

## 1. Warning: Incompatible RAW Mode

**Jaleo Version 2.5 uses a new raw disk storage mode.** You must convert existing raw disk directories/headers before you can use Jaleo 2.5 on an existing raw partition. If you do not do a conversion, **your existing content will be lost.** We do strongly recommend to make sure that you have backup copies of the data on the raw device, or that all data on the device can easily be recaptured. See “Raw Disk Directory Conversion”, on page 22, for more information.

This change does only affect you if you are using a raw disk partition; some Jaleo Composite setups do not have a raw partition set up. **Normal file system storage is not affected by the modified raw mode.**

Be careful not to run Jaleo 2.5 with a raw disk device written with Jaleo 2.1 if there is anything of importance on the raw drives – Jaleo 2.5 will not use the existing raw disk directory, but create its own, empty one, unless you convert the existing directory information. Here is why:

The new release features new raw disk management. Improvements include an optimization of raw mode access speed for both reading and writing and a new content manager system for improved operational security and maintenance. Old raw disk directory files (JALEO-ENV/etc/devices/raw\_info/RAW\_HEAD) are not compatible with the new management software.

The new raw mode functions and recovery tools are described in a later section of these release notes.

**Again, using Jaleo 2.5 with an existing raw volume as used by Jaleo 2.1 will overwrite the raw volume content. Jaleo 2.5 can not work with existing data on raw devices without prior directory conversion.** While Jaleo 2.1 and 2.5 installations can principally coexist on the same machine, they can not share the same raw disk volume (and not the same license manager). If you want to run both Jaleo 2.1 and 2.5 on the same machine, each using a raw disk, you would have to setup a separate raw disk for each of them.

Raw disk directory conversion is described in detail in section “New Raw Disk Management”, page 15.

## 2. Additional Installation Information

### Warning:

**Be careful when installing Jaleo 2.5 on a system with a existing Jaleo installation. The raw disk handling of version 2.5 is not compatible without conversion. Improper installation may result in raw disk data loss.**

See “New Raw Disk Management”, on page 15, for more information.

### 2.1 Prerequisites and Requirements

Jaleo 2.5 requires Irix 5.3. Also it does not run on R3000 based machines any more.

To use all the functionality regarding movie creation, in particular with audio, the `dmedia_eoe` and `dmedia_tools` software subsets, not older than version 5.5, must be installed on your system. You can use the `showprods -D 1` typed in a shell command to see the installed subsets, or you can use the graphical `swmgr` application. The subsets can be found on your system CD (it is part of the normal SGI system distribution).

**Note to users of older systems:** Older IRIX 5.3 system CDs do *not* contain all the tools required for Jaleo operation as described above. You must have `dmedia_eoe` with a version number of at least 5.5. If you have an older version, please contact SGI on how to update your system software.

**Hint:** Newer distribution media display the version number in `inst`, older ones do not.

#### 2.1.1 Jaleo for Impact 2.5

Jaleo for Impact 2.5 *requires Impact Video software version not older than 2.4*. This does only apply to systems with Ciprico and Impact Video; it does not affect Jaleo Composite running on Impact.

On older versions, we can not guarantee proper operation. If you have older versions, please contact your SGI office for help.

Hint: You can use `showprods -D 1 -d` to find out the date a software subsystem was build.

Note that the `-d` option is essential here, because otherwise the installation date is shown, instead of the date when the software was build by SGI. Impact software version 2.4 is shipped under the title Impact Digital Media 1.1 and has a copy data of December

11, 1995. Older software, at least for the `dmedia_eoe`, `impactvideo` and `impactdm` subsystems will most probably not work.

Furthermore, you *must have* operating system **Patch 1112** from SGI. Without this patch, capture and playout of video and audio in the best case will be out of sync, in the worst case will crash.

## 2.2 Update Information

While following the installation script, make sure that you do install the new license server software – Jaleo 2.5 will not work with the existing one.

### 2.2.1 Running Jaleo 2.1 and 2.5 on the same machine

Also, if you wish to keep a copy of Jaleo 2.1 running on your machine, make sure that you copy the directory `/usr/lib/elm/jaleo` to a safe place before you proceed – a license server installation for the new version will overwrite these files.

To run the old version after a successful installation of Jaleo 2.5, you have to do the following:

- Stop the license server for version 2.5. As root, run from a shell the command  
`/etc/init.d/jaleolic stop`
- Run the old license server. The command to type depends on the location to which you copied the original 2.1 license manager and its license files. Let's assume you put it to `/usr/local/safePlace`. Then your command would look like this:  
`/usr/local/safePlace/jaleolic -s0 -e /usr/local/safePlace`
- To run the Jaleo 2.5 license manager again, you would type  
`/etc/init.d/jaleolic stop`  
`/etc/init.d/jaleolic start`

Note that using this method, after a reboot of the machine you always would return to the Jaleo 2.5 license manager.

## 2.3 Basic Installation

Installation of Jaleo 2.5 has not changed in comparison to Jaleo 2.1. The information in the 2.1 installation guide is still perfectly valid.

## 2.4 Licensing

To operate Jaleo 2.5, you will require a new license. Aside of that, the license installation and management has not changed.

There is an important change, though:

- Licenses do not depend on the sysid and IP-address of the machine, but only on the sysid. That is, you can now change the IP address of the license server machine without having to fear to invalidate your license.

Because some clients experienced problems, we would like to repeat the following advice:

- Licenses are very sensitive to date changes. Before you install a license server, or a license, make sure your system date is correctly set – in all machines to run Jaleo or the license server all over your network. The license management system will not work if the clocks of the server and a client are significantly different.
- Jaleo licenses always begin with the current date. You can not have your system date be in a past century, year or month.
- Jaleo licenses are *very* sensitive against date changes, specifically against putting back the date. You will invalidate the license server system if you put back your clock.

## 2.5 Setup for Video Operation

Proper setup of the video hardware is absolutely crucial for proper operation of real time video capture and playout. In particular, without correct synchronization of all components, you will very likely not achieve frame accuracy.

Please see “Basic Video Hardware Setup for RtVideo Operation”, on page 26, for more information.

**Cosmo Compress** users may also need to redirect temporary storage for movie creation. See “Cosmo Compress Temporary Storage Setup”, on page 12, for more information.

## 3. Setup Information

For full setup information, please see the original installation and setup guide of version 2.1. This guide gives only supplementary information.

### 3.1 Setup of Ciprico Arrays for Jaleo for Impact

Jaleo for Impact systems need a Ciprico array for real time performance. The Ciprico setup is almost identical to the setup as required for a normal raw disk. There is one additional step necessary (a driver for the disk controller must be installed), and one step is slightly modified (partitioning of the disk).

Before you can configure the Ciprico for use with Jaleo, you must install the disk controller card and the appropriate driver software that is delivered with the array. The necessary documentation for board and driver installation is part of the Ciprico package. Once the hardware is running (as can be seen from the printouts delivered upon startup of the system), you can configure the array.

As any normal disk drive, the Ciprico array must be partitioned. The only difference in comparison with a normal disk drive is that you can not use the standard SGI disk preparation tool `fx`, but you must use the utilities delivered with the Ciprico. However, this even makes the task easier, as those have a graphical interface.

- Log in as root
- To prepare the Ciprico array, run the `cfxutil` utility program. Where this can be found depends on where you located the driver at installation time. After starting the program, you must select the drive to be prepared.
- To do so, select the `Open New Array` entry from the `Functions` menu.
- On the dialogue that appears, press the `Perform Scan` button. The software will scan your machine's disk controllers in search of a Ciprico drive. Once it has finished the search, a list of arrays will be shown in the list portion of the dialogue. Typically, this will only contain a single array, and this one should be selected automatically. If not, select it by clicking on it.
- Close the dialogue window.
- Select the entry `Volume Header` from the `Function` menu.

- In the dialogue that appears, you can set up the partitioning of the array. With a new array, you should see a default partitioning that roughly looks like this (for a 16 versus 32 GB array):

Partition	Size	Type
6	8/16 GB	EFS/XFS
7	8/16 GB	EFS/XFS
8	3/3 MB	Volume Header
10	16/32 GB	Entire Disk

As you can see, this is largely equivalent to the standard layout of an SGI drive:

- the Volume Header partition, that contains housekeeping information for the drive and *always* must be present.
- the Entire Disk partition, that covers every single block on the disk, including the volume header. This partition, although always present, must *never* be used, unless for one-to-one copies of a drive to an exactly equivalent one.
- A number of file system partitions. The default for the Ciprico has two partitions for efs (the older of the SGI filesystems) usage prepared.

For use with Jaleo, you typically want to have one large data partition.

The following steps explain how to delete one of the prepared default partitions and to change the size of the other one.

- Select the `Modify Partition` button. A dialogue pops up.
- In the dialogue, you will see two scrollable lists. The left one is the list of partitions. Here, select partition 6. In the size fields, the size of the partitions will appear. Set the size and start block of partition 6 to 0 and press the `Write Change to Partition Table` button.
- Select partition 7. Make it the size of the full array minus the size of the volume header, that is, the size of the array in Megabyte minus 4. The start megabyte value *must not be smaller than 3 to prevent that the volume header is overwritten*. Set the partition type to “data” using the list box at the right. Write the change to the partition table and exit the dialogue.
- In the volume table, make sure that the new full-size partition 7 is not starting below megabyte 3. It *must not overlap* the volume header partition.
- Write the volume header back to the disk using the appropriate button of the dialogue window.

