Université
Bordeaux I

La technologie Java Card™

Présentation de la carte à puce

La Java Card

SAUVERON Damien
sauveron@labri.u-bordeaux.fr

Présentation de la carte à puce

Qu'est ce qu'une carte à puce ?

- ***un morceau de plastique*** de la taille d'une carte de crédit
- ***un circuit électronique*** capable de manipuler (stocker, calculer, ...) des informations

Historique

- En 1968, deux Allemands J. Dethloff et H. Grötrupp introduisent un circuit intégré dans une carte plastique.
- En 1970, K. Arimura au Japon dépose un brevet sur la carte à puce.
- Entre 1974 et 1978, ***Roland* Moreno**, ***le père de la carte à puce*** dépose 47 brevets dans 11 pays.
- En 1979, la première carte est créée par Bull CP8 (1Ko de mémoire programmable et coeur à base de microprocesseur 6805).

Historique

- En 1983, premières cartes téléphoniques à mémoire.
- En 1984, adoption de la ”carte bleue” (sur un prototype Bull CP8). version actuelle B0’.
- Entre 1984 et 1987, normalisation sous la ***référence ISO 7816***.
- En 1997, apparition des premières ***Java Cards***.

Les différentes cartes à puce

Il existe plusieurs sortes de cartes à puce.

Plusieurs classements possibles :

- ***les cartes à mémoire*** versus ***les cartes à microprocesseur***
- ***les cartes à contact*** versus ***les cartes sans contact***

La carte à mémoire

- premier modèle de cartes à puce
- la majorité des cartes vendues dans le monde en 1999

Elle possède ***une puce mémoire et une logique cablée non programmable***.

La taille de la mémoire est de 1Ko à 4Ko.

En décembre 2000, annonce de cartes à mémoire optique (4Mo).

La carte à mémoire

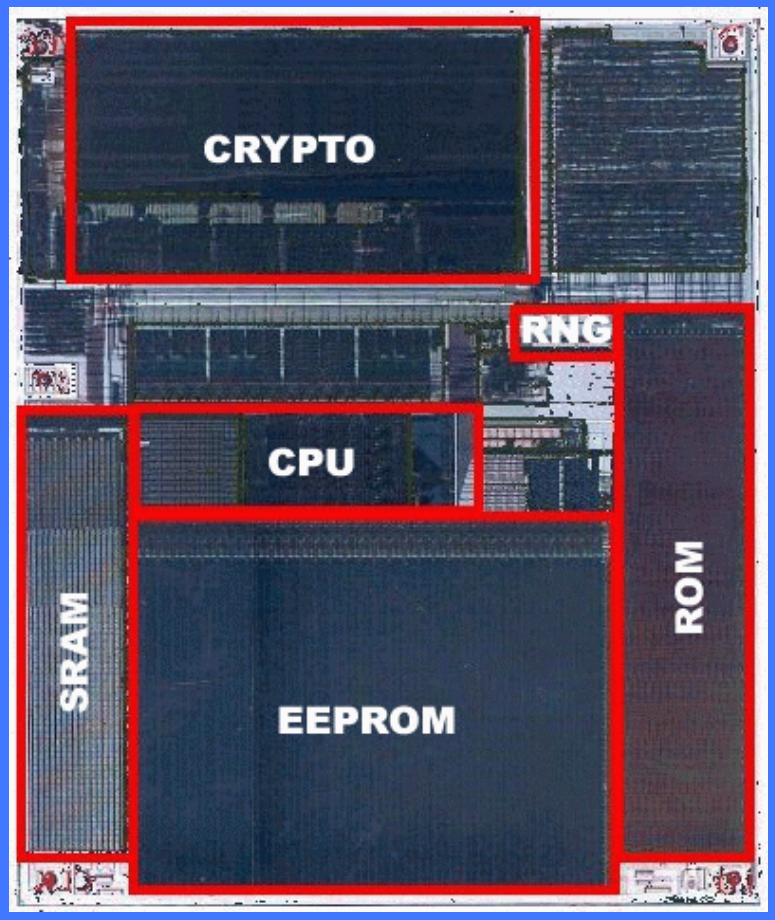
Avantages :

- sa technologie simple
- son faible coût (1$)

Inconvénients :

- sa dépendance vis-à-vis du lecteur de carte
- "assez" facile à dupliquer

La carte à microprocesseur



La carte à microprocesseur

Le microprocesseur :

- 8, 16 ou 32 bits, architecture RISC ou pas
- le plus souvent Motorola 6805 ou Intel 8051 avec une horloge à 5Mhz

La ROM :

- stocke le COS (Card Operating System) et des données permanentes
- figée en usine
- possède une taille de 16Ko à 24Ko

La carte à microprocesseur

L'EEPROM (Electrical Erasable Programmable Read Only Memory) :

- mémoire persistante
- sa taille est de 8Ko à 64Ko
- problèmes : durée de vie limitée et temps d'accès lent

La RAM :

- mémoire de stockage temporaire
- sa capacité est de 256 octets à 1Ko.
- avantages : durée de vie illimitée et temps d'accès rapide

Un co-processeur cryptographique + un générateur de nombre aléatoire RNG (Random Number Generator)

La carte à microprocesseur

Avantages :

