Appendix A

PERSONAL EDITION SINGLE DISK SM SERVER SOFTWARE

Powered by

A.1 Introduction

Yima is a streaming media architecture composed of *server, client,* and *utility* software components. Both the server and the client are currently Linux based. The full version of Yima and its internal operation are described in Chapter 10. As part of this book a simplified version of Yima is distributed: the *Yima Personal Edition* or *Yima PE* for short.

Yima PE is missing some of the features of its big sibling. Notably absent are multi-node support and retransmission capabilities in the server component. The provided client only recognizes and plays WAVE[1] audio streams. The idea behind the Personal Edition is to have a fully functional streaming media system that can be used for experiments and teaching while at the same time keeping the source code at a manageable size. This chapter describes the installation and use of Yima PE. It is a guide on how to compile and run the various components of the system as well as how to use the different configuration parameters.

---

[1]Uncompressed, 16-bit linear PCM, 44,100 Hz stereo files are supported in the current software. This is the same format that one would typically find on an audio CD.

The Yima PE software is included on a CD-ROM with this book. For updates and other online information about Yima PE and this book please refer to the following web site:

<div align="center">

`http://streamingmedia.usc.edu`

</div>

This guide makes several assumptions about the user's level of expertise:

- The user is familiar with the Linux operating system.

- The user has access to two Linux computers that are connected via an IP-based network (e.g., a LAN or the Internet). The software has been tested mostly on Red Hat Linux versions 7.x, however, any recent Linux distribution should work fine.

- The user has read Chapter 10; "Yima Case Study" and is familiar with some of the architectural features of Yima.

- The user has a CD-ROM or some other file archive of the software called *Yima Personal Edition.*

The rest of this chapter is organized as follows. We will start with an overview of the software components that are included with Yima PE and describe the contents and organization of the CD-ROM (or the file archive, if the user acquired the software as a `tar` or `zip` file). Subsequently, we will describe the installation and operation of both the server and the client components. Finally, we will elaborate on how to prepare new content for the system and make it available for streaming.

Please also refer to the `README` file included with the software for any changes and updates that were made after this book was printed.

A.2   Yima PE Components

Yima PE consists of the following software components:

1. A server executable called `yimaserver` to manage the storage, retrieval, and scheduling of the streaming media files. The server is also responsible for the communication with the clients. The server implementation is multi-threaded and some of its tasks include:

    - Media block storage and retrieval scheduling.
    - Media block caching.
    - RTP packetizing (see RFC 1889 [148]). The data is pre-packetized with the `yimasplit` utility (see item 3).

  - RTP packet scheduling and transmission.

  - RTSP communication with the clients (see RFC 2326 [149]).

2. A client executable called `yimaclient` capable of receiving and playing uncompressed (linear PCM) WAVE audio files. This software requires a computer with a soundcard that is supported under Linux. If you are having difficulty with sound support under Linux you may want to check the following web site `http://www.alsa-project.org` for more information.

3. A `yimasplit` utility to prepare media files for storage and streaming. A regular WAVE file needs to be pre-processed by this utility before the `yimaserver` process can serve the file to clients. The `yimasplit` program pre-packetizes the original file data into RTP packets and then combines the packets into fixed sized media blocks.

A.2.1   CD-ROM Contents

The Yima PE CD-ROM is organized with the following directory structure:

```
/YimaPE_v1.0
            /Server
            /Client
            /Splitter
            /Streams
```

The `Server` directory contains a tar archive with the source code of the complete Yima PE server software source tree. Section A.3 describes how to build the `yimaserver` executable.

The `Client` directory contains the software client capable of playing streamed WAVE files. The software is distributed as a tar archive that includes the complete software source tree. Section A.4 describes how to build the `yimaclient` executable.

The `Splitter` directory contains the `yimasplit` tool to make original WAVE files streamable. Again this is a tar archive with source code. See Section A.5 for installation instructions.

The `Streams` directory contains a short sample PCM audio WAVE file ("yimaintro.wav") with the following format: stereo, 16-bits/sample, 44,100 samples/second (1.4 Mb/s bandwidth). The `yimaintro.wav` file can be processed with the `yimasplit` program to test the correct operation of the installation (see Section A.5).

