfigurator -x "set_node_data;async_activity_logging,true"
   ```

   This command adds the following text to the `<default>` section of the `aspera.conf` file, located at: `/opt/aspera/etc/aspera.conf`.

   ```
   <CONF version="2">
   ...
   <default>
   ...
   <async_activity_logging>true</async_activity_logging>
   ...
   </default>
   ...
   </CONF>
   ```

2. Restart the Aspera Node API service to activate your changes.

   ```
   # service asperanoded restart
   ```

3. Confirm Sync jobs and transfers associated with them are reported in Console.
   After initiating a sync session, go to Console and go to Activity > Sync Jobs page to monitor the job.

   - **Note:** Sync job reporting (from the Sync Jobs screen) may not appear immediately.

Monitor Sync Jobs

To monitor Aspera Sync jobs, you must first configure Console to poll the Node API. For more information, see Enabling Async Server Node Reporting on page 55.

Once you have initiated a sync transfer, you can monitor it by going to Activity > Sync Jobs.

- **Note:** Sync jobs may not appear immediately.

From the Sync Jobs table, you can view a job's transfer details by clicking the corresponding row. The job's transfer details page displays the following:

- Local and remote server details
- Session statistics including the number of paths that are synced, pending, conflicted, deleted, or in error state
- Transfer rate graph, which is only active during the transfer
- **Remove log data** button, which deletes the job's slog data from the Console and Aspera Sync databases.

You can also remove log data from the Sync Jobs page by clicking **remove log data**.
Transferring Files

Starting a Simple Transfer

IBM Aspera Console can be used to initiate transfers between nodes when the Console user has the permission to start transfers. Console provides two types of transfer methods: simple transfers and smart transfers. Simple transfers are one-time transfer sessions that require entering all transfer information. Smart transfers are reusable templates with saved transfer settings.

1. Go to Transfer.
2. Click Simple Transfer.

3. Enter the transfer name and optional comments. The name and comments can be helpful if you want to search for this transfer later.

4. Optional: Add new tags or modify existing tags.
   Click the + button to add a new tag. Enter the tag name and the tag value. Click the × button to delete an existing tag. Select the 0 button to prevent a user from changing or deleting the locked tag when starting this transfer. For more information about tags, see Working with Tags on page 116.

5. Select the source node or saved endpoint from the Connect drop-down menu.
   - Node: A node appears as an entry with only the node name and IP address. A node does not have associated user credentials and prompts you to enter the login for an SSH user on the node and to authenticate by using either a password or an SSH key.
   - Endpoint: A saved endpoint is associated with an SSH user and either a password or an SSH key. Selecting a saved endpoint does not prompt you for credentials.

6. Click Browse and choose the files or folders you want to transfer. The files and folders you select are added to the Selected Source Items panel when you click Add.

   Note: When browsing the node, you can narrow your search by applying a filter. When specifying a filter, the asterisk (*) is not a wildcard. Any string you enter as a filter is treated as a "search within". In other words, the string "foo" matches "123foo", "foo456", and "123foo456".
By default, if a source item is a file, only the file is transferred. None of the folders in the file's path are transferred. If a source item is a folder, the folder and its entire contents are transferred, but none of the higher-level folders in its path are transferred. For example, if the source path is `aspera/tmp/sent_files`, the only folder that will be transferred to the destination is the `sent_files` folder. Neither `/aspera` nor `/tmp` appear at the destination location.

7. Select the destination node or saved endpoint from the **Connect** drop-down menu.
   - **Node**: A node appears as an entry with only the node name and IP address. A node does not have associated user credentials and prompts you to enter the login for an SSH user on the node and to authenticate by using either a password or an SSH key.
   - **Endpoint**: A saved endpoint is associated with an SSH user and either a password or an SSH key. Selecting a saved endpoint does not prompt you for credentials.

8. Click **Browse** and choose the files or folders you want to transfer. The files and folders you select are added to the Selected Source Items panel when you click **Add**.

9. Optional: Expand the settings under More Options to configure addition settings.

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Connection</strong></td>
<td>Configure <em>fasp</em> settings.</td>
</tr>
<tr>
<td><strong>Transfer</strong></td>
<td>Configure transfer rates and policies.</td>
</tr>
<tr>
<td><strong>Security</strong></td>
<td>Encrypt the transfer.</td>
</tr>
<tr>
<td><strong>File Handling</strong></td>
<td>Configure source file attributes, archive source files after transfer, and set filters for source files.</td>
</tr>
<tr>
<td><strong>Notifications</strong></td>
<td>Configure email notification options. For more information on email notifications, see Configuring Email Notifications on page 13.</td>
</tr>
<tr>
<td><strong>Advanced</strong></td>
<td>Configure transfer initiator, <em>fasp</em> MTU, and read and write block sizes on source and destination nodes.</td>
</tr>
<tr>
<td><strong>Transfer Time</strong></td>
<td>Schedule your transfer to run <strong>Now</strong> or <strong>Later</strong>. If you choose <strong>Later</strong>, click the **button and choose the date and time you want the transfer to run.</td>
</tr>
</tbody>
</table>

For information on these options, see **Simple Transfer Options** on page 153.

**Note:** If you schedule your simple transfer for a future time, you can cancel it by going to Activity > Transfers. Select "All" from the **Scheduled** drop-down menu, and click **Cancel**.
10. Click **Transfer** to start the transfer (or **Schedule** if you set a transfer time).

### Creating a Smart Transfer

IBM Aspera Console can be used to initiate transfers between nodes when the Console user has the permission to start transfers. Console provides two types of transfer methods: simple transfers and smart transfers. Simple transfers are one-time transfer sessions that require entering all transfer information. Smart transfers are reusable templates with saved transfer settings.

1. To create a smart transfer template, go to **Transfer > New Smart Transfer**.

2. Enter a transfer name.

3. Optional: Select **Share this smart transfer** to share this smart transfer with any user who has permissions for the transfer paths.

   For more information on sharing smart transfers, see *Sharing a Smart Transfer* on page 63.

4. Optional: Select **Allow changes to transfer settings at submit time** to allow the user who starts this smart transfer to change settings before submitting the transfer request.

5. Optional: Add new tags or modify existing tags.

   Click the button to add a new tag. Enter the tag name and the tag value. Click the button to delete an existing tag. Select the button to prevent a user from changing or deleting the locked tag when starting this transfer. For more information about tags, see *Working with Tags* on page 116.
The highlighted box in the Smart Transfer Diagram indicates whether you are configuring the Source or Destination for the smart transfer. Make sure Source is selected.

6. Select the source node or saved endpoint from the Connect drop-down menu.
   - Node: A node appears as an entry with only the node name and IP address. A node does not have associated user credentials and prompts you to enter the login for an SSH user on the node and to authenticate by using either a password or an SSH key.
   - Endpoint: A saved endpoint is associated with an SSH user and either a password or an SSH key. Selecting a saved endpoint does not prompt you for credentials.

7. Choose your Source directory.
   Click Choose Source Directory to browse the node for the directories and files you want to transfer. Console displays the source path you choose once you have chosen your source directory.

   **Note:** When browsing the node, you can narrow your search by applying a filter. When specifying a filter, the asterisk (*) is not a wildcard. Any string you enter as a filter is treated as a "search within". In other words, the string "foo" matches "123foo", "foo456", and "123foo456".

8. Select Specify base for source path(s) to place the transferred files directly into the destination folder without its hierarchy of directories. The specified base for the source path is removed from the source path when transferring directories.
   For example, if the source path is /shared_files/projects/presentation, a successful transfer results in the folder destination_folder/shared_files/projects/presentation on the destination node. A successful transfer with "/shared_files/projects" specified as the base path results in the folder destination_folder/presentation on the destination node.
   For more information on specifying a base path, see Specify Base for Source Path on page 160.

9. Select one of the following file-transfer rules from the Items to transfer drop-down list:
   - Always transfer the entire directory: The transfer always transfers all files in the source directory.
   - Allow initiator to select items when starting manually: The user starting this smart transfer can choose the items in the directory included in the transfer.

10. Expand the settings under More Options to configure addition settings.
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection</td>
<td>Configure fasp settings.</td>
</tr>
<tr>
<td>Transfer</td>
<td>Configure transfer rates and policies.</td>
</tr>
<tr>
<td>Security</td>
<td>Encrypt the transfer.</td>
</tr>
<tr>
<td>File Handling</td>
<td>Configure source file attributes, archive source files after transfer, and set filters for source files.</td>
</tr>
<tr>
<td>Notifications</td>
<td>Configure email notification options. For more information on email notifications, see Configuring Email Notifications on page 13.</td>
</tr>
<tr>
<td>Advanced</td>
<td>Configure transfer initiator, fasp MTU, and read and write block sizes on source and destination nodes.</td>
</tr>
<tr>
<td>Transfer Time</td>
<td>Schedule your transfer to run Now or Later. If you choose Later, click the button and choose the date and time you want the transfer to run.</td>
</tr>
</tbody>
</table>

For more information on these options, see Smart Transfer Options on page 155.

The highlighted box in the Smart Transfer Diagram indicates whether you are configuring the Source or Destination for the smart transfer. Make sure a Destination is selected. You can create additional destination endpoints by clicking the button. To remove a destination, click the button inside the destination box.
11. Select the destination node or saved endpoint from the Connect drop-down menu.

- Node: A node appears as an entry with only the node name and IP address. A node does not have associated user credentials and prompts you to enter the login for an SSH user on the node and to authenticate by using either a password or an SSH key.
- Endpoint: A saved endpoint is associated with an SSH user and either a password or an SSH key. Selecting a saved endpoint does not prompt you for credentials.

12. Select your Destination directory.

Click Choose Destination Directory to browse the node for the directories and files you want to transfer. Console displays the source path you choose once you have chosen your source directory.

Note: When browsing the node, you can narrow your search by applying a filter. When specifying a filter, the asterisk (*) is not a wildcard. Any string you enter as a filter is treated as a "search within". In other words, the string "foo" matches "123foo", "foo456", and "123foo456".

13. Optional: Allow the user starting this smart transfer to change the directory on this destination node. The Change Destination Path button appears for a destination with this option enabled.

14. Optional: Allow the user starting this smart transfer to remove this destination node. The button appears for a destination with this option enabled.

15. Optional: Configure additional settings for this individual destination node.

Note: This option is only available for a smart transfer with multiple destination nodes.

Select Set transfer options individually for this destination. The More Options appears at the bottom of the page.

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection</td>
<td>Configure fasp settings.</td>
</tr>
<tr>
<td>Transfer</td>
<td>Configure transfer rates and policies.</td>
</tr>
<tr>
<td>Security</td>
<td>Encrypt the transfer.</td>
</tr>
<tr>
<td>File Handling</td>
<td>Configure source file attributes, archive source files after transfer, and set filters for source files.</td>
</tr>
</tbody>
</table>
### For information on these options, see *Smart Transfer Options* on page 155.

16. Click **Save**.

Once a smart transfer template has been saved, it is accessible from the Transfer page. Go to **Transfer** to **start**, **edit**, **copy**, and **delete** existing smart transfers.

---

## Starting a Smart Transfer

IBM Aspera Console can be used to initiate transfers between nodes when the Console user has the permission to start transfers. Console provides two types of transfer methods: simple transfers and smart transfers. Simple transfers are one-time transfer sessions that require entering all transfer information. Smart transfers are reusable templates with saved transfer settings.

1. Go to **Transfer** to see all the smart transfers you have permission to access. For instructions on creating a smart transfer, see *Creating a Smart Transfer* on page 58.

2. Find the smart transfer listed under Saved Smart Transfers and click **Start**.

3. Optional: Modify the **Transfer Name** and leave a comment describing the transfer.

4. Optional: Add new tags or modify existing tags.

![Tags](image)

Click the **+** button to add a new tag. Enter the tag name and the tag value. Click the **-** button to delete an existing tag. Locked tags are greyed out and cannot be modified. For more information, see *Working with Tags* on page 116.

5. Optional: Configure email notification options.

   Expand the Notifications section. Add or delete email addresses and configure notifications for existing email addresses. For more information, see *Configuring Email Notifications* on page 13.

6. Optional: Schedule the transfer to run **Now** or **Later**.

   If you choose **Later**, click the **date** button and choose the date and time you want the transfer to run.
7. Click Start.

Sharing a Smart Transfer

Smart transfers are reusable templates with saved settings. The primary use case for sharing smart transfers is to set up pre-defined transfers for non-admin users to run. You can decide what transfers a user can monitor and start by limiting the user's permissions and access to a smart transfer. By default, shared transfers require you to use endpoints created by an admin using the Edit Nodes > Endpoints page. Once configured in Console settings, you can also share smart transfers saved with personal login credentials and domain names.

For more information about sharing smart transfers with personal logins, see Sharing a Smart Transfer with Personal Login Credentials on page 64

Note: These instructions assume you know how to configure a smart transfer. For more information, see Creating a Smart Transfer on page 58.

1. Create endpoints on the nodes you want to use for this smart transfer.
   For detailed instructions, see Adding Endpoints on page 27.

   Tip: To use domain names as transfer endpoints, create an unmanaged node using a domain name, then add an Endpoint to this unmanaged node.

2. Go to Transfer and click New Smart Transfer.

3. Select Share this smart transfer.

   Note: When creating a smart transfer with Any as an endpoint, you must first save the smart transfer before selecting Share this smart transfer.
4. Select endpoints for the Source and Destination.
   Tip: Create new personal saved endpoints by selecting the desired node and entering the SSH user login credentials.

5. Finish configuring the smart transfer. Click **Save** when finished.

6. Enable a user to start this smart transfer.
   - Create a group with permissions to start smart transfers for this transfer path (see *Creating Console Groups* on page 43).
   - Add the user to this group (see *Creating Console Users* on page 44).

Admin users have permissions to all transfers and do not need to be added to a group to use a shared smart transfer. By default, admins do not have the ability to edit Smart Transfers that are shared with them but owned by another admin. To enable admins to edit each other's smart transfers, go to **Configuration > Defaults** and select **Smart Transfer Editing: Allow administrators to edit each other's Smart Transfers**.

Note: Even with this feature enabled, admins can only edit smart transfers that do not contain personally saved login credentials.

Tip: Editing another admin's smart transfer changes ownership of the smart transfer to the admin who made the last change.

---

**Sharing a Smart Transfer with Personal Login Credentials**

Smart transfers are reusable templates with saved settings. The primary use case for sharing smart transfers is to set up pre-defined transfers for non-admin users to run. You can decide what transfers a user can monitor and start by limiting the user's permissions and access to a smart transfer. By default, shared transfers require you to use endpoints created by an admin using the **Edit Nodes > Endpoints** page. Once configured in Console settings, you can also share smart transfers saved with personal login credentials and domain names.

Personal login credentials are automatically created and saved when a user creates a transfer, chooses a node, and enters authentication credentials for an SSH user on that node. The following describes how to share a smart transfer with personal login credentials.

Note: These instructions assume you know how to configure a smart transfer. For more information, see *Creating a Smart Transfer* on page 58.

1. Go to **Configuration > Defaults** and select **Smart Transfer Sharing**.
2. Create a new smart transfer, select **Share this smart transfer**.

   ![Description](image)

   Note: When creating a smart transfer with Any as an endpoint, you must first save the smart transfer before selecting **Share this smart transfer**.

3. Select personal saved endpoints for the Source and Destination.
If you have no personal saved endpoints, create a new one by selecting the desired node and entering the SSH user login credentials.

4. Finish configuring the smart transfer. Click **Save** when finished.

5. Enable a user to start this smart transfer.
   - Create a group with permissions to start smart transfers for this transfer path (see *Creating Console Groups* on page 43).
   - Add the user to this group (see *Creating Console Users* on page 44).

Admin users have permissions to all transfers and do not need to be added to a group to use a shared smart transfer. By default, admins do not have the ability to edit Smart Transfers that are shared with them but owned by another admin. To enable admins to edit each other's smart transfers, go to **Configuration > Defaults** and select **Smart Transfer Editing: Allow administrators to edit each other's Smart Transfers**.

**Note:** Even with this feature enabled, admins can only edit smart transfers that do not contain personally saved login credentials.

**Tip:** Editing another admin's smart transfer changes ownership of the smart transfer to the admin who made the last change.

---

**Queue Transfers**

**Overview**

The Console queueing feature provides two useful capabilities:

- Admins can limit the number of Console-initiated transfers that can run concurrently for a given destination or from a given source. This can be useful if network connections have limited bandwidth or if particular destination nodes have difficulty handling more than a small number of transfers at a time. For example, if the concurrency limit for a connection is "2", and two transfers are in progress, any new transfers initiated while the first two are still in progress will be queued in the order in which they were initiated.

- All users can change the priority order of queued and in-progress transfers. This can be useful in situations where users need to respond to emergencies or shifting priorities.

**Important:** Queueing only applies to transfers started from Console or via its API. Transfers started outside Console are not subject to queueing and do not count towards concurrency limits.

Concurrency limits are always assigned on a per-node basis, and per outbound or inbound direction. However, the overall, actual limit on a set of concurrent transfers between two nodes is governed by the node with the lowest limit. That is, if NodeA has an outbound limit of "2" and NodeB has an inbound limit of "1", concurrent transfers from NodeA to NodeB are limited to one transfer at a time, with subsequent transfers queued up in the order in which they were initiated.

**Adjusting the Queueing Properties of In-Progress Transfers**

If queuing is enabled on a node (see *Configuring Queues for Nodes* on page 67), the queueing properties of in-progress transfers can be adjusted in several ways:

- Their relative priorities can be raised or lowered.
- They can be paused and resumed.
- The concurrency limit in effect can be raised or lowered.
- Concurrency (and therefore queueing) can be disabled completely.

These adjustments can be made while monitoring the node from the Node Detail page. You can view the Node Detail page by going to **Nodes** and clicking on the node.
**Tip:** You can also reach this page from the Queing page on the current queue contents link next to an enabled concurrency limit.

On the Node Detail page, below the transfer chart, you may see the Inbound Queue tab, the Outbound Queue tab, or both. These tabs are visible if the node is configured with inbound or outbound queueing. Clicking the tab displays the node’s inbound or outbound transfers - both those currently in progress and those in the queue.

To view past transfers, open the Transfers tab. The Transfers tab also shows both outbound and inbound transfers, but does not include controls to promote, demote, pause, or resume transfers.

### Controlling In-Progress Transfers

<table>
<thead>
<tr>
<th>Icon</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Pause Icon]</td>
<td>Pause Transfer</td>
</tr>
<tr>
<td>![Resume Icon]</td>
<td>Resume Transfer</td>
</tr>
<tr>
<td>Icon</td>
<td>Action</td>
</tr>
<tr>
<td>------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>▲</td>
<td>Promote Transfer to Highest Priority</td>
</tr>
<tr>
<td>▲</td>
<td>Promote Transfer</td>
</tr>
<tr>
<td>▼</td>
<td>Demote Transfer</td>
</tr>
<tr>
<td>▼ ▲</td>
<td>Demote Transfer to Lowest Priority</td>
</tr>
<tr>
<td>x</td>
<td>Cancel Transfer</td>
</tr>
<tr>
<td>🔄</td>
<td>Highlight this session on the graph</td>
</tr>
</tbody>
</table>

### Configuring Queues for Nodes

Both managed and unmanaged nodes can be configured for queuing. For more information on queuing, see [Queue Transfers](#) on page 65.

1. Go to **Nodes**. Find your node in the Managed Nodes or Unmanaged Nodes page. Click **edit** and then click **Queueing**.
2. Enable queueing by selecting **Limit concurrency** (disabled by default) for Inbound Transfers.
3. Choose the maximum number of concurrent transfers allowed for this node. The default setting is "1".
4. Repeat the previous two steps for Outbound Transfers.
5. Click **Update**.

In the example below, queueing is disabled for inbound transfers. For outbound transfers, queueing is enabled for at most three transfers in progress at the same time.

![Inbound Transfers](#)

- **Limit concurrency**: unchecked
- **Max. concurrent transfers**: 1

![Outbound Transfers](#)

- **Limit concurrency**: checked
- **Max. concurrent transfers**: 3 current queue contents...

### Configure Failover Groups

#### Overview

A failover group contains a group of different nodes that act as substitutes for the original node in the case that the original node becomes unavailable. When a node goes offline, Console also restarts any transfers in progress on that node, submitting them to a different node in the group.
Transfer failover only activates if the status of a node is set to error. Transfers that are inactive do not failover.

**Node Requirements**
Nodes must have identical passwords, transfer accounts, and docroots to be grouped together. Make sure each node has identical configurations for each item in the following list before adding them to a failover group:

- System User Accounts
- Transfer User Accounts
- Node API User Accounts
- Docroots
- Endpoints on Console
- Directory Structure

**Adding a Node to a Failover Group**
When adding or editing a node, select **Enable failover and load balancing for Console-initiated transfers on this node**. Add the node to an existing group or select **enter new name** from the **Failover Group Name** drop-down menu to create a new group.

If you select **enter new name**, enter a new failover group name in the prompt.

**Endpoint Synchronization**
Editing an endpoint on a node of a failover group makes those changes to the same endpoint on the other nodes in the failover group.

- **Note:** Only saved and synchronized endpoints should be selected as a destination when starting a transfer.

**Configuring Load Balancing**
Go to **Configuration > Console Defaults** and configure the Failover / Load balancing Behavior option.

- **Failover + Load Balancing:** The transfer uses the least busy nodes first.
- **Failover only:** The transfer will use the original endpoints that the user specified.

**Creating a Cookie Parsing Rule**

- **Note:** Cookie configuration applies only to the use of custom cookies. Console does not apply parsing rules to cookies it recognizes as standard cookies used by Aspera products.

In an **ascp** command-line transfer, you can specify the transfer cookie with an environment variable.

```bash
set ASPERA_SCP_COOKIE=custom_cookie
```

Using a rule, Console can match the set cookie string and then substitute it for selected transfer information.
1. Go to Configuration > Cookies. Click New Rule.
2. Name the rule.
3. Configure the cookie.
   Enter the regular expression Console uses to filter transfers. If this string matches a transfer, Console includes the cookie in the transfer and the information in the other fields is used in the transfer session.
   - **Tip:** The format used for regular expression is the RUBY format described here: [http://www.rubydoc.org/core/Regexp.html](http://www.rubydoc.org/core/Regexp.html).
4. Configure the cookie with the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Started via</td>
<td>Name of the transfer initiator.</td>
</tr>
<tr>
<td>Contact description</td>
<td>Description of this transfer initiator.</td>
</tr>
<tr>
<td>Transfer name</td>
<td>Name for this transfer.</td>
</tr>
</tbody>
</table>

5. Click Create.

When you have multiple cookie parsing rules, Console uses the first rule listed that matches the cookie string. To modify the order of the parsing rules, drag-and-drop the rules in the list. If two rules have identical regular expressions, the rule that is higher in the list is applied.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule name</td>
<td>MyCustomCookieRule</td>
</tr>
<tr>
<td>Regexp</td>
<td>^setcustomfields:(.+?):(.+?):(.+?):$</td>
</tr>
<tr>
<td>Started via</td>
<td>/1</td>
</tr>
<tr>
<td>Contact description</td>
<td>/2</td>
</tr>
<tr>
<td>Transfer name</td>
<td>/3</td>
</tr>
</tbody>
</table>

For example, the following cookie replaces Started via with "My App", Contact description with "My Contact", and Transfer name with "My Transfer".

```plaintext
set ASPERA_SCP_COOKIE="setcustomfields:My App:My Contact:My Transfer:"
```
Working with Watchfolders

Configuring a Watchfolder-enabled Transfer Node

Watchfolders uses the Aspera Watchfolder service (`asperawatchfolderd`) and the output from the Aspera Watch Service (`asperawatchd`) to automatically transfer new files added to a source folder on an Aspera node. Follow the instructions below to start the two services.

Note: You must perform the following instructions on the transfer node.

1. Select a user account to run your services.

Watchfolder services must be run under a user with access to every area of your file system in which you intend to create a watch folder. Watchfolders allows you to run multiple instances of these services under different users. Most users run these services under one user. Choose a user that has access to your entire file system.

If you need to run multiple instances of these services to access every area of your file system, see Choosing User Accounts to Run Watchfolder Services on page 73.

2. Configure a docroot for the chosen user.

The docroot is a security feature that allows you to restrict the area `asperawatchfolderd` can access. If you need to access the entire file system, you can set the docroot path as `/`, but you cannot leave it empty.

Set the docroot for the user using the `asconfigurator` utility:

```shell
# asconfigurator -x "set_user_data;user_name,username;absolute,docroot"
```

For example:

```shell
# asconfigurator -x "set_user_data;user_name,root;absolute,/
success
user_name: root
```

The `asconfigurator` command adds the following configuration to the `<aaa>` section of `aspera.conf`:

```xml
<aaa>
    <realms>
        <realm>
            <users>
                <user>
                    <name>root</name>
                    <file_system>
                        <access>
                            <paths>
                                <path>
                                    <absolute>/</absolute>
                                    <path/>
                                </path>
                            </paths>
                        </access>
                    </file_system>
                </user>
            </users>
        </realm>
    </realms>
</aaa>
```

3. Configure the `asperawatchd` and `asperawatchfolderd` services to run under your user.
The following commands add the services in the `asperarund` database; `asperarund` automatically starts and preserves services in its database. The `asperarund` service must be running.

**Windows Node:**

```plaintext
> asperawatchd --user username
> asperawatchfolderd --user username
```

Windows requires a password when running services as other users. Enter the password when prompted.

**OS X Node:**

```plaintext
$ /Library/Aspera/sbin/asperawatchd --user username
$ /Library/Aspera/sbin/asperawatchfolderd --user username
```

**Linux Node:**

```plaintext
# /opt/aspera/sbin/asperawatchd --user username
# /opt/aspera/sbin/asperawatchfolderd --user username
```

4. Verify that `asperawatchd` and `asperawatchfolderd` is running under the given user.

   Use the `aswatchadmin` and `aswatchfolderadmin` admin tools to retrieve a list of running daemons. Daemons are named with the username you passed in when starting the service. For example, if you used the root user to run your services, you should see the root daemon listed in the output.

   **Windows Node:**

   ```plaintext
   > aswatchadmin query-daemons
   [aswatchadmin query-daemons] Found a single daemon: svcaspera
   > aswatchfolderadmin query-daemons
   [aswatchfolderadmin query-daemons] Found a single daemon: svcaspera
   ```

   Windows requires a password when running services as other users. Enter the password when prompted.

   **Linux Node:**

   ```plaintext
   $ /opt/aspera/bin/aswatchadmin query-daemons
   [aswatchadmin query-daemons] Found a single daemon: root
   $ /opt/aspera/bin/aswatchfolderadmin query-daemons
   [aswatchfolderadmin query-daemons] Found a single daemon: root
   ```

5. Create a Node API User and map it to a system account. The user account must have administrative privileges to interact with `asperawatchfolderd`.

   Using the REST API provided by the `asperanoded` service, a user can create, remove, query and modify watchfolder instances. The `asperanoded` service forwards requests to the corresponding watch folder instances over Redis.

   **Windows Node:**

   ```plaintext
   > asnodeadmin -a -u node_username -p node_password -x admin_user --acl-set "admin,impersonation"
   ```
Linux Node:

```
# /opt/aspera/bin/asnodeadmin-a -u node_username -p node_password -x admin_user --acl-set "admin,impersonation"
```

For example:

Windows Node:

```
> asnodeadmin -a -u watchfolder_user -p X245lskd3 -x root --acl-set "admin,impersonation"
```

Linux Node:

```
# /opt/aspera/bin/asnodeadmin -a -u watchfolder_user -p X245lskd3 -x root --acl-set "admin,impersonation"
```

Adding, modifying, or deleting a node-user triggers automatic reloading of the user database and the node's configuration and license files. For more information on the Node API, see your transfer server's administrator guide.

6. Verify that you correctly added the node user.

```
# /opt/aspera/bin/bin/asnodeadmin -1
List of node user(s):
 user       system/transfer user                    acls
--------------------    ----------------------    ---------------
 node_api_user
 system_user  [admin]
```

Given the example in the previous step, the output should look like the following:

```
# /opt/aspera/bin/bin/asnodeadmin -1
List of node user(s):
 user       system/transfer user                    acls
--------------------    ----------------------    ---------------
 watchfolder_user
```

7. Add the transfer node to Console as a managed node.

This transfer node is now ready to be added to Console as a managed node. For instructions on adding nodes to Console, see *Creating a Managed Node in Console* on page 26.

Once you have successfully added the transfer node to Console as a managed node, you can use the node to create watch folders. For instructions on creating a watch folder, see *Creating a Watch Folder in Console* on page 72.

---

**Creating a Watch Folder in Console**

To use watch folders, you must enable the feature on the Console Defaults page. In Console, go to **Configuration > Defaults** and select **Enable Watchfolder management**.

Configure the transfer node. Start watchfolder services and configure the Node API so that Console can add the transfer node as a managed node. See *Configuring a Watchfolder-enabled Transfer Node* on page 70.

A watch folder is an automation of file transfers from a source to a destination system. Files placed into a source folder are automatically transferred to the destination. It runs on the client side only and the recipient Aspera server endpoint does not need additional software components to support receiving data. The following pro

1. Go to **Activity > Watchfolders** and click **Create New Watchfolder**.
2. Enter a name for your watch folder in the Name field.
3. Select a managed node configured to run watch folders from the **Local Node** drop-down menu.
Note: Only nodes configured to run watch folders appear on the drop-down list. If you do not see any nodes, make sure your nodes are properly configured in Console.

4. Click Browse to choose the source directory or enter the source path.

Tip: The source folder is relative to associated system user's docroot. For example, for the user root if you enter the path project1/final and the docroot for "root" is /root, the full path is /root/project1/final. For instructions on modifying the docroot, see Set a Docroot for a Node User or Group on page 31.

5. Enter the credentials to authenticate to the destination node.
   a) Enter hostname or IP address host address for the destination node in the Address field.
   b) Enter the SSH login credentials for the destination node.

      Enter the administrative account username and password to allow Console to connect to the node machine.
      Authenticate the account with either a password or a saved SSH key.

      • Password authentication: Enter the account password.
      • Public key authentication: Select Use SSH Key and select your uploaded key. To use public key authentication, you must have your SSH private key configured in Console. For instructions on how to configure SSH keys in Console, see Configuring SSH Keys on page 94.

6. Set the TCP and UDP ports. By default, TCP uses port 22 and UDP uses port 33001.

7. Click Browse to test the connection and choose the target directory.

8. Configure additional watchfolder settings.

    For more information on watch folder settings, see Watchfolder Options on page 84.

9. Click Create.

10. Test your watch folder.

    Add files to the source directory. If the configuration is correct, Watchfolder detects the new files, starts a transfer, and the added files appear in the target directory. You can see the transfer on the Activity Overview page.

Choosing User Accounts to Run Watchfolder Services

Watchfolder services must be run under a user with access to every area of your file system in which you intend to create a watch folder. Watchfolders allows you to run multiple instances of these services under different users. Most users run these services under one user. Choose a user that has access to your entire file system.

You may need to run multiple instances of these services to access every area of your file system. For example, your file system may include mounted storage from the marketing department and another from the release team. In such a scenario, you may not have a single user that can access files in both mounted storages, or your administrative account has access to the entire file system, but your policy prohibits running the asperawatchd and asperawatchfolderd services under that user account.

There are two ways to configure watch folders depending on your scenario. See the chart below for the configuration that matches your scenario:
Configuration #1
This is the simplest and most common configuration of watchfolder services. Choose an account that has read permissions for all your files and follow the instructions in *Creating a Watch Folder in Console* on page 72.

Configuration #2
If you cannot run watchfolder services under the administrative account or you do not have a single user that has access to the entire file system, you must run pairs of `asperawatchd` and `asperawatchfolderd` services under enough users to access your entire file system.

For example, if you have user mounted storage from the marketing department that can only be accessed by user xasp1, and another storage from the release team, which can only be accessed by user xasp2, you need to run a pair of `asperawatchd` and `asperawatchfolderd` services under each user. Aspera recommends using the Node API to configure services and manage watch folders in a multi-user context. You can interact with the Node API using IBM Aspera Console.

**Updating the Docroot of a Running asperawatchfolderd Service**

If, when creating a watch folder, `aswatchfolderadmin` errors out with error code `err=28672`, check that your docroot has been properly configured to provide access to the source directory specified in the JSON configuration file. You may have specified a destination that is not permitted by the docroot of the user running `asperawatchfolderd`, or you may have no docroot configured at all.

If you need to make changes to your docroot, follow the instructions below to update your docroot and restart the `asperawatchfolderd` service.

1. Update the docroot for the user running `asperawatchfolderd`.

   The docroot is a security feature that allows you to restrict the area `asperawatchfolderd` can access. If you need to access the entire file system, you can set the docroot path as `/`, but you cannot leave it empty.
Set the docroot for the user using the `asconfigurator` utility:

```bash
# asconfigurator -x "set_user_data;user_name,username;absolute,docroot"
```

For example:

```bash
# asconfigurator -x "set_user_data;user_name,root;absolute,/
success
user_name: root
```

The `asconfigurator` command adds the following configuration to the `<aaa>` section of `aspera.conf`:

```
<aaa>
  <realms>
    <realm>
      <users>
        <user>
          <name>root</name>
          <file_system>
            <access>
              <paths>
                <path>
                  <absolute>/</absolute>
                </path>
              </paths>
            </access>
          </file_system>
        </user>
      </users>
    </realm>
  </realms>
</aaa>
```

You can find the `aspera.conf` configuration file at:

```
/opt/aspera/etc/aspera.conf
```

2. **Restart the asperawatchfolderd service to pick up the changes to the docroot.**

Use the `asrun` utility to disable and enable the `asperawatchfolderd` service.

