CatDV Server 6.1

Release Notes and User Manual

Square Box Systems Ltd
29 Waterloo Place
Leamington Spa
CV32 5LA
United Kingdom

Table of Contents

1 Introduction

These release notes cover the combined release of the CatDV Workgroup or Enterprise Server for MySQL or Oracle, together with Live HTML Publisher.

IMPORTANT: You may install and use this software only in accordance with the terms of the enclosed software license agreement.

1.1 Products Covered

This is a universal installer. Files for several different products and platforms are present on the installation archive and not all will be relevant to you. Which products are available to you depends on which license you have and the activation code(s) you enter.

Likewise, this document includes instructions for different products and platforms. Please read the whole of this Introduction, then refer to the chapter(s) that relate to you.

The following products are covered

- CatDV Workgroup Server for MySQL
- CatDV Enterprise Server for MySQL
- CatDV Enterprise Server for Oracle
- CatDV Live HTML Publisher

The following platforms are covered

- Mac OS X
- Windows
- Generic Unix/Linux

The Enterprise Server for Oracle is available on Windows only at present, while the other products are available for all three platforms.

1.2 Before You Start

Thoroughly read these release notes and make sure you understand and are comfortable with the requirements and all the installation steps. Make sure you understand *all* the various components and steps that form part of the overall system:

- The CatDV server stores clip data in a central database, for which you need database software. **MySQL** is a third party, open-source database server that is widely used and widely available. **Oracle** is a third party, commercial database product. (The CatDV server would in principle work equally well with other SQL databases with minor code changes but at this time has only been converted to work with MySQL and Oracle).
- The **CatDV Server** is a server program that is specific to CatDV. It runs on the server machine (usually the same machine as the database server is running on) and communicates with the MySQL or Oracle server process to access the

database. The **Workgroup** and **Enterprise Server** share the same code but are controlled by which license key is installed.

- The CatDV Pro **Client Application** is a separate desktop application that runs under Windows, Mac OS 9 or Mac OS X and connects to the CatDV Server via a local area network. You use the Server menu commands in the client application to communicate with the CatDV server.
- CatDV **Live HTML Publisher** is an optional Java JSP- and servlet-based web application that runs within an application server such as Apache Tomcat to provide access to the CatDV database from a web browser. It is possible to customize the appearance and functionality of the web application, but in general the capabilities of the web interface are much more limited than those of the CatDV client application with the CatDV server.
- **Apache Tomcat** and **Oracle OC4X** are third party web application servers. You only need one of these if you want to use the Live HTML Publisher. OC4X is included as part of a normal Oracle installation while Apache Tomcat is free, open-source software available for a wide variety of platforms.
- **Java** is a programming language and environment for writing cross-platform software applications. It is pre-installed as part of the operating system software on many platforms but, if necessary, you can download a version for your platform for free. The CatDV Server, the Client Application and the Live HTML Publisher all require Java.

Thus you may need to download and install various third party software in addition to the CatDV server software from Square Box Systems Ltd. Full instructions and URLs are included in this documentation however.

Installing database software for the first time consists of downloading and installing the MySQL or Oracle server software, running scripts to set up a user account and permissions, and then creating the database or databases you want. CatDV is one such database but the same database server software can simultaneously provide many other separate databases for completely unrelated applications. Once you have performed the *one time* installation process you also need to make sure that the database server process or "daemon" is started *each time* the machine is restarted.

Similarly, for the CatDV Server you need to perform one time installation steps (unpack the archive containing the release notes you are now reading, place the CatDV server files in an appropriate directory location, and run a script to create the CatDV database) and ensure that the CatDV Server program starts each time the server machine is rebooted.

1.3 Skills Required

To install the MySQL or Oracle database and the CatDV Server on a machine you will require system administrator privileges and a working knowledge of administration procedures for your chosen operating system.

Remember that setting up *any* server (eg. a mail server, web server, file server, etc.) is typically a more specialised activity than installing a desktop application. It may require additional skills (such as familiarity with editing XML files or use of the command line) to those needed for day-to-day tasks.

In particular, some exposure to using the command line would be helpful as you may need to use this in certain situations (primarily for troubleshooting). Under Mac OS X you need the Terminal application (in Application > Utilities) and under Windows the MS-DOS Prompt, available via the Start button.

The CatDV Server is cross-platform software, originally developed for Unix-type platforms, so the installation process may appear slightly unfamiliar to you if you are primarily used to desktop applications under Mac OS X or Windows.

If you encounter any problems, please consult your local system support group, review any of the many online resources for MySQL, Java and Apache Tomcat, or contact support@squarebox.co.uk.

1.4 CatDV Server License Key

For the CatDV Server to operate correctly you require a server license registration code. This code is emailed to you when you purchase a Workgroup or Enterprise Server product. We can also issue time-limited evaluation licenses on request.

The registration code is stored in a text file called catdv.properties. This file also holds other configuration information, such as the database you are using. The same file is used to configure both the CatDV Workgroup or Enterprise Server and the Live HTML Publisher. The CatDV Control Panel can be used to edit this file (or you can use a text editor), either to update the registration details or to change the server configuration.

Please contact sales@squarebox.co.uk if you do not have your server registration code.

1.5 Structure of this Document

This document contains both release notes and an installation manual. It is structured as follows:

1. Introduction (common to all products and platforms)
2. Architectural Overview
3. Installing CatDV Server for MySQL (Mac OS X, Windows and generic Unix)
4. Installing CatDV Server for Oracle (Windows)
5. Installing CatDV Live HTML Publisher (Mac OS X, Windows and generic Unix)
6. Upgrading from earlier versions
7. Using the CatDV Pro client application with the workgroup server
8. Using and customising the Live HTML Publisher

1.6 Version History

The version history has moved to chapter 6, together with notes on updating an existing CatDV Server installation.

1.7 Contacting Us

For registration queries please email sales@squarebox.co.uk or visit our web site http://www.squarebox.co.uk. For technical queries email support@squarebox.co.uk. Please also check the sales and support FAQ at http://www.squarebox.co.uk/faq.html if you have any support queries. This is regularly updated with late-breaking news and has a section on server-related issues.

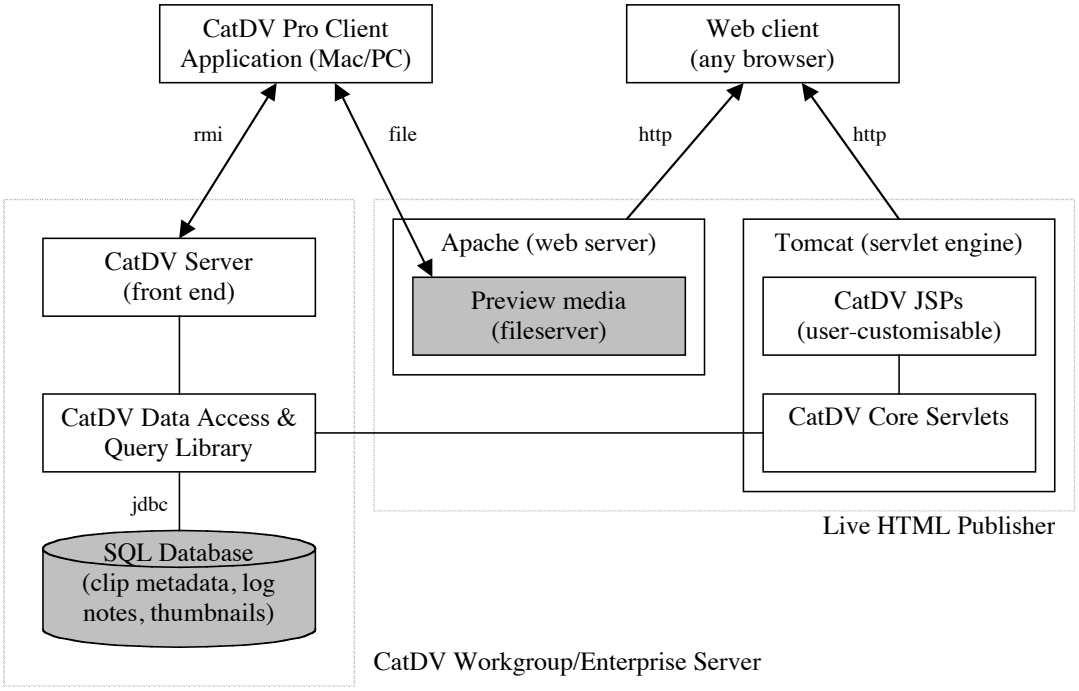
Please *do* contact us if you sport any inaccuracies in this document or if anything isn't clear! (Third party URLs in particular do change from time to time and we need your help to keep this document up to date.)

2 Architectural Overview

The CatDV Pro *client application* is a standalone desktop application, providing video logging and media cataloging capability and available for both Macintosh and Windows PCs. It is based around a catalog or database of clips, where each clip represents either a media file (identified by filename) or a scene within a movie file or video tape (identified by tape name and timecode values). Each clip is annotated with a rich collection of metadata, including name, log notes, date of recording, a thumbnail image, video and audio format, and so on, which can be used when searching for clips or to provide a record of the contents of a tape even if it's not currently online.

In addition to its well-established standalone mode of operation (whereby files and catalogs are stored on the local file system), the CatDV Pro application can also operate as a network client to an optional CatDV Pro Server.

The *CatDV Server* is a separate program, running on a server machine and accessible over a local area network from one or more client machines. The server stores the clip metadata in a relational database, rather than a local flat file. This allows the clip database to be searched and updated by different clients concurrently, and permits much more sophisticated searches to take place.



Using a CatDV Pro client communicating with the CatDV server, users can search for existing clips in the database, update these clips (eg. by adding further log notes) and create new clips in the database (eg. by scanning a new tape, importing new media files, or by creating secondary clips from existing clips). Exactly the same operations, with exactly the same user interface, can be performed as when CatDV Pro is being used as a

standalone application, the only difference being that the clips are saved and loaded over the network into a shared database.

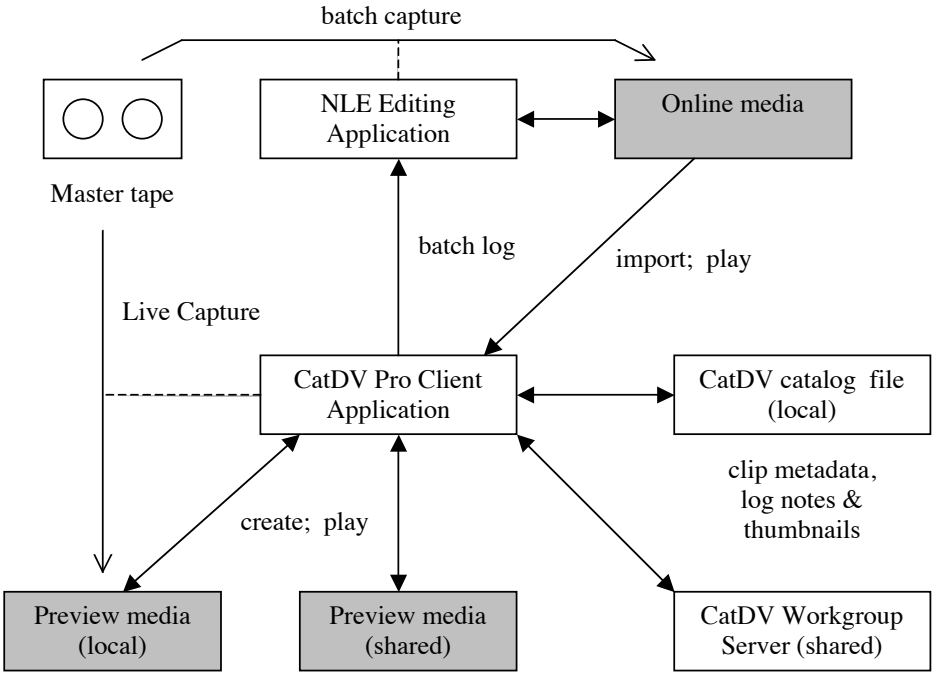
With the *Live HTML Publisher*, the same clip data is further made available through a web interface, allowing it to be viewed from any web browser without requiring the CatDV client software to be installed on that machine. The database can be browsed or searched but in the current implementation this interface is read-only – unlike with the full client application it is not possible to input data via the web interface.

2.1 Managing Media

The CatDV Server does not itself store any media, either full quality on-line media or low-resolution preview versions. Instead, the database primarily deals with metadata *about* the media, such as clip and tape name, video and audio format, log notes, thumbnails, and media file location. This metadata allows clips to be searched and identified, even if the media is currently unavailable.

Once the desired clips have been located using this metadata, the client application can then request the corresponding media. If the media is available online on the local disk this will be via the filename, and the media can be played within CatDV. Alternatively, the user can export a batch capture log of selected clips so the media can be recaptured from tape at full online quality and used within your non-linear video editing software (NLE).

The CatDV Pro Client Application can also create and manage low-resolution *preview* versions of original media. These previews act as “video thumbnails” to give a rough indication of the contents of a clip for those situations when the original full-resolution media are not available online. Although primarily intended for previewing clips within CatDV Pro, the preview files can also be exported and used as low-resolution video editing proxies if so desired. Previews are referenced using a search algorithm based on the tape name and timecode value of the clip, rather than a fixed file or pathname, which means they don’t need to be explicitly associated with a clip but are available for any clip referencing that tape.



Previews are normally stored locally on the client machine, but they can also be shared via the file system by configuring the preview directory to be on a remote volume. However, even though the preview movies are by their nature compressed and much smaller than the online media files, careful thought still needs to be given to both fileserver and network performance if the preview media are to be stored on a remote volume.

If previews are stored on a file server then they can also be made available to web clients via a web server. Again, careful consideration needs to be given to the performance requirements. CatDV does not aim to be a media server by itself but will support this configuration if used in conjunction with a suitable media file server. It is also possible to configure the Tomcat application server to serve up media files if desired.

2.2 Interoperability

As far as possible, CatDV Pro is based around non-proprietary, open standards. The client application is written in Java, and uses Apple's QuickTime technology, ensuring that it's available for both Macintosh and PC. The workgroup server and HTML publisher also use Java, and as a result are available for a wide range of server platforms.

By default, the CatDV Server uses MySQL as its database. MySQL is a widely available open-source database, available for platforms including Unix, Mac OS X, Windows NT/2000/XP, and Solaris (and included as standard with many Linux distributions). The CatDV Workgroup or Enterprise Server software communicates with the MySQL database using JDBC drivers. By using this open standard it is possible (with some minor code changes) to install the CatDV database on other vendors' database products rather than MySQL. Currently, Oracle is supported in addition to MySQL. JDBC also permits the database to be run on the same or on a different machine as the workgroup server.

Similarly, the HTML publisher uses Java servlets, and normally runs within the open-source Apache Tomcat servlet engine, which again is freely available for Windows, Mac OS X and other Unix platforms.

Preview media can be integrated into the web pages returned by the Live HTML Publisher by configuring a web server to return the appropriate files. This might typically be the open-source Apache web server, but other web servers such as Microsoft IIS or dedicated proprietary media servers can be integrated into the system also, as long as the Live HTML publisher is configured to return the appropriate URLs.

To maximise portability and interoperability, CatDV is developed with deliberately conservative assumptions about the platform and libraries it is being run against. Although extensively tested against the latest available versions of third party software (including beta and developer preview versions), care is taken not to rely on later features unless a compelling reason for their inclusion is found. By not depending on the "latest and greatest" (and possibly unstable) versions of software such as Java, QuickTime, MySQL or Tomcat, we can maximise the number of systems on which the CatDV product family will operate.

