QTKit

The QuickTimeKit (QTKit.framework) is a framework developed for working with QuickTime in Cocoa applications.

Two goals:

- Substitute the old Cocoa Application Kit classes NSMovie and NSMovieView.
- Provide a new API to cover more extensive QuickTime functions and data types.

QTKit

Features

- API to play Movies
- API to record Movies
- Support for a wide variety of formats to encode and decode Audio and Video.

QTKit

History

- First introduced in Tiger (OS X 10.4)
- Significantly enhanced in Leopard (OS X 10.5)

- Quicktime has a history of its own, starting in 1991 with Quicktime 1.0 on Apple Macintosh.

That said, Quicktime made its real Cocoa appearance on OS X almost 4 years late.
NSMovie and NSMovieView already came frome NeXTStep.

QTKit

Classes for Playback

- QTMovie (instead of NSMovie)
- QTMovieView (instead of NSMovieView)
- QTTrack and QTMedia, which provide access to the lower Carbon API.

QTKit

Classes for Recording

- QTCaptureDevice represents each connected device as instance
- QTCaptureDeviceInput as input source for media devices of all kind (cams and mics)
- Preview classes (QTCaptureView and QTCaptureAudioPreviewOutput). Both need a device for input and output.

QTKit

How to record... (the simple way)

- Collect all connected devices with '+inputDevicesWithMediaType' from QTCaptureDevice and
- Open them as needed.
- Use QTCaptureMovieFileOutput to record and save the data.

Delegation is common in the QTKit.

QTKit

How to record...

- Collect all connected devices and open them as needed.
- Use Preview-Classes to evaluate the media.
- Manipulate the Media
- Create a movie and write it to file.

QTKit

As for a movie...

- Collect the cameras and open them.
- Use the delegate method in QTCaptureView:

```
- (CIImage *)view:(QTCaptureView *)view
  willDisplayImage:(CIImage *)image
```

to manipulate every frame image

Inside this method CoreImage and its filters is the technique of choice.

QTKit

... and storing it in a file:

- Create the movie by adding frame after frame with time-intervals in QTMovie:

```
- (void)addImage:(NSImage *)image
   forDuration:(QTTime)duration
   withAttributes:(NSDictionary *)attributes
```

Convert a CIImage with filters and then to an NSImge to set it as frame in the movie. The dictionary's mandatory attribute is 'QTAddImageCodecType'.

QTKit

Summary

A lot of progress

- Cocoa is now really supported
- The cumbersome Quicktime-API is under the hood.
- Interface Builder is integrated and thus
- Basic functionality is easy to implement

QTKit

Summary

all but perfect

- Support in Interface Builder is not free of bugs
- Utilizing output media makes output devices necessary. To gain access to the CIImages in a stream a capture view needs to be displayed somehow, maybe hidden as UI-element.
- The underlying Quicktime-API is still there and lurking.

Quicktime still has its level of craptitude.

QTKit ...outlook

Snow

- The API has not changed that much.
- Quicktime X is coming...

Problems to just build and run some apps on Snow.

Time-laps Movie, ~ 6h

QTKit