- In the main window of `cfxutil`, make a note of the driver name that is written in the window. It should be something of the form:

```
/dev/rdisk/rfc0d0s0
```

The driver path to be used in the Jaleo raw disk configuration file is not quite this one; the last number in the name must be the partition you prepared for use as a raw disk partition. If you used partition 7, as in the example above, the path would be:

```
/dev/rdisk/rfc0d0s7
```

- You must give write access for everyone to this path. In a shell window, type  
`chmod ugo+w /dev/rdisk/rfc0d0*`

Your drive is now ready for use with Jaleo.

## 3.2 Setup Additions and Changes

### 3.2.1 .jaleorc

- A new option has been added for the auto snap: `MAGNETSNAP`. Its value is the maximum snap distance in frames.

A new option controls the RotoPaint temporary storage location:

- `PAINT_TMP_DIR`. Determines the directory where RotoPaint stores temporary files. Typically, you should provide for a directory with ample space, as RotoPaint, when processing sequences, typically stores a temporary frame for each frame processed, until the new sequence is saved back to the original.

Two new options are for sound management:

- `SOUND_SAMPLE_RATE`. This determines the audio sample rate to be currently used by Jaleo. It must be specified in sample per second. Currently supported values are: 8000, 11025, 16000, 22050, 32000, 44100, 48000.
- `SOUND_SERVER_HOST`. Determines the host where to run the sound server. If a local server (the default) is desired, just comment out or delete the keyword from `.jaleorc`. If you want to disable all sound playback, put the value `NONE`. If you wish to use a different host, place the appropriate hostname of the target machine. Note that Jaleo can currently not start up the sound server process automatically on remote machines. Please see the “Additional Documentation on Jaleo 2.5” for more information.

For Jaleo for Impact systems, the following option must be set:

- `Reality`.  
The value must be `OFF`.

For Jaleo Composite, the live video output is handled via a hidden screen copy of the output image. To do so, Jaleo must track a virtual window. This tracking for technical

reasons is not always accurate. There are two options to adjust incorrect Live Video output display:

- `LIVE_OFFSET_X`
- `LIVE_OFFSET_Y`

Both receive a numeric value in pixels.

### 3.2.2 Raw.Dev, Raw Disk Configuration File

The raw disk configuration file has a new configuration entry. This entry allows to configure Jaleo for different disk array types. The new field is called `TYPE_RAWDEVICE`. It *must* be set to one of the values given below.

Possible values are:

- `TYPE_RAWDEVICE`      `normal_disk`  
This entry is used for a single standard disk drive.
- `TYPE_RAWDEVICE`      `normal_disks_logical_volume`  
This value is used for logical volumes created from a number of standard disk drives. See the installation and setup guide for more information on logical volumes.
- `TYPE_RAWDEVICE`      `ciprico_6700_logical_volume`  
For Jaleo Plus, four Ciprico 6700 arrays can be used as a real time disk array if connected to four SCSI controllers. Please contact CIC if you wish to use Jaleo PLUS with Ciprico 6700 arrays.
- `TYPE_RAWDEVICE`      `ciprico_6900`  
For Jaleo for Impact only. Indicates that a Ciprico UltraSCSI array is used.

Jaleo Composite systems do only support the two normal disk entries (`normal_disk`, `normal_disks_logical_volume`). Jaleo Plus supports the normal disk entries and the Ciprico 6700. Real time performance of course only can be achieved with appropriately configured logical volumes on multiple SCSI controllers. The only array that makes sense for Jaleo for Impact is the Ciprico UltraSCSI array. Normal disks are supported, but do not provide real time – and in this case, it is more economical to run Jaleo Composite on the machine instead.

There is also a new entry for the raw disk recovery behaviour after critical failures.

- Version 2.1B had a count configuration variable `NUM_BACKUP` to handle backup counting. This variable is not needed any more, and thus not supported.

In Jaleo 2.5, raw disk directories are handled in a much more secure way. Even if you experience a full machine crash due to power failure during a raw disk operation, you will be able to recover from the failure. That is, an operation on the raw disk is guaranteed to work either completely, or not to affect the directory and data status (content) at all. Normally, the recovery from failure is automatic. A new entry in Raw.dev allows



you to request manual recovery from failures, using the new `hadmin` tool. See “The `hadmin` Raw Management Tool”, on page 17, for more information.

- **TRANSAUTORECOVERY**  
If set to `FALSE`, you must use `hadmin` to recover from raw operation failures. Otherwise, the recovery is done automatically and silently, only noted in the raw disk log file.

### 3.2.3 RtVideo.cfg with Jaleo for Impact

The configuration file for the RtVideo (in `JALEO-ENV/etc/devices`) has additional options for operation with the Impact. These options are only used in Jaleo for Impact configurations. See the video configuration section later in this guide (“Basic Video Hardware Setup for RtVideo Operation”, page 26) for more information.

- `IMG_CATCH_CORRECTION`, for images. Default is -1.
- `SND_CATCH_CORRECTION`, for sound. Default is 0.

The parameters can be used to correct regular capture errors. That is, if you experience a regular offset of your capture results from the desired capture points, you can correct this error by setting appropriate values. In the systems we have tried, the default value worked perfectly. A reason for a capture offset can, for example, be a delay caused by an AD/DA converter.

Note: Make sure your VLAN is synchronized and supplied with a GPI trigger properly before trying to change these values.

- `IMG_SEND_CORRECTION`, for images. Default is -1.
- `SND_SND_CORRECTION`, for sound. Default is 5.

Similar to the parameters described above, these are used to correct regularly occurring offsets while laying off video. In normal cases, with a properly synchronized VLAN, the default values should work fine.

Note: Make sure your VLAN is synchronized and supplied with a GPI trigger properly before trying to change these values

- **BACKLOAD.** The backload parameter is used to determine the amount of caching used while creating the preview images after capturing a clip. The default value of 10 means that 10 images are read as a block from disk, scaled down, and then written back as preview images. This is much faster than reading and writing single images; however, it requires quite some memory, as each buffer roughly takes up 1 megabyte. For a machine with 64 MB, 10 is a sensible value. If you have a lot of memory, for example 128MB or more, you can increase the value to 20 or maybe even 30, to speed up creation of the previews.

### 3.2.4 RtVideo.cfg for Cosmo Compress: Setup of Preview Image Format

When creating compressed clips with the RtVideo, the storage format for the preview images can either be uncompressed or software compressed MVC2 movies. The format can be setup in the `RtVideo.cfg` file.

- `LOWRESOLUTION` is used to set the movie format. Use `UNCOMPRESSED` for uncompressed preview images, and `MVC2` for MVC2 compression. Note that uncompressed preview movies can be larger than the compressed high res movie if the average compression rate is larger than about 16:1.

### 3.2.5 Cosmo Compress Temporary Storage Setup

If you are using a **Cosmo Compress** board, you may need to setup the system temporary directory to a place with sufficient disk space for movie creation. The system default, `/usr/tmp`, in many cases does not have appropriate free space. The temporary directory must have space for the largest movie you are going to make, as there will be a temporary copy required.

If you need to relocate the temporary storage directory, you must set the `TMPDIR` environment variable. You should do this by adding a line like

```
setenv TMPDIR /usr/DISK_WITH_LOTS_OF_SPACE
```

to the `.cshrc` file in the Jaleo user home directory. Of course, you would replace

```
/usr/DISK_WITH_LOTS_OF_SPACE
```

with the appropriate mountpoint of a disk with, well, a lot of free space.

*You must log out and log in again* after adding the line to the `.cshrc` file.

### 3.2.6 ScrToVideo.cfg

The ScreenToVideo configuration file, used to setup the “Live Video” option for still images in Jaleo systems, has a new possible value for the `DEVICECLASS` parameter. If you set `DEVICECLASS` to `NONE`, the Live Video option is disabled. Use this value if you do not have any video device (not even a DDR) connected to your machine.

### 3.2.7 Vlan.cfg

Vlan operation can now be configured with the following options:

- The `CONNECTION` field specifies the physical connection of the VLAN device. Supported Values are:
  - `VLAN_SIRIUS`. The built-in VLAN of the sirius video.
  - `VLAN_SERIAL`. A VLAN connected to a serial port.
- The `PORT` parameter specifies the port actually used. Possible values are:

- **SIRIUS.** Only applicable if **CONNECTION** is set to **VLAN\_SIRIUS**.
- Any serial interface device for **VLAN\_SERIAL**. A typical value for Indigo systems would be `/dev/ttyd2`.
- The **NODE** parameter determines the VLAN node address to be controlled. Any number between 1 and 16 can be specified.
- The **TIMEOUT** parameter gives the interval (in seconds) after which the system assumes a timeout in the connection if no messages are received from the VLAN.

### 3.2.8 VtrAccom\*.dev

The VtrAccom configuration files have a new configuration option to correct a problem that occurs with the Accom AutoEdit function. The auto edit requires a magic timing value that influences the accuracy of edits. If this value is set incorrectly for your VTR, the result of operations with the VtrAccom device will not be accurate.

- **CORRECTION.** The default value is 780.

Accom describes the parameter, named “Edit Timing” as follows in the WSD documentation (See the chapter on WSD AutoEdit):

*“Edit Timing: Three-digit number determines the accuracy of the WSD and VTR.*

*First digit: WSD advance (range 0-9 fields). Adjusts when the WSD performs “edit on” and “edit off” functions.*

*Second digit: VTR advance (range 0-9 fields). Adjusts when the VTR performs “edit on” and “edit off” functions.*

*Third digit: Unused. Always enter 0 (zero).”*

The default value, 780, thus means 7 fields of WSD advance and 8 fields of VTR advance.