2.3 Toolkit Philosophy

The approach taken with the design of the CatDV Server software is to use open standards and maintain maximum interoperability and flexibility. While the CatDV Pro client application is a shrink-wrapped end-user application, the server software, especially the Live HTML Publisher, can be viewed as being more of a toolkit, designed to be customised and integrated with other systems as part of the user's overall workflow and system infrastructure. It is even possible to license the source code for the CatDV workgroup server itself, permitting CatDV to be hooked in to an existing back end media management system, for example, instead of using its own database.

How to customise the JSPs that make up the user interface of the HTML publisher is described later in this document.

3 Installing CatDV Server for MySQL

This chapter describes how to install the MySQL database software and the CatDV Workgroup or Enterprise Server under Mac OS X, Windows and generic Unix-like systems for the first time. Please refer to the separate instructions later in this manual if you are upgrading an existing installation.

3.1 Requirements

The CatDV Server for MySQL requires the following:

- Operating system: Linux, Windows XP, or Mac OS X (10.3 or later), including Mac OS X Server
- 1024MB RAM minimum
- MySQL 4.1 or later (MySQL 5.1 recommended)
- Java JDK 1.4.2 or later (Java 1.6 recommended)

To install MySQL and the CatDV Server on a machine you will require system administrator privileges and a working knowledge of administration procedures (including exposure to use of command line tools) for your chosen operating system. If necessary please consult your local system support group.

3.2 MySQL Installation

Before you can install the CatDV Server you need to install MySQL database server software on your server machine as described below.

Please note that MySQL is an open source database available for download and installation at no cost. However, if you are using it for commercial purposes you should purchase a basic MySQL network license and support. The cost of the MySQL license is very reasonable and doing so will both support the MySQL development efforts and provide assistance if you should need it for the database itself.

Mac OS X Server

If you use Mac OS X Server (10.3 or later) you can skip the installation step as MySQL is already installed by default. However, you do need to make sure MySQL is running and that network connections are allowed:

1. Under 10.4 and earlier, launch Applications > Server > MySQL Manager

Alternatively, under 10.5 and later, launch Applications > Server > Server Admin.app go to the Services tab, and enable MySQL.

2. Check the “Allow network connections” box and press Start MySQL. If MySQL is already running you will need to stop it first, then restart it again afterwards.

Mac OS X

If you don't already have MySQL on your machine download the MySQL Community Edition from www.mysql.com. You need Adminstrator privileges to perform these once only steps:

1. Visit http://dev.mysql.com/downloads/mysql/5.1.html#macosx-dmg and download the 5.0.51b (or later) Installer Package installation appropriate for your Mac OS X version (there are separate downloads depending on your process and Mac OS X version). Select a 'Standard' release. Click "Pick a mirror" to download the software.

2. Double click the .dmg file to open the disk image. (Under Mac OS X 10.4 Safari will open the disk image automatically once it has downloaded.)

3.When you double click the .dmg file this will open a disk image with two installation files, mysql-standard-5.XX.pkg and MySQLStartupItem.pkg. The first is the MySQL database software itself (which will be installed in /usr/local/mysql). The second installs a startup item (in /Library/StartupItems) to start MySQL automatically each time your computer is started. We recommend you install both, by double clicking on them in turn and following the installer instructions, though you can omit the startup item and launch MySQL manually if you prefer.

4. MySQL also comes with a convenient System Preferences item to start and stop the server. Create a folder /Library/PreferencePanes if you don't already have one, then copy MySQL.prefPane to /Library/PreferencePanes.

5. To start the MySQL server the first time open the MySQL item in System Preferences. Alternatively, start the Terminal application and type the following

```
sudo /Library/StartupItems/MySQLCOM/MySQLCOM start
```

entering your administrator password when requested.

6. Make sure MySQL is installed correctly and you can type "/usr/local/mysql/bin/mysql" to start the text-based client program to access the database. If you successfully see the "mysql>" prompt type "quit" (or press Ctrl-C) to exit mysql and return to the command shell and then proceed to "CatDV Server Installation" (section 3.4 below).

7. Optionally, you can set a password for the MySQL root user (which is different from the 'root' user under Mac OS X or Unix) by typing the following in Terminal:

```
/usr/local/mysql/bin/mysqladmin –u root –password secret
```

where *secret* is the password you want to set.

Creating a my.cnf config file (Mac OS X and Mac OS X Server)

A MySQL installation has a large number of tuning and other configuration parameters that can be set by means of a my.cnf configuration file. Unlike Windows, where the installer automatically creates a my.ini config file, by default a MySQL configuration under Mac OS X won't automatically have a config file, which means that MySQL will run with default values for its configuration.

If you don't already have a my.cnf config file then it's a good idea to create one so you can set up various MySQL parameters as required. The easiest way to do this is copy an existing template file to /etc/my.cnf .

When you launch the CatDV Control Panel it will offer to install a default config file for you.

You can also install a config file manually, either coping my-sample.cnf from the CatDV installation directory or using one of the templates provided by MySQL itself (/usr/share/mysql/my-large.cnf under Mac OS X Server, /usr/local/mysql/support-files/my-large.cnf under Mac OS X). Use Terminal.app to run the following command and enter your password when prompted to do so by the sudo command:

```
sudo cp templateFile /etc/my.cnf
```

Note that MySQL looks for its config file in a number of places and while /etc/my.cnf is the standard location your site may already have a config file elsewhere. Consult your administrator if you're not sure.

Windows

If you don't already have MySQL on your machine download the MySQL Community Edition from www.mysql.com.

1. Download and launch mysql-essential-5.1.XX-win.msi (or later) Windows Essentials (x86) from http://dev.mysql.com/downloads/mysql/5.1.html#win32
2. Run the "Essential" installer to install MySQL (eg. in C:\Program Files\MySQL Serer 5.1). Run "Configure Installation" when the wizard prompts and choose a standard configuration.
    a. Check the "Install as Windows Service" and "Include Bin Directory in Windows PATH" options.
    b. On the next page, choose "Modify Security Settings" and enter a new root password such as *secret.* (Replace '*secret* with your actual chosen password.)
    c. If you prefer, you can choose a Detailed Configuration, in which case it is suggested that you choose the following options:
        - Server Machine
        - Non-Transactional Database Only
        - Decision Support (DSS)/OLAP
        - Enable TCP/IP Networking (port 3306)
        - Do *not* Enable Strict Mode (you can enable this option if you want but you might get errors if you try to save data with excessively long clip names etc.)
        - Best Support for Multi-lingualism
3. Alternatively, if you need to use an older version of MySQL 4.1 which doesn't have the "Essential" installer you will need to do some extra manual steps:

a. Install MySQL to a location such as C:\mysql. (This directory will contain both the programs that MySQL uses and the database files for CatDV and any other databases.) Select a typical install.

b. Assuming you used C:\mysql, add C:\mysql\bin to your PATH. (My Computer > Properties > Advanced > Environment Variables > System Variables)

c. Run winmysqladmin.exe (from C:\mysql\bin) to start MySQL as an NT service. You will be prompted to enter a username (choose 'root') and password for the MySQL administrator user. Enter a password (eg. *secret*) and note this for when you install the Workgroup Server.

d. Alternatively, open a DOS prompt and type "mysqladmin -u root password *secret*" to set a password for the MySQL root user. Replace '*secret* with your chosen secret password.

4. Make sure MySQL is installed correctly and you can type "mysql" to start the text-based client program to access the database. If you successfully see the "mysql>" prompt type "quit" (or press Ctrl-C) to exit mysql and return to the command shell and then proceed to "Java Installation" as described below.

Linux/Generic Unix

1. Many Linux installations come with MySQL pre-installed. If not, please visit http://dev.mysql.com/downloads/mysql/5.1.html and download and install the appropriate version for your platform (including Linux, Solaris, FreeBSD, AIX etc.)
2. Follow the instructions provided with MySQL to install a binary distribution, set appropriate permissions, and start the server.
3. Make sure MySQL is installed correctly and you can type "mysql" to start the text-based client program to access the database. If you successfully see the "mysql>" prompt type "quit" (or press Ctrl-C) to exit mysql and return to the command shell and then proceed to "CatDV Server Installation" as described below.

3.3 Java Installation

The CatDV Server is written in Java, so you need to ensure you have Java installed on your machine.

Mac OS X / Mac OS X Server

No action is required as Java is automatically installed as part of the operating system. You may want to use Software Update to ensure Java is up to date.

Windows

If you already have Java 1.3.1 or later installed you need take no further action. Otherwise, please visit http://java.sun.com/javase/downloads/index.jsp and download the J2SE Development Kit (JDK) version 1.6 (also referred to as Java 6). Please follow the instructions given by the installer.

(Note: while the CatDV Workgroup Server is compatible with both the JRE runtime environment and the JDK development kit, it is better to install the JDK in case you want to use the Live HTML Publisher.)

Assuming you installed Java to a directory such as c:\j2sdk1.5.0_04 then add the corresponding bin directory c:\j2sdk1.5.0_04\bin to your PATH (My Computer > Properties > Advanced > Environment Variables > System Variables)

Generic Unix/Linux

Java is likely to be pre-installed in /usr/java as part of the operating system. If not, please visit http://java.sun.com and download the J2SE Development Kit for your platform.

3.4 CatDV Server Installation

Once you have installed and started the MySQL database software you need to install the CatDV Server software. As part of the installation you will create a 'catdv' database in MySQL.

Unpack the CatDVServer.zip file to a temporary location if you have not already done so. (If you are reading these release notes you probably have done so!)

Mac OS X (including Mac OS X Server)

Double click on CatDV Installer.pkg (in install/macosx) and follow the instructions, entering your administrator password when prompted. This will create the catdv database, install the CatDV workgroup server in /usr/local/catdvServer and install a startup item to launch it automatically when the system is restarted.

For troubleshooting, or if you have special requirements, you can ignore the installer package and manually install the server under Mac OS X as per the generic Unix instructions below. This is *not* recommended for most users however.

You need to edit the catdv.properties configuration file and paste in your license code before starting the workgroup server. As part of the Mac OS X installation process the CatDV Control Panel will be launched to help you do this – see sections 3.5, 3.6 and 3.7 for further details on entering your registration code and starting the server.

Windows

Double click the installCatDVServer.bat batch file from the install\mysql_win folder.

This will prompt you for the following information:

- the location to install the software, eg. C:\Program Files\Square Box\CatDV Server.
- the MySQL root password you set earlier, if any (eg. *secret*)
- whether to install the workgroup server so it starts automatically (you should say yes)

(Under Windows Vista you need to right-click the batch file and Run As Administrator. You may also be asked to enter the location of the batch file.)

The installer batch file will create a catdv database in MySQL and install the workgroup server as an NT service so it starts automatically when you reboot.

You need to edit the catdv.properties configuration file and paste in your license code before starting the workgroup server. As part of the installation process the CatDV Control Panel will be launched to help you do this – see sections 3.5, 3.6 and 3.7 for further details on entering your registration code and starting the server.

(You can use launchServer.bat to launch the CatDV server manually if you prefer. Running launchServer from a DOS prompt may be useful during initial troubleshooting as it is easier to see any error output on the console.)

When you first launch the server, Windows may say that it is protecting your computer by blocking this Java program. Select Yes to allow the server to run.

Manual Windows installation (optional)

For troubleshooting or if you have special requirements you can ignore the installer script and manually install the server under Windows. This is not recommended for most users.

1. Unpack the CatDVServer zip file using WinZip or similar application direct to the location you want the server to run from, for example create a directory " C:\Program Files\Square Box\CatDV Server " or similar as the destination. If using WinZip, create a new folder with this name and make sure you have the “Use folder names” option checked.

2. Open a DOS command window, then type:

```
cd " C:\Program Files\Square Box\CatDV Server "
mysql -u root -psecret < install\mysql\create_catdv.sql
```

(Here *secret* is the MySQL root password you set earlier, with no spaces after –p. Omit the –p option if you didn’t set a password.)

3. Edit the launchServer.bat script (in install\mysql_win) with a text editor, checking the path to the Java bin directory is correct.

4. To launch the server type launchServer.bat (or double click the file icon), but see the section “Installing your registration code” before trying to start the workgroup server.

Generic Unix/Linux

1. Decide where the CatDV Server files (ie. the contents of the directory that contains the release notes you are now reading) will go. This will normally be /usr/local/catdvServer, though you can use another path if you choose. It is helpful if this path does not contain any spaces however.

2. Open a terminal window and change directory to the CatDV server directory, eg. type "cd /usr/local/catdvServer". Type "ls" and make sure you have the following files: catdv_initd, Software License.txt, Release Notes.pdf, launchServer, launchServer.bat, java.policy, sample.properties, plus lib and install sub-folders. The lib directory contains

various jar files which comprise the server software itself, while install contains SQL installation scripts as well as Mac and Windows specific files which you can ignore.

3. Perform the following one-time initialisation step to create the CatDV database:

mysql -u root -p*secret* < install/mysql/create_catdv.sql

Type spaces exactly as shown, ie. after -u but not after -p. *secret* is the database root password you chose earlier. (If there is no MySQL root password omit the –p option)

4. Ensure you can type the following to access the CatDV database:

mysql -u catdv -pcatdv catdv

5. Type "quit" to exit MySQL then proceed to "Installing your registration code" below. You will need to create a catdv.properties file containing your license code and start the server using the launchServer script. You may need to make the launchServer script executable by typing “chmod +x launchServer”. If there are any errors check the contents of the server.log file.

6. After setting up and testing the CatDV server you can set it up to run automatically whenever your system starts. Under Linux, one way of doing this is as follows:

- Log on as root and open a shell
- Type “cp catdv_initd /etc/rc.d/init.d/catdv”
- Type “chmod +x /etc/rc.d/init.d/catdv”
- Type “chkconfig catdv reset”
- Reboot the machine and verify that the CatDV server process starts

3.5 Installing your Registration Code

The operation of the server, including how many client connections are permitted, is controlled by a license key or registration code. You will have been mailed two lines of text such as the following example. (Please contact sales@squarebox.co.uk if you do not have your registration code.)

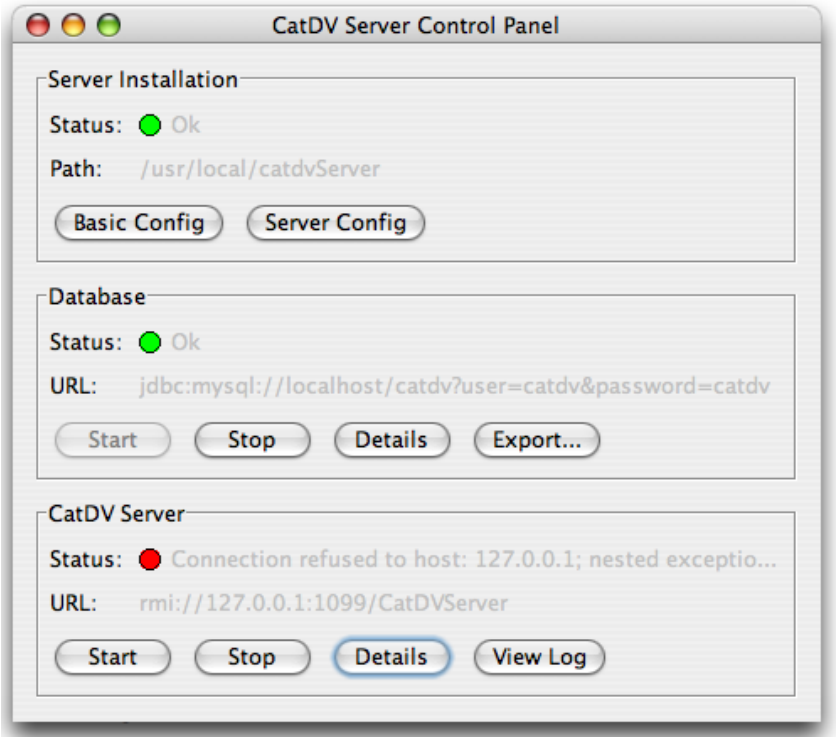
reg.user = Trial
reg.code = OGOP-JZXL-G1TD-65SI-7F7B-YZ3X-GVJD-6YQQ

The registration code is stored in the catdv.properties file in the CatDV server directory. If you want to you can edit this file with a text editor (copy and paste the lines you were sent into the catdv.properties file, replacing the example lines already there). It is normally easier to use the CatDV Control Panel to edit the configuration file however. For details see the next section.

