Android Disconnect Reasons

Disconnect Cause

Describes the cause of a disconnected call. This always includes a code describing the generic cause of the disconnect. Optionally, it may include a label and/or description to display to the user. It is the responsibility of the ConnectionService to provide localized versions of the label and description. It also may contain a reason for the disconnect, which is intended for logging and not for display to the user.

Summary

Constants

int ANSWERED_ELSEWHERE

Disconnected because the user did not locally answer the incoming call, but it was answered on another device where the call was ringing.

int BUSY

Disconnected because the other party was busy.

int CALL_PULLED

Disconnected because the call was pulled from the current device to another device.

int CANCELED

Disconnected because it has been canceled.

int CONNECTION_MANAGER_NOT_SUPPORTED

Disconnected because the connection manager did not support the call.

int	ERROR Disconnected because there was an error, such as a problem with the network.
int	LOCAL Disconnected because of a local user-initiated action, such as hanging up.
int	MISSED Disconnected because there was no response to an incoming call.
int	OTHER Disconnected for reason not described by other disconnect codes.
int	REJECTED Disconnected because the user rejected an incoming call.
int	REMOTE Disconnected because of a remote user-initiated action, such as the other party hanging up up.
int	RESTRICTED Disconnected because of a restriction on placing the call, such as dialing in airplane mode.
int	UNKNOWN Disconnected because of an unknown or unspecified reason.

Inherited constants

From interface android.os.Parcelable

Fields

public static finalCreator<DisconnectCause>	CREATOR

Public constructors

DisconnectCause(int code)

Creates a new DisconnectCause.

DisconnectCause(int code, String reason)

Creates a new DisconnectCause.

DisconnectCause(int code, CharSequence label, CharSequence description, String reason)

Creates a new DisconnectCause.

DisconnectCause(int code, CharSequence label, CharSequence description, String reason, int toneToPlay)

Creates a new DisconnectCause.

Public methods

int	describeContents() Describe the kinds of special objects contained in this Parcelable instance's marshaled representation.
boolean	equals(Object o) Indicates whether some other object is "equal to" this one.

int	getCode() Returns the code for the reason for this disconnect.
CharSequence	getDescription() Returns a description which explains the reason for the disconnect cause and is for display in the user interface.
CharSequence	getLabel() Returns a short label which explains the reason for the disconnect cause and is for display in the user interface.
String	getReason() Returns an explanation of the reason for the disconnect.
int	getTone() Returns the tone to play when disconnected.
int	hashCode() Returns a hash code value for the object.
String	toString() Returns a string representation of the object.
void	writeToParcel(Parcel destination, int flags) Flatten this object in to a Parcel.

Inherited methods

From class java.lang.Object

From interface android.os.Parcelable

Constants

ANSWERED_ELSEWHERE

Added in API level 25

public static final int ANSWERED_ELSEWHERE

Disconnected because the user did not locally answer the incoming call, but it was answered on another device where the call was ringing.

Constant Value: 11 (0x0000000b)

BUSY

Added in API level 23

public static final int BUSY

Disconnected because the other party was busy.

Constant Value: 7 (0x00000007)

CALL_PULLED

Added in API level 25

public static final int CALL_PULLED

Disconnected because the call was pulled from the current device to another device.

Constant Value: 12 (0x0000000c)

CANCELED

Added in API level 23

public static final int CANCELED

Disconnected because it has been canceled.

Constant Value: 4 (0x00000004)

CONNECTION_MANAGER_NOT_SUPPORTED

Added in API level 23

public static final int CONNECTION_MANAGER_NOT_SUPPORTED

Disconnected because the connection manager did not support the call. The call will be tried again without a connection manager. See PhoneAccount#CAPABILITY_CONNECTION_MANAGER.

Constant Value: 10 (0x0000000a)

ERROR

Added in API level 23

public static final int ERROR

Disconnected because there was an error, such as a problem with the network.

Constant Value: 1 (0x00000001)

LOCAL

Added in API level 23

public static final int LOCAL

Disconnected because of a local user-initiated action, such as hanging up.

Constant Value: 2 (0x00000002)

MISSED

Added in API level 23

public static final int MISSED

Disconnected because there was no response to an incoming call.

Constant Value: 5 (0x00000005)

OTHER

Added in API level 23

public static final int OTHER

Disconnected for reason not described by other disconnect codes.

Constant Value: 9 (0x00000009)

REJECTED

Added in API level 23

public static final int REJECTED

Disconnected because the user rejected an incoming call.

Constant Value: 6 (0x00000006)

REMOTE

Added in API level 23

public static final int REMOTE

Disconnected because of a remote user-initiated action, such as the other party hanging up up.

Constant Value: 3 (0x00000003)

RESTRICTED

Added in API level 23

public static final int RESTRICTED

Disconnected because of a restriction on placing the call, such as dialing in airplane mode.

Constant Value: 8 (0x00000008)

UNKNOWN

Added in API level 23

public static final int UNKNOWN

Disconnected because of an unknown or unspecified reason.

Constant Value: 0 (0x00000000)

Fields

CREATOR

Added in API level 23

public static final Creator<DisconnectCause> CREATOR

Public constructors

DisconnectCause

Added in API level 23

public DisconnectCause (int code)

Creates a new DisconnectCause.

Parameters

code	int: The code for the disconnect cause.

DisconnectCause

Added in API level 23

```
public DisconnectCause (int code,
                String reason)
```

Creates a new DisconnectCause.

Parameters

code	int: The code for the disconnect cause.
reason	String: The reason for the disconnect.

DisconnectCause

Added in API level 23

```
public DisconnectCause (int code,
                CharSequence label,
                CharSequence description,
                String reason)
```

Creates a new DisconnectCause.

