



**BRIGHT**  
TECHNOLOGIES  
TECHNOLOGIES  
BRIGHT



**DM** DataMover

# DataMover 4

Manual

## SUPPORT INFORMATION

Phone (US and CAN)	+1.877.399.9002
Phone (international)	+1.775.823.9002
Helpdesk page	<a href="http://support.4bright.com">support.4bright.com</a>
E-Mail	<a href="mailto:support@4bright.com">support@4bright.com</a>

## CONTENTS

ABOUT THIS DOCUMENT .....	3
Purpose .....	3
Notice .....	3
BTI Software Warranty.....	3
DATAMOVER OVERVIEW .....	4
DATAMOVER COMMAND USAGE AND OPTION DESCRIPTIONS .....	4
DATAMOVER COMMAND EXAMPLES.....	7
Basic Examples .....	7
Advanced Examples.....	8
DATAMOVER MEDIA FILE AWARENESS .....	9
patternfilter.xml - Pattern Rules.....	10
media.txt - File Type Recognition.....	13
BRIGHT SOFTWARE FILE SEQUENCE AWARENESS .....	13
3D STEREOSCOPIC OPTIMIZER.....	14
Basic Example .....	14
Advanced Examples.....	14
APPENDIX A: TERMS AND ACRONYMS.....	15
APPENDIX B: BRIGHTCLIP-ENABLED APPLICATIONS.....	15

## ABOUT THIS DOCUMENT

### Purpose

This document provides an overview of the Bright DataMover program along with some basic and advanced usage examples.

This document is intended for use by computer system professionals who support media studios in managing production workflow data.

### Notice

The information contained in this document may change without notice, due to any cause, including software and/or hardware changes. While every effort has been made to ensure the accuracy of the information contained within, Bright Technologies, Incorporated (hereinafter referred to and acknowledged as BTI) and the author of this document accept no liability for any errors that may occur. BTI hereby expressly disclaims all warranties, express and implied, including the implied warranties of merchantability and fitness for a particular purpose. Under no circumstances shall BTI be liable for any consequential, incidental, or special damages, including lost profits, lost data, or lost goodwill, however caused. The information in this document is protected by copyright: Copyright ©2006-2012 Bright Technologies, Incorporated. All Rights Reserved. Except as specifically permitted herein, no portion of the information in this document may be reproduced in any form or by any means without prior written permission of BTI.

BrightClip is a registered trademark of Bright Technologies, Inc. STORNEXT is a registered trademark of Quantum Corporation. HyperFS is a trademark of FalconStor Software, Inc. All other brand and product names are trademarks or registered trademarks of their respective owners.