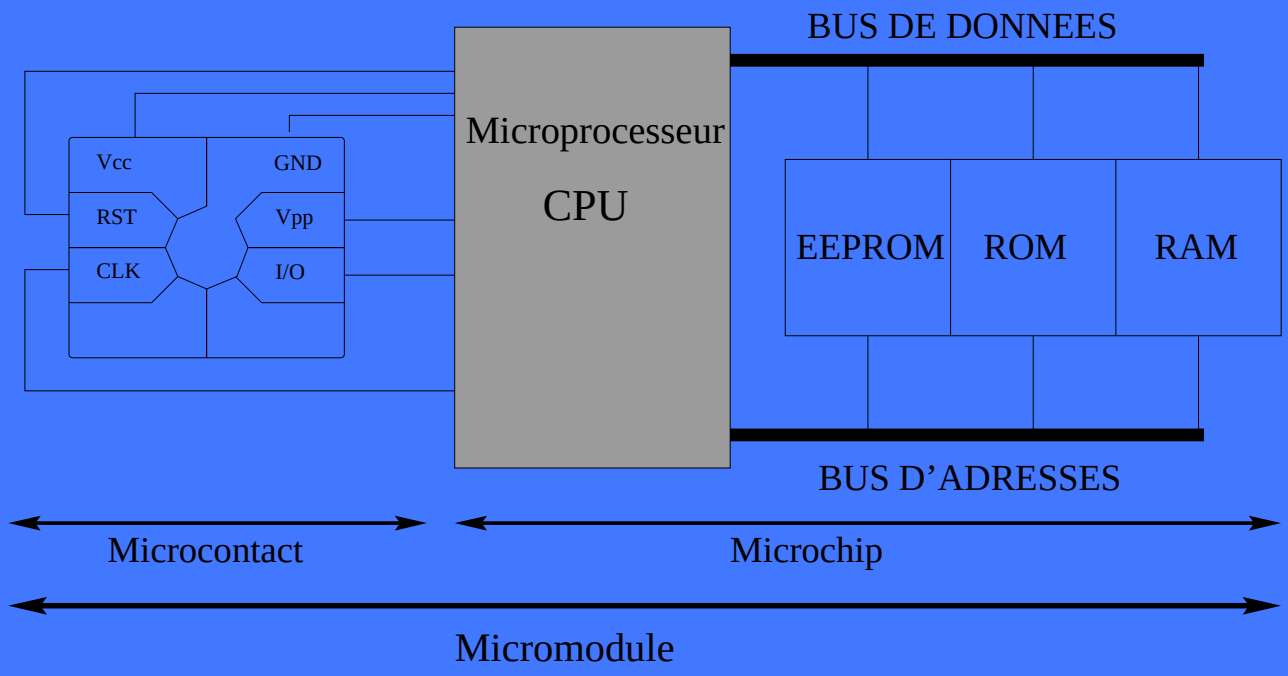
- le microprocesseur et le co-processeur cryptographique fournissent la sécurité
- le coût acceptable pour tant de sécurité (entre 1$ et 20$)

Pour résumer : la triple alliance électronique, informatique et cryptographie assure un très haut degré de sécurité

La carte à contact

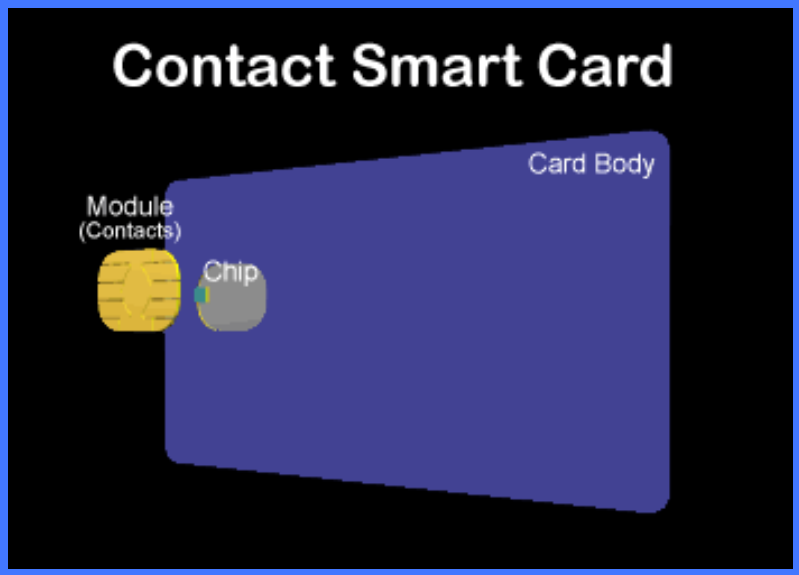
Son fonctionnement, l'oblige à être insérée dans un lecteur de carte.

Elle utilise une communication série via huit contacts.



La carte à contact

Suit le standard ISO 7816.

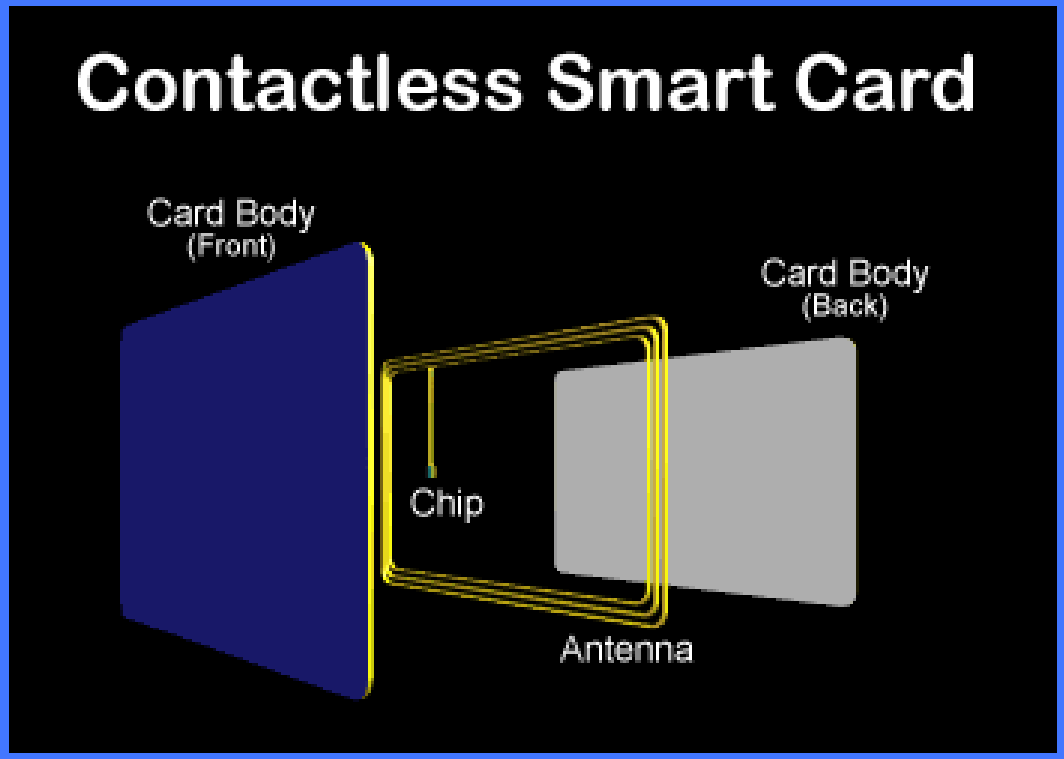


Problèmes :

- l'insertion et le retrait sont des facteurs d'usure de la carte
- il faut bien orienter la carte dans le lecteur.

