

# Hachat Documentation

{rohrer, dittler, wedel}@informatik.hu-berlin.de

January 22, 2013

## Contents

<b>Contents</b>	<b>1</b>
<b>1 Module gui</b>	<b>3</b>
1.1 Class gui . . . . .	3
1.1.1 Methods . . . . .	3
1.1.2 Properties . . . . .	3
<b>2 Module host</b>	<b>4</b>
2.1 Variables . . . . .	4
2.2 Class Host . . . . .	4
2.2.1 Methods . . . . .	4
<b>3 Module message</b>	<b>5</b>
3.1 Functions . . . . .	5
3.2 Variables . . . . .	5
3.3 Class ByeMessage . . . . .	5
3.3.1 Methods . . . . .	5
3.3.2 Properties . . . . .	5
3.4 Class HeloMessage . . . . .	6
3.4.1 Methods . . . . .	6
3.4.2 Properties . . . . .	6
3.5 Class History . . . . .	7
3.5.1 Methods . . . . .	7
3.6 Class HistoryExchangeMessage . . . . .	7
3.6.1 Methods . . . . .	8
3.6.2 Properties . . . . .	8
3.7 Class HostExchangeMessage . . . . .	8
3.7.1 Methods . . . . .	9
3.7.2 Properties . . . . .	9
3.8 Class Message . . . . .	9
3.8.1 Methods . . . . .	9
3.8.2 Properties . . . . .	10
3.9 Class MessageException . . . . .	10
3.9.1 Methods . . . . .	10
3.9.2 Properties . . . . .	11
3.10 Class TextMessage . . . . .	11
3.10.1 Methods . . . . .	11

---

3.10.2 Properties . . . . .	12
<b>4 Module peer</b>	<b>13</b>
4.1 Variables . . . . .	13
4.2 Class Peer . . . . .	13
4.2.1 Methods . . . . .	13
<b>Index</b>	<b>15</b>

# 1 Module gui

## 1.1 Class gui

object —  
gui.gui

### 1.1.1 Methods

**\_\_init\_\_**(*self*, *parent*)

x.\_\_init\_\_(...) initializes x; see help(type(x)) for signature

Overrides: object.\_\_init\_\_ `exitit`(inherited documentation)

**empfang**(*self*, *msg*)

**ende**(*self*)

**ende\_**(*self*, *event*)

**popup\_**(*self*, *event*)

**run**(*self*)

**senden**(*self*)

**senden\_**(*self*, *event*)

### *Inherited from object*

`__delattr__()`, `__format__()`, `__getattr__()`, `__hash__()`, `__new__()`, `__reduce__()`, `__reduce_ex__()`, `__repr__()`, `__setattr__()`, `__sizeof__()`, `__str__()`, `__subclasshook__()`

### 1.1.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

## 2 Module host

This module provides the Host Class which represents connections to other Peers in a Hachat-Peer

### 2.1 Variables

Name	Description
<code>__package__</code>	<b>Value:</b> None

### 2.2 Class Host

Class representing a connection to another peer

#### 2.2.1 Methods

<code>__del__(self)</code>
<code>__init__(self, peer, hostIP, hostPort)</code>
<code>addToMsgQueue(self, msg)</code> check if message is type Message and add to Queue
<code>constructKey(cls, hostIP, hostPort)</code> Class method: construct key to identify hosts in hostlist
<code>sendHello(self)</code> will send a HELO-Message to the corresponding peer

### 3 Module message

#### 3.1 Functions

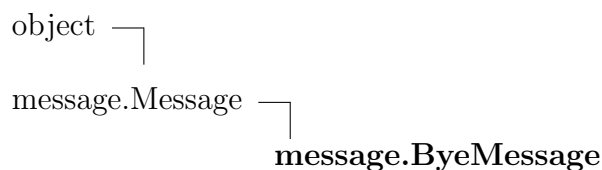
<b>toMessage</b> ( <i>string</i> )
------------------------------------