### BTI Software Warranty

LIMITED WARRANTY: BRIGHT TECHNOLOGIES WARRANTS THAT THIS SOFTWARE WILL OPERATE ON A CONTEMPORANEOUSLY PURCHASED BRIGHTDRIVE SERVER OR ANOTHER CONFORMING SYSTEM AS DESCRIBED IN THIS MANUAL (HEREINAFTER THE EQUIPMENT). BRIGHT TECHNOLOGIES WARRANTS THAT WHEN OPERATING ON THE EQUIPMENT, THE SOFTWARE SHALL BE CAPABLE OF PERFORMING IN ACCORDANCE WITH THE SPECIFICATIONS AND PARAMETERS SET OUT IN THIS MANUAL FOR A PERIOD OF TEN (10) MONTHS AFTER PURCHASE. THESE WARRANTIES DO NOT INCLUDE ANY REPRESENTATION OR WARRANTY THAT THE SOFTWARE IS FREE FROM "BUGS" AND OTHER MINOR DEFECTS, BUT RATHER ONLY THAT BUGS AND MINOR DEFECTS SHALL NOT MATERIALLY PREVENT THE USE OF THE SOFTWARE. BRIGHT TECHNOLOGIES SHALL HAVE NO OBLIGATION IN RESPECT OF ANY BREACH OF THESE WARRANTIES UNLESS THE CUSTOMER NOTIFIES BRIGHT TECHNOLOGIES OF THE BREACH WITHIN TWO (2) BUSINESS DAYS OF THE DISCOVERY OF ANY DEFECT IN THE SOFTWARE. NO BREACH OF THESE WARRANTIES SHALL BE DEEMED TO OCCUR IF THERE IS: (1) A FAILURE OF THE EQUIPMENT; (2) A FAILURE TO COMPLY WITH THE INSTRUCTIONS IN THIS MANUAL; (3) INACCURATE INPUT; (4) USE OF THE SOFTWARE OR EQUIPMENT BY A PERSON NOT SUFFICIENTLY QUALIFIED OR TRAINED; (5) ALTERATION TO THE EQUIPMENT OR SOFTWARE NOT APPROVED BY BRIGHT TECHNOLOGIES. NO OTHER WARRANTIES, EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE, ARE PROVIDED WITH THIS PRODUCT, INCLUDING WARRANTIES OF CONDITION, QUALITY, PERFORMANCE, DURABILITY, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. BRIGHT TECHNOLOGIES' AGGREGATE LIABILITY IN RESPECT OF ANY CAUSE OF ACTION RELATED TO THIS WARRANTY OR THE SOFTWARE SHALL NOT EXCEED AN AMOUNT EQUAL TO THE FEE PAID TO BRIGHT TECHNOLOGIES FOR THE SOFTWARE REDUCED BY TWENTY PERCENT (20%) FOR EVERY TWO (2) MONTHS OF OPERATION PRIOR TO THE CAUSE OF ACTION ARISING. ALTERNATIVELY, AT BRIGHT TECHNOLOGIES OPTION, FOR ANY LIABILITY IT INCURS, IT MAY SATISFY SUCH LIABILITY BY REPLACING DEFECTIVE SOFTWARE WITH SOFTWARE OF SIMILAR CAPABILITY AND PERFORMANCE. UNDER NO CIRCUMSTANCES SHALL BRIGHT TECHNOLOGIES BE LIABLE FOR THE COST OF LABOR FOR THE UNINSTALLATION OF DEFECTIVE SOFTWARE, THE INSTALLATION OF REPLACEMENT SOFTWARE OR FOR ANY COSTS RELATED TO DATA RECOVERY. BRIGHT TECHNOLOGIES SHALL NOT BE LIABLE FOR ANY CONSEQUENTIAL, PROXIMATE, INCIDENTAL OR SPECIAL DAMAGES RESULTING FROM A DEFECT OR FAILURE OF THIS PRODUCT. SPECIFICALLY, BRIGHT TECHNOLOGIES SHALL NOT BE LIABLE FOR ANY DAMAGES OR RESULTS OF LOST DATA, LOST PROFITS OR LOST GOODWILL. UNLESS EXPRESSLY STATED THEREIN, ANY SUPPORT CONTRACTS PURCHASED FROM BRIGHT TECHNOLOGIES RELATED TO THIS SOFTWARE DO NOT INCLUDE AN OBLIGATION BY BRIGHT TECHNOLOGIES TO REPAIR OR REPLACE DEFECTIVE SOFTWARE OR GUARANTY THAT THE SOFTWARE WILL BE BACKWARDS OR FORWARDS COMPATIBLE WITH OTHER PRODUCTS, EQUIPMENT OR SYSTEMS.

## DATAMOVER OVERVIEW

Bright DataMover (BDM) is a high-performance media file transfer application designed to take full advantage of available hardware resources in shared media environments. Use BDM to efficiently copy or move large amounts of data to and from high bandwidth storage systems, such as the BrightDrive SAN Server for Media, or other systems using StorNext® or HyperFS™ volumes. BDM includes support for allocation sessions.


BDM can be used with local storage, networked storage, or with a workspace. A workspace refers to shared storage space accessed through a high-speed storage area network (SAN), where a name-space or volume provided by the SAN is mounted on client workstations.


BDM includes media management features such as file sequence awareness and a 3D Stereoscopic Optimizer merge function. In addition, BrightClip® functionality is invoked when BDM is used on a BrightClip workspace.