The catdv.properties configuration file also contains other configuration information, most notably the connection URL for the database. It also contains configuration for the Live HTML Publisher but you can safely ignore these lines if you do not use the HTML Publisher. If you make any changes to catdv.properties you need to stop and restart the workgroup server for the changes to take effect.

3.6 Using the CatDV Control Panel

To configure the CatDV Server, check its status and control its operation you can use the new CatDV Server Control Panel:



The CatDV Control Panel can be launched in various ways as follows

- The control panel should be launched automatically as part of the installation process under Mac OS X and Windows
- Under Mac OS X, launch CatDV Control Panel.app from your Applications folder
- Under Windows, double click ControlPanel.bat from the CatDV Server directory (eg. C:\Program Files\Square Box\CatDV Server)
- Navigate to the CatDV Server directory then double click server.jar (an executable Jar file) in the lib subdirectory (or, from the command line, type "java –jar server.jar")

Once the control panel is running traffic light indicators show the status of the server: whether the server installation path looks ok, whether MySQL is running and the catdv database is accessible, and whether the CatDV server itself appears to be running.

Press "Basic Config" and enter the path if the server is installed somewhere other than the default location. When you first launch the control panel, and after editing the basic config, the database is checked to make sure it is running the correct schema version. If necessary, the schema will be updated automatically (after confirmation).

Once you have a green light for the Server Installation status, press “Server Config” to edit the catdv.properties file. Enter your registration code and user name, and any other parameters as required, then press Save.

The MySQL Database status shows whether the catdv database is accessible using the database URL specified in the server configuration. Once configured properly you should not normally need to Start or Stop the MySQL Server manually.

The CatDV Server status shows whether the control panel can make an RMI connection to the server, in the same way that the client application does. If you have problems you may to stop and then re-start the CatDV Server. “View Log” will display the server.log file that contains useful diagnostic information when the server is running.

3.7 Starting and stopping the CatDV Server

If configured as described above, under Mac OS X and Windows both MySQL and the CatDV Server should start automatically when the server machine is restarted and stop when the machine shuts down. However, during initial set up, for troubleshooting, and when running under Linux or Unix, you may need to start the server manually:

Starting the CatDV Server the first time

- Use the CatDV Control Panel as described above
- Alternatively, under Mac OS X, either restart your computer, or open a Terminal window (Applications > Utilities > Terminal) and type:
  sudo /Library/StartupItems/CatDVServer/CatDVServer start
  entering your administrator password when requested.
- Under Windows, bring up Start Menu > Control Panel > Administrative Tools > Services and start the CatDV Server service

Running the CatDV Server manually

If some problem is stopping the server from starting normally, to aid diagnosis you can run the server from a command prompt so you can see any error messages which may be displayed:

1. Launch a terminal window and change directory to the CatDV Server, eg. cd /usr/local/catdvServer , or cd C:\Program Files\Square Box\CatDV Server.

2. Under Mac OS X or Unix type the following to start the server running: ./launchServer. Under Windows, run launchServer.bat.

If it launches without error you should then see a message similar to the following:

```
Creating RMI Registry on port 1099
…
(various Java system settings displayed for diagnostic purposes)
…
Catalogs: 0
Clips: 0
Thumbnails: 0
```

SourceMedias: 0
ImportSources: 0
Max ID: 1
DB Schema: 1.03
Running: MyMacName.local

You are now ready to go and can now start the CatDV client application (from any machine on your local network) and connect to the server from the client application "Server" menu by entering the hostname or IP address of the server machine.

3. Leave the server program running and minimise or hide the terminal window. You can stop the server by opening this window and pressing Ctrl-C and can monitor any error messages by viewing the contents of the window.

Stopping the CatDV server

Details of how to shut down the CatDV server depend on how you started it:

- If you are using the CatDV Control Panel you can stop the server from there
- If you are logged on from a CatDV Pro client, you can shut down the server remotely from the Server Admin Panel under the "Server" menu
- If you started it manually from a terminal shell, press Ctrl-C
- If you started with the NT service, bring up Start Menu > Control Panel > Administrative Tools > Services and stop the CatDV Server service (Windows)
- If you started it with a Mac OS X startup item, type

  sudo /Library/StartupItems/CatDVServer/CatDVServer stop

Stopping the MySQL server

In the unlikely event you need to shut down the MySQL server software you can do so. This similarly depends on how you launched it:

- Press the MySQL Stop button in the CatDV Server Control Panel
- Type "/usr/local/mysql/bin/mysqladmin shutdown" (Mac OS X or Linux)
- Type "sudo /Library/StartupItems/MySQL/MySQL stop" (Mac OS X startup item) (or use the MySQL System Preferences item if you installed it)
- Use the Windows "Services" control panel (or type "mysqladmin –u root –p*secret* shutdown" from a DOS prompt)

3.8 Troubleshooting

MySQL problems

- If mysql_install_db gives an error about hostname use the --force option.
- If you get errors when starting the MySQL server, look at the .err files in the data directory for details.

- Note that "root" means the mysql administrator, ie. the name of a database user as understood by MySQL, not the Unix root login user as understood by the operating system. On the other hand "mysql" *does* refer to a login user account. This can be somewhat confusing, so pay careful attention to the context so you know what is being referred to.
- If you get an SQL error "Server configuration denies access to data source" the most likely reason is a DNS issue and your machine is not 'localhost'. Make a note of the hostname after the '@' then type the following (inserting the actual hostname as appropriate):

  mysql -u root -p*secret (where secret is the MySQL root password)*
  GRANT ALL PRIVILEGES ON catdv.* to catdv@hostname ;
  GRANT RELOAD,SHUTDOWN on *.* TO catdv@hostname;
  FLUSH PRIVILEGES ;

  Press Ctrl-C to come out of MySQL and run ./launchServer again.
- If you are unable to determine which hostname to use, then, as a last resort for testing purposes, you could try:

  GRANT ALL PRIVILEGES ON catdv.* to catdv@"%" ;
  FLUSH PRIVILEGES ;

  but beware, this leaves a gaping hole in your database security and once you have identified the issue you should fix the problem.

Mac OS X-specific issues

- Under Mac OS X 10.2 and later, /usr/local/bin/ is no longer in your path and you will need to explicitly type /usr/local/mysql/bin/mysql instead of just mysql. You also need to download the special Jaguar release of MySQL..
- With Mac OS X 10.3 and later you no longer need to create a mysql user. If you are using an older version of Mac OS X and don't have a mysql user you can create one using the Users pane of System Preferences. You may also want to consult http://developer.apple.com/internet/macosx/osdb.html.
- If the CatDV Mac Installer reports a failed installation, do File > Show Log, then select "Show Everything". The most common cause of errors is that MySQL is not installed or is not running.
- If you are using Mac OS X 10.5 and start MySQL using the CatDV control panel then MySQL will inherit your own personal tmpdir temporary directory which it doesn't have access to. This can cause the CatDV client to fail with error messages such as "prepareQuery failed (Can't create table 'tmpclips') (errno:13))" when you try to perform a query. To resolve this, make sure you have a my.cnf config file which explicitly specifies a tmpdir to use (see 3.2 above).

Windows XP firewall settings

- If you can connect to the server with a client running on the same machine but not from another machine on your network then access to the server is probably being blocked by the Windows Firewall software.
- Open the Windows Firewall settings (Start menu > Control Panel > Security Settings > Manage security settings for Windows Firewall). It is recommended that you leave the firewall settings ON, but permit access to the CatDV server using the Exceptions tab:
    - Click on Add Program… then browse to the directory where you installed the CatDV server (eg. C:\Program Files\Square Box\CatDV Server 5.0) and select JSHelper.exe
    - Click on Add Port… and enter "Java RMI" for the Name, "1099" for the port number, and check the TCP button.

Other Windows-specific issues

- When installing on Windows 2000 the installer may fail saying command REG not found. This is because you may not have a copy of the registry editor REG.EXE in your path. If so, you may need to copy REG.EXE from another machine (contact support@squarebox.co.uk if you need further assistance).
- Check the CatDVServer logs in C:\TEMP if the server fails to start
- Try launching the CatDV Server manually from the NT Services control panel. If it fails there the message may display additional useful information. Check the properties of the service to make sure things like the directory it's running from are what you expect.

CatDV Server troubleshooting

- Launch the CatDV Server Control Panel and press the Details and View Log buttons to display diagnostic information.
- If the server does not start automatically using the NT service or OS X startup item, try launching it manually from the command line as this may make it easier to see any error messages that are displayed.
- For testing purposes you may find it convenient to install the CatDV Pro client application on the server machine. It is not necessary to do so however, and you do not need to install QuickTime on the server machine for the Workgroup Server software to function.
- The most likely cause of the server not starting is that you haven't inserted a valid registration cause into the catdv.properties file.
- Other common causes are that the MySQL database isn't running, that you didn't create a catdv database, or that the RMI port 1099 is already in use by another application (in which case pick a different port in the catdv.properties file).

- If you can connect to the server with a client running on the same machine but not from another machine then one possible explanation is a firewall issue. Another, more insidious, possibility relates to the way Java RMI works. When you enter the IP address of the server in the Connect To Server dialog on the client you are actually specifying the location of a naming registry which returns the actual CatDV server location. In order for this to work, the server needs to be told *its own* IP address as used by clients when connecting to it. In certain situations (such as if the server has multiple network adapters) the naming service may not correctly identify its own IP address, so clients will be unable to connect properly. To resolve this issue, use the CatDV Control Panel to edit the Server Config and enter the server's own IP address under Host & Port. (If you edit the catdv.properties file then you need to set java.rmi.server.hostname.)

Log files

- A log file called server.log is often written in the server directory. This contains diagnostic information. You can also view this log file from the CatDV Control Panel.
- Under Windows, you may find c:\temp\CatDVServer.stdout.log and c:\temp\CatDVServer.stderr.log files that also contain useful information.
- Under Mac OS X the console.log and system.log files, accessible via the Application > Utilities > Console application, may also contain useful diagnostic information.

3.9 Miscellaneous Topics

MySQL backups

To backup the entire CatDV database to a single flat file shut down the CatDV server and use the commands:

```
mysqldump -u catdvadmin -pcatdv catdv > backupXXXX.sql
gzip backupXXXX.sql
```

where backupXXXX.sql is a filename you choose (perhaps including today's date), and gzip is used to compress the file, which would otherwise be quite large (you can use other compression tools or omit this step if you prefer).

The CatDV Control Panel has a convenient command to simplify exporting the CatDV database using the mysqldump program.

MySQL gives you other options for backing up the database files as well. You can directly backup the MySQL data files (in /usr/local/mysql/data/catdv, /var/mysql/catdv or C:\mysql\data\catdv or a similar location) using your regular backup utility, or consult the MySQL documentation for details of other backup strategies (such as writing a transaction log to a remote file system).

IMPORTANT: Because it needs to fit in with your existing backup strategy please note that the CatDV server does **not** perform any backups automatically by itself. It is your

responsibility to ensure that you create regular and adequate backups that meet your requirements. You should also test that your strategy works by performing a restore before you rely on it.

Restoring database or moving MySQL to another machine

If you need to migrate the CatDV server and database to another machine install MySQL and the CatDV Server on the new machine then shut down MySQL on both machines and copy the MySQL data files from one machine to the other. The data files are typically in the directory in /usr/local/mysql/data/catdv, /var/mysql/catdv or C:\mysql\data\catdv or a similar location, depending on where you installed MySQL. Never simply overwrite an existing catdv data directory, always rename it first (eg. to catdv_old) in case you make a mistake!

You can also copy the catdv database to another machine by loading a backup file. The instructions are similar for restoring a database from a backup. First, shutdown the MySQL server, then locate any existing catdv data files (in /usr/local/mysql/data/catdv, /var/mysql/catdv or C:\mysql\data\catdv or a similar location) and rename the existing directory if it exists to catdv_old. You can then run the backup script to recreate the catdv database and load the data:

mysql –u root –p*secret* catdv < backupXXX.sql

where backupXXXX.sql is the name of your backup file and *secret* is the MySQL root password if any. Warning: restoring from a backup script will silently delete any existing catdv database at the destination, so you should always backup the catdv data directory first!

If you are moving the data to a new machine you will need create an empty 'catdv' database first before loading the script (CREATE DATABASE catdv;). Also, if you are moving from a non-case preserving system to a case-sensitive file system (eg. from Windows to Unix) you might need to rename the tables (RENAME TABLE sourcemedia TO sourceMedia; etc.)

MySQL replication

If you require, you set up replication from one MySQL instance to another, ensuring that you have a second copy of the database on another machine which is automatically kept in sync with the first.

This will involve editing your my.cnf or my.ini config file. Under Mac OS X you may need to create the /etc/my.cnf file first, see 3.2 above. Under Windows the file is called my.ini and is normally located in C:\Program Files\MySQL\MySQL Server 5.0. In both cases you should review the existing file and merge in the changes below.

1. On the master machine, edit your my.cnf or my.ini config file and add the following

```
[mysqld]
log-bin=mysql-bin
server-id=1
```

2. On the master, execute the following mysql command:

```
GRANT REPLICATION SLAVE ON *.* TO 'repl'@'%.mydomain.com'
IDENTIFIED BY 'slavepass' ;
```

(You can change the replication user ‘repl’ and password ‘slavepass’ as required. Replace mydomain.com with your domain name.)

3. On the master, execute the following mysql command and note the file and position:

```
FLUSH TABLES WITH READ LOCK;
SHOW MASTER STATUS;
```

4. On the slave machine, edit my.cnf/my.ini and add the following:

```
[mysqld]
log-bin=mysql-bin
server-id=2
```

(You can choose different server-id’s if necessary, the important thing is that the master and slave are different from each other.)

5. On the slave, execute the following mysql command:

```
CHANGE MASTER TO MASTER_HOST='hostname', MASTER_USER='repl',
MASTER_PASSWORD='slavepass', MASTER_LOG_FILE='filename',
MASTER_LOG_POS=filepos ;
```

Replace hostname with the host of the master. Replace ‘slavepass’ (and ‘repl’ if necessary) with the user name and password you set up at step 2. Replace filename and filepos with the values you noted at step 3 (use ‘’ and 4 if not otherwise specified).

6. You can start and stop the slave and check its status using the following mysql commands:

```
STOP SLAVE;
START SLAVE;
SHOW SLAVE STATUS;
```

Refer to the MySQL documentation for further details if required.

MySQL authentication

As set up by default the catdv database user is configured to use a relatively easy to guess password, on the assumption that unauthorised people will not have direct access to the server machine. If you require additional security you should set up a database password for the catdv database user as follows. (Note that the the catdv database password is not the same as the MySQL root password, which applies to all databases, not just the catdv one.)

