

Cat Mouse

High Distinction Project for HIT2302 Object Oriented Programming

Generated by Doxygen 1.8.5

Sat Nov 9 2013 22:17:28

Contents

1 Objective-C Decoupled Implementation	1
2 Class Documentation	1
2.1 CMAntClass Class Reference	1
2.1.1 Detailed Description	3
2.1.2 Method Documentation	4
2.1.3 Member Data Documentation	4
2.1.4 Property Documentation	4
2.2 CMCat Class Reference	4
2.2.1 Detailed Description	6
2.2.2 Method Documentation	6
2.3 CMEvent Class Reference	7
2.3.1 Detailed Description	8
2.3.2 Method Documentation	8
2.3.3 Member Data Documentation	9
2.3.4 Property Documentation	9
2.4 <CMEventAnnouncer> Protocol Reference	9
2.4.1 Detailed Description	10
2.5 CMEEventManager Class Reference	11
2.5.1 Detailed Description	12
2.5.2 Method Documentation	12
2.6 <CMEventProcessor> Protocol Reference	13
2.6.1 Detailed Description	15
2.6.2 Method Documentation	15
2.7 <CMEventSubscriber> Protocol Reference	16
2.7.1 Detailed Description	17
2.8 CMGame Class Reference	17
2.8.1 Detailed Description	18
2.9 CMGUI Class Reference	18
2.9.1 Detailed Description	20
2.9.2 Method Documentation	21
2.9.3 Member Data Documentation	21
2.10 CMKeyboard Class Reference	21
2.10.1 Detailed Description	22
2.10.2 Method Documentation	22
2.11 CMLog Class Reference	22
2.11.1 Detailed Description	24
2.11.2 Method Documentation	25

2.12	CMMouse Class Reference	25
2.12.1	Detailed Description	27
2.13	CMNetwork Class Reference	28
2.13.1	Detailed Description	30
2.13.2	Method Documentation	30
2.13.3	Member Data Documentation	32
2.13.4	Property Documentation	33

1 Objective-C Decoupled Implementation

Version

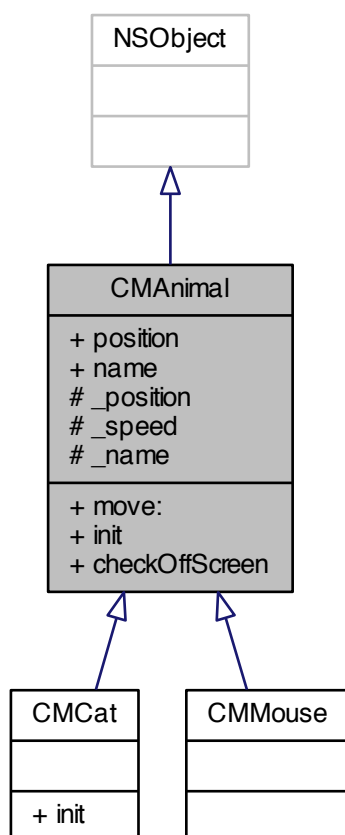
2

2 Class Documentation

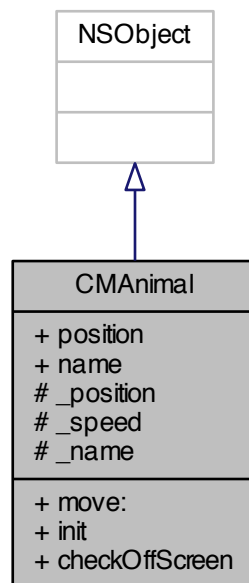
2.1 CMAnt Class Reference

```
#import <CMAnt.h>
```

Inheritance diagram for CMAntimal:



Collaboration diagram for CMAbstract:



Instance Methods

- (void) - **move:**
- (id) - **init**[implementation]
- (void) - **checkOffScreen**[implementation]

Protected Attributes

- CGPoint2D * **_position**
- int **_speed**
- NSString * **_name**

Properties

- CGPoint2D * **position**
- NSString * **name**

2.1.1 Detailed Description

Defines an abstract, base class for a playable 'thing' on the screen which can move around etc.

Author

Alex Cummaudo

Date

23 Oct 2013

2.1.2 Method Documentation

2.1.2.1 -(void) move: (dirs) dir

Move implementation for a Animal to move an animal in a direction at its speed.

Parameters

<i>dir</i>	Direction the animal is told to move in (alters x and y axis position of poosition accordingly)
------------	---

2.1.2.2 -(id) init [implementation]

Default constructor for initialising `_position` and `_speed` for all new Animals.

Returns

The self class pointer

Reimplemented in **CMCat** (p. 7).

2.1.2.3 -(void) checkOffScreen [implementation]

Off screen check that prevents any Animal from going outside the borders of the screen.

2.1.3 Member Data Documentation

2.1.3.1 -(CGPoint2D*)_position [protected]

Centrepont position of the animal.

2.1.3.2 -(int)_speed [protected]

Speed at which animals move at, set to a value of 3.

2.1.3.3 -(NSString*)_name [protected]

Name of animals, overridden by children (i.e. 'Cat' or 'Mouse')

2.1.4 Property Documentation

2.1.4.1 -(CGPoint2D*) position [read],[write],[atomic],[retain]

Readwrite property to update position, used by **CMGame** (p. 17).

2.1.4.2 -(NSString*) name [read],[atomic],[assign]

Readonly property to name, used by **CMGUI** (p. 18) and **CMNetwork** (p. 28).

