

2007-11-01

JSR 248 Mobile Service Architecture – Change Log

#	Change Description	Reason	Spec version	Status
1	<p>Section: 6.3.3.2 Mandatory ContactList Attributes</p> <p>ATTR_NONE has been removed from the list of mandatory attributes and from the table in the requirement text. The text changed to clarify that method PIMlist.getSupportedAttributes() should not return ATTR_NONE in the attribute array even though that meta-attribute is implicitly supported.</p>	<p>The current specification lists some ContactList attributes that must be supported. The problem is that the list includes a special meta-attribute that is meant to signify the absence of attributes. For that reason it should not be on the list of attributes to avoid confusion.</p>	1.1.0	PROPOSED
2	<p>7.3.2.1 Mapping Permissions to Function Groups in Protected Domains</p> <p>Section headings added before tables 3 – 9 as none of them belong under the previous section heading: “Network Access Requirements”.</p>	<p>Tables 3 – 9 appear under section titled “Network Access Requirements”. The problem is that the tables are not about networking. Some subsection headings are missing. The tables also appear in the document without any explanatory text around them.</p>	1.1.0	PROPOSED
3	<p>7.3.2.1 Mapping Permissions to Function Groups in Protected Domains</p> <p>Clarified the MSA requirement on Product Token “UNTRUSTED/1.0” to avoid confusion and to comply with the HTTP specification.</p> <p>The new wording: “HttpConnection and HttpsConnection MUST include the Product-Token “UNTRUSTED/1.0” in the User-Agent header. Product-Tokens in the User-Agent header supplied by the application MUST NOT be deleted.”</p>	<p>The specification is not clear on how Product-Token “UNTRUSTED/1.0” should be included in the User-Agent header.</p> <p>There seems to be a conflict between how it was defined in JTWI and MSA.</p>	1.1.0	PROPOSED
4	<p>Updated references to new component JSR versions</p> <p>Since maintenance releases of JSRs 180, 226 and 234 include all MSA clarifications in MSA 1.0, these clarifications are removed</p>	<p>The following MSA component JSRs have made maintenance releases after MSA 1.0 was finalized:</p> <ul style="list-style-type: none"> * JSR 135: Mobile Media API 1.1 → 1.2 * JSR 226: Scalable 2D Vector 	1.1.0	PROPOSED

2007-11-01

	<p>from the MSA specification.</p> <p>System property table in Appendix A section A.1.1 is also updated accordingly.</p> <p>Due to updated SATSA specification, microedition.satsa.*.version system property values are also updated to 1.0.1 from 1.0.</p>	<p>Graphics API 1.0 → 1.1</p> <p>* JSR 177: Security and Trust Services API 1.0 → 1.0.1</p> <p>* JSR 180: SIP API 1.0.1 → 1.1.0</p> <p>* JSR 234: AMMS 1.0 → 1.1</p> <p>These include new important clarifications and also incorporate many of the clarifications in the MSA 1.0 specification.</p> <p>New component JSR versions should be used to reduce fragmentation and redundancy between MSA and the component JSR specifications.</p>		
5	<p>JSR 184 requirement 6.10.3.1</p> <p>(1) Wording changed from "If the load method is used for loading content from a resource that is accessed by using a networking protocol such as HTTP, this method has the same requirements for security checks as the corresponding API for using the networking protocol directly (when using, for example, javax.microedition.io.HttpConnection)." to "If the load method is used for loading content from a resource that is accessed by using a networking protocol such as HTTP, this method MUST perform the same security checks as the corresponding API for using the networking protocol directly (when using, for example, javax.microedition.io.HttpConnection)." (2) Font for method name "load" corrected.</p>	<p>JSR 184 requirement 6.10.3.1 is not stated in typical requirement format (with a MUST or MAY).</p> <p>Also, the text, "If the load method is used..." should have the word "load" in a different font (since this represents a method).</p>	1.1.0	PROPOSED
6	<p>JSR 205 requirement 6.11.3.4</p> <p>Changed requirement text from: "In MessagePart objects, the encoding parameter maps to the character set indicated by the charset= parameter in the</p>	<p>The text in "requirement text" contains no MUST or MAY statements. The requirements should be written using standard requirement text conventions.</p>	1.1.0	PROPOSED

