G24-J Customer Notes

Table of Contents

Software Development Kits

To support application development "Motorola M2M Wireless Modules" set up two software development kits. These development kits are:

- ***MOTO2MOTO Wireless Toolkit 1.0 Powered by Sun Java(TM) Technology***
  - Includes the "Motorola M2M API" javadoc version 3.6.2.
  - Supports emulation of the following JSRs: JSR 139 (CLDC 1.1), JSR 118 (MIDP 2.0), JSR 120 (WMA 1.1), JSR 205 (WMA 2.0), JSR 135 (MMAPI 1.1), JSR 75 (FC, PIM).
  - Supports emulation of "Motorola M2M API".

- ***Motorola G24 Java(TM) ME SDK Tool v3.0***
  - Includes the "Motorola M2M API" javadoc version 3.6.2.
  - Supports emulation of the following JSRs: JSR 139 (CLDC 1.1), JSR 118 (MIDP 2.0), JSR 120 (WMA 1.1), JSR 205 (WMA 2.0), JSR 135 (MMAPI 1.1), JSR 75 (FC, PIM).
  - Does not support emulation of the "Motorola M2M API"; only stub functions and default return values are supported within the emulation.

Product Known Issues

Software Version: G24_G_0C.11.95R

Flex Version: GCEG24xJ240000AB072

Default MIDlet Version: 2.2.2

General Note: For any issue listed below, a fix may be already available. Contact M2MCare@motorola.com to receive an engineering release with the fix.

- Identifier: LIBmm58373
  After a socket connection open failure, the next socket connection open fails for no reason.
  Workaround: A second open attempt will resolve this problem. Add SocketConnectionOpen class (see below) to your code and use SocketConnectionOpen.open(url) instead of Connector.open(url) to implement this workaround.

- Identifier: LIBmm56842
  Default MIDlet SMS updates / requests problem. Cannot disable a fault report address. An empty fault report address is considered as invalid.
  Workaround: Send a dummy fault report address instead of an empty string, for example: "sms://".

- Identifier: LIBmm55919
  JTool manager is stuck if JTool command is entered while JAR file is being downloaded.
  Workaround: Pay attention and avoid this kind of operation. In case JTool is stuck switch to MIDlet manger and back to JTool; if this fails, restart the unit.

- Identifier: LIBmm41311
  Rare messages are routed to UART2 although logs are disabled or routed to another serial.

- Identifier: LIBmm31469
  While in JTool mode, arriving browser message may overrun current settings of provisioning web session. Note that java unit on the field is not likely to be in JTool mode.

- Identifier: LIBmm31094
  OTAdownload.startOTA(...) changes the current web session and doesn't restore the original once done. User should take care of restoring the current web session.

- Identifier: LIBmm30254
  Incoming cell broadcast message (CBS) contains corrupted data**.**

- Identifier: LIBll52655
  ”JTool Ready” output depends on the speed of switching KJAVA Manager GPIO pin.

Workaround: Minimum of 1-2 seconds between switches ensures good operation.

- Identifier: LIBmm24591
  For some operators, GPRS username and password settings might not work.
  Workaround: Do not set password and username. All the operators support the APN parameter alone as well. Address M2M customer support.

- Identifier: LIBmm38860
  No proper failure reason is outputted when a MIDlet download fails because an unsigned MIDlet is trying to replace a signed MIDlet.
  Workaround: When a MIDlet download fails for no apparent reason, delete the existing MIDlet first using the JTool command: "oemconfig_delete:um".

- Identifier: LIBmm28588
  KDWP problems:
  1. When switching to JTool mode, the MIDlet does not stop.
     Workaround: After each KDWP session restart the unit.
  2. In some cases, after power cycle, the MIDlet becomes invalid.
     Workaround: Use JTool command "oemconfig_clearflag:um" to reset the MIDlet invalidity flag.

  See KDWP document for more troubleshooting.

- Identifier: LIBmm11531
  Validation of A2D threshold configuration ignores the case where the low threshold upper limit equals the high threshold lower limit. In this case the configuration has no meaning, but no exception would be thrown.

- Identifier: LIBmm49336
  Server socket connection object cannot be used twice. After the client side closes the connection, a connection cannot be reestablished using the same object.
  Workaround: When read method returns -1 (EOF), the server socket connection must be closed. New server socket connection should be created in order to reestablish a listen socket.