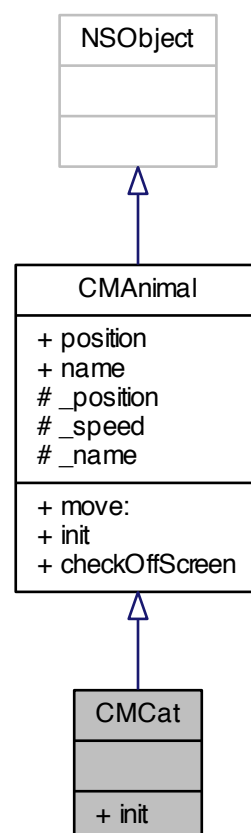
The documentation for this class was generated from the following files:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMAnimal.h**
- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMAnimal.m**

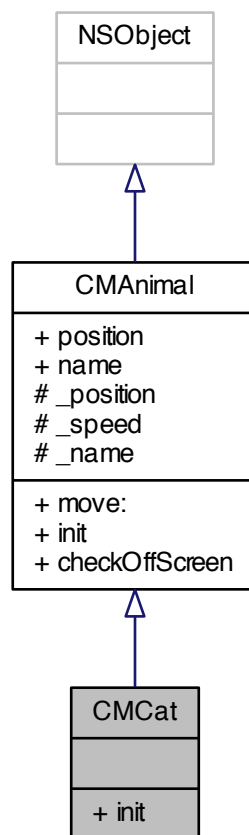
2.2 CMCat Class Reference

```
#import <CMCatMouse.h>
```

Inheritance diagram for CMCat:



Collaboration diagram for CMCat:



Instance Methods

- (id) - **init**[implementation]

Additional Inherited Members

2.2.1 Detailed Description

Defines an class for a playable chaser (i.e. the chasing cat)

Author

Alex Cummaudo

Date

24 Oct 2013

2.2.2 Method Documentation

2.2.2.1 -(id) init [implementation]

The default constructor for the cat constructs parent and sets position on lefthand-side of screen.

Returns

The self class pointer

Reimplemented from **CMAntimal** (p. 4).

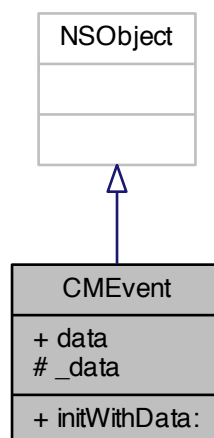
The documentation for this class was generated from the following files:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMCatMouse.h**
- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMCatMouse.m**

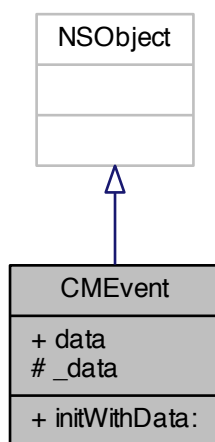
2.3 CMEvent Class Reference

```
#import <CMEvent.h>
```

Inheritance diagram for CMEvent:



Collaboration diagram for CMEvent:



Instance Methods

- (id) - **initWithData:**

Protected Attributes

- NSDictionary * **_data**

Properties

- NSDictionary * **data**

2.3.1 Detailed Description

Defines event class of what to pass a view, thereby allowing a link between each view and each model.

Author

Alex Cummaudo

Date

22 Oct 2013

2.3.2 Method Documentation

2.3.2.1 - (id) initWithData: (NSDictionary*) *initialData*

Constructor for new event object to initialise fields.

Parameters

<i>initialData</i>	Textual data to insert as data to this event
--------------------	--

Returns

The self class pointer

2.3.3 Member Data Documentation

2.3.3.1 -(NSDictionary*)_data [protected]

Textual data contained within the Event.

2.3.4 Property Documentation

2.3.4.1 -(NSDictionary*) data [read],[atomic],[assign]

Readonly property to data.

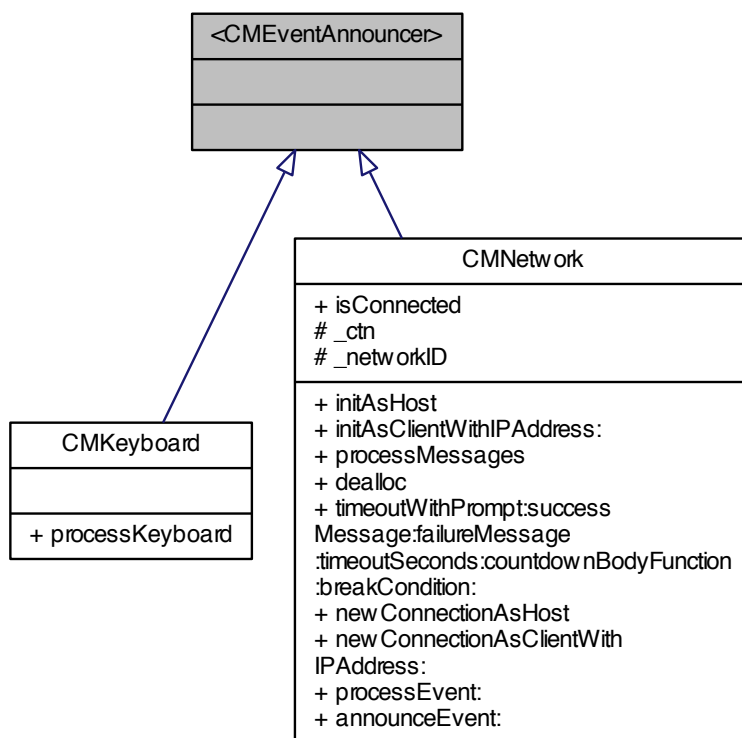
The documentation for this class was generated from the following files:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMEvent.h**
- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMEvent.m**

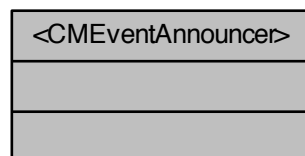
2.4 <CMEventAnnouncer> Protocol Reference

```
#include <CMEventAnnouncer.h>
```

Inheritance diagram for <CMEventAnnouncer>:



Collaboration diagram for <CMEventAnnouncer>:



2.4.1 Detailed Description

An protocol that defines all the methods that each announcer of events must implement.