La carte sans contact

- communique via une antenne dans la carte
- tire son énergie d'un couplage capacitif ou d'un couplage inductif



La carte sans contact

Il n'y a plus les problèmes des cartes à contact.

Problèmes :

- distance de communication limitée (environ 10cm)
- temps de transaction est de l'ordre de 200ms $\Longrightarrow$ limite les données à échanger
- le coût élévé

La carte combi

C'est une combinaison entre :

- la carte à contact
- et la carte sans contact

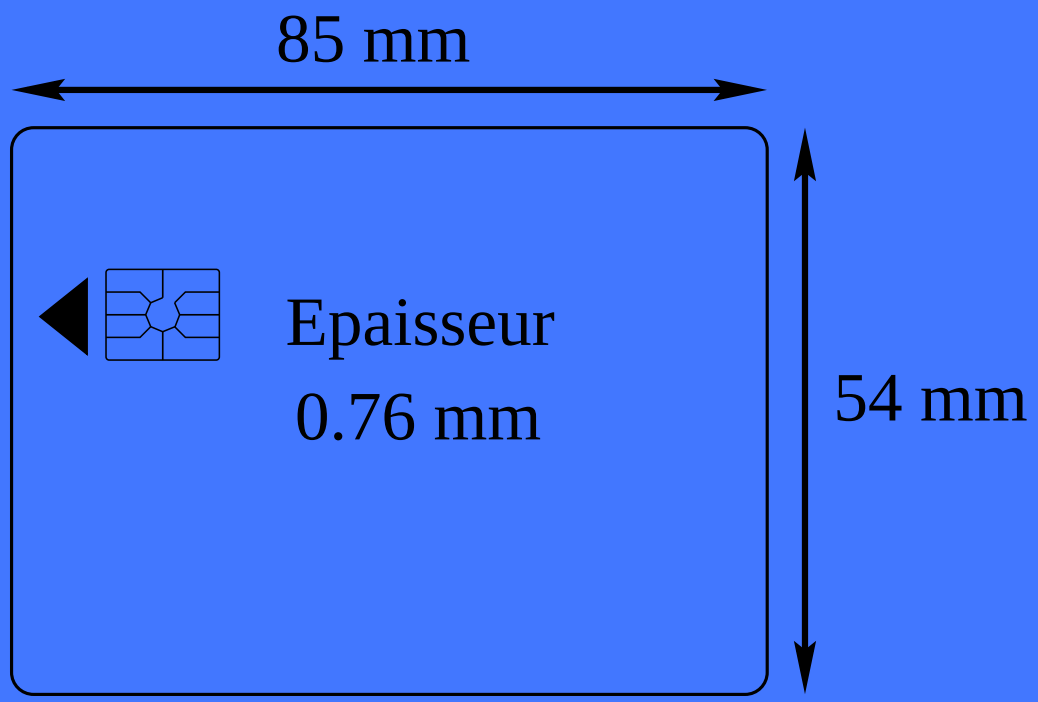
Ces deux possibilités de communication en font une carte "idéale".

Le standard ISO 7816

- Partie 1 : Caractéristiques physiques
- Partie 2 : Dimensions et position des contacts
- Partie 3 : Signaux électroniques et protocoles de transmission
- Partie 4 : Commandes inter-industrie pour l'échange
- Partie 5 : Identificateur d'application
- Partie 6 : Eléments de données inter-industrie
- Partie 7 : Commandes inter-industrie pour SCQL (Structured Card Query Language)
- Partie 8 : Sécurité de l'architecture et des commandes inter-industrie