Prior to setting up the database you can edit the create script and add "IDENTIFIED BY *xxxx*;" to the end of the "GRANT" lines, where " *xxxx* " is your chosen password (for the catdv user, not the MySQL root user). You can also set a password after creating the database by typing "mysqladmin -u catdv password *xxxx* ".

If you set a database password then you will need to edit the catdv.properties configuration file (the easiest way to do this is by launching the CatDV Control Panel). Edit this line to include your chosen password.

catdv.database=jdbc:mysql://localhost/catdv?user=catdv&password=*xxxx*

You can also edit this line to specify a database server running on a different machine from the the CatDV server (ie. not localhost). If you do this you will need to grant database access to a user other than "catdv@localhost" or you will get an access denied error when you try to start the CatDV Server. Refer to the MySQL user manual for more details.

CatDV server configuration parameters

There are two ways of specifying configuration parameters such as the database URL to use: via the catdv.properties file and by specifying a system property in the launchServer batch file with –Dproperty=value. You should avoid trying to set the same property in two different ways as that is likely to be confusing. For reference though, the system property setting takes precedence.

Normally you should use the CatDV Control Panel (see 3.6) to edit these settings and are unlikely ever to need to edit the catdv.properties file by hand.

The following properties are used:

catdv.logfile	Filename of server log file to create
catdv.properties	Filename of properties file to load
catdv.rmiport	Server port to use (defaults to 1099)
java.rmi.server.hostname	The server's own IP address (if the machine has multiple network adapters you need to specify which IP address clients will use to connect to the CatDV Server)
catdv.server	The RMI service name
catdv.database	Database URL to use
catdv.debug	Set to true (and restart the server) to display additional debug information in the server.log file
catdv.allowAnonymous	Set to true if you are using Workgroup edition clients connecting to the Enterprise server and want to allow them to create and edit catalogs in the default system group without logging on to the server.
catdv.terminationFile	Filename of a file which, if present, signals the server should terminate
catdv.jdbc.class	Java class name of the the JDBC driver class
reg.user	Registered user name (license details)
reg.code	Registration code (license details)
reg.info	Human-readable description of license (license details)

reg.expiry	Human-readable description of license (license details)
preview.dir	Filename of preview root directory (HTML Publisher)
preview.url	Corresponding URL (HTML Publisher)
media.dir	Filename of media root director (HTML Publisher)
media.equiv	Equivalent media path prefix (HTML Publisher)
media.url	Corresponding URL (HTML Publisher)
timestamp.format	How dates and times are formatted (HTML Publisher)
web.user	Which CatDV user the web publisher uses to access Enterprise server. Determines which catalogs will be accessible via web interface (HTML Publisher)

4 Installing CatDV Server for Oracle

This chapter covers the installation of the CatDV Enterprise Server for Oracle. This is available for Windows only at present.

4.1 Before You Start

Requirements

The CatDV Enterprise Server for Oracle currently requires the following:

- Operating system: Windows 2000 or Windows XP
- 512MB RAM minimum, 1024MB or more recommended
- Oracle 10g (although it may work with Oracle 9, this is untested)

To install Oracle and the CatDV Server on a machine you will require system administrator privileges and a working knowledge of administration procedures and command line tools for your chosen operating system. If necessary please consult your local system support group.

Oracle DBA skills required

Oracle Database 10g is a large and complex suite of programs which you (or other people in your organization) are assumed to be familiar with already. The instructions provided here cover the bare minimum needed to install Oracle and set up a CatDV database on a single machine. Oracle administration topics such as storage management, security, network operation, load balancing, backups and recovery etc. are well beyond the scope of this guide. Instead, please refer to the documentation provided with Oracle or to any full-time Oracle DBAs within your organization for assistance with installing, configuring and administering the Oracle database.

A note on passwords

In the following instructions, replace *topsecret* with a password of your own choice for administering Oracle as a whole. Replace *secret* with another password for use by the CatDV application; this password is less secure and appears in plain text form in scripts or registry entries and should not be the same as any other passwords you use.

It is possible to configure Oracle to use Windows user authentication (host credentials) for controlling access to the database but in the notes below we are assuming that Oracle user names and passwords (database credentials) will be independent of Windows user accounts.

4.2 Installing Oracle 10g Full Version

(These notes describe using the full version of Oracle 10g. See 4.3 for a simpler alternative using Oracle 10g Express Edition.)

1. Obtain the Oracle Database 10g Installer, either by downloading it from Oracle's web site or ordering a CD from your Oracle vendor. (Please note that, although the Oracle database software is in general freely available to download and install under a developer/evaluation license, you are responsible for purchasing any required licenses and ensuring you adhere to Oracle's license conditions.)
2. Make sure you have Windows administrator privileges for your machine. Run the Oracle installer and select a Basic Installation of the Enterprise Edition
3. Select the option to create a starter database, with global database name ORCL, password *topsecret*. (You can skip this step and use an existing database instance if you have one, or use a different name if you choose.)
4. Let the Oracle installer and configuration assistants run to completion (approx. 30 minutes)
5. When the Database Configuration Assistant completes, make a note of the summary information displayed, eg.

   Global Database Name: orcl

   System Identifier (SID): orcl

   Server Parameter Filename:
   C:\oracle\product\10.1.\Db_1\database\spfileorcl.ora

   Enterprise Manager URL: http://localhost:5500/em

   Ultra Search URL: http://localhost:5620/ultrasearch

   iSQL*Plus URL: http://localhost:5560/isqlplus

   If you need to connect to and administer the database you can either use the web based Enterprise Manager, or run "mysql /nolog" then type "connect sys/*topsecret* as sysdba" followed by SQL commands at the SQL> prompt. Refer to the Oracle documentation for further information.

   Once you have installed the Oracle database you need to create a catdv user.

Creating a 'catdv' user in the Oracle database

6. The Oracle Enterprise Manager is automatically loaded at the end of the Oracle installation process, or you can open a web browser with the Enterprise Manager URL you noted earlier (eg. http://localhost:5500/em).
7. Log on to Enterprise Manager: User Name: SYS, Password: *topsecret*, Connect As: SYSDBA
8. Go to the Adminstration tab, then Users (under Security). Click on Create.
9. In the Create User form enter these details: User Name: catdv, Password: *secret*. Leave other options blank or at their default values. Go to the Roles tab, click on Modify, find the "Resource" role in available roles and Move to right. Press OK twice.

4.3 Using Oracle 10g Express Edition

As a simpler alternative to the full version of Oracle you can use the new Oracle 10g Express Edition, which is free to download and deploy but omits some of the more advanced features of Oracle. (For example, the Express Edition doesn't include a Java VM or application server, and many aspects of the installation is simpler because you are limited to a single Oracle database instance.)

1. Start by downloading the OracleXE.exe installer from Oracle's web site. Make sure you have Windows administrator privileges for your machine. Run the installer and follow the instructions. You can leave the destination as its default value of C:\oraclexe.
2. Choose a password *topsecret* for the SYS and SYSTEM accounts.
3. On completion (this takes several minutes) the installer will launch the Oracle web interface (http://localhost:8080 by default) which you use to administer Oracle.
4. In the web interface, log in as user SYSTEM with your password *topsecret*.
5. Click on Administration, then Manage Database Users
6. Click Create and create a new user with name catdv and password *secret*. Leave the other options as their default value (ie. with CONNECT and RESOURCE roles). Click Create to confirm.

4.4 Installing the CatDV Server

After creating a catdv user in Oracle you will run an installation script to create the tables for use by CatDV and install the CatDV Server as an NT service:

1. If you have not already done so, unpack the CatDVServer zip file to a temporary location. If using WinZip, create a new folder with this name and make sure you have the "Use folder names" option checked.
2. Double click installer\oracle_win\installCatDVServer.bat. You will need to enter the global database name and *secret* password that you entered earlier. Press Enter to accept all the other default answers.
3. See the following section "Editing the catdv.properties file" before starting the CatDV service.
4. Once you have entered your license key and database URL in the catdv.properties file you can go to Start Menu > Administrative Tools > Services, scroll down to "CatDV Server" and click on Start Service.

4.5 Editing the catdv.properties File

The operation of the server, including how many client connections are permitted, is controlled by a license key or registration code stored in the catdv.properties file. The easiest way to edit this file is using the CatDV Server Control Panel as described in 3.6. However, you can also edit the catdv.properties file manually if you prefer.

You will have been mailed four lines of text such as the following example. (Please contact sales@squarebox.co.uk if you do not have your registration code.)

```
reg.user = Trial
reg.code = OGOP-JZXL-G1TD-65SI-7F7B-YZ3X-GVJD-6YQQ
reg.expires = Jun 30, 2005
reg.info = Workgroup Server for MySQL (5 clients) with Live HTML Publisher
```

Open the catdv.properties file in the CatDV server directory with a text editor (or better still, using the CatDV Control Panel), then copy and paste these lines in to the file, replacing the example lines included in the file as distributed.

This file also contains other configuration information, most notably the connection URL for the database. You need to make sure the value of catdv.database is correct for your database. Comment out or delete the mysql line and uncomment the oracle line, editing it to include your *secret* password that you entered earlier.

The catdv.properties file also contains configuration for the Live HTML Publisher but you can safely ignore these lines if you do not use the HTML Publisher. If you make any changes to the catdv.properties configuration file you need to stop and restart the workgroup server for the changes to take effect.

4.6 Troubleshooting

- The CatDV workgroup server is a Java program, whose class fiiles are in lib\server.jar. The Java Service helper (JSHelper.exe) is used to install and start this program as an NT service but you can also start it manually by typing the following command (which may be useful for diagnostic purposes):

```
java -cp lib\server.jar;lib\jaxp.jar;lib\parser.jar;
lib\jakarta-regexp-1.2.jar;lib\ojdbc14.jar
-Djava.security.policy=java.policy
squarebox.catdv.server.CatDVServer
```

- The server will run with any J2SE or J2EE compliant Java implementation, including Java 1.3.1 or later from Sun Microsystems or the bundled JVM within Oracle Database.
- The CatDV client application and server communicate using Java RMI, which normally uses port 1099. You may need to open up this port in your firewall, or you may need to select a different port if there is a conflict, which you can do by uncommenting and editing the catdv.rmiport parameter in catdv.properties then restarting the server (eg. catdv.rmiport=1100). You will need to enter the same number when connecting to the server from a client.
- If you need to edit the CatDV Server startup parameters (eg. to change the port number, change the database URL, or change the location of the server log file) you can use regedit to edit values within the following registry topic:

  HKLM\SYSTEM\CurrentControlSet\Services\CatDV Server\Parameters.

- To check that the CatDV server has started successfully there are various log files you can consult. Problems executing the server itself (eg. Java not available) may be

reported in the Windows event log (Start menu > Administrative Tools > Event Viewer) or in C:\TEMP\CatDVServer.stderr.log. Once the server has started it creates a log file called server.log in the server current directory, eg. c:\Program Files\Square Box\CatDV Server\server.log.

Database URLs

The CatDV server and the HTML Publisher use a JDBC connection string to refer to the database. Although various forms are supported by Oracle the format used here is:

jdbc:oracle:thin:user/password@//hostname:port/global.database.name

For example, jdbc:oracle:thin:catdv/*secret*@//localhost:1521/orcl

If you are using Oracle 10g Express Edition you only need the user name and password, for example jdbc:oracle:thin:catdv/*secret*

If you are running Oracle and the CatDV server on different machines you can specify a different URL by editing the catdv.properties file (You can also provide a different catdv.database startup parameter to the server, or by editing the webapps/catdv/WEB-INF/web.xml file for the HTML publisher, but setting it via catdv.properties is the recommended method.)

(See 3.9 for details of other general configuration parameters.)

5 Installing Live HTML Publisher

This chapter describes installation of the optional Live HTML Publisher component. The Live HTML Publisher is a web application that provides access to clips in a CatDV workgroup server database from a web browser rather than via the CatDV client application.

5.1 Before You Start

Requirements

The HTML Publisher requires the following components:

- MySQL database software (or Oracle 10g Database)
- A CatDV database installed by the CatDV Server
- Java 1.3.1 or later (Java 1.5 required if using Tomcat 5.5)
- Apache Tomcat servlet engine (or Oracle OC4X)

Description

Instructions are given below on how to install HTML Publisher with both Oracle OC4J and Apache Tomcat. The steps are similar in both cases, namely move the catdv.war file into the web apps directory then restart the application server. This will expand the web archive, creating a catdv web app folder containing JSP pages and static resources, with a WEB-INF subfolder that contains the web.xml configuration file and the class or jar files needed by the web application.

After the initial installation you can edit the web.xml file or JSP pages as required to configure and customize the web application. If you have a CatDV server source license you can recompile the servlets and tag library and replace the catdvservlets.jar file. To see the effect of a change you may need to restart the web application server whenever you change the web.xml configuration file or any of the other resources.

After installing the CatDV web app and configuring catdv.properties with the database URL you can view the HTML Publisher welcome screen by opening the appropriate URL in your web browser, eg. http://localhost:8080/catdv/. Check that you can navigate to all the pages linked from this home page.

To test the HTML Publisher you will need to store some clips in the remote database. Use the CatDV Pro client to import a directory containing media files and then use Publish Window to publish this to the server. You will be prompted for a catalog name. In your web browser, follow the Browse Database link and ensure the catalog name you entered previously is listed there.

The HTML Publisher provides a read only interface to the metadata stored in the CatDV database. To enter or maintain this data you need to use the CatDV Pro client application and the workgroup server.

Overview of the installation process

If you have not already done so, please first ensure that you have successfully installed and tested the CatDV Workgroup Server software. The steps to deploy the Live HTML Publisher are then quite straightforward:

1. Download and install Apache Tomcat (or use another Java servlet container, such Oracle OC4X)
2. Copy the file catdv.war from this archive to the Tomcat or OC4X webapps directory
3. Restart the web application server
4. Visit http://localhost:8080/catdv/ to view the Live HTML Publisher pages (or other port as may be configured for the web application server)

Once you have installer the catdv webapp, see chapter 8 for further details on using and customizing the Live HTML Publisher.

5.2 Installation under Apache Tomcat (Mac OS X)

There are a number of different versions of Tomcat you can use with Mac OS X.

- Use Mac OS X Server, which has Tomcat pre-installed.
- You can download a Unix version of Tomcat from the Apache web site at http://jakarta.apache.org/site/binindex.cgi. You will need to unpack the tar archive and install the files yourself, then start Tomcat by running the bin/startup.sh script. (This is the recommended approach in most cases.)
- You can install Tomcat, JBoss and other development tools as part of the XCode tools CD (in the application server package).
- You can download CompleteTomcat-4.1.24.dmg installer from Server Logistics (PowerPC only).

Installation from Apache web site

1. Visit http://tomcat.apache.org and download apache-tomcat-5.5.20.tar.gz (or later) under Download → Tomcat 5.x → Binary Distributions → Core. This requires Mac OS X 10.4 and Java 1.5. (If you have an older version of Java you can download Tomcat 5.0.28 instead.) Double click this file in the Finder to expand it to a folder called “apache-tomcat-5.5.20”.
2. For simple standalone testing you can run Tomcat from where you downloaded it (eg. your desktop):
   - Drag “catdv.war” from install/webapp to the Tomcat “webapps” folder.
   - Open a Terminal window (Applications > Utilities > Terminal.app) and drag the file “startup.sh” from the Tomcat “bin” folder to the Terminal window, then press enter.