- Identifier: LIBmm22677
  Few types of SMS messages cannot be received using Java. For example: Fax or Voicemail notification, concatenated non-port (regular) messages.
  Workaround: For concatenated messages, send messages with port.

- Identifier: LIBmm76793
  FileConnection API cannot create a new directory anywhere in the file system.

- Identifier: LIBmm76797
  Contact entries form SIM card can be read using PIM API only if the SIM was inserted at power up.

- Identifier: LIBmm76808
  Java Virtual Machine enters suspended state if there is an incoming call while the VM loads. This scenario is very rare.

- Identifier: LIBmm76812
  Using WebSessionManager.getWebSession(index) with SESSION_INDEX_PROV (provisioning related web session) might return a web session object with invalid fields. Meaning after processing a Browser Message, the returned web session object might not load as is to a different web session.
  Workaround: Use correctWebSession(websession) method (see below) to reset the invalid fields to their default values.

LIBmm58373 Workaround

File: **SocketConnectionOpen.java**

```
import java.io.IOException;
import javax.microedition.io.Connection;
import javax.microedition.io.Connector;

/**
 * Wrapper class for the method that overcomes the socket connection openning bug (identifier: LIBmm58373).
 */
public class SocketConnectionOpen {

    /**
     * Create and open a Connection.
     * <p>
     * This method overcomes the socket connection openning bug (identifier: LIBmm58373).
     * To overcome the bug it tries to open the connection twice, while enabling timeout exceptions,
     * and sleeping between the tries.
     * </p>
     * @param name The URL for the connection.
     * @return A new Connection object.
     * @throws IllegalArgumentException - If a parameter is invalid.
     * @throws ConnectionNotFoundException - If the target of the name cannot be found, or if the requested protocol type is not supported.
     * @throws IOException - If some other kind of I/O error occurs.
     * @throws SecurityException - May be thrown if access to the protocol handler is prohibited.
     */
    public static Connection open(String name) throws IOException {

        // Number of times to try open the connection.
        // Two is enough to overcome the bug.
        final int numberOfTries = 2;

        int i = 0;

        IOException ioe = null;

        while(i++ < numberOfTries) {

            try {
                return Connector.open(name, Connector.READ_WRITE, true);
            } catch(IOException e) {
                ioe = e;
            }

            try {
                Thread.sleep(10000);
            } catch(InterruptedException ie) {
                ie.printStackTrace();
            }

        }

        throw ioe;

    }

}
```

