

**CLAIMS**

- 5           1.    A method of reducing the memory footprint of a Java Card application, the method comprising loading (31a, 31b) two distinct packages into the memory of a Java Card, namely a first, main package (11) containing a main application and dedicated to the installation and transaction phase of said main application in the Java Card, and a second package, capable of sharing resources with the main package and intended to be used during execution of the personalization phase of the main application and to be deleted after said execution of said main application.
- 10           2.    The method of Claim 1, wherein the main package and the second package use Java Card Shareable Interface to be capable of sharing resources with each other.
- 15           3.    The method of any one of Claims 1 and 2, wherein the main package is associated to a transaction phase and the second package is associated to a personalization phase, respectively.
4.    The method of Claim 3, wherein the second package is a Card Personalization Specification, CPS, package (21 ) associated to the personalization phase, said CPS package (21 ) containing a CPS applet (25) dedicated to the creation of a CPS object (32) into the memory of the Java Card.
- 20           5.    The method of Claim 4, wherein the CPS package has a firewall context and an ownership, and is configured to execute a personalization code used for the personalization of the main application.
- 25           6.    The method of any one of Claims 4 and 5, further comprising changing (34) the firewall context of the CPS object by assigning thereto a firewall context of the main applet and changing the ownership of the CPS object from the CPS applet instance ownership to a main applet instance ownership.
- 30           7.    The method of any one of Claims 4 to 6, further comprising handling (35) the personalization of the main application by having the main application access the CPS object and execute the personalization code which initializes main application data.

8. The method of any one of Claims 6 and 7, further comprising:
- changing back (39) the ownership of the CPS object from the main applet ownership to the CPS applet ownership; and,
  - deleting (40) the CPS package.

5           9. The method of any one of Claims 6 to 8, wherein the firewall context of the CPS object is changed by a first Application Programming Interface, API.

10           10. The method of any one of Claims 6 to 9, wherein the ownership of the CPS object is changed and/or changed back by the first API, or by a second API distinct from said first API.

10           11. The method of any one of Claims 9 and 10, wherein the first API and/or the second API, used for changing the ownership and/or the firewall context of the CPS object, respectively, are provided by the Java Card platform or are created specifically.

15           12. The method of any one of Claims 10 and 11, wherein the first API and/or the second API used for changing the ownership and the firewall context of the CPS object is/are deleted from the Java Card memory after the change back of the CPS object ownership from the main applet to the CPS applet has been performed.

20           13. The method of any one of Claims 9 to 11, wherein the first API and/or the second API used for changing the ownership and the firewall context of the CPS object is/are rendered unusable after having performed the change back of the CPS object ownership from the main applet to the CPS applet.

25           14. The method of any one of Claims 6 to 13, wherein the firewall context of the CPS object is changed by implementing a specific parameter in the standard install command used to install the CPS applet and defined by the Global Platform Card specification.

          15. The method of any one of Claims 1 to 14, wherein the main application installed from the main package is a Europay Mastercard Visa (EMV) application.

16. The method of any one of Claims 1 to 15, wherein the main package is loaded into the Read-Only Memory (ROM) of the Java Card.

17. The method of any one of Claims 4 to 16, wherein the CPS package is loaded into the Non Volatile Memory (NVM) of the Java Card.

5           18. The method of any one of Claims 1 to 17, further comprising dividing the Java Card application into the main package and the second package.

10           19. A computer program product comprising one or more stored sequences of instructions that are accessible to a processor and which, when executed by the processor, cause the processor to perform the steps of any of Claims 1 to 18.

20. A device for reducing the memory footprint of a Java Card Application, comprising means for carried out all the steps of the method of any one of Claims 1 to 18.