Package com.sun.javacard.apduio

Provides a client API for exchanging APDUs with smart cards.

See:
Description

Interface Summary	
CadClientInterface	An interface describing a client connection to a card.

Class Summary	
Apdu	This class represent a pair of C-APDU and R-APDU.
CadDevice	Factory class producing CardClientInterface instances (connections to a card)

Exception Summary	
CadTransportException	This abstract class is the parent class of all CAD exception classes.
T1Exception	This class represents error states for T=1 CAD devices.
TLP224Exception	Exceptions related to TLP224 protocol violations

Package com.sun.javacard.apduio Description

Provides a client API for exchanging APDUs with smart cards.

It is used internally by several components of the Java Card development kit.

com.sun.javacard.apduio

Class Apdu

```
java.lang.Object
  └com.sun.javacard.apduio.Apdu
```

```
public class Apdu
```

extends java.lang.Object

This class represent a pair of C-APDU and R-APDU. It is used internally by several components of the Java Card development kit.

Field Summary

static int	**APDU_TYPE_MASK** Mask to extract APDU type information: either ISO ecoding or not.
static int	**CASE_1** The value which indicates this Apdu is an ISO 7816-3 Case 1 apdu
static int	**CASE_2E** The value which indicates this Apdu is an ISO 7816-3 Case 2 Extended apdu
static int	**CASE_2S** The value which indicates this Apdu is an ISO 7816-3 Case 2 Short apdu
static int	**CASE_3E** The value which indicates this Apdu is an ISO 7816-3 Case 3 Extended apdu
static int	**CASE_3S** The value which indicates this Apdu is an ISO 7816-3 Case 3 Short apdu

static int	**CASE_4E** The value which indicates this Apdu is an ISO 7816-3 Case 4 Extended apdu
static int	**CASE_4S** The value which indicates this Apdu is an ISO 7816-3 Case 4 Short apdu
static int	**CLA** The offset in the command array to the ISO 7816-3 CLA octet.
byte[]	**command** Internal representation of the C-APDU header
byte[]	**dataIn** Data part of the C-APDU
byte[]	**dataOut** Data part of the R-APDU
static int	**INS** The offset in the command array to the ISO 7816-3 INS octet.
boolean	**isExtended** Extended APDU flag
int	**Lc** Value of Lc
int	**Le** Value of Le
static int	**LOGICAL_CHN_MASK** Mask to extract channel information out of the CLA byte.
static int	**P1** The offset in the command array to the ISO 7816-3 P1 octet.
static int	**P2** The offset in the command array to the ISO 7816-3 P2 octet.
static int	**P3** The offset in the command array to the ISO 7816-3 P3 octet.
byte[]	**sw1sw2** Status as byte array

Constructor Summary

Apdu()	Creates a new instance of Apdu.

Method Summary

int	**getCase**() Return the ISO 7816-3 Case of this Apdu
byte	**getChannelInfo**() Determines the channel number
byte[]	**getCommand**() Get internal representation of the header of the C-APDU
byte[]	**getCommandApduBytes**() returns the command APDU as a byte array.
byte[]	**getDataIn**() Get the data part of the C-APDU
byte[]	**getDataOut**() Get the data part of the R-APDU
int	**getLc**() Get Lc
int	**getLe**() Get Le
byte[]	**getResponseApduBytes**() returns the response APDU as a byte array.
int	**getStatus**() Get status (SW bytes) as integer
byte[]	**getSw1Sw2**() Get status bytes
void	**setDataIn**(byte[] dataIn) Set the data of the C-APDU
void	**setDataIn**(byte[] dataIn, int length) Set the data of the C-APDU
void	**setDataOut**(byte[] dataOut) Set the data part of the R-APDU

void	**setDataOut**(byte[] dataOut, int length) Set the data part of the R-APDU
void	**setLc**(int Lc) Set the value of Lc
void	**setLe**(int Le) Set Le
java.lang.String	**toString**() Generate a string representation of this Apdu in the ApduTool output format.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Field Detail

CLA

public static final int **CLA**

The offset in the command array to the ISO 7816-3 CLA octet.

See Also:
Constant Field Values

INS

public static final int **INS**

The offset in the command array to the ISO 7816-3 INS octet.

See Also:
Constant Field Values

P1

```
public static final int P1
```

The offset in the command array to the ISO 7816-3 P1 octet.

See Also:
Constant Field Values

P2

```
public static final int P2
```

The offset in the command array to the ISO 7816-3 P2 octet.

See Also:
Constant Field Values

P3

```
public static final int P3
```

The offset in the command array to the ISO 7816-3 P3 octet.