Parameters

code	int: The code for the disconnect cause.
label	CharSequence: The localized label to show to the user to explain the disconnect.
description	CharSequence: The localized description to show to the user to explain the disconnect.
reason	String: The reason for the disconnect.

DisconnectCause

Added in API level 23

```
public DisconnectCause (int code,
                CharSequence label,
```

CharSequence description,

String reason,

int toneToPlay)

Creates a new DisconnectCause.

Parameters

code	int: The code for the disconnect cause.
label	CharSequence: The localized label to show to the user to explain the disconnect.
description	CharSequence: The localized description to show to the user to explain the disconnect.
reason	String: The reason for the disconnect.
toneToPlay	int: The tone to play on disconnect, as defined in ToneGenerator.

Public methods

describeContents

Added in API level 23

public int describeContents ()

Describe the kinds of special objects contained in this Parcelable instance's marshaled representation. For example, if the object will include a file descriptor in the output of writeToParcel(android.os.Parcel, int), the return value of this method must include the CONTENTS_FILE_DESCRIPTOR bit.

Returns

int	a bitmask indicating the set of special object types marshaled by this Parcelable object instance. Value is either 0 or CONTENTS_FILE_DESCRIPTOR

equals

Added in API level 23

public boolean equals (Object o)

Indicates whether some other object is "equal to" this one.

The equals method implements an equivalence relation on non-null object references:

It is reflexive: for any non-null reference value x, x.equals(x) should return true.

It is symmetric: for any non-null reference values x and y, x.equals(y) should return true if and only ify.equals(x) returns true.

It is transitive: for any non-null reference values x, y, and z, if x.equals(y) returns true and y.equals(z)returns true, then x.equals(z) should return true.

It is consistent: for any non-null reference values x and y, multiple invocations of x.equals(y) consistently return true or consistently return false, provided no information used in equals comparisons on the objects is modified.

For any non-null reference value x, x.equals(null) should return false.

The equals method for class Object implements the most discriminating possible equivalence relation on objects; that is, for any non-null reference values x and y, this method returns true if and only if x and y refer to the same object (x == y has the value true).

Note that it is generally necessary to override the hashCode method whenever this method is overridden, so as to maintain the general contract for the hashCode method, which states that equal objects must have equal hash codes.

Parameters

o	Object: the reference object with which to compare.

Returns

boolean	true if this object is the same as the obj argument; false otherwise.

getCode

Added in API level 23

public int getCode ()

Returns the code for the reason for this disconnect.

Returns

int	The disconnect code.

getDescription

Added in API level 23

public CharSequence getDescription ()

Returns a description which explains the reason for the disconnect cause and is for display in the user interface. This optional text is generally a longer and more descriptive version of getLabel(), however it can exist even if getLabel() is empty. The In-Call UI should display this relatively prominently; the traditional implementation displays this as an alert dialog. The ConnectionService is responsible for providing and localizing this message. If there is no string provided, returns null.

Returns

CharSequence	The disconnect description.

getLabel

Added in API level 23

public CharSequence getLabel ()

Returns a short label which explains the reason for the disconnect cause and is for display in the user interface. If not null, it is expected that the In-Call UI should display this text where it would normally display the call state ("Dialing", "Disconnected") and is therefore expected to be relatively small.
The ConnectionService is responsible for providing and localizing this label. If there is no string provided, returns null.

Returns	
CharSequence	The disconnect label.

getReason

Added in API level 23

public String getReason ()

Returns an explanation of the reason for the disconnect. This is not intended for display to the user and is used mainly for logging.

Returns	
String	The disconnect reason.

getTone

Added in API level 23

public int getTone ()

Returns the tone to play when disconnected.

Returns	
int	the tone as defined in ToneGenerator to play when disconnected.

hashCode

Added in API level 23

public int hashCode ()

Returns a hash code value for the object. This method is supported for the benefit of hash tables such as those provided by HashMap.

The general contract of hashCode is:

Whenever it is invoked on the same object more than once during an execution of a Java application, the hashCode method must consistently return the same integer, provided no information used in equalscomparisons on the object is modified. This integer need not remain consistent from one execution of an application to another execution of the same application.

If two objects are equal according to the equals(Object) method, then calling the hashCode method on each of the two objects must produce the same integer result.

It is not required that if two objects are unequal according to the equals(java.lang.Object) method, then calling the hashCode method on each of the two objects must produce distinct integer results. However, the programmer should be aware that producing distinct integer results for unequal objects may improve the performance of hash tables.

As much as is reasonably practical, the hashCode method defined by class Object does return distinct integers for distinct objects. (This is typically implemented by converting the internal address of the object into an integer, but this implementation technique is not required by the Java™ programming language.)

Returns

int	a hash code value for this object.

toString

Added in API level 23

public String toString ()

Returns a string representation of the object. In general, the toString method returns a string that "textually represents" this object. The result should be a concise but informative representation that is easy for a person to read. It is recommended that all subclasses override this method.

The toString method for class Object returns a string consisting of the name of the class of which the object is an instance, the at-sign character `@', and the unsigned hexadecimal representation of the hash code of the object. In other words, this method returns a string equal to the value of:

```
 getClass().getName() + '@' + Integer.toHexString(hashCode())
```

Returns

String	a string representation of the object.s

writeToParcel

Added in API level 23

```
public void writeToParcel (Parcel destination,
                int flags)
```

Flatten this object in to a Parcel.

Parameters

destination	Parcel: The Parcel in which the object should be written.
flags	int: Additional flags about how the object should be written. May be 0 or Parcelable.PARCELABLE_WRITE_RETURN_VALUE. Value is either 0 or a combination of Parcelable.PARCELABLE_WRITE_RETURN_VALUE, and android.os.Parcelable.PARCELABLE_ELIDE_DUPLICATES