## 3.3 New Files

### 3.3.1 PlugIns

PlugIn files are located in `JALEO-ENV/lib`. Examples for developers are in the home directory of the Jaleo user, in the directory `EXAMPLES`.

PlugIn data, stored by some PlugIns that have a private user interface and store their own data, is kept in the directory `.PLUGIN_DATA` in the current project. For old projects, this directory will be created whenever you select the project in the project manager.

### **3.3.2 Raw Disk Directory Files**

Jaleo now stores raw disk information in `/usr/local/jaleoHeaders`

### **3.3.3 RotoPaint Default Cutouts**

RotoPaint default cutouts reside in `JALEO-ENV/etc/cutout`. When you create a new project, the default cutouts are copied to the `PAINT` directory of the new project.

### **3.3.4 Setup Manager Setup Files**

The setup manager stores setups in `JALEO-ENV/etc/setups`.

### **3.3.5 Colour Store Files**

The content of the “colour warehouse” of the new colour selector is kept in `JALEO-ENV/etc/colors`.

## 4. New Raw Disk Management

As described above, the raw disk handling has been changed to meet the following objectives:

- Better performance
- More reliable operation
- Easier recovery in case of failure

The new raw mode software also includes new tools for maintenance and recovery.

**Jaleo 2.5 uses raw disk devices with a different directory management software. To preserve existing raw disk content, you must convert your existing directory, or the content will be lost. Note: We recommend to have all raw data backed up before directory conversion is attempted.** See “Raw Disk Directory Conversion”, on page 22, for more information.

The new raw disk mode permits the simultaneous operation of multiple applications that access raw mode. The potential danger for the data integrity of the raw volume that did exist in previous versions of Jaleo has been eliminated by introduce a transaction processing-like management that ensures that an operation is either executed successfully and completely, or not at all. Even if you experience a fatal failure, like for example a power drop, in the mid of an operation, the system can ensure that the previous state can be recovered.

While an operation is performed, the raw directory header is not accessible to other applications.

Also, raw disk compaction is now never done automatically. Instead, the new administration tool permits to do so whenever it is convenient.

All raw disk administration can now safely be done using the administration tool. Even advanced power users must *never* try to edit the raw directory files manually. Although this has been possible with previous versions, it is now absolutely discouraged – data loss is very probable if the file is manipulated manually.

Jaleo raw disk content information is stored in various places:

- `/usr/local/jaleoHeaders`
- The last block of the raw disk volume itself

The directory in `/usr/local` contains the actual copy of the directory and a list of numbered backups. Each backup contains the state of the system previous to the last operation; that is, it does not necessarily contain exactly the same information as the current header. This ensures that in the unlikely case of a corruption of the main header file as much information as possible is preserved.

## 4.1 Raw Disk Directory Organization

**WARNING:** *Never* edit the raw disk directory file manually.

A raw disk is organized as a collection of consecutively numbered “blocks”. Image sequences are stored starting at certain “offsets”. This is the block number where the sequence begins. The first frame stored for a sequence actually does not contain visible image data, but informations describing the image sequence. This information is used for crash recovery with the `hview` tool.

The location information is stored inside of the Directory Header file, aside with other information:

Each image sequence has a name that is used inside of Jaleo clip files to address the sequence. A name consists of a “/”, the project name the clip has been created from, plus a “/” and the clip name used when the sequence was created. Furthermore, the clip name will have the characters “H\_” or “L\_” prepended, depending if it is a preview or fullres clip. An example:

```
/JALEOTUTORIAL/H_testme
```

is the name for a sequence created in the project JALEOTUTORIAL when the clip testme was created. The prepended H\_ tells us that this clip is probably full res.

Aside of the name, there is:

- The number of images for the sequence
- A code for the image type (RGB24, RGB32, YUV)
- The starting offset for the sequence
- The number of bytes per image pixel
- Image size in x and y
- The number of disk blocks required to store one image. This number is calculated internally for best performance and is based on the image size and depth (bytes per pixel), as well as on the type of disk array used.

Here is an example entry (name – number of images – start offset – image type – bytes/pixel – sizex – sizey – blocksize per image):

```
/NewProject/H_test1 26 3952 1 3 720 576 2430
```

Directory entries are always ordered with ascending start offset values. *Never* reorder or otherwise edit raw directory files, or you guarantee data loss.

## 4.2 The `hadmin` Raw Management Tool

Note: See “Raw Disk Recovery Guidelines”, on page 24, for basic information on recovery strategies in case of trouble with your raw disk.

The `hadmin` (for “header administration”) utility is a shell based tool program for raw volume administration. It can be used to display and manipulate raw disk content, and to perform other administration tasks, like raw disk compaction.

To use `hadmin`, open a shell window from the toolchest menu. `Hadmin` needs a lot of screen space, so best make your window as wide as the monitor.

By default, `hadmin` asks for confirmation for every command. If you do want to prevent `hadmin` from asking for confirmation, you can run it with the `-x` option. **Note: This is strictly discouraged, as even destructive commands do not warn before they are performed.**

`Hadmin` must not be run at the same time as other Jaleo applications, with the exception of `hview`.

Upon start-up, `hadmin` automatically loads the current header.

`Hadmin` commands consist of a sequence of letters. To see the available commands type `<h>` and `<return>` at any time. You will see a list of commands available. In the command list, letters enclosed in `[ ]` are the letters you must type to execute the command. For example, for the command listed as `[l]oad [i]nternal` you must type

```
l i<return>
```

Note that there must be a space between the command letters.

`Hadmin` offers the following commands:

### 4.2.1 Loading Header Information

You can load header informations from different locations:

- `load` and `load header` load the current header file.
- `load backup <number>` loads the backup file with the number given. For example,  

```
l b 2
```

loads the backup file number 2.

- `load internal` loads the header information stored on the raw partition itself. The directory is not written to the raw disk itself automatically; that is, this command can only succeed after you have written the raw directory to the raw disk explicitly. A reason why you might want to do this is when you wish to connect a raw array to a different machine. To transport the raw directory on the raw disk itself may be much more convenient than transporting it on a separate DAT.
- `load external <file name>` loads header information from an external ASCII file named `<file name>`.

#### 4.2.2 Displaying Header Information

- `see` and `see header` display the currently loaded header in a compact format
- `see expert` displays the currently loaded header in an extended format. The BPI field gives the number of disk blocks per image.
- `see log` displays the operation log that contains a list of all operations performed on the raw disk.
- `see backup` permits you to see the list of backup files currently available.
- `see fragment` shows the empty spaces between image sequences. Empty areas can be the result of deleting sequences from the raw device. The listing also shows which sequences would be moved if a `compact` command were issued.
- `see backup_last_num` shows you the number of the backup file that contains the last backup. This information may be very important for recovery.

#### 4.2.3 Saving Header Information

- `save` writes the header file currently loaded to the main directory header file. Be careful with this command – if you have loaded an old backup file and issue the `save` command, the old (and probably outdated) backup file will replace the current directory header.
- `save <file name>` writes the currently edited header to the file pointed to by `<file name>`.
- `save internal` writes the directory to the raw disk itself.

This is useful if you wish to connect the raw disk to another machine and you do not wish to transport a DAT or other additional media with the raw directory information. See also `load internal` as described above.

#### 4.2.4 Compacting the Raw Disk

- `compact` minimizes unused space between sequences on the disk by moving sequences physically. If the disk is heavily fragmented, this can be a time consuming operation. Use the `see fragment` command to see the fragmentation of the disk.



### 4.2.5 Moving Sequences

- `move <name> <offset>` moves the sequence with the given name to the offset given by the `<offset>` parameter. Move does not overwrite existing sequences; the operation is only performed if no collisions result from the move.

### 4.2.6 Adding Sequences

The `add_seq` command is essential for crash recovery. With the `add` command, a new directory entry can be added to the raw directory. Together with the `hview` utility that gives visual support in the search for lost image sequences, it can be used to comparatively quickly reconstruct lost or damaged directory header files.

The `add_seq` command receives a command string consisting of keywords and values:

```
add_seq      <seq      name>      SIZE      <noImages>      FORMAT
<RGB24|RGB32|YUV422> IMAGESIZEX <sizeX> IMAGESIZEY <sizeY>
OFFSET <starting block>
```

The parameters are:

- `<seq name>`: The name the sequence will have in the raw directory header.
- `<noImages>`: The number of frames stored in the raw device
- `<RGB24|RGB32|YUV422>`: The format is designated using ONE of the three possible format identifiers. Format can be either RGB24, that is RGB without alpha, RGB32 (RGB with alpha) or YUV.
- `<sizeX>`, `<sizeY>`: The size of the images in pixels
- `<starting block>`: The block where the sequence begins. This is the block where the image marker is located, that is, actually one frame before the first image.

Be careful with `add_seq`: It does not only change the directory file. Also, a new marker frame is written at the offset position specified. The scan mode of `hview` takes this into account.

### 4.2.7 Deletion Commands

- `delete <sequence name>`: Deletes a named sequence from the raw disk.
- `delete internal`: Deletes the header copy written into the last block of the raw volume. The internal header only exists after you have written it out explicitly using `save internal`.
- `delete header`: Deletes the current header file. Caution: make sure that you always create another header file after deleting one – a raw disk without header is not useable, or at least its content is inaccessible and in danger of deletion.
- `delete log`: Clear the log file.

