3GPP TSG-CN Meeting #23bis *virtual meeting for email approval of OSA Java CRs* **NP-040154**
29 March - 19 April 2004

Source: **CN5**

Title: **Rel-5 CR 29.198-01: Correct Java Code to conform with Java Rulebook in TS 29.198-01 and to remove errors**

Agenda item: **8.2**

Document for: **APPROVAL**

Meeting-1st-Level	Doc-1st-Level	Spec	CR	R	Phase	Subject	Cat	Vers Curr	Doc-2nd-Level	Worki tem
NP-23bis	NP-040154	29.198-01	028	-	Rel-5	Correct Java Code to conform with Java Rulebook in TS 29.198-01 and to remove errors	F	5.4.0	N5-040226	OSA2

CR-Form-v7

CHANGE REQUEST

⌘ **29.198-01** **CR** **028** ⌘ **rev** **-** ⌘ Current version: **5.4.0** ⌘

*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 ☐ Radio Access Network ☐ Core Network **X**

Title:	⌘	Correct Java Code to conform with Java Rulebook in TS 29.198-01 and to remove errors
Source:	⌘	CN5 (AePONA – Eamonn Murray)
Work item code:	⌘	OSA2
Date:	⌘	22/03/2004
Category:	⌘	**F**
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:	⌘	Java code, and a rulebook for developing it, were introduced into the OSA specifications in September 2003. Recently, errors in code production and misalignment with the published rules were identified in the December 2003 issue of the OSA specifications. Corrections to the code to align with the rulebook and to enable implementation of the December 2003 version of Release 5 are urgently requried.
Summary of change:	⌘	Provide an updated, corrected and validated J2SE Java code package to accompany the specification. This code package is an integral part of the specification. Note that there are no changes required to the specification document itself, other than the reference to the new code package.
Consequences if not approved:	⌘	The existing J2SE code package will be inconsistent with the published Java rulebook. It will be difficult to implement the existing Java code, and will therefore discourage companies from developing implementations which use the Java code part of the OSA specifications.

Clauses affected:	⌘	Annex C.1

Other specs affected:	⌘	Y	N		⌘	
		X		Other core specifications		All parts of TS 29.198 Release 5
			X	Test specifications		
			X	O&M Specifications		

Other comments:	⌘	This CR forms part of a package introducing corrected J2SE code in all parts of TS 29.198 Rel-5.

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.

Annex C (informative):
Java Realisation API

C.1 Java Realisation Overview

The Parlay/OSA UML specifications are defined in a technology neutral manner. This annex aims to deliver for Java, a developer API, provided as a realisation, supporting a Java API that represents the UML specifications.

C.1.1 J2SE API

The J2SE API supports a J2SE development environment that

- provides an abstraction of the Parlay/OSA APIs that provides a local API for J2SE developers
- supports a listener based API for SCFs and a callback API for the Framework
- uses local object references as correlation mechanisms as Java developers are familiar with object correlation
- is a local API without visibility to the underlying transport

C.1.2 J2EE API

The J2EE API supports a development environment which allows the creation of J2EE and Java RMI interfaces for both the server and client, ensuring consistent interfaces for interoperability. These interfaces may be used for Java RMI on either JRMP or IIOP (RMI/IIOP), allowing use in J2EE environments. The interfaces may also be used as a thin layer on other transports, similar to other Java technologies that provide a RMI programming interface.

The J2EE API is a suitable base for Java across Java platforms, allowing creation of implementations that:

- may be a thin layer on transport protocols
- may support J2EE remote interfaces
- may support J2EE local interfaces

The Java files created with the realisation will be made available with the Parlay/OSA specifications.

The remaining sections of this annex deal with the following areas:

- section C.2 covers the tools and languages used to produce and define the Java Realisation
- section C.3 covers the mappings that are common across both Java Realisation APIs
- section C.4 covers the mappings specific to the J2SE API
- section C.5 covers the mappings specific to the J2EE API

C 1.3 Javadoc

The Javadoc that accompanies the J2SE realisation of the Parlay/OSA API specification is provided as archive 291980V5501J2SE.ZIP that accompanies the present document.

The Javadoc that accompanies the J2EE realisation of the Parlay/OSA API specification is provided as archive 2919801J2EE.ZIP that accompanies the present document.