```
Peerlmpl
                                              Attributes
public String DEFAULT_PEER = "default@cn.com"
private String peers[0..*]
private int maxPeers
private String resources[0..*]
private String DATA_PATH = "jumbo/"
private int checkTime
private String RESOURCE_PATH = "jumbo/resources/"
private String queryExpression
private boolean isStopped
                                              Operations
public PeerImpl( String serverHost, String email, String user, String pwd, int ttl, int checkTime, int maxPeers )
public void mayAddUser( String peer )
public QueryHit[0..*] getQueryHits( )
public String getRecentQueryExpression( )
public String mkInputRequest( String msg )
public void stop()
public String[0..*] getPeers( )
public String[0..*] getResources( )
public Hashtable < String, Command > getCommands( )
private String[0..*] searchResources(String regex)
private void loadResources()
private void initCommands()
private void addResource(String resourceName)
                                  Operations Redefined From PowerPeer
protected void handleMessage( PING message )
protected void handleMessage(FPING message)
protected void handleMessage( PONG message )
protected void handleMessage(QUERY message)
public void sendQUERY( String to, String expression )
protected void handleMessage(FQUERY message)
protected void handleMessage( QUERYHIT message )
protected void handleMessage( GET message )
protected void handleMessage( POST message )
                                Operations Redefined From UserInterface
public void error(String s)
```

public void log(String s)

public void start()

public void display(Strings)

```
PowerPeer
                                   Attributes
private String serverHost
private String email
private String user
private String pwd
private int ttl
private boolean isRunning
                                   Operations
public PowerPeer( String serverHost, String email, String user, String pwd, int ttl )
public void handleMessage( PPMessage message )
protected void handleMessage( PING message )
protected void handleMessage(FPING message)
protected void handleMessage(PONG message)
protected void handleMessage(QUERY message)
protected void handleMessage(FQUERY message)
protected void handleMessage(QUERYHIT message)
protected void handleMessage(GET message)
protected void handleMessage(POST message)
protected void startListener(int sec)
protected void stopListener()
public void sendPING(String to)
protected void sendFPING( String to, String requestor, int ttl )
protected void sendPONG(String to)
public void sendQUERY( String to, String expression )
protected void sendFQUERY( String to, String requestor, String expression, int ttl )
protected void sendQUERYHIT( String to, String expression, String resources[0..*])
public void sendGET( String to, String resource )
protected void sendPOST( String to, File resource )
public String getUser( )
                   Operations Redefined From MessageHandler
```

public boolean handleMessage( Message m )

Main

Attributes

Operations

public void main( String args[0..\*] )

public void usage()