## DATAMOVER COMMAND USAGE AND OPTION DESCRIPTIONS

**USAGE:** bdm [ -<flag> [<val>] | --<name>[ {=| }<val>] ]... source... dest

<i>Flg</i>	<i>Arg</i>	<i>Option-Name</i>	<i>Description</i>
-L	Str	--license	Path to license file.
-v		--verbose	Display additional information about the transfer process and result; use when additional detail about the status of the transfer is required. Will display additional information on the terminal/console as well as capture the same information in the BDM-generated log file. Commonly used to capture a record of files transferred as well as a troubleshooting tool.
-q		--quiet	Display minimum of progress information for the file transfer process.
Copy options:			
-r		--recurse	Recursively transfer a directory tree - include all sub-directories of the source. Use when there are many subdirectories to be transferred along with the target files. Disabled as --dont-recurse (default).
-u		--update	Transfer files to destination only if (a) the file exists in the destination and the source file is newer (based on last modified time) or (b) the destination file doesn't exist. Use to make certain that every file in the source file set exists in the destination file set, including source files that have changed since the last time the source files were copied to the destination. Disabled as --dont-update (default).

<i>Flg</i>	<i>Arg</i>	<i>Option-Name</i>	<i>Description</i>
-m		--move	Move files from the source directory to the destination directory - the source files are deleted after a successful copy. Unlike the similar Linux command `mv`, this option still copies the files to the destination, even if the destination is on the same storage space. Disabled as --dont-move (default).
-o		--overwrite	If files with the same name as source files exist in the destination, overwrite the destination files <b>without</b> asking the user for permission. Use to copy over or replace files in a file set <b>without a warning prompt</b> for each file, such as within a controlled usage script. Disabled as --dont-overwrite (default).
-i		--interactive	Provides ability to dynamically modify the data transfer rate during program operation. The +/- keys will change the target transfer rate by increments of 10 MB/s. Disabled as --not-interactive (default).
		--continue-on-error	Continue a copy even if errors occur. Errors that occur will be captured in the log file. Disabled as --dont-continue-on-error (default) - transfers are stopped when an error occurs.
			 <b>WARNING:</b> Use this option with caution and review the copy results, as indiscriminate use may lead to data loss due to copy errors.
-p		--preserve	Preserve file attributes if possible. Disabled as --dont-preserve (default).
		--verify	Verify transfer operation upon completion. Use for an extra level of data integrity verification if desired. Forces a re-read of each destination file for direct comparison to the source file. Disabled as --dont-verify (default).
		--contig	Write destination files as optimally as possible for best playback performance (default). This feature provides complete optimal file arrangement when the destination directory is on a BrightDrive BrightClip workspace. However, BDM (with this option as default) will provide the most optimal file arrangement possible with any workspace. In the event that this is not desired (highly churned destination space, archival purposes, absolute shortest elapsed copy time required), it can be disabled as --no-contig.


**NOTE:** --no-contig should be used carefully, since the playback performance of the copied files is not likely to meet real-time standards, even if the underlying storage supports sufficient speed.

<i>Flg</i>	<i>Arg</i>	<i>Option-Name</i>	<i>Description</i>
		<code>--clobber</code>	Respond affirmative if volume needs to be cleaned up - only used to automatically clean up temporary files left from an abnormal termination of a BDM run (for example, if the workstation running BDM suffers a power failure).
		<code>--sequence</code>	Enable sequence awareness (default) - parse the files being transferred to identify associated image sequences for optimal placement on destination storage space. Disabled as <code>--no-sequence</code> .
		<code>--merge-3d</code>	3D Stereoscopic Optimizer - merges source left-eye and right-eye image sequences to create a contiguous data arrangement of the stereoscopic image files in a separate directory. Provides maximum storage performance for real-time stereo playback. Prohibits the <code>--recurse</code> option. Disabled as <code>--no-merge-3d</code> (default).
<b>Options controlling performance:</b>			
<code>-d</code>		<code>--directio</code>	Attempt to use direct I/O (default). Disabled as <code>--no-directio</code> .
<code>-b</code>	Str	<code>--buffersize</code>	Specifies the buffer size in bytes used for I/O (default = 4 MB). Can add 'm' to the option string to indicate megabytes.
<code>-n</code>	Str	<code>--numbuffers</code>	Specifies the number of buffers to use (default = 4).
<code>-t</code>	Str	<code>--numthreads</code>	Specifies the number of worker threads to use (default = 4).
<code>-M</code>	Str	<code>--maxbps</code>	Specifies maximum transfer rate in bytes per second. Can add 'k' or 'm' to the option string to indicate kilobytes or megabytes. Use to reduce the effect on read performance while transferring files to a given volume. Elapsed time will increase when BDM performance is throttled.
	Str	<code>--policy</code>	Worker thread scheduling policy.
	Str	<code>--priority</code>	Worker thread priority.
<b>Options controlling logging:</b>			
<code>-O</code>	Str	<code>--outputstats</code>	Output block transfer statistics, for debugging purposes.
	Str	<code>--performancstats</code>	Log performance statistics, for debugging purposes.
<b>Version and help options:</b>			
<code>-V</code>	opt	<code>--version</code>	Display version information.
<code>-h</code>		<code>--help</code>	Display help information.
<code>-!</code>		<code>--more-help</code>	Display help information with page scrolling feature.