2007-11-01

	<p>Content-Type header. The encoding parameter in the MessagePart object does not affect the choice of Content-Transfer-Encoding, which is chosen automatically by the implementation."</p> <p>to</p> <p>"The encoding parameter in MessagePart objects MUST be mapped to the character set indicated by the charset= parameter in the Content-Type header. The encoding parameter in the MessagePart object does not affect the choice of Content-Transfer-Encoding, which is chosen automatically by the implementation."</p>			
7	<p>JSR 205 requirement 6.11.3.1</p> <p>Section title changed to: "Message Handling and Buffering"</p>	<p>Title of JSR 205 requirement 6.11.3.1 is "Handling of Messages if the Associated Java Application Isn't Running" is incorrect in the original specification.</p>	1.1.0	PROPOSED
8	<p>6.2.3.9 RMS Data Size per MIDlet Suite - Behavior for larger than 64kB RMS sizes</p> <p>Change 1: Old text: "If a MIDlet from the suite creates a single (empty) RecordStore, the RecordStore.getSizeAvailable() method returns a value greater than or equal to the value of the MIDlet-Data-Size attribute." New text: "If a MIDlet creates a single (empty) RecordStore, the RecordStore.getSizeAvailable() method returns a value greater than or equal to the value of the MIDlet-Data-Size attribute."</p> <p>Change 2: Old text: "At any time, a MIDlet from the suite can store a single byte array of at least RecordStore.getSizeAvailable() bytes using the</p>	<p>The specification text includes two requirements: 1."A compliant implementation MUST be able to honour requests by MIDlet suites for an RMS data size of at least 64 kB." 2."...if an implementation honours a request for a certain size of RMS data, this amount of memory MUST be available to MIDlets from the suite during the entire time the suite is installed on the device." The problem is that the three bullet points following text "More precisely, a compliant implementation MUST ensure the following:" include some unclarities in their wording that may lead to an incorrect assumption that requirement number 2 above would not apply to all RMS sizes granted to the application. The misleading sentences are: "* If a MIDlet from the suite creates a single (empty)</p>	1.1.0	PROPOSED

2007-11-01

	<p>RecordStore.addRecord() method. In particular, a single byte array size of the value of the MIDlet-Data-Size attribute can be stored in an empty record store." New text: "At any time, a MIDlet can store a single byte array of at least RecordStore.getSizeAvailable() bytes using the RecordStore.addRecord() method. In particular, a single byte array size of the value of the MIDlet-Data-Size attribute can be stored in an empty record store."</p>	<p>RecordStore, the RecordStore.getSizeAvailable() method returns a value greater than or equal to the value of the MIDlet-Data-Size attribute. * At any time, a MIDlet from the suite can store a single byte array of at least RecordStore.getSizeAvailable() bytes using the RecordStore.addRecord() method. In particular, a single byte array size of the value of the MIDlet-Data-Size attribute can be stored in an empty record store."</p> <p>It can be easily misunderstood that these bullet points refer to the bullet point right before them that talks about the specific case of a MIDlet suite requesting 64kB or less. However, these points describe the more general requirement that is not tied to the 64kB limit.</p>		
9	<p>6.1.3.1 Minimum Heap Size - Redundant requirement</p> <p>The clarification in Section 6.1.3.1 to be removed. Section heading to be left as it is. This will prevent latter section numbers from changing.</p>	<p>Section "6.1.3.1 Minimum Heap Size" requires that 1024 kB must be available for the Java VM. Section "7.2.3 Java Heap size Available to MIDlet" sets a stricter and fully overlapping requirement. It requires 1024kB to be available for the MIDlet. The requirement in section 6.1.3.1 seems to be fully covered by the latter requirement. In addition the first requirement is untestable unlike the latter requirement.</p>	1.1.0	PROPOSED
10	<p>6.1.3.1 Minimum Heap size - Reference to a nonexistent chapter in the document changed to reference the correct chapter</p>	<p>The text refers to a chapter called "Hardware Requirements and Recommendations". The real name of the chapter is "7.2 Hardware Requirements".</p>	1.1.0	PROPOSED
11	<p>American English spelling applied throughout the document.</p>	<p>Both British English and American English used in the document text.</p>	1.1.0	PROPOSED
12	<p>6.3.2.13 Unescaping of URLs Before Security Checks</p>	<p>The section title is slightly misleading. The requirement text itself describes how URLs must</p>	1.1.0	PROPOSED