- `delete backup <number|*>`: Deletes the numbered backup file given by `<number>`. If you give an asterisk instead of the number, all backup files will be deleted.

#### 4.2.8 Recovery

In case a transaction is interrupted in a non-complete state, for example by a kernel crash or power failure, the system leaves around appropriate recovery information. By default, the recovery is silently and automatically done the next time you start up a Jaleo application accessing the raw disk. However, if you have set the option `TRANSAUTORECOVERY` in the `Raw.dev` configuration file, the system merely warns you that a failed operation exists and expects you to clean up manually. To do so, use `hadmin`'s `recover` command:

- `recover`: Clear the transaction state and make sure the system is back to the state it has been in before the operation that was left unfinished.

See the section “Raw Disk Recovery Guidelines”, on page 24, for more information on recovery strategies.

### 4.3 The `hview` Recovery Tool

The `hview` utility permits the user to recover raw disk content semi-automatically after a directory content has been damaged. `hview` is based on the fact that even after a directory damage or even a disk crash typically parts of the disk content are still intact.

Note: See “Raw Disk Recovery Guidelines”, on page 24, for basic information on recovery strategies in case of trouble with your raw disk.

The `hview` utility can scan the raw disk for the sequence markers placed before each sequence in the raw disk (see above) to find image sequences. Once a marker is found, the information contained is printed out in a message directly useable for the `add_seq` command of the `hadmin` tool, to regenerate a directory entry for the sequence found. As scanning a huge raw array can be extremely time consuming (the number of blocks to be read is very large), `hview` provides interactive facilities to permit the user to start the scan in the vicinity of a stored sequence.

To do so, the `hview` program presents a display window for images, and a number of tool buttons and clip format controls/display elements at the bottom of the window. There is also a large slider that permits to position the raw disk to any block on the disk.

With the image format controls, the user can tell `hview` to try to interpret the blocks read from the disk as images in any of the supported disk formats. There are presets for the standard PAL and NTSC formats, and the common preview resolutions for each of these. Furthermore, the user may define her/his own custom resolutions. There is also a

selector for the image format – the three supported formats RGB24, RGB32 and YUV can be selected here.

While browsing through the disk using the slider, `hview` tries to interpret the data read, starting at the current disk block, as an image of the given format. The image read is displayed in the large display window.

Of course, the display only makes sense if the slider is positioned at a reasonable position. However, if the image format is set more or less correctly, image sequences usually can be found quite rapidly just with the slider. Once you believe you are close to the start of the sequence, you can go back a few blocks (using the Previous Image button or the slider arrow buttons) and scan for the precise starting block of the sequence. Start/Stop Scan begins or interrupts a scanning process. Once it finds a sequence marker, it will display a suitable command string for `hadmin` in the message line. You can just copy this string with the mouse to a copy of `hadmin` running in a shell window at the same time.

Once a sequence marker is found, you can use the Next/Previous Image buttons to verify that the images themselves are correct. Next/Previous image will forward or backup the current block by the blocksize determined by the current image format selection.

#### 4.3.1 Recovery Of Lost Sequence Markers

While the manual approximation / scanning approach works fine while the sequence markers are intact, recovery is much more complicated in case a sequence marker is lost. In this case you may still be able, using backup copies of the directory header or browsing with the slider, to find a sequence. However, you need to find the precise location of the first usable image block, or the results will not work. To find precise image starts, you first should try to find a location somewhere in the sequence. This will typically involve playing with the image resolution and format fields, in case you do not know the resolution and format of the images. Once you have images on screen, just not aligned properly to the screen, you can use the shift field at the very right of the tool line to adjust the current block position. This is just a more precise and fast adjustment than the scrollbar can offer. Once you have managed to centre one image of the sequence properly on the screen, you can use the Previous Image button to backup to the start of the sequence. Write down the block count for the first image. Step back one more frame and write down this value too. Now use Next Image to find the end of the sequence. Write down the block number of the last frame of the sequence. Forward one more frame (you will probably see trash) and write down the block number for the end. Now you have everything you need to calculate the missing parameters:

- The imagesize and format is given in the sequence format fields.
- You must find out the number of frames in the sequence. If the sequence is short, you can simply count the frames. For longer sequences, you may be faster with the following approach:

- The number of blocks per image is given by the difference between the block numbers of any two consecutive frames. Use for example the numbers written down for the last image, and the image immediately after. Subtract the smaller block number from the larger one. Make sure that the values are from consecutive frames, or the result will be totally wrong.
- Subtract the block number for the first image from the block number of the frame after the last usable image of the sequence. Divide the number by the BPI value computed before. This will give you the number of images in the sequence. In case the sequence is short, you can instead simply count the frames while stepping through it using Next Image.
- As the `add_seq` command of the `hadmin` tool writes a marker frame at the first frame position you give it, you have to use the number for the frame immediately before the first image for the `OFFSET` value in the command line.

## 4.4 Raw Disk Directory Conversion

Jaleo 2.1B has been using a file stored in `JALEO-ENV/etc/devices/raw_info` to maintain the raw disk directory. There also have been a number of backup locations for this file.

To use existing raw disk content from Jaleo 2.1B, you must convert this directory file into a Jaleo 2.5 raw disk directory. The disk content itself does not need to be touched: While raw disk files written by Jaleo 2.5 use a different storage format, files written by older Jaleo versions can still be read, although possibly with slower performance than native 2.5 files. Although header conversion is possible, it is preferable to start with a “fresh” raw disk. Jaleo 2.1-written raw sequences do not have the recovery functionality as it is present for Jaleo 2.5-written data. If you have a chance between projects to clean up your disk and to delete everything, we recommend this strongly.

Even if you must maintain data, make sure to either back up the data or to have available original video tapes for quick recapture of the material, in case of a failure. You can backup material

- by copying the clips to a normal disk using the IO
- or by using the `jbackup` utility

### 4.4.1 Converting Directories

**Warning: Do NOT convert a directory while you are in the middle of a production and the data on the drive is not replaceable. Also, make a backup before attempting to switch from Jaleo 2.1 to 2.5 raw disks.**

If you want to convert a raw disk header, these are the steps to do:

- Install Jaleo 2.5 as usual. Configure the software for raw device operation, **but do NOT run any of the Jaleo 2.5 applications**, as this would corrupt the existing raw disk partition.
- Make a backup of your data. There are basically two ways to backup raw content for the transfer from 2.1 to 2.5:
  1. You can use the Jaleo backup utility with Jaleo 2.1. Note that on real time systems a quality reduction may be the result if analogue VTRs are used and/or RGB imagery is backed up – real time backup via the Sirius or Impact video is 4:2:2 YUV.
  2. You can use the IO of 2.1 to copy from the raw device to a clip stored on a normal disk. Be careful to maintain the clip names, or your environments may not work any more.
- run the utility `hconvert` to create a Jaleo 2.5 header file from your existing raw head. `Hconvert` must be run from a shell. It takes two input parameters:

```
hconvert <2.1header> <2.5header>
```

The first parameter is a valid Jaleo 2.1 directory header file (typically a copy of `JALEO-ENV/etc/devices/raw_info/RAW_HEAD`), the second one is the name for the file to generate – this will be a Jaleo 2.5 directory file. Typically, you will want to place it in `/usr/tmp`. An example run would be:

Login as `jaleo25`. Type in a shell:

```
cp JALEO-ENV/etc/devices/raw_info/RAW_HEAD /usr/tmp/oldHead
hconvert /usr/tmp/oldHead /usr/tmp/newHead
```

After this, `/usr/tmp/newHead` should be a valid Jaleo 2.5 header file. You can now validate this generated file using the other 2.5 raw disk management utilities as described before.

- Use `hadmin` to load and verify the generated header. Try the option `load external` on the file `/usr/tmp/newHead` (provided you did the steps as described above). Now use the `show` options to see if the file looks sensible. If it looks good (that is it contains all the clips it should have, and these clips show data that at least at first look makes sense), save it using the `save` option. Do **not** run `jaleo25` now.
- Use the Jaleo 2.5 flipbook to view all the clips that use sequences from the raw disk. Do **not** use any application that potentially can write to the raw disk, that is, neither Jaleo, nor the IO, dustbin or RtVideo applications. Make sure the clips work fine.
- If your clips work fine, you principally can run Jaleo 2.5. **However, running 2.5 on a raw disk previously used by 2.1 means that you can not run Jaleo 2.1 any more.** Make sure you do not need 2.1 to access the given raw disk before you run 2.5. While Jaleo 2.1 and 2.5 installations can coexist in the same machine, they can not share the same raw disk. **To make sure you do not accidentally run the old version on the raw disk, you must now reconfigure Jaleo2.1** (in case you want to maintain it), so that it will not use raw disk access any more.

Of course, if you have an extra disk, you can configure one as raw device for 2.1 and the other one for 2.5.

Alternatively to use `hconvert`, you could use one of two different conversion strategies (also we recommend strongly to use `hconvert`):

1. You can use the Jaleo backup utility with Jaleo 2.1 and restore with Jaleo 2.5. Note that on real time systems a quality reduction may be the result if analogue VTRs are used and/or RGB imagery is backed up – real time backup via the Sirius or Impact video is 4:2:2 YUV.
2. You can use the IO of 2.1 to copy from the raw device to a clip stored on a normal disk. After installing 2.5, you can use the IO of 2.5 to copy the disk clip back to a clip in raw mode. Be careful to maintain the clip names, or your environments may not work any more.

