3GPP TSG-T (Terminals) Meeting #17
Biarritz, France, 4 - 6 September, 2002

Tdoc TP-020202

Agenda Item: 5.2.3

Source: T2

Title: "MExE" Change Requests

Document for: Approval

Spec	CR	Rev	Rel	Subject	Cat	Vers-Current	Vers-New	T2 doc	Workitem
23.057	119	-	Rel-6	CC/PP section cleanup	D	6.0.0	6.1.0	T2-020652	MEXE6-ENHANC
23.057	120	-	Rel-6	Adding new attributes to the JAR manifest file	C	6.0.0	6.1.0	T2-020660	MEXE6-ENHANC
23.057	121	-	Rel-5	Adding new attributes to the JAR manifest file	C	5.0.0	5.1.0	T2-020663	MEXE5-ENHANC

3GPP TSG-T2 #18
Velen, Germany
12 -16 August 2002

T2-020652

CR-Form-v7

CHANGE REQUEST

⌘ **23.057** **CR** **119** ⌘ **rev** **-** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ [ ] ME [X] Radio Access Network [ ] Core Network [ ]

Title:	⌘	CC/PP section cleanup
Source:	⌘	T2
Work item code:	⌘	MEXE6-ENHANC
Date:	⌘	08/07/2002
Category:	⌘	**D**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	To move the sections into the right context and avoid sections appearing twice in the same specification.
Summary of change:	⌘	Removed section 5.2.3 and moved section 5.2.7 to 10.2.5 where it belongs
Consequences if not approved:	⌘	Specification is difficult to read and understand.

Clauses affected:	⌘	5.2.3 and 5.2.7

		Y	N		
Other specs affected:	⌘		X	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	

Other comments:	⌘	

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

5.2 Capability and content negotiation

...

5.2.1 User profile and capability negotiation relationship

...

5.2.2 Capability negotiation characteristics

...

~~5.2.3 CC/PP over HTTP or WSP (Classmark 4)~~

~~In Classmark 4 the CC/PP is carried over by using CC/PP over HTTP [15] and optionally CC/PP over WSP, [17].~~

5.2.4 Client content capability report

The client may perform content negotiation capabilities to the server by using appropriate HTTP/1.1 or WSP request headers. The following methods are available for content negotiation:

- Client software (product): `User-Agent` header;
- MIME media types: `Accept` header;
- Character set: `Accept-Charset` header;
- Content encoding: `Accept-Encoding` header;
- Language: `Accept-Language` header.

There is no need for MExE to specify any tokens for content negotiation, as these headers are already defined in HTTP and WSP. The formats for these headers are specified in [9] and [6].

5.2.5 Server role in capability negotiation

The server may request the capabilities of a client whenever required, and shall enquire of the client's capabilities prior to making each transaction resulting in a set of transfers to the client; the characteristics which may be reported in the client capability report are identified in the list above.

In server-driven negotiation the server signals to the client that the response entity was selected from a set of available representation.

5.2.6 Client-driven negotiation

If the server cannot specify an optimal version for the client basing on the CC/PP sent over to the server, the server may also indicate to client which type of versions are available and let the client make the decision. This method is already available in HTTP1.1. In client-driven negotiation the client selects the best representation after having received an initial response from the server.

~~5.2.7 CC/PP over WSP (Classmark 1)~~

~~In Classmark 1, according to the WAP User Agent Profile Specification [17], the CC/PP description is encoded with WBXML [45] after which it is carried over by WSP, [17].~~

…

7.2.4 CC/PP over WSP (Classmark 1)

In Classmark 1, according to the WAP User Agent Profile Specification [17], the CC/PP description is encoded with WBXML [45] after which it is carried over by WSP, [17].

…

10.2.5 CC/PP over HTTP or WSP (Classmark 4)

In Classmark 4 the CC/PP is carried over by using CC/PP over HTTP [15] and optionally CC/PP over WSP, [17].