A.3   Server Installation and Operation

To install the Yima PE server please complete the following steps. Step 1 may be skipped if you already have a functioning Linux installation.

1. Install a recent Linux distribution on your computer and create an appropriate user account (for example "yima"). The software has been tested on Red Hat Linux 7.x. Boot the machine and login.

2. Open a terminal (shell) window.

3. Mount the CD-ROM and copy the `YimaPE_Server_v1.0.tar` source file from the CD-ROM (from the `YimaPE_v1.0/Server` directory) to the user home directory.

4. Untar the `YimaPE_Server_v1.0.tar` file. A source directory structure will be created:

   ```
   $tar xvf YimaPE_Server_v1.0.tar
   ```

5. Change to the subdirectory `YimaPE_v1.0/Server/qpthread-1.3.1`.

6. Install the `qpthread` library version 1.3.1 by following the directions contained in the `README` file. `QpThread` is a third-party software and is not covered by the Yima PE license.
   (See http://lin.fsid.cvut.cz/~kra/index.html for more information.)

7. Set the shell environment variable `LD_LIBRARY_PATH` to include the qpthread library directory path (refer to the man pages of the shell that you are using, e.g., bash, csh, etc., on how to do this). You will need to do this with superuser privileges so that it applies to all users.

8. Create a directory that is used to store the movie data. This directory can be located on a different path or disk, for example. We will refer to this directory as the *media directory*.

9. Change to the `YimaPE_v1.0/Server` directory in the yima user home directory.

   Note: The tar file contains a precompiled executable file of the server called `yimaserver`. If you want to rebuild the software, then type `make` in the `Server` directory.

10. Copy the `config_Sample` file to a file called `config` (while still being in the `YimaPE_v1.0/Server` directory):

    ```
    $cp config_Sample config
    ```

11. Edit the server `config` file: on line 2 change `/home/yima/Movies` to the path of the media directory that you designated on this machine (see item 8).

At this point the server is ready to run and you can test it by becoming superuser and invoking the server executable.

1. Change to become superuser:

```
$su
 password:
```

2. Start the Yima PE server:

```
$./yimaserver
```

The server needs superuser privileges because it listens on TCP port 554 for incoming RTSP connections from clients[2]. If everything went well then you should see:

```
<YimaPE 1.0> begin scheduler
<YimaPE 1.0> begin rtsps
```

If you see an error message like the following:

```
(RTSPS/frontend/interface): Permission denied
Abort.
(RTSPS/backend): frontend suddenly closed the link!
(RTSPS/backend/link): Success
Abort.
(RTSPS/frontend): socket suddenly closed by RTSP server!
: Success
```

then the server executable is not running with superuser privileges and the RTSP port 554 could not be opened.

If you see an error message like this:

```
./yimaserver: error while loading shared libraries:
./yimaserver: undefined symbol: __tiQ25qpthr10QpCondBase
```

then the dynamic qpthread library could not be found and loaded. Make sure that the `LD_LIBRARY_PATH` environment variable points to the location where `libqpthr.so` can be found.

---

[2]If you change this port number in both the client and server code to a number > 1024 then no superuser privileges are required. However, port 554 is the standard port for RTSP connections.

Even though the server can run at this point, we first need to prepare some media content for streaming to make it useful. See Section A.5 on how to prepare a WAVE file for streaming.

A.4   Client Installation and Operation

To install the Yima client please complete the following steps. Step 1 may be skipped if you already have a functioning Linux installation.

1. Install a recent Linux distribution on your computer and create an appropriate user account (for example "yima"). Boot the machine and log in.

2. Open a terminal (shell) window.

3. Mount the CD-ROM and copy the `YimaPE_Client_v1.0.tar` source file from the CD-ROM (from the `YimaPE_v1.0/Client` directory) to the user home directory.

4. Untar the `YimaPE_Client_v1.0.tar` file. A source directory structure will be created:

   ```
   $tar xvf YimaPE_Client_v1.0.tar
   ```

   Note: The tar file contains a pre-built executable file of the client, `yimaclient`. If you want to rebuild the software, then type `make` in the `YimaPE_v1.0/Client` directory.

   The client code makes use of a graphical user interface library called `XForms` which is a third-party software and is not covered by the Yima PE license.
   (See http://world.std.com/~xforms/ for more information.)