Also, in case you have an extra disk, you can setup the extra disk exclusively for Jaleo 2.5, thus maintaining the originals 2.1 raw disk system until all data is out of use.

## 4.5 Raw Disk Recovery Guidelines

There are different possible causes of raw disk failure and different means to resolve problems. This section tries to give hints on how to deal with raw disk trouble caused by system failure.

### System Crash, Power Failure or Software Crash during a Raw Operation

An interruption in a raw disk operation generally should lead to automatic recovery of state the raw disk had before the operation was attempted. However, if you set up the system for manual recovers by placing the field `TRANSAUTORECOVERY` with a value of `FALSE` in your `Raw.dev` file, you will have to use the `recover` option of `hadmin` to achieve recovery manually.

### Corrupted Raw Disk Directory

If you encounter that the content of your raw disk directory, as listed by the `Raw Status` command or `hadmin`, is corrupted, there are a number of possible recovery strategies. A corrupted hard disk directory is a very unlikely possibility, but it might happen as the result of a disk head crash or other failure of the system.

- If the disk that hosts the `jaleoHeaders` directory (the `/usr/local` filesystem) can be recovered after the crash, even if the raw directory file was corrupted by the crash there are a number of backup files. Which file is the last backup can be retrieved using `hadmin`. The steps to be taken would be:
  - Start `hadmin`
  - Save the current header to a file with a private name
  - Get the name of the last backup file
  - Load this file as current header and save it

- Try out the recovered header
- If this did not work, use a header backup file prior to the last one. That is, if the last backup for example had the number 34, try number 33.
- If the file content is satisfactory, leave the system as is and proceed. Note that if you use a prior backup, your last operation may have been lost. If you step back further, you may have more losses, but this may still be preferable to a total loss.
- If you very recently saved a raw directory to the raw disk itself, you can also load the internal header for recovery purposes. This may make sense if the backup file listed by `hadmin` does not exist, looks invalid or contains obviously nonsense.

To do so,

- Start `hadmin`
- Load the internal header
- Save it back as the current header
- Test

### **In case of total directory loss or a partial crash of the Raw disk itself**

In case no directory can be recovered by none of the previous recovery method, or if the raw disk itself has either been damaged physically or overwritten accidentally, you can still use `hview` to partially extract image content from the disk. Even if you may have losses, you may still be able to use the data partially.

Recovery with `hview` gives two basic possibilities:

- Each sequence on the raw disk is preceded by some header blocks that contains the basic information on the sequence. If this block, for a given sequence is intact, `hview`'s scan mode will quickly find the necessary information to rebuild a directory entry.
- If the header blocks are lost, a directory entry for the intact sequence or parts of it must be rebuilt manually.

Both recovery methods are described above. See “The `hview` Recovery Tool”, on page 20, for more information.

## 5. Basic Video Hardware Setup for RtVideo Operation

Correct video hardware setup is absolutely essential for frame accurate capture. This holds true especially for the systems using stand alone VLAN controllers, that is, for Jaleo Composite and Jaleo for Impact.

Unfortunately, there is virtually an unlimited number of variants of connections, devices, and setups. It is thus not possible for us to give a detailed setup description for every possible configuration. Instead, we have tried to collect some universal facts on the matter.

The central issue is:

- All video hardware must be properly synchronized using a stable reference signal

The components involved are the following:

- The SGI machine and its video hardware. Depending on the machine and video board, the video side may be analog or digital.
- A VTR capable of insert cuts. This again can either be digital or analog.
- A control monitor, either digital or analog
- A VLAN controller for machine control
- Optionally, there may be a digital-to-analog or analog-to-digital converter necessary to connect analog equipment of some kind to digital ones.
- Optionally, there may be a source for a studio reference signal.

The trick for successful operation is to keep all these components in sync on the video side.

A common necessity on the SGI side is **the proper setup of video inputs, outputs and synchronisation** using the `vcp` (video control panel) application. You can run the control panel from any shell by typing `vcp`. Although the `vcp` applications look slightly different, depending on the hardware capabilities of the boards, all permit to setup the video synchronisation. **Do not forget to set the board to the sync source used!**

### 5.1 A Note on VLAN

VLAN is a universal controller for video equipment. A VLAN system consists of two base components:

- A VLAN Transmitter. This device translates commands expressed as simple text into an internal protocol for video device control. Commands do look like “PY” for “Play” or “GT 00:00:12:22” for “Goto location...”. These commands typically are issued by a



program or edit controller and sent to the transmitter via a serial interface. Sometimes, transmitters are also integrated in video equipment. As an example, the SIRIUS board from SGI has a VLAN transmitter on board.

- One or more VLAN Receiver. The receiver translates the internal protocol of the VLAN system into device specific remote control commands. The major strength of VLAN is that drivers for basically all common video devices are available.

A single VLAN transmitter can control up to 32 different devices at once. These devices are called “nodes”, each identified by a node number. For each node to be controlled, a separate receiver is required. Connections between transmitters and receivers are done via normal video BNC cables (75 Ohm).

### 5.1.1 VLAN Timeout Handling

The RtVideo (as well as the realtime backup applications for Jaleo Plus and Jaleo for Impact) will intent up to three retries if the VLAN times out during an operation. A time-out of a VLAN command might occur, for example, because the tape transport takes to long to position the tape appropriately.

### 5.1.2 VLAN Flavours

VLAN comes in various flavours:

- There are the old small black plastic boxes that contain a single transmitter or receiver each.
- There are integrated transmitters on video boards.
- There are little metal boxes containing either a transmitter and a receiver, a single receiver or two receivers. The newest model, called VLXi, is of this type. For computer graphics operation, a VLXi configuration with one transmitter and one receiver, saving an explicit connection between transmitter and receiver, is the most common case.

### 5.1.3 VLAN and Frame Accuracy

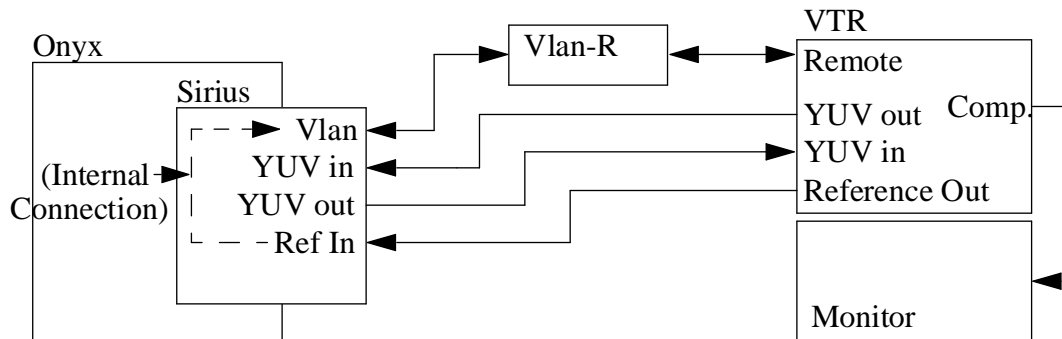
Vlan controllers are absolutely frame accurate, provided the transmitter is supplied with a stable and reliable sync signal (the same of course that drives the rest of the video equipment). The transmitter is equipped with a diagnostic LED to show if a valid sync comes in through the Reference Input. **Without a proper sync, no frame accuracy can be achieved.**

## 5.2 Jaleo Plus

To begin with Jaleo Plus makes sense, as this is the easiest setup on the video side: The Sirius video, the only option for the ONYX machines, has a built in VLAN transmitter,

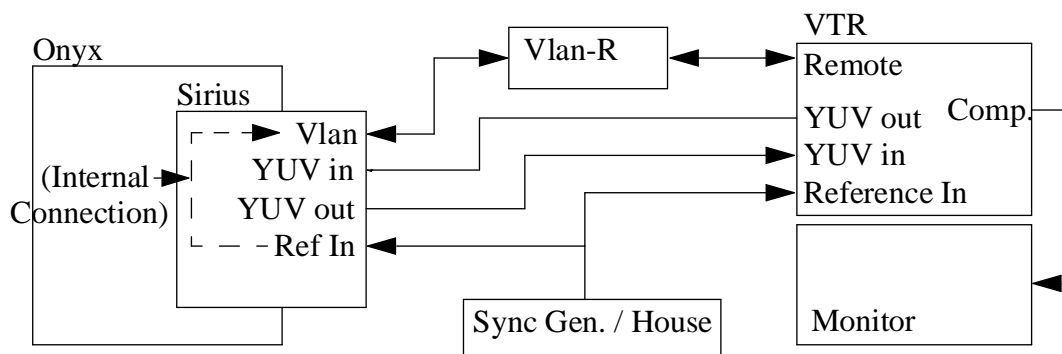
that automatically is in sync with the video itself. Synchronisation thus is a bit easier, as only one device needs to be synchronized.

A basic setup looks like this:



In this example, an analog VTR is used to sync the Sirius video with its reference output; a BetaCam recorder has a built-in sync. It is absolutely crucial that the software sync setup for the Sirius video matches the cabling. That is, if a sync comes in via the reference input (genlock) of the Sirius, this must be setup in the vcp control panel application. Failing to do so can cause the Sirius to crash (the Sirius does not like to be left without a sync).

A slightly more advanced setup would use an external sync generator:



In this case, the sync source may either be a small stand-alone generator, or the house sync of a big studio.