2007-11-01

	Section title to be changed to: "6.3.2.13 Unescaping of URLs Presented to the User".	be unescaped before presenting them to the user. "Security check" is a much wider concept and can mean many other things than a security prompt.		
13	6.11.3.7 System property for Wireless Messaging API version Incorrect version used in the example corrected to the right one (2.0).	The table in the section explains the system property like this: "Version of the Wireless Messaging API supported by the device. For example, 1.0". The problem is that 1.0 is not an allowed value for the property in an MSA compliant implementation.	1.1.0	PROPOSED
14	7.3 Security Requirements Removed the redundant statement: "The security requirements are normative, and therefore all compliant devices MUST adhere to the requirements in this section."	The whole section 7 is normative as indicated in the section title. It is also stated in section 3.1. Repeating that statement right after discussing the recommended security policy seems confusing and unnecessary as the normativeness has already been stated elsewhere.	1.1.0	PROPOSED
15	6.3.3.3 Mandatory ContactList Fields Specification clarified to require that at least one of the contact lists supports all of the listed fields.	The specification can be incorrectly interpreted to require the following fields to be supported in all ContactLists (including the list on SIM/USIM): ADDR, EMAIL, NOTE, TEL, URL, PHOTO It is not clear if the intent has really been to mandate these to SIM based contact lists as well or if it is enough if these are supported in the main contact list in the device memory. Requirement unclear and too strict for SIM cards.	1.1.0	PROPOSED
16	Section "2.1 Normative References" updated. References added to the component JSR specifications. Unused references removed.	The list of references in "2.1 Normative References" needs to be updated, when the specification is changed to reference new maintenance releases of some of the component JSRs. Another problem is that the list of references includes 42 documents, but only 4 of them are actually referred to in the specification. The reason is, that most of the references are missing from the specification	1.1.0	PROPOSED

2007-11-01

		text and some references have no normative relevance to this specification and should be removed, for example, [SCPROV] and [WIM].		
17	List of abbreviations in section "3.2 Abbreviations" updated.	The list of abbreviations is missing some abbreviations used in the document, for example: SIM and USIM. It also includes abbreviations that are not used in the document.	1.1.0	PROPOSED
18	Summary Table – System Properties Changed fileconn.dir.roots.name to fileconn.dir.roots.names in Appendix A.	The informative Appendix A that lists all system properties has a typo in one system property name. The property fileconn.dir.roots.names has been incorrectly listed in the appendix as fileconn.dir.roots.name. The TCK tests the correct name from the normative part of the JSR 248 document (fileconn.dir.roots.names).	1.1.0	PROPOSED
19	Removal of JSR 229 from the MSA specification This includes: - Removing the reference from section "2.1 Normative References". - Removing JSR 229 from the picture in section "5. MSA Component JSRs" - Removing JSR 229 from the table in section "5. MSA Component JSRs" - Removing section "6.14 Payment API (JSR 229)" and all its subsections. This should be done so that other section numbers do not change. - Removing system property "microedition.payment.version" from table in section A.1.1 of Appendix A. - Removing Table 11 from the security section.	JSR 229 TCK cannot currently be licensed and for those who already have a license, do not get support for the TCK, so the TCK cannot be passed. This creates a problem as JSR 229 is a mandatory component for a full JSR 248 implementation.	1.1.0	PROPOSED
20	Adding a missing method to the Location API permission table In Table 7, method name "LocationProvider.getLastKnownLocation()" to be added to the row	The Location API version used by MSA is 1.0.1. However, the permission table is still from Location API 1.0. One missing method name that was added in the Location API maintenance	1.1.0	PROPOSED

2007-11-01

	with permission "javax.microedition.location.Location".	release needs to be also added to the MSA specification.		
21	<p>Permission table 4 (Bluetooth) aligned with the JSR 82 specification.</p> <p>Change: "Connector.open("irdaobex://addr...") Connector.open("irdaobex://conn...") Connector.open("irdaobex://name...")" added after: "Connector.open("irdaobex://discover...")"</p>	Permission table 4 in the MSA specification has left out a few method names that are in the corresponding table in the Bluetooth API specification (JSR 82). These methods should be added to avoid confusion caused by an unnecessary difference compared to JSR 82.	1.1.0	PROPOSED
22	Method "Tuner.setPresetName(int preset, java.lang.String name)" added below method "Tuner.setPreset(int preset, int freq, java.lang.String mod, int stereomode)" in Permission table 12 under security requirements.	Permission table 12 in the MSA specification has left out one method name that is in the corresponding table in the AMMS API specification (JSR 234). This missing method should be added.	1.1.0	PROPOSED
23	<p>Summary Table – System Properties:</p> <p>"JSR 248" removed from the row containing the system property microedition.profiles</p>	The specification table A.1.1 incorrectly states that JSR 248 defines the system property microedition.profiles.	1.1.0	PROPOSED
24	Typo correction: "MIDlet-PermissionsOpt" to be replaced with "MIDlet-Permissions-Opt" in the specification	There is a typo in one instance of the attribute name MIDlet-Permissions-Opt. The incorrect form is "MIDlet-PermissionsOpt".	1.1.0	PROPOSED
25	An informative table listing all Function Groups with permissions that belong to those Function Groups added under "Appendix A. Summary Tables".	It is difficult to quickly find out which permissions each Function Group contains. An informative summary table with this information should be added.	1.1.0	PROPOSED
26	Other minor editorial corrections, such as fixing typos.	The specification text included some minor typos.	1.1.0	PROPOSED