See Also:
Constant Field Values

CASE_1

```
public static final int CASE_1
```

The value which indicates this Apdu is an ISO 7816-3 Case 1 apdu

See Also:
Constant Field Values

CASE_2S

```
public static final int CASE_2S
```

The value which indicates this Apdu is an ISO 7816-3 Case 2 Short apdu

See Also:
Constant Field Values

CASE_3S

```
public static final int CASE_3S
```

The value which indicates this Apdu is an ISO 7816-3 Case 3 Short apdu

See Also:
Constant Field Values

CASE_4S

```
public static final int CASE_4S
```

The value which indicates this Apdu is an ISO 7816-3 Case 4 Short apdu

See Also:
Constant Field Values

CASE_2E

`public static final int` **`CASE_2E`**

The value which indicates this Apdu is an ISO 7816-3 Case 2 Extended apdu

See Also:
Constant Field Values

CASE_3E

`public static final int` **`CASE_3E`**

The value which indicates this Apdu is an ISO 7816-3 Case 3 Extended apdu

See Also:
Constant Field Values

CASE_4E

`public static final int` **`CASE_4E`**

The value which indicates this Apdu is an ISO 7816-3 Case 4 Extended apdu

See Also:
Constant Field Values

LOGICAL_CHN_MASK

`public static final int` **`LOGICAL_CHN_MASK`**

Mask to extract channel information out of the CLA byte.

See Also:
Constant Field Values

APDU_TYPE_MASK

`public static final int` **`APDU_TYPE_MASK`**

Mask to extract APDU type information: either ISO ecoding or not.

See Also:
Constant Field Values

command

`public byte[]` **`command`**

Internal representation of the C-APDU header

Lc

`public int` **`Lc`**

Value of Lc

dataIn

`public byte[]` **`dataIn`**

Data part of the C-APDU

Le

`public int` **`Le`**

Value of Le

---

dataOut

`public byte[] dataOut`

Data part of the R-APDU

---

sw1sw2

`public byte[] sw1sw2`

Status as byte array

---

isExtended

`public boolean isExtended`

Extended APDU flag

Constructor Detail

Apdu

`public Apdu()`

Creates a new instance of Apdu.

Method Detail

getCommand

`public byte[] **getCommand**()`

Get internal representation of the header of the C-APDU

Returns:
The header of the C-APDU

---

getDataIn

`public byte[] **getDataIn**()`

Get the data part of the C-APDU

Returns:
The data

---

setDataIn

`public void **setDataIn**(byte[] dataIn)`

Set the data of the C-APDU

Parameters:
`dataIn` - The data

---

setLc

`public void **setLc**(int Lc)`

Set the value of Lc

Parameters:
`Lc` - value of the Lc

---

setDataIn

```
public void setDataIn(byte[] dataIn,
                      int length)
```

Set the data of the C-APDU

Parameters:

`dataIn` - The data
`length` - the data length

---

getLc

```
public int getLc()
```

Get Lc

Returns:

The Lc

---

setDataOut

```
public void setDataOut(byte[] dataOut)
```

Set the data part of the R-APDU

Parameters:

`dataOut` - The data

---

setLe

```
public void setLe(int Le)
```

Set Le

Parameters:

`Le` - value of Le

---

setDataOut

```
public void setDataOut(byte[] dataOut,
                       int length)
```

Set the data part of the R-APDU

Parameters:

`dataOut` - The data
`length` - The data length

---

getLe

```
public int getLe()
```

Get Le

Returns:

The Le

---

getDataOut

```
public byte[] getDataOut()
```

Get the data part of the R-APDU

Returns:

The data part of the R-APDU

---

getSw1Sw2

```
public byte[] getSw1Sw2()
```

Get status bytes

Returns:
The status bytes

---

getStatus

```
public int getStatus()
```

Get status (SW bytes) as integer

Returns:
The status

---

getCase

```
public int getCase()
```

Return the ISO 7816-3 Case of this Apdu

Returns:
An int value indicating the ISO case.

---

getChannelInfo

```
public byte getChannelInfo()
```

Determines the channel number

Returns:
The channel number

---

getCommandApduBytes

`public byte[] getCommandApduBytes()`

returns the command APDU as a byte array.

Returns:
The command APDU

---

getResponseApduBytes

`public byte[] getResponseApduBytes()`

returns the response APDU as a byte array.

Returns:
The response APDU

---

toString

`public java.lang.String toString()`

Generate a string representation of this Apdu in the ApduTool output format.

Overrides:
`toString` in class `java.lang.Object`
Returns:
String representation of this Apdu

---

com.sun.javacard.apduio

Interface CadClientInterface

```
public interface CadClientInterface
```

An interface describing a client connection to a card.