In case digital equipment is involved on one side only, a digital-to-analog converter must be placed into the signal flow. Note that this converter must also receive a sync signal. As this setup is less common for the ONYX with Sirius (the Sirius can be equipped with both digital and analog in- and outputs to match external equipment, we leave it out here.

### 5.2.1 Video Out with Sirius Video

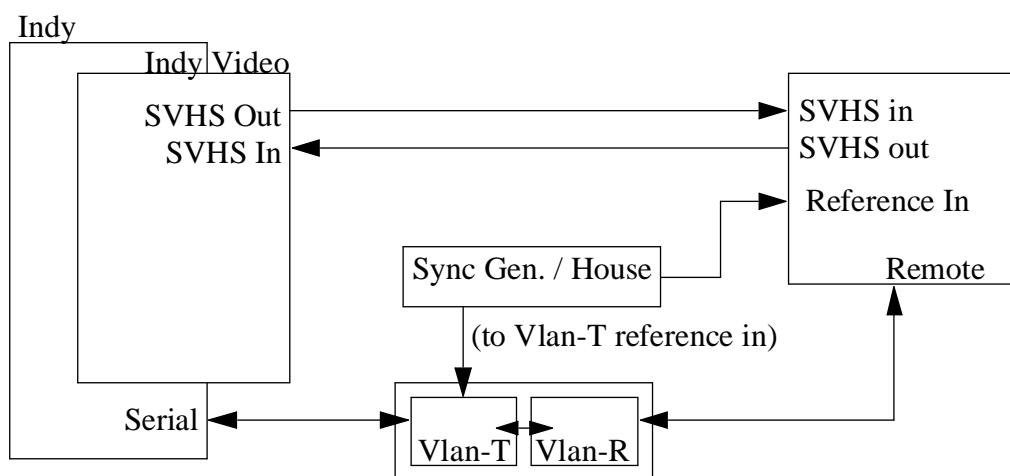
Live Video output with the Sirius board does only work if Jaleo's working resolution is set to one of two possible combination for PAL and NTSC each. The vcp video resolution settings for the board must match the resolution chosen.

	PAL Analog	PAL CCIR	NTSC Analog	NTSC digital
<b>IMAGESIZEX</b>	768	720	646	720
<b>IMAGESIZEY</b>	576	576	486	486
<b>vcp mode</b>	625 Square Pix	CCIR 625	525 Square Pix	CCIR 525

## 5.3 Jaleo Composite Systems

In comparison to the ONYX setup, the situation here is slightly complicated by the need of a separate VLAN transmitter, as neither the IndyVideo, nor the Galileo have it built in.

A setup for the Indy might look like this (the setup is shown with a sync generator, although the reference out of the VTR, provided it is stable, might be used as well):

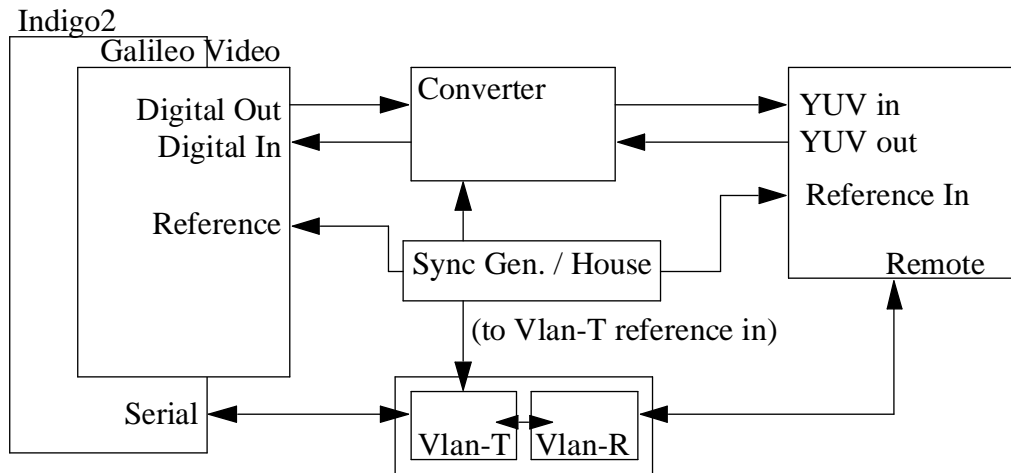


The new aspects here are as follows:

- The Indy does not have a reference input. It must extract its video reference from the video input. Still, it must be set up appropriately (genlock mode).
- As the VLAN used has a separate transmitter, **the transmitter must receive a sync signal, or it will not operate correctly**. The sync signal may come either from the VTR or from a separate sync source, but it must be stable. Note that all VLAN transmitter models have LEDs to show if they receive a valid sync signal. If you have trouble, check, first check your VLAN!.

As described above, VLANs come in various flavours. The transmitter and receiver may physically be in the same box, or in separate boxes.

A setup for a Galileo with Digital option looks like this (again, the sync might alternatively come from the VTR):



The differences against the setup described above:

- The image shows a digital Galileo board, and a converter to an analog VTR. Note that the converter should also be synced to the studio sync.
- The Galileo board has a reference input. Use it! And do not forget to set up the vcp appropriately, that is, to switch on the Genlock mode with the appropriate input selected for sync.
- **Again, the VLAN transmitter must be connected to a valid sync source, or it will not operate correctly.**

### 5.3.1 Video Out with IndyVideo

Live Video output with the Indy Video board does only work if Jaleo's working resolution is set to 768\*576 (PAL) or 646\*486 (NTSC) pixels. Unfortunately, the board does not support other resolutions.

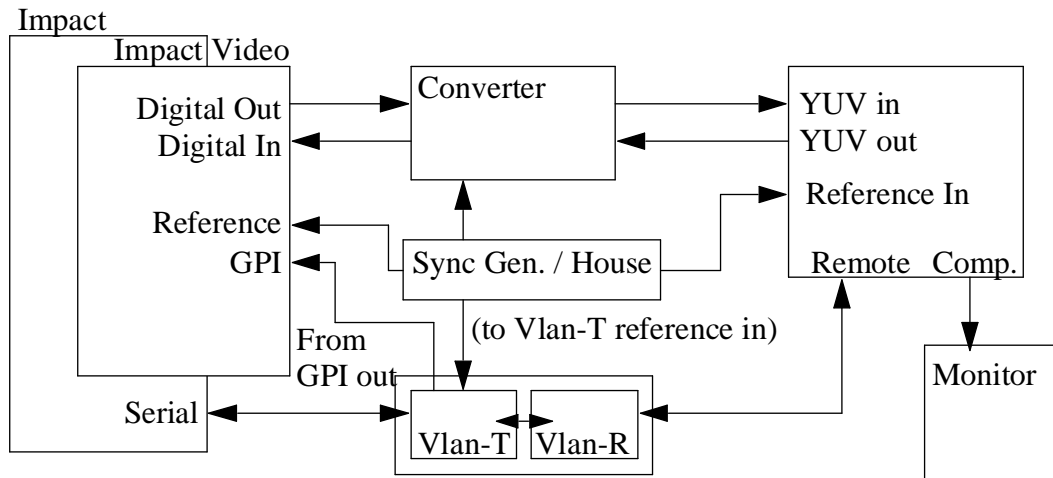
### 5.3.2 Video Out with Galileo Video

Live Video output with the Galileo does only work if Jaleo's working resolution is set to one of two possible combination for PAL and NTSC each. The vcp video resolution settings for the board must match the resolution chosen.

	PAL Analog	PAL CCIR	NTSC Analog	NTSC digital
<b>IMAGESIZEX</b>	768	720	646	720
<b>IMAGESIZEY</b>	576	576	486	486
<b>vcp mode</b>	625 Square Pix	CCIR 625	525 Square Pix	CCIR 525

## 5.4 Setup for Jaleo for Impact Systems

On Jaleo for Impact systems, another addition is necessary. The setup looks like this:



In comparison to the Galileo setup shown above,

- one additional cable is necessary to connect the GPI trigger output of the VLAN transmitter with the Trigger input of the Impact Video. The GPI (General Purpose Interface) of the VLAN gives trigger information that is necessary for synchronization on the Impact. **Do not omit the GPI connection, or RtVideo will not be able to capture with VTR control.**

Let us repeat the other essentials:

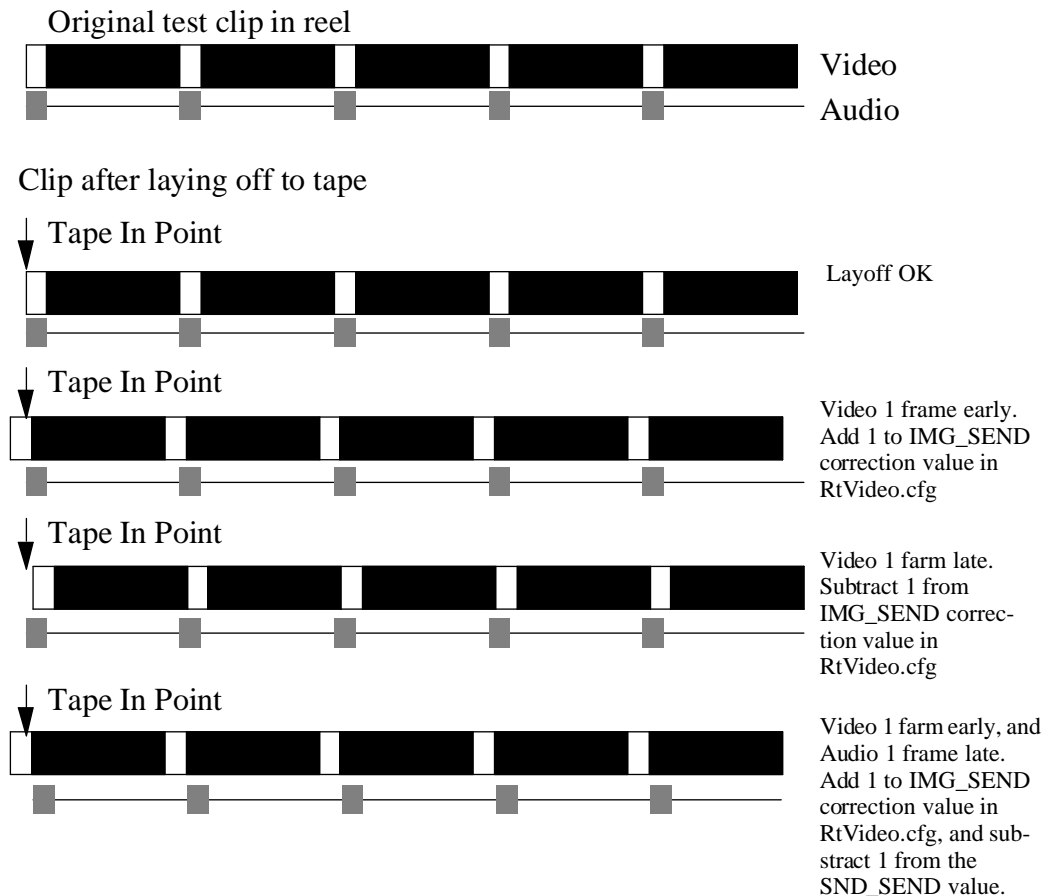
- The VLAN transmitter must be connected to a valid sync source. **Without that, no frame accurate operation is possible.** Again, the transmitter provides a diagnostic LED to check if a sync is present.
- The video control panel must be setup properly for external sync, and the reference sync must be connected to the reference input of the Impact Video.
- In case an analog/digital converter is used, the converter should also be synced to the common reference.
- Where no separate sync generator is available, most professional VTRs can be used as reference source, providing a separate reference output. The reference signal can then be daisy-chained to all the devices, the VLAN, the converter and the Impact.

### 5.4.1 Calibrating Video/Audio Synchronization

A simple test procedure for Video/Audio synchronization is as follows:

- Create a clip consisting of a second of black images, with the first two images set to white. You could do this by using an Empty effect for the black frames and a RGB Filter without input for the white ones.
- Add Audio in a way that you have a test signal that takes exactly the time of the white frames. A good candidate is any wave signal, recorded over time.

- Repeat that sequence so that it fills various seconds (at least 10 to 20), always being 2 frames white, then the rest of a second black, with sound exactly on the white frames.
- Render this sequence out to the raw disk.
- Use RtVideo to lay it off to video.
- Check if audio and video are in sync.
- If there are any deviations, use the appropriate correction values in the RtVideo.cfg configuration file (IMG\_SEND\_CORRECTION, SND\_SEND\_CORRECTION) to get the tracks aligned. You can pull a track forward by subtracting from the default value. For example, if you change the IMG\_SEND\_CORRECTION from -1 to -2, the video track will start one frame earlier. See “RtVideo.cfg with Jaleo for Impact”, on page 11, for more information. The main reason for correction in the video track would be a delay in the AD/DA converter if you use one; do not use video track correction if the video starts at the right position on the tape, as specified in RtVideo. On the other hand, if the audio starts aligned properly to the record in point, do not adjust the audio, but the video track. Examples:



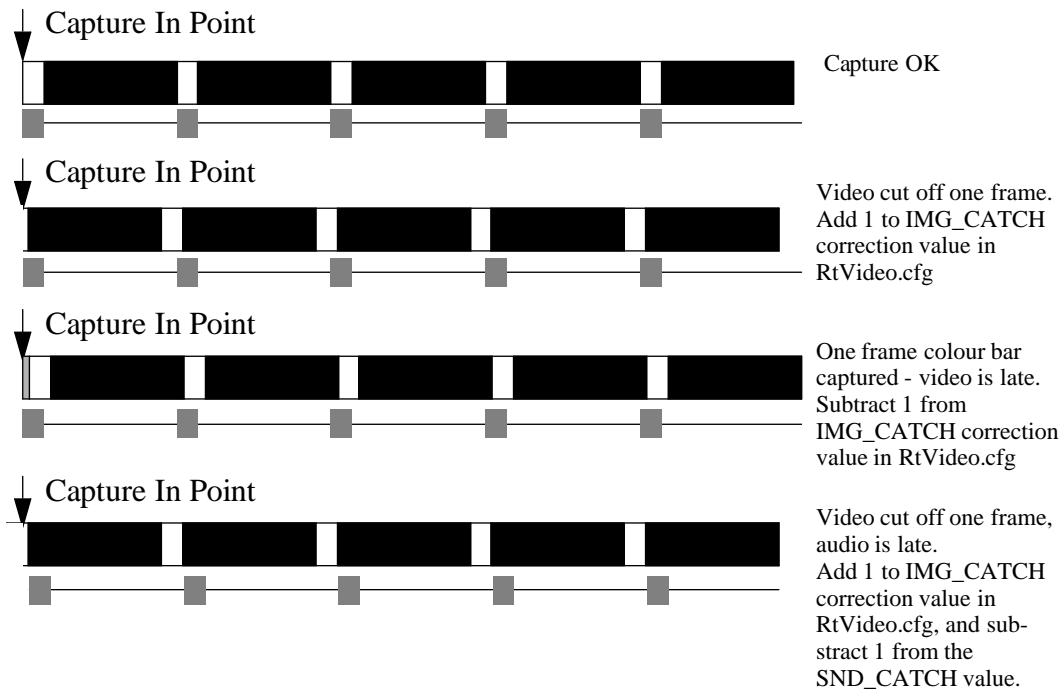
- Lay off again and cross check.
- Once you achieved proper sync, try to recapture the sequence from the video tape.
- Check synchronization in the Reel.

- Use the `IMG_CATCH_CORRECTION` and `SND_CATCH_CORRECTION` to calibrate the capture as desired. Examples:

TestClip as recorded on tape, with colour bars before and after



Clip in Reel after capture:



## 5.4.2 Video Out with Impact Video

Live Video output with the Impact Video board does only work if Jaleo's working resolution is set to proper resolution settings. The `vcp` video resolution settings for the board must match the resolution chosen.

	PAL CCIR	NTSC digital
<b>IMAGESIZEX</b>	720	720
<b>IMAGESIZEY</b>	576	486
<b>vcp mode</b>	CCIR 625	CCIR 525

## 5.5 Audio

The sketches above have left out audio connection, for the sake of clarity. Audio must be connected straight from the VTR to/from the workstations audio in/outputs. The audio control panel can then be used to select the desired input, to control levels, etc.

In playback mode, you should switch off audio monitoring, to prevent feedback effects.

## 5.6 Summary

For proper and frame accurate video operation, the availability of a stable reference source is of uttermost importance. **Without syncing the VTR, the VLAN and the SGI Video to the same reference, no frame accurate operation is possible.** Also, you must make sure that the video control panel software settings for the video hardware you use in your SGI match the physical cabling.

Typically, you will either want to use a separate sync generator or the VTR as sync source (provided your VTR can be used that way). We do not recommend to use the SGI as sync source – always genlock the SGI video to an external reference.



## 6. Usage Note Regarding Flicker Problems and DVE Moves

Some users had problems with flickering DVE moves. It turned out that the problem resulted from the fact that the user did not render in interlace mode. We thus wish to repeat the comment from the manual: Whenever you use material that is in interlace mode, you should render in interlaced mode to get good results. Whenever you do fast DVE moves, wipes etc. you even should render in interlace if your material is not interlaced.

In short: As most video material, the typical raw material for Jaleo, is interlaced, it almost always is a good idea to render in interlaced mode.

There is one important exception: If you have a single frame of video material to be stretched out (repeated) over many frames, you *must* make sure this frame is not interlaced. Otherwise, a strong flicker will occur. To de-interlace a single frame for use in stretching, the best way to achieve this is to make a quick render of the frame in question, switching off the FIELD option while doing so. Jaleo will actually present you with a warning if you try to render a single frame in field mode.

Addition for Sierra Quickframe owners: Due to a particular property of this machine, you must set the NTSCFIELDS variable in the Jaleo setup file `.jaleorc` to FALSE, even if you are working in NTSC. If you put it to TRUE, you will get inverted fields on the DDR - definitely ugly for DVE moves.

## 7. Bug Fixes

Aside of the bug fixes listed here, there is a large number of fixes and improvements related to changed, improved or replaced features mentioned elsewhere in this text.

### 3D-DVE

Antialiasing bugs have been fixed.

### Display Size Bug for Flipbook and Paint in NTSC

In NTSC mode, RotoPaint and Flipbook previews in low resolution would be scaled incorrectly.

### DVE2D

A bug has been fixed that caused the system to hang (while leaving the CPU very active in gr\_osview).

### Groups and Clone

Groups can now be cloned and ungrouped without problems of “homeless” effect scope indicators.

### IO and Abekas, Sierra, HP DDRs

A bug in the driver for the mentioned DDRs caused some versions of Jaleo to render only a single frame to the DDR. The IO would then freeze. This has been corrected.