3. Alternatively, you can install Tomcat so it launches automatically when the computer starts up by placing it in /Library/Tomcat and using launchd (this requires Mac OS X 10.4 or later):
    - Use the Finder to move the Tomcat directory to /Library and rename it to "Tomcat"
    - Copy "catdv.war" from install/webapps into the /Library/Tomcat/webapps folder
    - Copy "tomcat.plist" from install/webapp to the /Library/LaunchDaemons folder
    - Restart your Macintosh
    - If you're using launchd you can stop and restart Tomcat from a Terminal window by typing the following:

```
sudo launchtl unload /Library/LaunchDaemons/tomcat.plist
sudo launchtl load /Library/LaunchDaemons/tomcat.plist
```

4. Once Tomcat is running and the catdv webapp is installed, test it by opening http://localhost:8080/catdv/ in your web browser. Ensure you can see the CatDV webapp welcome screen and access the server status and other linked pages.

Using "Complete Tomcat"

If you have no need to run a more recent version of Tomcat, and are not running on a MacIntel, an easy solution is to use the Server Logistics installer. This works well and installs a startup item to automatically launch Tomcat:

1. Download CompleteTomcat-4.1.24.dmg.zip from http://serverlogistics.com/tomcat.php.
2. Double click to unzip the file, and again to mount the disk image. (If you use Safari under Mac OS X 10.4 the image is mounted automatically once download completes.)
3. Run the Tomcat.pkg installer package to install Tomcat in /Library/Tomcat
4. Copy Tomcat.prefPane to /Libray/PreferencePanes if you require a System Preferences item to manually start and stop Tomcat.
5. Copy catdv.war (from the install/webapp folder of the installation archive) to the Tomcat webapps folder, eg. /Library/Tomcat/webapps.
6. Open http://localhost:8080/catdv/ in your web browser and ensure you can see the CatDV webapp welcome screen and access the server status and other linked pages.

5.3 Installation under Apache Tomcat 5 (Windows)

If you are using CatDV Workgroup Server for MySQL, or if you prefer not to use the Oracle OC4J application server, then you can download the open source Tomcat server to run the CatDV Web Publisher.

1. Download apache-tomcat-5.5.20.exe from http://tomcat.apache.org (Download → Tomcat 5.x → Binary Distributions → Core → Windows Service Installer)
2. Run this installer and install it to a location such as C:\Program Files\Apache Software Foundation\Tomcat 5.5

3. Go to Start Menu > All Programs > Apache Tomcat 5.5 > Configure Tomcat

   Leave the settings as their default values, then press OK. Connector port: 8080 (or you can choose another), user: admin, password: leave blank.

4. Open a web browser and open http://localhost:8080/ and check you see the Tomcat welcome page. If you do not see the Tomcat page, check the log files in the Tomcat logs directory. In particular, you might get a port 8080 already in use error message. If so, see the next section.

5. Copy catdv.war to the Tomcat webapps folder, eg. C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps.

6. Shutdown and restart the Apache Tomcat service, either via the NT Services control panel (Start Menu > Administrative Tools > Services) or the Tomcat systray icon.

   The catdv.war archive will be automatically unpacked and a catdv directory is created.

7. Edit the catdv web app config file, webapps\catdv\WEB-INF\web.xml, to make sure it has the correct location of your catdv.properties file. The catdv.properties file contains the database URL such as jdbc:oracle:thin:catdv/*secret*@//localhost:1521/orcl.mydomain.com or similar, based on the password and global database name you chose earlier. (Consult the Using and Configuring Live HTML Publisher chapter for details on other parameters you can configure, such as media paths.)

8. Go to http://localhost:8080/catdv/ and ensure you can see the CatDV webapp welcome screen and access the server status and other linked pages.

Note: The CatDV Web Publisher is also compatible with earlier versions of Tomcat (such as 4.1.27) if you are unable to use Tomcat 5.5 because of its dependency on Java 1.5

5.4 Installation under Oracle OC4J (Windows)

Oracle Database includes a J2EE application server called OC4J but this is not enabled by default unless you install the Oracle Application Server. You can use the OC4J included in an Oracle Database installation but will need to start the application server manually each time. These instructions assume you do not have Oracle Application Server.

1. Locate the OC4J home directory, eg. C:\oracle\product\10.1\Db_1\oc4j. In the remaining steps this location is referred to as OC4J_HOME.

2. Copy catdv.war to OC4J_HOME\j2ee\home\applications

3. Edit OC4J_HOME\config\application.xml and add the following line after the other <web-module> definitions:

   ```
   <web-module id="catdv" path="../../home/applications/catdv.war" />
   ```

4. Edit OC4J_HOME\config\http-web-site.xml and add the following line after the other <web-app> definitions:

   ```
   <web-app application="default" name="catdv" root="/catdv" />
   ```

5. Start the OC4J application server by typing

```
cd OC4J_HOME\j2ee\home
java –jar oc4j.jar
```

   The catdv.war archive will be automatically unpacked and a catdv directory is created.

6. Edit the catdv web app config file, webapps\catdv\WEB-INF\web.xml, to make sure it has the correct location of your catdv.properties file. The catdv.properties file contains the database URL such as jdbc:oracle:thin:catdv/*secret@*//localhost:1521/orcl.mydomain.com or similar, based on the password and global database name you chose earlier. (Consult the Using and Configuring Live HTML Publisher chapter for details on other parameters you can configure, such as media paths.)

7. Go to http://localhost:8888/catdv in a web browser to test the CatDV webapp.

5.5 Installation on Other Platforms

Installation under Apache Tomcat on other platforms, including Linux, Solaris and other Unix platforms, and even under other J2EE web application servers, is very straightforward. Although the paths are different on other platforms, the steps are essentially exactly the same as described above and involve copying the catdv.war file to the webapps folder. Please consult the documentation with the web application server for details of any specific differences.

5.6 Troubleshooting the Web App

- Tomcat needs to have the Java bin directory in its path (in particular, javac.exe) or it will fail to compile JSPs. (Windows only)
- If you see a “fork is true, ignoring compiler setting” error message ensure that the JDK bin directory is in your path. Open a DOS prompt and ensure that you can type “javac” and get a usage message. If Tomcat is set to start automatically make sure you rebooted the machine after altering the path. (Windows only)
- Tomcat 5.5 requires Java 1.5. If you are using an older version of Java, please download Tomcat 5.0.28 (or even 4.1.24) instead. Live HTML Publisher does not require newer Tomcat features, so you are safe in using an older version.
- If you see an error message about the licence file not being found, edit the webapps/catdv/WEB-INF/web.xml file to point to the catdv.properties file in the CatDV Server directory. You need to restart tomcat for changes to this file to take effect.
- If you see an error message about your registration code not being valid, ensure that the catdv.properties file contains the correct license key that you were sent and that this license is valid for Live HTML Publisher.
- Tomcat logs files are placed in the ‘logs’ subdirectory of the Tomcat folder. There are different logs relating to startup of the Tomcat application server itself and operation of particular web applications within it, so you may need to order the files by date and

look at several log files before you find the file (or files) that are relevant to your problem.

- If copying the catdv.war file to the webapp directory doesn't result in an expanded catdv folder check the Tomcat server.xml config file: check the <Host> tag for the virtual host you are using and make sure unpackWARs and autoDeploy properties are set to true and that appBase and docBase have the correct values. You should also check that the webapp directory is writable by the tomcat user.
- Visit http://localhost:8080/catdv/serverStats.jsp and check that server statistics are displayed (you may need to change the port depending on your installation) and all have the expected values.

Disabling Oracle XML Services on port 8080

If you are using the Live HTML Publisher with Apache Tomcat there is one thing to note when using Oracle. By default, Oracle Database TNSLSNR will install XDB (Oracle XML database web services) on port 8080, creating a conflict with the default Tomcat port. Although you can change the Tomcat port, it is a strong convention to use 8080 for Tomcat, so it is recommended to change XDB instead.

The simplest way to change the port used by XDB, for example to 8081, is to launch the Oracle SQL*Plus application (or the web based iSql*Plus interface) and paste the following SQL command in:

```
call dbms_xdb.cfg_update(updateXML(
dbms_xdb.cfg_get(),'/xdbconfig/sysconfig/protocolconfi
g/httpconfig/http-port/text()', 8081))
```

You can lauch SQL*Plus by typing "sqlplus sys/*topsecret* as sysdba" at a DOS prompt.

For further details on XDB see:

http://www.idevelopment.info/data/Oracle/DBA_tips/xmldb/XMLDB_2.shtml

6 Upgrading From An Earlier Release

This chapter contains information for existing CatDV Server users who want to update their server installation. Please read the important warnings in 6.3 before commencing an upgrade.

6.1 Version history

6.1.6 19 January 2010
Fix a problem with schema updating.

6.1.5 11 January 2010
Audit log queries now search action field. Fix issue with service not starting on some Windows installations. Fix compatibility with MySQL 5.1.16 and later. Client connections are automatically removed after being idle for 24 hours. Updated schema to 3.11 (added userID to clip).

6.1.4 28 November 2009
Updated schema to 3.10 (adding archiveStatus field). Fix for queries involving a NOT term..

6.1.3 8 October 2009
Updated schema to 3.09 (adding bigNotes and events fields). Performance improvement when polling server. Add support for displaying version number of connected clients. Minor fixes.

6.1.2 7 August 2009
Updated schema to 3.08. Fix deadlock that could occur on disconnection after an error occurred. Minor API enhancement.

6.1.1 7 June 2009
Updated schema to 3.07. Added support for saving clip lists and smart folders to the server. Added support for explicitly keeping track of the order of clips within a catalog. Other minor fixes.

6.1.0 26 May 2009
Major performance update, with new APIs to allow clients to request just those objects which have been modified or which haven't yet been loaded on the client. Added support for broadcasting messages to users from the administrator.

6.0.8 13 May 2009
Minor update to increase memory allocation and track peak memory usage.

6.0.7 5 May 2009
Implement recycle bin. New permission to control deletion of clips. Schema upgrade to 3.06. Add button to control panel to perform housekeeping (clean old clips from recycle bin etc.). Minor fix to 59.94 timecode support.

6.0.6 14 Apr 2009
Fix occasional error about aspectRatio when using MySQL in strict mode. Fix incorrect

URLs in web publisher clip baskets. Ensure log viewer doesn't fail with out of memory error when viewing huge log files.

6.0.5 19 Mar 2009
Patch for issue relating to handling of new timecode formats.

6.0.4 14 Mar 2009
Upgrade schema to 3.05 to ensure catalog names are unique, add support for new clip types, and reduce delay when shutting down the server. Add support for 59.94 timecode format. New ability to search any log or metadata field (not just any log field). Added relative date search. Fix related to saving import sources. Minor fix to gmtDate field.

6.0.3 30 Jan 2009
Add support for new timecode format. Backups are now recorded in the audit log. Fixed RecDate field in LiveHTML Publisher. Other minor fixes.

6.0.2 22 Oct 2008
Fixed issue relating to saving sequences. Better checking and reporting of errors.

6.0.1 20 Sept 2008
Upgrade schema to 3.04 with support for new clip status and history fields. Make control panel slightly more robust. Bug fix relating to fps field and queries involving timecodes and durations.

6.0.0 11 July 2008
No change since b10 (except to remove beta designation from the version number).

6.0b10 26 May 2008
Upgrade schema to 3.03 and add support for saving multi-track sequences. Updated the control panel to make it more robust and so it automatically applies schema updates when required. Under Mac OS X it will also offer to create a default my.cnf file. Updated the server release notes.

6.0b9 24 Apr 2008
Added support for saving metaclips and image sequences. Performance improvement when publishing catalogs. Note: this release involves a schema change, see section 6.3 if you are upgrading an earlier installation.

6.0b8 4 Apr 2008
Added support for new tape library management search functions. Better recording of both logged on and operating system user in audit messages.

6.0b7 15 Feb 2008
Interim beta release(s),

6.0b3 18 Sep 2007
Interim beta release. Includes support for saving settings to the server.

6.0b1 25 Aug 2007
Beta release. Improved recovery from MySQL connection errors. Live HTML Publisher enhancements: more advanced queries, added clip basket and email response servlets.

5.1.2 10 Jul 2007
Minor update to fix “You don’t have permission to save tape infos” message if you don’t have permission in the system group, plus some reporting improvements.

5.1.1 9 Jan 2007
Minor update to fix occasional ‘tape already exists’ error and add extra reporting.

5.1.0 6 Nov 2006
Bug fixes and enhancements, mainly to Live HTML Publisher. (See section 6.2 for details)

5.0.0 30 Sep 2006
Production release, including bug fixes and support for Oracle.

5.0b4 9 July 2006
Add support required by worker nodes. Live HTML Publisher is now supported again. Automatically reconnect if database connection goes down. Other minor fixes.

5.0b3 7 June 2006
Added support for Mac OS X Server. Updated installers so control panel is launched automatically on installation. Various bug fixes. Revised the release notes.

5.0b2 30 May 2006
Renamed sever version from 4.0 to 5.0 to keep it in line with the version number of the client application. Added support for saving sequences and included a new server control panel.

4.0b1 5 May 2006
First public release including new Enterprise server (featuring users, groups and permissions to implement access control).

3.6.0b 30 Nov 2005
Private beta release of some of the 4.0 changes, including tape library management support.

3.5.2 22 Feb 2006
Fix incorrect use of executeQuery that affected some jdbc drivers. Improved diagnostics (if server is configured incorrectly it now starts up and reports an error to clients on connection, instead of silently failing to launch).

3.5.1 8 Nov 2005
Fix problem with Publish Changes command under MySQL. Fix problem with certain registration codes. Fix occasional ‘Error parsing timecode’ errors when publishing catalogs.

3.5.0 5 Aug 2005
New combined release. Add compatibility with CatDV 4.0 clients. Fix a problem with missing thumbnails when loading a catalog that has been updated.

3.5b1 28 July 2005
Beta release of the above changes

3.1.0 30 December 2004
First Oracle release

3.0.4 13 December 2004
Fix expiry bug.

3.0.3 21 January 2004
New installer and startup item for Mac OS X.

3.0.2 20 May 2003
Updated scripts to work with MySQL4. Added NT Service install and uninstall batch files. Updated release notes for Windows 2000.

3.0.1 5 May 2003
Ensure that licenses are released again if a client dies without disconnecting. Ensure server terminates if you press ctrl-C from the console. Fix <Any logging field> queries when searching for catalogs (as opposed to clips).

3.0.0 14 Oct 2002
Public release of CatDV Workgroup Server, incorporating a minor bug fix to queries including a NOT term.

6.2 Functional Changes

Version 6.0

6.0 has further enhanced the Live HTML Publisher by supporting more complex queries and allowing users to create clip baskets and send these by email (for example, to place an order from a stock footage library). It includes support for saving metaclips and image sequences and for saving preference settings and group documents to the server.

The CatDV Control Panel has been extended so it automatically applies database schema updates when necessary, simplifying upgrades. It also features a button to quickly back up the database.

(Please note that version 6.0 is not currently compatible with Oracle.)

Version 5.1

5.1 primarily updated the Live HTML Publisher:

1. When using Live HTML Publisher with the Enterprise server it now uses access control rather than giving access to all the catalogs.
2. A new GetMedia servlet provides access to preview and media files (if they are available on the server machine) directly from Tomcat. There is no longer any need to configure an external web server to do this (though you can still configure the Live HTML Publisher to use external URLs if desired).
3. Added the media.equiv property so media can be displayed even if it's stored with a different path in the database from that the server uses to access the file.
4. Made a number of additional fields available when displaying clips and catalogs via JSPs, including a display of multiple thumbnails for one clip.