construct Message type from string
------------------------------------

#### 3.2 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

#### 3.3 Class ByeMessage



Message Type to be send when leaving

##### 3.3.1 Methods

<b>--init--</b> ( <i>self</i> , <i>origin</i> , <i>uid</i> =None)
---

build Message with supplied uid or otherwise get a random uid
---

Overrides: object.--init-- extit(inherited documentation)
---

<b>--str--</b> ( <i>self</i> )
--------------------------------

implements interface
----------------------

Overrides: object.--str--
---------------------------

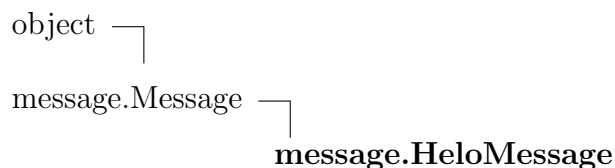
##### *Inherited from object*

`--delattr--()`, `--format--()`, `--getattr--()`, `--hash--()`, `--new--()`, `--reduce--()`, `--reduce_ex--()`, `--repr--()`, `--setattr--()`, `--sizeof--()`, `--subclasshook--()`

##### 3.3.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>--class--</code>	

### 3.4 Class *HeloMessage*



regularly sent HELO Message which exchange information on IP and Port Message layout:  
 | type | uid | recipientIP | recipientPort | senderIP | senderPort |

#### 3.4.1 Methods

```
--init__(self, recipientIP, recipientPort, senderIP, senderPort, uid=None)
```

build Message with supplied uid or otherwise get a random uid

Overrides: `object.__init__` extit(inherited documentation)

```
--str__(self)
```

implements interface

Overrides: `object.__str__`

#### *Inherited from object*

```
--delattr__(), --format__(), --getattr__(), --hash__(), --new__(), --reduce__(), --reduce_ex__(),
--repr__(), --setattr__(), --sizeof__(), --subclasshook__()
```

#### 3.4.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>--class--</code>	

### 3.5 Class History

Klasse History speichert und ueberprueft Text-Msgs

#### 3.5.1 Methods

```
__init__(self, msgLimit, hashLimit)
```

```
addMsg(self, msg)
```

```
getListMsgObjects(self, msgQuant)
```

```
getMsgHashes(self, msgQuant)
```

```
getMsgObjects(self, msgHash)
```

```
msgExists(self, msghash)
```

```
msgSafed(self, msg)
```

```
removeMsg(self, msgQuant=0, hashQuant=0)
```

### 3.6 Class HistoryExchangeMessage



for request and pushing Hosts Message layout: | type | uid | recipientIP | recipientPort |  
origin key | level | quant | liste |

### 3.6.1 Methods

```
__init__(self, recipientIP, recipientPort, origin, level, quant=None, liste=None, uid=None)
```

build Message with supplied uid or otherwise get a random uid

Overrides: object.\_\_init\_\_ extit(inherited documentation)

```
__str__(self)
```

implements interface

Overrides: object.\_\_str\_\_

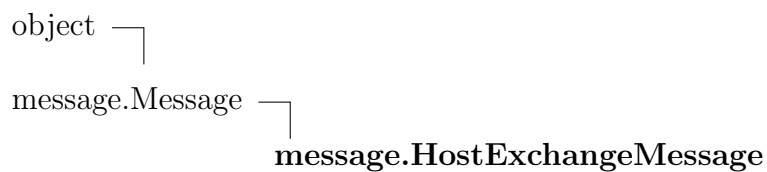
### Inherited from object

```
__delattr__(), __format__(), __getattr__(), __hash__(), __new__(), __reduce__(), __reduce_ex__(),
__repr__(), __setattr__(), __sizeof__(), __subclasshook__()
```

### 3.6.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

## 3.7 Class HostExchangeMessage



for request and pushing Hosts Message layout: | type | uid | recipientIP | recipientPort |  
origin key | level | quant | listofHosts |



### 3.7.1 Methods

```
__init__(self, recipientIP, recipientPort, origin, level, quant=None,
listofHosts=None, uid=None)
```

build Message with supplied uid or otherwise get a random uid

Overrides: object.\_\_init\_\_ extit(inherited documentation)

```
__str__(self)
```

implements interface

Overrides: object.\_\_str\_\_

#### *Inherited from object*

```
__delattr__(), __format__(), __getattr__(), __hash__(), __new__(), __reduce__(), __reduce_ex__(),
__repr__(), __setattr__(), __sizeof__(), __subclasshook__()
```

### 3.7.2 Properties

Name	Description
<i>Inherited from object</i>	
__class__	

## 3.8 Class Message



**Known Subclasses:** message.ByeMessage, message.HeloMessage, message.HistoryExchangeMessage, message.HostExchangeMessage, message.TextMessage