The client can be run at this point by executing `yimaclient`. However, we first need to prepare some media content for streaming on the server and then we will need to tell the client where to find this content. So let us continue with Section A.5 on how to prepare a WAVE file for streaming.

A.5   Media Preparation

To prepare a new media file for Yima PE do the following. First you will need to install the splitter utility from the CD-ROM. This utility called `yimasplit` is used to pre-packetize media content and to break it into fixed sized data blocks.

1. Mount the CD-ROM and copy the `YimaPE_Splitter_v1.0.tar` source file from the CD-ROM (from the `YimaPE_v1.0/Splitter` directory) to the user home directory.

2. Untar the `YimaPE_Splitter_v1.0.tar` file. A source directory structure will be created:

   ```
   $tar xvf YimaPE_Splitter_v1.0.tar
   ```

   Note: The tar file contains a pre-built executable file of the splitter, `yimasplit`. If you want to rebuild the software, then type `make` in the `YimaPE_v1.0/Splitter` directory.

3. You may want to add the `YimaPE_v1.0/Splitter` directory to your shell search path expression so that the `yimasplit` can be executed from any directory location. Refer to the man pages of your shell for instructions on how to do this.

Once the `yimasplit` utility is installed and ready we can proceed with preparing our media content.

1. Obtain a WAVE file. A short audio file is contained on the CD-ROM ("yimaintro.wav") in the `YimaPE_v1.0/Streams` directory.

2. Copy the file (e.g., `yimaintro.wav`) from the CD-ROM (or your own file) to some temporary directory. Let us assume that directory is called `/tmp`.

3. Execute the following command in the temporary directory:

   ```
   $/<path_to_yimasplit>/yimasplit yimaintro.wav 25
   ```

   You may need to specify the full path to the `yimasplit` utility as shown in the example. Alternatively, you can add the location of `yimasplit` to your shell search path and then you will be able to execute it without a path.

   The parameters of `yimasplit` are as follows: (1) the movie name ("yimaintro.wav" in the example) and (2) the movie length in seconds ("25" in the example). A subdirectory `BLOCKS` with many files will be created; these files are the media blocks. Also, a `config` file is created that contains information about the movie for the server.

4. Copy all the files in the `BLOCKS` directory into the media directory: the one that you configured in the `config` file for the Yima PE server.

5. Change to the `YimaPE_v1.0/Server` directory and append the movie information from the temporary directory to the server `config` file as follows (assuming that the temporary directory is called `tmp`):

   ```
   $cat /tmp/config >> config
   ```

   Now the server can access this new content called `yimaintro.wav` and can serve it to any clients. Note however that the server program `yimaserver` only reads the `config` file at startup. If any changes are made to that file then the server needs to be stopped (e.g., with control C) and restarted.

6. Finally, we need to add the new content to the clients' list of known media files. Ideally, the media name could be entered at runtime, however, currently the client reads this information from a configuration file. To update the client configuration, edit the `Yima.cfg` file that is contained in the `YimaPE_v1.0/Client` directory. Append the new content information in the following format and update the `movies: X` counter:

   ```
   rtspX://<name_or_IP_address_of_server>/<medianame>
   ```

   For example (X is replaced with an ascending number):

   ```
   rtsp1://192.168.1.100/yimaintro.wav
   rtsp2://yima.usc.edu/yimaintro.wav
   ```

Congratulations! The Yima PE system is now ready for streaming. Make sure that the server is running. Execute the client, click on the [`List`] button, select your content by double-clicking and then press the [`Play`] button. If everything went well, you should hear the audio of your media content after a few seconds of buffering delay.

Note that uncompressed, linear PCM WAVE files require about 1.4 Mb/s of bandwidth for their smooth transmission. Therefore, you will not be able to run this demonstration over a regular phone modem, DSL, or cable modem connection. However, if the client is extended with, for example, an MP3 or other audio decompression module so that it can accept compressed streams, then much less transmission bandwidth is required. There will be no changes necessary to the server side. Refer to Appendix C under Chapter 10 for this and other exercises.

Once again, for updates, help and other online information about Yima PE and this book please refer to the following web site:

```
http://streamingmedia.usc.edu
```