5. Added extra fields (ModifiedDate, ImportDate, MediaDate) and operators (Fuzzy Match, Before and After date) to the query page.
6. Added clipAll.jsp with notes on how to customize the JSPs.
7. Added 'Allow anonymous' and 'Web user name' fields to the config editor.
8. Added basic support for "Fuzzy" text matches (requires CatDV 5.1 Client). Currently this matches clips containing all the (space-separated) words specified. In future this operator may be extended to do more general free text searches.
9. Other bug fixes and enhancements, for example adding an optional tomcat.xml launchd file for use in Mac OS X, and performance improvements when deleting catalogs or shutting down the server.

Version 5.0

The following changes were made in 5.0:

1. Support for separate Workgroup and Enterprise versions of the server, the latter supporting access control and other additional functionality.
2. New tables to support users, groups and definitions, and new columns to identify the user and group a catalog belongs to.
3. A new table to support saving clip sequences.
4. New table to support tape library management (this was actually introduced in 3.6)
5. Other schema changes, for example to the audit log.
6. New CatDV Server Control Panel to display status about the server installation and help in managing it.
7. Support for MySQL 4.1 and later, including full Unicode character set support.
8. Compatible with CatDV Pro 5.0 clients.

Version 3.5

Version 3.5.0 did not add any significant functionality changes over the previous release (3.0.4 for MySQL, 3.1.0 for Oracle) other than a few bug fixes. It was primarily an organizational "tidy up".

The key differences are as follows:

1. This version of the server is compatible with CatDV Pro 4.0 clients.
2. License authentication has changed. Previously, there was no registration code stored on the server. Instead, the CatDV Pro 3.0 client licenses encoded whether access to the workgroup server was permitted or not. Now, a license registration code on the server determines whether access to the Workgroup Server and/or Live HTML Publisher is permitted, and with how many client connections. Any CatDV Pro 3.0 or 4.0 client can connect, subject to the server license conditions.
3. Configuration of settings for the workgroup server, such as the database url to use, has changed. Instead of being passed on the command line (in the launchServer script) the

settings are now stored in a catdv.properties file. This file also stores the registration code.

4. Configuration of settings for the Live HTML Publisher has similarly changed. Previously, properties such as the database and preview or media URLs were set in the web.xml file. Now, these too are stored in the catdv.properties file.
5. A bug was fixed where thumbnails would disappear from a catalog after attaching media and then publishing changes.
6. There is now a single universal installation archive for all CatDV server products and for all platforms.
7. The version number of the Live HTML Publisher now matches that of the workgroup server. (The previous HTML Publisher version was 1.01)
8. Other bugs are fixed as per the version history at the start of this document.

6.3 Overview of the Upgrade Process

When upgrading an existing CatDV Server installation there are generally two steps involved:

- updating the database schema (defining new tables or columns to support new functionality)
- updating the CatDV Server software.

(Updating the Live HTML Publisher is covered separately in a later section.)

Whether a database scheme change is needed depends on the version you are upgrading from. Often a minor point release won't require a change whereas a major version update will, but for details see the relevant section below. In either case, before attempting the upgrade you should always shut down the CatDV Server and **make a complete backup of the 'catdv' database** as described in 3.9 or elsewhere. It is not possible to roll back a schema update other than by deleting the entire database and restoring it from the backup!

Updating the server software involves replacing the files in the server installation directory, especially lib/server.jar but possibly others. Before doing so, you should **make a backup copy of the installation directory** (typically /usr/local/catdvServer or C:\Program Files\Square Box\CatDV Server), especially of the configuration file catdv.properties as this might get overwritten.

The installer batch file (Windows) and installer package (Mac OS X) are primarily intended for *new* installations only. However, they will also update the server software and launch the CatDV Control Panel (which in turn will apply any necessary database schema updates), so they can also be used to upgrade an existing installation.

6.4 Automatic Upgrade of the CatDV Server

Starting with version 6.0, in addition to performing new installations, the installer batch file (Windows) and installer package (Mac OS X) will also upgrade an existing installation.

After updating the server software itself the installer will launch the CatDV Control Panel application. Each time it runs the Control Panel will check the CatDV database schema and automatically offer to apply any upgrades if necessary. Follow the instructions that are displayed to perform an automatic upgrade. If any problems occur (for example, MySQL is not running, you are using an unsupported configuration, the upgrade scripts could not be located, or you have forgotten the MySQL root password you set up previously) please correct the problem and relaunch the Control Panel. If necessary, perform a manual upgrade as described in the next section.

6.5 Manual Upgrade of the CatDV Server

Updating the database schema from 5.1 to 6.0

If you are currently using CatDV Server for MySQL 5.1.2 you should use the command line and run the following scripts in turn to update the CatDV database:

mysql -u root -p*secret* < install/mysql/update_catdv301.sql

mysql -u root -p*secret* < install/mysql/update_catdv302.sql

mysql -u root -p*secret* < install/mysql/update_catdv303.sql

mysql -u root -p*secret* < install/mysql/update_catdv304.sql

mysql -u root -p*secret* < install/mysql/update_catdv305.sql

If you are using an earlier version of the CatDV Server you need to upgrade to 5.1 first.

Updating the database schema from 3.x to 5.1

If you are currently using CatDV Server for MySQL 3.0.4 or 3.1.0 you should use the command line and run the following scripts to update the CatDV database:

mysql -u root -p*secret* < install/mysql/update_catdv103.sql

mysql -u root -p*secret* < install/mysql/update_catdv104.sql

mysql -u root -p*secret* < install/mysql/update_catdv201.sql

mysql -u root -ps*ecret* < install/mysql/update_catdv202.sql

If you are currently using CatDV Server 3.5 you should run the following scripts:

mysql -u root -p*secret* < install/mysql/update_catdv104.sql

mysql -u root -p*secret* < install/mysql/update_catdv201.sql

mysql -u root -p*secret* < install/mysql/update_catdv202.sql

If you are running 3.6 you only need to run the last two scripts.

Replace *secret* with the MySQL database password you set at the time of installation. (If you did not set a password then you should omit the –p option.)

Updating the CatDV Server software

For further explanation of the following steps please refer to the instructions for a new installation given elsewhere in this document.

1. Identify where the workgroup server software is currently installed on your server machine. For example, /usr/local/catdvServer, /Library/CatDVServer, or C:\Program Files\CatDV Server. This directory will contain files such as launchServer and java.policy, amongst others, and a lib subdirectory containing various .jar files.
2. Shut down the CatDV server. If you are using Windows, run the UnInstall.bat script to de-register the existing "CatDV Server" NT service
3. As administrator, delete or move the server installation directory to a backup location. This step is optional but is recommended in case you need to revert.
4. Use the regular installation script to install the server in a new location, answering No when it asks whether to create the CatDV database (as it already exists). Do not simply install over the top of an existing installation as the file layout may have changed slightly and old files will remain.
5. Enter your license registration code (you will need to purchase an upgrade and receive a new license code as the old licenses are no longer valid in this version). Review the other configuration settings in catdv.properties and copy any settings that you need over from the old configuration file.
6. Relaunch the CatDV Server

6.6 Upgrading the Live HTML Publisher

Because the Live HTML Publisher is intended as a toolkit, designed to be customized and integrated with your existing web site, it is not possible for the installer to automatically upgrade an existing installation. Please use the following instructions as a guideline and refer to your systems integrator or web site maintainer for further details as required.

Updating the Live HTML Publisher (full upgrade)

1. Shut down the Apache Tomcat (or OC4X) server
2. Identify the Tomcat directory, eg. /Library/Tomcat or /usr/local/jakarta-tomcat-4.1.24 or C:\Tomcat4.1 or C:\Program Files\Apache Software Foundation\Tomcat 5.0.25 which you are running from.
3. Rename the webapps/catdv directory to catdv.old
4. Copy catdv.war from the install/webapp directory of the installation archive to the Tomcat webapps directory. (If necessary, overwrite any old catdv.war file there.)

5. Restart Tomcat (or OC4X) and visit http://localhost:8080/catdv/index.html (or other port number as you previously used) with a web browser. This will automatically unpack the catdv.war file and create a new catdv web application in the webapps folder. The installation is not yet complete however.
6. If you customised any of the JSP files or added custom resources such as icons or static pages, copy these from webapps/catdv.old to the webapps/catdv directory or manually reapply your changes.
7. Edit the webapps/catdv/WEB-INF/web.xml file with a text editor and change the catdv.properties value to point to the catdv.properties file in the workgroup server directory you installed in the previous section. (Although the catdv webapp automatically searches various standard locations for this file it is better to configure the correct path explicitly.)
8. Open the old webapps/catdv.old/WEB-INF/web.xml configuration file. If you entered values for media.path, media.url, preview.path, preview.url or timecode.format here, copy these values to the catdv.properties file. Note that while the format of the file is quite different, the meaning of these settings is the same, but beware that any backslashes in Windows file paths now need to be doubled up, eg. "C:\\CatDV\\Preview Files". This is because backslash is used as a special 'escape' character to quote the next character.
9. Using your web browser, visit the Server Statistics and other pages to check that they operate correctly as before.

Updating the Live HTML Publisher (minor updates)

Alternatively, if you are performing a minor version update, it may not be necessary to replace all the files and a simpler update procedure may be sufficient:

1. Shut down the Apache Tomcat (or OC4X) server
2. Identify the Tomcat directory, eg. /Library/Tomcat or /usr/local/jakarta-tomcat-4.1.24 or C:\Tomcat4.1 or C:\Program Files\Apache Software Foundation\Tomcat 5.0.25 which you are running from.
3. Make a backup copy of the file webapps/catdv/WEB-INF/lib/catdvServlets.jar and replace it with the file of the same name from the current installation archive.
4. Restart the Tomcat server and check you can access the Live HTML Publisher pages.

7 Using the CatDV Pro Client Application

This chapter includes notes on how to use the CatDV Pro client application with the workgroup server, including an excerpt from the CatDV Pro client user manual.

7.1 Installing the CatDV Pro Client

To aid in testing and managing the database you may also choose to install the CatDV Pro Client Application on the server machine (you would normally install CatDV Pro on client machines as a desktop application). This is optional and is not required for normal operation of the server.

The CatDV Pro client needs the following software components:

• Java 1.3.1 or later

  Note that the CatDV Pro client is not compatible with the Oracle JVM and you will need to install Java J2SE 1.3.1 or later from Sun Microsystems if you want to run the client on the same machine.

• QuickTime

  CatDV Pro uses QuickTime, which is available from http://www.apple.com/quicktime/download/standalone.html. Follow the instructions that come with the client application for further details (in particular, you need to do a custom installation and select *all* components of QuickTime).

• CatDV Pro client application

  You can obtain the latest CatDV Pro client (eg. CatDVPro4.0.zip) from http://www.squarebox.co.uk/download.html

Follow the installation instructions provided in the CatDV Pro Read Me file to install the client (and Java and QuickTime if necessary).

Launch the client application and ensure you can open the supplied Sample.cdv catalog.

Use Log On To Server (from the “Server” menu) to connect to the workgroup server. If the server is on the same machine enter ‘localhost’ as the hostname. (If you started the workgroup server on a port other than the default RMI port of 1099 then include that after the hostname, eg. “localhost:1100”.)

7.2 Using the CatDV Pro Workgroup Features

Using the workgroup features provided by CatDV Pro several members of a team can share clip catalogs by saving clip information to a central shared database, rather than to files held locally on disk. They access the central server from the CatDV Pro client application by using commands in the “Server” menu (in older versions of the CatDV Pro client this was called the “Workgroup” menu).

Connecting to the server

Use the Server > Log On To Server command (Cmd/Ctrl-K) and enter the hostname or IP address of the machine running the CatDV Server. When you press OK you will be logged on to the server and the other "Server" menu commands will be enabled, or you may see a message that a connection failure occurred.

To check that you have established a connection with the server program use the Server > Server Admin Panel command to display statistics about the operation of the server, such as how many catalogs and clips are contained in the remote database.

If you are using the Enterprise version of the client you should also enter a CatDV user name and password to identify you within the CatDV server (see 7.3 "Using Enterprise Features" for more details).

Publishing catalogs

If you have created catalogs and saved them locally on your hard disk you need to publish them to make them available to other users via the shared database.

Once they are stored in the shared database you no longer need the local catalog files, though you may choose to keep these files somewhere as a backup or in case you need access to them should the server be unavailable. Once published to the database you should make all your changes there, however, rather than in the local files, as the local files will not be kept in sync with the database.

You publish a catalog by opening it in CatDV Pro and then using the Server > Publish Catalog command. This will publish the catalog from the current window (even if you have just created it and it has never been saved to disk - if you don't require a local copy you can then close the window without saving changes).

You can also publish an entire directory full of catalog files directly from your local hard disk by using the Server > Bulk Publish Catalogs command (previously this command was known as "Publish Folder").

(Note that initially the "Publish Catalog" command, and other commands that modify or delete data on the server, are disabled on the client application. You need to enable them via the Advanced tab of CatDV Preferences. This is a safety feature to prevent inexperienced or occasional users of CatDV from accidentally overwriting data on the server.)

Opening a remote catalog

Use the Server > Browse Database command to view a list of all the catalogs in the remote database, including a short summary of the contents of each catalog. You can open a catalog by double clicking its name in the list. From this window you can also delete catalogs, or search for all the catalogs containing a particular keyword. This will search in both the catalog descriptions and the individual clip details.

Querying the remote database

Use Server > Perform Query (Cmd/Ctrl-Shift-F) and enter search criteria to search for matching clips across the entire remote database.

When you run the query a window is displayed containing the query results, combining all the clips that match, even if they are in different catalogs.

You can save the query results to a new local catalog file, print them out, export them as a batch list, or make changes to the clips returned, perhaps adding new logging annotations and then publishing the changes back to the remote database.

When you perform a remote query you use the same enhanced query dialog as when performing local searches, though you have several additional options to return additional clips similar to the ones matched by the query:

For example, if you find a clip whose log notes contain a particular keyword you might also want to return the clips immediately before and after it on the same tape. You can also choose whether to return all the thumbnails associated with a clip or just the poster thumbnail.

Press Find Clips to create a new query results window showing all the matching clips, regardless of which catalog they are in.

Press Find Catalogs to display a list of matching catalogs within the database, including a count of how many clips in each catalog match the query.

Managing catalogs

Although all the clips on the remote server are stored in the same database, for convenience they are still grouped into logical groupings called catalogs. You should normally create a separate catalog for each tape, or perhaps each shoot or each project, rather than trying to store all your clips in one large catalog.

This will make it easier to manage your clips. For example, you can use the Delete Catalog button in Browse Database to delete a catalog from the database. You also minimise the risk of creating a catalog that is too large to open reliably if you only have limited memory available on the client.

Once you have opened a remote catalog you then work on a local copy of the clips and thumbnails from that catalog, held in memory on your machine. If another user on your network edits these same clips and publishes their changes to the database you can use Server > Refresh Window to update your window with the latest version from the remote database. The time at which the contents of the window were last synchronised with the remote database is shown as part of the window title. If you have had a window open for a long time it's a good idea to refresh the window before starting to make any changes.

Publishing changes

When you open a remote catalog, or perform a query and are working with the query results, you can edit the clips in your window exactly as if you were working on a normal local catalog file. However, rather than saving any changes to disk with File > Save Catalog, you normally want to update the clips in the remote database instead. For this

you use the Server > Publish Changes command. (When working on remote catalogs the Cmd/Ctrl-S shortcut activates Publish Changes rather than Save.)

You can add logging notes, change clip names, make selections, select new poster thumbnails, delete unwanted clips, split a clip into two or create new secondary clips, and all these changes will be saved when you use Publish Changes.