Author

Alex Cummaudo

Date

20 Oct 2013

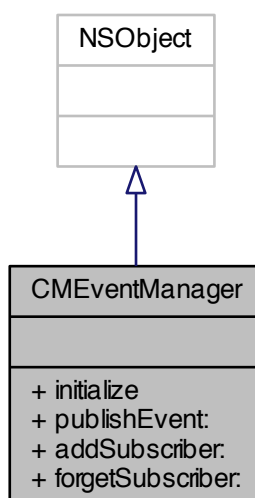
The documentation for this protocol was generated from the following file:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMEventAnnouncer.h**

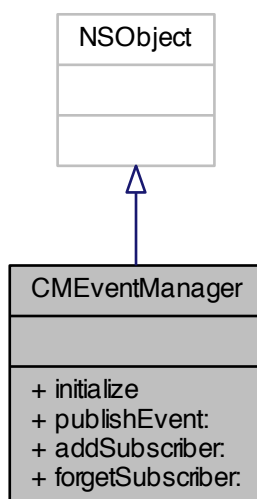
2.5 CEventManager Class Reference

```
#import <CEventManager.h>
```

Inheritance diagram for CEventManager:



Collaboration diagram for CEventManager:



Class Methods

- (void) + **initialize**
- (void) + **publishEvent:**
- (void) + **addSubscriber:**
- (void) + **forgetSubscriber:**

2.5.1 Detailed Description

Defines Event Manager class which processes each event to each kind of event subscriber.

Author

Alex Cummaudo

Date

22 Oct 2013

Note

This is a factory member class; so that clients do not need to make an instance of an **CEventManager** (p. 11) (since there's only ever going to be one processor). Therefore we invoke EventManager by calling directly on the class (i.e. [EventManager notifySubscribers:Event])

2.5.2 Method Documentation

2.5.2.1 + (void) initialize

This method is automatically called on a creation of the CEventManager object, and is used to initialise the NS-MutableArray that contains the subscribers.

This method is automatically called on a when class objects are created, to initialise the _subs Collection.

2.5.2.2 + (void) publishEvent: (CMEvent*) eventToPublish

Processes the event for each kind subscriber (i.e. CMEventProcessors ONLY!)

Parameters

<i>eventToPublish</i>	CMEvent (p. 7) to publish to all CMEventProcessors
-----------------------	---

2.5.2.3 + (void) addSubscriber: (NSObject<CMEventSubscriber>*) subscriberToAdd

Adds a subscriber to the _subs collection (can be any **NSObject** (p. ??) as long as it uses the **CMEventSubscriber** (p. 16) interface—either a: **CMEventAnnouncer** (p. 9) or **CMEventProcessor** (p. 13).

Parameters

<i>subscriberToAdd</i>	Subscriber to now manage
------------------------	--------------------------

2.5.2.4 + (void) forgetSubscriber: (NSObject<CMEventSubscriber>*) subscriberToForget

Kills a subscriber.

Parameters

<i>subscriberToKill</i>	Subscriber to now forget
-------------------------	--------------------------

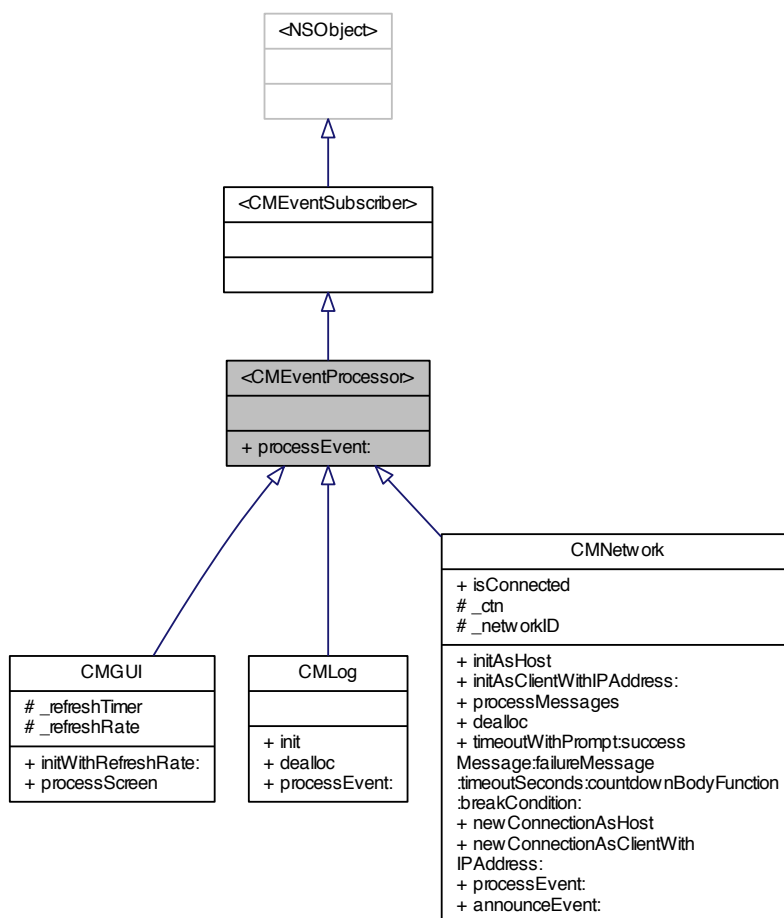
The documentation for this class was generated from the following files:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMEventManager.h**
- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMEventManager.m**