Method Summary

void	**exchangeApdu**(Apdu apdu) Exchanges com.sun.javacard.apduio.Apdu with a card.
void	**powerDown**() Powerdown a card or simulator and then disconnect
void	**powerDown**(boolean disconnect) Send powerdown to a card
byte[]	**powerUp**() Send powerup to a card or a simulator

Method Detail

exchangeApdu

```
void exchangeApdu(Apdu apdu)
                  throws java.io.IOException,
                         CadTransportException
```

Exchanges com.sun.javacard.apduio.Apdu with a card.

Parameters:

`apdu` - A pair of C-APDU and R-APDU

Throws:

`java.io.IOException` - Communication error

`CadTransportException` - Protocol error

---

powerDown

```
void powerDown(boolean disconnect)
               throws java.io.IOException,
                      CadTransportException
```

Send powerdown to a card

Parameters:

`disconnect` - This parameter is applicable to card simulators only. true = disconnect from the simulator false = powedown, but leave the connection open

Throws:

`java.io.IOException` - Communication error

`CadTransportException` - Protocol error

---

powerDown

```
void powerDown()
               throws java.io.IOException,
                      CadTransportException
```

Powerdown a card or simulator and then disconnect

Throws:

`java.io.IOException` - Communication error

`CadTransportException` - Protocol error

---

powerUp

```
byte[] powerUp()
               throws java.io.IOException,
                      CadTransportException
```

Send powerup to a card or a simulator

Returns:

ATR bytes

Throws:

`java.io.IOException` - Communication error

`CadTransportException` - Protocol error

---

com.sun.javacard.apduio

Class CadDevice

```
java.lang.Object
  └com.sun.javacard.apduio.CadDevice
```

```
public abstract class CadDevice
```

extends java.lang.Object

Factory class producing CardClientInterface instances (connections to a card)

Field Summary

protected static java.util.ResourceBundle	**_messages** Properties file with messages
protected boolean	**debug** Internal debug flag.
protected byte	**deviceProtocol** The currentry used protocol
protected java.io.InputStream	**in** The input stream which data is read from.
protected java.io.OutputStream	**out** The output stream to which data is written.
static byte	**PROTOCOL_PCSC** PC/SC constant
static byte	**PROTOCOL_T0** Protocol constant
static byte	**PROTOCOL_T1** Protocol constant

static byte	**PROTOCOL_TCL** Protocol constant

Constructor Summary

protected	**CadDevice**(byte devProtocolType) this form of constructor is used by PCSC streams are not applicable
protected	**CadDevice**(byte devProtocolType, java.io.InputStream in, java.io.OutputStream out) this form of constructor is used by T=0, T=1 implementations

Method Summary

void	**close**() Flush and close the input and output streams.
static CadClientInterface	**getCadClientInstance**(byte protocolType, java.io.InputStream in, java.io.OutputStream out) A factory method
byte	**getDeviceProtocol**() Get the value of the protocol byte
static CadClientInterface	**getPCSCClientInstance**(int readerNumber) A factory method for connecting to a PCSC card readed

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field Detail

_messages

protected static java.util.ResourceBundle **_messages**

Properties file with messages

---

in

protected java.io.InputStream **in**

The input stream which data is read from. This field is used to store the reference to the input stream which an instance of the Cad class reads from.

---

out

`protected java.io.OutputStream` **`out`**

The output stream to which data is written. This field is used to store the reference to the output stream which an instance of the Cad class writes to.

---

debug

`protected boolean` **`debug`**

Internal debug flag. Set the system property apduIODebug to 0x01 to activate transport level debugging.

PROTOCOL_T0

`public static final byte` **`PROTOCOL_T0`**

Protocol constant

See Also:
Constant Field Values

PROTOCOL_T1

`public static final byte` **`PROTOCOL_T1`**

Protocol constant

See Also:
Constant Field Values

PROTOCOL_TCL

```
public static final byte PROTOCOL_TCL
```

Protocol constant

See Also:
Constant Field Values

PROTOCOL_PCSC

```
public static final byte PROTOCOL_PCSC
```

PC/SC constant

See Also:
Constant Field Values

---

deviceProtocol

```
protected byte deviceProtocol
```

The currentry used protocol

Constructor Detail

CadDevice

```
protected CadDevice(byte devProtocolType,
                    java.io.InputStream in,
                    java.io.OutputStream out)
```

this form of constructor is used by T=0, T=1 implementations

Parameters:
`devProtocolType` - Procol constant: T1 or T1
`in` - Input stream from a simulator or a TLP224 card reader
`out` - Output stream to a simulator or a TLP224 card reader