If you have opened a remote catalog you can also create brand new clips, eg. by importing a file or using New Log Entry. You can't add new clips if you are viewing query results, though, because in that case it's not defined which catalog the new clips belong to.

Resolving conflicts

If two users try to make changes to the same catalog or clips at the same time then only the first set of changes that are published will be saved to the remote database.

The second person who attempts to publish changes will receive a warning message stating there were conflicting edits (eg. trying to add a comment to a clip which the previous user has just deleted).

All the changes which can be saved without conflict are saved, and the main window is refreshed to show the current contents as per the remote database. Any clips which weren't able to be saved are displayed in a new "unsaved changes" window. The second user then needs to manually re-apply those changes in the main window, deciding whether and how to resolve any conflicts before trying to publish the changes again.

Re-publishing a catalog

If you publish a catalog with the same name and creation time as an existing catalog in the remote database (and your local catalog is newer than the one in the database) then you will overwrite that catalog in the database with the newer one. Normally, however, you should always use Publish Changes, as this automatically merges your changes and attempts to resolve any conflicting edits.

There are two main situations where you may want to overwrite an entire catalog by re-publishing it with Publish Catalog, rather than just publishing the changes:

- If you need to rearrange the order of clips in a catalog. (Publish Changes will insert, delete or update clips in the database but not change their order.)
- If you saved a remote catalog locally for offline working and now want to publish the changes that you made to the local files.

In both cases note that any change history associated with the old catalog will be lost, and if another user has the same catalog open they will be unable to publish their changes.

Read-only access to server

Initially when you launch the CatDV client application and connect to the server you have *read only* access. All the commands which might modify data held on the server (such as Publish Changes or Delete Catalog) are disabled by default. This is because many users only need to be able to browse and search the database without wanting to

risk accidentally deleting any data. Once a user becomes more familiar with the system they can enable the "Allow write access to server" option in the Advanced tab of Preferences.

It is also possible that a particular copy of the client application has a "Browse Only" license which never allows write access, even if you try to set a preference option. Finally, if you use the Enterprise client and server (see below) the logged on user may be restricted in terms of whether they can save data to the database.

Checking the server status

If you are currently on the server machine itself you can check that the server is running and display statistics such as the number of clips and catalogs in the database using the CatDV Server Control Panel (see 3.6). You can also view the server log and start and stop the server.

You also have access to the server statistics from the CatDV client application, using the Server Admin window. The Connections tab shows which clients are currently connected to the server so you can check whether all your licenses are in use.

Other tabs in the Server Admin window are relevant to the Enterprise edition only.

7.3 Using the Enterprise Features

With the Enterprise version of the CatDV server you gain the ability to control who has access to which parts of the database by defining users and groups. For example, you might have two quite separate projects going on and want to set it up so users can only view and edit catalogs that were created within their team.

To support access control, you now need to both connect to the CatDV server and log on as a particular CatDV user. A CatDV user is created by an administrator using the CatDV user admin panel and is distinct from the operating system user.

Note that creating CatDV users and setting their permissions only controls access to the metadata held in the CatDV database. It does not prevent users physically accessing the media files if they have access to them via the file system.

Users, groups and permissions

The first thing to when setting up access control in CatDV is to define your production groups, for example one group for each team or department or particular project that is being worked on. You can then create users and assign them to groups, giving them different roles in each group if you choose. Catalogs belong to one group or another, and only members of that group will have normally have access to those catalogs.

Users can belong to one ore more groups. A single user can have different permissions in different groups (for example no access to one group, read only access in another, and admin rights in a third). Likewise, different members of the same group can have different levels of access. You might set up a group "Project X" with user "Fred" as administrator, while "Susan" might have read and write access to catalogs in that group but can't create new users.

There is a special predefined group, the System Group (with id 0), to which you can assign users if you want to give them access to all catalogs and groups throughout the system (perhaps they're a manager, or the system administrator).

A "system administrator" can create new groups and edit anyone's permission. A "group administrator" can create new users and assign them to his or her groups only (and can also grant existing users of other groups access to his groups) but can't create new groups or edit permissions for other groups.

A catalog only belongs to one particular group (or is "unowned" and belongs to no group), so if you're a member of more than one group and create a new catalog you need to specify which of those groups the catalog belongs to. You do this by specifying which group you are currently working in (ie. which "hat" you're wearing today) when you log on. If you are a member of more than one group you can change your current group without logging off by bringing up the connection dialog again.

Administrator user

When you first install the CatDV Enterprise Server software a single user called "Administrator" with no password is created in the database. You should log on with this user and set a password for this account, then create additional users and groups as required.

Catalog details

You can view the group any catalog belongs to, and change the group if you have permission, by bringing up the details of the catalog using the Get Info button on the "Browse Database" screen.

If you check the "Show all" box this screen lists all the catalogs in the database. Any which you don't have permission to access are greyed out. The permission column summarises the access you have to a catalog using letters:

r – read (can open the catalog)
w – write (can publish changes)
d – delete.

The cataog details also shows information such as who initially created the catalog or when it was last modified.

User administration

To create users and groups and edit their permissions you use the User Admin panel of the Server Admin Panel.

First, check the "Show all" groups box (otherwise only groups you are a member of are shown) to see if the group you want exists. If it doesn't already exist press the Create button to create the group (you will need to be a member of the System Group with administration privileges to do this).

Select the group you want to add users to and check the "Show all" users box. If the user you want already exists, click on them and either assign them to the group (if they're not

a member yet) or edit their permissions as required. If not, press Create button to create the user.

When you have selected both a group and a user, and if you yourself have admin privileges in that group, you can edit that user's permissions in the group by checking the appropriate boxes.

Assign somebody to the special "System group" to give them a particular level of access to any catalog in the database. For example, read access to any catalog belonging to any group.

Shared settings

With the Enterprise server, an administrator can save preference settings to the server so that when a user logs on he or she will automatically load the appropriate settings (user defined fields, pick list values, view layouts, and other settings) rather than using locally stored preferences which may be inconsistent. Settings are saved under a particular production group, so if a user is a member of more than one production group (each of which might relate to a separate TV show that is being worked on, with different user fields in use for each) then they can load whichever settings are relevant to the work they are doing at the time.

Production blog

It is possible to simple documents such as notes or to-do lists on the server for communicating information among the team. The documents can contain links to specific catalogs or clips and appear under the "Production blog" tree node on the client. A 'document' consists of 'entries' so you can maintain a discussion thread as required.

Audit log

The audit log displays an audit history of significant events to the database, including when the server was started and stopped, when users logged on and off, and when catalogs were published or modified. Select the type of message or date range to filter messages, or type in text to search for in the description column.

Tape library management

The Library Management screen gives you a list of all the tape records in the CatDV database. You can store information against each particular tape, such as as a description, its shelf location, a barcode number. This information is linked to all the media clips that refer to that tape identifier. You can therefore use the library management screen as an alternative way to browse the contents of the database – you can browse clips either by catalog (using Browse Database) or by tape (using Library Management).

Shared clip lists

Using the "Server" tree you can create named clip lists on the server by right clicking on the Clip Lists node. A clip list is a collection of clips (like a playlist in iTunes). Clips are stored by reference, so you can include the same clip in more than one clip list. You can

add clips by dragging them from another a remote catalog or query results to the clip list, and remove them by opening up the clip list for editing and then deleting the clip.

Shared smart folders

As well as shared clip lists (with an explicit list of clips), you can create smart clip lists (called smart folders) which are based on a query. When the user clicks on a smart folder the query is performed and the matching clips are shown. Users can create their own personalized smart folders which are saved locally on their machine but with the Enterprise edition of the server you can also create shared smart folders, where the query is stored on the server.

Automatic smart folders

Any clip property which is configured as a grouping field appears under the Automatic Smart Folders node so you can quickly perform queries based on that field.

8 Using and Configuring the Live HTML Publisher

This chapter describes the functionality provided by the Live HTML Publisher and gives details of how to customise this interface. Note that the HTML Publisher currently only provides a subset of the full workgroup functionality available with the CatDV Pro client application. In particular, it is not possible to import media files or tapes to create new clip records, or to edit existing records, using the web interface.

8.1 Overview

The Live HTML Publisher consists of Java servlets and JSPs (Java Server Pages) that communicate with the CatDV MySQL database to display the contents of the database. The interface is based around separate web pages. The following pages are available as standard. Although functional as provided, these pages are intended as templates that can be customised by the user as required.

browseDatabase.jsp

> Browses the database, returning a list of all the catalogs in the database together with a brief summary of the contents of each catalog (dates, tapes, and clip types included in that catalog).

catalog.jsp

> Provides details of a particular catalog, including a list of all the clips in that catalog.

clip.jsp

> Provides details of a particular clip, including thumbnail, a list of all the main clip properties (organized into logging fields, technical fields, and other metadata fields), and a link to the preview media if available.

clipAll.jsp

> Provides a list of *all* the properties for a clip. (This is designed to aid customization of the web interface by helping you choose which clip properties to include in your custom JSP pages.)

searchForm.jsp

> A form in which to enter the parameters for a search.

searchDatabase.jsp

> The page that displays the results of a search, which may either be a list of matching catalogs or of matching clips.

serverStats.jsp

> Displays brief statistics about the database (total number of clips stored, etc.)

printenv.jsp

Display various system properties. This isn't part of the Live HTML Publisher user interface but may be useful for diagnostic purposes during initial set up.

8.2 Customising JSPs

Java Server Pages are HTML pages that contain special tags that execute Java code on the server to query the CatDV database and insert data into the page. JSPs are processed by the Apache Tomcat servlet engine. JSPs are automatically compiled into Java servlets when they're accessed for the first time. Although you can use normal HTML editing tools to customise the layout of JSP pages it is important to get the syntax of the tags correct otherwise you will get Java errors when you display the page.

The following are some of the special tags that will be encountered and are likely to be unfamiliar to most HTML authors:

<%@ taglib uri="/WEB-INF/catdv.tld" prefix="catdv" %>

This directive occurs at the top of each JSP that wants to use the CatDV tag library (special tags of the form <catdv:xxx> that perform particular functions, detailed below).

<%@ include file="bodyStart.inc" %>

The include directive can be used to insert common HTML into the file, such as style sheets or JavaScript functions, so that all pages have a common appearance.

<form action="searchDatabase.jsp" method="POST">

A JSP can be the target of a form that is submitted.

<% … %>

This is used to insert raw Java code. In general, you should not alter any of these from the examples provided in the sample JSPs as it's very easy to introduce syntax errors that will prevent the page from compiling, eg. by misplacing a brace or accessing a variable that isn't defined in a particular context.

<%= ….%>

This includes output from a Java expression in the HTML page that is being produced, for example a variable defined by a CatDV tag or declared in an earlier <%…%> statement.

<%= clip.get("Name") %>

This accesses the value of a particular clip attribute.

```
<% if (condition) { %>
        ...
 <% } else { %>
        …
<% } %>
```

This example will conditionally include one or other piece of output. As noted previously, irrespective of the non-Java contents between the tags, if you want to avoid syntax errors it is vital that the braces continue to match!

<%! … %>

If you are familiar with Java you can define helper functions to carry out useful tasks such as formatting data for a table by placing your method definitions inside a global declaration tag like this.

8.3 CatDV Tag Library

To access data from the CatDV database, you need to include special tags in your JSP pages. Each tag is of the form <catdv:xxx>…</catdv:xxx>. If there is no separate closing tag then XML syntax needs to be used, as in <catdv:xxx/>.

Tags normally take one or more attributes (similar to arguments of a function) that identify the data being queried.

Although some tags directly expand to produce HTML or other text that is inserted into the output, most of the tags in the CatDV tag library define variables which must explicitly be included in the page, as this gives the JSP more control over where and how the data is presented in HTML. Many of the tags therefore define a variable such as ‘catalog’ or ‘clip’ that stores the values being returned. How to access these variables is covered in the next section.

The tags below are currently available. The current list of available tags is also shown in the catdv.tld file.

The serverStats tag

The <catdv:serverStats/> tag takes no parameters and returns a text string listing how many objects of each type are stored in the database.

The browseDatabase tag

The <catdv:browseDatabase> tag accepts a single http request parameter “order” defining the order in which catalogs are returned.

It iterates over all the catalogs in the database and defines the “catalog” variable for each row being returned in turn. It also defines the “rowcount” variable to display how many catalogs are contained in the database.

The catalog tag

The <catdv:catalog> tag returns details about a specific catalog. It accepts a single http request parameter “id”, defining the numeric id of the catalog. It defines “catalog” and “rowcount” variables, and can contain a <catdv:cliplist> tag to iterate over the clips within that catalog.

The cliplist tag

The <catdv:cliplist> tag is an iterator tag that returns details for a variable number of clips. It it called as many times as necessary, setting the "clip" variable for each clip in turn. The <catdv:cliplist> tag only occurs within another tag to define the list of clips, either <catdv:catalog> to return all the clips in a given catalog, or <catdv:doQuery> to return the clips matching a query.

For example, the <catdv:catalog> and <catdv:cliplist> tags might be combined as follows:

```
<catdv:catalog>
<p>The catalog <b><%= catalog.get("Name") %></b> has <%= rowcount %>
clips:</p>
    <catdv:cliplist>
        <%= clip.get("Name") %> (<%= clip.get("In") %>-<%=
clip.get("Out") %>)
        <br>
    </catdv:cliplist>
</catdv:catalog>
```

The doQuery tag

The <catdv:doQuery> tag accepts the request parameters below to define a clip query.

Up to 20 query terms can be defined by defining attr1/op1..20

For a text-based query you should specify the following parameters:

attr*N*	Name of attribute being queried (one of: Name, Notes, AnyField, NameNotes, Tape, Bin, Format, ImportNotes, SourceMedia, User1..UserN)
op*N*	Query operation (0=contains, 1=equals, 2=starts with, 3=ends with, 4=is blank, 5=is one of, 6=contains one of, 7=regular expression, 8=all words, 9=fuzzy match)
param*N*	Text being compared against. In the case of 'one of' comparisions this is a comma-separated list of values
ignore*N*	Case insensitive comparision

For a date-based query use the following parameters:

attr*N*	Name of date attribute being queried (one of: RecDate, MediaDate, ImportDate, ModifiedDate, CatalogCreated, CatalogExended, CatalogSaved or CatalogPublished)
op*N*	Query operation (16=equals param1, 17=before param1, 18=after param1, 19=between param1 and param2, 21=date is blank)
param*N*a, param*N*b	Date values being compared with, in d/m/y [h:m[:s]] format.

By default a Boolean AND operation is assumed (ie. all conditions must be true to find a clip) but you can specify more complex queries like in the CatDV client application by setting

or*N*	This term OR any other terms with the 'or' flag set must match

not*N*	This term must be false for a clip to match

Additionally, you can specify attributes which aren't in the list of known attribute names above by giving their attribute IDs preceded by a '#', eg. attr5=#12.

A <catdv:cliplist> tag within the scope of the doQuery tag will then iterate over the matching clips and display the results, one row at a time.

The queryDatabase tag

The <catdv:queryDatabase> tag accepts the same request parameters as <catdv:doQuery> to define a query. However, rather than returning a list of clips matching the query it returns a list of catalogs. The "catalog" variable is defined, as per <catdv:browseDatabase>

The clipBasket tag

A clip basket can be used to collect together results from different queries like a shopping basket. Baskets are created and managed automatically as part of the user's session context. They expire automatically after 30 minutes.

The <catdv:clipBasket> tag accepts the following parameters

basketAction	Either 'add' or 'remove' to add or remove all the clip with specified clip ids from the basket
c_*ID*	Which clips to add or remove, for example c_1234=on to add or remove the clip whose id is 1234.