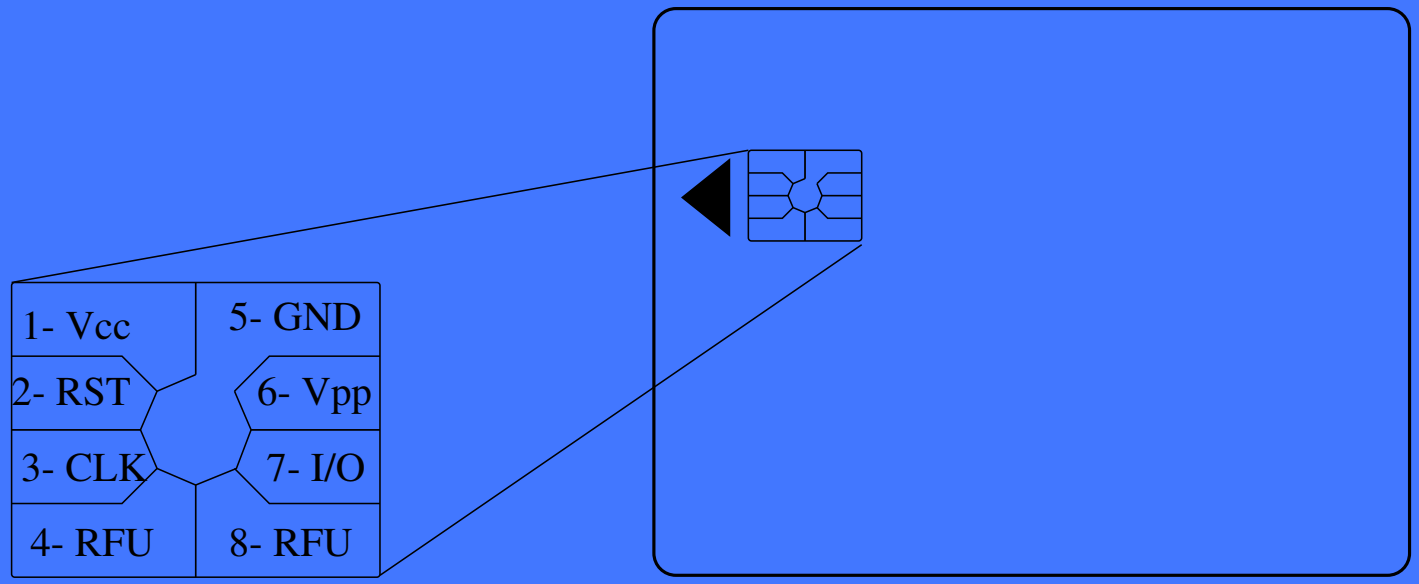
ISO 7816-1

Définit les caractéristiques physiques : dimensions, flexibilité, résistivité.



ISO 7816-2

Définit les aspects électriques et la situation des contacts sur la carte.



ISO 7816-3

Cette partie normalise :

- ***les caractéristiques électriques*** comme :
  - la fréquence d'horloge (entre 1Mhz et 5Mhz)
  - la vitesse des communications (jusqu'a 115200 bauds)
- ***les protocoles de transmission*** (TPDU : Transmission Protocol Data Unit) :
  - T=0, protocole orienté octet
  - T=1, protocole orienté paquet
  - T=14 réservé pour les protocoles propriétaires
- ***la sélection du type de protocole*** (PTS : Protocol Type Selection)
- ***la réponse au reset*** (ATR : Answer To Reset)

ISO 7816-4

ISO 7816-4 vise à assurer une inter-opérabilité.

Il spécifie :

- ☞ ***le contenu des messages*** entre la carte et le lecteur
  - les commandes
  - les réponses
- ☞ ***les structures des fichiers et de données*** :
  - l'accès à ces données
  - l'architecture de sécurité
  - la sécurisation des communications

ISO 7816-4

Le protocole APDU

Protocole de niveau application.

- La ***commande APDU*** (C-APDU) : émise par le CAD vers la carte

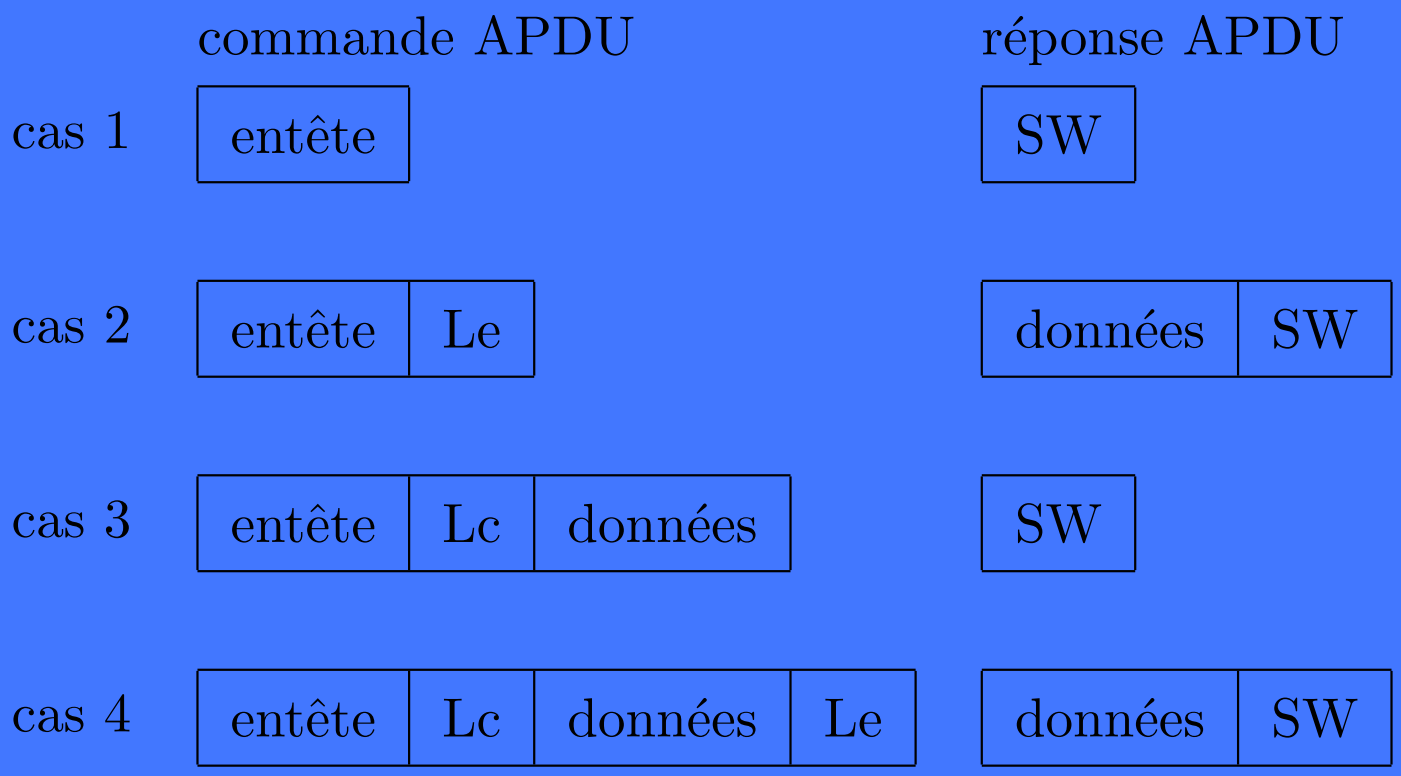
Entête obligatoire				Corps optionnel		
CLA	INS	P1	P2	Lc	Champ de données	Le

- La ***réponse APDU*** (R-APDU) : transite de la carte vers le CAD

Corps optionnel	Enqueue obligatoire	
Champ de données	SW1	SW2

ISO 7816-4

Les différents cas d'échanges APDU :