CadDevice

```
protected CadDevice(byte devProtocolType)
```

this form of constructor is used by PCSC streams are not applicable

Parameters:
`devProtocolType` - Protocol type (PCSC)

Method Detail

getDeviceProtocol

```
public byte getDeviceProtocol()
```

Get the value of the protocol byte

Returns:
the value of the protocol byte

---

getCadClientInstance

```
public static CadClientInterface getCadClientInstance(byte protocolType,
                                                      java.io.InputStream in,
                                                      java.io.OutputStream out)
```

A factory method

Parameters:
`protocolType` - Protocol constant
`in` - Input stream from a simulator
`out` - Output stream to a simulator
Returns:
New instance of a card connection

---

getPCSCClientInstance

```
public static CadClientInterface getPCSCClientInstance(int readerNumber)
```

A factory method for connecting to a PCSC card readed

Parameters:

`readerNumber` - PC/SC reader number (0, 1, 2...)

Returns:

Connection to the reader

---

close

```
public void close()
           throws java.io.IOException
```

Flush and close the input and output streams.

Throws:

`java.io.IOException` - If there is an error closing or flushing the input or output streams.

---

com.sun.javacard.apduio

Class CadTransportException

```
java.lang.Object
  └java.lang.Throwable
      └java.lang.Exception
          └com.sun.javacard.apduio.CadTransportException
```

All Implemented Interfaces:

ct Known Subclasses:

T1Exception, TLP224Exception

`public abstract class` **`CadTransportException`**

extends java.lang.Exception

This abstract class is the parent class of all CAD exception classes. It serves as an ancestor for T1Exception (for T=1 CAD) as well as TLP225Exception (for T=0 CAD) classes.

See Also:

Field Summary

protected static java.util.ResourceBundle	**_messages** Properties file with messages

protected int	**status** Error code
static int	**STATUS_UNKNOWN** Error code constant

Constructor Summary

CadTransportException(int status) Constructor method for exception class.

Method Summary

java.lang.String	**getMessage**() Returns a String describing this exception.
int	**getStatus**() Returns the status associated with the exception.

Methods inherited from class java.lang.Throwable
fillInStackTrace, getCause, getLocalizedMessage, getStackTrace, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Field Detail

_messages

protected static java.util.ResourceBundle **_messages**

Properties file with messages

---

status

`protected int **status**`

Error code

STATUS_UNKNOWN

`public static int **STATUS_UNKNOWN**`

Error code constant

Constructor Detail

CadTransportException

`public **CadTransportException**(int status)`

Constructor method for exception class.

Parameters:
`status` - Initialization status for this exception.

Method Detail

getStatus

`public int **getStatus**()`

Returns the status associated with the exception.

Returns:
The error status

---

getMessage

`public java.lang.String` **`getMessage`**`()`

Returns a String describing this exception.

Overrides:

`getMessage` in class `java.lang.Throwable`

Returns:

the String describing this exception

---

com.sun.javacard.apduio

Class T1Exception

```
java.lang.Object
  └java.lang.Throwable
      └java.lang.Exception
          └com.sun.javacard.apduio.CadTransportException
              └com.sun.javacard.apduio.T1Exception
```

All Implemented Interfaces:

`public class` **`T1Exception`**

extends CadTransportException

This class represents error states for T=1 CAD devices.

See Also:

Field Summary

static int	**T1_BAD_CHECKSUM**
static int	**T1_BLOCK_IO_ERROR**
static int	**T1_FUNCTION_NOT_SUPPORTED**

static int	**T1_PROTOCOL_VIOLATION**
static int	**T1_UNINITIALIZED**
static int	**T1_WRONG_LENGTH**

Fields inherited from class com.sun.javacard.apduio.CadTransportException
_messages, status, STATUS_UNKNOWN

Constructor Summary

T1Exception(int status) Constructor method for this exception class.

Method Summary

java.lang.String	**toString**() Returns a String describing this exception.

Methods inherited from class com.sun.javacard.apduio.CadTransportException
getMessage, getStatus

Methods inherited from class java.lang.Throwable
fillInStackTrace, getCause, getLocalizedMessage, getStackTrace, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Field Detail

T1_WRONG_LENGTH

```
public static final int T1_WRONG_LENGTH
```

See Also:
Constant Field Values

T1_UNINITIALIZED

```
public static final int T1_UNINITIALIZED
```

See Also:
Constant Field Values

T1_BLOCK_IO_ERROR

```
public static final int T1_BLOCK_IO_ERROR
```

See Also:
Constant Field Values

T1_FUNCTION_NOT_SUPPORTED

```
public static final int T1_FUNCTION_NOT_SUPPORTED
```

See Also:
Constant Field Values

T1_PROTOCOL_VIOLATION

`public static final int` **`T1_PROTOCOL_VIOLATION`**

See Also:
Constant Field Values

T1_BAD_CHECKSUM

`public static final int` **`T1_BAD_CHECKSUM`**

See Also:
Constant Field Values

Constructor Detail

T1Exception

`public` **`T1Exception`**`(int status)`

Constructor method for this exception class.