<i>Flg</i>	<i>Arg</i>	<i>Option-Name</i>	<i>Description</i>
->	opt	--save-opts	Save current options to a configuration file for later use. The full path of the file to be saved (with name) must be specified.
--	Str	--load-opts	Load options from a configuration file. The full path of the file (with name) must be specified. Disabled as --no-load-opts; may appear multiple times on the command line.

To view option preset mechanisms for a specific client operating system, view the command usage information on the client using the "-h" option (the mechanisms are shown toward the end).

## DATAMOVER COMMAND EXAMPLES

Examples of using BDM are shown below. In the examples, the source and destination directories could be local storage space, portable storage devices, or workspaces.

### Basic Examples

```
$ bdm <source> <destination>
```

Optimized copy. The most common use of BDM is to copy source data onto a workspace in a manner suitable for high-performance playback without having to perform additional optimization afterward.

```
$ bdm --no-contig <source> <destination>
```

Basic copy. Here, BDM is used without the attempt of contiguous file layout on the storage. Although the sequence awareness feature is still invoked by default, the data transfer will be subject to the client file system data arrangement capability. Useful when the condition of the target workspace won't allow "proper" playback at appropriate speeds, but the data still needs to be copied to the target.



**NOTE:** Playback performance of files copied with --no-contig is completely dependent on the state of the target workspace when the copy is performed. In many cases, these files will require optimization to obtain real-time playback (assuming the target workspace supports sufficient speeds in the first place).

```
$ bdm -p /mnt/prjsource/flight_day1.mpg /mnt/ws03/demos/
```

Optimized copy of a large media file to a directory on workspace "ws03", with option to preserve file attributes.

```
$ bdm -rp /mnt/prjsource/prj_mverse1/ /mnt/ws03/
```

Optimized copy of a project directory onto workspace "ws03", with options for recursive data transfer (for the source directory tree) and preserve file attributes.

## Advanced Examples

```
$ bdm -rp -b 13m /mnt/drivefx/flight_day1/ /mnt/ws03/
```

Optimized copy of source files to a project directory on workspace “ws03”, with options for recursive data transfer, preserve file attributes, and buffer size set to 13 megabytes.



**NOTE:** When transferring a large sequence of files, all of which being the same resolution or file size, use a buffer size slightly larger than a single file size for best transfer rate. The example above would be applicable for 2K image files (at a size of 12.16 MB per file).

```
$ bdm -rp --verify -b 13m /mnt/drivefx/flight_day1/ /mnt/ws03/
```

Optimized copy - same as previous example, along with forced verification of transferred files against source files.

```
$ bdm -rp -M 50m /mnt/prj5/flight_day2/ /mnt/ws03/fltd2/
```

Optimized copy of source files to a project directory on workspace “ws03”, with options for recursive data transfer, preserve file attributes, and maximum transfer rate of 50 megabytes/second. This is useful for limiting impact on high-performance storage currently being accessed by other users or applications.



## DATAMOVER MEDIA FILE AWARENESS

BDM includes a set of media file filters to help optimize workflow performance with a StorNext® file system workspace. These filters define the characteristics of files to be stored onto the workspace. In this way, relevant “media” files will be available using the maximum performance capabilities of the workspace storage.