LIBmm76812 Workaround

```
/**
 * Reset the web session invalid fields to their default values.
 * @param websession Web session to correct.
 */
private static void correctWebSession(WebSession websession) {

   if(websession.getServiceType1() != WebSession.SERVICE_TYPE_HTTP &&
      websession.getServiceType1() != WebSession.SERVICE_TYPE_HTTP_SECURE &&
      websession.getServiceType1() != WebSession.SERVICE_TYPE_WAP &&
      websession.getServiceType1() != WebSession.SERVICE_TYPE_WAP_CONNECTIONLESS &&
      websession.getServiceType1() != WebSession.SERVICE_TYPE_WAP_SECURE &&
      websession.getServiceType1() != WebSession.SERVICE_TYPE_WAP_SECURE_CONNECTIONLESS) {

      websession.setServiceType1(WebSession.SERVICE_TYPE_HTTP);

   }

   if(websession.getPort1() == 0) {
      websession.setPort1(8080);
   }

   if(websession.getServiceType2() != WebSession.SERVICE_TYPE_HTTP &&
      websession.getServiceType2() != WebSession.SERVICE_TYPE_HTTP_SECURE &&
      websession.getServiceType2() != WebSession.SERVICE_TYPE_WAP &&
      websession.getServiceType2() != WebSession.SERVICE_TYPE_WAP_CONNECTIONLESS &&
      websession.getServiceType2() != WebSession.SERVICE_TYPE_WAP_SECURE &&
      websession.getServiceType2() != WebSession.SERVICE_TYPE_WAP_SECURE_CONNECTIONLESS) {

      websession.setServiceType2(WebSession.SERVICE_TYPE_HTTP);

   }

   if(websession.getPort2() == 0) {
      websession.setPort2(9201);
   }

   if(websession.getTimeOut() != WebSession.TIMEOUT_1_MIN &&
      websession.getTimeOut() != WebSession.TIMEOUT_2_MIN &&
      websession.getTimeOut() != WebSession.TIMEOUT_5_MIN &&
      websession.getTimeOut() != WebSession.TIMEOUT_10_MIN &&
      websession.getTimeOut() != WebSession.TIMEOUT_15_MIN) {

      websession.setTimeOut(WebSession.TIMEOUT_15_MIN);

   }

   if(websession.getCsdSpeed1() != WebSession.CSD_SPEED_2400_BAUD &&
      websession.getCsdSpeed1() != WebSession.CSD_SPEED_4800_BAUD &&
      websession.getCsdSpeed1() != WebSession.CSD_SPEED_9600_BAUD &&
      websession.getCsdSpeed1() != WebSession.CSD_SPEED_14400_BAUD) {

      websession.setCsdSpeed1(WebSession.CSD_SPEED_14400_BAUD);

   }

   if(websession.getLineType1() != WebSession.CSD_LINE_TYPE_MODEM &&
      websession.getLineType1() != WebSession.CSD_LINE_TYPE_ISDN) {

     websession.setLineType1(WebSession.CSD_LINE_TYPE_MODEM);

   }
```

```
    if(websession.getCsdSpeed2() != WebSession.CSD_SPEED_2400_BAUD &&
        websession.getCsdSpeed2() != WebSession.CSD_SPEED_4800_BAUD &&
        websession.getCsdSpeed2() != WebSession.CSD_SPEED_9600_BAUD &&
        websession.getCsdSpeed2() != WebSession.CSD_SPEED_14400_BAUD) {

        websession.setCsdSpeed2(WebSession.CSD_SPEED_14400_BAUD);

    }

    if(websession.getLineType2() != WebSession.CSD_LINE_TYPE_MODEM &&
        websession.getLineType2() != WebSession.CSD_LINE_TYPE_ISDN) {

       websession.setLineType2(WebSession.CSD_LINE_TYPE_MODEM);

    }

}
```

M2M WTK Known Issues

Software Version: 1.0 final

See "Known Issues and Bugs" chapter at MOTO2MOTO Wireless Toolkit Powered by Sun Java™ Technology → Documentation → Release Notes.

Below is a list of additional known issues:

ID	Feature	Description
73	Browse to some location to open a project	If we browse to some location to open a project the following error message is displayed: Error loading Comm property java.io.FileNotFoundException…
74	Call class	Call.release() does not throw exception if there is no call to release
72	Create new project	File -> Create project from JAD/JAR does not create the project
75	Network class	Method getNITZClock() - wrong exception message
76	NetworkException class	Displays the message: NetworkException was thrown with error code [101] errorMessage = [Network: Unable to register confirm event]
67	Osc class	Emulator: after the alarm expired power up, the Power State of the External Event Generator display "Power ON" while also the System Status is "Power ON".
68	Osc class	Emulator: After MIDlet Watchdog expire, The Power State of the External Event Generator display "Power OFF" while also the System Status is "Power OFF".
69	Osc class	Emulator: there is no power up after the swReset() power down.
70	Osc class	Emulator: When Pressing on "Power OFF", the Midlet stop running, but destroyApp is not called.
71	Osc class	Emulator: After MIDlet Watchdog expire power down, there is no power up.
77	Osc class	OSC.setDateTime(dateTime), OSC.setAlarm(dateTimeAlarm); If dateTime or dateTimeAlarm parameters are out of rage an exception must be thrown.
78	Osc class	OSC.enableAlarm(false), OSC.enableAlarm(true) If DateTime is not available an exception should be thrown.
79	Osc class	No date & time change events

81	Osc class	OSC.enableLowPowerMode(true);
		OSC.enableLowPowerMode(true,1);
		OSC.enableRFOperations(false);
		OSC.enableRFOperations(true);
		OSC.isRFOperationsEnabled();
		OSC.enableDateTimeNetworkUpdate()
		Trying to call these functions trows exceptions

82	Osc class	OSC.getProperty("stam"); Who sends the message "The key was not found" ?
83	Osc class	Emulator: when Clock Supported is True, the onDateTimeStateChanged set the time correctly in the first time that the midlet run, but after power cycle (power off - power on), the time & date change to 14-AUG-2009 10:00:00.
N/A	Osc class	setProperty() values are not validated.