Parameters:
`status` - Initialization status for this exception.

Method Detail

toString

`public java.lang.String` **`toString`**`()`

Returns a String describing this exception.

Overrides:
`toString` in class `java.lang.Throwable`

---

com.sun.javacard.apduio

Class TLP224Exception

```
java.lang.Object
  └java.lang.Throwable
      └java.lang.Exception
          └com.sun.javacard.apduio.CadTransportException
              └com.sun.javacard.apduio.TLP224Exception
```

All Implemented Interfaces:

---

public class **TLP224Exception**

extends CadTransportException

Exceptions related to TLP224 protocol violations

See Also:

Field Summary

Fields inherited from class com.sun.javacard.apduio.CadTransportException
_messages, status, STATUS_UNKNOWN

Constructor Summary

TLP224Exception(int status) Constructor	

Method Summary

java.lang.String	**toString**() Returns a String describing this exception.

Methods inherited from class com.sun.javacard.apduio.CadTransportException
getMessage, getStatus

Methods inherited from class java.lang.Throwable
fillInStackTrace, getCause, getLocalizedMessage, getStackTrace, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Detail

TLP224Exception

public **TLP224Exception**(int status)

Constructor

Parameters:

status - status code

Method Detail

toString

`public java.lang.String` **`toString`**`()`

Returns a String describing this exception.

Overrides:

`toString` in class `java.lang.Throwable`

Returns:

String describing this exception.

Package com.sun.javacard.apduio

Provides a client API for exchanging APDUs with smart cards.

See:

Description

Interface Summary	
CadClientInterface	An interface describing a client connection to a card.

Class Summary	
Apdu	This class represent a pair of C-APDU and R-APDU.
CadDevice	Factory class producing CardClientInterface instances (connections to a card)

Exception Summary	
CadTransportException	This abstract class is the parent class of all CAD exception classes.
T1Exception	This class represents error states for T=1 CAD devices.
TLP224Exception	Exceptions related to TLP224 protocol violations

Package com.sun.javacard.apduio Description

Provides a client API for exchanging APDUs with smart cards.

It is used internally by several components of the Java Card development kit.

Hierarchy For Package com.sun.javacard.apduio

Class Hierarchy

- java.lang.Object
  - com.sun.javacard.apduio.**Apdu**
  - com.sun.javacard.apduio.**CadDevice**
  - java.lang.Throwable
    - java.lang.Exception
      - com.sun.javacard.apduio.**CadTransportException**
        - com.sun.javacard.apduio.**T1Exception**
        - com.sun.javacard.apduio.**TLP224Exception**

Interface Hierarchy

- com.sun.javacard.apduio.**CadClientInterface**

Deprecated API

Contents

Package Class Tree Deprecated Index Help
PREV NEXT FRAMES NO FRAMES All Classes

A

Apdu - Class in com.sun.javacard.apduio
This class represent a pair of C-APDU and R-APDU.
Apdu() - Constructor for class com.sun.javacard.apduio.Apdu
Creates a new instance of Apdu.
APDU_TYPE_MASK - Static variable in class com.sun.javacard.apduio.Apdu
Mask to extract APDU type information: either ISO ecoding or not.

C

CadClientInterface - Interface in com.sun.javacard.apduio
An interface describing a client connection to a card.
CadDevice - Class in com.sun.javacard.apduio
Factory class producing CardClientInterface instances (connections to a card)
CadDevice(byte, InputStream, OutputStream) - Constructor for class com.sun.javacard.apduio.CadDevice
this form of constructor is used by T=0, T=1 implementations
CadDevice(byte) - Constructor for class com.sun.javacard.apduio.CadDevice
this form of constructor is used by PCSC streams are not applicable
CadTransportException - Exception in com.sun.javacard.apduio
This abstract class is the parent class of all CAD exception classes.
CadTransportException(int) - Constructor for exception com.sun.javacard.apduio.CadTransportException
Constructor method for exception class.
CASE_1 - Static variable in class com.sun.javacard.apduio.Apdu
The value which indicates this Apdu is an ISO 7816-3 Case 1 apdu
CASE_2E - Static variable in class com.sun.javacard.apduio.Apdu
The value which indicates this Apdu is an ISO 7816-3 Case 2 Extended apdu
CASE_2S - Static variable in class com.sun.javacard.apduio.Apdu
The value which indicates this Apdu is an ISO 7816-3 Case 2 Short apdu
CASE_3E - Static variable in class com.sun.javacard.apduio.Apdu