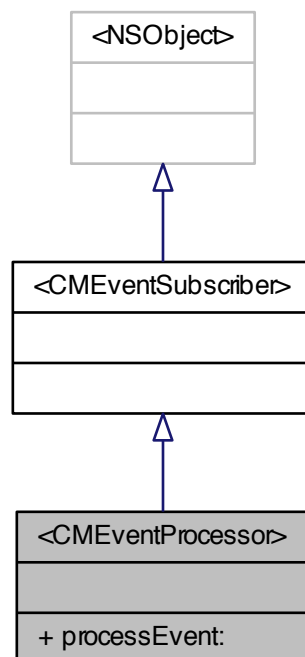
2.6 <CMEventProcessor> Protocol Reference

```
#import <CMEventProcessor.h>
```

Inheritance diagram for <CMEventProcessor>:



Collaboration diagram for <CMEventProcessor>:



Instance Methods

- (void) - **processEvent:**

2.6.1 Detailed Description

A protocol class that defines all the methods that each processor of events must implement.

Author

Alex Cummaudo

Date

20 Oct 2013

2.6.2 Method Documentation

2.6.2.1 - (void) processEvent: (CMEvent *) eData

Defines that whoever uses this interface must process an event in anyway with the given Event.

Parameters

<i>eData</i>	Event Data to process
--------------	-----------------------

Reimplemented in **CMNetwork** (p. 32), and **CMLog** (p. 25).

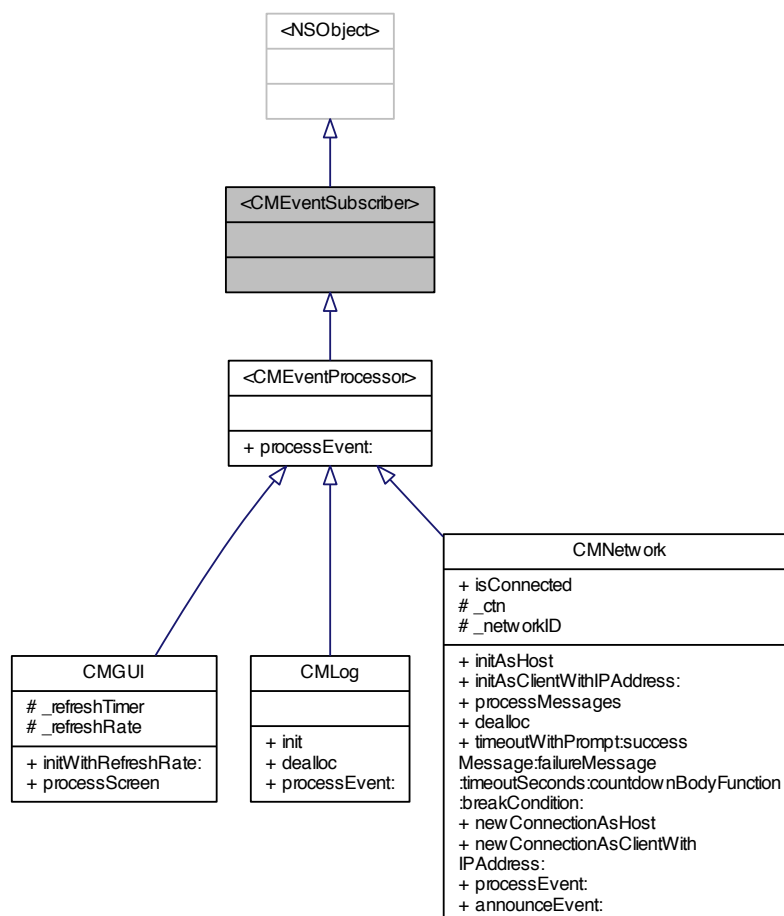
The documentation for this protocol was generated from the following file:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMEventProcessor.h**

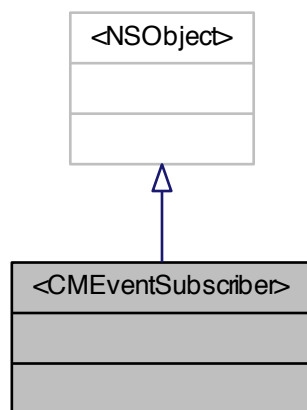
2.7 <CMEventSubscriber> Protocol Reference

```
#import <CMEventSubscriber.h>
```

Inheritance diagram for <CMEventSubscriber>:



Collaboration diagram for <CMEventSubscriber>:



2.7.1 Detailed Description

Acts as a parent to subscribers and announcers so that the Event manager knows what to manager (i.e. both Announcers and Processors, and this allows this relationship to occur via inheritance)

Author

Alex Cummaudo

Date

22 Oct 2013

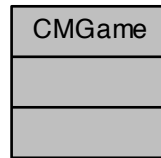
The documentation for this protocol was generated from the following file:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMEventSubscriber.h**

2.8 CMGame Class Reference

```
#import <CMGame.h>
```

Collaboration diagram for CMGame:



2.8.1 Detailed Description

Defines class for the general 'game' of the cat and mice.

