## + (STKDataSource \*)dataSourceFromURL:(NSURL \*)url; - (instancetype)init; - (instancetype)initWithOptions:(STKAudioPlayerOptions)optionsIn; - (void)play:(NSString \*)urlString; - (void)play:(NSString \*)urlString withQueueItemID:(NSObject \*)queueItemId; - (void)playURL:(NSURL \*)url; - (void)playURL:(NSURL \*)url withQueueItemID:(NSObject \*)queueItemId; (void)playDataSource:(STKDataSource \*)dataSource; - (void)playDataSource:(STKDataSource \*)dataSource withQueueItemID:(NSObject \*)queueItemId; - (void)queue:(NSString \*)urlString; - (void)queue:(NSString \*)urlString withQueueItemId:(NSObject \*)queueItemId; - (void)queueURL:(NSURL \*)url; - (void)queueURL:(NSURL \*)url withQueueItemId:(NSObject \*)queueItemId; - (void)queueDataSource:(STKDataSource \*)dataSource withQueueItemId:(NSObject \*)queueItemId; - (void)setDataSource:(STKDataSource \*)dataSourceIn withQueueItemId:(NSObject \*)queueItemId; - (void)seekToTime:(double)value; - (void)clearQueue; (void)pause; (void)resume; - (void)stop; - (void)mute; - (void)unmute; (void)dispose; (NSObject \*)currentlyPlayingQueueItemId; - (void)removeFrameFilterWithName:(NSString \*)name; - (void)appendFrameFilterWithName:(NSString \*)name block:(STKFrameFilter)block; - (void)addFrameFilterWithName:(NSString \*)name afterFilterWithName:(nullable)SString; (float)peakPowerInDecibelsForChannel:(NSUInteger)channelNumber; (float)averagePowerInDecibelsForChannel:(NSUInteger)channelNumber;

(void)setGain:(float)gain forEqualizerBand:(int)bandIndex;

STKAudioPlayer

## STKLocalFileDataSource

+ (AudioFileTypeID)audioFileTypeHintFromFileExtension:(NSString \*)fileExtension;
- (instancetype)initWithFilePath:(NSString \*)filePath;

## *STKDataSourceDelegate* STKQueueEntry (void)dataSourceDataAvailable:(STKDataSource \*)dataSource; OSSpinLock spinLock; - (void)dataSourceErrorOccured:(STKDataSource \*)dataSource; BOOL parsedHeader; - (void)dataSourceEof:(STKDataSource \*)dataSource; Float64 sampleRate; - (void)dataSourcePrematureEOF:(STKDataSource \*)dataSource; double packetDuration; - (void)dataSourceRangeOutOfBounds:(STKDataSource \*)dataSource; UInt64 audioDataOffset; - (void)dataSourceConnectionDrop:(STKDataSource \*)dataSource; UInt64 audioDataByteCount; (void)dataSourceConnectionRegained:(STKDataSource \*)dataSource; UInt32 packetBufferSize; volatile Float64; volatile SInt64; volatile SInt64; volatile SInt64; volatile int processedPacketsCount; volatile int processedPacketsSizeTotal; STKDataSourceWrapper AudioStreamBasicDescription audioStreamBasicDescription; double durationHint; (instancetype)initWithDataSource:(STKDataSource \*)dataSource andQueueItemId:(NSObject \*)queueItemId; - (instancetype)initWithDataSource: (void)reset; (STKDataSource \*)innerDataSource; (double)duration; (Float64)progressInFrames; (double)calculatedBitRate; (BOOL) is Definitely Compatible: (AudioStreamBasicDescription \*) basicDescription; STKDataSource (BOOL)registerForEvents:(NSRunLoop \*)runLoop; (void)unregisterForEvents; ${\tt STKAutoRecoveringHTTPDataSource}$ (void)close; (void)seekToOffset:(SInt64)offset; (int)readIntoBuffer:(UInt8 \*)buffer withSize: (instancetype)initWithHTTPDataSource: (int)size; (STKHTTPDataSource \*)innerDataSource; (AudioFileTypeID)audioFileTypeHint; STKCoreFoundationDataSource CFReadStreamRef stream; BOOL isInErrorState; NSRunLoop \*eventsRunLoop; (BOOL)reregisterForEvents; (void)open; (void)openCompleted; - (void)dataAvailable; - (void)eof; STKHTTPDataSource (void)error0ccured; (CFStreamStatus)status; (AudioFileTypeID)audioFileTypeHintFromMimeType:(NSString \*)fileExtension; (instancetype)initWithURL:(NSURL \*)url; (instancetype)initWithURL:(NSURL \*)url httpRequestHeaders:(NSDictionary \*)httpRequestHeaders;

- (instancetype)initWithURLProvider:(STKURLProvider)urlProvider;

- (nullable)SRunLoop;
- (void)reconnect;

- (instancetype)initWithAsyncURLProvider:(STKAsyncURLProvider)asyncUrlProvider;