The value which indicates this Apdu is an ISO 7816-3 Case 3 Extended apdu
CASE_3S - Static variable in class com.sun.javacard.apduio.Apdu
The value which indicates this Apdu is an ISO 7816-3 Case 3 Short apdu
CASE_4E - Static variable in class com.sun.javacard.apduio.Apdu
The value which indicates this Apdu is an ISO 7816-3 Case 4 Extended apdu
CASE_4S - Static variable in class com.sun.javacard.apduio.Apdu
The value which indicates this Apdu is an ISO 7816-3 Case 4 Short apdu
CLA - Static variable in class com.sun.javacard.apduio.Apdu
The offset in the command array to the ISO 7816-3 CLA octet.
close() - Method in class com.sun.javacard.apduio.CadDevice
Flush and close the input and output streams.
com.sun.javacard.apduio - package com.sun.javacard.apduio
Provides a client API for exchanging APDUs with smart cards.
command - Variable in class com.sun.javacard.apduio.Apdu
Internal representation of the C-APDU header

D

dataIn - Variable in class com.sun.javacard.apduio.Apdu
Data part of the C-APDU
dataOut - Variable in class com.sun.javacard.apduio.Apdu
Data part of the R-APDU
debug - Variable in class com.sun.javacard.apduio.CadDevice
Internal debug flag.
deviceProtocol - Variable in class com.sun.javacard.apduio.CadDevice
The currentry used protocol

E

exchangeApdu(Apdu) - Method in interface com.sun.javacard.apduio.CadClientInterface
Exchanges com.sun.javacard.apduio.Apdu with a card.

G

getCadClientInstance(byte, InputStream, OutputStream) - Static method in class com.sun.javacard.apduio.CadDevice
A factory method
getCase() - Method in class com.sun.javacard.apduio.Apdu
Return the ISO 7816-3 Case of this Apdu
getChannelInfo() - Method in class com.sun.javacard.apduio.Apdu
Determines the channel number
getCommand() - Method in class com.sun.javacard.apduio.Apdu
Get internal representation of the header of the C-APDU
getCommandApduBytes() - Method in class com.sun.javacard.apduio.Apdu
returns the command APDU as a byte array.
getDataIn() - Method in class com.sun.javacard.apduio.Apdu
Get the data part of the C-APDU
getDataOut() - Method in class com.sun.javacard.apduio.Apdu
Get the data part of the R-APDU
getDeviceProtocol() - Method in class com.sun.javacard.apduio.CadDevice
Get the value of the protocol byte
getLc() - Method in class com.sun.javacard.apduio.Apdu
Get Lc
getLe() - Method in class com.sun.javacard.apduio.Apdu
Get Le
getMessage() - Method in exception com.sun.javacard.apduio.CadTransportException
Returns a String describing this exception.
getPCSCClientInstance(int) - Static method in class com.sun.javacard.apduio.CadDevice
A factory method for connecting to a PCSC card readed
getResponseApduBytes() - Method in class com.sun.javacard.apduio.Apdu
returns the response APDU as a byte array.
getStatus() - Method in class com.sun.javacard.apduio.Apdu
Get status (SW bytes) as integer
getStatus() - Method in exception com.sun.javacard.apduio.CadTransportException
Returns the status associated with the exception.
getSw1Sw2() - Method in class com.sun.javacard.apduio.Apdu
Get status bytes

I

in - Variable in class com.sun.javacard.apduio.CadDevice
The input stream which data is read from.

INS - Static variable in class com.sun.javacard.apduio.Apdu
The offset in the command array to the ISO 7816-3 INS octet.
isExtended - Variable in class com.sun.javacard.apduio.Apdu
Extended APDU flag

L

Lc - Variable in class com.sun.javacard.apduio.Apdu
Value of Lc
Le - Variable in class com.sun.javacard.apduio.Apdu
Value of Le
LOGICAL_CHN_MASK - Static variable in class com.sun.javacard.apduio.Apdu
Mask to extract channel information out of the CLA byte.

O

out - Variable in class com.sun.javacard.apduio.CadDevice
The output stream to which data is written.