### IO, Disk Device

- When the image format for images in Disk.dev was set incorrectly, the IO would die. This has been corrected.
- Related: When the format was set to YUV, the IO would die after processing approximately 100 frames. This has been corrected.

### Movie Files

The frame rate and field order flags for movie files are now written correctly.

### Movie File Formats

Movie files would use the same file format for preview and highres movies, even if different movie compressions were selected. This does not happen any more.

### Noise

The RollZ parameter did not give good results in all situations

## **PickEditor**

A bug that sometimes caused a crash when closing the PickEditor has been fixed.

## **RotoPaint Previews**

When rotoscoping images, the preview displayed when moving the frame slider always would show the original images, no matter which changes had been applied. This has been fixed – the modified images are now shown.

## **RenderGroup**

- Render Group now detects Out of Disk Space conditions
- In case a clip already exists, the Render Group now puts up a warning message.

## **Save in a Navigation Level**

Using Save in a Navigation level of the Reel caused the system to save only the content of this navigation level. This has been fixed.

## **Setup Constraints**

The x and y constraint setups now are correct. In the last version, they have been switched.

## **Trimming of Sound Clips**

Trimming of sound clips did not readjust the curves properly.

## **Undo/Redo**

A number of bugs in the UNDO mechanism have been fixed. Some of these involved incomplete UNDOes in some situations, while the most grave one caused the system to exit after repeated drop/undo operation sequences.

## **YUV File Read**

While older versions of Jaleo could render YUV files properly, trying to read them caused a crash. This has been fixed.

## 8. Known Problems and Bugs

### 8.1 General

#### 8.1.1 Hotkeys do not Work if CapsLock or NumLock are Active

It is a strange behaviour of SGI's X Windows server implementation, but unfortunately, if CapsLock or NumLock modes are active, our windows (like those of many other applications on the SGI) do not receive proper hot key events. In particular, the keys on the cursor block and the insert/delete/page up/down keys will not work. There is not much we can do about that on our side.

If you experience trouble with the hot keys (that is, the software does not seem to react), please make sure you have NumLock and CapsLock deactivated.

#### 8.1.2 Temporary Files not always deleted

Some small temporary files written by Jaleo in `/usr/tmp` are not always deleted. The files in question have names starting with `tmp...`. The files are not big, so they should not bother you. You can delete them manually if so desired.

#### 8.1.3 TIFF File Format does not support Alpha

The TIFF file reader does not support images with alpha. Although the images can be read, the alpha information will be lost.

### 8.2 Reel

#### 8.2.1 PlugIns

PlugIns that only use the Jaleo time editor as their sole interface work fine without limitations. PlugIns with a private interface work fine if you are using a single Reel window. If you have opened a second reel window, the PlugIn interface does not recognize in which reel the edited effect is. As a workaround, we recommend to place PlugIns with a private interface only in a single Reel window.

#### 8.2.2 Large Movie Files (Cosmo)

Large movie clips seem to be handled incorrectly in the Reel sometimes – they appear black. The problem actually is in the movie file creation process, not in the reel. Fortunately, the problem is easy to work around.

If RtVideo runs out of disk space in system temporary storage during movie creation, an invalid movie file is created. Unfortunately, not always a warning is posted to inform the user of this fact. To run out of space is of course much more likely with large movie files.

You can redirect temporary storage to a directory with sufficient free storage. See “Cosmo Compress Temporary Storage Setup”, on page 12, for more information.

### **8.2.3 Lost Monitor Cursor after Loading New Environment**

In some occasions, you may loose the monitor cursor if you load in a new environment. This of course can only happen if you have at least one monitor window open. After the new environment has been loaded, you will simply not see the monitor cursor any more, not even in the Overview window. The workaround is to close the monitor window left without its controlling cursor and to open a new one from the Tools menu.

## **8.3 RotoPaint**

### **8.3.1 Animation Mode: Documentation Omission**

The documentation on RotoPaint is not very explicit on the relationship between animated and non-animated work. The issue is rather simple: You can not have non-animated and animated elements, be they vector based or immediate, at the same time. The reason for that is that otherwise you could paint on top of an animated shape. This sounds fine, but it becomes difficult to handle in the case you modify or change the animation afterwards – as non-animated paint actions are sticked to the image when you change frames, the effect of paint actions that originally may have depended on the presence of an animates shape, becomes totally unpredictable.

To solve the problem, RotoPaint enters a special animation mode whenever you add the first keyframe to any shape, that is, whenever you start to build an animation. You remain in animation mode until you use the New or Open options from the File menu.

In animation mode, all images previously rendered to the temporary storage (that is, all non-animated work you have done up to this point) will be lost. All future non-animated work will be ignored. As it is sometimes helpful to use these non-animated actions for temporary markers etc., RotoPaint does not prohibit to do non-animated actions while you are in animation mode, but these actions will be lost once you change the current-frame.

As entering animation mode is potentially destructive to work done before, RotoPaint puts up a warning when you select the Add Keyframe command for the first time. The warning informs you of the fact that your previous (future) non animated work will be lost (ignored) until you exit animation mode. You may cancel the operation or accept.

Once you entered animation mode, the border color of the image shown on the canvas changes to yellow.

### **Layering Animation and Non-Animated Paint Work**

If you wish to add vector based paint animation on top of rotoscoping or other non-animated paint work, you should do the following:

- First complete your non animated paint work.
  - Save the clip you created or modified.
  - Now, create one or more vector shapes you want to animate, either by painting or with the vector shape tools.
  - If you already had created a shape and decided later you want to animate it, you can still save your clip. Nothing of your non-animated work is lost until you actually use and confirm the Add Keyframe command for the first time.
  - If you wish to add non-animated paint work to your animation, you will have to
    - render your animation.
    - If you like the result, use Save Clip or Save As... from the File menu.
    - You may wish to save your vector animation as well – select the vector shapes you wish to preserve and use Save Shape from the file menu.
- Remember that you can also load animated shapes into the reel as vector objects, provided your shapes are closed. They will automatically be encapsulated in a DVE Shape effect when loaded in the reel.
- Use New from the file menu to exit from animation mode.
  - Reload the saved clip
  - Proceed with either non-animated paint work, rotoscoping or vector animation.

### **8.3.2 Necessary Disk Space Calculations**

If the directory used as temporary storage for RotoPaint is mounted with Automount, the calculations for required disk space for rendering are not reliable. The error does not occur on normally mounted network volumes, or local file systems.

Workarounds:

- Use a normal fstab mount
- Use a local file system
- Look for the free disk space on the machine that exports the file system

### **8.3.3 Save As... File Format**

Although RotoPaint can read any file format supported by Jaleo from disk (when opening clips), on save, it always writes Vista format files if the option Save As... is used.

If you use Save Clip it will write the clip in exactly the same format as it had before loading.

#### **8.3.4 RotoPaint and Movie Files**

If you have a clip with movie files, the Save command does only write lowres images – the generation of high res images in movie format does not work properly. As a workaround, save your work with Save As..., which writes the clip as Vista files. Then you can later convert it using Render Selection or the IO.

### **8.4 RtVideo**

#### **8.4.1 Drag&Drop in Clip->Video Mode**

Sometimes the clipname, if drag&drop is used from an SGI file manager window, is not set properly if the field is not cleared manually before the drop.

Workaround: Please clear the clipname field by selecting its content and pressing backspace before dropping a clip or sound clip file to it.

#### **8.4.2 Laying Off with AIFF Files**

Although RtVideo can lay off Aiff files to video, pure aiff files can not be added to the job list. To add something to the job list, it must be a .nclp or .sclp file. These are the wrapper files created by the IO, the RtVideo or the Reel upon capture/rendering.

Laying pure aiff files off to video works fine with the Do This option, though.

#### **8.4.3 Large Movie Files with Cosmo Compress**

Movie files with a large number of images seem to be created incorrectly sometimes. The reason for this is that the system run out of temporary disk space during movie creation. Although normally there should pop up a warning if this is happening, this is not always the case.

You can redirect temporary storage for movie creation to a directory with sufficient disk space. See “Cosmo Compress Temporary Storage Setup”, on page 12, for more information.

## 8.5 Backup/Restore

### 8.5.1 Remote Tape Drives

If during a backup using a remote tape drive (that is a drive connected via a network) errors occur, the graphical interface does not catch error messages. That is, the integrity of the backup can not be inferred from the absence of error warnings in JBackup. However, the messages appear in the console window, where they easily can be tracked.

For remote backups with jBackup, we thus *strongly recommend to check the messages printed in the console window* for errors. When using a remote tape drive, the absence of error message posted inside the graphical interface of JBackup is *no measure for an error-free backup* – you must at least cross-check the console messages.



## 9. Copyright Notice

Jaleo Composite, Jaleo for Impact and Jaleo PLUS are property of Comunicación Integral.

(c) 1994-1995 by Comunicación Integral. All rights reserved.

No part of the software or the associated documentation may be copied, photocopied, reproduced, translated or reduced to any electronic medium or machine readable form without the prior written consent of Comunicación Integral.

Comunicación Integral

c/ Huelva 8

28002 Madrid

Spain

Tel. +34 1 413 7497

Fax +34 1 416 9914

The license management portion of Jaleo is based on: Élan License Manager, (c) 1989-1994 Elan Computer Group, Inc. All rights reserved.

Élan License Manager is a trademark of Elan Computer Group, Inc.