As an alternative to setting basketAction and a c_ you can specify a single clip id to add with the parameter 'add', for example add=1234

A <catdv:cliplist> tag within the scope of the clipBasket tag will then iterate over the clips in your current basket and display the results, one row at a time.

The sendEmail tag

You can email the contents of the user's clip basket using the <catdv:sendEmail> tag. This accepts the following parameters:

to	Recipient email address
name	Sender's name
email	Sender's email address
cc	Optional CC email
message	Optional message (in addition to the clip basket)
subject	Subject header

You need to configure the sendEmail servlet by specifying various context parameters in the web.xml file:

mail.host	Which SMTP server to use
mail.user	Which user to log on as (for smtp authentication)
mail.password	Password for smtp authentication

mail.preamble	Text to include at start of email message
mail.postamble	Text to include at end of message
mail.success	Text that the sendEmail tag displays to user on successful sending (otherwise an error message is displayed)
mail.fields	Comma separated list of fields to include when listing clips in the email message

The getThumbnail servlet

The JPEG data for a thumbnail image is returned using the getThumbnail servlet. This is a separate servlet rather than a tag because the data is returned in a separate http request, not as part of an HTML page. The servlet takes a single parameter "id" which defines the id of the thumbnail object in the database, for example <IMG SRC="/catdv/getThumbnail?id=12345" WIDTH=64 HEIGHT=48>.

Developers do no normally need to worry about the details of how to use the getThumbnail servlet as the complete <IMG SRC …> expression to use is predefined and returned as part of a "clip" object.

The getMedia servlet

This is a new servlet that will display media and preview files directly from within Tomcat (without having to configure a separate web server such as Apache). If the media files are accessible to the server and you configure media.dir to be the root directory of the media, for example /Volumes/Media, then "/catdv/media/xyz/file.mov" will return the contents of /Volumes/Media/xyz/file.mov. Similarly /catdv/preview/… will append the given path onto the value of the preview.dir variable.

8.4 Displaying Objects and Lists of Objects

Many of the CatDV tags listed above define a clip or catalog variable that contains the values being displayed. It is up to the JSP itself to display the desired attributes of the object. (Note that 'attributes' might also be referred to as fields or properties, or as columns when they're displayed in a table.) Each attribute has a name such as "Name" or "Tape" by which it is known – all the available clip and catalog attribute names are shown in the lists below.

The value of an attribute can be displayed using syntax such as <%= clip.get("Name") %> within a JSP page. There are several variants of this call, all of which accept at least one parameter that defines the attribute name. get() will display the value as a string of text, or display an empty string if the attribute has no value. get_t() is similar but returns " " if the attribute is empty, which is sometimes necessary when displaying values in an HTML table. More generally, get() with two parameters returns the second string if the attribute is empty, for example get("WhoCreated", "N/A"). get_limit() takes a parameter which defines how many characters to display at most – if the text is longer than this limit it is truncated and an ellipsis "…" is shown. If you're unsure which attributes might be available then the keys() method returns a Java Enumeration of all the property keys.

When a tag is intended to display a *list* of objects (such as all the clips in catalog) it will iterate over the list and set the variable to each row in succession.

Catalog object

The following attributes are defined for catalogs:

ID	The numeric id of the catalog.
NumClips	Number of clips in the catalog. (Number of matches if performing a query.)
Name	The catalog name.
Comment	The catalog notes.
WhoCreated	Which user first created the catalog.
WhoSaved	Which user last modified the catalog.
WhoPublished	Who published the catalog to the database.
Modified	When was the catalog last modified.
Version	Which version of CatDV Pro last saved the catalog.
Tapes	All the unique tape name(s) of clips in the catalog.
ClipTypes	All the different clip types represented in the catalog (eg. DV, MOV, JPEG, LOG, etc.)
Info	A brief status text about the query.
MinGMTDate	The earliest record date of any clip in the catalog.
MaxGMTDate	The latest record date of any clip in the catalog.
GMTRange	A summary of the range of dates covered by clips in the catalog.
MinFileDate	The earliest file modification date of any media file in the catalog.
MaxFileDate	The latest file modification date of any media file in the catalog.
UserName	CatDV user name who owns the catalog (Enterprise server only)
GroupName	CatDV group that owns the catalog (Enterprise server only)

Clip object

The following attributes are defined for clips:

ID	The numeric id of the clip.
CatalogID	The numeric id of the catalog that this clip belongs to.
CatalogName	The name of the catalog the clip belongs to.
CatalogNotes	The comments of the catalog the clip belongs to.
Name	Clip name.
Tape	Tape or reel name
In	Start of clip (timecode)
Out	End of clip (timecode)
In2	Start of selection within clip (timecode)
Out2	End of selection within clip (timecode)
Duration	Duration of clip (timecode)
Bin	Bin name
Timecode	The timecode format (eg. 2997 for drop frame NTSC, or 25 for PAL)

Notes	Clip notes field
Type	Clip type
SeqNum	Sequence number of clip within its import source
Transition	Transition (for clips that came from importing an EDL)
Modified	Date the clip entry was last modified
ImportNotes	Import warnings
DVTC	The timecode valid embedded in the DV stream (usually same as In)
Marked	Whether the clip is marked or not
Hidden	Whether the clip is marked as being hidden
SourceMedia	The source media filename (for clips imported from a media file)
Audio	The audio format of the clip (if imported from a media file)
Video	The video format of the clip (if imported from a media file)
Importer	The QuickTime importer used to import the media file
QTTracks	A list of tracks in the movie (if imported from a media file)
Metadata	QuickTime user data (eg. Exif metadata tags)
DataRate	The data rate (if known)
MediaSize	The size of the media file (if known)
IsStill	Is this a still image?
Format	A summary of the format of the media
RecDate	The original date of recording (if known, eg. from DV metadata)
ClockAdjust	Any camera clock adjustment (from local time to GMT)
GMTDate	The record date or media file date (corrected for GMT if relevant)
DVInfo	An encoded version of the DV metadata
PosterID	The ID of the poster thumbnail
Poster.IMG	This returns an <IMG SRC=…> tag that will display the poster thumbnail for the clip at maximum available size.
Poster.SMALLIMG	This returns an <IMG SRC=…> that returns the poster thumbnail for the clip scaled so it is no larger than 48x36 pixels in size.
Preview.URL	The URL of the preview movie (if available)
Preview.LINK	This returns a complete link including an appropriate icon, whether or not the preview is available, ie. <A href=…><IMG SRC=…></A>
User1, …, UserN	These return the user-defined attributes of the clip. Note that the user-defined names of these attributes are not stored in the database and are used for display in the client application only. Within the get() call it is necessary to use the numeric id of the user attributes, for example “User11” might correspond to “Project”, but if you customize the JSP you can of course display your own label that correctly describes the field.
DataRate.RAW	Unformatted data rate (bytes/sec)
MediaSize.RAW	Unformatted media file size (bytes)
AllThumbnails.IMG	Sequence of <IMG…> tags containing all the available thumbnails for the clip (between the clip’s In and Out point). (For practical reasons this is limited to a maximum of 16 thumbnails.)
MediaDate	Modification time of the media file

Not all these properties will be set for all clips or catalogs.

Sorting tables

Tags that display a list of objects (<catdv:browseDatabase> and <catdv:cliplist>) may accept a request parameter "order" that defines the sort order of the results returned. This is the name of one of the properties (as listed above), and may be preceded by "–" to reverse the sort order.

The example browseDatabase.jsp shows how to display a header row by iterating over the column names and highlighting the background colour of the column that is being sorted on.

8.5 Configuring the HTML Publisher

The HTML publisher normally uses the Apache/Jakarta Tomcat servlet engine (though in practice any J2EE-compliant servlet container can be used). Details on how to install Tomcat are provided in the release notes and further documentation on how to install and configure Tomcat is included with Tomcat itself. Once Tomcat is started the documentation can be viewed by connecting to Tomcat's own web server. By default, Tomcat runs on port 8080, so test the installation and view the documentation by viewing http://localhost:8080 with your web browser.

Tomcat uses a standard directory structure for all its web applications (webapps). Assuming <tomcat> is the Tomcat root directory, the CatDV HTML Publisher would be installed in <tomcat>/webapps/catdv. <tomcat> might be /usr/local/jakarta-4.1.3 under Unix or C:\Tomcat4.1 under Windows.

<tomcat>/bin	Programs to start/stop the server etc.
<tomcat>/conf	Server-wide configuration files (shouldn't need modification)
<tomcat>/logs	Location of error log files
<tomcat>/webapps/ROOT	Location of the homepage for the Tomcat-managed web site
<tomcat>/webapps/catdv	Location of the CatDV HTML Publisher web app
…/catdv/WEB-INF/lib	The CatDV servlets and supporting libraries
…/catdv/WEB-INF/web.xml	CatDV HTML Publisher configuration file
…/catdv/*.jsp	JSP server pages (may be modified)
…/catdv/*.gif	Icons used within the JSP pages (other resources can also be placed here)
…/catdv/*.html	Static web pages
…/catdv/*.inc	HTML and Java fragments designed to be included in JSPs
<tomcat>/…	Other files and directories required for the operation of Tomcat itself

Within the webapps/catdv/WEB-INF/web.xml file you will see definitions of all the servlets provided by CatDV. These lines should not be modified. This file also defines the location of the catdv.properties file that contains a license key and properties that affect the functioning of the Live HTML Publisher. You should edit this so the Live HTML Publisher points to the same copy of catdv.properties that the Workgroup or Enterprise Server is using. (If you edit the web.xml file be careful to adhere to correct XML syntax or the catdv web application will fail to start.)

The following settings in catdv.properties are relevant to the HTML Publisher:

catdv.database	JDBC URL to the MySQL database
preview.dir	File path to the preview directory (if available locally on the machine running Tomcat)
preview.url	URL to access the preview directory (if available, eg. via Apache web server). Leave this blank to use the built-in getMedia servlet.
media.dir	File path to root directory where online media is stored (if available locally). This is relative to the server's filesystem.
media.equiv	Optional equivalent path prefix that relates to how paths are stored in the database (ie. relative to the client that published the clips to the database)
media.url	URL to the online media location (if available via a web or file url). Leave this blank to use the built-in getMedia servlet.
timestamp.format	How dates are formatted for display (0=D/M/Y, 1=M/D/Y, 2=Y/M/D, 3=verbose, 4=yyyy-mm-dd. Add 16 for 24-hour clock.)
web.user	Which CatDV user to connect to the Enterprise server as. This determines which catalogs are accessible via the web interface.
catdv.allowAnonymous	Alternatively, you can leave web.user blank and allow anonymous access, giving access to unowned catalogs.

If you make any changes to this configuration file you need to stop and restart Tomcat.

Tomcat can be configured to automatically recompile any JSPs when you change them so their effect is instantly apparent. This is recommended during development, but once finalised you can turn this option off by changing the "reloading" parameter in <tomcat>/conf/web.xml. You can also configure the port Tomcat runs on and other Tomcat server parameters in <tomcat>/conf/server.xml.

To view the CatDV HTML Publisher web pages, visit http://localhost:8080/catdv/. If you get any errors check the log files in <tomcat>/logs for additional details.

Access control

If you are using the Workgroup edition of the CatDV Server then the Live HTML Publisher automatically has (read only) access to all the catalogs and clips in the database.

If you are using the Enterprise edition of the server then you need to specify which catalogs are available through the Live HTML Publisher. The way to do this is use the Server Admin Panel of the CatDV client application to create a "Live HTML" user and assign that user as a member of all the groups whose catalogs you want to be visible with the web interface. In the catdv.properties file, set web.user to the name of the user you created.

You can also set the catdv.allowAnonmyous property to true, which will give the Live HTML Publisher access to any unowned catalogs, ie. catalogs which aren't assigned to a particular group. (These same catalogs will also be available to the CatDV client application to users who connect but don't log on with a CatDV user name and password.)

When using the Enterprise server you must specify at least one of web.user or allowAnonymous (or both) otherwise the HTML Publisher won't have access to *any* clips or catalogs in the database.

Playing media

If you wish, you can configure the Live HTML Publisher to make the media files themselves, not just the metadata and thumbnails, available to a web browser.

There are several important points to bear in mind before doing so:

First, the original uncompressed (or DV compressed) media files may be very large. Even if they are available online, it is probably not practical to download multi-gigabyte files via HTTP. For this reason, it usually makes more sense to provide lower resolution, preview quality files for users to view the contents of a clip when using the the web interface, and provide direct access to the media files via the file system or a SAN when full quality files are needed for editing. (Providing access to the original media files via the web interface is primarily intended for supplementary media files such as still images.)

Second, in order to determine whether media for a clip is available or not, the Live HTML Publisher needs direct access to the media files via the file system. Also, the path stored for the media files in the CatDV catalog must be correct for the server to access them.

Finally, because the web server does not have a capability to construct movies on the fly, separate preview movies should be created for each clip that might be returned from the web server.

(This differs from how the CatDV Pro client application normally handles previews. Because the previews are available to it via the file system the client application can efficiently construct a movie for display using portions of one or more preview movies, based on tape name and timecode value. For example, the preview directory might

contain a single preview for the whole of one tape, and separate previews for each shot on another tape. If the user tries to display a short 5 second clip from within the first tape a reference movie referring just to the required portion of the tape is created within the client application. Similarly, if a sequence (or composition) or clips is being shown, a single reference movie combining portions of several preview files is assembled automatically, giving the client application the ability to assemble and display a cuts only version of a imported EDL on the fly. The HTML publisher, on the other hand, will only display previews if they directly match the start timecode and duration of the clip being shown.)

Configuring Tomcat to act as a media server

The CatDV client application uses two mechanisms to locate media for a clip. Original media files are accessed via the source media file path, and preview movies are looked up using tape name and timecode values of the clip being displayed.

The Live HTML Publisher is similar, so to play media you need to configure the file path either to media files (media.dir) or the preview root directory (preview.dir) respectively, or both. Because the media path for clips as stored in the database may be different from the local path the server uses to access the same files you may also need to set the media.equiv property.

Once the Live HTML Publisher has determined via the file system that a media or preview file is available it then translates this to a URL that the browser can use to access the file. As part of the HTML Publisher configuration you therefore need to tell it what URL to present to the web browser by setting the corresponding media.url and preview.url properties.

You have a number of options as to how you serve the media and preview files:

- If you have an existing web server such as Apache httpd, or Microsoft IIS, then you can set up aliases to the relevant media directories and enter the URL prefix. This will likely be on a different port, and possibly even a different server, from the Live HTML Publisher itself.
- Alternatively, you can use the new getMedia servlet built in to Live HTML Publisher 5.1 to serve the media. Configuring this is very easy – just leave the media.url and preview.url fields blank.
- Finally, it's also possible to configure directory aliases for the built-in web server included in the standalone Tomcat server. You do this by creating a configuration file for each directory mapping you want to set up. In the following example, *tomcat* is the Tomcat installation directory, eg. /Library/Tomcat or C:\Program Files\Apache Software Foundation\Tomcat 5.5.

  Assume you have media files on /Volumes/XDrive/Media and want to make these available via a URL of the form http://server:8080/xmedia/ .

  Create a file media.xml in *tomcat*/conf/Catalina/localhost containing the following line:

  <Context path="xmedia" docBase="/Volumes/XDrive/Media" />

(You can also add this fragment of xml to the *tomcat*/conf/server.xml file). As with the case of an external web server you then need to tell the Live HTML Publisher which URL prefix to use.

While configuring an external web server to serve media files may provide better performance, using the built-in getMedia servlet is a simple and effective way to make previews and media files available with the minimum of configuration.