P

P1 - Static variable in class com.sun.javacard.apduio.Apdu
The offset in the command array to the ISO 7816-3 P1 octet.
P2 - Static variable in class com.sun.javacard.apduio.Apdu
The offset in the command array to the ISO 7816-3 P2 octet.
P3 - Static variable in class com.sun.javacard.apduio.Apdu
The offset in the command array to the ISO 7816-3 P3 octet.
powerDown(boolean) - Method in interface com.sun.javacard.apduio.CadClientInterface
Send powerdown to a card
powerDown() - Method in interface com.sun.javacard.apduio.CadClientInterface
Powerdown a card or simulator and then disconnect
powerUp() - Method in interface com.sun.javacard.apduio.CadClientInterface
Send powerup to a card or a simulator

PROTOCOL_PCSC - Static variable in class com.sun.javacard.apduio.CadDevice
PC/SC constant
PROTOCOL_T0 - Static variable in class com.sun.javacard.apduio.CadDevice
Protocol constant
PROTOCOL_T1 - Static variable in class com.sun.javacard.apduio.CadDevice
Protocol constant
PROTOCOL_TCL - Static variable in class com.sun.javacard.apduio.CadDevice
Protocol constant

S

setDataIn(byte[]) - Method in class com.sun.javacard.apduio.Apdu
Set the data of the C-APDU
setDataIn(byte[], int) - Method in class com.sun.javacard.apduio.Apdu
Set the data of the C-APDU
setDataOut(byte[]) - Method in class com.sun.javacard.apduio.Apdu
Set the data part of the R-APDU
setDataOut(byte[], int) - Method in class com.sun.javacard.apduio.Apdu
Set the data part of the R-APDU
setLc(int) - Method in class com.sun.javacard.apduio.Apdu
Set the value of Lc
setLe(int) - Method in class com.sun.javacard.apduio.Apdu
Set Le
status - Variable in exception com.sun.javacard.apduio.CadTransportException
Error code
STATUS_UNKNOWN - Static variable in exception com.sun.javacard.apduio.CadTransportException
Error code constant
sw1sw2 - Variable in class com.sun.javacard.apduio.Apdu
Status as byte array

T

T1_BAD_CHECKSUM - Static variable in exception com.sun.javacard.apduio.T1Exception

T1_BLOCK_IO_ERROR - Static variable in exception com.sun.javacard.apduio.T1Exception

T1_FUNCTION_NOT_SUPPORTED - Static variable in exception com.sun.javacard.apduio.T1Exception

T1_PROTOCOL_VIOLATION - Static variable in exception com.sun.javacard.apduio.T1Exception

T1_UNINITIALIZED - Static variable in exception com.sun.javacard.apduio.T1Exception

T1_WRONG_LENGTH - Static variable in exception com.sun.javacard.apduio.T1Exception

T1Exception - Exception in com.sun.javacard.apduio
This class represents error states for T=1 CAD devices.
T1Exception(int) - Constructor for exception com.sun.javacard.apduio.T1Exception
Constructor method for this exception class.
TLP224Exception - Exception in com.sun.javacard.apduio
Exceptions related to TLP224 protocol violations
TLP224Exception(int) - Constructor for exception com.sun.javacard.apduio.TLP224Exception
Constructor
toString() - Method in class com.sun.javacard.apduio.Apdu
Generate a string representation of this Apdu in the ApduTool output format.
toString() - Method in exception com.sun.javacard.apduio.T1Exception
Returns a String describing this exception.
toString() - Method in exception com.sun.javacard.apduio.TLP224Exception
Returns a String describing this exception.

_

_messages - Static variable in class com.sun.javacard.apduio.CadDevice
Properties file with messages
_messages - Static variable in exception com.sun.javacard.apduio.CadTransportException
Properties file with messages

How This API Document Is Organized

This API (Application Programming Interface) document has pages corresponding to the items in the navigation bar, described as follows.

Package

Each package has a page that contains a list of its classes and interfaces, with a summary for each. This page can contain four categories:

- Interfaces (italic)
- Classes
- Enums
- Exceptions
- Errors
- Annotation Types

Class/Interface

Each class, interface, nested class and nested interface has its own separate page. Each of these pages has three sections consisting of a class/interface description, summary tables, and detailed member descriptions:

- Class inheritance diagram
- Direct Subclasses
- All Known Subinterfaces
- All Known Implementing Classes
- Class/interface declaration
- Class/interface description

- Nested Class Summary
- Field Summary
- Constructor Summary
- Method Summary

- Field Detail

- Constructor Detail
- Method Detail

Each summary entry contains the first sentence from the detailed description for that item. The summary entries are alphabetical, while the detailed descriptions are in the order they appear in the source code. This preserves the logical groupings established by the programmer.