```bash
# /opt/aspera/bin/asrun send --disable="username"
# /opt/aspera/bin/asrun send --enable="username"
```

---

## Working with Growing Files

Growing files are "in progress" files that continue to grow in size. For example, videos generated at live events continue to grow in size and duration. Watchfolders can detect these growing files and start transferring them to a target directory using Aspera faspstream technology.

**Important:** Both the source host and remote host must be running IBM Aspera Enterprise Server 3.6.0+ to support the faspstream technology used to transfer growing files. Also, the remote host transfer user for the watchfolder must not have a docroot.

To configure a watchfolder to work with growing files, first add the source host to Console as a managed node. Then, follow the instructions below to a new watchfolder and configure the Growing Files options.

**Important:** The source host must be properly configured for watchfolders. For more information, see *Configuring a Watchfolder-enabled Transfer Node* on page 70.
1. Go to **Activity > Watchfolders** and click **Create New Watchfolder**.
2. Name your watchfolder. For example, name your watchfolder "Camera Feed Transfers".
3. Select a managed node configured to run watch folders from the **Local Node** drop-down menu.

   **Note:** Only nodes configured to run watch folders appear on the drop-down list. If you do not see any nodes, make sure your nodes are properly configured in Console.

4. Enter the credentials to authenticate to the destination node.
   a) Enter hostname or IP address host address for the destination node in the Address field.
   b) Enter the SSH login credentials for the destination node.

   Enter the administrative account username and password to allow Console to connect to the node machine. Authenticate the account with either a password or a saved SSH key.

   - Password authentication: Enter the account password.
   - Public key authentication: Select Use SSH Key and select your uploaded key. To use public key authentication, you must have your SSH private key configured in Console. For instructions, on how to configure SSH keys in Console, see [Configuring SSH Keys](#) on page 94.

5. Set the TCP and UDP ports. By default, TCP uses port 22 and UDP uses port 33001.
6. Click **Browse** to test the connection and choose the target directory.

7. Configure the Growing Files section to configure the session bandwidth policy and the growing files detection policy.

   **Note:** Sessions with growing files are not affected by settings in the Transfer section.

   Expand the **Growing Files** section. Configure the following options.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target rate</td>
<td>Specify the target transfer rate.</td>
<td>10000</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
<td>Example</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>Transport encryption</td>
<td>Select aes-128 to transfer with this encryption method.</td>
<td>aes128</td>
</tr>
<tr>
<td>TCP port</td>
<td>Enter the TCP port to use for this watch folder.</td>
<td>22</td>
</tr>
<tr>
<td>fasp™ port (UDP)</td>
<td>Enter the UDP port to use for this watch folder.</td>
<td>33001</td>
</tr>
<tr>
<td>Completion timeout</td>
<td>Specify the amount of time to wait for the file to no longer change for the session to finish.</td>
<td>10</td>
</tr>
<tr>
<td>Growing file filters</td>
<td>The growing file filter allows watchfolders to define which files are considered growing files. Click the button to add a new filter to identify the growing file. You can set a filter to include or exclude files by globbing or by regular expression.</td>
<td>*.m2p</td>
</tr>
</tbody>
</table>

For more information on watchfolder settings, see *Watchfolder Options* on page 84.

![File Handling (show)](image)

8. Click Create.
Working with Package Sending

A user can control the order of transfer for files added to a watchfolder by configuring the watchfolder to send with packages. Watchfolder uses a file list to determine the files in a package and the order in which they are transferred. The last file in a file list is designated to be transferred last. It will not be transferred until all other files included in the file list have been transferred. Using watchfolders with file lists supports workflows that trigger when detecting the presence of a particular file, but require the presence of a particular set of files to function correctly. For example, a workflow that builds a project dependent on the existence of other files will fail if the trigger file is transferred before those other files have been transferred. Using a file list ensures the trigger file is not transferred until the rest of the files have been transferred.

To configure a watchfolder to work with package sending, first add the source host to Console as a managed node. Then, follow the instructions below to create a new watchfolder and configure the Package List options.

**Important:** The source host must be properly configured for watchfolders. For more information, see Configuring a Watchfolder-enabled Transfer Node on page 70.

1. On the source host, create a file list of the files you want transferred and the file that needs to be transferred last. The first line has to be "#aspera-filelist". Enter the names of the files you want transferred in the following lines. The last file in the file list is not transferred until the rest of the preceding files are finished transferring. The last file transferred is always the last file in the filelist. Here is an example file list:

```
#aspera-filelist
superhero_epic_the_musical-english.mp4
superhero_epic_the_musical-spanish.mp4
superhero_epic_the_musical-chinese.mp4
superhero_epic_the_musical.png
superhero_epic_the_musical-english-trailer.mp4
ADI.XML
```

**Note:** The paths listed in the file list should be entered as paths to files relative to the watchfolder directory. Paths to directories are not supported.

2. Go to Activity > Watchfolders and click Create New Watchfolder.
3. Name your watchfolder. For example, name your watchfolder "A Package of Files".
4. Select a managed node configured to run watch folders from the Local Node drop-down menu.

**Note:** Only nodes configured to run watch folders appear on the drop-down list. If you do not see any nodes, make sure your nodes are properly configured in Console.
5. Enter the credentials to authenticate to the destination node.
   a) Enter hostname or IP address host address for the destination node in the Address field.
   b) Enter the SSH login credentials for the destination node.
      Enter the administrative account username and password to allow Console to connect to the node machine.
      Authenticate the account with either a password or a saved SSH key.
      • Password authentication: Enter the account password.
      • Public key authentication: Select Use SSH Key and select your uploaded key. To use public key authentication,
        you must have your SSH private key configured in Console. For instructions, on how to configure SSH keys in
        Console, see Configuring SSH Keys on page 94.

6. Set the TCP and UDP ports. By default, TCP uses port 22 and UDP uses port 33001.

7. Click Browse to test the connection and choose the target directory.

8. Configure the Transfers section to configure the session bandwidth policy.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target rate</td>
<td>Specify the transfer target rate.</td>
<td>10000</td>
</tr>
<tr>
<td>Minimum rate</td>
<td>Set the transfer minimum rate</td>
<td>0</td>
</tr>
<tr>
<td>Bandwidth policy</td>
<td>Choose a transfer policy among fixed/high/fair/low.</td>
<td>fair</td>
</tr>
<tr>
<td>Transport Encryption</td>
<td>Select aes-128 to transfer with this encryption method.</td>
<td>aes128</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
<td>Example</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>Retry policy</td>
<td>Specify the number of attempts and the duration between each retry.</td>
<td>• 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• 30</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• 50</td>
</tr>
</tbody>
</table>

9. Configure the Packages / Drops section to configure the file list detection policy.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package timeout</td>
<td>The package timeout defines the time in which watchfolder waits for required files. If the required files do not appear within the duration, files with dependencies are marked as not transferred because of unsatisfied dependencies.</td>
<td>10</td>
</tr>
<tr>
<td>Final transfer</td>
<td>• Last file in list: The last file in the package list is transferred last. • File list: The last file transferred is the file list itself.</td>
<td>Last file in list</td>
</tr>
</tbody>
</table>

![Watchfolder Settings (show)](image)

![File Handling (show)](image)
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>File list filters</td>
<td>The file list filter allows watchfolders to define which files are considered file lists. Click the + button to add a new filter to identify file lists. You can set a filter to include or exclude files by globbing or by regular expression.</td>
<td>*.package</td>
</tr>
</tbody>
</table>

For more information on watchfolder settings, see *Watchfolder Options* on page 84.

10. Click **Create**.

Your watchfolder is now ready to transfer files with file lists. Once you add the files and the file list into the source directory, watchfolder detects the addition and starts transferring the files. Watchfolder transfers all files before transferring the last file in the list.
In this example, watchfolders transfers all files before transferring the last file: "ADI.XML".
Working with Filters

When given a large set of files, you can configure filters to determine what files watchfolders detects and transfers. Watchfolders support both globbing and regular expression filters.

To configure a watchfolder with filters, first add the source host to Console as a managed node. Then, follow the instructions below to create a new watchfolder and configure the Package List options.

Configuring Filters

Go to Activity > Watch Folders and click Create New Watchfolder or edit an existing watchfolder. Expand the Watchfolder Settings section.

Next to the File filters field, click the button to add a new filter to identify file lists. You can set a filter to include or exclude files by globbing or by regular expression.

For example, to exclude text and backup files from a fileset, create two filters that exclude files that match the glob expressions "*.txt" and "*.bak".
Watchfolder Options

The following tables provide information on additional configurable settings that are available when creating watchfolders.

Watchfolder Settings

**Note:** A watchfolder groups new or updated files it detects in its source folder into "drops". A drop is defined by the duration set by the snapshot creation period. All files in a given drop are transferred in the same transfer session, post-processed together, and reported as a unit.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Drop detection strategy       | The strategy this watchfolder uses to detect files dropped into the source folder.  
• Cool off only: Create a drop that includes new files added within the duration by the snapshot creation period.  
• Top level files: Create a drop for each file placed in the top level of the source folder.  
• Top level directories: Create a drop for each directory placed in the top level of the source folder. This drop also includes the sub-directories and files in the top level directory.                                                                                                                                                                                                                                                                                                                                 |  
| Drop detection cool off       | The duration allowed for new files to be included in a drop. Aspera recommends choosing a multiple of the specified `snapshot_creation_period` for predictable results.                                                                                                                                                                                                                                                                                                                                                                   |
| Snapshot creation period      | The duration used to determine what files are included in the current drop.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Connect timeout               | The duration the source node waits to connect to the destination node.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Sample period                 | The frequency of the system estimating the available bandwidth.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Queue threshold               | The duration watchfolder adds files to a session. Use this feature to limit the number of files transferred based on the computed available bandwidth.                                                                                                                                                                                                                                                                                                                                                                                                  |
| Retry duration                | The duration in which the source node tries to establish a connection with the destination node.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| Wait between retries          | The duration the source node waits in between retries.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| File detection cool off       | The duration watchfolder in which placing a new file in the source folder does not trigger a new drop.  
**Note:** This setting does not apply to the **Cool off only** detection strategy.                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |  
| File filters                  | Click the button to add a new filter to identify file lists. You can set a filter to include or exclude files by globbing or by regular expression.                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

Transfer

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target rate</td>
<td>The transfer target rate.</td>
</tr>
<tr>
<td>Minimum rate</td>
<td>The transfer minimum rate</td>
</tr>
<tr>
<td>Bandwidth policy</td>
<td>Choose a transfer policy among fixed</td>
</tr>
<tr>
<td>Transport Encryption</td>
<td>Select aes-128 to transfer with this encryption method.</td>
</tr>
</tbody>
</table>
## Working with Watchfolders

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retry policy</td>
<td>The number of attempts and the duration between each retry.</td>
</tr>
</tbody>
</table>

### Security

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Content Protection | Select **Encrypt transferred files with a password** to enable content encryption. Enter and confirm the password the recipient must use to decrypt the transferred files.  
**Note:** When editing a watchfolder with content protection enabled, you must re-
the content protection password. A password must be provided in order to save changes to the watchfolder. |

### File Handling

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resume policy</td>
<td>Specify a resume policy and the overwrite rule when the file exists on the destination.</td>
</tr>
<tr>
<td>File attributes</td>
<td>Preserve file UIDs, GIDs, or timestamps.</td>
</tr>
</tbody>
</table>
| Source Archiving     | The designated directory source files are moved to after completing a transfer. The transfer's session details page display the archive directory's filepath as the **After transfer** path.  
**Note:** The **After transfer** path will only be visible in the session details of the Console that initiated the transfer. Another Console monitoring the same managed nodes will not have access to the **After transfer** path.  
**Note:** Re-running the transfer may generate a "No such file or directory" error since the source files were moved to the archive directory. You can use archive directory variables in the filepath to define specific archive paths for each drop. Hover over the **Archive directory variables** link for a list of available variables. |
| Source deletion      | Delete the transferred files from the source computer after transfer.        |

### Growing Files

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum parallel transfers</td>
<td>The maximum number of concurrent transfers of growing files watchfolder can initiate.</td>
</tr>
<tr>
<td>Target rate</td>
<td>The target transfer rate.</td>
</tr>
<tr>
<td>Bandwidth policy</td>
<td>The bandwidth policy.</td>
</tr>
<tr>
<td>Transport encryption</td>
<td>Select aes-128 to transfer with this encryption method.</td>
</tr>
<tr>
<td>TCP port</td>
<td>The TCP port to use for this watchfolder.</td>
</tr>
<tr>
<td>fasp™ port (UDP)</td>
<td>The UDP port to use for this watchfolder.</td>
</tr>
<tr>
<td>Completion timeout</td>
<td>The amount of time to wait for the file to no longer change for the session to finish.</td>
</tr>
<tr>
<td>Memory</td>
<td>The maximum amount of memory that the faspstream binary is allowed to use.</td>
</tr>
<tr>
<td>Chunk size</td>
<td>The size of data to pack before sending over the network.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Growing file filters</td>
<td>Click the button to add a new filter to identify growing files. You can set a filter to include or exclude files by globbing or by regular expression.</td>
</tr>
</tbody>
</table>

### Packages / Drops

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package timeout</td>
<td>A package in watchfolder defines a set of files with dependencies. The package timeout defines the time in which watchfolder waits for required files. If the required files do not appear within the duration, files with dependencies are marked as not transferred because of unsatisfied dependencies.</td>
</tr>
<tr>
<td>Final transfer</td>
<td>Defines which file has to be transferred last.</td>
</tr>
<tr>
<td></td>
<td>• Last file in list: The last file in the package list is transferred last.</td>
</tr>
<tr>
<td></td>
<td>• File list: The files are transferred without any specific order.</td>
</tr>
<tr>
<td>File list filters</td>
<td>Click the button to add a new filter to identify file lists. You can set a filter to include or exclude files by globbing or by regular expression.</td>
</tr>
</tbody>
</table>
Running Reports

Creating a Basic Report

Console allows you to create and export custom reports, as well as apply filters and scheduling options. The steps below demonstrate how to configure new, basic reports. To view an example of a basic report, see the three samples in this topic. To learn about creating advanced reports within Console, seeCreating an Advanced Report on page 88. To create an advanced report, click the New Advanced button instead. You can also copy and edit Console's built-in, advanced reports, which are listed on the Manage Report Types page. For further information on advanced reporting, see Creating an Advanced Report on page 88.

1. Go to Reports > Manage Report Types. Click the New Basic button.
2. Enter a name for your report (limited to 75 characters) and a detailed description about the report.
3. Choose the level of detail to show on your report.
   Select a field from the drop-down list to be used as the basis for organizing your report. Console generates a report with a row for each item that matches a chosen field. If you choose more than one field, Console generates a multi-level report. The data in the generated report is grouped in ascending order by the fields selected from the drop-down list. For example, if you select Client address, the data in the report is grouped by the transfer initiator IP addresses. For example, Console groups the five transfers initiated by IP Address 1 in the first grouping, the three transfers initiated by IP Address 2 in the second grouping, and so on.
   Note: Once a field is selected, the drop-down list updates automatically to allow for multiple levels of organization. To remove a level of organization, click the Remove link that appears next to the selected field.
   The drop-down list includes all Console built-in fields and custom fields. For a list of built in fields, see Reference: Basic Report Organization Options on page 161. For more information on custom fields, see Creating Custom Fields on page 91.
4. Select the data columns to include in your report. These include built-in and custom fields.
   Select whether to use basic fields only or both basic fields and advanced fields from the Available Columns drop-down menu. Use the blue arrows to add and remove selected data columns.
   Note: The columns available in the list are determined by the organizational fields chosen in the step before.
5. Configure result sorting.
   Select fields to sort by and whether to sort the data in ascending or descending order
   Grouping and sorting options appear based on the data columns that you chose to include in the report. By default, the report is sorted by the organization field selected in the previous step.
6. Add a filter to show only results matching the entered value.
   For detailed information on Console's filters, please see Reference: Reporting Filters on page 163.
7. Create your report. You can also run it at this time.
   - Click Create: Save the report without running it. You are redirected to the Manage Report Types page where you can see the new report in the list of reports. Custom reports have edit and delete links, which differentiate them from Console's built-in reports. Both custom reports and built-in reports include a copy link for duplicating the report and a run link to view run settings and generate the report.
   - Click Create and Run: Save the report and run it. The new report is added to the Manage Report Types page, but first, you are redirected to the New Report page where you must finalize the report run settings and click the Run Report button to run the report.
Creating an Advanced Report

The following instructions describe how to create advanced reports. To view an example of an advanced report, see Advanced Report Example: Transfer Sessions with High Packet Loss on page 193. For more information about creating basic reports, see Creating a Basic Report on page 87.

**Important:** Aspera recommends you read through the Advanced Report Usage Notes on page 175 before configuring an advanced report.

1. Go to Reports > Manage Report Types. Click the New Advanced button.

   **Note:** You may also modify an advanced report by clicking the edit action for an advanced report that is listed on the Manage Report Types page.

2. Enter a name for your report (limited to 75 characters) and a detailed description about the report.

3. Configure the SQL script text.

   For information on available SQL variables or database field references, click on the **Help** link.

```
CREATE TABLE $FINAL_RESULT_TABLE

SELECT DISTINCT
  t.name,
  t.contact,
  t.last_source_ip AS 'from',
  t.last_dest_ip AS 'to',
  t.started_at,
  t.stopped_at,
  t.status,
  t.bytes_transferred,
  t.files_complete,
  t.files_failed,
  t.files_skipped
FROM
  $STBL_TRANSFERS t
WHERE
  t.started_at < 'REPORT_PERIOD_END'
  AND (t.stopped_at >= 'REPORT_PERIOD_START'
    OR t.stopped_at IS NULL)
ORDER BY
  t.started_at,
  t.id;
```

For a list of available reference variables, see:

- Reference: SQL Variables for Advanced Reports on page 164
- Reference: Database Fields for Advanced Reports on page 166
- Creating Custom Fields on page 91

When creating advanced reports, you may specify a custom variable within the **WHERE clause** (for example, $custom_variable). Once declared within the SQL script text, you can to view and edit the variable by clicking...
**Edit Parameters** the Edit Advanced Report Template page. You are prompted to enter a value for the variable when you run the report.

4. **Optional:** Add a filter in the `WHERE` section of your script. For example, this example script filters out transfers that do not have a reported policy and transfers that do not fall within the specified date range.

```sql
... WHERE
  ts.reported_policy IS NOT NULL
  AND
  ts.started_at < '$REPORT_PERIOD_END'
  AND (ts.stopped_at >= '$REPORT_PERIOD_START'
       OR ts.stopped_at IS NULL)
...`

For a list of available SQL variables you can use, *Reference: SQL Variables for Advanced Reports* on page 164.

5. Create your report. You can also run it at this time.
   - Click **Create**: Save the report without running it. You are redirected to the Manage Report Types page where you can see the new report in the list of reports. Custom reports have edit and delete links, which differentiate them from Console's built-in reports. Both custom reports and built-in reports include a copy link for duplicating the report and a run link to view run settings and generate the report.
   - Click **Create and Run**: Save the report and run it. The new report is added to the Manage Report Types page, but first, you are redirected to the New Report page where you must finalize the report run settings and click the Run Report button to run the report.

**Finalizing and Running a Report**

Console requires you to finalize the report’s run settings before running a report.

1. You can initiate finalizing and running a report in the following ways:
   - After configuring your basic or advanced report, click **Create and Run**.
   - Go to Reports > Manage Report Types from the Console menu and clicking the run link From the Actions column.
   - Go to Reports > Run a Report. Select a built-in or custom report from the list.
   - Go to Reports and click the rerun link from the Actions column for a recently run report.

You are redirected to the New Report page.

2. Name the report.

3. Run the report now or schedule it to run later.
   - Select **Run Now**: Run this report immediately.
   - Select **Run Later**: Schedule a report by setting the run date. You may also select **Repeat** to schedule a repeating report.

4. Define the report period.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Report on</td>
<td>Select a pre-defined time period from the drop-down list.</td>
</tr>
<tr>
<td></td>
<td>• last hour</td>
</tr>
<tr>
<td></td>
<td>• last 24 hours</td>
</tr>
<tr>
<td></td>
<td>• last week</td>
</tr>
<tr>
<td></td>
<td>• month to date</td>
</tr>
</tbody>
</table>
### Option Description

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>• last month • custom</td>
<td></td>
</tr>
<tr>
<td>Start date</td>
<td>Select the start date of this report. You must select custom in the dropdown menu to modify this field.</td>
</tr>
<tr>
<td>End date</td>
<td>Select the end date of this report. You must select custom in the dropdown menu modify this field.</td>
</tr>
<tr>
<td>Time zone</td>
<td>Select the time zone for this report.</td>
</tr>
</tbody>
</table>

5. Enter values for your custom SQL variables under **Report Parameters**. If there are no values, no custom variables were specified for this report.

   For more information on custom variables, see *Editing Custom Variables* on page 90.

6. Optional: Enter an email address and click the Add button to email a recipient a copy of this report.

   After adding an email address, select whether the report is sent as an XLSX or a CSV file.

7. Optional: Choose additional file formats (XLSX and CSV). These files can be downloaded after the report has been generated.

8. Click **Run Report** after finalizing your settings.

   Your generated report is listed on the Scheduled and Recently Run Reports page. When viewing your report, you have the following options:

   • For a custom report, click **Edit Report Type** to configure report.
   • To run the report again, click **Rerun**.
   • If you chose to export your report in CSV or XLSX, click the respective button to download the files.

---

### Editing Custom Variables

When creating advanced reports, you can specify a custom variable within the **WHERE clause**. For example, to create a search by contact, enter:

```sql
... WHERE contact = '$CONTACT_MATCH'; # $CONTACT_MATCH is the custom variable.
...```

Once you declare the variable within the SQL script text, you can view and edit the variable on the Edit Advanced Report Template page.

1. To edit a custom SQL variable used in an advanced report, go to **Reports** and click the **Manage Report Types** button. Find the advanced report and click **edit**. Click **Edit Parameters**.

2. Find the custom variable you want to configure and click **edit**.

3. Select the desired variable type from the **Type** drop-down menu.

<table>
<thead>
<tr>
<th>Variable Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>The value of this variable must be a string.</td>
</tr>
<tr>
<td>integer</td>
<td>The value of this variable must be an integer.</td>
</tr>
<tr>
<td>date</td>
<td>The value of this variable must be a valid date. Click the calendar icon to select a valid date.</td>
</tr>
<tr>
<td>ip</td>
<td>The value of this variable must be a valid IP Address.</td>
</tr>
</tbody>
</table>
4. Optional: Allow the user running the report to leave the variable undefined by clearing **Is field required?**. Custom variables are required by default.

If **Is field required?** is selected, a user running this report is required to enter a value for the custom variable to run the report.

**Note:** If the custom variable is *not* required *and* it is used with the **AND** operator, then write the report query as follows:

```
WHERE
...
AND (
    t.status = '$FOO'
    OR '$FOO' = ''
)
...
```

Failure to include *OR* '$FOO' = '' results in an empty report because the data is filtered by t.status = '', which is always false.

5. Optional: Define the variable name that is displayed when Console asks for the value of this variable.

For example, if you want to search by contacts and included a custom variable named $CONTACT_MATCH in your SQL script, Console by default prompts the user running the report to enter a value for "Contact Match." If you enter "User Name" in the **Label** field, Console asks for a value called "User Name" instead and matches the result to $CONTACT_MATCH.

6. Optional: Add a hint to remind the user the purpose of this variable.

Continuing the previous example, if your custom variable, $CONTACT_MATCH, is used to search your database for contacts matching the value of this variable, a possible hint is: "Search by this CONTACT name."

When running the report, the user is prompted with the following:

```
... WHERE ...
AND (  
    t.status = '$FOO'
    OR '$FOO' = ''
)  
...
```

7. When finished, click **Update**.

---

### Creating Custom Fields

Custom fields are used to specify rules for automatically populating fields in basic and advanced reports.

1. Go to **Configuration > Custom Fields**.
2. Click **New Custom Field**.
3. Select **transfer** or **file** from the **Level** drop-down list, depending on whether the new custom field stores transfer- or file-related content.
4. Enter a name for the custom field. The name must be unique and lowercase.

   The resulting SQL name is prefixed with "cf_." For example, the field name "metadata" appears as "cf_metadata".

   **Note:** Custom fields appear in the database with the "cf_" prefix. Custom fields are utilized in the $TBL_FILES and $TBL_TRANSFER tables.
5. Enter the start date (date on which to start custom field calculation).
6. Enter a custom field description.
7. Click **Create**.
8. Create and associate new rules for your custom field.

Rules are conditions that define when the custom field to comes into effect. To set up the rule's conditions, configure the following settings:

- Select a built-in field from the drop-down list.
- Enter an operator.
- Enter an expression.
- Enter the value Console uses to populate the custom field if conditions are met.

For a list of field names and definitions, see Reference: Built-In Fields for Custom Field Rules on page 161.

For example, to create a custom field that is populated with your company name, create a new custom field and associate it with the following rule:

![Editing Rule for Custom Field: company_name](image)

9. Click Create.

For each custom field, you can create multiple rules that populate with different values based on various conditions. When multiple rules are present, Console uses the first rule listed (as long as it matches the condition). To modify the order of the custom field rules, use the drag-and-drop function to move the rules in the list.

When creating an advanced report, you can find your available custom fields by clicking the Help link in the SQL Script Text section.
For an example of using a custom field in a report, see *Advanced Report Example: Transfer Sessions with High Packet Loss* on page 193.
Configuring SSH Keys

SSH Keys

SSH keys provide a more secure way to authenticate than using passphrases. Console generally uses SSH keys for two purposes:

- Authentication to administer and configure a node.
- Authentication to make a transfer from one node to another.

You can store keys and find a list of existing keys by navigating to the SSH Private Key page in either of two locations:

- **Personal Preferences:** Select Preferences from the drop-down menu next to your username in the upper right-hand corner. Then, select the SSH Keys tab.
- **Console Configuration:** Go to Configuration > SSH Keys from the Console menu.

For more information on storing keys, see *Storing SSH Keys on Console* on page 94.

The steps to using an SSH key differ if you are using an SSH key to make a transfer or using one to make a transfer to nodes with endpoints that use SSH keys.

**Using SSH Keys in Transfers**

A user must add an SSH key in his personal preferences before he can use that key in a transfer. Even if the SSH key is configured in Console Configuration settings, if the user did not the key in his personal preferences, the key does not appear when he enters the credentials for a node to set up a transfer.

**Making Transfers to Nodes With Endpoints that Use SSH keys**

When making transfers to nodes with endpoints using SSH keys, the transfer user on the initiating node also needs to have the private key in the .ssh folder. For a walkthrough of this process, see *Transferring Files with an Endpoint Using SSH Keys* on page 95.

**Storing SSH Keys on Console**

Console uses a node machine's private key to authenticate into the machine using public key authentication. You must first store in Console the private key paired with the public key on the node machine. You can store private keys privately in your user preferences or globally in Console configurations. These SSH keys can then be used to authenticate endpoints or transfers.

1. Go to your private SSH keys or Console's stored SSH Keys.
   - **Personal Preferences:** Select Preferences from the drop-down menu next to your username in the upper right-hand corner.
Then, click the **SSH Private Keys** tab.

- **Console Configuration:** Go to **Configuration > SSH Keys.**

2. Click **New SSH Private Key.**

3. Enter a descriptive name to represent the SSH key in Console.

4. Enter the filename of the key as it exists on the node. Do not include the directory.

5. Upload the private key file provided by the node administrator.

6. Enter and confirm the passphrase of the key, if any.

7. Click **Save.**

8. Click **Test** to test the new SSH private key.

   Provide the following information:
   - The address of the computer that has the paired public key installed in their **authorized_keys** file.
   - The corresponding user name.

   Then, click **Connect with SSH Key** to test against the computer.

   - **Tip:** If the connection fails, contact the node administrator to make sure the public key is properly installed in the **authorized_keys** file.

---

### Transferring Files with an Endpoint Using SSH Keys

The objective of this example is to set up two nodes in Console to transfer files from one node to the other using public key authentication.

- **User A:** Transfer user found on Node A.
- **User B:** Transfer user found on Node B.
- **Node A:** The node initiating the transfer. This node holds the private key.
- **Node B:** The node receiving the files. This node holds the public key matching the private key in Node A. We set up an endpoint using SSH keys for this node.

   - **Note:** For the purpose of this example, both nodes are Linux machines.

1. Generate a private key as User A on Node A with the following command:

   ```
   # ssh-keygen -t rsa
   ```

   Choose the default location to store this new private key (Default is `~/.ssh`).

2. Make sure **User A** has read and write permissions for the private key file.

   ```
   $ chmod 600 ~/.ssh/id_rsa
   $ chmod 644 ~/.ssh/id_rsa.pub
   ```

3. Copy the SSH public key into User B’s **authorized_keys** file on Node B.

   ```
   # cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
   ```

4. In Console, add Node A as a managed node and Node B as an unmanaged node.

5. Go to **Configuration > SSH Keys** and upload the private key to Console. This key should be paired with the public key copied to Node B.

6. Go to **Nodes** and edit Node B. Click **Endpoints** and add a new endpoint. Choose to use the SSH key that was uploaded to Console.

7. Make a transfer from User A on Node A to the saved endpoint on Node B.
Installing a Signed SSL Certificate Provided by Authorities

In a default IBM Aspera Shares installation, Apache generates and uses a self-signed SSL certificate. You can find this certificate at the following location:

- /opt/aspera/common/apache/conf/server.crt
- /opt/aspera/common/apache/conf/server.key

To set up a signed SSL certificate, follow these steps:

1. Enter the OpenSSL command to generate your Private Key and Certificate Signing Request (CSR).

Run the following command (where key_name.key is the name of the unique key that you are creating and csr_name.csr is the name of your CSR):

```
$ openssl req -new -nodes -newkey rsa:2048 -keyout key_name.key -out csr_name.csr
```

After entering the command, you are prompted to enter several pieces of information, which are the certificate's X.509 attributes.

**Important:** The Common Name field must be filled in with the fully qualified domain name of the server to be protected by SSL. If you are generating a certificate for an organization outside of the US, see [https://www.iso.org/obp/ui/#search/code/](https://www.iso.org/obp/ui/#search/code/) for a list of 2-letter, ISO country codes.

```
Generating a 1024 bit RSA private key
.........................++++++
.........................++++++
writing new private key to 'my_key_name.key'
-----
You are about to be asked to enter information that will be incorporated into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:Your_2_letter_ISO_country_code
State or Province Name (full name) [Some-State]:Your_State_Province_or_County
Locality Name (eg, city) [J]:Your_City
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Your_Company
Organizational Unit Name (eg, section) []:Your_Department
Common Name (i.e., your server's hostname) []:secure.yourwebsite.com
Email Address []:johndoe@yourwebsite.com
```

**Note:** You are prompted to enter "extra" attributes, including an optional challenge password. Manually entering a challenge password when starting the server can be problematic in some situations (for example, when starting the server from the system boot scripts). You can skip entering values for any extra attribute by hitting the "enter" button.