3GPP TSG-T2 #18
Velen, Germany
12 -16 August 2002

T2-020660

CR-Form-v7

CHANGE REQUEST

⌘ **23.057** **CR** **120** ⌘ **rev** **-** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ [ ] ME [X] Radio Access Network [ ] Core Network [ ]

Title:	⌘	Adding new attributes to the JAR manifest file
Source:	⌘	T2
Work item code:	⌘	MEXE6-ENHANC
Date:	⌘	01/08/2002
Category:	⌘	**C**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Adding more attributes to the manifest file in the jar file format, to update the handset with a set of non root certificates this way.
Summary of change:	⌘	Added the attributes
Consequences if not approved:	⌘	There will be proprietary ways of doing this in different handsets from different manufacturers, which all will have to be supported by the operators.

Clauses affected:	⌘	2, 6.8.1

		Y	N			
Other specs affected:	⌘		X	Other core specifications	⌘	
			X	Test specifications		
			X	O&M Specifications		

Other comments:	⌘	

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

2 References

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] Void.

[2] 3GPP TS 22.057: "Mobile Execution Environment (MExE); Stage 1".

[3] Personal Java 1.1.1 or higher, Sun Microsystems http://www.javasoft.com/products/personaljava/

[4] JavaPhone API version 1.0, http://java.sun.com/products/javaphone/ .

[5] Void.

…

[49] PKCS#1 “RSA Cryptographic Standard” " version 2.0, RSA Laboratories, October 1998
URL: http://www.rsasecurity.com/rsalabs/pkcs/pkcs-1/index.html

[50] Common Language Infrastructure, ECMA specification ECMA-335,
http://www.ecma.ch/ecma1/STAND/ecma-335.htm

[51] Simple Object Access Protocol version 1.1, (SOAP), URL : http://www.w3.org/TR/2000/NOTE-SOAP-20000508/

[52] PKCS#7 “Cryptographic Message Syntax Standard” " version 1.5, RSA Laboratories, November 1993
URL: http://www.rsasecurity.com/rsalabs/pkcs/pkcs-7/index.html

6.8 Usage of Signed Content

6.8.1 Signed packages used for installation

If the 3 MExE security domains defined in clause 6.1 "Generic security" are not supported, then the signed packages used for installation, described in this clause, are~~is~~ optional.

The Java Archive (JAR) file format shall be supported on classmark 2 and 3 MExE devices for securely packaging objects that are to be downloaded and installed on the ME. The method for securely packaging objects for MExE classmark 1 devices may be referenced from the WAP specifications in a future release of this specification. A MExE device may support other proprietary means of downloading and installing objects.

The JAR file shall contain a manifest file that has at least the following attribute:

`MExE-Implementation-Type`
The information contained within the manifest file is represented as so-called "name: value" pairs, where "name" is represented by `MExE-Implementation-Type`. Groups of name-value pairs are known as a "section", where sections are separated from other sections by empty lines.

The `MExE-Implementation-Type` value shall be one of the following:-

- "**MExENativeLibrary**"

  in the case of a MExE Native Library (as described in 8.3.2 "Installing MExE native libraries");

- "**TTPCertificate**"

  in the case of a certificate containing a 3rd party root public key (as described in 6.8.2 "Installation of root certificates in a signed data package");

- "**ManufacturerCertificate**"

  in the case of a certificate containing a manufacturer root public key (as described in 6.8.2 "Installation of root certificates in a signed data package");

- "**OperatorCertificate**"

  in the case of a certificate containing an operator root public key (as described in clause 6.8.2 "Installation of root certificates in a signed data package");

- "**AdminCertificate**"

  in the case of an administrator certificate, which shall consist of a section containing both the administrator certificate and a CCM (as described in clause 6.8.2 "Installation of root certificates in a signed data package"); or

- "**OrdinaryTTPCertificate**"

  in the case of a certificate or certificate list containing 3rd party public key(s). An example of a certificate list syntax can be found in [52]

- "**OrdinaryManufacturerCertificate**"

  in the case of a certificate or certificate list containing manufacturer public key(s). An example of a certificate list syntax can be found in [52]

- "**OrdinaryOperatorCertificate**"

  in the case of a certificate or certificate list containing operator public key(s). An example of a certificate list syntax can be found in [52]

- "**CCM**"

  in the case of a CCM (as described in clause 6.8.2 "Installation of root certificates in a signed data package"); or

- *-free-format-value-*

  in the case of proprietary binaries or Java classes such as native DSP code, provisioned functionality upgrades and patches (as described in clause 6.8.3 "Installation of other signed data").