**NOTE:** When BDM is used with a BrightClip workspace, the BrightClip media file awareness is invoked.

The filter rules are contained in two filter files on a client workstation:

- `patternfilter.xml` - Contains pattern rules based on file attributes.
- `media.txt` - Defines file type recognition.

The filter files are located in the directory `/usr/bti/config/` (for Linux and Mac clients) or in `%ProgramData%\bti\config\` or `%ProgramFiles%\bti\config\` (for Windows clients).

The default filter files contain settings for recognizing standard media file types and attributes, and are appropriate for most environments. Using a text editor, the filter files can be modified on a client workstation to recognize file characteristics for a specific application or non-standard media file type in order to achieve a customized optimum workflow. Changes to the filter files are not data destructive. However, if not done properly, there could be a significant reduction in data transfer performance.



**NOTES:**

1. Independent copies of the filter files reside on each client workstation (filtering is not centric or server-based). Any changes should be applied to the filter files on all clients to maintain media file awareness compatibility on shared workspaces.
2. To reset the filter files to default, add “.bak” to the end of the existing file names (to keep them available for reference). Upon the next run of BDM, default filter files will be re-created.

The filter rules for BDM media file awareness are processed in the following manner:

1. The file or sequence is tested against the pattern rules in the `patternfilter.xml` filter file. In the global section of this filter file (described on page 10), the rules are processed from top to bottom.
  - The moment a file or sequence is deemed as “included” by a pattern rule match, all filter testing stops and the file or sequence is recognized as media.
  - The moment a file or sequence is deemed as “excluded” by a pattern rule match, all filter testing stops and the file or sequence is treated as generic .
  - If neither of the above actions occurs (meaning the file or sequence has not been specifically “included” or “excluded”), it is then tested for file type recognition based on the contents of the `media.txt` filter file (described in the next step).

2. If the file or sequence is not “included” or “excluded” as described in step 1, then it is tested for file type recognition. The extension of the file or sequence name is compared to the contents of the `media.txt` filter file.
  - If a match is found, the file or sequence is recognized as media.
  - If a match is not found, the file or sequence is treated as generic.

## patternfilter.xml - Pattern Rules

The `patternfilter.xml` file contains pattern rules for including or excluding certain content from being recognized as media. The pattern rule format provides the ability to create filters based on detailed file attributes beyond just the file name extension (as done with the `media.txt` filter file).

Shown below is the “global” section of the default `patternfilter.xml` file:

LINE NUMBER	LINE CONTENTS
7	<code>&lt;global&gt;</code>
8	<code>&lt;patterns&gt;</code>
9	<code>&lt;pattern name="" path="*/.*/" applyto="all" action="exclude"&gt;</code>
10	<code>&lt;size&gt;</code>
11	<code>&lt;![CDATA[]]&gt;</code>
12	<code>&lt;/size&gt;</code>
13	<code>&lt;/pattern&gt;</code>
14	<code>&lt;pattern name="." path="" applyto="all" action="exclude"&gt;</code>
15	<code>&lt;size&gt;</code>
16	<code>&lt;![CDATA[]]&gt;</code>
17	<code>&lt;/size&gt;</code>
18	<code>&lt;/pattern&gt;</code>
19	<code>&lt;pattern name="*" path="*" applyto="sequence" action="exclude"&gt;</code>
20	<code>&lt;size&gt;</code>
21	<code>&lt;![CDATA[&lt;100M]]&gt;</code>
22	<code>&lt;/size&gt;</code>
23	<code>&lt;/pattern&gt;</code>
24	<code>&lt;pattern name="*" path="*" applyto="nonsequence" action="exclude"&gt;</code>
25	<code>&lt;size&gt;</code>
26	<code>&lt;![CDATA[&lt;100M]]&gt;</code>
27	<code>&lt;/size&gt;</code>
28	<code>&lt;/pattern&gt;</code>
29	<code>&lt;/patterns&gt;</code>
30	<code>&lt;/global&gt;</code>

Notice that there are four pattern rules contained in this “global” section of the default `patternfilter.xml` file:

1. Lines 9-13 - Exclude hidden directories (for directory names beginning with “.”);
2. Lines 14-18 - Exclude hidden files (for file names beginning with “.”);
3. Lines 19-23 - Exclude file sequences less than 100 megabytes;
4. Lines 24-28 - Exclude general files (not part of a sequence) less than 100 megabytes.