Annotation Type

Each annotation type has its own separate page with the following sections:

- Annotation Type declaration
- Annotation Type description
- Required Element Summary
- Optional Element Summary
- Element Detail

Enum

Each enum has its own separate page with the following sections:

- Enum declaration
- Enum description
- Enum Constant Summary
- Enum Constant Detail

Tree (Class Hierarchy)

There is a Class Hierarchy page for all packages, plus a hierarchy for each package. Each hierarchy page contains a list of classes and a list of interfaces. The classes are organized by inheritance structure starting with `java.lang.Object`. The interfaces do not inherit from `java.lang.Object`.

- When viewing the Overview page, clicking on "Tree" displays the hierarchy for all packages.
- When viewing a particular package, class or interface page, clicking "Tree" displays the hierarchy for only that package.

Deprecated API

The Deprecated API page lists all of the API that have been deprecated. A deprecated API is not recommended for use, generally due to improvements, and a replacement API is usually given. Deprecated APIs may be removed in future implementations.

Index

The Index contains an alphabetic list of all classes, interfaces, constructors, methods, and fields.

Prev/Next

These links take you to the next or previous class, interface, package, or related page.

Frames/No Frames

These links show and hide the HTML frames. All pages are available with or without frames.

Constant Field Values

The Constant Field Values page lists the static final fields and their values.

This help file applies to API documentation generated using the standard doclet.

All Classes

Apdu
CadClientInterface
CadDevice
CadTransportException
T1Exception
TLP224Exception

Constant Field Values

Contents

- com.sun.*

com.sun.*

com.sun.javacard.apduio.Apdu		
public static final int	APDU_TYPE_MASK	240
public static final int	CASE_1	1
public static final int	CASE_2E	5
public static final int	CASE_2S	2
public static final int	CASE_3E	6
public static final int	CASE_3S	3
public static final int	CASE_4E	7
public static final int	CASE_4S	4
public static final int	CLA	0
public static final int	INS	1
public static final int	LOGICAL_CHN_MASK	3
public static final int	P1	2
public static final int	P2	3
public static final int	P3	4

com.sun.javacard.apduio.CadDevice

public static final byte	PROTOCOL_PCSC	-85
public static final byte	PROTOCOL_T0	0
public static final byte	PROTOCOL_T1	1
public static final byte	PROTOCOL_TCL	91

com.sun.javacard.apduio.T1Exception		
public static final int	T1_BAD_CHECKSUM	-123
public static final int	T1_BLOCK_IO_ERROR	-126
public static final int	T1_FUNCTION_NOT_SUPPORTED	-125
public static final int	T1_PROTOCOL_VIOLATION	-124
public static final int	T1_UNINITIALIZED	-127
public static final int	T1_WRONG_LENGTH	-128

Hierarchy For All Packages

Package Hierarchies:

com.sun.javacard.apduio

Class Hierarchy

- java.lang.Object
  - com.sun.javacard.apduio.**Apdu**
  - com.sun.javacard.apduio.**CadDevice**
  - java.lang.Throwable
    - java.lang.Exception
      - com.sun.javacard.apduio.**CadTransportException**
        - com.sun.javacard.apduio.**T1Exception**
        - com.sun.javacard.apduio.**TLP224Exception**

Interface Hierarchy

- com.sun.javacard.apduio.**CadClientInterface**

Copyright © 2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at http://www.sun.com/patents and one or more additional patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights - Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements. Use is subject to license terms.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and in other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, Java, Solaris, Sparc, Java Card, Java Developer Connection, Javadoc, JDK, JVM, J2ME, NetBeans and J2SE are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Products covered by and information contained in this service manual are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright © 2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, Etats-Unis. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuels relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plus des brevets américains listés à l'adresse http://www.sun.com/patents et un ou les brevets supplémentaires ou les applications de brevet en attente aux Etats - Unis et dans les autres pays. L'utilisation est soumise aux termes de la Licence.

Cette distribution peut comprendre des composants développés par des tierces parties.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, Java, Solaris, Sparc, Java Card, Java Developer Connection, Javadoc, JDK, JVM, J2ME, NetBeans et J2SE sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

Les produits qui font l'objet de ce manuel d'entretien et les informations qu'il contient sont regis par la legislation americaine en matiere de controle des exportations et peuvent etre soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucleaires, des missiles, des armes biologiques et chimiques ou du nucleaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou reexportations vers des pays sous embargo des Etats-Unis, ou vers des entites figurant sur les listes d'exclusion d'exportation americaines, y compris, mais de maniere non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une facon directe ou indirecte, aux exportations des produits ou des services qui sont regi par la legislation americaine en matiere de controle des exportations et la liste de ressortissants specifiquement designes, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.