```
... Enter the following 'extra' attributes to be sent with your certificate request
A challenge password []:
```
After finalizing the attributes, the private key and CSR will be saved to your root directory.

**Important:** If you make a mistake when running the OpenSSL command, you may discard the generated files and run the command again. After successfully generating your key and Certificate Signing Request, be sure to guard your private key, as it cannot be re-generated.

2. Send CSR to your signing authority.

You now need to send your unsigned CSR to a Certifying Authority (CA). Once the CSR has been signed, you have a real Certificate. Follow the key provider's instructions to generate and submit both your private key and the Certificate Signing Request (CSR) to acquire the certificate. Here is a list of commonly used certificate authorities.

**Important:** Some Certificate Authorities provide a Certificate Signing Request generation tool on their Website. Check with your CA for additional information.

At this point, you may need to generate a self-signed certificate because:

- You don't plan on having your certificate signed by a CA.
- You wish to test your new SSL implementation while the CA is signing your certificate.

For information on how to generate a self-signed certificate for temporary use, see Generating a New Self-Signed SSL Certificate on page 97.

3. Copy key and certificate into target directory.

After receiving your signed certificate from your CA, copy the files into Apache's `/conf` directory and edit your `httpd-ssl.conf` file. Note that you can store the certificate and key in any directory, as long as the paths are updated in your configuration file.

4. Store your certificates on your machine.

For example:

- `~/my_server.crt`
- `~/my_server.key`

Your certificate provider may require you to also install an Intermediate CA Certificate file. Copy the file to the following location:

```
/opt/aspera/common/apache/conf/server-ca.crt
```

5. Install the SSL certificate with the following command:

```
$ asctl apache:install_ssl_cert cert_file key_file [chain_file]
```

For example:

```
$ asctl apache:install_ssl_cert ~/my_server.crt ~/my_server.key /opt/aspera/common/apache/conf/server-ca.crt
```

You can find the installed certificate at the following location:

- `/opt/aspera/common/apache/conf/server.crt`
- `/opt/aspera/common/apache/conf/server.key`

---

**Generating a New Self-Signed SSL Certificate**

You may need to generate a self-signed certificate because:

- You don't plan on having your certificate signed by a CA.
- You wish to test your new SSL implementation while the CA is signing your certificate.
Generate a self-signed certificate using OpenSSL. This temporary certificate will generate an error in the client's browser that warns the client that the signing certificate authority is unknown and not trusted. To generate a temporary certificate (which is good for 365 days), run the following command:

```bash
# openssl x509 req -days 365 -in csr_name.csr -signkey key_name.key -out cert_name.crt
```

### Regenerating Self-Signed SSL Certificate (Apache)

When you initially set up Console on your system a pregenerated, self-signed SSL certificate is also installed. If you have changed your Apache hostname, regenerate the self-signed certificate by following the instructions below.

1. Open a terminal window and run the `asctl` command.

   In a terminal window, run the following command to generate a new, self-signed SSL certificate for your installation of Aspera Console (where you will replace the `HOSTNAME` with your Apache server's IP address or host name):

   ```bash
   $ asctl apache:make_ssl_cert HOSTNAME
   ```

   Answer `yes` when prompted to overwrite the existing certificate.

2. Confirm that your certificates are updated.

   Check the following location to confirm your self-signed SSL certificates have been updated:
   
   - `/opt/aspera/common/apache/conf/server.crt`
   - `/opt/aspera/common/apache/conf/server.key`
Working with Shares and Directory Services

Console and Shares on Same Machine

Important: This topic assumes that you have already installed IBM Aspera Shares. If you have not installed Shares yet, please see the IBM Aspera Shares Administrator's Guide.

If you installed Console on the same machine as Shares, you must update the host and port settings in Shares' database.yml file. Your database.yml file can be found in the following directory:

/opt/aspera/shares/www/config/database.yml

Open database.yml with a text editor and perform the following modifications:

• Comment out the socket location.
• Change the host to 127.0.0.1.
• Change the TCP port to 4406.

After performing these modifications, your database.yml file should look similar to the example below.

```
production:
  adapter: mysql2
  encoding: utf8
  reconnect: false
  database: web_production
  pool: 5
  username: admin
  password: v00d00
  # socket: /tmp/mysql.sock
  # socket: /var/lib/mysql/mysql.sock
  host: 127.0.0.1
  port: 4406
```

Configuring the Directory Service

Important: You must install IBM Aspera Shares locally or on a remote host before you can configure a directory service. For information on installing the latest version of Shares, please review the Administrator's Guide.

Before continuing, please ensure that the following prerequisites have been satisfied:

• Shares is installed locally or on a remote host: For instructions on how to install Shares on the same machine as Console, see Console and Shares on Same Machine on page 99.
• Console is installed on the same machine as Shares: Configure your Shares web server to use a non-standard HTTPS port (for example, 8443, rather than 443). See the Shares Administrator's Guide (the "Setting up Shares" topic).
• Console is installed on the same machine as Shares: Configure your Shares database with the correct host and port settings. For more information, see Working with Shares and Directory Services on page 99.

1. Go to Accounts > Directories from the Console menu.
2. Select Remote Authentication to enable remote authentication so that Console can access the groups and users on your Shares server.
3. Enter the base URL.

Shares users and groups are authenticated through this Node API base URL. The standard base URL is https://shares_IP_address/api/v1.
Note: Because you must use HTTPS to connect to your Shares directory service, ensure that your Node API base URL uses HTTPS, rather than HTTP.

4. Enter the Shares Node API user credentials.

5. Click **Save and test settings**.

Once the directory service is successfully connected, you can add remote users and remote groups by going to **Users** > **Groups**. For more information, see *Adding Remote Users* on page 100 and *Adding Remote Groups* on page 100.

---

### Adding Remote Users

1. Go to the **Users** menu. Click **Add Remote User**.

   Note: The **Add Remote User** button does not appear if you have not configured the directory service.

2. Enter the full or partial name of an existing remote user. Click **Search**.

3. Once your search results appear, select the remote user by clicking **Add**.

4. Configure the remote user's Console permissions and assign the user to a group.

---

### Adding Remote Groups

1. Go to **Groups**. Click **Add Remote Group**.

   Note: The **Add Remote User** button does not appear if you have not configured the directory service.

2. Enter the full or partial name of an existing remote user. Click **Search**.

   Note: Console does not find remote groups that start with a backslash (\) or an asterisk (*). Avoid naming groups that start with these characters.

3. Once your search results appear, select the remote group by clicking **Add**.

4. Configure the remote group's transfer paths and members.
**Working with SAML**

IBM Aspera Console supports Security Assertion Markup Language (SAML) 2.0, an open, XML-based standard that allows secure web domains to exchange user authentication and authorization data. With the SAML model, you can configure the Console web application as a SAML "online service provider" (SP) that contacts a separate online "identity provider" (IdP) to authenticate users who will use Console to access secure content.

With SAML enabled and configured, a user logging into Console is redirected to the IdP sign-on URL. If the user has already signed in with the IdP, the IdP sends a SAML assertion back to Console. The user is now logged into Console.

When SAML is enabled, Console creates a user account based on the information provided by a SAML response, and therefore the Console user account does not need to be created manually. However, any changes to the account that are made on the DS server are not picked up by SAML.

These instructions assume you are already familiar with SAML and already have an identity provider (IdP) -- either third-party or internal -- that meets the following requirements:

- Can be configured to use an HTTP POST binding
- Can be connected to the same directory service being used by Console (however, SAML and DS cannot be used together)
- Are not configured to use pseudonyms
- Can be configured to return assertions to the SP (Console) that include the entire contents of the signing certificate

**Note:** SAML and directory services should not be enabled together. Although there is a directory service behind a SAML IdP, Console users will not have access to it. If Console is being set up to use SAML, the following is recommended: (1) directory service sync should be disabled; and (2) existing directory service users should first be removed from the Console system.

Console provides a mechanism for administrators to bypass the SAML login and log in using a local username and password. This allows administrators to log in and correct server settings, including a misconfigured SAML setup. To bypass the SAML login and sign in with the regular login, add local=true to the end of the login URL. For example: https://10.0.0.1/aspera/console/login?local=true

**Configuring SAML**

Before following the instructions below, have the following information on hand:

- IdP Single Sign-On URL (SSO)
- IdP Certificate Fingerprint OR IdP Certificate

1. Go to **Accounts > SAML** to open the SAML Configuration page.
2. Select **SAML Authentication**.
3. Enter the SSO URL and the SHA1 certificate fingerprint.
4. Click **Save**.

**Configuring Your Identity Provider (IdP)**

1. Provide the Console Entity ID, Assertion Consumer Service (ACS), and base URL. These are the endpoints that the IdP will use to retrieve information about the Console and deliver the SAML response.

| Entity ID: | https://server_name_or_ip/aspera/console/auth/saml/metadata |
ACS:  https://server_name_or_ip/aspera/console/auth/saml/callback

Base URL:  https://server_name_or_ip/aspera/console

2. Set up the SAML attribute mapping to provide information to Console.

The attribute mapping specifies which values to be extracted from the user datastore (LDAP, Active Directory, or database) and passed back to the Console as SAML assertions.

<table>
<thead>
<tr>
<th>Element</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAML_SUBJECT</td>
<td>yes</td>
</tr>
<tr>
<td>email</td>
<td>yes</td>
</tr>
<tr>
<td>given_name</td>
<td>optional</td>
</tr>
<tr>
<td>surname</td>
<td>optional</td>
</tr>
</tbody>
</table>

3. Extract the IdP certificate fingerprint (SHA1). The certificate fingerprint will be specified in Console and used to validate the SAML response from the IdP.

User Accounts Being Provisioned by SAML Just-In-Time (JIT) Provisioning

When new user accounts are being provisioned through SAML JIT Provisioning, new SAML groups are created when the SAML response contains group information, and that group does not yet exist in IBM Aspera Console. A SAML user belonging to multiple groups will get permissions and settings of all groups the user belongs to. For example, if group A disallows sending to external users but group B does not, users who belong to both groups are allowed to send to external users. Settings that require specific handling are as follows:

- Account expiration is only enabled if all groups to which a user belongs specify account expiration. If account expiration is enabled, the expiration date is set to the latest expiration date from among all groups.
- For the settings “Server Default”, “Yes” or “Allow”, and “No” or “Deny”, the setting is set to “Yes” if any group specifies yes, and it is set to “No” if all groups are set to no. Otherwise it is set to the server default.
- For package deletion policy, override is enabled if all groups specify override, or the least restrictive group setting is less restrictive than the server-wide setting. If override is enabled, the least restrictive group setting is used. “Do nothing” is less restrictive than “Delete files after all recipients download all files,” which in turn is less restrictive than “Delete files after any recipient downloads all files.”
- For advanced transfer settings, override is enabled if all groups specify override or if any group specifies any transfer rate that is higher than the server default. If override is enabled, each transfer rate is set to the higher of the highest value from among the groups and the server default. The minimum rate policy is locked only if all groups specify the setting.
Backing Up Console Database

Back Up Console with asctl

There are two different ways to back up the Console database:

1. Through the asctl command, which backs up only the MySQL database. Use this method before a Console upgrade procedure, or to guard against possible database corruption.

2. Through the Console web UI, which backs up the MySQL database in addition to all the files required to fully restore the Console application. Use this method for disaster recovery purposes, in order to restore Console when the entire server is lost.

Back up Console's database using the following asctl command in a Terminal:

```bash
$ asctl -v console:backup_database
```

This command uses mysqldump to create Console's MySQL database backup. The backup file, aspera_console.sql, is saved in the following directory:

```
/opt/aspera/console/backup/<year-month-day_time>
```

For instructions on restoring your Console database, see Restoring the Console Database on page 103.

Back Up Console with the Web UI

There are two different ways to back up the Console database:

1. Through the asctl command, which backs up only the MySQL database. Use this method before a Console upgrade procedure, or to guard against possible database corruption.

2. Through the Console web UI, which backs up the MySQL database in addition to all the files required to fully restore the Console application. Use this method for disaster recovery purposes, in order to restore Console when the entire server is lost.

1. Select Configuration > Database from the Console menu. Click Back Up.
2. Enter the desired path of the Console machine into the Save to field. This path is the destination folder for the console_full_backup_YYYY-MM-DD_hhmmss backup file.
3. Schedule the backup to Run Now or to Run Later.
   - Click Run now: Back up the database immediately.
   - Click Run later: Specify a time in the future or configure a repeating backup operation.
4. Click Back Up Now.

Once Console has been backed up, the backup file appears on the Database Backups page, where scheduled, current, and recent backups are listed. To view details on a particular backup, click anywhere in the backup's row.

Restoring the Console Database

You can restore any back up of a Console database as long as you have access to the backup file.

1. Stop Console.
   ```bash
   $ asctl console:stop
   ```
2. Restore the Console database.
• If you made a back up of the Console database with the asctl command, you can restore it with the following command:

```
$ asctl -v console:restore_database /path/to/dir
```

For example:

```
$ asctl -v console:restore_database /opt/aspera/console/backup/2013-1-16_164305
```

• If you made a back up of the Console database with the web UI, you can restore it with the following command:

```
$ asctl -v console:restore /absolute/path/to/dir
```

For example:

```
$ asctl -v console:restore /tmp/console_full_backup_2013-1-16_00.57.28_UTC
```

Important: The restore command does not support relative paths to the backup directory. The path must be an absolute path in order for the restore command to work.


```
$ asctl console:start
```

---

**Backing Up the Current Console Configuration**

If you need to revert to a previous configuration of Console or want to upload a preset configuration file to a new Console, follow these instructions to backup and restore your Console configurations.

Note: This Console backup will not back up the following files:

- Node users
- SSL certificates

1. Stop Console background processes.
   Go to **Configuration > Background** from the Console menu and stop the background processes.

2. Back up SSH keys.
   Back Up the following directory: `/root/.ssh`.

3. Back up node users.
   ```
   # /opt/aspera/bin/asnodeadmin --backup=/backup/api-xfer-mapping
   ```

4. Back up SSL Certificates.
   Note: This step is only applicable if you have purchased SSL certificates and IBM Aspera Shares is installed on your machines.
   ```
   # cp /opt/aspera/shares/conf/cert.key /backup/cert.key
   # cp /opt/aspera/shares/conf/cert.pem /backup/cert.pem
   ```

5. Navigate to **Configuration > Save/Restore** and select **Download Current Configuration**.
   Important: If you use the Safari web browser, you need to make sure the Open "safe" files after downloading option is unchecked in Safari's general settings, before downloading the backup file.
Otherwise, the file will be downloaded as a .tar file, rather than a .tar.gz file, and will not work when the user attempts to restore the server with this file.

6. Restart Console background processes from **Configuration > Background**.

To restore Console to a saved configuration, see *Restoring the Current Console Configuration* on page 105.

---

**Restoring the Current Console Configuration**

1. Stop Console background processes.
   - Go to **Configuration > Background** from the Console menu and stop the background processes.

2. Navigate to the **Configuration > Save/Restore** tab and upload a saved configuration file from your computer and select **Restore**.

3. Restore your SSH key directory to the following location: `/root/.ssh`.

4. Restore node users.

   ```bash
   # /opt/aspera/bin/asnodeadmin --restore=/backup/api-xfer-mapping
   ```

5. Restart the Aspera NodeD service.

   ```bash
   # service asperanoded restart
   ```

6. Restart Console background processes from **Configuration > Background**.
Managing the MySQL Database

Configure MySQL Settings

You may want to modify the MySQL settings for security or management purposes.

**Change the Database root Login Password**
MySQL database's root account's password is set during the setup process. For security reason, it is recommended to update the password. Use the following command to change the password. Enter the new and old password when prompted:

```
$ asctl mysql:set_root_password
```

**Change the MySQL Port**
By default, Console's MySQL uses TCP port 4406. Use the following command to change it.

```
$ asctl mysql:port 1234
```

If the MySQL's port number is changed, you will need to provide the updated Console settings to all the nodes, and reflect the new settings in all the nodes' `aspera.conf` files.

Running MySQL on a Separate Machine

After you have installed Console, you can further configure it to run the MySQL database on a remote machine. Follow these steps to run the web application and the MySQL database on two separate machines:

- **Note:** This setup procedure involves steps on the Console machine and the MySQL machine. A MySQL machine or Console machine is indicated at each step.

1. **(MySQL machine) Download and install common files only**
   On the MySQL machine, download Aspera Common Files and run it with the following command:

   ```
   $ rpm -Uvh aspera-common-version.rpm
   ```

2. **(MySQL machine) Setup MySQL database.**
   On the MySQL machine, execute this command to configure MySQL:

   ```
   asctl mysql:setup
   ```

   When started, the configuration program will ask you to use streamlined or detailed setup. Expect the following setup items in each setup method: (In detailed setup, answer `y` in the first question.)

<table>
<thead>
<tr>
<th>Item</th>
<th>Streamlined</th>
<th>Detailed</th>
</tr>
</thead>
<tbody>
<tr>
<td>MySQL will run on this machine (y/n)? (default: y)</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>What port would you like MySQL to listen on? (default: 4406)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Where would you like MySQL to store data: (default: C:/Program files/Common Files/Aspera/Common/mysql/data)</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>MySQL will need to start/restart during configuration. Continue (y/n)? (Current: y)</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
Lastly, a setup summary shows your settings. Enter y to confirm, n to change settings, or x to quit the program without saving.

When finished, execute this command to allow access for the Console machine. Replace the highlighted items to match your configuration (Enter the Console machine's address in <Console_server_IP>, and your MySQL password in <mysql_password>):

```
asctl mysql:grant_remote_access <Console_server_IP> root <mysql_password>
```

3. **(Console machine)** Configure Console to use a remote MySQL database.

On the Console machine, execute this command to configure it to run MySQL on a remote machine:

```
$ asctl console:setup
```

Answer n to the following question:

MySQL will run on this machine (y/n)? (default: y)

---

### Purging Data from Console

You can archive or purge data from Console (for example, purge all sessions before January 1, 2000) by clicking the **Purge** button from the Database Backups page and completing the fields.

1. Schedule Console to purge the data now or at a later date.
   - **Run now**: Back up the database immediately.
   - **Run later**: Specify a time in the future or configure a repeating purge operation.

2. Select time frame of data to purge.

   Choose the date by entering a number and selecting **day**, **week**, or **month** from the drop-down menu. Make sure the automatically updated date displayed next to the drop-down menu is the desired day before proceeding.

3. Choose the type of transfers to include.

   Select **All closed transfers** or choose from the following list:
   - All successful transfers
   - All cancelled transfers
   - All error transfers
   - All inactive transfers
   - All zero-byte transfers

4. Save the data being purged for archiving purposes.

   Select **Save data being purged?** and enter the desired absolute path into the **Save to** field (for example, /tmp/data or D:\data). The purged data will then be stored in the file **purged_data.sql** in the directory: [absolute path]/console_purge/YYYY-MM-DD_hhmmss/.

   **Tip:** Saved purged data can be restored by following the instructions in **Restoring Purged Data** on page 108.

5. When finished, click the **Purge Now** or **Schedule Purge** button (depending on whether you selected **Run now** or **Run later** above).
Restoring Purged Data

To restore purged data, you can run a MySQL data import (as shown below). It may be necessary to provide a full path to the MySQL binary. For example, the full path to the command is `/opt/aspera/common/mysql/bin/mysql`.

```
# asctl console:stop
# cd /opt/aspera/common/mysql/bin
# ./mysql -uUSERNAME -pPASSWORD aspera_console < /path/to/purged_data.sql
# asctl console:start
```
Troubleshooting Console

Updating your Console License

IBM Aspera Console requires a valid license key before you can configure users and send or receive packages. If your Console license has expired or cannot be found, the Console login screen displays the following message:

![Console license not found](image)

An administrator must update the license before any other usage of Console is allowed for any user, including the administrator.

The license can be updated in the Console web UI or by running a rake task on the computer where Console is installed.

**From the GUI:**

1. Login with an administrator account and go to Configuration > License.
2. Click **Upload a license file** or paste the license text into the text window.
3. Click **Save**.

**From the Command Line (Rake task):**

1. Set the license text as an environment variable.

   ```
   # export LICENSE_TEXT='\<ASPERA_LICENSE\> <DETAILS expiration_date=... \</KEY\> \</ASPERA_LICENSE\>
   ```

In this example, only part of the license text is shown. You must paste the entire license text for the license to be valid.
2. Update the Aspera license:

```
# asctl console:rake aspera:update_license
```

## Restart Console Services

If Console is not working properly, it is recommended you restart the Console service utilizing the `asctl` command, so that Apache and MySQL continue to run uninterrupted. If the Console server's MySQL service is stopped, then Aspera Central will need to be restarted on all nodes to re-establish a connection.

Console installs the following services on your Linux system:

<table>
<thead>
<tr>
<th>Service</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aspera Console</td>
<td>Aspera Console main application.</td>
</tr>
<tr>
<td>MySQL Server (Aspera)</td>
<td>MySQL Database for Aspera Console.</td>
</tr>
</tbody>
</table>

Right-click any of these services select **Restart** from the menu.

Execute the following `asctl` command to restart Console (while keeping Apache and MySQL running):

```
$ asctl console:restart
```

For more `asctl` commands, see *asctl Command Reference* on page 119

## Resetting Console Admin Password

To reset Console's administrator password, execute the following `asctl` command in a Terminal, replacing name with your existing admin login, email with the current admin password, and password with the new admin password:

```
$ asctl console:admin_user name email password
```

## Log Files

Console's log files are located in the following directories:

<table>
<thead>
<tr>
<th>OS Version</th>
<th>Path</th>
</tr>
</thead>
</table>
| Linux      | • Console: /opt/aspera/console/log/
|            | • asctl: /opt/aspera/common/asctl/asctl.log
|            | • MySQL: /opt/aspera/common/mysql/data/mysqld.log
|            | • Apache: /opt/aspera/common/apache/logs/

In Console's Apache HTTP server logs directory, you will find the following files:

- access_log
- error_log
- ssl_access_log
- ssl_error_log
• ssl_request_log

**Important:** All Apache logs are, by default, rotated by size (defaulting to 10MB files and only retaining the last 10 rotated logs).

**httpd_template_linux.conf**

```
/opt/aspera/common/apache/conf/httpd_template_linux.conf
```

• ErrorLog "${log_path}bin/asrotatelogs ${log_path}logs/error_log 10M 10"

• CustomLog "${log_path}bin/asrotatelogs ${log_path}logs/access_log 10M 10" common

**httpd-ssl_template.conf**

```
/opt/aspera/common/apache/conf/extra/httpd-ssl_template.conf
```

• ErrorLog "${log_path}bin/asrotatelogs ${log_path}logs/ssl_error_log 10M 10"

• TransferLog "${log_path}bin/asrotatelogs ${log_path}logs/ssl_access_log 10M 10"

• CustomLog "${log_path}bin/asrotatelogs ${log_path}logs/ssl_request_log 10M 10" "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x "%r" %b"

You can further configure Console's Apache log settings by running the following commands in a Terminal:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify an Apache log level (for example, error level)</td>
<td><code>$ asctl apache:log_level error</code></td>
</tr>
<tr>
<td>Enable Apache log (set to notice)</td>
<td><code>$ asctl apache:enable_logs</code></td>
</tr>
<tr>
<td>Disable Apache log (set to emerg level)</td>
<td><code>$ asctl apache:disable_logs</code></td>
</tr>
</tbody>
</table>

**Locate Configuration Files**

**Important:** Aspera recommends that you DO NOT modify Console's configuration files manually. Instead, use the `asctl` command. For additional information on utilizing `asctl` commands, see the topic *asctl Command Reference* on page 119.

Console's configuration files are listed below. If you plan to modify these files, Aspera encourages backing up Console through the GUI or by using the `asctl` command. The `asctl` command is limited to backing up the Console database, while the GUI backs up the database, as well as all files required to fully restore the system. For instructions on backing up Console through the GUI, please see *Backing Up Console with the Web UI* on page 103. To back up Console's database using the `asctl` command, please see *Back Up Console with asctl* on page 103.

<table>
<thead>
<tr>
<th>Component</th>
<th>Configuration File Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache</td>
<td><code>/opt/aspera/console/config/console.apache.conf</code></td>
</tr>
<tr>
<td>MySQL</td>
<td><code>/opt/aspera/common/mysql/my.cnf</code></td>
</tr>
<tr>
<td>Console</td>
<td><code>/opt/aspera/console/config/*.yml</code></td>
</tr>
</tbody>
</table>
Disabling SELinux

SELinux (Security-Enhanced Linux), an access control implementation, can affect web UI access. To disable SELinux, do the following:

1. Open the SELinux configuration file:

   `/etc/selinux/config`

2. Locate the following line:

   `SELINUX=enforcing`

3. Change the value to `disabled`:

   `SELINUX=disabled`

   Save your changes.

4. On the next reboot, SELinux is permanently disabled. To dynamically disable it before the reboot, run the following command:

   `# setenforce 0`

Configuring Console Defaults

The Console Defaults configuration page lets you set up system defaults for Console, such as IP address and SSH timeout, as well as defaults for transfers (target rate, minimum rate, bandwidth policy, and so on) and login security. To access the Console and Transfer Defaults configuration page, select **Configuration > Defaults** from the Console menu.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Console database IP address</td>
<td>Enter the Console database IP address.</td>
</tr>
<tr>
<td>Warn when database free space less than</td>
<td>The space watcher background jobs warns you when available space drops below the set number of gigabytes. Set to zero to disable space watcher warnings.</td>
</tr>
<tr>
<td>Skip non-error transfers older than</td>
<td>If a submitted transfer doesn't start after the specified number of minutes, then flag it as having an error.</td>
</tr>
<tr>
<td>Mongrel Timeout</td>
<td>Enter the number of seconds to wait for a response when testing mongrels.</td>
</tr>
<tr>
<td>Node Polling Timeout</td>
<td>Enter the number of seconds the SOAP Poller background process waits for a response when testing a node.</td>
</tr>
<tr>
<td>Mark Inactive Timeout</td>
<td>Enter the number of seconds Console waits before marking a session as inactive.</td>
</tr>
<tr>
<td>File Browsing Timeout</td>
<td>Enter the number of seconds to wait for a response from a node when browsing file lists (over and above the SSH timeout to connect).</td>
</tr>
<tr>
<td>Item</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>File Browsing Max Items</td>
<td>Enter the maximum number of items to retrieve from a node when browsing file lists.</td>
</tr>
<tr>
<td>Default SSH Encryption</td>
<td>Select the default SSH encryption algorithm for non-Console nodes. Note: Console presents this algorithm as the standard, but you can change the algorithm when adding a new node.</td>
</tr>
<tr>
<td>Remote Login Connection Timeout</td>
<td>Enter the number of seconds Console waits before timing out when establishing a connection to a remote server.</td>
</tr>
<tr>
<td>Remote Login Response Timeout</td>
<td>Enter the number of seconds Console waits before timing out when waiting for the remote server's response.</td>
</tr>
<tr>
<td>SSH Timeout</td>
<td>Enter the timeout value in seconds for the SSH connection.</td>
</tr>
<tr>
<td>SSH Tunnel Start Port</td>
<td>Start assigning SSH tunnel ports at the specified port number.</td>
</tr>
<tr>
<td>Advanced Search Timeout</td>
<td>Enter the timeout value in seconds before advanced search returns current results.</td>
</tr>
<tr>
<td>Email Notification Delay</td>
<td>Enter the number of seconds to wait after initiating a transfer before producing notification emails.</td>
</tr>
<tr>
<td>Total Bandwidth Graph</td>
<td>Select this option to track total bandwidth usage across all notes on the Dashboard graph.</td>
</tr>
<tr>
<td>Advanced File Search</td>
<td>Select this option to allow users to search the entire database for filenames when using advanced search.</td>
</tr>
<tr>
<td></td>
<td>Note: This may slow down Console if your database contains a large number of files.</td>
</tr>
<tr>
<td>Email Recipients</td>
<td>Select this option to allow email recipients to see each other's addresses.</td>
</tr>
<tr>
<td>Session notifications</td>
<td>Select this option to allow non-admins to access the session notifications page.</td>
</tr>
<tr>
<td>Smart Transfer Start Permissions</td>
<td>Select this option to allow users whose transfer path includes &quot;Any&quot; or addresses without a username to start any matching smart transfer that is shared and uses non-personal endpoints. For example, userA is authorized to use a transfer path that has one endpoint set to 10.0.123.45 and the other set to &quot;Any&quot;. If userB's shared smart transfer is set up with non-personal endpoints on 10.0.123.45 (source) and 10.0.111.11 (destination), it will appear in userA's smart transfers list and can be started by userA.</td>
</tr>
<tr>
<td>Smart Transfer Sharing</td>
<td>Select this option to allow users to share smart transfers with personal logins.</td>
</tr>
<tr>
<td>Smart Transfer Editing</td>
<td>Select this option to allow administrators to edit each other's smart transfers.</td>
</tr>
<tr>
<td>Failover / Load balancing Behavior</td>
<td>Select <strong>Failover + Load balancing</strong> for Console to use the least busy node(s) first. For more information, see <strong>Configure Failover Groups</strong> on page 67.</td>
</tr>
<tr>
<td>Watchfolders</td>
<td>Enable the watchfolder feature in Console.</td>
</tr>
<tr>
<td>Watchfolders per page</td>
<td>Configure the number of watch folders to display per page when browsing configured watch folders.</td>
</tr>
<tr>
<td>Proxy</td>
<td>Select this option to turn on the proxy. This feature enables Console to remotely browse nodes when Console is prohibited from making SSH connections to public IP addresses.</td>
</tr>
<tr>
<td>Proxy: Address</td>
<td>Enter the IP address of the proxy.</td>
</tr>
<tr>
<td>Proxy: Port</td>
<td>Enter the port number for the proxy.</td>
</tr>
<tr>
<td>Proxy: Use SSL</td>
<td>Select this option to use SSL with your proxy.</td>
</tr>
<tr>
<td>Item</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>Proxy: Login</td>
<td>Enter the login for the proxy user.</td>
</tr>
<tr>
<td>Proxy: Password</td>
<td>Enter the password for the proxy user.</td>
</tr>
</tbody>
</table>

### Transfer Defaults

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Rate</td>
<td>Set the default target rate.</td>
</tr>
<tr>
<td>Minimum Rate</td>
<td>Set the minimum rate.</td>
</tr>
<tr>
<td>Bandwidth Policy</td>
<td>Set the default transfer policy (choose among low, high, fair, and fixed).</td>
</tr>
<tr>
<td>Max. Retry Attempts</td>
<td>Set the maximum retry attempts.</td>
</tr>
<tr>
<td>Retry Interval</td>
<td>Set the retry interval in seconds.</td>
</tr>
<tr>
<td>Transport Encryption</td>
<td>Select between not-encrypted or aes-128 encryption.</td>
</tr>
<tr>
<td>File Compare Type</td>
<td>Select a file comparison type to verify transferred files.</td>
</tr>
<tr>
<td>File Overwrite Policy</td>
<td>Select an overwrite policy.</td>
</tr>
</tbody>
</table>

### Report Generation

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retention Period</td>
<td>The number of days to keep generated reports before deleting them automatically.</td>
</tr>
<tr>
<td>Maximum Email Attachment Size</td>
<td>The maximum size in megabytes of CSV/XLSX files that may be sent by email. (Generated files can still be downloaded from the Reports page.)</td>
</tr>
<tr>
<td>File Maximum Data Length</td>
<td>The maximum size in megabytes of the result table for which CSV/XLSX files may be generated. (CSV/XLSX files are not generated if the result table is larger than this.) This setting is useful for preventing Console from trying to convert a giant data set into a file and running out of disk space.</td>
</tr>
<tr>
<td>Maximum XLS file rows</td>
<td>The maximum number of rows allowed for generated XLS files.</td>
</tr>
</tbody>
</table>

### Security

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Session Timeout</td>
<td>Sessions will timeout after the specified number of minutes of inactivity.</td>
</tr>
<tr>
<td>Deactivate Users</td>
<td>Deactivate a Console user if there has been &quot;X&quot; failed login attempts within &quot;X&quot; minutes.</td>
</tr>
<tr>
<td>Prevent concurrent login</td>
<td>If this checkbox is enabled, users can only be logged in from one client at a time.</td>
</tr>
<tr>
<td>Suppress logging of transfer tokens</td>
<td>Select this option to suppress tokens from being written to the database. Existing tokens already in the database are unaffected.</td>
</tr>
</tbody>
</table>

**Note:** After enabling this feature, you may experience some lag before the setting takes effect if a request is already in progress and the node is taking a long time to reply.
### Console Password Options

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Password Expiration</td>
<td>Select this option to expire number of days</td>
</tr>
<tr>
<td>Password Duration</td>
<td>Enter the number of days before passwords expire. Setting the value to 0 will disable this feature.</td>
</tr>
<tr>
<td>Password Reuse Limit</td>
<td>Enter the number of passwords users need to go through before they can reuse an old password. Setting the value to 0 disables this feature.</td>
</tr>
<tr>
<td>Password Requirement Regular Expression</td>
<td>Enter a regular expression to specify password requirements. Leave blank to set no requirements.</td>
</tr>
</tbody>
</table>

**Note:** You can select the **Restore Default** link to reset the password requirement to the following: "Passwords must be at least six characters long, with at least one letter, one number, and one symbol."

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Password Requirement Message</td>
<td>Set a message describing the password requirements for users setting a new password.</td>
</tr>
</tbody>
</table>

### Empty sessions (successfully completed with 0 bytes transferred)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leave in database</td>
<td>Log no-transfer sessions in the database.</td>
</tr>
<tr>
<td>Delete if hot folder</td>
<td>Delete no-transfer sessions that are hot folder sessions.</td>
</tr>
<tr>
<td>Delete all</td>
<td>Delete all no-transfer sessions.</td>
</tr>
</tbody>
</table>

### Understanding Space Watcher

Space watcher is a background process that checks the amount of free space in the database and gives warning when space is running low.

**Space Watcher Functionality**

Once a minute, space watcher runs a `ls` or `dir` command, then writes the free space in bytes to a table named `aspera_db_disk_space_free`. The exact command it executes is:

```
df -k -P "aspera_console_db_directory_path"
```

It only writes one record, always with "id=1". The `aspera_db_disk_space_free` table will never have more than one record in it. This table only has three fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Always equal to 1.</td>
</tr>
<tr>
<td>bytes_free</td>
<td>BIGINT, max value = 9223372036854775807, which is approximately 8191 petabytes</td>
</tr>
<tr>
<td>last_reported_at</td>
<td>The time space watcher last stored an entry in the table.</td>
</tr>
</tbody>
</table>

If the process fails to figure out free space for any reason or fails to connect to MySQL, it does nothing and logs nothing. Successful or not, it then closes its connection and then sleeps for a minute before repeating the process.
**Space Watcher Messages in Console**
Unless warnings have been disabled (by setting the warning threshold to zero), Console checks the `aspera_db_disk_space_free` table when rendering a page. If it sees that there are no records in the table, or that it has been longer than 10 minutes since space watcher last reported, Console displays the following message at the top of the page: "**WARNING: No recent data from database free space watcher**". If the last entry is recent (within 10 minutes) but the number of free bytes is less than the configured warning level (default: 10 gigabytes), it shows a message such as the following: "**WARNING: Database free space low (7.5 GB remaining)**".

---

**Working with Tags**

Tags in Aspera products are JSON (JavaScript Object Notation) strings. Console uses tags to identify transfers and to label Console-initiated transfers. You can find a specific transfer's tags by navigating to a transfer's Session Details page and selecting the Session ID link under the Session State column.

Tags are used in the following tasks:
- Creating simple transfers.
- Creating and starting smart transfers.
- Creating advanced rulesets to filter by tags.
- Creating custom fields with rules involving tags.
- Searching using the Advanced Search.

**The JSON Match Comparison Operator**

Console includes a JSON match operator in the Custom Fields and Advanced Rulesets features, which provide a simple syntax for matching JSON formatted tags included in Aspera transfers. Below are examples of transfer tags in Console and Faspex transfers and instructions for matching them using the JSON match operator.

**Console Transfers**

A Console transfer is defined as any transfer initiated by Console using simple or smart transfers. Tags can be specified in both simple and smart transfers. A Console transfer tag is formatted in the following way:

```json
{"aspera":
{"console":
{"user_specified":
{"key1":"val1", ... , "key3":"val3"}
}
}
}
```

An example of a corresponding JSON match value is shown below:

```
[aspera][console][user_specified][key1]val1
```

**Faspex Transfer**

A Faspex Transfer is any transfer initiated by Faspex. A Faspex transfer tag is formatted in the following way:

```json
{"aspera":
{"faspex":
{"key1":"val1", ... , "key3":"val3"}
}
}
```
The corresponding JSON match value is shown below:

```
[aspera][faspex][key1]val
```

**Note:** It is recommended to use the Faspex Metadata filter for Faspex transfers instead. See *Basic Report Example: Faspex Metadata* on page 189 for more information on Faspex Metadata.

### Regular Expressions in JSON Matches

You can also use regular expressions in a JSON match. Define the regular expression using forward slashes ( / ) like in the example below:

```
[aspera][console][user_specified]/.+/
```

**Important:** Aspera advises against using regular expressions in keys, because the result will be the first value that matches the regular expression. In the example below, Console will return the first Faspex transfer it hits without backtracking to check for other transfers that meet the requirements.