### 3.8.1 Methods

```
__init__(self, uid)
```

build Message with supplied uid or otherwise get a random uid

Overrides: object.\_\_init\_\_

<code>__str__(self)</code>
----------------------------

cast Message to string
------------------------

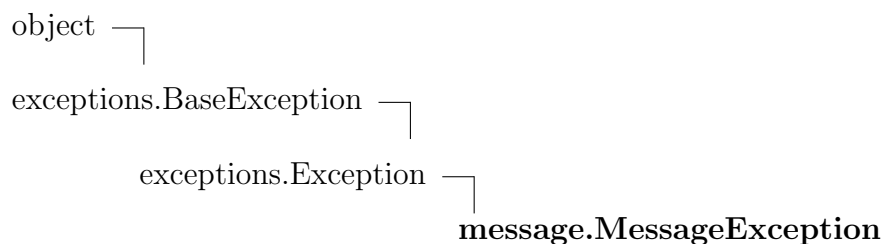
Overrides: <code>object.__str__</code>
--

***Inherited from `object`***

`__delattr__()`, `__format__()`, `__getattribute__()`, `__hash__()`, `__new__()`, `__reduce__()`, `__reduce_ex__()`,  
`__repr__()`, `__setattr__()`, `__sizeof__()`, `__subclasshook__()`

**3.8.2 Properties**

Name	Description
<i>Inherited from <code>object</code></i>	
<code>__class__</code>	

**3.9 Class `MessageException`**

Custom Exception Type for Messages

**3.9.1 Methods**

<code>__init__(self, value)</code>
------------------------------------

<code>x.__init__(...)</code> initializes x; see <code>help(type(x))</code> for signature
--

Overrides: <code>object.__init__</code> <code>exitit</code> (inherited documentation)
---

<code>__str__(self)</code>
----------------------------

<code>str(x)</code>
---------------------

Overrides: <code>object.__str__</code> <code>exitit</code> (inherited documentation)
--

***Inherited from `exceptions.Exception`***

`__new__()`

***Inherited from `exceptions.BaseException`***

`__delattr__()`, `__getattr__()`, `__getitem__()`, `__getslice__()`, `__reduce__()`, `__repr__()`,  
`__setattr__()`, `__setstate__()`, `__unicode__()`

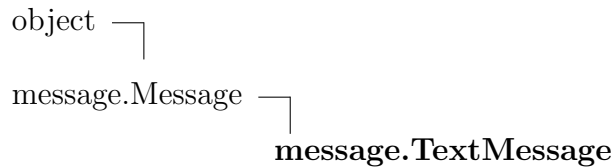
***Inherited from `object`***

`__format__()`, `__hash__()`, `__reduce_ex__()`, `__sizeof__()`, `__subclasshook__()`

### 3.9.2 Properties

Name	Description
<i>Inherited from <code>exceptions.BaseException</code></i>	
<code>args</code> , <code>message</code>	
<i>Inherited from <code>object</code></i>	
<code>__class__</code>	

## 3.10 Class `TextMessage`



normal Text Messages Message layout: | type | uid | hash | sender name | origin key | lastHop  
key | text |

### 3.10.1 Methods

**`__init__(self, name, origin, lastHop, text, uid=None)`**

build Message with supplied uid or otherwise get a random uid

Overrides: `object.__init__` extit(inherited documentation)

**`__str__(self)`**

implements interface

Overrides: `object.__str__`

***Inherited from `object`***

`--delattr--()`, `--format--()`, `--getattr--()`, `--hash--()`, `--new--()`, `--reduce--()`, `--reduce_ex--()`,  
`--repr--()`, `--setattr--()`, `--sizeof--()`, `--subclasshook--()`

### 3.10.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>--class--</code>	

## 4 Module peer

### 4.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 4.2 Class Peer

Peer Klasse

#### 4.2.1 Methods

**HistoryControl**(*self*, *neighbour*, *historyList*)

checks if the historyList from neighbour contains msgs wich are not in own History. If so, these lostMsg Hahses will be pushed back and by this the associated msgObjects are requested.