Author

Alex Cummaudo

Date

16 Oct 2013

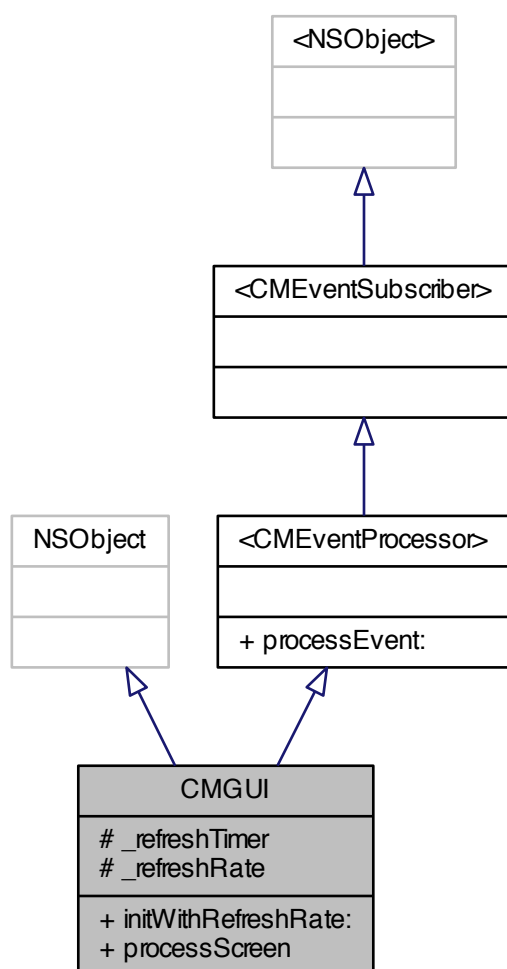
The documentation for this class was generated from the following file:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMGame.h**

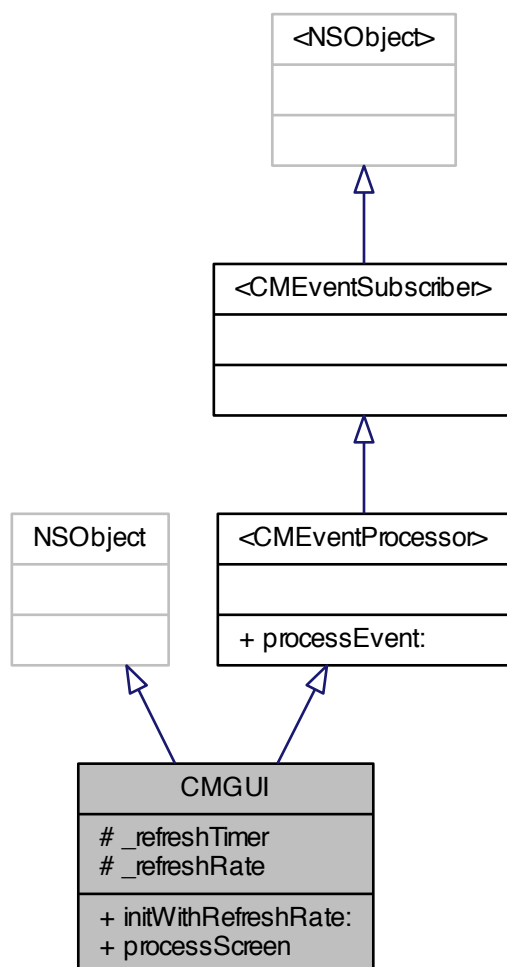
2.9 CMGUI Class Reference

```
#import <CMGUI.h>
```

Inheritance diagram for CMGUI:



Collaboration diagram for CMGUI:



Instance Methods

- (id) - **initWithRefreshRate:**
- (void) - **processScreen**

Protected Attributes

- SGTimer * **_refreshTimer**
- float **_refreshRate**

2.9.1 Detailed Description

Provides GUI View for the game to display the game on in a graphics window.

Author

Alex Cummaudo

Date

22 Oct 2013

2.9.2 Method Documentation

2.9.2.1 - (id) initWithRefreshRate: (float) *refRate*

2.9.2.2 - (void) processScreen

2.9.3 Member Data Documentation

2.9.3.1 - (SGTimer*) _refreshTimer [protected]

Timer used to refresh the screen at the by clearing the screen and resetting at refreshRate given

2.9.3.2 - (float) _refreshRate [protected]

Seconds to refresh the screen at.

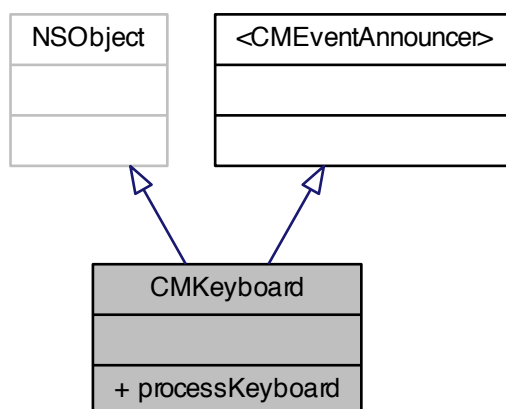
The documentation for this class was generated from the following file:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMGUI.h**

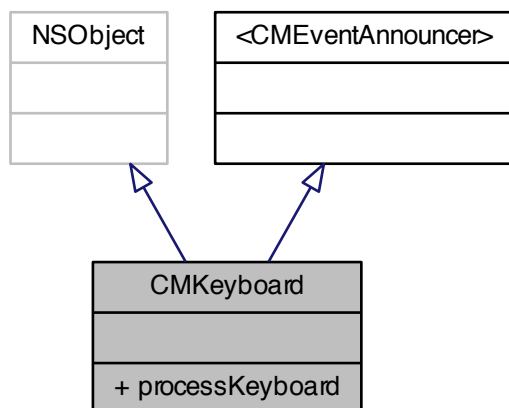
2.10 CMKeyboard Class Reference

```
#import <CMKeyboard.h>
```

Inheritance diagram for CMKeyboard:



Collaboration diagram for CMKeyboard:



Instance Methods

- (void) - **processKeyboard**

2.10.1 Detailed Description

Defines class to capture keyboard events and pass them to the processor.