```
[aspera][faspex]/.+/[user_specified]/.+/
```

### Configure Background Processes

The Background Processes configuration page displays all Console processes and allows you to perform the following tasks:

- View a process log
- Edit a process
- Stop a process (although this is not available for all processes)
- Restart a process (although this is not available for all processes)

To access the Background Processes page, select **Configuration > Background** from the Console menu.

The following background processes can be accessed from the table:

- Controller
- Mongrel Manager
- Database Ingest
- Session Data Collector
- Node Info Collector
- File Data Collector
- Data Canonicalizer
- Custom Field
- Database Utility
- Transfer Initiator
- Email
- Report

To modify the settings for a given process, click the **edit** link in the corresponding table row. After clicking **edit**, the *Editing Background Process* page appears, along with the following options:

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Startup type</td>
<td>Select the way that the background process starts (that is, manually or automatically) or disable the process from starting altogether.</td>
</tr>
<tr>
<td>Log level</td>
<td>Select the preferred level of logging for the log file output to control the verbosity of the log file output. Choose debug, info, warn, error or fatal.</td>
</tr>
</tbody>
</table>
### Configure the Apache HTTP Server

You may configure Console's Apache HTTP Server to use a different host name, communication port, and namespace using asctl commands.

#### Change the Number of Mongrel servers

By default, Console opens four mongrel servers. To change it, for example, from the default (4) to 10, use the following command:

```
$ asctl console:mongrel_count 10
```

#### Update the Hostname

During the installation, you should have configured the Console's hostname. Use this command to print the current hostname:

```
$ asctl apache:hostname
```

To change the hostname, use the following command. Replace `HOSTNAME` with the new hostname:

```
$ asctl apache:hostname HOSTNAME
```

**Important:** When changing the hostname, the server's SSL certificate should be regenerated. Select (y) when prompted to generate a new SSL certificate.

When the hostname is updated, advise your clients of the new URL. In this example, use the following address: `http://HOSTNAME/aspera/console`

#### Change HTTP and HTTPS ports

By default, Console's web servers are running on TCP/80 (HTTP) and TCP/443 (HTTPS). Use the following commands to update these ports (where, in this example, we TCP/7080 for HTTP and TCP/7443 for HTTPS):

<table>
<thead>
<tr>
<th>Item</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP</td>
<td>$ asctl apache:http_port 7080</td>
</tr>
<tr>
<td>HTTPS</td>
<td>$ asctl apache:https_port 7443</td>
</tr>
</tbody>
</table>
**Change Console namespace**
Console uses the namespace `/aspera/console` by default. Use this command to print the current namespace:

```
$ asctl console:uri_namespace
```

To set the namespace to, for example, `/console`, use the following command:

```
$ asctl console:uri_namespace /console
```

When the namespace is updated, advise your client of the new URL. For example, if your Console server's address is `10.0.0.10`, use this URL:

```
https://10.0.0.10/console
```

**Note:** Refer to *asctl Command Reference* on page 119 for a complete asctl command reference.

### asctl Command Reference
You can use `asctl` commands in a Terminal window to display or modify IBM Aspera Console's component settings. Console configuration options that can be modified using `asctl` are listed below. If there are modifications that cannot be accomplished with `asctl`, notify Aspera Support.

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache</td>
<td>Apache web server.</td>
</tr>
<tr>
<td>Console</td>
<td>Console main application.</td>
</tr>
<tr>
<td>MySQL</td>
<td>MySQL database.</td>
</tr>
</tbody>
</table>

#### All components commands

**Important:** The commands in this section control all Console components.

<table>
<thead>
<tr>
<th>Task</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show config info</td>
<td>asctl all:info</td>
<td>Print info about all components.</td>
</tr>
<tr>
<td>Restart all components</td>
<td>asctl all:restart</td>
<td>Restart all components.</td>
</tr>
<tr>
<td>Setup status</td>
<td>asctl all:setup_status</td>
<td>Information about configuring all components.</td>
</tr>
<tr>
<td>Start</td>
<td>asctl all:start</td>
<td>Start all components.</td>
</tr>
<tr>
<td>Show status</td>
<td>asctl all:status</td>
<td>Display the status of each component.</td>
</tr>
<tr>
<td>Stop</td>
<td>asctl all:stop</td>
<td>Stop all components.</td>
</tr>
<tr>
<td>Show version</td>
<td>asctl all:version</td>
<td>Display the current version of each component.</td>
</tr>
<tr>
<td>Task</td>
<td>Command</td>
<td>Additional Information</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>------------------------</td>
</tr>
<tr>
<td>Create a setup file</td>
<td><code>asctl apache:create_setup_file file</code></td>
<td>Create a reusable file that contains answers to the setup questions. Replace <code>file</code> with a file name.</td>
</tr>
<tr>
<td>Disable Apache</td>
<td><code>asctl apache:disable</code></td>
<td>Disable the Aspera Apache server. When disabled, the service will not start when rebooting computer, does not print reminders or update its configurations.</td>
</tr>
<tr>
<td>Disable Apache logs</td>
<td><code>asctl apache:disable_logs</code></td>
<td>Set the Apache's log level to 'emerg'.</td>
</tr>
<tr>
<td>Enable Apache logs</td>
<td><code>asctl apache:enable_logs</code></td>
<td>Set the Apache's log level to 'notice'.</td>
</tr>
<tr>
<td>Re-generate conf</td>
<td><code>asctl apache:generate_config</code></td>
<td>Generate the component's configuration file using the current settings.</td>
</tr>
<tr>
<td>Display hostname</td>
<td><code>asctl apache:hostname</code></td>
<td>Display the hostname or IP address of the server.</td>
</tr>
<tr>
<td>Change hostname</td>
<td><code>asctl apache:hostname host</code></td>
<td>Change the hostname or IP address of the server. Replace <code>host</code> with a new hostname or IP address.</td>
</tr>
<tr>
<td>Display HTTP port</td>
<td><code>asctl apache:http_port</code></td>
<td>Display the HTTP port the web server listens to.</td>
</tr>
<tr>
<td>Change HTTP port</td>
<td><code>asctl apache:http_port port</code></td>
<td>Change the HTTP port the web server listens to. Replace <code>port</code> with a new port number.</td>
</tr>
<tr>
<td>Display HTTPS port</td>
<td><code>asctl apache:https_port</code></td>
<td>Display the HTTPS port the web server listens to.</td>
</tr>
<tr>
<td>Change HTTPS port</td>
<td><code>asctl apache:https_port port</code></td>
<td>Change the HTTPS port the web server listens to. Replace <code>port</code> with a new port number.</td>
</tr>
<tr>
<td>Show config info</td>
<td><code>asctl apache:info</code></td>
<td>Print configuration info about Apache.</td>
</tr>
<tr>
<td>Copy your SSL files into the Aspera default location (under default names)</td>
<td><code>asctl apache:install_ssl_cert cert_file key_file [chain_file]</code></td>
<td>After upgrading Faspex and Common, use this command to copy your original SSL certificate, key and optional chain file to <code>/opt/aspera/common/apache/conf</code> and give them Aspera-standard names. The <code>httpd-ssl.conf</code> file is also re-rendered and permissions/ownership is set for the cert files.</td>
</tr>
<tr>
<td>Set Apache log level</td>
<td><code>asctl apache:log_level option</code></td>
<td>Specify the Apache's log level. Replace <code>option</code> with <code>crit</code>, <code>error</code>, <code>warn</code>, <code>notice</code>, <code>info</code> or <code>debug</code>.</td>
</tr>
<tr>
<td>Task</td>
<td>Command</td>
<td>Additional Information</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Create SSL certificate</td>
<td>asctl apache:make_ssl_cert hostname</td>
<td>Create a self-signed SSL certificate for the specified hostname. Replace hostname with your hostname.</td>
</tr>
<tr>
<td>Restart Apache</td>
<td>asctl apache:restart</td>
<td></td>
</tr>
<tr>
<td>Configure Apache</td>
<td>asctl apache:setup</td>
<td></td>
</tr>
<tr>
<td>Configure Apache using saved file</td>
<td>asctl apache:setup_from_file filename</td>
<td>Run setup using the answers from a file created using the &quot;create_setup_file&quot; command.</td>
</tr>
<tr>
<td>Start Apache</td>
<td>asctl apache:start</td>
<td></td>
</tr>
<tr>
<td>Show Apache status</td>
<td>asctl apache:status</td>
<td></td>
</tr>
<tr>
<td>Stop Apache</td>
<td>asctl apache:stop</td>
<td></td>
</tr>
<tr>
<td>Upgrade Apache</td>
<td>asctl apache:upgrade</td>
<td></td>
</tr>
<tr>
<td>Show Apache's version</td>
<td>asctl apache:version</td>
<td></td>
</tr>
</tbody>
</table>

**Console**

<table>
<thead>
<tr>
<th>Task</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create or update admin</td>
<td>asctl console:admin_user login email [password]</td>
<td>Create a new admin, or update an existing admin account. Replace login with a login, email with its email. You can add the account's password in the command ([password]), or enter it when prompted. If the login you have entered exists, the account is updated with new email and password.</td>
</tr>
<tr>
<td>Backup database</td>
<td>asctl console:backup_database dir</td>
<td>Backup Console database and associate files to the specified directory. Replace dir with a path to store the backup.</td>
</tr>
<tr>
<td>Display base port</td>
<td>asctl console:base_port</td>
<td>Display the base port of the mongrels.</td>
</tr>
<tr>
<td>Create setup file</td>
<td>asctl console:create_setup_file file</td>
<td>Create a reusable file that contains answers to the setup questions. Replace file with a file name.</td>
</tr>
<tr>
<td>Disable Console</td>
<td>asctl console:disable</td>
<td>Disable Console. When disabled, the service will not start when rebooting computer, does not print reminders or update its configurations.</td>
</tr>
<tr>
<td>Re-generate conf</td>
<td>asctl console:generate_config</td>
<td>Generate Console component's configuration file using the current settings.</td>
</tr>
<tr>
<td>Task</td>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Config info</td>
<td>asctl console:info</td>
<td>Print Console configuration info.</td>
</tr>
<tr>
<td>Update database</td>
<td>asctl console:migrate_database</td>
<td>Update database to the latest schema.</td>
</tr>
<tr>
<td>Display mongrel count</td>
<td>asctl console:mongrel_count</td>
<td>Display the number of mongrels to spawn.</td>
</tr>
<tr>
<td>Change mongrel count</td>
<td>asctl console:mongrel_count arg</td>
<td>Change the number of mongrels to spawn. Replace arg with a number.</td>
</tr>
<tr>
<td>Rake command</td>
<td>asctl console:rake arg</td>
<td>Evoke a rake command.</td>
</tr>
<tr>
<td>Restart Console</td>
<td>asctl console:restart</td>
<td>Restart mongrel web servers and all background processes.</td>
</tr>
<tr>
<td>Restore config and data</td>
<td>asctl console:restore dir</td>
<td>Restore Console database and configuration from a backup directory.</td>
</tr>
<tr>
<td>Restore database</td>
<td>asctl console:restore_database dir</td>
<td>Restore Console database from a backup directory.</td>
</tr>
<tr>
<td>Configure Console component</td>
<td>asctl console:setup</td>
<td>Configure this component.</td>
</tr>
<tr>
<td>Configure Console using saved file</td>
<td>asctl console:setup_from_file file</td>
<td>Run setup using the answers from a file created using the &quot;create_setup_file&quot; command.</td>
</tr>
<tr>
<td>Start Console</td>
<td>asctl console:start</td>
<td>Starts mongrel web servers and all background processes.</td>
</tr>
<tr>
<td>Show Console status</td>
<td>asctl console:status</td>
<td>Display Console status.</td>
</tr>
<tr>
<td>Stop Console</td>
<td>asctl console:stop</td>
<td>Stops mongrel web servers and all background processes.</td>
</tr>
<tr>
<td>Upgrade</td>
<td>asctl console:upgrade</td>
<td>Upgrade Console from a previous version.</td>
</tr>
<tr>
<td>Display namespace</td>
<td>asctl console:uri_namespace</td>
<td>Display Console's URL namespace.</td>
</tr>
<tr>
<td>Change namespace</td>
<td>asctl console:uri_namespace arg</td>
<td>Change Console's URL namespace. Replace arg with the new namespace.</td>
</tr>
<tr>
<td>Show Console's version</td>
<td>asctl console:version</td>
<td>Display the currently set up version.</td>
</tr>
<tr>
<td>Generate email templates</td>
<td>asctl console:generate_email_templates</td>
<td>Recreate email template files.</td>
</tr>
</tbody>
</table>

**MySQL**

<table>
<thead>
<tr>
<th>Task</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create setup file</td>
<td>asctl mysql:create_setup_file file</td>
<td>Create a reusable file that contains answers to the setup questions. Replace file with a file name.</td>
</tr>
<tr>
<td>Display database directory</td>
<td>asctl mysql:data_dir</td>
<td>Display the directory that the databases are kept in.</td>
</tr>
<tr>
<td>Task</td>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>Disable MySQL</td>
<td>asctl mysql:disable</td>
<td>Disable the Aspera MySQL. When disabled, the service will not start when rebooting computer, does not print reminders or update its configurations.</td>
</tr>
<tr>
<td>Grant access on MySQL-only server</td>
<td>asctl mysql:grant_remote_access host mysql_user password</td>
<td>If MySQL server is running on a different computer, use this command on the MySQL machine to allow access from the specified machine. Replace <code>host</code>, <code>mysql_user</code> and <code>mysql_password</code> with the server's hostname, MySQL's user name, and the user's password, respectively.</td>
</tr>
<tr>
<td>Show config info</td>
<td>asctl mysql:info</td>
<td>Print configuration info about MySQL.</td>
</tr>
<tr>
<td>Show port</td>
<td>asctl mysql:port</td>
<td>Display the port the MySQL server listens to.</td>
</tr>
<tr>
<td>Change port</td>
<td>asctl mysql:port port</td>
<td>Change the port the MySQL server listens to. Replace <code>port</code> with a new port number.</td>
</tr>
<tr>
<td>Restart MySQL</td>
<td>asctl mysql:restart</td>
<td>Restart the Aspera MySQL.</td>
</tr>
<tr>
<td>Set root password</td>
<td>asctl mysql:set_root_password</td>
<td>Set the password for 'root' in MySQL.</td>
</tr>
<tr>
<td>Configure MySQL-only server</td>
<td>asctl mysql:setup</td>
<td>If MySQL server is running on a different computer, use this command on the MySQL machine to configure it.</td>
</tr>
<tr>
<td>Configure MySQL using saved file</td>
<td>asctl mysql:setup_from_file file</td>
<td>Run setup using the answers from a file created using the &quot;create_setup_file&quot; command.</td>
</tr>
<tr>
<td>Start MySQL</td>
<td>asctl mysql:start</td>
<td>Start the Aspera MySQL.</td>
</tr>
<tr>
<td>Show MySQL status</td>
<td>asctl mysql:status</td>
<td>Display the Aspera MySQL status.</td>
</tr>
<tr>
<td>Stop MySQL</td>
<td>asctl mysql:stop</td>
<td>Stop the Aspera MySQL.</td>
</tr>
<tr>
<td>Upgrade MySQL-only server</td>
<td>asctl mysql:upgrade</td>
<td>If MySQL server is running on a different computer, use this command on the MySQL machine to upgrade the database.</td>
</tr>
<tr>
<td>Show MySQL's version</td>
<td>asctl mysql:version</td>
<td>Display the currently set up version.</td>
</tr>
</tbody>
</table>

**Advanced Search**
You can search for a transfer from any page in IBM Aspera Console by using the search bar in the top right corner of the page. If you want to refine your search, you can access the Advanced Search dialog by selecting the blue drop-down arrow next to the search bar.

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transfer Name</td>
<td>Include transfers with this name</td>
</tr>
<tr>
<td>Contact</td>
<td>Include transfers initiated by this user.</td>
</tr>
<tr>
<td>SSH User</td>
<td>Include transfers involving this SSH user.</td>
</tr>
<tr>
<td>Session ID</td>
<td>Include transfers with this unique session ID</td>
</tr>
<tr>
<td>File Name Start</td>
<td>Include transfers with files that start with this string.</td>
</tr>
<tr>
<td>Source Path</td>
<td>Include transfers with files that originated from this location.</td>
</tr>
<tr>
<td>Destination Path</td>
<td>Include transfers with files transferred to this location.</td>
</tr>
<tr>
<td>Filter</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>Node</td>
<td>Include transfers involving this selected node or this node IP address.</td>
</tr>
<tr>
<td>From</td>
<td>Include transfers started from this date and onwards.</td>
</tr>
<tr>
<td>To</td>
<td>Include transfers from this date and onwards.</td>
</tr>
</tbody>
</table>
| Status | Include transfers with the current state designated:  
  • Active  
  • Completed  
  • Cancelled  
  • Error   |
| Results| The number of results you want Console to display. |

### Setting Up the Console Environment

**Setup Example #1: Monitoring Transfers with Another Organization**

This example shows how to create a user group that can monitor all transfers between your organization and a partner organization. The configuration in this example uses the following values:

- Partner node address: 10.0.0.0
- Console user: partner-1-staff (non-admin)

1. Add the partner's node as an unmanaged node, and add an endpoint.
   - To do so, go to **Nodes** from the Console menu, click **List Unmanaged Nodes > New Unmanaged Node**. Enter the partner node's information. You can add the partner's node as an unmanaged node without further configuration.

   ![New Unmanaged Node](image)

2. Create a new endpoint with saved login information.
   - The addition of a new node creates an endpoint in Console with a wildcard (*@10.0.0.0). Wildcard endpoints require a user to enter login credentials every time the user uses it for a transfer. To create an endpoint with saved login credentials, go to the **Endpoints** tab and click **Add Endpoint**. Enter the login credentials for a user on the node and click **Create**.
For more information on the actions listed in this step, see *Adding Unmanaged Nodes* on page 27.

3. Create a group with permission to monitors transfers with the partner's node.
   Go to **Groups** from the Console menu and click **New Group**. In the Create New Group page, enter the group's name and description. Click **Create** when finished.

   ![](image)

When the group is created, select **Transfer Path > Add Path**. The following is an example of the Transfer Path settings:
Field | Description
---|---
Endpoint1 | Choose the Partner's node or endpoint.
Direction | Set to/from for inbound and outbound transfers.
Endpoint2 | Choose Any so that users can make transfers with any node.
Group permissions | Check these options:
• View transfers started by others
• Opt-in to email notifications

Click Create.

For more information on the tasks listed in this step, see Creating Console Groups on page 43.

4. Create a Console user and add it to the group.

To create a user account to monitor the transfers, go to Users from the Console menu and click New User. Fill out the form and click Create.
5. Assign the user to the group.
   In the User Maintenance page's **Groups** tab, select the group from the drop menu and click **Add**.

When logging into Console with this user account, you can monitor all the transfers with the partner's node.
For more information on the tasks listed in this step, see *Creating Console Users* on page 44.

**Setup Example #2: Managing Aspera Faspex Transfers**

This example shows how to monitor and control transfers on a node running IBM Aspera Faspex, a file exchange application built upon IBM Aspera Enterprise Server for a centralized transfer solution. The configuration in this example is as follow:

- **Faspex node address**: 10.0.0.0
- **Console user**: faspex-monitor-1

1. Add Faspex as a managed node, and create an endpoint.
   Go to **Nodes** from the Console menu and click **New Managed Node**. Enter the Faspex node information and click **Create**.
Important: If you wish to configure the Faspex node transfer settings using Console, go to the Node Maintenance page, select the Credentials tab, and enter the SSH login.

When the node is added, go to the Endpoints tab and click Add Endpoint. Enter Faspex in the Login field, leave password fields blank. When finished, click Create.

For more information on the tasks listed in this step, see Adding Nodes to Console on page 22.

2. Create a group with the proper permissions.
   
   Go to Groups from the Console menu and click New Group. Enter the group's information and click Create.
When the group is created in the *Group Maintenance* page, go to the **Transfer Paths** tab and click **Add Path**. Use the following settings for this Transfer Path (change the Faspex node address to match yours):

<table>
<thead>
<tr>
<th><strong>Item</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Endpoint1</td>
<td>faspex@22.33.44.55</td>
</tr>
<tr>
<td>Direction</td>
<td>to / from</td>
</tr>
<tr>
<td>Endpoint2</td>
<td>Any</td>
</tr>
<tr>
<td>Options</td>
<td>Check these options:</td>
</tr>
<tr>
<td></td>
<td>• View Transfers started by others</td>
</tr>
<tr>
<td></td>
<td>• Opt-in to email notifications</td>
</tr>
</tbody>
</table>

For more information on the tasks listed in this step, see *Creating Console Groups* on page 43.

3. Add users to the group.

In the *Group Maintenance* page, go to the Members tab, select the Console user from the menu and click **Add**. When added, the Console user can monitor and control transfers on the Faspex node through Console.
For more information on the tasks listed in this step, see *Creating Console Users* on page 44.

**Setup Example #3: Create Groups of Different Permissions**

This example shows how to assign different permissions to different project members. The configuration in this example is as follows:

- **Endpoint1**: project@1.2.3.4
- **Endpoint2**: project@5.6.7.8
- **User1 (Project admin)**: admin1
- **User2 (Project member)**: member1
- **Group - Project admin**: All permissions between Endpoint1 and Endpoint2
- **Group - Project member**: Limited permissions between Endpoint1 and Endpoint2

1. Prepare the nodes, endpoints and the Console user accounts.
   - *Add a managed or unmanaged node with an endpoint for transfer*: project@1.2.3.4
   - *Add a managed or unmanaged node with an endpoint for transfer*: project@5.6.7.8
   - *Add a Console user*: admin1
   - *Add a Console user*: member1

   For more information on the tasks listed in this step, see *Adding Nodes to Console* on page 22, *Adding Unmanaged Nodes* on page 27, and *Creating Console Users* on page 44.

2. Create a group for the project administrator.

   Go to **Groups** from the Console menu and click **New Group**. Enter the group's information and click **Create**.
In the Group Maintenance page, click the Transfer Paths tab. Enter the following information:

### Accounts

| Users | Groups | Directories | Access Log | New Path for 'Project admin' |

- **New Transfer Path**
  - **Endpoint 1**: project@1.2.3.4
  - **Direction**: to/from
  - **Endpoint 2**: project@5.6.7.8
  - **Group permissions**
    - Start Simple Transfers
    - Start Smart Transfers
    - Create Smart Transfers
    - Share Smart Transfers
    - Control Transfers started by others
    - View Transfers started by others
    - Opt-in to email notifications

- **Description**
Click Create.

In the Group Maintenance page, go to the Members tab and add the user admin1 into this group.

For more information on the tasks listed in this step, see Creating Console Groups on page 43.

3. Create a group for the project members.

Go to Groups from the Console menu and click New Group. Enter the group's information and click Create.

In the Group Maintenance page, click the Transfer Paths tab. Enter the following information:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Endpoint1</td>
<td>project@1.2.3.4</td>
</tr>
<tr>
<td>Direction</td>
<td>to / from</td>
</tr>
<tr>
<td>Endpoint2</td>
<td>project@5.6.7.8</td>
</tr>
<tr>
<td>Options</td>
<td>Select All.</td>
</tr>
</tbody>
</table>

Check these options:
- Start Smart Transfers
- View Transfers started by others
- Opt-in to email notifications
When finished, click **Create**.

In the Group Maintenance page, go to the **Members** tab and add the user **member1** into this group.

For more information on the tasks listed in this step, see *Creating Console Groups* on page 43.

4. Create a Smart Transfer template with the project admin account.

   At this point, both Console users should have the proper permissions. Use the project admin account (**admin1**) to create a Smart Transfer template.

   When the user **admin1** creates and share a Smart Transfer template, **member1** will be able to execute the Smart Transfer template.

   For more information on the tasks listed in this step, see *Sharing a Smart Transfer* on page 63

5. Execute the Smart Transfer with the project member account.

   The **Project member** group, which has the same Transfer Path as the **Project admin** group, has access to the shared Smart Transfer templates. Go to **Transfer** from the Console menu, the Project member will see the Smart Transfer template listed in the Saved Smart Transfers table.

---

**Email Template Examples**

**Email Template Example: Creating a Simple Notification for a Successful Transfer**

The following example shows how to create an email template that notifies a user of a successful transfer with minimal information.

1. Select **Create new transfer success email template** and then **edit**.
2. Name your template "Client Success Email".
3. Enter a **From Name** and **Reply-to Address** if you don't want the notification to come from the default email address.
4. Enter a new email subject: "Client Transfer Notification - Success".
5. Click **Edit Plain Template** and make remove variables to limit information provided to the recipient.
For example:

```
Client Transfer Notification

| Description of the Transfer:        | DESCRIPTION             |
| Client Name:                       | CONTACT                 |
| Total Bytes Transferred:           | BYTES TRANSFERRED       |
| Total Time for Transfer:           | ELAPSED_TIME            |
| Average Transfer Rate:             | AVERAGE_RATE            |

You are receiving this message because your Aspera Console preferences are set to receive these notifications or someone else thought you should know about this particular transfer.
```

The end result should look like the following:

```
Client #1 Transfer Notification

| Description of the Transfer:        | TEMPLATE TEST: File from Sydney to LA |
| Client Name:                       | econ1 (ssh)                        |
| Total Bytes Transferred:           | 0 Bytes                             |
| Total Time for Transfer:           | 44s                                 |
| Average Transfer Rate:             |                                    |

You are receiving this message because your Aspera Console preferences are set to receive these notifications or someone else thought you should know about this particular transfer.
```

6. **Edit HTML Template** to match the information in the basic template. The end result should look like the following:

```
Client #1 Transfer Notification

| Description of the Transfer:        | TEMPLATE TEST: File from Sydney to LA |
| Client Name:                       | econ1 (ssh)                        |
| Total Bytes Transferred:           | 0 Bytes                             |
| Total Time for Transfer:           | 44s                                 |
| Average Transfer Rate:             |                                    |

You are receiving this message because your Aspera Console preferences are set to receive these notifications or someone else thought you should know about this particular transfer.
```

7. Click the **Send Test Email** button to test the new email template.

**Email Template Example: Adding Company Branding to Your Template**

The following example shows how to create an email template that shows company branding when opened in HTML format.

1. On the Template preview screen, click the **Edit HTML Template** button to modify the template's HTML code.
2. Locate the URL of your company logo. Your image must be hosted on a server that is accessible to the recipient.
3. Open the HTML Template and insert the following code in the desired location.