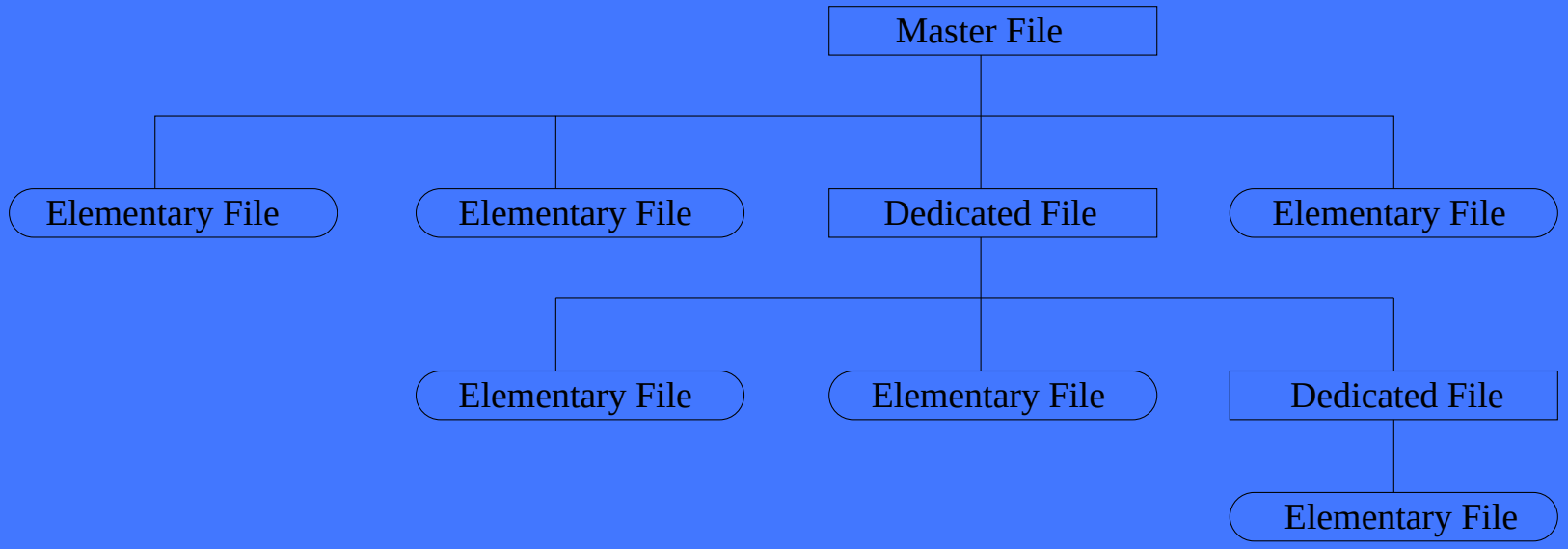


Le système de fichiers des cartes à puce

Supporte un ***système de fichiers hiérarchique*** qui peut contenir trois types de fichiers :

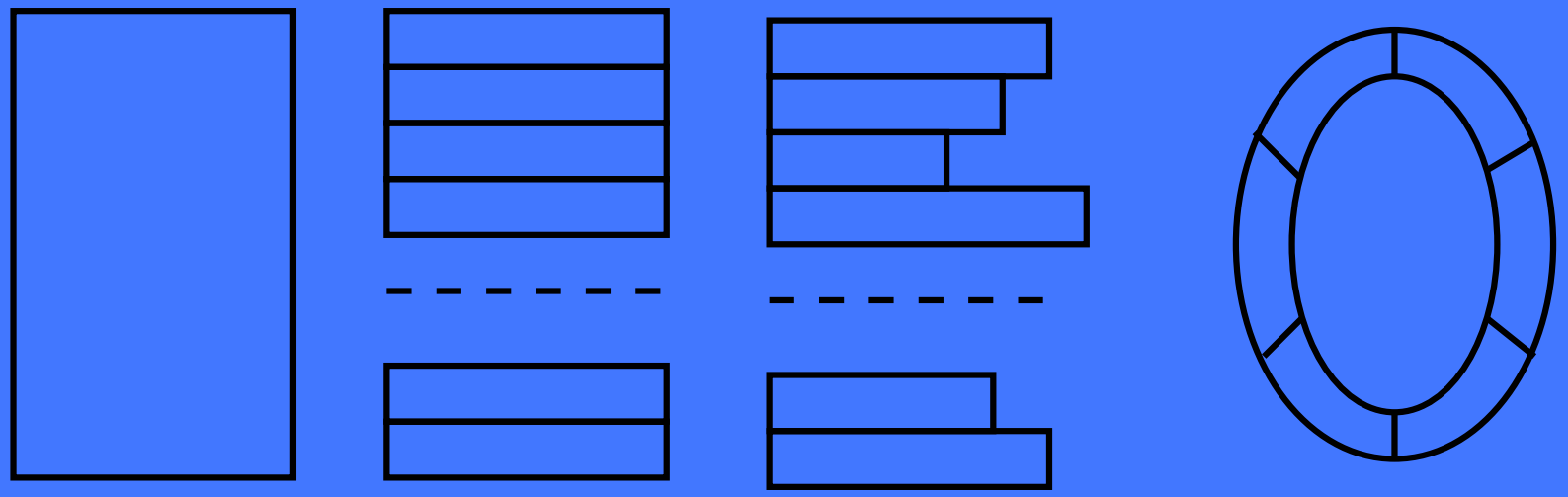
- ”Master File” (MF)
- ”Dedicated File” (DF)
- ”Elementary File” (EF)

ISO 7816-4


Master File
Elementary File
Elementary File
Dedicated File
Elementary File
Elementary File
Elementary File
Dedicated File
Elementary File

ISO 7816-4

Les différents types d'"Elementary File" (EF)



Transparent file | Linear fixed | Linear variable | cyclic fixed

ISO 7816-5

Spécifie le ***système de numérotation*** et les procédures d'enregistrement ***des identifiants d'applications*** (AID pour Application IDentifier).