Author

Alex Cummaudo

Date

22 Oct 2013

2.10.2 Method Documentation

2.10.2.1 - (void) processKeyboard

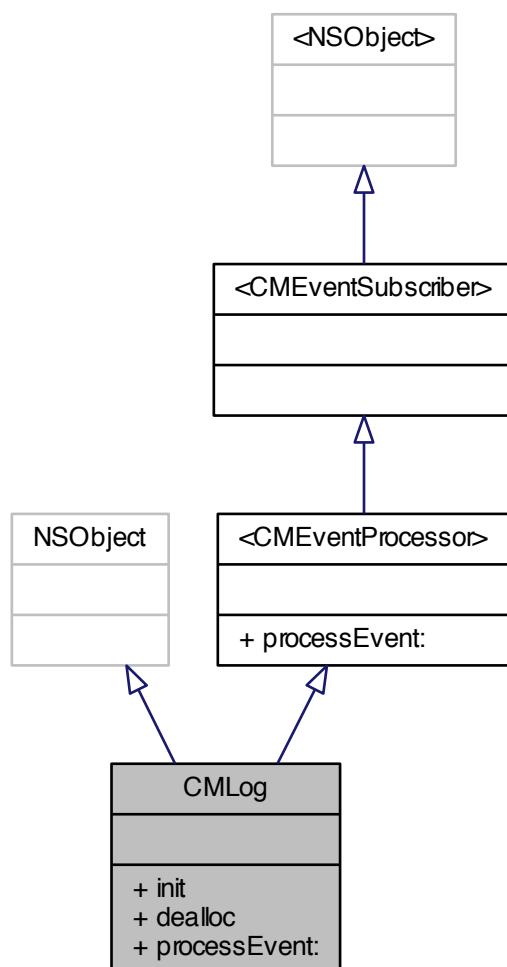
The documentation for this class was generated from the following file:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMKeyboard.h**

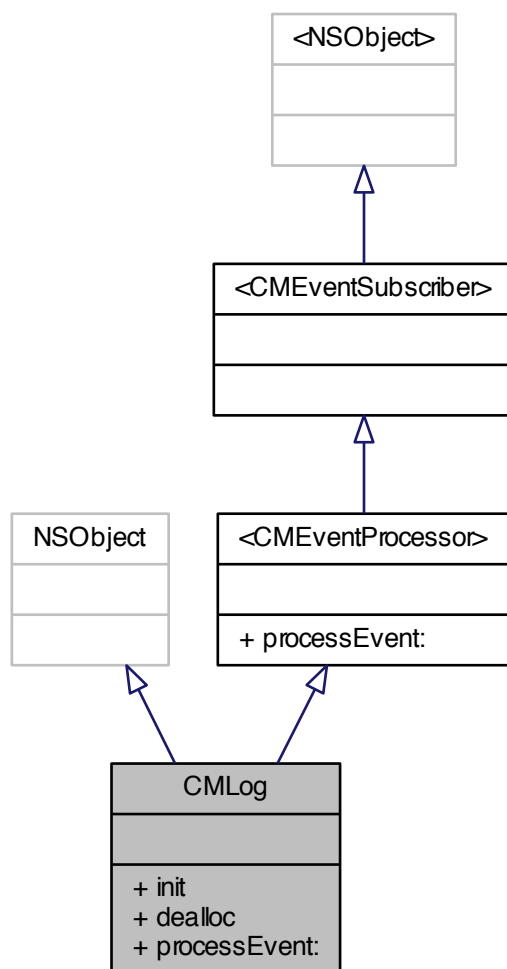
2.11 CMLog Class Reference

```
#import <CMLog.h>
```


Inheritance diagram for CMLog:



Collaboration diagram for CMLog:



Instance Methods

- (id) - **init**[implementation]
- (void) - **dealloc**[implementation]
- (void) - **processEvent:**[implementation]

2.11.1 Detailed Description

CLI view to game.

Author

Alex Cummaudo

Date

22 Oct 2013

2.11.2 Method Documentation

2.11.2.1 `-(id) init` [implementation]

Constructor asks **CMEEventManager** (p. 11) to start handling me.

2.11.2.2 `-(void) dealloc` [implementation]

Destructor asks **CMEEventManager** (p. 11) to forget about me.

2.11.2.3 `-(void) processEvent: (CMEvent *) eData` [implementation]

Processes a data depending on the the data passed by printing each key/value pair in the event data's dictionary.

Parameters

<i>eData</i>	Event Data to process
--------------	-----------------------

Reimplemented from `<CMEventProcessor>` (p. 15).

