

InvitationListener

INHERITS FROM

Listener : Object

DECLARED IN

OpponentApp/InvitationListener.h

CLASS DESCRIPTION

The InvitationListener class receives invitations to play from the Ragnarok application. If an invitation is accepted, the inviter may then send a game to the InvitationListener.

INSTANCE VARIABLES

Inherited from Object

Class

isa;

Inherited from Listener

char
port_t
port_t
id delegate;

*portName;
listenPort;
signaturePort;

```
int timeout;  
int priority;
```

Declared in InvitationListener (none)

METHOD TYPES

Receiving an invitation - invitationFrom:onHost:RSVP:Face:length:

Receiving a game - Game:length:Side:
YourPort:MyPort:
MyFace:length:
From:onHost:

INSTANCE METHODS

invitationFrom:onHost:RSVP:Face:length:
-(int)invitationFrom : (char *) *username*

onHost : (char *) *hostname*
RSVP : (port_t) *rsvpPort*
Face : (char *) *face*
length : (int) *length*

Receives an invitation from user *username* running Ragnarok on host *hostname*. The TIFF data for the user's preferred face image is in *face*, which of length *length*. Either an **invitationAcceptedMyFace:length:** or **invitationRejected** message should be sent to *rsvpPort*.

See also: - **invitationAcceptedMyFace:length:** (ReplyListener),
- **invitationRejected** (ReplyListener)

Game:length:Side:YourPort:MyPort:MyFace:length:From:onHost:
-(int)**Game** : (char *) *game*
length : (int) *length*
Side : (int) *side*
YourPort : (port_t *) *yourPort*
MyPort : (port_t) *myPort*
MyFace : (char *) *myFaceData*

length : (int) *mflen*
From : (char *) *username*
onHost : (char *) *hostname*

Receives a game from user *username* running Ragnarok on host *hostname*. The game is a GameState object, encoded by

NXWriteRootObjectToBuffer() into *game* as *length* bytes of data. The TIFF data for the user's preferred face image is in *myFaceData*, which is of length *mflen*. Further communication with the Ragnarok application should be sent to *myPort*. The port of the RagnarokListener that will receive further messages from the Ragnarok application should be put into *yourPort*.