Un AID = identification unique d'une application de la carte et de certains types de fichiers

Un AID se compose de deux parties :

RID (5 octets)	PIX (0-11 octects)

ISO 7816-6

Spécifie les ***éléments de données inter-industrie*** tels que le nom du porteur de carte, de la date d'expiration, etc.

Etiquette	Longueur	valeur

ISO 7816-7

Propose des ***commandes inter-industrie pour SCQL***.

ISO 7816-8

Concerne la ***sécurité de l'architecture et des commandes inter-industrie.***

Pourquoi utiliser la carte à puce ?

Avantages :

- la sécurité
- la portabilité
- la facilité d'utilisation
- la personnalisation

Pourquoi utiliser la carte à puce ?

La sécurité

- ***au niveau physique*** : techniques d'impression sophistiquées
- ***au niveau hardware*** :
  - un numéro de série unique
  - l'utilisation de mémoire de type PROM
  - blindage physique du composant
  - des détecteurs de conditions anormales (température,...)
  - brouillage des informations dans le composant
  - co-processeurs cryptographiques

Pourquoi utiliser la carte à puce ?

- ***au niveau software*** :
  - contrôles d'accès aux données
  - maintien de l'intégrité des données
  - entrées/sorties sécurisées
- ***au niveau de l'environnement de production*** qui doit être physiquement sécurisé

Pourquoi utiliser la carte à puce ?

La portabilité :

- taille de la carte à puce
- informations disponibles à l'endroit où se trouve le porteur de carte

La facilité d'utilisation :

- insérer sa carte dans le lecteur puis la retirer
- naissance des Java Cards une technologie simple et facile d'accès $\Longrightarrow$ naissance de cartes multi-applicatives. Un concurrent : la WinCard proposée par Microsoft.

La personnalisation : objet marketing avec la personnalisation graphique ou l'embossage.

Applications

- l'industrie des télécommunications
- l'industrie bancaire et monétaire
- le secteur de la santé
- l'industrie audiovisuelle avec la télévision à péage, ...
- le porte-monnaie électronique
- les transports en commun.
- le contrôle d'accès physique de personnes à des locaux, ...
- l'identification : à des sites sur l'Internet, ...
- les applications de fidélité
- les "e-services"

La Java Card

Qu'est ce que Java Card ?

Technologie permettant de faire fonctionner des applications écrites en langage Java pour :

- carte à puce
- autres périphériques à mémoire limitée

La technologie Java Card définit une plateforme sécurisée pour cartes à puce, portable et multi-application qui incorpore beaucoup des avantages du langage Java.

Historique

- En Novembre 1996, un groupe d'ingénieurs de Schlumberger cherche à simplifier la programmation des cartes à puces tout en préservant la sécurité.
  $\Longrightarrow$ la spécification Java Card 1.0
- En Février 1997, Bull et Gemplus se joignent à Schlumberger pour cofonder le ***"Java Card Forum"***.
- En Novembre 1997, Sun présente les spécifications Java Card 2.0.

Historique

- En Mars 1999 sort la version 2.1 des spécifications Java Card.
  Elles consistent en trois spécifications :
  - The Java Card 2.1 API Specification.
  - The Java Card 2.1 Runtime Environment Specification.
  - The Java Card 2.1 Virtual Machine Specification.

  Contribution la plus significative :
  - Définition explicite de la machine virtuelle de la Java Card.
  - Le format de chargement des applets.
- En Mai 2000, sort une petite correction $\Longrightarrow$ version 2.1.1
- En Octobre 2000, plus de 40 entreprises ont acquis la licence d'exploitation de la technologie Java Card.

Les avantages de la technologie Java Card

La facilité de développement des applications grâce :

- à la programmation orientée objet offerte par Java
- à l'utilisation des environnements de développement existants pour Java.
- à une plateforme ouverte qui définit des APIs et un environnement d'exécution standard.
- à l'encapsulation de la complexité fondamentale du système des cartes à puce.

Les avantages de la technologie Java Card

La sécurité grâce :

- à plusieurs niveaux de contrôle d'accès aux méthodes et aux variables (`public`, `protected`, `private`).
- à un langage fortement typé.
- à l'impossibilité de construire des pointeurs.
- à un "firewall"

Les avantages de la technologie Java Card

L'indépendance au hardware réalisée grâce au langage Java
$\Longrightarrow$ "Write Once, Run Anywhere"

La capacité de stockage et de gestion de multiples applications.
$\Longrightarrow$ possiblité de mises à jour des applications de la Java Card sans avoir besoin de changer de cartes.

La ***compatibilité*** avec les standards existants sur les cartes à puces.

Présentation de son architecture

Contraintes mémoires : 1Ko de RAM, 16Ko d'EEPROM et de 24Ko de ROM

$\Longrightarrow$ **Problèmes** pour construire une carte Java

$\Longrightarrow$ **Solutions** :

- ☞ ***supporter seulement un sous-ensemble des caractéristiques du langage Java***
- ☞ ***découper la machine virtuelle Java*** (JCVM : Java Card Virtual Machine) ***en deux parties***
  - une partie en dehors de la carte
  - une partie sur la carte

Présentation de son architecture

$\Longrightarrow$ **Problème** :

Beaucoup de tâches ne sont plus vérifiées à l'exécution.