```html
<IMG SRC="http://<IMAGE_URL>">
```

In this example, we've inserted the logo into the header.

The result may look like the following:

```
Aspera Console Transfer Notification

Aspera

Description: TEMPLATE TEST: File from Sydney to LA
Started by: econ1 (ssh)

Source: Sydney (10.0.75.201)
First 5 Source Paths: 
   C:/demo files/test
   C:/demo files

Destination: LA (10.0.85.108)
   /home/econ1/uploads
```

## Node References

### Node-Level Configuration Options

To start node configuration, go to **Nodes** in the Console menu. Click **edit** for an existing node that you wish to configure. The server's admin credentials are required for the configuration. See [Updating a Node's Admin Credentials](#) on page 26.

The node configuration options can be found in the Configuration tab. The following is a summarized chart for navigating and changing values when you click on an individual section. Click **Save changes** when finished:

Configuration at the node level will affect all user accounts and group accounts on that node performing Aspera transfers.

<table>
<thead>
<tr>
<th>Section</th>
<th>Configuration Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>Configuring policy and logging level settings.</td>
</tr>
<tr>
<td>Transfer Server</td>
<td>Setting transfer server IP address and port.</td>
</tr>
<tr>
<td>HTTP Fallback Server</td>
<td>Enable and configure HTTP / HTTPS fallback server.</td>
</tr>
<tr>
<td>Docroot</td>
<td>Setting document root and its access permissions.</td>
</tr>
<tr>
<td>Authorization</td>
<td>Connection permissions, token key, and encryption requirements.</td>
</tr>
<tr>
<td>Section</td>
<td>Configuration Details</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>Incoming and outgoing transfer bandwidth and policy settings.</td>
</tr>
<tr>
<td>Advanced File Handling</td>
<td>File handling settings, such as file block size, overwrite rules, and exclude pattern.</td>
</tr>
<tr>
<td>Advanced Network Options</td>
<td>Network IP address, port, and socket buffer settings.</td>
</tr>
</tbody>
</table>

### Database

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Host IP</td>
<td>Enter the Aspera Console server's IP address, default 127.0.0.1</td>
<td>valid IPv4 address</td>
<td>127.0.0.1</td>
</tr>
<tr>
<td>2</td>
<td>Port</td>
<td>The default value for an Aspera Console installation is 4406. Valid port numbers range between 1 and 65535.</td>
<td>Integer between 1 and 65535</td>
<td>4406</td>
</tr>
<tr>
<td>3</td>
<td>User</td>
<td>User login for the database server.</td>
<td>text string</td>
<td>blank</td>
</tr>
<tr>
<td>4</td>
<td>Database Name</td>
<td>Name of the database used to store Aspera transfer data.</td>
<td>text string</td>
<td>blank</td>
</tr>
<tr>
<td>5</td>
<td>Threads</td>
<td>The number of parallel connections used for database logging. A higher value may be useful when a large number of files are being transferred within a given time frame.</td>
<td>Integer between 1 and 40</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>Stop Transfers on Database Error</td>
<td>Quits all ongoing transfers and no new transfers are permitted when a database error prevents data from being written to the database. Set this to true if all transfers must be logged by your organization.</td>
<td>• true</td>
<td>false</td>
</tr>
<tr>
<td>7</td>
<td>Session Progress</td>
<td>Setting this value to <strong>true</strong> will log transfer status such as number of files transferred, and bytes transferred, at a given interval.</td>
<td>• true</td>
<td>true</td>
</tr>
<tr>
<td>8</td>
<td>Session Progress Interval</td>
<td>The frequency at which an Aspera node logs transfer session information, up to 65535 seconds.</td>
<td>Positive integer</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>File Events</td>
<td>Setting this value to <strong>true</strong> enables the logging of complete file paths and file names. Performance may be improved when transferring datasets containing thousands of files. Also see File Per Session for setting a threshold for the number of files to log per session.</td>
<td>• true</td>
<td>true</td>
</tr>
<tr>
<td>10</td>
<td>File Progress</td>
<td>Setting this value to <strong>true</strong> will log file status such as bytes transferred at a given interval.</td>
<td>• true</td>
<td>true</td>
</tr>
<tr>
<td>11</td>
<td>File Progress Interval</td>
<td>The frequency at which an Aspera node logs file transfer information, up to 65535 seconds. The default setting of 1 logging sessions every second.</td>
<td>Integer between 1 and 65535</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>Files Per Session</td>
<td>The value is the cut-off point for file names logged in a given session. For example, if the value is set to 50, the first 50 file names will be</td>
<td>Positive integer or zero</td>
<td>0</td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td>----</td>
<td>--------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------</td>
<td>---------</td>
</tr>
<tr>
<td>13</td>
<td>Ignore Empty Files</td>
<td>Setting this to <strong>true</strong> will block the logging of zero-byte files.</td>
<td>• true</td>
<td>false</td>
</tr>
<tr>
<td>14</td>
<td>Ignore No-transfer Files</td>
<td>Setting this to <strong>true</strong> will block the logging of files that have not been transferred because they exist at the destination at the time the transfer started.</td>
<td>• true</td>
<td>false</td>
</tr>
<tr>
<td>15</td>
<td>Rate Events</td>
<td>Setting this to <strong>true</strong> will log changes made to the Target Rate, Minimum Rate, and Transfer Policy of a transfer by any user or Aspera node administrator during a transfer.</td>
<td>• true</td>
<td>true</td>
</tr>
</tbody>
</table>

**Transfer Server**

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bind Address</td>
<td>This is the network interface address on which the transfer server listens. The default value <strong>127.0.0.1</strong> enables the transfer server to accept transfer requests from the local computer. Setting the value to <strong>0.0.0.0</strong> allows the Aspera transfer server to accept transfer requests on all network interfaces for this node. Alternatively, a specific network interface address may be specified.</td>
<td>Valid IPv4 address</td>
<td>127.0.0.1</td>
</tr>
<tr>
<td>2</td>
<td>Bind Port</td>
<td>The port at which the transfer server will accept transfer requests.</td>
<td>Integer between 1 and 65535</td>
<td>40001</td>
</tr>
</tbody>
</table>

**HTTP Fallback Server**

*Note:* While Console can change a node's settings for HTTP fallback, Console does not support HTTP fallback for transfers it initiates.

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cert File</td>
<td>The absolute path to an SSL certificate file. If left blank, the default certificate file that came with your Aspera Enterprise Server will be used.</td>
<td>file path</td>
<td>blank</td>
</tr>
<tr>
<td>2</td>
<td>Key File</td>
<td>The absolute path to an SSL key file. If left blank, the default certificate file that came with your Aspera Enterprise Server will be used.</td>
<td>file path</td>
<td>blank</td>
</tr>
<tr>
<td>3</td>
<td>Bind Address</td>
<td>This is the network interface address on which the HTTP Fallback Server listens. The default value <strong>0.0.0.0</strong> allows the Aspera HTTP Fallback Server to accept transfer requests on all network interfaces for this node.</td>
<td>valid IPv4 address</td>
<td>0.0.0.0</td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td>----</td>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------</td>
<td>---------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>interfaces for this node. Alternatively, a specific network interface address may be specified.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Restartable Transfers</td>
<td>Setting this to <code>true</code> allows interrupted transfers to resume at the point of interruption.</td>
<td><code>true</code></td>
<td><code>true</code></td>
</tr>
<tr>
<td>5</td>
<td>Session Activity Timeout</td>
<td>Any value greater than 0 sets the amount of time, in seconds, that the HTTP Fallback Server will wait without any transfer activity before canceling the transfer. Notice that this option cannot be left at 0, otherwise interrupted HTTP Fallback sessions will get stuck until server or asperacentral is restarted.</td>
<td>Positive integer</td>
<td><code>0</code></td>
</tr>
<tr>
<td>6</td>
<td>Enable HTTP</td>
<td>Enables the HTTP Fallback Server that allows failed UDP transfers to continue over HTTP.</td>
<td><code>true</code></td>
<td><code>false</code></td>
</tr>
<tr>
<td>7</td>
<td>HTTP Port</td>
<td>The port on which the HTTP server listens. Valid port numbers range between 1 and 65535.</td>
<td>positive integer</td>
<td><code>8080</code></td>
</tr>
<tr>
<td>8</td>
<td>Enable HTTPS</td>
<td>Enables the HTTPS Fallback Server that allows failed UDP transfers to continue over HTTPS.</td>
<td><code>true</code></td>
<td><code>false</code></td>
</tr>
<tr>
<td>9</td>
<td>HTTPS Port</td>
<td>The port on which the HTTPS server listens. Valid port numbers range between 1 and 65535.</td>
<td>positive integer</td>
<td><code>8443</code></td>
</tr>
</tbody>
</table>

**Docroot**

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Absolute Path</td>
<td>The Absolute Path describes the area of the file system that is accessible by Aspera users. The default empty value gives users access to the entire file system.</td>
<td>file path</td>
<td>N/A</td>
</tr>
<tr>
<td>2</td>
<td>Read Allowed</td>
<td>Setting this to true allows users to transfer from the designated area of the file system as specified by the Absolute Path value.</td>
<td><code>true</code></td>
<td>N/A</td>
</tr>
<tr>
<td>3</td>
<td>Write Allowed</td>
<td>Setting this to true allows users to transfer to the designated area of the file system as specified by the Absolute Path value.</td>
<td><code>true</code></td>
<td>N/A</td>
</tr>
<tr>
<td>4</td>
<td>Browse Allowed</td>
<td>Setting this to true allows users to browse the directory.</td>
<td><code>true</code></td>
<td>N/A</td>
</tr>
</tbody>
</table>

**Authorization**

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Incoming Transfers</td>
<td>The default setting of <code>allow</code> allows users to transfer to this computer. Setting this to <code>deny</code> will prevent transfers to this computer. When set to <code>require token</code>, only transfers initiated with valid tokens will be allowed to transfer to this computer. Token-based</td>
<td><code>allow</code></td>
<td><code>allow</code></td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td>----</td>
<td>--------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>----------</td>
</tr>
<tr>
<td>1</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td></td>
<td></td>
<td>babbiagb a web applications such as Faspex and require a Token Encryption Key.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Incoming External Provider URL</td>
<td>The value entered should be the URL of the external authorization provider for incoming transfers. The default empty setting disables external authorization. Aspera servers can be configured to check with an external authorization provider. This SOAP authorization mechanism can be useful to organizations requiring custom authorization rules.</td>
<td>HTTP URL</td>
<td>blank</td>
</tr>
<tr>
<td>3</td>
<td>Incoming External Provider SOAP Action</td>
<td>The SOAP action required by the external authorization provider for incoming transfers. Required if External Authorization is enabled.</td>
<td>text string</td>
<td>blank</td>
</tr>
<tr>
<td>4</td>
<td>Outgoing Transfers</td>
<td>The default setting of <strong>allow</strong> allows users to transfer from this computer. Setting this to <strong>deny</strong> will prevent transfers from this computer. When set to <strong>require token</strong>, only transfers initiated with valid tokens will be allowed to transfer from this computer. Token-based transfers are typically employed by web applications such as Faspex and require a Token Encryption Key.</td>
<td>• allow</td>
<td>allow</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• deny</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• require token</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Outgoing External Provider URL</td>
<td>The value entered should be the URL of the external authorization provider for outgoing transfers. The default empty setting disables external authorization. Aspera servers can be configured to check with an external authorization provider. This SOAP authorization mechanism can be useful to organizations requiring custom authorization rules.</td>
<td>HTTP URL, default blank</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Outgoing External Provider Soap Action</td>
<td>The SOAP action required by the external authorization provider for outgoing transfers. Required if External Authorization is enabled.</td>
<td>Text string</td>
<td>blank</td>
</tr>
<tr>
<td>7</td>
<td>Token Encryption Cipher</td>
<td>The cipher used to generate encrypted authorization tokens.</td>
<td>• aes-128</td>
<td>aes-128</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• aes-192</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• aes-256</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Token Encryption Key</td>
<td>This is the secret token that will be used to authorize those transfers configured to require token. Token generation is part of the Aspera SDK. See the Aspera Developer's Network (Token-based Authorization Topic) for more information.</td>
<td>Text string</td>
<td>blank</td>
</tr>
<tr>
<td>9</td>
<td>Token Life (seconds)</td>
<td>Sets token expiration for users of web-based transfer applications.</td>
<td>Positive integer</td>
<td>1200</td>
</tr>
<tr>
<td>10</td>
<td>Encryption Allowed</td>
<td>Describes the type of transfer encryption accepted by this computer. When set to <strong>any</strong> the computer allows both encrypted and non-encrypted transfers. When set to none the computer restricts transfers to non-encrypted transfers only. When set to aes-128 the computer restricts transfers to encrypted transfers only.</td>
<td>• any</td>
<td>any</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• none</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• aes-128</td>
<td></td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td>---</td>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>---------</td>
</tr>
<tr>
<td>1</td>
<td>Incoming Vlink ID</td>
<td>The value sets the Vlink ID for incoming transfers. Vlinks are a mechanism to define aggregate transfer policies. The default setting of 0 disables Vlinks. One Vlink—the virtual equivalent of a network trunk—represents a bandwidth allowance that may be allocated to a node, group, or user. Vlink ID are defined in each Vlink created in Aspera Console. The Vlink ID is a unique numeric identifier. See Configuring Virtual Links on page 33.</td>
<td>Pre-defined value</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Incoming Target Rate Cap (Kbps)</td>
<td>The value sets the Target Rate Cap for incoming transfers. The Target Rate Cap is the maximum target rate that a transfer can request, in kilobits per second. No transfer may be adjusted above this setting, at any time. The default setting of Unlimited signifies no Target Rate Cap. Clients requesting transfers with initial rates above the Target Rate Cap will be denied.</td>
<td>Positive integer</td>
<td>Unlimited</td>
</tr>
<tr>
<td>3</td>
<td>Incoming Target Rate Default (Kbps)</td>
<td>This value represents the initial rate for incoming transfers, in kilobits per second. Users may be able to modify this rate in real time as allowed by the software in use. This setting is not relevant to transfers with a Fixed policy.</td>
<td>Positive integer</td>
<td>10000</td>
</tr>
<tr>
<td>4</td>
<td>Incoming Target Rate Lock</td>
<td>After an incoming transfer is started, its target rate may be modified in real time. The default setting false gives users the ability to adjust the transfer rate. A setting of true prevents real-time modification of the transfer rate.</td>
<td>• true</td>
<td>false</td>
</tr>
<tr>
<td>5</td>
<td>Incoming Minimum Rate Cap (Kbps)</td>
<td>The value sets the Minimum Rate Cap for incoming transfers. The Minimum Rate Cap is a level specified in kilobits per second, below which an incoming transfer will not slow, despite network congestion or physical network availability. The default value of Unlimited effectively turns off the Minimum Rate Cap.</td>
<td>Positive integer</td>
<td>Unlimited</td>
</tr>
<tr>
<td>6</td>
<td>Incoming Minimum Rate Default (Kbps)</td>
<td>This value represents the initial minimum rate for incoming transfers, in kilobits per second. Users may be able to modify this rate in real time as allowed by the software in use. This setting is not relevant to transfers with a Fixed policy.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>Incoming Minimum Rate Lock</td>
<td>After an incoming transfer is started, its minimum rate may be modified in real time. The default setting of false gives users the ability to adjust the transfer's minimum rate. A setting of true prevents real-time modification of the transfer rate. This setting is not relevant to transfers with a Fixed policy.</td>
<td>• true</td>
<td>false</td>
</tr>
<tr>
<td>8</td>
<td>Incoming Bandwidth Policy Default</td>
<td>The value chosen sets the default Bandwidth Policy for incoming transfers. The default policy value is fair.</td>
<td>• fixed</td>
<td>fair</td>
</tr>
</tbody>
</table>

**Bandwidth**
<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>may be overridden by client applications initiating transfers.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Incoming Bandwidth Policy Allowed</td>
<td>The value chosen sets the allowed Bandwidth Policy for incoming transfers. Aspera transfers use fixed, high, fair and low policies to accommodate network-sharing requirements. When set to <strong>any</strong>, the server will not deny any transfer based on policy setting. When set to <strong>high</strong>, transfers with a Policy of high and less aggressive transfer policies (such as, fair or low) will be permitted. Fixed transfers will be denied. When set to low, only transfers with a Bandwidth Policy of <strong>low</strong> will be allowed.</td>
<td>• fixed • high • fair • low</td>
<td>fair</td>
</tr>
<tr>
<td>10</td>
<td>Incoming Bandwidth Policy Lock</td>
<td>After an incoming transfer is started, its Policy may be modified in real time. The default setting of <strong>false</strong> gives users the ability to adjust the transfer's Policy. A setting of <strong>true</strong> prevents real-time modification of the Policy.</td>
<td>• true • false</td>
<td>false</td>
</tr>
<tr>
<td>11</td>
<td>Outgoing Vlink ID</td>
<td>The value sets the Vlink ID for outgoing transfers. Vlinks are a mechanism to define aggregate transfer policies. The default setting of 0 disables Vlinks. One Vlink—the virtual equivalent of a network trunk—represents a bandwidth allowance that may be allocated to a node, group, or user. Vlink ID are defined in each Vlink created in Aspera Console. The Vlink ID is a unique numeric identifier. See <a href="#">Configuring Virtual Links</a> on page 33.</td>
<td>Pre-defined value</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>Outgoing Target Rate Cap (Kbps)</td>
<td>The value sets the Target Rate Cap for outgoing transfers. The Target Rate Cap is the maximum target rate that a transfer can request, in kilobits per second. No transfer may be adjusted above this setting, at any time. The default setting of Unlimited signifies no Target Rate Cap. Clients requesting transfers with initial rates above the Target Rate Cap will be denied.</td>
<td>Positive integer</td>
<td>Unlimited</td>
</tr>
<tr>
<td>13</td>
<td>Outgoing Target Rate Default (Kbps)</td>
<td>This value represents the initial rate for outgoing transfers, in kilobits per second. Users may be able to modify this rate in real time as allowed by the software in use. This setting is not relevant to transfers with a Fixed policy.</td>
<td>Positive integer</td>
<td>10000</td>
</tr>
<tr>
<td>14</td>
<td>Outgoing Target Rate Lock</td>
<td>After an outgoing transfer is started, its target rate may be modified in real time. The default setting of <strong>false</strong> gives users the ability to adjust the transfer rate. A setting of <strong>true</strong> prevents real-time modification of the transfer rate.</td>
<td>• true • false</td>
<td>false</td>
</tr>
<tr>
<td>15</td>
<td>Outgoing Minimum Rate Cap (Kbps)</td>
<td>The value sets the Minimum Rate Cap for outgoing transfers. The Minimum Rate Cap is a level specified in kilobits per second, below which an incoming transfer will not slow, despite network congestion or physical network availability. The values may be overridden by client applications initiating transfers.</td>
<td>Positive integer</td>
<td>Unlimited</td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td>----</td>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------</td>
<td>---------</td>
</tr>
<tr>
<td>16</td>
<td>Outgoing Minimum Rate Default</td>
<td>This value represents the initial minimum rate for outgoing transfers, in kilobits per second. Users may be able to modify this rate in real time as allowed by the software in use. This setting is not relevant to transfers with a Fixed policy.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
</tbody>
</table>
| 17 | Outgoing Minimum Rate Lock    | After an outgoing transfer is started, its minimum rate may be modified in real time. The default setting of false gives users the ability to adjust the transfer's minimum rate. A setting of true prevents real-time modification of the transfer rate. This setting is not relevant to transfers with a Fixed policy. | • true  
• false | false |
| 18 | Outgoing Bandwidth Policy Default | The value chosen sets the default Bandwidth Policy for outgoing transfers. The default policy value may be overridden by client applications initiating transfers. | • fixed  
• high  
• fair  
• low | fair |
| 19 | Outgoing Bandwidth Policy Allowed | The value chosen sets the allowed Bandwidth Policy for outgoing transfers. Aspera transfers use fixed, high, fair and low policies to accommodate network-sharing requirements. When set to any, the server will not deny any transfer based on policy setting. When set to high, transfers with a Policy of high and less aggressive transfer policies (for example, fair or low) will be permitted. Fixed transfers will be denied. When set to low, only transfers with a Bandwidth Policy of low will be allowed. | • any  
• high  
• fair  
• low | any |
| 20 | Outgoing Bandwidth Policy Lock | After an outgoing transfer is started, its Policy may be modified in real time. The default setting of false gives users the ability to adjust the transfer's Policy. A setting of true prevents real-time modification of the Policy. | • true  
• false | false |

### Advanced File Handling

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>File Create Mode</td>
<td>Specify file creation mode (permissions). If specified, create files with these permissions (for example 0755), irrespective of File Create Grant Mask and permissions of the file on the source computer. Only takes effect when the server is a non-Windows receiver.</td>
<td>Positive integer (octal)</td>
<td>undefined</td>
</tr>
<tr>
<td>2</td>
<td>File Create Grant Mask</td>
<td>Used to determine mode for newly created files if File Create Mode is not specified. If specified, file modes will be set to their original modes plus the Grant Mask values. Only takes effect when the server is a non-Windows receiver and when File Create Mode is not specified.</td>
<td>Positive integer (octal)</td>
<td>0644</td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td>---</td>
<td>--------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------</td>
<td>----------</td>
</tr>
<tr>
<td>3</td>
<td>Directory Create Mode</td>
<td>Specify directory creation mode (permissions). If specified, create directories with these permissions irrespective of Directory Create Grant Mask and permissions of the directory on the source computer. Only takes effect when the server is a non-Windows receiver.</td>
<td>Positive integer (octal)</td>
<td>undefined</td>
</tr>
<tr>
<td>4</td>
<td>Directory Create Grant Mask</td>
<td>Used to determine mode for newly created directories if Directory Create Mode is not specified. If specified, directory modes will be set to their original modes plus the Grant Mask values. Only takes effect when the server is a non-Windows receiver and when Directory Create Mode is not specified.</td>
<td>Positive integer (octal)</td>
<td>0755</td>
</tr>
<tr>
<td>5</td>
<td>Read Block Size (bytes)</td>
<td>This is a performance tuning parameter for an Aspera sender. It represents the number of bytes an Aspera sender reads at a time from the source disk drive. Only takes effect when server is sender. The default of 0 will cause the Aspera sender to use its default internal buffer size, which may be different for different operating systems.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>Write Block Size (bytes)</td>
<td>This is a performance tuning parameter for an Aspera receiver. Number of bytes an ascp receiver writes data at a time onto disk drive. Only takes effect when server is receiver. The default of 0 will cause the Aspera sender to use its default internal buffer size, which may be different for different operating systems.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>Use File Cache</td>
<td>This is a performance tuning parameter for an Aspera receiver. Enable or disable per-file memory caching at the data receiver. File level memory caching improves data write speed on Windows platforms in particular, but will use more memory. We suggest using a file cache on systems that are transferring data at speeds close to the performance of their storage device, and disable it for systems with very high concurrency (because memory utilization will grow with the number of concurrent transfers).</td>
<td>• true</td>
<td>true</td>
</tr>
<tr>
<td>8</td>
<td>Max File Cache Buffer (bytes)</td>
<td>This is a performance tuning parameter for an Aspera receiver. This value corresponds to the maximal size allocated for per-file memory cache (see Use File Cache). Unit is bytes. The default of 0 will cause the Aspera receiver to use its internal buffer size, which may be different for different operating systems.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>Resume Suffix</td>
<td>Extension name of a class of special files holding metadata information of regular data files. Useful in the context of resuming partially completed transfers. During resume mode (-k1/2/3), each data</td>
<td>text string</td>
<td>aspx</td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td>----</td>
<td>---------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------</td>
<td>---------</td>
</tr>
<tr>
<td>10</td>
<td>Preserve Attributes</td>
<td>Configure file creation policy. When set to none, do not preserve the timestamp of source files. When set to times, preserve the timestamp of the source files at destination.</td>
<td>none / times</td>
<td>undefined</td>
</tr>
</tbody>
</table>
| 11 | Overwrite           | Overwrite is an Aspera server setting that determines whether Aspera clients are allowed to overwrite files on the server. By default it is set to allow, meaning that clients uploading files to the servers will be allowed to overwrite existing files as long as file permissions allow that action. If set to deny, clients uploading files to the server will not be able to overwrite existing files, regardless of file permissions. | • allow  
• deny   | allow   |
| 12 | File Manifest       | When set to text a text file "receipt" of all files within each transfer session is generated. If set to disable no File Manifest is created. The file manifest is a file containing a list of everything that was transferred in a given transfer session. The filename of the File Manifest itself is automatically generated based on the transfer session's unique ID. The location where each manifest is written is specified by the File Manifest Path value. If no File Manifest Path is specified, the file will be generated under the destination path at the receiver, and under the first source path at the sender. | • text  
• disable | none    |
| 13 | File Manifest Path  | Specify the location to store manifest files. Can be an absolute path or a path relative to the transfer user's home.                                                                                         | text string | blank   |
| 14 | Pre-Calculate Job Size | Configure the policy of calculating total job size before data transfer. If set to any, follow client configurations (-o PreCalculateJobSize={yes|no}). If set to no, disable calculating job size before transferring. If set to yes, enable calculating job size before transferring. | • any  
• yes  
• no   | any     |
| 15 | Storage Rate Control | Enable/Disable disk rate control. When enabled, adjust transfer rate according to the speed of receiving I/O storage, if it becomes a bottleneck.                                                           | • true  
• false | false   |
| 16 | File checksum method | Specify the type of checksum to calculate for transferred files. The content of transfers can be verified by comparing the checksum value at the destination with the value read at the source.                                                        | • any  
• md5  
• sha1 | any     |
| 16 | Partial Suffix      | Set the file suffix for partially downloaded files.                                                                                                                                                       | .aspx      |         |
### Advanced Network Options

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bind IP Address</td>
<td>Specify an IP address for server-side ascp to bind its UDP connection. If a valid IP address is given, ascp sends and receives UDP packets ONLY on the interface corresponding to that IP address.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Values</td>
</tr>
<tr>
<td>2</td>
<td>Bind UDP Port</td>
<td>Specify a port number for server-side ascp to bind its UDP connection. This also prevents client ascp processes from binding to same UDP port. Valid port numbers range between 1 and 65535.</td>
</tr>
<tr>
<td>3</td>
<td>Disable Packet Batching</td>
<td>When set to <strong>true</strong>, send data packets back to back (no sending a batch of packets). This results in smoother data traffic at a cost of higher CPU usage.</td>
</tr>
<tr>
<td>4</td>
<td>Maximum Socket Buffer (bytes)</td>
<td>Upper bound the UDP socket buffer of an ascp session below the input value. The default of 0 will cause the Aspera sender to use its default internal buffer size, which may be different for different operating systems.</td>
</tr>
<tr>
<td>5</td>
<td>Minimum socket buffer (bytes)</td>
<td>Set the minimum UDP socket buffer size for an ascp session.</td>
</tr>
</tbody>
</table>

### Node Account-Level Configuration Options

When configuring users and groups on a node from Console, both group-level and user-level settings share the same configuration options. This topic covers the following configuration sections:

<table>
<thead>
<tr>
<th>Section</th>
<th>Configuration Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Docroot</td>
<td>Setting document root and its access permissions.</td>
</tr>
<tr>
<td>Authorization</td>
<td>Connection permissions, token key, and encryption requirements.</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>Incoming and outgoing transfer bandwidth and policy settings.</td>
</tr>
<tr>
<td>Advanced File Handling</td>
<td>File handling settings, such as file block size, overwrite rules, and exclude pattern.</td>
</tr>
<tr>
<td>Advanced Network Options</td>
<td>Network IP, port, and socket buffer settings.</td>
</tr>
</tbody>
</table>

### Docroot

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Absolute Path</td>
<td>The Absolute Path describes the area of the file system that is accessible by Aspera users. The default empty value gives users access to the entire file system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Values</td>
</tr>
<tr>
<td>2</td>
<td>Read Allowed</td>
<td>Setting this to true allows users to transfer from the designated area of the file system as specified by the Absolute Path value.</td>
</tr>
<tr>
<td>3</td>
<td>Write Allowed</td>
<td>Setting this to true allows users to transfer to the designated area of the file system as specified by the Absolute Path value.</td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---</td>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>4</td>
<td>Browse Allowed</td>
<td>Setting this to true allows users to browse the directory.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Authorization**

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Incoming Transfers</td>
<td>The default setting of <strong>allow</strong> allows users to transfer to this computer. Setting this to <strong>deny</strong> will prevent transfers to this computer. When set to <strong>require token</strong>, only transfers initiated with valid tokens will be allowed to transfer to this computer. Token-based transfers are typically employed by web applications such as Faspex and require a Token Encryption Key.</td>
<td>• allow</td>
<td>allow</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• deny</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• require token</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Incoming External Provider URL</td>
<td>The value entered should be the URL of the external authorization provider for incoming transfers. The default empty setting disables external authorization. Aspera servers can be configured to check with an external authorization provider. This SOAP authorization mechanism can be useful to organizations requiring custom authorization rules.</td>
<td>HTTP URL</td>
<td>blank</td>
</tr>
<tr>
<td>3</td>
<td>Incoming External Provider SOAP Action</td>
<td>The SOAP action required by the external authorization provider for incoming transfers. Required if External Authorization is enabled.</td>
<td>text string</td>
<td>blank</td>
</tr>
<tr>
<td>4</td>
<td>Outgoing Transfers</td>
<td>The default setting of <strong>allow</strong> allows users to transfer from this computer. Setting this to <strong>deny</strong> will prevent transfers from this computer. When set to <strong>require token</strong>, only transfers initiated with valid tokens will be allowed to transfer from this computer. Token-based transfers are typically employed by web applications such as Faspex and require a Token Encryption Key.</td>
<td>• allow</td>
<td>allow</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• deny</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• require token</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Outgoing External Provider URL</td>
<td>The value entered should be the URL of the external authorization provider for outgoing transfers. The default empty setting disables external authorization. Aspera servers can be configured to check with an external authorization provider. This SOAP authorization mechanism can be useful to organizations requiring custom authorization rules.</td>
<td>HTTP URL, default blank</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Outgoing External Provider Soap Action</td>
<td>The SOAP action required by the external authorization provider for outgoing transfers. Required if External Authorization is enabled.</td>
<td>Text string</td>
<td>blank</td>
</tr>
<tr>
<td>7</td>
<td>Token Encryption Cipher</td>
<td>The cipher used to generate encrypted authorization tokens.</td>
<td>• aes-128</td>
<td>aes-128</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• aes-192</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• aes-256</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Token Encryption Key</td>