The “global” section of the `patternfilter.xml` file is the area that can be modified to meet specific workflow needs.



**NOTE:** The pattern rules in the “global” section are processed from top to bottom. The first pattern rule that results in a match with the attributes of a target file or sequence is invoked according to the action specified, and subsequent pattern rules are not processed. Therefore, if modifying this filter file, one must give great consideration to the logic inherent in the ordering of the pattern rules.

**To add** a pattern rule to the `patternfilter.xml` file, start by deciding where the new rule will fit logically among the existing rules, keeping in mind that the pattern rules are processed from top to bottom.

Next, add the following lines to the filter file at the decided location:

```
<pattern name="" path="" applyto="" action="">
  <size>
    <![CDATA[ ]]>
  </size>
</pattern>
```

Finally, edit the variable settings to specify the desired pattern rule. Here is a list of valid settings for each variable:

VARIABLE	DESCRIPTION	VALID SETTINGS
name	File name pattern. NOT case sensitive.	Any valid file name designation, including the “*” wildcard. <i>or</i> Blank designation, " " (no space in between). In this case, the file name is not part of the pattern rule logic.
path	Directory path name pattern. NOT case sensitive.	Any valid directory path designation, including the “*” wildcard. Use the forward slash, “/”, for directory level separators. <i>or</i> Blank designation, " " (no space in between). In this case, the file directory path is not part of the pattern rule logic.
applyto	Pattern logic for files vs. sequences.	all nonsequence sequence
action	Pattern logic for specifying the rule as exclusive or inclusive.	exclude include none
CDATA	Pattern logic for specifying file size condition (in bytes). Ensure that there are no spaces within the setting designation.	[ <NUMBER ] [ >NUMBER ] [ ]  <i>NUMBER format: n{K M G T}</i> <i>(n - integer, in bytes; a multiplier is optional with integer)</i> <i>Examples: [ &lt;60M ]</i> <i>[ &gt;1G ]</i>

**To edit** any of the default pattern rules, edit the variable settings as described above to specify the desired pattern rule. Again, keep in mind that the pattern rules are processed from top to bottom; moving a rule to a different location in the list may be logically necessary when editing it.

**To remove** any of the default pattern rules, delete the corresponding lines from the filter file.

### EXAMPLES:

To include “.avi” files larger than 40 megabytes while maintaining the exclusion of hidden directories or files, place this rule in the default filter file beginning at line 19:

```
<pattern name="*.avi" path="" applyto="nonsequence" action="include">
  <size>
    <![CDATA[>40M]]>
  </size>
</pattern>
```

Assume an application is copying or moving extra, non-critical files and directories, such as a large sequence of thumbnail images in a sub-directory named “128x96”, where the contents are being recognized as media by BrightClip. To exclude such a directory, place this rule in the default filter file beginning at line 9:

```
<pattern name="" path="*/128x96/*" applyto="all" action="exclude">
  <size>
    <![CDATA[]]>
  </size>
</pattern>
```

Here is an example of including a sequence file type not listed in the `media.txt` filter file, such as “.tcs”, with additional attributes of the file path including a specific directory name and a sequence size greater than 400 megabytes. This rule could be placed in the default filter file beginning at line 9:

```
<pattern name="*.tcs" path="*/ahpprj/*" applyto="sequence" action="include">
  <size>
    <![CDATA[>400M]]>
  </size>
</pattern>
```

## media.txt - File Type Recognition

The `media.txt` file is simple to understand and modify. It contains a list of file name extensions (not case sensitive) for file type recognition. Here is a sample of the default file contents:

```
# List Version 1.0
*.dpx
*.tga
*.tif
...
```

Other file type extensions can be added to the file, or an existing extension can be commented out by adding the hash symbol (#) at the beginning of a line (recommended method instead of deleting a line).

## BRIGHT SOFTWARE FILE SEQUENCE AWARENESS

DataMover is one of Bright Technologies' high-performance, sequence-aware media file management applications. By using multiple threads reading and writing multiple buffers, it can fully utilize the available hardware resources and improve throughput over standard utilities. Copy or move large sets of file sequences with ease - just specify the directory locations, and the transfer of any sequences contained therein is handled automatically, no matter the sequence size.