Refer to [42] for full details of how to encode the "name: value" pairs and "section" in a JAR manifest file.

See figure 15 "Signed packages". When a download of a JAR file is completed, the system installer shall read the manifest to determine what types of files are contained in the JAR, and install them appropriately.

Note that a signed package containing a library which does not have a manifest attribute "MExE-`Implementation-Type: MExENativeLibrary"` shall be considered to be some type of upgrade to libraries that are intrinsically part of the MExE device implementation rather than a "MExE native library". E.g.

`MExE-Implementation-Type: ManufacturerUpgrade (something.dll)`

(Recommended behaviour for the server is that it uses the capability information supplied from the MExE device to determine how to offer appropriate upgrades.)

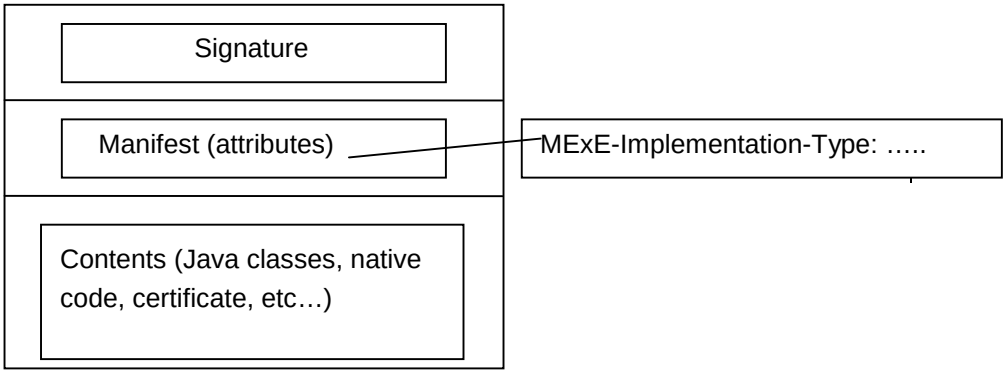

Figure 15: Signed packages

6.8.2 Installation of root certificates in a signed data package

Root certificates in a signed package (whose signature verifies as described in clause 6.6 "Root Public keys" to the Manufacturer root, Operator root, or the Administrator root), may be installed to the root public key store on the MExE device. Note that the certificate thus packaged does not necessarily belong to the manufacturer domain. The types of certificate that can be present and installed by packages are given in table 9 "Allowed certificate types in signed packages". The MExE device shall store the root public key as indicated by the certificate type.

When a certificate containing an Administrator root public key is thus contained in a signed package, the signed package (e.g. a JAR file in the case of Java based MExE classmarks) shall contain two files: the Administrator root public key and the CCM.

Table 9: Allowed certificate types in signed packages

Signature on Package	Allowed Certificate types in package
Administrator	Third Party
Manufacturer	Administrator, Manufacturer, Operator, Third Party
Operator	Administrator, Operator, Third Party

6.8.3 Installation of other signed data

A signed package of proprietary binaries or Java classes such as native DSP code, provisioned functionality upgrades and patches, whose signature verifies as described in clause 6.6.2 "Manufacturer root public key" as belonging to the Manufacturer Domain may be installed. The use of such binaries is outside the scope of MExE, but the manufacturer shall be responsible for ensuring that the integrity of MExE is not compromised.

Support of this feature is optional.

3GPP TSG-T2 #18
Velen, Germany
12 -16 August 2002

T2-020663

CR-Form-v7

CHANGE REQUEST

⌘ **23.057** **CR** **121** ⌘ **rev** **-** ⌘ Current version: **5.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ [ ] ME [X] Radio Access Network [ ] Core Network [ ]

Title:	⌘	Adding new attributes to the JAR manifest file
Source:	⌘	T2
Work item code:	⌘	MEXE5-ENHANC
Date:	⌘	14/08/2002
Category:	⌘	**C**
Release:	⌘	Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Adding more attributes to the manifest file in the jar file format, to update the handset with a set of non root certificates this way.
Summary of change:	⌘	Added the attributes
Consequences if not approved:	⌘	There will be proprietary ways of doing this in different handsets from different manufacturers, which all will have to be supported by the operators.