<td>This is the secret token that will be used to authorize those transfers configured to require token. Token generation is part of the Aspera SDK. See</td>
<td>Text string</td>
<td>blank</td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td>----</td>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
<td>------------</td>
</tr>
<tr>
<td>9</td>
<td>Token Life (seconds)</td>
<td>Sets token expiration for users of web-based transfer applications.</td>
<td>Positive integer</td>
<td>1200</td>
</tr>
<tr>
<td>10</td>
<td>Encryption Allowed</td>
<td>Describes the type of transfer encryption accepted by this computer. When set to any the computer allows both encrypted and non-encrypted transfers. When set to none the computer restricts transfers to non-encrypted transfers only. When set to aes-128 the computer restricts transfers to encrypted transfers only.</td>
<td>• any</td>
<td>any</td>
</tr>
</tbody>
</table>

**Bandwidth**

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Incoming Vlink ID</td>
<td>The value sets the Vlink ID for incoming transfers. Vlinks are a mechanism to define aggregate transfer policies. The default setting of 0 disables Vlinks. One Vlink—the virtual equivalent of a network trunk—represents a bandwidth allowance that may be allocated to a node, group, or user. Vlink ID are defined in each Vlink created in Aspera Console. The Vlink ID is a unique numeric identifier. See Configuring Virtual Links on page 33</td>
<td>Pre-defined value</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Incoming Target Rate Cap (Kbps)</td>
<td>The value sets the Target Rate Cap for incoming transfers. The Target Rate Cap is the maximum target rate that a transfer can request, in kilobits per second. No transfer may be adjusted above this setting, at any time. The default setting of Unlimited signifies no Target Rate Cap. Clients requesting transfers with initial rates above the Target Rate Cap will be denied.</td>
<td>Positive integer</td>
<td>Unlimited</td>
</tr>
<tr>
<td>3</td>
<td>Incoming Target Rate Default (Kbps)</td>
<td>This value represents the initial rate for incoming transfers, in kilobits per second. Users may be able to modify this rate in real time as allowed by the software in use. This setting is not relevant to transfers with a Fixed policy.</td>
<td>Positive integer</td>
<td>10000</td>
</tr>
<tr>
<td>4</td>
<td>Incoming Target Rate Lock</td>
<td>After an incoming transfer is started, its target rate may be modified in real time. The default setting false gives users the ability to adjust the transfer rate. A setting of true prevents real-time modification of the transfer rate.</td>
<td>• true</td>
<td>false</td>
</tr>
<tr>
<td>5</td>
<td>Incoming Minimum Rate Cap (Kbps)</td>
<td>The value sets the Minimum Rate Cap for incoming transfers. The Minimum Rate Cap is a level specified in kilobits per second, below which an incoming transfer will not slow, despite network congestion or physical network availability. The default value of Unlimited effectively turns off the Minimum Rate Cap.</td>
<td>Positive integer</td>
<td>Unlimited</td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td>----</td>
<td>-------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------</td>
<td>----------</td>
</tr>
<tr>
<td>6</td>
<td>Incoming Minimum Rate Default (Kbps)</td>
<td>This value represents the initial minimum rate for incoming transfers, in kilobits per second. Users may be able to modify this rate in real time as allowed by the software in use. This setting is not relevant to transfers with a <strong>Fixed</strong> policy.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
</tbody>
</table>
| 7  | Incoming Minimum Rate Lock                 | After an incoming transfer is started, its minimum rate may be modified in real time. The default setting of **false** gives users the ability to adjust the transfer's minimum rate. A setting of **true** prevents real-time modification of the transfer rate. This setting is not relevant to transfers with a **Fixed** policy. | • true  
• false | false    |
| 8  | Incoming Bandwidth Policy Default          | The value chosen sets the default Bandwidth Policy for incoming transfers. The default policy value may be overridden by client applications initiating transfers. | • fixed  
• high  
• fair  
• low | fair     |
| 9  | Incoming Bandwidth Policy Allowed          | The value chosen sets the allowed Bandwidth Policy for incoming transfers. Aspera transfers use fixed, high, fair and low policies to accommodate network-sharing requirements. When set to **any**, the server will not deny any transfer based on policy setting. When set to **high**, transfers with a Policy of high and less aggressive transfer policies (such as, fair or low) will be permitted. Fixed transfers will be denied. When set to low, only transfers with a Bandwidth Policy of **low** will be allowed. | • fixed  
• high  
• fair  
• low | fair     |
| 10 | Incoming Bandwidth Policy Lock             | After an incoming transfer is started, its Policy may be modified in real time. The default setting of **false** gives users the ability to adjust the transfer's Policy. A setting of **true** prevents real-time modification of the Policy. | • true  
• false | false    |
<p>| 11 | Outgoing Vlink ID                          | The value sets the Vlink ID for outgoing transfers. Vlinks are a mechanism to define aggregate transfer policies. The default setting of 0 disables Vlinks. One Vlink—the virtual equivalent of a network trunk—represents a bandwidth allowance that may be allocated to a node, group, or user. Vlink ID are defined in each Vlink created in Aspera Console. The Vlink ID is a unique numeric identifier. See <em>Configuring Virtual Links</em> on page 33 | Pre-defined value | 0        |
| 12 | Outgoing Target Rate Cap (Kbps)            | The value sets the Target Rate Cap for outgoing transfers. The Target Rate Cap is the maximum target rate that a transfer can request, in kilobits per second. No transfer may be adjusted above this setting, at any time. The default setting of Unlimited signifies no Target Rate Cap. Clients requesting transfers with initial rates above the Target Rate Cap will be denied. | Positive integer | Unlimited|</p>
<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>Outgoing Target Rate Default (Kbps)</td>
<td>This value represents the initial rate for outgoing transfers, in kilobits per second. Users may be able to modify this rate in real time as allowed by the software in use. This setting is not relevant to transfers with a Fixed policy.</td>
<td>Positive integer</td>
<td>10000</td>
</tr>
<tr>
<td>14</td>
<td>Outgoing Target Rate Lock</td>
<td>After an outgoing transfer is started, its target rate may be modified in real time. The default setting of false gives users the ability to adjust the transfer rate. A setting of true prevents real-time modification of the transfer rate.</td>
<td>• true</td>
<td>false</td>
</tr>
<tr>
<td>15</td>
<td>Outgoing Minimum Rate Cap (Kbps)</td>
<td>The value sets the Minimum Rate Cap for outgoing transfers. The Minimum Rate Cap is a level specified in kilobits per second, below which an incoming transfer will not slow, despite network congestion or physical network availability. The default value of Unlimited effectively turns off the Minimum Rate Cap.</td>
<td>Positive integer</td>
<td>Unlimited</td>
</tr>
<tr>
<td>16</td>
<td>Outgoing Minimum Rate Default</td>
<td>This value represents the initial minimum rate for outgoing transfers, in kilobits per second. Users may be able to modify this rate in real time as allowed by the software in use. This setting is not relevant to transfers with a Fixed policy.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
<tr>
<td>17</td>
<td>Outgoing Minimum Rate Lock</td>
<td>After an outgoing transfer is started, its minimum rate may be modified in real time. The default setting of false gives users the ability to adjust the transfer's minimum rate. A setting of true prevents real-time modification of the transfer rate. This setting is not relevant to transfers with a Fixed policy.</td>
<td>• true</td>
<td>false</td>
</tr>
<tr>
<td>18</td>
<td>Outgoing Bandwidth Policy Default</td>
<td>The value chosen sets the default Bandwidth Policy for outgoing transfers. The default policy value may be overridden by client applications initiating transfers.</td>
<td>• fixed</td>
<td>fair</td>
</tr>
<tr>
<td>19</td>
<td>Outgoing Bandwidth Policy Allowed</td>
<td>The value chosen sets the allowed Bandwidth Policy for outgoing transfers. Aspera transfers use fixed, high, fair and low policies to accommodate network-sharing requirements. When set to any, the server will not deny any transfer based on policy setting. When set to high, transfers with a Policy of high and less aggressive transfer policies (for example, fair or low) will be permitted. Fixed transfers will be denied. When set to low, only transfers with a Bandwidth Policy of low will be allowed.</td>
<td>• any</td>
<td>any</td>
</tr>
<tr>
<td>20</td>
<td>Outgoing Bandwidth Policy Lock</td>
<td>After an outgoing transfer is started, its Policy may be modified in real time. The default setting of false gives users the ability to adjust the transfer's Policy. A setting of true prevents real-time modification of the Policy.</td>
<td>• true</td>
<td>false</td>
</tr>
</tbody>
</table>
### Advanced File Handling

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>File Create Mode</td>
<td>Specify file creation mode (permissions). If specified, create files with these permissions (for example 0755), irrespective of File Create Grant Mask and permissions of the file on the source computer. Only takes effect when the server is a non-Windows receiver.</td>
<td>Positive integer (octal)</td>
<td>undefined</td>
</tr>
<tr>
<td>2</td>
<td>File Create Grant Mask</td>
<td>Used to determine mode for newly created files if File Create Mode is not specified. If specified, file modes will be set to their original modes plus the Grant Mask values. Only takes effect when the server is a non-Windows receiver and when File Create Mode is not specified.</td>
<td>Positive integer (octal)</td>
<td>0644</td>
</tr>
<tr>
<td>3</td>
<td>Directory Create Mode</td>
<td>Specify directory creation mode (permissions). If specified, create directories with these permissions irrespective of Directory Create Grant Mask and permissions of the directory on the source computer. Only takes effect when the server is a non-Windows receiver.</td>
<td>Positive integer (octal)</td>
<td>undefined</td>
</tr>
<tr>
<td>4</td>
<td>Directory Create Grant Mask</td>
<td>Used to determine mode for newly created directories if Directory Create Mode is not specified. If specified, directory modes will be set to their original modes plus the Grant Mask values. Only takes effect when the server is a non-Windows receiver and when Directory Create Mode is not specified.</td>
<td>Positive integer (octal)</td>
<td>0755</td>
</tr>
<tr>
<td>5</td>
<td>Read Block Size (bytes)</td>
<td>This is a performance tuning parameter for an Aspera sender. It represents the number of bytes an Aspera sender reads at a time from the source disk drive. Only takes effect when server is sender. The default of 0 will cause the Aspera sender to use its default internal buffer size, which may be different for different operating systems.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>Write Block Size (bytes)</td>
<td>This is a performance tuning parameter for an Aspera receiver. Number of bytes an ascp receiver writes data at a time onto disk drive. Only takes effect when server is receiver. The default of 0 will cause the Aspera sender to use its default internal buffer size, which may be different for different operating systems.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>Use File Cache</td>
<td>This is a performance tuning parameter for an Aspera receiver. Enable or disable per-file memory caching at the data receiver. File level memory caching improves data write speed on Windows platforms in particular, but will use more memory. We suggest using a file cache on systems that are transferring data at speeds close to the performance of their storage device, and disable it for systems with very high concurrency (because memory</td>
<td>• true  • false</td>
<td>true</td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td>----</td>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------</td>
<td>---------</td>
</tr>
<tr>
<td>8</td>
<td>Max File Cache Buffer (bytes)</td>
<td>This is a performance tuning parameter for an Aspera receiver. This value corresponds to the maximal size allocated for per-file memory cache (see Use File Cache). Unit is bytes. The default of 0 will cause the Aspera receiver to use its internal buffer size, which may be different for different operating systems.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>Resume Suffix</td>
<td>Extension name of a class of special files holding metadata information of regular data files. Useful in the context of resuming partially completed transfers. During resume mode (-k1/2/3), each data file has a corresponding metadata file with the same name and the pre-specified resume suffix.</td>
<td>text string</td>
<td>aspx</td>
</tr>
<tr>
<td>10</td>
<td>Preserve Attributes</td>
<td>Configure file creation policy. When set to none, do not preserve the timestamp of source files. When set to times, preserve the timestamp of the source files at destination.</td>
<td>none / times</td>
<td>undefined</td>
</tr>
<tr>
<td>11</td>
<td>Overwrite</td>
<td>Overwrite is an Aspera server setting that determines whether Aspera clients are allowed to overwrite files on the server. By default it is set to allow, meaning that clients uploading files to the servers will be allowed to overwrite existing files as long as file permissions allow that action. If set to deny, clients uploading files to the server will not be able to overwrite existing files, regardless of file permissions.</td>
<td>• allow / • deny</td>
<td>allow</td>
</tr>
<tr>
<td>12</td>
<td>File Manifest</td>
<td>When set to text a text file &quot;receipt&quot; of all files within each transfer session is generated. If set to disable no File Manifest is created. The file manifest is a file containing a list of everything that was transferred in a given transfer session. The filename of the File Manifest itself is automatically generated based on the transfer session's unique ID. The location where each manifest is written is specified by the File Manifest Path value. If no File Manifest Path is specified, the file will be generated under the destination path at the receiver, and under the first source path at the sender.</td>
<td>• text / • disable</td>
<td>none</td>
</tr>
<tr>
<td>13</td>
<td>File Manifest Path</td>
<td>Specify the location to store manifest files. Can be an absolute path or a path relative to the transfer user's home.</td>
<td>text string</td>
<td>blank</td>
</tr>
<tr>
<td>14</td>
<td>Pre-Calculate Job Size</td>
<td>Configure the policy of calculating total job size before data transfer. If set to any, follow client configurations (-o PreCalculateJobSize={yes</td>
<td>no}). If set to no, disable calculating job size before transferring. If set to yes, enable calculating job size before transferring.</td>
<td>• any / • yes / • no</td>
</tr>
<tr>
<td>#</td>
<td>Field</td>
<td>Description</td>
<td>Values</td>
<td>Default</td>
</tr>
<tr>
<td>----</td>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>---------</td>
</tr>
<tr>
<td>15</td>
<td>Storage Rate Control</td>
<td>Enable/Disable disk rate control. When enabled, adjust transfer rate according to the speed of receiving I/O storage, if it becomes a bottleneck.</td>
<td>true, false</td>
<td>false</td>
</tr>
<tr>
<td>16</td>
<td>File checksum method</td>
<td>Specify the type of checksum to calculate for transferred files. The content of transfers can be verified by comparing the checksum value at the destination with the value read at the source.</td>
<td>any, md5, sha1</td>
<td>any</td>
</tr>
<tr>
<td>16</td>
<td>Partial Suffix</td>
<td>Set the file suffix for partially downloaded files.</td>
<td>.aspx</td>
<td></td>
</tr>
</tbody>
</table>

### Advanced Network Options

<table>
<thead>
<tr>
<th>#</th>
<th>Field</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bind IP Address</td>
<td>Specify an IP address for server-side ascp to bind its UDP connection. If a valid IP address is given, ascp sends and receives UDP packets ONLY on the interface corresponding to that IP address.</td>
<td>Valid IPv4 address</td>
<td>blank</td>
</tr>
<tr>
<td>2</td>
<td>Bind UDP Port</td>
<td>Specify a port number for server-side ascp to bind its UDP connection. This also prevents client ascp processes from binding to same UDP port. Valid port numbers range between 1 and 65535.</td>
<td>Positive integer</td>
<td>33001</td>
</tr>
<tr>
<td>3</td>
<td>Disable Packet Batching</td>
<td>When set to true, send data packets back to back (no sending a batch of packets). This results in smoother data traffic at a cost of higher CPU usage.</td>
<td>true, false</td>
<td>false</td>
</tr>
<tr>
<td>4</td>
<td>Maximum Socket Buffer (bytes)</td>
<td>Upper bound the UDP socket buffer of an ascp session below the input value. The default of 0 will cause the Aspera sender to use its default internal buffer size, which may be different for different operating systems.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>Minimum socket buffer (bytes)</td>
<td>Set the minimum UDP socket buffer size for an ascp session.</td>
<td>Positive integer</td>
<td>0</td>
</tr>
</tbody>
</table>

### Transfer References

#### Simple Transfer Options

The following tables provide information on additional configurable settings that are available when creating simple transfers.

### Connection

<table>
<thead>
<tr>
<th>Fasp Port (UDP)</th>
<th>Specify the UDP port for FASP file transfers.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fasp proxy</td>
<td>Enable transferring through a FASP proxy server, and specify the proxy host address, port, username, and password. This feature enables the source node to bypass restrictions to the destination node for this specific transfer by using a proxy.</td>
</tr>
</tbody>
</table>
### Security

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content protection</td>
<td>Check the option to enable the content protection that encrypts the files on destination, using the entered password.</td>
</tr>
<tr>
<td>Transport encryption</td>
<td>Select aes-128 to transfer with this encryption method.</td>
</tr>
</tbody>
</table>

### Transfer

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target rate</td>
<td>Specify the transfer target rate.</td>
</tr>
<tr>
<td>Minimum rate</td>
<td>Set the transfer minimum rate.</td>
</tr>
<tr>
<td>Bandwidth policy</td>
<td>Choose a transfer policy among fixed/high/fair/low.</td>
</tr>
<tr>
<td>Retry policy</td>
<td>Check the option to enable the retry policy, as well as specify the number of attempts and the duration.</td>
</tr>
</tbody>
</table>

### Notifications

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Email address</td>
<td>To send status notifications for transfer events (start, success, or error), enter an email address and click Add. When the email address appears in the table, specify which email template to use for each transfer event.</td>
</tr>
</tbody>
</table>

### File Handling

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timestamp Filtering</td>
<td>Select this option to exclude files modified in the designated number of seconds.</td>
</tr>
<tr>
<td>Resume policy</td>
<td>Specify a resume policy and the overwrite rule when the file exists on the destination.</td>
</tr>
<tr>
<td>File attributes</td>
<td>Check the option to preserve the file permissions on the destination.</td>
</tr>
<tr>
<td>Symlinks</td>
<td>Specify how to deal with symbolic links: follow, copy, copy and force, or skip. Leave this option blank if the source is on Windows. For all others, leaving it blank is the same as choosing &quot;follow&quot;.</td>
</tr>
<tr>
<td>Source Archiving</td>
<td>Move source files to a designated directory after completing a transfer. The transfer's session details page will display the archive directory's filepath as the After transfer path. For more information on session details, see Transfer Details on page 50.</td>
</tr>
<tr>
<td>Note: The After transfer path will only be visible in the session details of the Console that initiated the transfer. Another Console monitoring the same managed nodes will not have access to the After transfer path.</td>
<td></td>
</tr>
<tr>
<td>Note: Rerunning the transfer may generate a &quot;No such file or directory&quot; error since the source files were moved to the archive directory.</td>
<td></td>
</tr>
<tr>
<td>Delete empty source subdirectories</td>
<td>This option becomes available if you selected Source Archiving. Select this option to delete any subdirectory that is emptied by the source archiving.</td>
</tr>
<tr>
<td>Note: Console does not delete the top-most directory in the source path.</td>
<td></td>
</tr>
<tr>
<td>Source Deletion</td>
<td>Check the option to delete the transferred files from the source computer.</td>
</tr>
<tr>
<td>Exclude filter</td>
<td>Enter file-name pattern Console uses to determine what files to exclude from the transfer. You can use the following two symbols in the pattern:</td>
</tr>
<tr>
<td></td>
<td>• * : The wildcard (<em>) symbol represents zero to many characters in a string. For example, the &quot;</em>.tmp&quot; pattern matches &quot;.tmp&quot; and &quot;abcde.tmp&quot;.</td>
</tr>
</tbody>
</table>
• ? : The question mark (?) represents any one character. For example, the "t?p" pattern matches "tmp" but not "temp".

<table>
<thead>
<tr>
<th>Include filter</th>
<th>Enter file-name pattern Console uses to determine what files to include in the transfer. Only files matching the filter are transferred. You can use the following two symbols in the pattern:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• * : The wildcard (<em>) symbol represents zero to many characters in a string. For example, the &quot;</em>.tmp&quot; pattern matches &quot;.tmp&quot; and &quot;abcde.tmp&quot;.</td>
</tr>
<tr>
<td></td>
<td>• ? : The question mark (?) represents any one character. For example, the &quot;t?p&quot; pattern matches &quot;tmp&quot; but not &quot;temp&quot;.</td>
</tr>
</tbody>
</table>

**Advanced**

<table>
<thead>
<tr>
<th>Initiator</th>
<th>Check this option to initiate transfers from the destination node (if possible). Console normally initiates transfers from the source node unless the source is an unmanaged node.</th>
</tr>
</thead>
<tbody>
<tr>
<td>fasp datagram size (MTU)</td>
<td>Select the option and enter the datagram size in bytes.</td>
</tr>
<tr>
<td>Read block size</td>
<td>Check the option and enter the read block size in bytes.</td>
</tr>
<tr>
<td>Write block size</td>
<td>Check the option and enter the write block size in bytes.</td>
</tr>
</tbody>
</table>

**Transfer Time**

| Transfer | Specify when to submit the transfer. **Note:** If you schedule your simple transfer for a future time, you can cancel it by going to Activity > Transfers. Select "All" from the Scheduled drop-down menu, and click Cancel. |

**Smart Transfer Options**

The following tables provide information on additional configurable settings that are available when creating smart transfers.

**Connection**

<p>| Fasp Port (UDP) | Specify the UDP port for FASP file transfers. |</p>
<table>
<thead>
<tr>
<th>Fasp proxy</th>
<th>Enable transferring through a FASP proxy server, and specify the proxy host address, port, username, and password. This feature enables the source node to bypass restrictions to the destination node for this specific transfer by using a proxy.</th>
</tr>
</thead>
</table>

### Security

| Content protection | Check the option to enable the content protection that encrypts the files on destination, using the entered password. |
| Transport encryption | Select aes-128 to transfer with this encryption method. |

### Transfer

| Target rate | Specify the transfer target rate. |
| Minimum rate | Set the transfer minimum rate |
| Bandwidth policy | Choose a transfer policy among fixed/high/fair/low. |
| Retry policy | Check the option to enable the retry policy, as well as specify the number of attempts and the duration. |

### Notifications

| Email address | To send status notifications for transfer events (start, success, or error), enter an email address and click Add. When the email address appears in the table, specify which email template to use for each transfer event. |

### File Handling

| Timestamp Filtering | Select this option to exclude files modified in the designated number of seconds. |
| Resume policy | Specify a resume policy and the overwrite rule when the file exists on the destination. |
| File attributes | Check the option to preserve the file permissions on the destination. |
| Symlinks | Specify how to deal with symbolic links: follow, copy, copy and force, or skip. Leave this option blank if the source is on Windows. For all others, leaving it blank is the same as choosing “follow”. |
| Source Archiving | Move source files to a designated directory after completing a transfer. The transfer's session details page will display the archive directory's filepath as the After transfer path. For more information on session details, see Transfer Details on page 50.  

**Note:** The After transfer path will only be visible in the session details of the Console that initiated the transfer. Another Console monitoring the same managed nodes will not have access to the After transfer path.  

**Note:** Rerunning the transfer may generate a "No such file or directory" error since the source files were moved to the archive directory.  

| Delete empty source subdirectories | This option becomes available if you selected Source Archiving. Select this option to delete any subdirectory that is emptied by the source archiving.  

**Note:** Console does not delete the top-most directory in the source path. |
| Source Deletion | Check the option to delete the transferred files from the source computer. |
Exclude filter  Enter file-name pattern Console uses to determine what files to exclude from the transfer.
You can use the following two symbols in the pattern:
- \* : The wildcard (*) symbol represents zero to many characters in a string. For example, the "*.tmp" pattern matches ".tmp" and "abcde.tmp".
- ? : The question mark (?) represents any one character. For example, the "t?p" pattern matches "tmp" but not "temp".

Include filter  Enter file-name pattern Console uses to determine what files to include in the transfer. Only files matching the filter are transferred.
You can use the following two symbols in the pattern:
- \* : The wildcard (*) symbol represents zero to many characters in a string. For example, the "*.tmp" pattern matches ".tmp" and "abcde.tmp".
- ? : The question mark (?) represents any one character. For example, the "t?p" pattern matches "tmp" but not "temp".

Advanced

Initiator  Check this option to initiate transfers from the destination node (if possible). Console normally initiates transfers from the source node unless the source is an unmanaged node.

**fp** datagram size (MTU)  Select the option and enter the datagram size in bytes.

Read block size  Check the option and enter the read block size in bytes.

Write block size  Check the option and enter the write block size in bytes.

Scheduling

Start  Click the calendar icon to select a date and time that serves as the starting basis for your recurring smart transfers. Based on the "Start" entry, Console will calculate the run time for the next occurrence (that matches the repeat rules). For example, if your start date is Friday, April 8, but your transfer is scheduled to run on Saturdays, then the first transfer will occur on Saturday, April 9.

Repeat every  Select the number of minutes, hours, days, weeks, or months to repeat this transfer. When \*weeks\* is selected, you can enable the requisite days of the week. When \*months\* is selected, you can specify whether to perform the transfer on a specific day of the month or on the "nth ___day" of the month (for example, 1st Sunday).

Until  Click the calendar icon to select a "do not go beyond" date and time. Your smart transfer will not repeat beyond this entry.

Time zone  Select your timezone from the drop-down list.

**Important:** When you have more than one destination, you can override the default smart transfer settings (with the exception of scheduling) shown in the More Options panel for each individual destination.

Watchfolder Options

The following tables provide information on additional configurable settings that are available when creating watchfolders.
Watchfolder Settings

**Note:** A watchfolder groups new or updated files it detects in its source folder into "drops". A drop is defined by the duration set by the snapshot creation period. All files in a given drop are transferred in the same transfer session, post-processed together, and reported as a unit.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Drop detection strategy | The strategy this watchfolder uses to detect files dropped into the source folder.  
  • Cool off only: Create a drop that includes new files added within the duration by the snapshot creation period.  
  • Top level files: Create a drop for each file placed in the top level of the source folder.  
  • Top level directories: Create a drop for each directory placed in the top level of the source folder. This drop also includes the sub-directories and files in the top level directory. |
| Drop detection cool off | The duration allowed for new files to be included in a drop. Aspera recommends choosing a multiple of the specified `snapshot_creation_period` for predictable results. |
| Snapshot creation period | The duration used to determine what files are included in the current drop. |
| Connect timeout | The duration the source node waits to connect to the destination node. |
| Sample period | The frequency of the system estimating the available bandwidth. |
| Queue threshold | The duration watchfolder adds files to a session. Use this feature to limit the number of files transferred based on the computed available bandwidth. |
| Retry duration | The duration in which the source node tries to establish a connection with the destination node. |
| Wait between retries | The duration the source node waits in between retries. |
| File detection cool off | The duration watchfolder in which placing a new file in the source folder does not trigger a new drop.  
  **Note:** This setting does not apply to the Cool off only detection strategy. |
| File filters | Click the button to add a new filter to identify file lists. You can set a filter to include or exclude files by globbing or by regular expression. |

Transfer

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target rate</td>
<td>The transfer target rate.</td>
</tr>
<tr>
<td>Minimum rate</td>
<td>The transfer minimum rate</td>
</tr>
<tr>
<td>Bandwidth policy</td>
<td>Choose a transfer policy among fixed</td>
</tr>
<tr>
<td>Transport Encryption</td>
<td>Select aes-128 to transfer with this encryption method.</td>
</tr>
<tr>
<td>Retry policy</td>
<td>The number of attempts and the duration between each retry.</td>
</tr>
</tbody>
</table>
### Security

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Content Protection   | Select **Encrypt transferred files with a password** to enable content encryption. Enter and confirm the password the recipient must use to decrypt the transferred files.  
  **Note:** When editing a watchfolder with content protection enabled, you must re-enter the content protection password. A password must be provided in order to save changes to the watchfolder. |

### File Handling

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resume policy</td>
<td>Specify a resume policy and the overwrite rule when the file exists on the destination.</td>
</tr>
<tr>
<td>File attributes</td>
<td>Preserve file UIDs, GIDs, or timestamps.</td>
</tr>
</tbody>
</table>
| Source Archiving     | The designated directory source files are moved to after completing a transfer. The transfer's session details page display the archive directory's filepath as the **After transfer** path.  
  **Note:** The **After transfer** path will only be visible in the session details of the Console that initiated the transfer. Another Console monitoring the same managed nodes will not have access to the **After transfer** path.  
  **Note:** Re-running the transfer may generate a "No such file or directory" error since the source files were moved to the archive directory. You can use archive directory variables in the filepath to define specific archive paths for each drop. Hover over the **Archive directory variables** link for a list of available variables.  
  Source deletion | Delete the transferred files from the source computer after transfer.                                                           |

### Growing Files

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum parallel</td>
<td>The maximum number of concurrent transfers of growing files watchfolder can initiate.</td>
</tr>
<tr>
<td>transfers</td>
<td>Target rate</td>
</tr>
<tr>
<td>Bandwidth policy</td>
<td>The bandwidth policy.</td>
</tr>
<tr>
<td>Transport encryption</td>
<td>Select aes-128 to transfer with this encryption method.</td>
</tr>
<tr>
<td>TCP port</td>
<td>The TCP port to use for this watchfolder.</td>
</tr>
<tr>
<td>fasp™ port (UDP)</td>
<td>The UDP port to use for this watchfolder.</td>
</tr>
<tr>
<td>Completion timeout</td>
<td>The amount of time to wait for the file to no longer change for the session to finish.</td>
</tr>
<tr>
<td>Memory</td>
<td>The maximum amount of memory that the faspstream binary is allowed to use.</td>
</tr>
<tr>
<td>Chunk size</td>
<td>The size of data to pack before sending over the network.</td>
</tr>
<tr>
<td>Growing file filters</td>
<td>Click the + button to add a new filter to identify growing files. You can set a filter to include or exclude files by globbing or by regular expression.</td>
</tr>
</tbody>
</table>
Packages / Drops

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package timeout</td>
<td>A package in watchfolder defines a set of files with dependencies. The package timeout defines the time in which watchfolder waits for required files. If the required files do not appear within the duration, files with dependencies are marked as not transferred because of unsatisfied dependencies.</td>
</tr>
<tr>
<td>Final transfer</td>
<td>Defines which file has to be transferred last.</td>
</tr>
<tr>
<td></td>
<td>• Last file in list: The last file in the package list is transferred last.</td>
</tr>
<tr>
<td></td>
<td>• File list: The files are transferred without any specific order.</td>
</tr>
<tr>
<td>File list filters</td>
<td>Click the [+ ] button to add a new filter to identify file lists. You can set a filter to include or exclude files by globbing or by regular expression.</td>
</tr>
</tbody>
</table>

Specify Base for Source Path

When selecting the source for a simple or smart transfer, you have the option to select Specify base for source path(s) to specify a portion of the source path to remove to place the transferred files directly into the destination folder without its hierarchy of directories.

For example, a source computer has a sent_files/project directory containing the following folders and files:

• /shared_files/project/presentation
• /shared_files/project/video_footage/take1
• /shared_files/project/video_footage/take2
• /shared_files/project/video_footage/take3

If you select the shared_files/project directory as the source, by default, the transfer includes the sent_files directory and the entirety of its contents, including its hierarchy of directories. If the destination directory is specified as /incoming, your transferred files appear as follows on the destination computer:

```
docroot/incoming/shared_files/project/presentation
docroot/incoming/shared_files/project/video_footage/take1
docroot/incoming/shared_files/project/video_footage/take2
docroot/incoming/shared_files/project/video_footage/take3
```

By selecting Specify base for source paths(s), the project folder can be excluded. Entering "/shared_files/project" in the field removes that part of the source path. Only the presentation and video_footage directories are transferred. The transferred files appear as follows on the destination computer:

```
docroot/incoming/presentation
docroot/incoming/video_footage/take1
docroot/incoming/video_footage/take2
```

If any files or folders selected for transfer fall outside the specified base path, they are omitted from the transfer. For example, if the specified path is /shared_files/project/video_footage, then presentation is not transferred at all because it is not in video_footage. Only take1, take2, and take3 are transferred. The transferred files appear as follows on the destination computer:

```
docroot/incoming/take1
docroot/incoming/take2
```

Tip: Specify base for source paths(s) can also be used to include more path depth than the default. If the source-base path is specified as /shared_files, then project and all files and folders in its folder hierarchy are
included. Similarly, if the source-base path is specified as /, the entire source path and all fields and folders in its folder hierarchy are transferred.

Report References

Reference: Basic Report Organization Options

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Address</td>
<td>Organize / summarize report by Client IP Address (client = initiator of the transfer)</td>
</tr>
<tr>
<td>Contact</td>
<td>Organize by the 'Contact' shown for a transfer. This might be a Console user name, a Faspex Server user name, SSH account, or customized value obtained from a transfer cookie. Examples: &quot;admin (console)&quot;, &quot;aspera (ssh)&quot;, &quot;aspera (faspex)&quot;.</td>
</tr>
<tr>
<td>File</td>
<td>Display a detail row for every file in every transfer.</td>
</tr>
<tr>
<td>File Extension</td>
<td>Organize / summarize report by file extension.</td>
</tr>
<tr>
<td>Server Address</td>
<td>Organize / summarize report by Server IP Address.</td>
</tr>
<tr>
<td>Session</td>
<td>Display a row for every transfer session. A transfer session represents one attempt to transfer.</td>
</tr>
<tr>
<td>Transfer</td>
<td>Display a row for every transfer. A transfer may have multiple sessions if it took multiple attempts to finish.</td>
</tr>
</tbody>
</table>

Reference: Built-In Fields for Custom Field Rules

Built-In Fields Available for Creating Custom Field Rules (for Transfer-Level Fields)

<table>
<thead>
<tr>
<th>Transfer Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Address</td>
<td>IP address of transfer initiator.</td>
</tr>
<tr>
<td>Client User</td>
<td>Client-side username. Null for all transfers, except for transfers initiated by the Console.</td>
</tr>
<tr>
<td>Contact</td>
<td>Contact assigned by Console. This can be a Console user name, a Faspex Server user name, SSH account, or customized value obtained from a transfer cookie. Examples: &quot;admin console&quot;, &quot;aspera ssh&quot;, &quot;aspera faspex&quot;.</td>
</tr>
<tr>
<td>Cookie</td>
<td>Custom identifying text attached to a transfer session. This text is used by the Console to identify and name transfers.</td>
</tr>
<tr>
<td>Destination Address</td>
<td>IP address of transfer destination (use for general purpose).</td>
</tr>
<tr>
<td>Destination Path</td>
<td>The file path on the destination machine.</td>
</tr>
<tr>
<td>Destination User</td>
<td>If upload, dest_user is the server user. If download, dest_user is client user (NULL, unless initiated from</td>
</tr>
<tr>
<td><strong>Transfer Field</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td><strong>Direction</strong></td>
<td>The direction of the transfer from the perspective of the client. &quot;Upload&quot; if the transfer is a push; &quot;Download&quot; if the transfer is a pull.</td>
</tr>
<tr>
<td><strong>Meta-tags</strong></td>
<td>JSON hash used to tag transfers with additional data.</td>
</tr>
<tr>
<td><strong>Faspex Metadata</strong></td>
<td>Information provided by Faspex, encoded in the transfer cookie. See <em>Basic Report Example: Faspex Metadata</em> on page 189.</td>
</tr>
<tr>
<td><strong>Server Address</strong></td>
<td>IP address of the server.</td>
</tr>
<tr>
<td><strong>Server User</strong></td>
<td>SSH account specified when the transfer starts (should always be displayed).</td>
</tr>
<tr>
<td><strong>Source Address</strong></td>
<td>IP address of transfer source.</td>
</tr>
<tr>
<td><strong>Source Paths</strong></td>
<td>File paths on the source machine.</td>
</tr>
<tr>
<td><strong>Source User</strong></td>
<td>If upload, source_user is the client user. If download, source_user is server user.</td>
</tr>
<tr>
<td><strong>Started Via</strong></td>
<td>The name of the application (Aspera or custom) that is responsible for initiating the transfer (for example, aspera scp, aspera sync, etc.).</td>
</tr>
<tr>
<td><strong>Token</strong></td>
<td>Security token used for the transfer (note that this depends on whether or not the application that started the transfer is configured to use tokens).</td>
</tr>
<tr>
<td><strong>Transfer Name</strong></td>
<td>Human-readable name assigned to a transfer. This name may have been keyed in by the user or automatically set by an application.</td>
</tr>
</tbody>
</table>

**Built-In Fields Available for Creating Custom Field Rules (for File-Level Fields)**

*Note:* Setting up file-level custom fields is NOT recommended for customers that transfer many small files, as this will result in scaling issues.

<table>
<thead>
<tr>
<th><strong>File Field Name</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File Bytes Transferred</strong></td>
<td>Total bytes successfully received over the network.</td>
</tr>
<tr>
<td><strong>File Error Desc</strong></td>
<td>Error message for the file, if any.</td>
</tr>
<tr>
<td><strong>File Extension</strong></td>
<td>Portion of the filename after the last period (.)</td>
</tr>
<tr>
<td><strong>File Full Destination Path</strong></td>
<td>File's full path from the destination's point-of-view.</td>
</tr>
<tr>
<td><strong>File Full Source Path</strong></td>
<td>File's full path from the source's point-of-view.</td>
</tr>
<tr>
<td><strong>File Name</strong></td>
<td>Name of the file, without its path (for example, &quot;my_file.txt&quot; rather than &quot;C:\temp\my_file.txt&quot;)</td>
</tr>
<tr>
<td><strong>File Size</strong></td>
<td>Size of the file in bytes.</td>
</tr>
<tr>
<td><strong>File Status</strong></td>
<td>Status of transfer or file (for example, &quot;running,&quot; &quot;completed,&quot; &quot;canceled&quot; or &quot;error&quot;).</td>
</tr>
</tbody>
</table>
Reference: Reporting Filters

IBM Aspera Console provides built-in filters that allow you to specify conditions for limiting the data included in your report.

<table>
<thead>
<tr>
<th>Column Heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter By</td>
<td>Select from a list of parameter names.</td>
</tr>
<tr>
<td>NOT</td>
<td>Appears as a checkbox, where unchecked represents &quot;is&quot; and checked represents &quot;is not&quot; (for example, file extension is not equal to tmp).</td>
</tr>
<tr>
<td>Comparison</td>
<td>Select from a list of operators (for example, equal to, greater than, etc.).</td>
</tr>
<tr>
<td>Value</td>
<td>Input a parameter value to complete the filter expression.</td>
</tr>
</tbody>
</table>

**Important:** Once you have added a filter, you may remove it by clicking the Remove hyperlink.

The following filter parameters are available within the Filter By drop-down list:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Parameter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{Custom Field Names}</td>
<td>Displays custom fields that you have configured for the SQL database.</td>
</tr>
<tr>
<td>File Bytes Transferred</td>
<td>File bytes successfully received over the network by the destination.</td>
</tr>
<tr>
<td>File Bytes Written</td>
<td>Files bytes successfully received over the network by the destination, plus bytes skipped for data already present at the destination.</td>
</tr>
<tr>
<td>File Error Description</td>
<td>File's error message, if any.</td>
</tr>
<tr>
<td>File Extension</td>
<td>Portion of the filename after the last period (.)</td>
</tr>
<tr>
<td>File Fullpath</td>
<td>File's directory tree hierarchy.</td>
</tr>
<tr>
<td>File Name</td>
<td>Name of the file, without its path (for example, &quot;my_file.txt,&quot; rather than &quot;C:\temp \my_file.txt&quot;).</td>
</tr>
<tr>
<td>File Session Status</td>
<td>Status of file session (for example, &quot;running,&quot; &quot;completed,&quot; &quot;canceled&quot; or &quot;error&quot;), where a file session is one file in a transfer session. A file record may group together more than one file session record if, during a transfer session, one of the files fails or is interrupted. In the next transfer session (when the transfer is retried or a hot folder handles the next batch of files to arrive), then that particular file may be retried. This will result in another file session record being created.</td>
</tr>
<tr>
<td>File Size</td>
<td>Size of the file in bytes.</td>
</tr>
<tr>
<td>File Status</td>
<td>The file status will be the status of the last/most recent file session for the file (for example, &quot;running,&quot; &quot;completed,&quot; &quot;canceled&quot; or &quot;error&quot;).</td>
</tr>
<tr>
<td>SSH Account</td>
<td>SSH account specified when the transfer starts.</td>
</tr>
<tr>
<td>Transfer Average Rate</td>
<td>Average transfer rate in bits per second.</td>
</tr>
<tr>
<td>Transfer Bytes Lost</td>
<td>Number of bytes sent by source for a particular file, but never received by destination, or never written to disk.</td>
</tr>
<tr>
<td>Transfer Bytes Transferred</td>
<td>Total bytes successfully received over the network by the destination.</td>
</tr>
<tr>
<td>Parameter Name</td>
<td>Parameter Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Transfer Bytes Written</td>
<td>Total bytes successfully received over the network by the destination, plus bytes skipped for data already present at the destination.</td>
</tr>
<tr>
<td>Transfer Client Address</td>
<td>IP address of transfer initiator.</td>
</tr>
<tr>
<td>Transfer Contact</td>
<td>Contact assigned by Console. This can be a Console user name, a Faspex user name, SSH account, or customized value obtained from a transfer cookie. Examples: &quot;admin (console)&quot;, &quot;aspera (ssh)&quot;, &quot;michael (faspex)&quot;</td>
</tr>
<tr>
<td>Transfer Cookie</td>
<td>Custom identifying text attached to a transfer session. This text is used by the Console to identify and name transfers.</td>
</tr>
<tr>
<td>Transfer Destination Address</td>
<td>IP address of transfer destination.</td>
</tr>
<tr>
<td>Transfer Destination Path</td>
<td>The file path on the destination machine.</td>
</tr>
<tr>
<td>Transfer Error Description</td>
<td>Error message for transfer or file, if any.</td>
</tr>
<tr>
<td>Transfer Files Completed</td>
<td>Number of files successfully verified at destination (i.e., the number of files actually transferred plus the number of files that were already at destination).</td>
</tr>
<tr>
<td>Transfer Files Failed</td>
<td>Number of files that failed to transfer.</td>
</tr>
<tr>
<td>Transfer Name</td>
<td>Human-readable name assigned to a transfer. This name may have been keyed in by the user or automatically set by an application.</td>
</tr>
<tr>
<td>Transfer Server Address</td>
<td>IP address of transfer server.</td>
</tr>
<tr>
<td>Transfer Session Status</td>
<td>Indicates status of transfer session (for example, &quot;running,&quot; &quot;completed,&quot; &quot;canceled&quot; or &quot;error&quot;), where the transfer session represents one execution of ascp (i.e., one attempt to transfer).</td>
</tr>
<tr>
<td></td>
<td>Note: When a transfer session is interrupted or fails and is configured to retry, a second transfer session will begin after the configured retry interval has elapsed.</td>
</tr>
<tr>
<td>Transfer Source Address</td>
<td>IP address of transfer source.</td>
</tr>
<tr>
<td>Transfer Source Paths</td>
<td>File paths on the source machine.</td>
</tr>
<tr>
<td>Transfer Status</td>
<td>A transfer will group together transfer sessions into a single item. The transfer status will be the status of the last/most recent transfer session for the transfer (for example, &quot;running,&quot; &quot;completed,&quot; &quot;canceled&quot; or &quot;error&quot;).</td>
</tr>
</tbody>
</table>

**Reference: SQL Variables for Advanced Reports**

When creating your advanced report, you may utilize the SQL variables listed below. These variables also appear within Console’s built-in, SQL script text help.

<table>
<thead>
<tr>
<th>SQL Variable</th>
<th>Description</th>
</tr>
</thead>
</table>
| $TBL_FILES     | Files table. One record in this table represents one file. At run time, this variable gets replaced with the SQL name of the table containing the file data (currently 'rpt_transfer_files'). Please note the following distinction:  
  • A FILE record can have multiple associated TRANSFER SESSION FILE records (if a file took more than one attempt to transfer).  
  • A FILE record has one and only one associated TRANSFER record ($STBL_TRANSFER_FILES.transfer_id = $STBL_TRANSFERS.id). |
<table>
<thead>
<tr>
<th>SQL Variable</th>
<th>Description</th>
</tr>
</thead>
</table>
| $TBL_TRANSFER_SESSIONS       | Transfer sessions table. One record in this table represents one attempt to transfer data. If you start a transfer and it fails, then automatically retries and succeeds, there will be two records in this table, one for the initial attempt and one for the automatic retry. For hot folder transfers, each session represents one attempt to transfer a batch of files that are currently available. If new files become available while the first batch is in progress, these may be transferred in a subsequent session, resulting in an additional record in this table. At run time, this variable gets replaced with the SQL name of the table containing the transfer session data (currently 'rpt_transfer_sessions'). Please note the following distinction:  
  • A TRANSFER SESSION record can have multiple TRANSFER SESSION FILE records (if the session attempted to transfer more than one file).  
  • A TRANSFER SESSION record has one and only one associated TRANSFER record ($TBL_TRANSFER_SESSIONS.transfer_id = $TBL_TRANSFERS.id). |
| $TBL_TRANSFER_SESSION_FILES  | Files within a transfer session. One record in this table represents one attempt to transfer a file. At run time, this variable gets replaced with the SQL name of the table containing the file session data (currently 'rpt_transfer_session_files'). Please note the following distinction:  
  • A TRANSFER SESSION FILE record has one and only one associated FILE RECORD ($TBL_TRANSFER_SESSION_FILES.transfer_file_id = $TBL_FILES.id).  
  • A TRANSFER SESSION FILE record has one and only one associated TRANSFER SESSION record ($TBL_TRANSFER_SESSION_FILES.transfer_session_id = $TBL_TRANSFER_SESSIONS.id). |
| $TBL_NODES                   | A table containing one record for each node, whether managed or unmanaged. At run time, this variable gets replaced with the SQL name of the table containing the node data (currently 'rpt_transfer_nodes').                                                                                                                                                                                                                       |
| $TBL_TRANSFERS              | A TRANSFER groups together TRANSFER SESSIONS to tie together retry attempts and hot folder file batches. Related TRANSFER SESSIONS are grouped together so that no matter how many times the session was interrupted and retried, only a single record will be present in this table. At run time, this variable gets replaced with the SQL name of the table containing the transfer data (currently 'rpt_transfers'). Please note the following distinction:  
  • A TRANSFER record can have multiple TRANSFER SESSION records (if multiple attempts or batches were required to transfer all the data).  
  • A TRANSFER record can have multiple FILE records (if the transfer consisted of more than one file). |
<p>| $FINAL_RESULT_TABLE         | This is the table where you place your final results. The data displayed on reports comes directly from this table. At run time, this variable gets replaced with a name based on an auto-generated numeric id (for example, 'report_100_results').                                                                                                                                                                                                                   |
| $TMP_TABLENAME              | If you need any temporary tables for intermediate record processing, give them names starting with &quot;$TMP_&quot; (for example,                                                                                     |</p>
<table>
<thead>
<tr>
<th>SQL Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$TMP_UNIQUE_IP_ADDRESSES</td>
<td>At run time, these variables get replaced with a name based on an auto-generated numeric id (for example, 'report_100_temp_unique_ip_addresses').</td>
</tr>
<tr>
<td>$USER_ID</td>
<td>This is the login id of user requesting report. At run time, this variable gets replaced with the numeric id of the user requesting the report.</td>
</tr>
<tr>
<td>$REPORT_PERIOD_START</td>
<td>Report period start. The user running this report will be prompted for a value at request time. (Value is converted to UTC before substitution).</td>
</tr>
<tr>
<td>$REPORT_PERIOD_END</td>
<td>Report period end. The user running this report will be prompted for a value at request time. (Value is converted to UTC before substitution).</td>
</tr>
<tr>
<td>$ANYTHING_ELSE</td>
<td>Any $NAME that does not match one of the variables is presumed to be a custom variable whose value will be provided by the report requester. See Editing Custom Variables on page 90 for instructions on how to create and configure a custom variable.</td>
</tr>
</tbody>
</table>

**Reference: Database Fields for Advanced Reports**

When creating your advanced report, you may utilize the database fields listed below. These fields (and corresponding descriptions) also appear within Console's built-in, SQL script text help.

**Note:** The term "client" refers to the machine initiating a transfer request. The term "server" refers to the machine receiving the request. These terms do not describe the direction of the file transfer. As long as a machine is the transfer initiator, it does not matter whether the machine is sending a file or receiving a file.

<table>
<thead>
<tr>
<th>Database Field</th>
<th>Description</th>
<th>Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>args_attempted</td>
<td>Number of items specifically selected by the user (either in GUI or command line).</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>args_completed</td>
<td>Out of the number of arguments attempted, the number completed successfully.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>aspera_version</td>
<td>Aspera product version for the node machine.</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>avg_loss_pct</td>
<td>Average packet loss over the network, which is calculated as a percentage.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>avg_rate</td>
<td>Average transfer rate in bits per second.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>bytes_config</td>
<td>The number of contiguous bytes that have been transferred to the destination.</td>
<td>$TBLTRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>bytes_lost</td>
<td>Number of bytes sent by source for a particular file, but never received by destination, or never written to disk.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS, $TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>bytes_pretransfer</td>
<td>If the server is configured to do so, calculates size of the transfer before the transfer starts. On the server, this corresponds to the &quot;pre-calculate job size&quot; setting.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>Database Field</td>
<td>Description</td>
<td>Table</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>bytes_remaining</td>
<td>Total bytes waiting to be sent over the network to the destination.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>bytes_transferred</td>
<td>Total bytes successfully received over the network by the destination.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS, $TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>bytes_written</td>
<td>Total bytes successfully received over the network by the destination, plus bytes skipped for data already present at the destination.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS, $TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>cipher</td>
<td>Encryption algorithm.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>client_console_ip</td>
<td>The client's IP address from the perspective of the Aspera Console application (advanced / debugging field).</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>client_err_code</td>
<td>Error code reported by the client.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>client_err_desc</td>
<td>Error code description reported by the client.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>client_external_fasp_port</td>
<td>The client's UDP port from the perspective of the server (advanced / debugging field).</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>client_external_ip</td>
<td>The client's IP address from the perspective of the server (advanced / debugging field).</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>client_file_basename</td>
<td>File's basename from client's point-of-view.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>client_file_extension</td>
<td>File's extension from client's point-of-view.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>client_file_fullpath</td>
<td>File's full path from the client's point-of-view.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>client_file_index</td>
<td>Arbitrary, unique number assigned to each file within a transfer session (on the client).</td>
<td>$TBL_TRANSFER_SESSION_FILES</td>
</tr>
<tr>
<td>client_ip</td>
<td>IP address of the transfer initiator (use for general purpose).</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>client_node_id</td>
<td>ID number assigned to the client node.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>client_node_uuid</td>
<td>Universally, unique ID number assigned to the client node.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>Database Field</td>
<td>Description</td>
<td>Table</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>client_status</td>
<td>Either the file status (running, completed, error) or the session status</td>
<td>STBL_TRANSFER_SESSION_FILES,</td>
</tr>
<tr>
<td></td>
<td>reported by the client.</td>
<td>STBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> In some cases, client and server can see different statuses</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(for example, canceled versus error).</td>
<td></td>
</tr>
<tr>
<td>client_user</td>
<td>Client-side username. Null for all transfers, except for transfers initiated</td>
<td>STBL_TRANSFERS,</td>
</tr>
<tr>
<td></td>
<td>by the Console.</td>
<td>STBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>contact</td>
<td>Contact assigned by Console. This can be a Console user name, a Faspex</td>
<td>STBL_TRANSFERS,</td>
</tr>
<tr>
<td></td>
<td>user name, SSH account, or customized value obtained from a transfer cookie.</td>
<td>STBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td></td>
<td>Examples: &quot;admin (console)&quot;, &quot;aspera (ssh)&quot;, &quot;michael (faspex)&quot;</td>
<td></td>
</tr>
<tr>
<td>cookie</td>
<td>Custom identifying text attached to a transfer session. This text is used</td>
<td>STBL_TRANSFERS,</td>
</tr>
<tr>
<td></td>
<td>by the Console to identify and name transfers.</td>
<td>STBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>dest_endpoint_id</td>
<td>ID number assigned to the destination endpoint.</td>
<td>STBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>dest_file_basename</td>
<td>File's basename from destination's point-of-view.</td>
<td>STBL_FILES</td>
</tr>
<tr>
<td>dest_file_extension</td>
<td>File's extension from destination's point-of-view.</td>
<td>STBL_FILES</td>
</tr>
<tr>
<td>dest_file_fullpath</td>
<td>File's full path from the destination's point-of-view.</td>
<td>STBL_FILES</td>
</tr>
<tr>
<td>dest_ip</td>
<td>IP address of transfer destination (use for general purpose).</td>
<td>STBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>dest_node_id</td>
<td>ID number assigned to the destination node.</td>
<td>STBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>dest_path</td>
<td>The file path on the destination machine.</td>
<td>STBL_TRANSFERS,</td>
</tr>
<tr>
<td></td>
<td>STBL_TRANSFER_SESSIONS</td>
<td></td>
</tr>
<tr>
<td>dest_user</td>
<td>If upload, dest_user is the server user. If download, dest_user is client</td>
<td>STBL_TRANSFERS,</td>
</tr>
<tr>
<td></td>
<td>user (NULL, unless initiated from Console). For everyday purposes,</td>
<td>STBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td></td>
<td>recommend using contact field instead.</td>
<td></td>
</tr>
<tr>
<td>dirs_pretransfer</td>
<td>If the server is configured to do so, calculates number of directories to</td>
<td>STBL_TRANSFERS,</td>
</tr>
<tr>
<td></td>
<td>be transferred. Only calculated if &quot;pre-calculate job size&quot; setting is</td>
<td>STBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td></td>
<td>turned on.</td>
<td></td>
</tr>
<tr>
<td>dirscans_completed</td>
<td>Number of directory scans completed.</td>
<td>STBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>Database Field</td>
<td>Description</td>
<td>Table</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>err_desc</td>
<td>Error message for transfer or file, if any.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>fallback_protocol</td>
<td>If the transfer has been configured to retry using the HTTP fallback protocol, then this field will report &quot;http.&quot; If not, will be NULL.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>filebasename</td>
<td>Name of the file, without its path (for example, &quot;my_file.txt,&quot; rather than &quot;C:\temp\my_file.txt&quot;)</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>file_extension</td>
<td>Portion of the filename after the last period (.)</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>filefullpath</td>
<td>Full path to the file.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>files_attempted</td>
<td>Number of files attempted to be sent over the network.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>files_complete</td>
<td>Number of files successfully verified at destination, that is, the number of files actually transferred + number of files that were already at destination.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>files_failed</td>
<td>Number of files that failed to transfer.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>files_pretransfer</td>
<td>If the server is configured to do so, calculates number of files to be transferred. Only calculated if &quot;pre-calculate job size&quot; setting is turned on.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>files_skipped</td>
<td>Number of files skipped.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>filescans_completed</td>
<td>The number of file scans completed.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>hostname</td>
<td>The local name of the node machine (which is only filled in for managed nodes). Note that a node machine will be called &quot;localhost&quot; if it hasn't been previously named.</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>id</td>
<td>Unique integer ID assigned by the Console (used as an internal field).</td>
<td>$TBL_NODES, $TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS, $TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>initiated_by_source</td>
<td>Identifies an &quot;upload.&quot; If this field is equal to 1, then whoever started the transfer is uploading.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS, $TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>last_client_ip</td>
<td>The client's IP address from the last session of a multiple session transfer.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>Database Field</td>
<td>Description</td>
<td>Table</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>last_client_node_id</td>
<td>ID number assigned to the client node during the last session of a multiple session transfer.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>last_dest_ip</td>
<td>The destination's IP address from the last session of a multiple session transfer.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>last_dest_node_id</td>
<td>ID number assigned to the destination node during the last session of a multiple session transfer.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>last_err_desc</td>
<td>Error description from the last session of a multiple session transfer.</td>
<td>$TBL_TRANSFERS, $TBL_FILES</td>
</tr>
<tr>
<td>last_network_delay</td>
<td>The lag on the network (RTT, measured in milliseconds) from the last session of a multiple session transfer.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>last_restarted_at</td>
<td>The last date/time that the node machine was restarted (for managed node's only).</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>last_retry_timeout</td>
<td>The number of seconds that the server waited to try again (after a failure), during the last session of a multiple transfer session.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>last_server_ip</td>
<td>The server's IP address from the last session of a multiple session transfer.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>last_server_node_id</td>
<td>ID number assigned to the server node during the last session of a multiple session transfer.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>last_source_ip</td>
<td>The source's IP address from the last session of a multiple session transfer.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>last_source_node_id</td>
<td>ID number assigned to the source node during the last session of a multiple session transfer.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>last_transfer_session_file_id</td>
<td>ID number assigned to the file during the last transfer session.</td>
<td>$TBL_FILES</td>
</tr>
<tr>
<td>last_transport</td>
<td>Transport mechanism (&quot;fasp2&quot; for Aspera protocol, &quot;http&quot; for fallback protocol) from the last session of a multiple session transfer.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>max_rate</td>
<td>Maximum transfer rate.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>min_rate</td>
<td>Minimum transfer rate.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>mkdirs_attempted</td>
<td>Number of directories that were attempted to be created at the destination.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>Database Field</td>
<td>Description</td>
<td>Table</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>mkdirs_failed</td>
<td>Number of directories that failed to be created at the destination.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>mkdirs_passed</td>
<td>Number of directories that were created successfully at the destination.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>name cf</td>
<td>Human-readable name assigned to a transfer. This name may have been keyed in by the user or automatically set by an application.</td>
<td>$TBL_NODES, $TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>network_delay</td>
<td>Lag on the network (RTT), which is measured in milliseconds.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>operation</td>
<td>Either upload or download from the perspective of the client (the initiator). Upload if pushing; download if pulling.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>os</td>
<td>The node machine's Operating System.</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>os_version</td>
<td>The version of the node machine's Operating System.</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>paths_attempted</td>
<td>The total number of files and directories attempted.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>paths_excluded</td>
<td>The number of files and directories that were not transferred because of an exclusion rules.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>paths_failed</td>
<td>The number of files and directories that failed to transfer. A failure is counted if the sender was unable to read a source file or the destination was unable to write the file.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>paths_irreg</td>
<td>This is the total number of special files (for example, nodes, pipes, memory mapped files, page files or /proc files). These files are never transferred.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>pct_complete</td>
<td>Percent (%) of transfer that has been completed. NULL if the node is server is not configured to pre-calculate job size.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>pretransfer_stats_changed</td>
<td>Between one attempt to the next (retries and sync), whether or not the size of the transfer has changed (grew or reduced in size).</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>primary_address</td>
<td>The node's actual IP address (which has been keyed into the Console interface).</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>priority</td>
<td>Normal or high (only valid when the policy if adaptive).</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>Database Field</td>
<td>Description</td>
<td>Table</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------</td>
</tr>
<tr>
<td>reported_by_both_sides</td>
<td>Database logger added information to the Console from both ends of the transfer (both source and destination are managed nodes and both are sending data back to the database).</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>reported_by_server</td>
<td>If this field is equal to 1, then Console received data from the server node. If this field is equal to 0, then the server was not a managed node or failed to log.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>reported_policy</td>
<td>High, fixed, adaptive or trickle.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>reported_priority</td>
<td>Normal or high (only valid when the policy is adaptive).</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>retry_timeout</td>
<td>After a transfer fails, the number of seconds the server will wait before trying again.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>seconds_remaining</td>
<td>Seconds remaining for the file transfer.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>server_console_ip</td>
<td>Internal IP address of the server (inputted into the nodes page inside Console). Note that this field is primarily used for testing.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>server_err_code</td>
<td>Error code reported by the server.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>server_err_desc</td>
<td>Error code description reported by the server.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>server_external_fasp_port</td>
<td>External fasp (UDP) port of the server. Note that this field is primarily used for testing.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>server_external_ip</td>
<td>External IP address of the server. Note that this field is primarily used for testing.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>server_file_basename</td>
<td>File's basename from server's point-of-view.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>server_file_extension</td>
<td>File's extension from server's point-of-view.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>server_file_fullpath</td>
<td>File's full path from the server's point-of-view.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>server_file_index</td>
<td>Arbitrary, unique number assigned to each file within a transfer session (on the server)</td>
<td>$TBL_TRANSFER_SESSION_FILES</td>
</tr>
<tr>
<td>server_ip</td>
<td>IP address of transfer server.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>server_node_id</td>
<td>ID assigned to the server node.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>Database Field</td>
<td>Description</td>
<td>Table</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>server_node_uuid</td>
<td>Universally, unique ID assigned to the server node.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>server_status</td>
<td>Either the file status (running, completed, error) or the session status reported by the server. Note that in some cases, client and server can see different statuses (for example, canceled versus error).</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>server_user</td>
<td>SSH account specified when the transfer starts (should always be displayed).</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>session_count</td>
<td>Number of sessions required for the transfer.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Hot folders can span many sessions.</td>
<td></td>
</tr>
<tr>
<td>session_file_count</td>
<td>Number of sessions required to send a particular file.</td>
<td>$TBL_FILES</td>
</tr>
<tr>
<td>session_id</td>
<td>ID assigned to transfer session.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>size</td>
<td>Size of the file in bytes.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>size_changed</td>
<td>The change in file size from one transfer attempt to another.</td>
<td>$TBL_FILES</td>
</tr>
<tr>
<td>soap_active_sessions</td>
<td>Number of transfer sessions running on the node.</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>source_endpoint_id</td>
<td>ID assigned to the source endpoint.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>source_file_basename</td>
<td>File's basename from source's point-of-view.</td>
<td>$TBL_FILES</td>
</tr>
<tr>
<td>source_file_extension</td>
<td>File's extension from source's point-of-view.</td>
<td>$TBL_FILES</td>
</tr>
<tr>
<td>source_file_fullpath</td>
<td>File's full path from the source's point-of-view.</td>
<td>$TBL_FILES</td>
</tr>
<tr>
<td>source_ip</td>
<td>IP address of transfer source.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>source_node_id</td>
<td>ID assigned to the source node.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>source_paths</td>
<td>File paths on the source machine.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>source_paths_changed</td>
<td>Between one transfer attempt to the next, whether or not the file source paths have changed.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td>source_user</td>
<td>If upload, source_user is the client user. If download, source_user is server user.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>ssh_port</td>
<td>Node machine's SSH port.</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>Database Field</td>
<td>Description</td>
<td>Table</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td>ssh_tunnel_port</td>
<td>Node machine's SSH tunnel port.</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>start_byte</td>
<td>Displays the point at which data from the file started transferring to the destination (relevant if some of the file has already been transferred). If the file has already been transferred to the destination, then the start byte equals the total file size.</td>
<td>$TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>started_at</td>
<td>Date and time that a transfer or file started.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS, $TBL_FILES</td>
</tr>
<tr>
<td>started_via</td>
<td>The name of the application (Aspera or custom) that is responsible for initiating the transfer (for example, aspera.scp, aspera.sync, etc.).</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>status</td>
<td>Status of transfer or file (for example, &quot;running,&quot; &quot;completed,&quot; &quot;canceled&quot; or &quot;error&quot;).</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS, $TBL_FILES</td>
</tr>
<tr>
<td>stopped_at</td>
<td>Date and time that a transfer or file stopped (value is blank if transfer or file is still active).</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS, $TBL_FILES</td>
</tr>
<tr>
<td>target_rate</td>
<td>Target transfer rate.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>tmp_actual_rate</td>
<td>Reserved for future use.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>tmp_actual_rate_calculated_at</td>
<td>Reserved for future use.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>tmp_loss_pct</td>
<td>Reserved for future use.</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>token</td>
<td>Security token used for the transfer (note that this depends on whether or not the application that started the transfer is configured to use tokens).</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>transfer_file_id</td>
<td>Corresponds to the file ID field</td>
<td>$TBL_TRANSFER_SESSION_FILES</td>
</tr>
<tr>
<td>transfer_id</td>
<td>Corresponds to ID field.</td>
<td>$TBL_TRANSFER_SESSIONS, $TBL_FILES</td>
</tr>
<tr>
<td>transfer_session_id</td>
<td>Corresponds to transfer session ID field</td>
<td>$TBL_TRANSFER_SESSION_FILES</td>
</tr>
<tr>
<td>transfer_uuid</td>
<td>Universally, unique ID that is used to identify the transfer as a whole. May contain multiple sessions and is generated by application that started the transfer. Generally only populated by transfers started by Console.</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>Database Field</td>
<td>Description</td>
<td>Table</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
<td>-------</td>
</tr>
<tr>
<td>transport</td>
<td>fasp2 for Aspera protocol, &quot;http&quot; for fallback protocol</td>
<td>$TBL_TRANSFER_SESSIONS</td>
</tr>
<tr>
<td>type</td>
<td>Managed or unmanaged node.</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>udp_port</td>
<td>Node machine's UDP port.</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>use_ssh_tunnel</td>
<td>Set up an SSH tunnel for database logging (for managed nodes only).</td>
<td>$TBL_NODES</td>
</tr>
<tr>
<td>usecs</td>
<td>Length of the transfer session (in milliseconds). Not authoritative (use only for transfer sessions and transfers).</td>
<td>$TBL_TRANSFERS, $TBL_TRANSFER_SESSIONS, $TBL_TRANSFER_SESSION_FILES, $TBL_FILES</td>
</tr>
<tr>
<td>uuid</td>
<td>Universally, unique identifier that is generated on the node when installing Aspera software (for managed nodes only)</td>
<td>$TBL_NODES</td>
</tr>
</tbody>
</table>

**Important:** If you have configured custom fields, they will be prefixed with "cf_". Custom fields are utilized in the $TBL_FILES and $TBL_TRANSFER tables. Please note that if you would like to add additional custom fields, you may do so via the **Configuration > Custom Fields**. For instructions on setting up a custom field, see *Creating Custom Fields* on page 91.

## Advanced Report Usage Notes

### Advanced Report Usage Notes: Avoid Duplicating Identical Records

Console's security filtering prioritizes speed over the cost of potentially returning duplicate records. It is up to the report writer to remove duplicate records returned when querying report tables directly.

For example, a user unaware of Console internals might expect the following to always return no more than a single record:

```sql
SELECT ts.*
FROM $TBL_TRANSFER_SESSIONS ts
WHERE ts.session_id='ed0a9b4039bb40dfa86690ff7e1f6fa2';
```

However, depending on the user's group memberships and permissions, the above could return multiple identical records. To correct this, use **SELECT DISTINCT**. For example:

```sql
SELECT DISTINCT ts.*
FROM $TBL_TRANSFER_SESSIONS ts
WHERE ts.session_id='ed0a9b4039bb40dfa86690ff7e1f6fa2';
```

Be aware that this means you cannot directly perform aggregate computations--such as **SUM**, **AVERAGE**, or **COUNT**--on the reporting tables. For example, in the following, **total_bytes_transferred** could count some sessions multiple times:

```sql
SELECT DISTINCT
```
ts.contact, SUM(ts.bytes_transferred) AS total_bytes_transferred
FROM $TBL_TRANSFER_SESSIONS ts
WHERE ... ;

Instead, first extract just the data of interest to a temporary table, then summarize from there:

# Create holding table for filtered raw data
CREATE TABLE $TMP_FILTERED_TRANSFER_SESSIONS (  `id` INT(11) NOT NULL AUTO_INCREMENT PRIMARY KEY ,  `contact` VARCHAR(255) ,  `bytes_transferred` BIGINT(20) );

# Extract relevant data (very important to include ts.id)
INSERT INTO $TMP_FILTERED_TRANSFER_SESSIONS
SELECT DISTINCT  ts.id ,  ts.contact ,  ts.bytes_transferred
FROM $TBL_TRANSFER_SESSIONS ts
WHERE  ts.started_at < '$REPORT_PERIOD_END'
AND (  ts.stopped_at >= '$REPORT_PERIOD_START'
OR ts.stopped_at IS NULL );

# Summarize by contact
CREATE TABLE $FINAL_RESULT_TABLE
SELECT  fts.contact ,  SUM(fts.bytes_transferred) AS total_bytes_transferred
FROM $TMP_FILTERED_TRANSFER_SESSIONS fts
GROUP BY  fts.contact
ORDER BY  fts.contact ;

Advanced Report Usage Notes: Avoid Duplicating Redundant Records

Transfers between two managed nodes create two records per file, one in $TBL_TRANSFER_SESSION_FILES and one in $TBL_FILES.

If both source and destination are managed nodes, then both sides log to the database. These records will not be identical—the record logged by the server reports the server-side path, while the record logged by the client reports the client-side path. Sometimes other fields, such as err_desc, may differ as well.

There are several fields in the canonical tables supplied specifically to address this issue:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
<th>Tables</th>
</tr>
</thead>
<tbody>
<tr>
<td>reported_by_both_sides</td>
<td>0 if transfer was only logged by one side.</td>
<td>$TBL_TRANSFERS</td>
</tr>
<tr>
<td></td>
<td>1 if transfer was logged by both server and client.</td>
<td></td>
</tr>
<tr>
<td>Field Name</td>
<td>Description</td>
<td>Tables</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------------------------</td>
</tr>
</tbody>
</table>
| reported_by_server | 0 if the file record was logged by the client.  
1 if the file record was logged by the server. | $TBL_FILES  
$TBL_TRANSFER_SESSION_FILES |
| initiated_by_source | 0 if transfer is a pull (client is the destination).  
1 if transfer is a push (client is the source). | $TBL_TRANSFERS  
$TBL_TRANSFER_SESSIONS  
$TBL_FILES  
$TBL_TRANSFER_SESSION_FILES |

To ensure that each file is present only once in a result set, we need to use the above fields to give precedence to the record from one side or the other.

*Note:* The previous caveat about record duplication ([Advanced Report Usage Notes: Avoid Duplicating Identical Records](#)) also applies (i.e. the file record reported by the server node could itself be returned multiple times, as well as the record reported by the client node).

*Note:* Certain edge cases cause a problem even when using the above filter. For example, if both nodes start reporting a transfer session and one node loses its connection to the database, then `reported_by_both_sides` will equal 1, but not all of the file records will have two records in the file tables.

The following SQL example, taken from the built-in Activity Summary By Contact report, gives the *destination-side* file record precedence in cases where both sides logged the transfer.