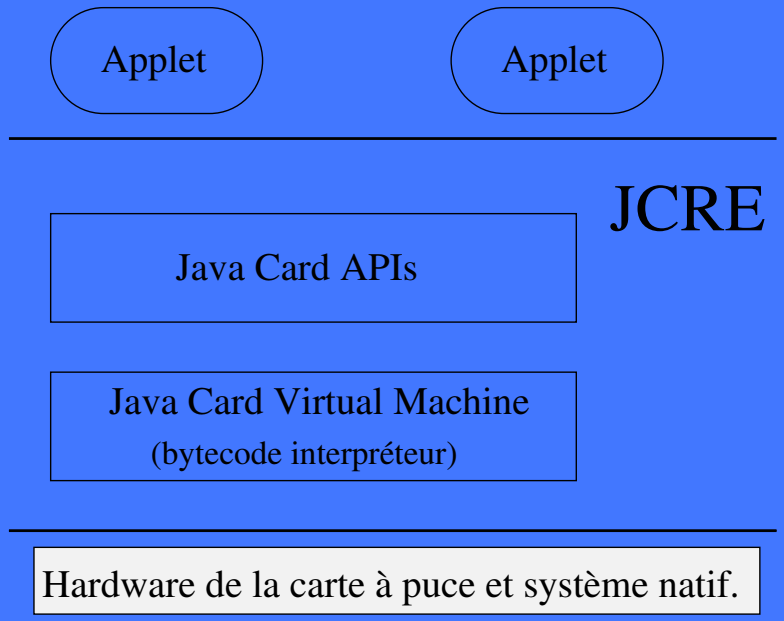
$\Longrightarrow$ **Solution** :

Un environnement d'exécution, ***le JCRE*** (Java Card Runtime Environment) ***chargé de fournir des mécanismes sécuritaires***.

Présentation de son architecture

- ***Le JCRE encapsule la complexité fondamentale*** du système des cartes à puce.
- A cause du découpage de la JCVM, ***la plateforme est distribuée*** entre la carte à puce et la machine de développement ***dans le temps et dans l'espace***.

Présentation de son architecture



Cette architecture de la technologie Java Card est définie par :

- The Java Card 2.1 Virtual Machine Specification.
- The Java Card 2.1 Runtime Environment Specification.
- The Java Card 2.1 API Specification.

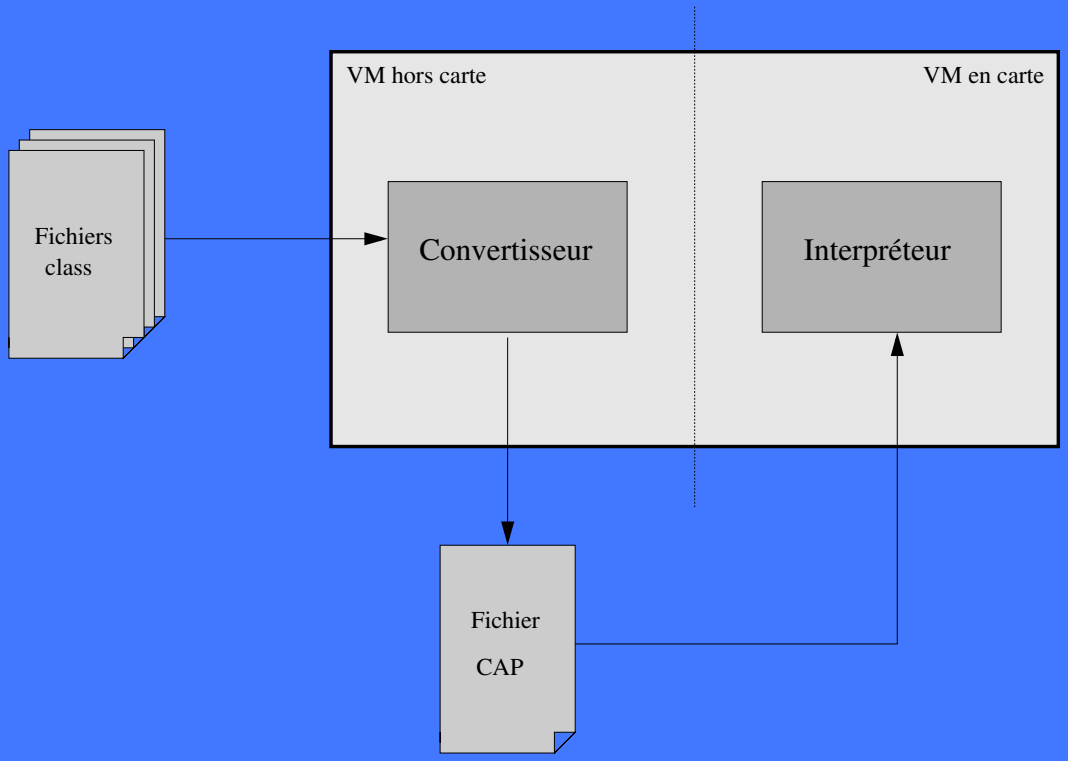
Caractéristiques Java non supportées

- ✗ Type simple de donnée de grosse taille : `long`, `double`, `float`
- ✗ Tableau plusieurs dimensions
- ✗ Caractères et chaines
- ✗ Chargement dynamique des classes
- ✗ Security Manager
- ✗ Ramasse-miettes et finalisation
- ✗ Threads
- ✗ Serialisation d'objet
- ✗ Clonage d'objet

Caractéristiques Java supportées

- ✓ Type simple de donnée de petite taille : `boolean`, `byte`, `short`
- ✓ Tableau à une dimension
- ✓ Paquetage Java, classes, interfaces et exceptions
- ✓ Caractéristique orientée objet : héritage, méthodes virtuelles, surcharge et création dynamique d'objet, access scope et binding rules
- ✓ Le mot clé `int` et le support des entiers sur 32 bits sont optionnels

La machine virtuelle de la Java Card : JCVM



Les deux parties implémentent toutes les fonctions d'une machine virtuelle.

La machine virtuelle de la Java Card : JCVM

Fichier CAP

Le fichier CAP est le format standard de fichier pour la compatibilité binaire de la plateforme Java Card.

Un fichier CAP contient une représentation binaire exécutable des classes d'un paquetage Java.
Un fichier CAP est un fichier JAR qui contient un ensemble de composants.
Chaque composant décrit un aspect du contenu d'un fichier CAP comme les informations sur les classes, les bytecodes exécutables, les informations de "linkage", les informations de vérifications, etc.

La machine virtuelle de la Java Card : JCVM

Fichier `export`

Un fichier `export` contient les informations publiques sur les API pour un paquetage de classes complet.