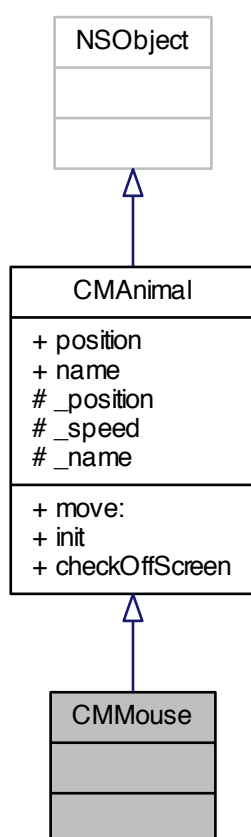
The documentation for this class was generated from the following files:

- `/Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/CMLog.h`
- `/Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/CMLog.m`

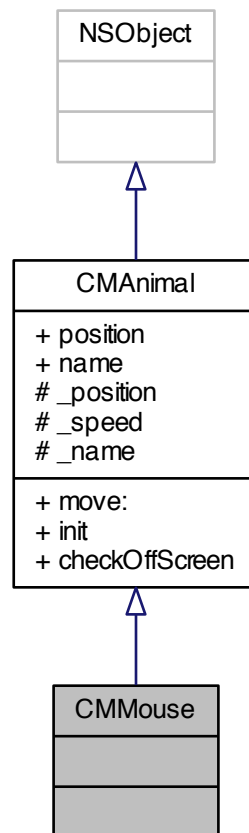
2.12 CMMouse Class Reference

```
#import <CMCatMouse.h>
```

Inheritance diagram for CMMouse:



Collaboration diagram for CMMouse:



Additional Inherited Members

2.12.1 Detailed Description

Defines an class for a playable chasee (i.e. the hunted mouse)

Author

Alex Cummaudo

Date

18 Oct 2013

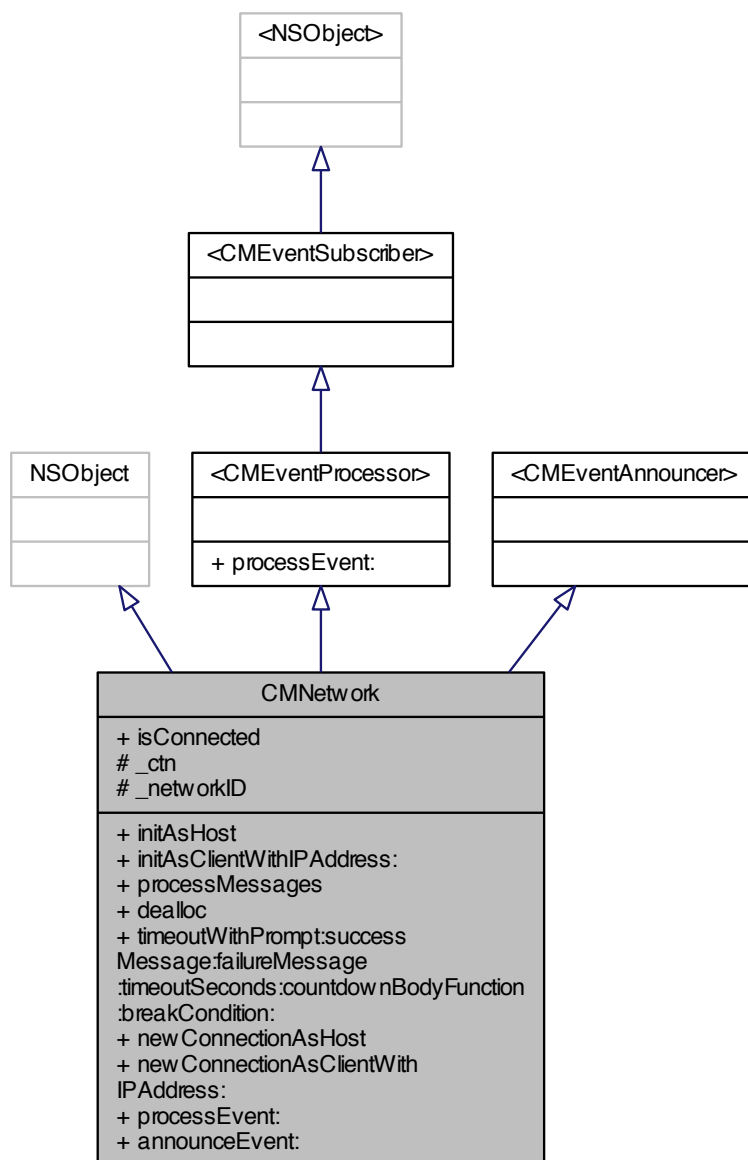
The documentation for this class was generated from the following file:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMCatMouse.h**

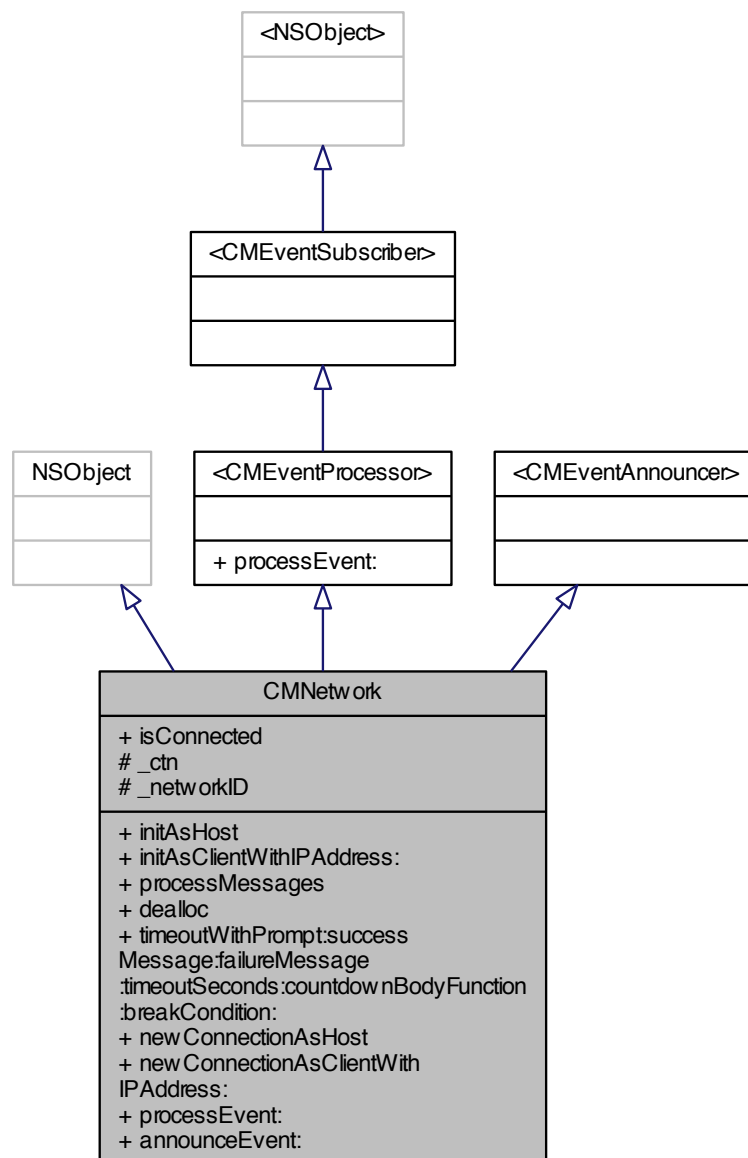
2.13 CMNetwork Class Reference

```
#import <CMNetwork.h>
```