When a file transfer is started, the source arguments are analyzed. If any arguments are directories, the contents are sorted by name and any sequences are detected.

A sequence is defined as consecutive files which have names in the format of `<prefix>[n+]<suffix>` where `[n+]` is a variable length number of digits. Files which are selected as belonging to the same sequence will have the same prefix, suffix, and number of sequence digits. The file sequence number must be contiguous with previous files in the sequence, and it must also be the last numeric field in the file name. For example, the following two files would be detected as belonging to the same sequence:

```
mverse-day17-000500.dpx
mverse-day17-000501.dpx
```

Alternatively, these two files would not be considered as belonging to the same sequence:

```
scene8-000500-2011-09-25.dpx
scene8-000501-2011-09-25.dpx
```

Once any sequences are detected, the source files are opened for transfer access (copy or move as dictated by the command options), and corresponding destination files along with any prerequisite directories are created. The program then iterates over the source files, transferring the data to the destination files. The read and write operations are performed in sequence - all the data is read from the first file before beginning on the second file. The destination files are written in the same order that the source files are read, providing contiguous allocation on the storage space as much as possible.

## 3D STEREOSCOPIC OPTIMIZER

The BDM “--merge-3d” option provides a powerful method for creating a contiguous sequence of 3D image files on your high-performance storage space. Just specify the “--merge-3d” option with the right-eye and left-eye source directories to be copied, the destination directory to hold the merged data, and run it. The 3D files in the merge directory will be optimally stored and available for review by any real-time stereo 3D media player application.

By copying the left-eye and right-eye 3D source files to a directory space where the image files are stored in a contiguous stereoscopic order, the storage access time is drastically reduced - unnecessary disk drive head seeks are avoided. This eliminates the need for two workspaces or volumes typically required for 3D workflows.



**NOTE:** The client operating system will display the merge directory contents in alphanumeric order. The client is not able to decipher the actual layout order of files on the file system storage. Therefore, it is recommended that the word “merged” (or similar notation of choice) be used as part of the merge directory name.

### Basic Example

```
$ bdm --merge-3d /mnt/drive1/prj1-r-eye/ /mnt/drive2/prj1-l-eye/  
/mnt/ws02/prj1-merged/
```

Optimized merge of two directories containing 3D source image files into a merge directory on workspace “ws02”.

### Advanced Examples

```
$ bdm --merge-3d -b 13m /mnt/drv1/mverse-r-eye/ /mnt/drv2/mverse-l-eye/  
/mnt/ws02/mverse-merged/
```

Optimized merge of two 3D source directories into a merge directory on workspace “ws02”, with option for using a buffer size of 13 megabytes (best performance option for 2K resolution image files).

```
$ bdm --merge-3d -M 10m /mnt/drvfield/hcptr_r_eye/  
/mnt/drvfield/hcptr_l_eye/ /mnt/ws02/hcptr_merged/
```

Optimized merge of two 3D source directories into a merge directory on workspace “ws02”, with option for setting the maximum transfer rate to 10 megabytes/second (minimize performance impact on a shared workspace).

## APPENDIX A: TERMS AND ACRONYMS

BDM - Bright DataMover.

BDO - Bright DataOptimizer.

BDW - Bright DataWrangler.

BRLM - Bright License Manager utility.

BTI - Bright Technologies, Inc.

LUNs - Logical unit numbers; typically refers to partitions or logical volumes on storage devices.

MDC - Metadata controller in a SAN environment.

RAID - Redundant array of independent disks.

SAN - Storage area network.

Workspace - shared storage space accessed through a high-speed storage area network (SAN), where a name-space or volume provided by the SAN is mounted on client workstations.

## APPENDIX B: BRIGHTCLIP-ENABLED APPLICATIONS

- ARRIsan  
*by ARRI*
- Bones,  
Bones Dailies  
*by Digital Film Technology*
- DaVinci Resolve,  
DaVinci Revival  
*by Blackmagic Design*
- Director  
*by Lasergraphics*
- diTTo  
*by Cintel International*
- Mistika  
*by SGO*
- Typhon  
*by Drastic Technologies*