Clauses affected:	⌘	2, 8.11

		Y	N		
Other specs affected:	⌘		X	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	

Other comments:	⌘	

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

2 References

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] Void.

[2] 3GPP TS 22.057: "Mobile Execution Environment (MExE); Stage 1".

[3] Personal Java 1.1.1 or higher, Sun Microsystems http://www.javasoft.com/products/personaljava/

[4] JavaPhone API version 1.0, http://java.sun.com/products/javaphone/ .

[5] Void.

…

[49] PKCS#1 “RSA Cryptographic Standard” " version 2.0, RSA Laboratories, October 1998
URL: http://www.rsasecurity.com/rsalabs/pkcs/pkcs-1/index.html

[50] Common Language Infrastructure, ECMA specification ECMA-335,
http://www.ecma.ch/ecma1/STAND/ecma-335.htm

[51] Simple Object Access Protocol version 1.1, (SOAP), URL : http://www.w3.org/TR/2000/NOTE-SOAP-20000508/

[52] PKCS#7 “Cryptographic Message Syntax Standard” " version 1.5, RSA Laboratories, November 1993
URL: http://www.rsasecurity.com/rsalabs/pkcs/pkcs-7/index.html

8.11 Signed packages used for installation

If the 3 MExE security domains defined in clause 8.1 "Generic security" are not supported, then the signed packages used for installation, described in this clause, are~~is~~ optional.

The Java Archive (JAR) file format shall be supported on classmark 2 and 3 MExE devices for securely packaging objects that are to be downloaded and installed on the ME. The method for securely packaging objects for MExE classmark 1 devices may be referenced from the WAP specifications in a future release of this specification. A MExE device may support other proprietary means of downloading and installing objects.

The JAR file shall contain a manifest file that has at least the following attribute:

`MExE-Implementation-Type`
The information contained within the manifest file is represented as so-called "name: value" pairs, where "name" is represented by `MExE-Implementation-Type`. Groups of name-value pairs are known as a "section", where sections are separated from other sections by empty lines.

The `MExE-Implementation-Type` value shall be one of the following:-

- "**`MExENativeLibrary`**"

  in the case of a MExE Native Library (as described in 8.10.1 "Installing MExE native libraries");

- "**TTPCertificate**"

  in the case of a certificate containing a 3rd party root public key (as described in 8.10.2 "Installation of root certificates in a signed data package");

- "**ManufacturerCertificate**"

  in the case of a certificate containing a manufacturer root public key (as described in 8.10.2 "Installation of root certificates in a signed data package");

- "**OperatorCertificate**"

  in the case of a certificate containing an operator root public key (as described in clause 8.10.2 "Installation of root certificates in a signed data package");

- "**AdminCertificate**"

  in the case of an administrator certificate, which shall consist of a section containing both the administrator certificate and a CCM (as described in clause 8.10.2 "Installation of root certificates in a signed data package"); or

- "**OrdinaryTTPCertificate**"

  in the case of a certificate or certificate list containing 3rd party public key(s). An example of a certificate list syntax can be found in [52]

- "**OrdinaryManufacturerCertificate**"

  in the case of a certificate or certificate list containing manufacturer public key(s). An example of a certificate list syntax can be found in [52]

- "**OrdinaryOperatorCertificate**"

  in the case of a certificate or certificate list containing operator public key(s). An example of a certificate list syntax can be found in [52]

- "**CCM**"

  in the case of a CCM (as described in clause 8.10.2 "Installation of root certificates in a signed data package"); or

- *-free-format-value-*

  in the case of proprietary binaries or Java classes such as native DSP code, provisioned functionality upgrades and patches (as described in clause 8.10.3 "Installation of other signed data").

Refer to [42] for full details of how to encode the "name: value" pairs and "section" in a JAR manifest file.