Inheritance diagram for CMNetwork:



Collaboration diagram for CMNetwork:



Instance Methods

- (id) - **initWithHost**
- (id) - **initWithClientWithIPAddress:**
- (void) - **processMessages**
- (void) - **dealloc**[implementation]
- (void) - **timeoutWithPrompt:successMessage:failureMessage:timeoutSeconds:countdownBodyFunction:breakCondition:**[implementation]
- (SGConnection *) - **newConnectionAsHost**[implementation]
- (SGConnection *) - **newConnectionAsClientWithIPAddress:**[implementation]
- (void) - **processEvent:**[implementation]
- (void) - **announceEvent:**[implementation]

Protected Attributes

- SGConnection * **_ctn**
- NSString * **_networkID**

Properties

- BOOL **isConnected**

2.13.1 Detailed Description

Packages up data recieved from the controller and passes it to a given network.

Author

Alex Cummaudo

Date

22 Oct 2013

2.13.2 Method Documentation**2.13.2.1 - (id) initWithHost**

Constructor initiates a connection as a host.

Returns

The self class pointer

2.13.2.2 - (id) initWithIPAddress: (NSString*) ipAddr

Constructor initiates a connection to a given ip address (as a client)

Parameters

<i>ipAddr</i>	The IP Address of the host this client will connect to
---------------	--

Returns

The self class pointer

2.13.2.3 - (void) processMessages

Process messages that are being recieved (i.e. incoming network string to event)

Note

Messages recieved in the format: key:value,key:value, etc. Hence we want to parse the msg back into its event kind

2.13.2.4 - (void) dealloc [implementation]

Destructor sends goodbye message and asks **CMEEventManager** (p. 11) to forget about me and closes all connections.

2.13.2.5 - (void) timeoutWithPrompt: (NSString*) *msgPrompt* successMessage:(NSString*) *msgSuc*
failureMessage:(NSString*) *msgFail* timeoutSeconds:(int) *timeoutSecs* countdownBodyFunction:(void (^)(
countdownBody breakCondition:(BOOL (^)(*breakCondition* [implementation]

The timeout connection; runs the passed function success on a success, and error function on error; allow passing of two block objects can be passed in as parameters to the timeout.

Parameters

<i>msgPrompt</i>	Prompt message announced when timeout begins (i.e., why we're having a timeout).
<i>msgSucc</i>	Message announced when timeout did not run out and the break condition was met
<i>msgFail</i>	Message announced when timeout did ran out of timeoutSecs and the break condition was never met
<i>timeoutSecs</i>	How long to run timeout for
<i>countdownBody</i>	Function to run every second on timeout
<i>breakCondition</i>	Function that returns a bool to check whether or not the timeout should break

2.13.2.6 - (SGConnection *) newConnectionAsHost [implementation]

Initiates the connection as a host, returning true or false on a success or error.

Returns

A new host connection to work with

Define newConnection as __block to allow access to the

Force break condition to be true

2.13.2.7 - (SGConnection *) newConnectionAsClientWithIPAddress: (NSString *) ipAddr [implementation]

Initiates the connection as a client, returning true or false on a success or error.

Parameters

<i>ipAddr</i>	IP Address that this client should connect to
---------------	---

Returns

A new client connection to work with

Force break condition to be true

2.13.2.8 - (void) processEvent: (CMEvent *) eData [implementation]

Process the event data by packaging it and sending it over the network as a string (i.e. an outgoing event to a network string)

Parameters

<i>eData</i>	Event Data to process
--------------	-----------------------

Reimplemented from <CMEventProcessor> (p. 15).

2.13.2.9 - (void) announceEvent: (NSString *) message [implementation]

Sends an event to all subscribers with the given message.

Parameters

<i>message</i>	Message to announce when creating an Event
----------------	--

2.13.3 Member Data Documentation**2.13.3.1 - (SGConnection*) _ctn [protected]**

Network connection controller between client and host.

2.13.3.2 - (NSString*)_networkID [protected]

Defines a unique address of this machine.

2.13.4 Property Documentation

2.13.4.1 - (BOOL)isConnected [read],[atomic],[assign]

Gets connection status.

Returns

YES on a connection with another machine or NO when not

The documentation for this class was generated from the following files:

- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMNetwork.h**
- /Users/Alex/Dropbox/Swinburne/HIT2302 - OOP/Projects/Cat and Mouse/#2_CatMouse_ObjC_DE-Coupled/src/**CMNetwork.m**