```sql
#==================================================
# Set variables to hold report datetime parameters  
# (all datetimes are converted to UTC)  
#==================================================
SET @report_period_start = '$REPORT_PERIOD_START';  
SET @report_period_end = '$REPORT_PERIOD_END';
#===================================================
# PRE-FILTER RECORD IDS  
# Initially retrieve just the id columns from  
# base tables (improves performance by avoiding  
# queries with more than one join)  
#===================================================
# Create tables to hold the prefiltered record IDs  
#===================================================
CREATE TABLE $TMP_TRANSFER_IDS (  
id INT NOT NULL PRIMARY KEY  
, reported_by_both_sides TINYINT(1) NOT NULL DEFAULT 0  );
CREATE TABLE $TMP_TRANSFER_SESSION_IDS (  
id INT NOT NULL PRIMARY KEY  
, reported_by_both_sides TINYINT(1) NOT NULL DEFAULT 0  );
CREATE TABLE $TMP_FILE_SESSION_IDS (  
id INT NOT NULL PRIMARY KEY  
, transfer_session_id INT NOT NULL  );
#---------------------------------------------------
# Retrieve IDs  
#---------------------------------------------------
# Transfers  
#---------------------------------------------------
INSERT INTO $TMP_TRANSFER_IDS  
SELECT DISTINCT
```
```
INSERT INTO $TMP_TRANSFER_SESSION_IDS
SELECT DISTINCT
    ts.id,
    t.reported_by_both_sides
FROM $TBL_TRANSFER_SESSIONS ts
JOIN $TMP_TRANSFER_IDS t
    ON ts.transfer_id = t.id
WHERE
    (ts.started_at < @report_period_end
    AND (ts.stopped_at >= @report_period_start
    OR ts.stopped_at IS NULL
    )
    )
;
#---------------------------------------------------
# File Sessions (choose destination-side
# info if both sides logged to db)
#---------------------------------------------------
INSERT INTO $TMP_FILE_SESSION_IDS
SELECT DISTINCT
    fs.id,
    fs.transfer_session_id
FROM $TBL_TRANSFER_SESSION_FILES fs
JOIN $TMP_TRANSFER_SESSION_IDS ts
    ON fs.transfer_session_id = ts.id
WHERE
    (fs.started_at < @report_period_end
    AND (fs.stopped_at >= @report_period_start
    OR fs.stopped_at IS NULL
    )
    )
    AND (ts.reported_by_both_sides=0
    OR (fs.reported_by_server=1
    AND fs.initiated_by_source=1)
    OR (fs.reported_by_server=0
    AND fs.initiated_by_source=0)
    )
;
CREATE INDEX idx_transfer_session_id
ON $TMP_FILE_SESSION_IDS (transfer_session_id);
```
Advanced Report Usage Notes: Filter on Raw Values

Filtering on computed values in most cases prevents MySQL from being able to take advantage of indexes. For example, the following will force a scan of every record in TBL_TRANSFER_SESSIONS, because MySQL has to perform the CONVERT( ) on ts.started_at for every record:

```sql
SELECT DISTINCT ts.*
FROM
 $TBL_TRANSFER_SESSIONS ts
WHERE
 CONVERT(ts.started_at, DATE) = DATE(NOW())
;
```

Instead, compute the correct criteria to compare the raw value against:

```sql
SET @todays_date = DATE(NOW());
SET @tomorrows_date = DATE_ADD(@todays_date, INTERVAL 1 DAY);
SELECT DISTINCT ts.*
FROM
 $TBL_TRANSFER_SESSIONS ts
WHERE
 ts.started_at >= @todays_date
AND ts.started_at < @tomorrows_date
;
```

⚠️ Note: Even the above will only give expected results if you are in GMT time zone, as NOW( ) will return UTC time.

The builtin report variables $REPORT_PERIOD_START and $REPORT_PERIOD_END contain datetimes converted from local time zone of input into UTC and are usually a better choice for date filtering (unless recipient is fine with UTC-based filtering).

Advanced Report Usage Notes: Filter Strings by Using "Begins With"

If possible, filter strings by matching “begins with” rather than “contains” or “ends with”. If that's not possible, consider creating a custom field. For example, the following will not be able to use the index on ts.contact:

```sql
SELECT DISTINCT ts.*
FROM
 $TBL_TRANSFER_SESSIONS ts
WHERE
 ts.contact LIKE '%Euro2012_Livex%'
;
```

If you know all the possible ways the string could begin, you could enumerate them like this:

```sql
SELECT DISTINCT ts.*
FROM
 $TBL_TRANSFER_SESSIONS ts
WHERE
 ts.contact LIKE 'AA_Euro2012_Livex%'
OR ts.contact LIKE 'BB_Euro2012_Livex%'
OR ts.contact LIKE 'CC_Euro2012_Livex%'
;
```

If the report is only to be run for a small date range and there are few transfer sessions then you may not need to worry about this. If you expect to be running over large date ranges and large numbers of sessions, then you should create a custom field that detects the presence of the match string and then copies it to the custom field -- you could then filter on the custom field instead.
Advanced Report Usage Notes: Always Include a Date Filter

To avoid creating a report that might try to work on the entire database you should always include some kind of date filter.

The recommended option is to use the built-in report variables $REPORT_PERIOD_START and $REPORT_PERIOD_END to filter data. If an advanced report contains these variables, the web UI will include date pickers when the user runs the report. A standard filter to find all transfers that were active at any time during the report period would look like the following:

```sql
SELECT DISTINCT
    ts.id,
    ts.contact,
    ts.bytes_transferred
FROM
    $TBL_TRANSFER_SESSIONS ts
WHERE
    ts.started_at < '$REPORT_PERIOD_END'
    AND (
        ts.stopped_at >= '$REPORT_PERIOD_START'
        OR ts.stopped_at IS NULL
    );
```

Note the clause `OR ts.stopped_at IS NULL`. Without this, the report would exclude any transfers that were still running at the time the report was run. Depending on the intended purpose of the report, you might need to prorate data for transfers that were active for only part of the reporting period, such as cases 2, 3, and 4 in the following:

As an alternative, you can avoid the use of $REPORT_PERIOD_START and $REPORT_PERIOD_END if you are creating a report that always looks at the last X hours:

```sql
SET @min_start = DATE_SUB(NOW(), INTERVAL 24 HOUR);
CREATE TABLE $FINAL_RESULT_TABLE
SELECT DISTINCT ts.*
FROM
    $TBL_TRANSFER_SESSIONS ts
WHERE
    ts.started_at >= @min_start
```
Note: As of Console 1.6 there is a bug in the report engine that causes the creation of Excel/CSV files to fail if you do not reference `$REPORT_PERIOD_START` and `$REPORT_PERIOD_END` at all. To work around this, include a dummy reference in the report SQL. For example:

```sql
SET @dummy = '$REPORT_PERIOD_START';
```

When running the report, users are then asked for report period dates, but they will be ignored.

**Advanced Report Usage Notes: Always Name Your Computed or Aggregated Columns**

Always name your computed or aggregate columns, and avoid names that might be reserved words. In particular, do not call a final result column "name", "count", "id", and so on.

**INCORRECT:**

```sql
CREATE TABLE $FINAL_RESULT_TABLE
SELECT
  fts.contact
  , COUNT(*)
  , SUM(fts.bytes_transferred)
...
```

**CORRECT:**

```sql
CREATE TABLE $FINAL_RESULT_TABLE
SELECT
  fts.contact
  , COUNT(*) AS session_count
  , SUM(fts.bytes_transferred) AS total_bytes_transferred
...
```

**Advanced Report Usage Notes: Avoid Joining Reporting Views**

MySQL often mis-optimizes queries that join reporting views directly to each other. The fact that the views can show the same record multiple times can cause a geometric explosion in the number of temporary records inspected.

**EXAMPLE:**

```sql
# Find all sessions that contained file "foo.txt"
# List both session info and file info
# ASSUMES NO SESSIONS WERE BETWEEN TWO MANAGED NODES
SET @report_period_start = '$REPORT_PERIOD_START';
SET @report_period_end = '$REPORT_PERIOD_END';
CREATE TABLE $FINAL_RESULT_TABLE
SELECT DISTINCT
  ts.session_id
  , ts.source_ip
  , ts.dest_ip
  , ts.started_at
  , ts.stopped_at
  , ts.status
  , tsf.file_fullpath
  , tsf.size
  , tsf.started_at AS file_started_at
  , tsf.stopped_at AS file_stopped_at
  , tsf.status AS file_status
FROM
```
$TBL_TRANSFER_SESSIONS ts
JOIN $TBL_TRANSFER_SESSION_FILES tsf
  ON ts.id = tsf.transfer_session_id
WHERE
  tsf.started_at < @report_period_end
  AND (tsf.stopped_at >= @report_period_start
    OR tsf.stopped_at IS NULL)
  AND tsf.file_basename = "foo.txt"
ORDER BY
  ts.started_at,
  tsf.started_at;

Although the above report uses SELECT DISTINCT, contains no aggregate functions such as COUNT and SUM, and generates a correct final result (unless any of the transfer sessions were between two managed nodes), it is potentially slow. For greater speed (and to prevent query misoptimization from MySQL), it is better to decompose the above query into smaller steps, and join your temporary tables to the report views instead of joining the report views together directly.

Note: In order to avoid complexity in the SQL, the example below assumes no sessions were between two managed nodes. Therefore, the code for dealing with this has been left out (see Advanced Report Usage Notes: Avoid Duplicating Redundant Records on page 176).

EXAMPLE

SET @report_period_start = '$REPORT_PERIOD_START';
SET @report_period_end = '$REPORT_PERIOD_END';

#------------------------------------------------
# Create tables to prefilter base table record ids
#------------------------------------------------
CREATE TABLE $TMP_TRANSFER_SESSION_IDS (
  id INT NOT NULL PRIMARY KEY
);
CREATE TABLE $TMP_TRANSFER_SESSION_FILE_IDS (
  id INT NOT NULL PRIMARY KEY
, transfer_session_id INT NOT NULL
);

#------------------------------------------------
# Create table to hold all desired fields from
# transfer_sessions
#------------------------------------------------
CREATE TABLE $TMP_TRANSFER_SESSION_DATA (
  `id` INT(11) NOT NULL AUTO_INCREMENT PRIMARY KEY
, `session_id` VARCHAR(36)
, `source_ip` VARCHAR(255)
, `dest_ip` VARCHAR(255)
, `started_at` DATETIME
, `stopped_at` DATETIME
, `status` VARCHAR(255)
);

#------------------------------------------------
# Create table to hold all desired fields from
# transfer_session_files
#------------------------------------------------
CREATE TABLE $TMP_TRANSFER_SESSION_FILE_DATA (  'id' INT(11) NOT NULL AUTO_INCREMENT PRIMARY KEY  , 'transfer_session_id' INT(11)  , 'started_at' DATETIME  , 'stopped_at' DATETIME  , 'status' VARCHAR(255)  , 'file_fullpath' TEXT  , 'size' BIGINT(20) );

#========================================
# PRE-FILTER BASE TABLE IDS
#========================================

# For this report, we know we are
# filtering on file name and can use
# the index on that column, so it is
# faster to find the records from
# transfer_session_files first
#------------------------------------------------

# Transfer Session Files
#------------------------------------------------

INSERT INTO $TMP_TRANSFER_SESSION_FILE_IDS
SELECT DISTINCT
  tsf.id
 , tsf.transfer_session_id
FROM $TBL_TRANSFER_SESSION_FILES tsf
WHERE
  tsf.started_at < @report_period_end
AND (
    tsf.stopped_at >= @report_period_start
  OR tsf.stopped_at IS NULL
)
  AND tsf.file_basename = "foo.txt" ;

#------------------------------------------------

# Create an index on the join field -
# for speed, we wait until table is
# populated instead of defining the index
# during initial creation of table
#------------------------------------------------

CREATE INDEX idx_transfer_session_id ON
$TMP_TRANSFER_SESSION_FILE_IDS (transfer_session_id);

#-------------------
# Transfer Sessions
#-------------------

INSERT INTO $TMP_TRANSFER_SESSION_IDS
SELECT DISTINCT ts.id
FROM $TBL_TRANSFER_SESSIONS ts
JOIN $TMP_TRANSFER_SESSION_FILE_IDS tsf
  ON ts.id = tsf.transfer_session_id
WHERE
  ts.started_at < @report_period_end
AND (
    ts.stopped_at >= @report_period_start
  OR ts.stopped_at IS NULL

Remove transfer file sessions that don't have an associated transfer_session record
(normally not supposed to happen, but we want to protect against bad data that might be caused by system crash, logger errors, console purge errors, Canonicalizer shutdown, etc.)
For this particular report, this is not needed since the final join will weed out such records, but it is a good habit to maintain, this report could be modified later into one where it would make a difference.

DELETE tsf.*
FROM $TMP_TRANSFER_SESSION_FILE_IDS tsf
LEFT JOIN $TMP_TRANSFER_SESSION_IDS ts
ON tsf.transfer_session_id = ts.id
WHERE ts.id IS NULL;

Get all desired fields

# transfer_session_files

INSERT INTO $TMP_TRANSFER_SESSION_FILE_DATA (id, `transfer_session_id`, `started_at`, `stopped_at`, `status`, `file_fullpath`, `size`)
SELECT DISTINCT
  tsf.id, tsf.transfer_session_id, tsf.started_at, tsf.stopped_at, tsf.status, tsf.file_fullpath, tsf.size
FROM $TMP_TRANSFER_SESSION_FILE_IDS tsf_ids
STRAIGHT_JOIN $TBL_TRANSFER_SESSION_FILES tsf
  ON tsf_ids.id = tsf.id

# Add index to speed joins

CREATE INDEX idx_transfer_session_id ON $TMP_TRANSFER_SESSION_FILE_DATA (`transfer_session_id`);

# transfer_sessions
Example Reports

Basic Report Example: Faspex User Activity

The following example demonstrates the process of creating a new, basic report (following the instructions described in Creating a Basic Report on page 87) for Faspex users. In our example, we will generate a report that displays transfer activity by Faspex users only. The example report, once generated, will display total bytes transferred by each Faspex Server user, along with file and transfer-level detail (where a transfer groups together transfer sessions into a single item).
1. Go to the *Manage Report Types* page.
   Select **Reports** from the Console menu, and then click the **Manage Report Types** button. On the *Manage Report Types* screen, click the **New Basic** button.

2. Configure your basic report to display file- and transfer-level details, organized by Faspex Users.
   On the *Create New Report Type* page (for **basic** reports), enter the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Basic Faspex User Report</td>
</tr>
<tr>
<td>Description</td>
<td>Basic Faspex Server User report, which includes total bytes per Faspex User, as well as file- and transfer-level details.</td>
</tr>
<tr>
<td>How would you like to organize this report?</td>
<td>Select &quot;Contact,&quot; &quot;Transfer,&quot; and &quot;File&quot; as the fields by which to organize this report. In doing so, the report will be grouped by the following fields:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Contact</strong> (Contact assigned by Console. This can be a Console user name, a Faspex Server user name, SSH account, or customized value obtained from a transfer cookie. Examples: &quot;admin (console)&quot;, &quot;aspen (ssh)&quot;, &quot;michael (faspex)&quot;).</td>
</tr>
<tr>
<td></td>
<td>• <strong>Transfer</strong> (Human-readable name assigned to a transfer. A transfer represents one or multiple executions of <em>ascp</em> (that is, one or multiple attempts to transfer).)</td>
</tr>
<tr>
<td></td>
<td>• <strong>File</strong> (File's name)</td>
</tr>
<tr>
<td>Columns to include</td>
<td>Select the following basic fields to include as columns:</td>
</tr>
<tr>
<td></td>
<td>• bytes transferred</td>
</tr>
<tr>
<td></td>
<td>• average rate</td>
</tr>
<tr>
<td></td>
<td>• files completed</td>
</tr>
<tr>
<td></td>
<td>• files failed</td>
</tr>
<tr>
<td></td>
<td>• started at</td>
</tr>
<tr>
<td></td>
<td>• stopped at</td>
</tr>
<tr>
<td></td>
<td>• status</td>
</tr>
<tr>
<td></td>
<td>• error description</td>
</tr>
<tr>
<td></td>
<td>• source address</td>
</tr>
<tr>
<td></td>
<td>• destination address</td>
</tr>
</tbody>
</table>
### Field | Description
---|---
| **Note:** When you select a field, its definition will appear in the box below.
| Sort | Select the following fields to sort data inside your groups:
• Sort your contact groups by contact name.
• Sort your transfer groups by the time that the transfer started.
• Sort your file groups by file name.
Select ascending order for all fields.
| Filters | To narrow down the report so that only Faspex Users are displayed, specify the Transfer Contact field as ending with the value (faspex).

3. **Save, finalize run settings, and run your report.**

    Next, click the **Create and Run** button. Confirm the following settings on next page:
    
    • Title is as described above.
    • Report is scheduled to **Run Now**.
    • Report period is **Month to date** and time zone is **Pacific**.
    • Sorting is as described above.
    • Filter is as described above.
    • XLSX file format is checked.

    Once confirmed, click the **Run Report** button.

4. **View your Web and XLSX reports.**

    After clicking the **Run Report** button, the page updates to display the report queuing and then running. Once generated, the Web version of your basic report appears as shown below.

    ![Report: Basic Faspex User Report](image)

As you can see, the report's data is grouped and sorted in the following manner:

• Faspex Users
• Transfers (per Faspex User), which are sorted by the time they started
• File name (per Transfer)

In addition, all data columns appear as selected on the **Create Advanced Report Type** page. To download the Excel version of the report for use in other applications, click the **XLSX** button.
Basic Report Example: Hot Folder Activity

The following example demonstrates the process of creating a new, basic report (following the instructions described in the topic Creating a Basic Report on page 87) for Hot Folder transfers. In our example, we will generate a report that displays transfer activity for Hot Folders that have been set up within Aspera Enterprise (or Connect) Server, Point_to_Point and Client. The example report, once generated, will display Hot Folder transfer start time, end time and the number of files transferred.

1. Go to the Manage Report Types page

   Select Reports from the Console menu, and then click the Manage Report Types button. On the Manage Report Types screen, click the New Basic button.

2. Configure your basic report to display file- and transfer-level details, organized by Faspex Users

   On the Create New Report Type page (for basic reports), enter the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Basic Hot Folder Transfer Report</td>
</tr>
<tr>
<td>Description</td>
<td>Basic Hot Folder Transfer report, which includes start time, stop time and the number of files that were transferred.</td>
</tr>
<tr>
<td>How would you like to organize this report?</td>
<td>Select &quot;Transfer&quot; as the field by which to organize this report. In doing so, the report will be grouped the human-readable name that has been assigned to each hot</td>
</tr>
</tbody>
</table>
Field | Description
---|---
folder transfer. A transfer represents one or multiple executions of `ascp` (that is, one or multiple attempts to transfer).

Columns to include | Select the following basic fields to include as columns:
- started at
- stopped at
- files completed
- average rate

Note: When you select a field, its definition appears in the box below.

Sort | Select the "transfer name" field (in ascending order) to sort data inside your group.

Filters | In this example, we must set a filter that checks the value of the transfer cookie. When files are transferred using Hot Folders, the transfer cookie contains the following information:

`aspera.sync2:`

Thus, the filter must be set to only include transfers that have a transfer cookie starting with the value `aspera.sync2:`.

3. Save, finalize run settings, and run your report.

Next, click the **Create and Run** button. Confirm the following settings on next page:

- Title is as described above
- Report is scheduled to **Run Now**
- Report period is **Month to date** and time zone is **Pacific**
- Sorting is as described above
- Filter is as described above

Once confirmed, click the **Run Report** button.

4. View your Web report.

After clicking the **Run Report** button, the page will update to display the report queuing and then running. Once generated, the Web version of your basic report appears as shown below.

Basic Report Example: Faspex Metadata

The following example demonstrates the process of creating a new, **basic** report (following the instructions described in the topic **Creating a Basic Report** on page 87) for Faspex metadata. In our example, we will generate a report that
displays the metadata that is entered into a "Create New Package" form within Faspex, which is accomplished by creating a new, custom field called "Event" within Console.

**Note:** This example assumes that the "event" (metadata) field has already been set up on the Faspex node. When creating a new Faspex package, Faspex users can select from a predefined (drop-down) list of events, which populates the database for this custom field.

The example report, once generated, will display the purpose (or "Event") of the Faspex package, as well as file-level detail, transfer-level detail (where a **transfer** groups together transfer sessions into a single item), and which Faspex user sent the package.

1. Set up a Console database custom field for the metadata.

   Within Console, select **Configuration** from the main menu, and then the **Custom Fields** tab. Create a new, custom field with the following attributes:

   - **Level:** Select "transfer"
   - **Name:** Enter the name "event"
   - **Start Date:** Enter "2011-01-01"
   - **Description:** Since this custom field is for the metadata report, enter the description "Faspex Metadata report demo"

   For more information on custom fields, see **Creating Custom Fields** on page 91.

   Next, click the **Create** button and enter the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Built-in field</td>
<td>Faspex Metadata</td>
</tr>
<tr>
<td>Operator</td>
<td>matching regular expression</td>
</tr>
<tr>
<td>Expression</td>
<td>(.*&quot;Event&quot;:&quot;(?&lt;event&gt;[^&gt;\d]+)&quot;)</td>
</tr>
<tr>
<td>Custom Field Value</td>
<td>&lt;event&gt;</td>
</tr>
</tbody>
</table>

   In the rule example above, a rule is created that states if the conditions match the regular expression, then set the "event" custom field value to the Faspex metadata value.

   The regular expression is interpreted as follows:
The following example of decoded metadata for a Faspex cookie shows what the regular expression matches:

\{.*"Event":.*\}\n
- Find a left curly bracket character
- followed by 0 or more characters that don't match "Event":.* (this is text to ignore, such as other metadata fields)
- "Event":.* — followed by the exact text "Event":.*
- followed by at least 1 character that is not a quotation mark (save these characters and call them event).
- followed by a quotation mark
- followed by 0 or more characters that don't match a right curly bracket \}

When finished, click Create to create the new rule.

On the next page, click the Back to Custom Fields tab or the Custom Fields tab. Locate the entry for the field you just created ("event" in this case), and click recalculate.

2. Go to the Manage Report Types page

Select Reports from the Console menu, and then click the Manage Report Types button. On the Manage Report Types screen, click the New Basic button.

3. Configure your basic report to display contact, file-level, and transfer-level details, organized by Faspex metadata (the "event").

On the Create New Report Type page (for basic reports), enter the following information:

![Create New Basic Report Type form](image-url)
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Faspex meta data report</td>
</tr>
<tr>
<td>Description</td>
<td>Based on the custom field &quot;event.&quot; Includes metadata, contact, file-level, and transfer-level details.</td>
</tr>
</tbody>
</table>
| How would you like to organize this report? | Select "Event" (which is a custom field), "Contact," "Transfer" and "File" as the fields by which to organize this report. In doing so, the report will be grouped by the following:  
  - **Event** (Based on a transfer-level rule that states if the conditions match the regular expression, then set the "event" custom field value to the Faspex metadata value.)  
  - **Contact** (Contact assigned by Console. This can be a Console user name, a Faspex Server user name, SSH account, or customized value obtained from a transfer cookie. Examples: "admin (console)", "aspera (ssh)", "michael (faspex)").  
  - **Transfer** (Human-readable name assigned to a transfer. A transfer represents one or multiple executions of ascp (i.e., one or multiple attempts to transfer).)  
  - **File** (File's name)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Columns to include       | Select the following basic fields to include as columns:  
  - started at  
  - stopped at  
  - bytes transferred  
  - status  
  - average rate  
  - cookie  
  - Note: When you select a field, its definition will appear in the box below.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Sort                     | Select the following fields to sort data inside your groups:  
  - Sort your metadata groups by event/metadata name  
  - Sort your contact groups by contact name  
  - Sort your transfer groups by transfer name  
  - Sort your file groups by file name  
  - Select ascending order for all fields.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Filters                  | Filter the report so that only fields with metadata appear (that is, event is not NULL) and only data from Faspex Users is displayed (that is, transfer contact contains the value faspex).                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
After clicking the **Run Report** button, the page will update to display the report queuing and then running. Once generated, the Web version of your basic report will appear as shown below.

As you can see, the report's data is grouped and sorted in the following manner:

- Metadata
- Faspex Users that selected the corresponding event/metadata
- Transfers (per Faspex User), which are sorted by the time they started
- File name (per Transfer)

In addition, all data columns appear as selected on the **Create Basic Report Type** page.

**Advanced Report Example: Transfer Sessions with High Packet Loss**

The following example demonstrates the process of creating a new, **advanced** report (following the instructions described in the topic **Creating an Advanced Report** on page 88) for transfers with high packet loss. In our example, we will generate a report that displays a list of all transfers that have high packet loss, where high loss is user specified. The report includes transfers that started before the report period start, as well as ones that ended after the report period end, as long as part of the transfer fell within the reporting period. Note that the data is not prorated, meaning that the "bytes transferred," "files complete" and other values show totals for the entire transfer, even if part of the transfer took place outside the reporting period.

1. Go to the **Manage Report Types** page.
   Select **Reports** from the Console menu, and then click the **Manage Report Types** button. On the **Manage Report Types** screen, click the **New Advanced** button.

2. Input your advanced report's name and description.
   On the **Create New Advanced Report Type** page, enter the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Transfer Sessions with High Packet loss</td>
</tr>
<tr>
<td>Description</td>
<td>Displays a list of all transfers that have high packet loss.</td>
</tr>
</tbody>
</table>

3. Create your SQL script.
   **Important:** For assistance on SQL variables and a fields reference guide, please click the **Help** link.
CREATE TABLE $FINAL_RESULT_TABLE

SELECT DISTINCT -- prevents duplicate rows (that is, overlapping permissions)
  ts.name,
  ts.contact,
  ts.bytes_transferred,
  ts.bytes_lost,
  TRUNCATE((ts.bytes_lost)*100/(ts.bytes_transferred + ts.bytes_lost),
  1) AS `packet loss %`,
  ts.source_ip AS `from`,
  ts.dest_ip AS `to`,
  ts.started_at,
  ts.stopped_at,
  ts.status,
  ts.files_complete,
  ts.files_failed,
  ts.files_skipped
 FROM
 $TBL_TRANSFER_SESSIONS ts

WHERE
 ((ts.bytes_lost * 100) / (ts.bytes_lost + ts.bytes_transferred)) >= $PACKET_LOSS /* Custom/configurable variable */
 AND
  ts.started_at < '$REPORT_PERIOD_END'
 AND (
   ts.stopped_at >= '$REPORT_PERIOD_START'
 OR ts.stopped_at IS NULL
  )

ORDER BY
  5 DESC,
  8
;

Important: For demonstration purposes, we have created a configurable/custom variable called $PACKET_LOSS in the SQL script text above. You may, alternatively, utilize the built-in SQL database field avg_loss_pct, to display the average packet loss over the network (as a percentage). Please see the Help link in the application for details.

4. Save, finalize run settings and run your report.

Next, click the Create and Run button. Confirm the following settings on next page:

- Title is as described above.
- Report is scheduled to Run Now.
- Report period is Last 24 hours and time zone is Pacific.

Once confirmed, click the Run Report button.

5. View your Web report.

After clicking the Run Report button, the page will update to display the report queuing and then running. Once generated, the Web version of your basic report will appear as shown below.
 Report: Transfer Sessions with High Packet loss

As of 2013-08-12 15:10:12 (GMT-06:00) Pacific Time (US & Canada)
05/11/2013 03:10 PM - 06/12/2013 03:10 PM (PDT)
Packet loss: 10

<table>
<thead>
<tr>
<th>Name</th>
<th>Contact</th>
<th>Bytes Transferred</th>
<th>Bytes Lost</th>
<th>Packet Loss %</th>
<th>From</th>
<th>To</th>
<th>Started At</th>
<th>Stopped At</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows to Linux</td>
<td>admin (Canada)</td>
<td>194,461,656</td>
<td>18,757,624</td>
<td>11.1</td>
<td>10-00-00-00</td>
<td>10-00-25-00</td>
<td>2013-05-11</td>
<td>14:51:11</td>
<td>completed</td>
</tr>
<tr>
<td>Windows to Linux</td>
<td>admin (Canada)</td>
<td>17,799,218</td>
<td>1,686,165</td>
<td>9.41</td>
<td>10-00-21-02</td>
<td>10-00-25-00</td>
<td>2013-06-01</td>
<td>14:08:11</td>
<td>completed</td>
</tr>
<tr>
<td>Windows to Linux</td>
<td>admin (Canada)</td>
<td>35,715,328</td>
<td>2,977,348</td>
<td>8.32</td>
<td>10-00-21-02</td>
<td>10-00-25-00</td>
<td>2013-06-01</td>
<td>13:34:11</td>
<td>completed</td>
</tr>
<tr>
<td>Raw Koller</td>
<td>raw koller</td>
<td>194,461,656</td>
<td>18,757,624</td>
<td>11.1</td>
<td>10-00-00-00</td>
<td>10-00-25-00</td>
<td>2013-05-11</td>
<td>14:51:11</td>
<td>completed</td>
</tr>
</tbody>
</table>
Technical Support

Support Websites

For an overview of IBM Aspera Support services, go to http://asperasoft.com/company/support/.

To view product announcements, webinars, and knowledgebase articles, as well as access the Aspera Support Community Forum, sign into the IBM Aspera Support site at support.asperasoft.com using your email address (not your company Aspera credentials), or set up a new account. You can click on a heading then click Follow to receive notifications when new knowledgebase articles are available; if you follow RELEASE NOTES under a specific product, you will be automatically notified of new releases.

Personalized Support

You may contact an Aspera support technician 24 hours a day, 7 days a week, through the following methods, with a guaranteed 4-hour response time.

If you have an emergency, create a ticket using the Support Request Form with as many details as you have available and then call. If you are asked to leave a voice message, include the ticket number.

<table>
<thead>
<tr>
<th>Email</th>
<th><a href="mailto:support@asperasoft.com">support@asperasoft.com</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Phone (North America)</td>
<td>+1 (510) 849-2386, option 2</td>
</tr>
<tr>
<td>Phone (Europe)</td>
<td>+44 (0) 207-993-6653 option 2</td>
</tr>
<tr>
<td>Phone (Singapore)</td>
<td>+81 (0) 3-4578-9357 option 2</td>
</tr>
<tr>
<td>Support Request Form</td>
<td><a href="https://support.asperasoft.com/anonymous_requests/new/">https://support.asperasoft.com/anonymous_requests/new/</a></td>
</tr>
</tbody>
</table>
Legal Notice

© 2008-2016-2017 Aspera, Inc., an IBM Company. All rights reserved.

Licensed Materials - Property of IBM
5725-S595737-A72
US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Aspera, the Aspera logo, and FASP transfer technology are trademarks of Aspera, Inc., registered in the United States. Aspera Connect Server, Aspera Drive, Aspera Enterprise Server, Aspera Point-to-Point, Aspera Client, Aspera Connect, Aspera Cargo, Aspera Console, Aspera Orchestrator, Aspera Crypt, Aspera Shares, the Aspera Add-in for Microsoft Outlook, and Aspera Faspex are trademarks of Aspera, Inc. All other trademarks mentioned in this document are the property of their respective owners. Mention of third-party products in this document is for informational purposes only. All understandings, agreements, or warranties, if any, take place directly between the vendors and the prospective users.