**--del--**(*self*)

**--init--**(*self*, *firstHost*=None, *port*=None, *name*='temp', *ip*=None, *testmode*=False)

**addToHosts**(*self*, *addr*)

check if already in hostlist otherwise add

**forwardMsg**(*self*, *msg*, *Oneneighbour*=None)

forwarding TextMessage, but not to initial sender if host is set, it will only forward to this single host

**generateMsgParts**(*self*, *quant*=5, *length*=2000)

generates random TextMsgs, if length > 1000 there will be more then one msg-part

**getHistory**(*self*, *neighbour*, *initial*=False)

request History from neighbour; initial is true for a initial history exchange: this will skip pushing Hashes and immediately request msg.objects

<b>maintenanceLoop</b> ( <i>self</i> )
--

<b>processMessage</b> ( <i>self</i> , <i>msg</i> , <i>fromAddr</i> )
--

processes the received messages
---------------------------------

<b>pushHistory</b> ( <i>self</i> , <i>neighbour</i> , <i>quant</i> )
--

push own List of History-Hashes to neighbour
--

<b>pushHosts</b> ( <i>self</i> , <i>neighbour</i> , <i>quant</i> )
--

give Hosts from hostExchange to a neighbour
---

<b>pushMsgObjects</b> ( <i>self</i> , <i>neighbour</i> , <i>lostMsgHashes</i> =None)
--

pushes requested msgObjects back to neighbour
---

<b>requestHosts</b> ( <i>self</i> , <i>neighbour</i> , <i>quant</i> =None)
--

request Hosts from neighbour
------------------------------

<b>sendLoop</b> ( <i>self</i> , <i>test</i> =False)
---

send Message objects of all hosts from Queue as string
--

<b>sendText</b> ( <i>self</i> , <i>text</i> )
---

<b>startRecvLoop</b> ( <i>self</i> )
--------------------------------------

general receive loop of a peer
--------------------------------

## Index

- gui (*module*), 3–5
  - gui.gui (*class*), 4–5
    - gui.gui.empfang (*method*), 5
    - gui.gui.ende (*method*), 5
    - gui.gui.ende\_ (*method*), 5
    - gui.gui.popup\_ (*method*), 5
    - gui.gui.run (*method*), 5
    - gui.gui.senden (*method*), 5
    - gui.gui.senden\_ (*method*), 5
- host (*module*), 6
  - host.Host (*class*), 6
    - host.Host.\_\_del\_\_ (*method*), 6
    - host.Host.\_\_init\_\_ (*method*), 6
    - host.Host.addToMsgQueue (*method*), 6
    - host.Host.constructKey (*class method*), 6
    - host.Host.sendHello (*method*), 6
- message (*module*), 7–14
  - message.ByeMessage (*class*), 7–8
  - message.HeloMessage (*class*), 8
  - message.History (*class*), 8–9
    - message.History.\_\_init\_\_ (*method*), 9
    - message.History.addMsg (*method*), 9
    - message.History.getListMsgObjects (*method*), 9
    - message.History.getMsgHashes (*method*), 9
    - message.History.getMsgObjects (*method*), 9
    - message.History.msgExists (*method*), 9
    - message.History.msgSafed (*method*), 9
    - message.History.removeMsg (*method*), 9
  - message.HistoryExchangeMessage (*class*), 9–10
  - message.HostExchangeMessage (*class*), 10–11
  - message.Message (*class*), 11–12
  - message.MessageException (*class*), 12–13
  - message.TextMessage (*class*), 13–14
  - message.toMessage (*function*), 7
- peer (*module*), 15–16
  - peer.Peer (*class*), 15–16
    - peer.Peer.\_\_del\_\_ (*method*), 15
    - peer.Peer.\_\_init\_\_ (*method*), 15
    - peer.Peer.addToHosts (*method*), 15
    - peer.Peer.forwardMsg (*method*), 15
    - peer.Peer.generateMsgParts (*method*), 15
    - peer.Peer.getHistory (*method*), 15
    - peer.Peer.HistoryControl (*method*), 15
    - peer.Peer.maintenanceLoop (*method*), 15
    - peer.Peer.processMessage (*method*), 16
    - peer.Peer.pushHistory (*method*), 16
    - peer.Peer.pushHosts (*method*), 16
    - peer.Peer.pushMsgObjects (*method*), 16
    - peer.Peer.requestHosts (*method*), 16
    - peer.Peer.sendLoop (*method*), 16
    - peer.Peer.sendText (*method*), 16
    - peer.Peer.startRecvLoop (*method*), 16