

LocationAgent.h

(motorola a925/1000 agps header api file)

PositionSource

PositionSource is the entry point for every AGPS based application. It can request position fix from GPS driver and receive from it coordinate and notifications when response become available

We can obtain this Objects using:

```
PositionSource posSource = PositionSource::NewL();
```

Fields

```
static final int POSITION_OK = 0;
static final int ERR_TIMEOUT = 1;
static final int ERR_NO_RESPONSE = 2;
static final int ERR_GPS_NOT_PRESENT = 3;
static final int ERR_GPS_DISABLED_BY_USER=4;
static final int ERR_NOT_ENOUGH_SAT = 5;
static final int ERR_RECEIVER = 6;
static final int ERR_UNAVAILABLE = 7;
static final int ERR_INTERNAL = 8;
```

Public Methods

public void generatePosition(int accuracy, int age, int timeout)

This method requests a position fix and returns immediately.

The "int age" parameter indicates the milliseconds age of a cached position fix.

The "int timeout" parameter indicates the timeout milliseconds needed to compute a position.

Objects that have been registered as listeners to the PositionSource object are notified when the fix is available or when the request times out.

The accuracy parameter indicates the accuracy of the position information (specified in meters). Setting the timeout parameter to zero indicates the system should use its default timeout value.

public void close()

Shuts down PositionSource connection with GPS driver.

public void addPositionListener(PositionListener listener)

Adds the specified position listener to receive position updates and GPS receiver errors.

public void removePositionListener(PositionListener listener)

Removes the specified position listener so that it no longer receives position updates and of GPS receiver errors.

public int getStatus()

Returns the status of the last position fix operation. See the following table:

Name Description

POSITION_OK Position fix completed successfully.

ERR_TIMEOUT Request timed out before receiver was able to complete fix

ERR_NO_RESPONSE GPS receiver not responding.

ERR_GPS_NOT_PRESENT GPS receiver is not connected to device

ERR_GPS_DISABLED_BY_USER GPS receiver was disabled by the user

ERR_NOT_ENOUGH_SAT Not enough GPS satellites to be received.

ERR_RECEIVER GPS receiver error (a catchall error code).

ERR_UNAVAILABLE Position not available (indicates GPS receiver is unable to perform fix).

ERR_INTERNAL Internal error (to be used for debugging purposes only).

Position2D

Position2D contains basic position information.

Public methods

public long getLatitude()

Returns the latitude of the position in arc minutes divided by 1/100,000, or UNAVAILABLE if latitude data is not available.

1 arc minutes = 0.0166666667 degree

public long getLongitude()

Returns the longitude of the position in arc minutes divided by 1/100,000, or UNAVAILABLE if longitude data is not available.

1 arc minutes = 0.0166666667 degree

public int hasLatLon()

Returns true if lat/lon data is available and false otherwise.

public int getLatLonAccuracy()

Returns the latitude/longitude accuracy in meter.

AggregatePosition

This object encapsulates position information, extending 2D.

Public methods

public TInt64 getTimeStamp()

Returns the time of the measurement, as the difference, in milliseconds, between the time the measurement was taken and midnight, January 1, 1970 UTC.

public long getAltitude()

Returns the altitude (above sea level) of the position in meters. The returned value is a signed integer, with negative being below sea level, and positive being above sea level. UNAVAILABLE is returned if the altitude data is not available.

public long getSpeed()

Returns the speed value in millimeter per second. UNAVAILABLE is returned if the speed data is not available.

public long getTravelDirection()

Returns the direction of travel, in degree with 0.001 precision from true North. The return value ranges from 0 to 360,000, with 90,000 indicating East. UNAVAILABLE is returned if the direction data is not available.

public int getPointingDirection()

Returns the current pointing direction of device, in degrees with 0.001 precision from true north. The return value ranges from 0 to 360,000, with 90,000 indicating East. The pointing direction is different from travel direction. Assuming a car is currently pointing to north, and the driver is doing reverse operation on the car. In this case, the pointing direction is north, and the travel direction is south. UNAVAILABLE is returned if the direction data is not available.

public int hasAltitude()

Returns true if altitude is available in this position object and false otherwise.

public int hasSpeed()

Returns true if speed is available in this position object and false otherwise.

public int hasTravelDirection()

Returns true if travel direction is available in this position object and false otherwise.

public int hasPointingDirection()

Returns true if pointing direction is available in this position object and false otherwise.

PositionListener

An interface implemented by applications that need to be informed when responses to position requests become available.

Public Methods

public void newPosition(AggregatePosition pos)

When PositionSource.generatePosition() generate a position fix results the system call newPosition(AggregatePosition) AggregatePosition contains many info about position.

If a satellite fix request was unsuccessful the AggregatePosition is set to null by listener.

Giuseppe Morreale
guGPS project on www.sourceforge.net
morreale.g@tiscali.it

send me an e-mail if you find this document useful.