Convertisseur Java Card

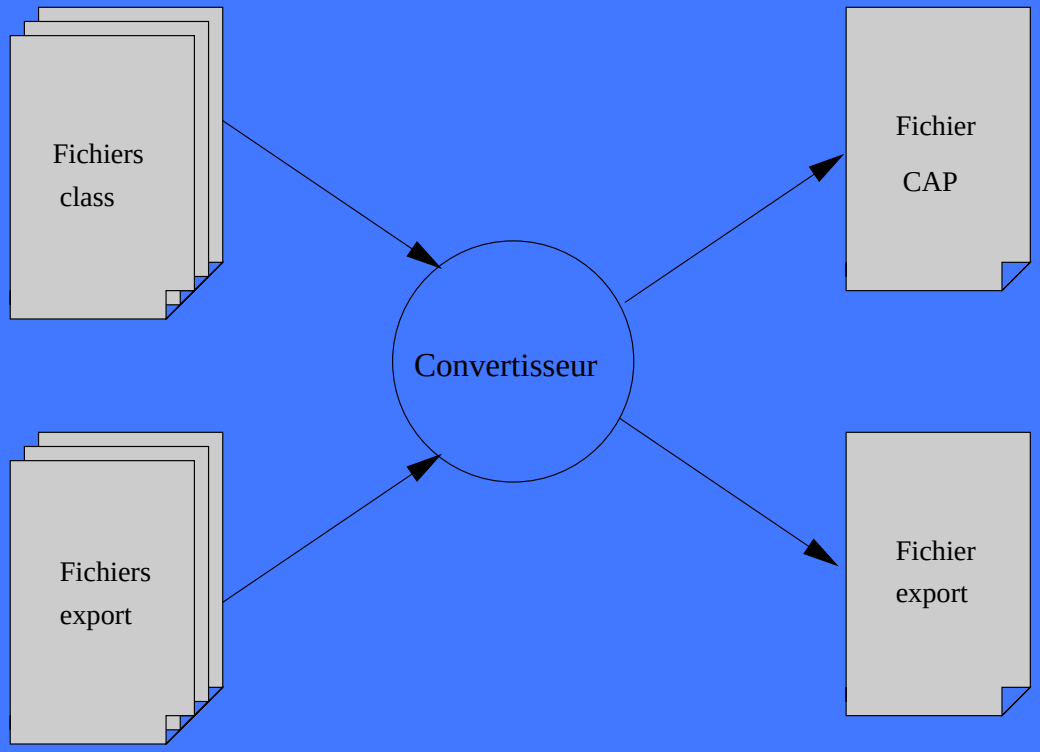
Le convertisseur réalise les taches que la machine virtuelle Java sur une station de travail doit réaliser au chargement des classes :

- ***Vérifie*** que le chargement des ***images des classes Java*** sont bien formées.
- ***Contrôle des violations*** du langage Java Card.
- ***Réalise des initialisations*** de variables `static`.

La machine virtuelle de la Java Card : JCVM

- ***Résout les références symboliques*** aux classes, méthodes et champs dans la forme la plus compact qui peut être traitée efficacement sur la carte.
- ***Optimise le bytecode*** en tirant avantage des informations obtenu au chargement des classes et au ”linkage”.
- ***Alloue l'espace et créer les structures de données de la machine virtuelle*** pour représenter les classes.

La machine virtuelle de la Java Card : JCVM



La sortie produite par le convertisseur est un fichier CAP et un fichier `export` pour le paquetage converti.

La machine virtuelle de la Java Card : JCVM

Interpréteur Java Card

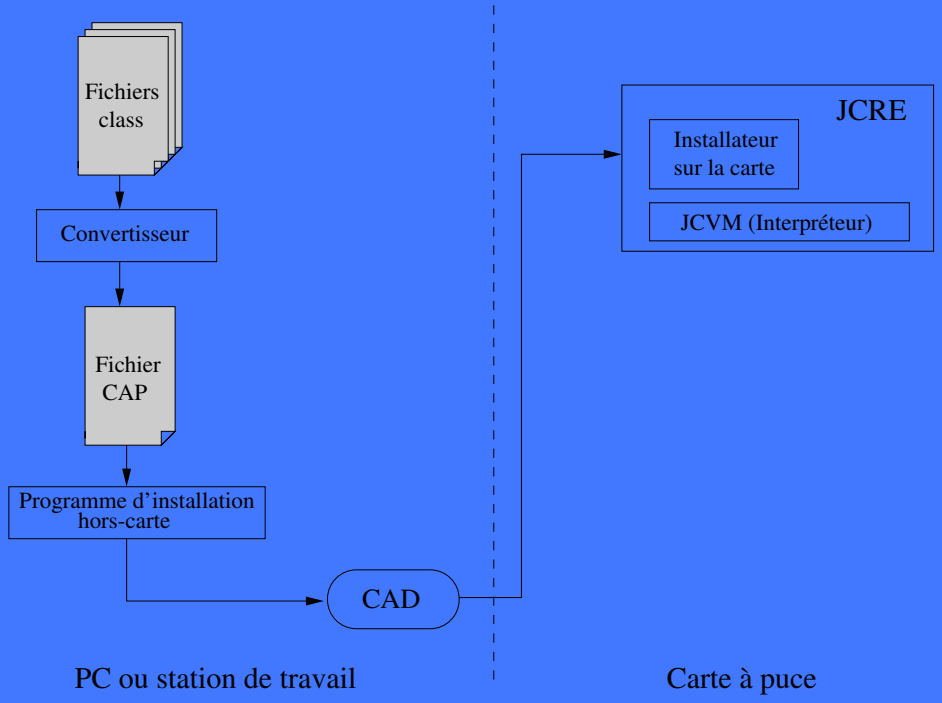
L'interpréteur Java Card fournit le ***support d'exécution*** du modèle du langage Java qui ***autorise une indépendance au hardware*** du code de l'applet.

L'interpréteur réalise les taches suivantes :

- ***Exécute les instructions du bytecode*** et ultimement les exécutions des applets.
- ***Contrôle les allocations de mémoire et les créations d'objets.***
- Joue un rôle crucial pour s'assurer de la ***sécurité*** lors ***de l'exécution.***

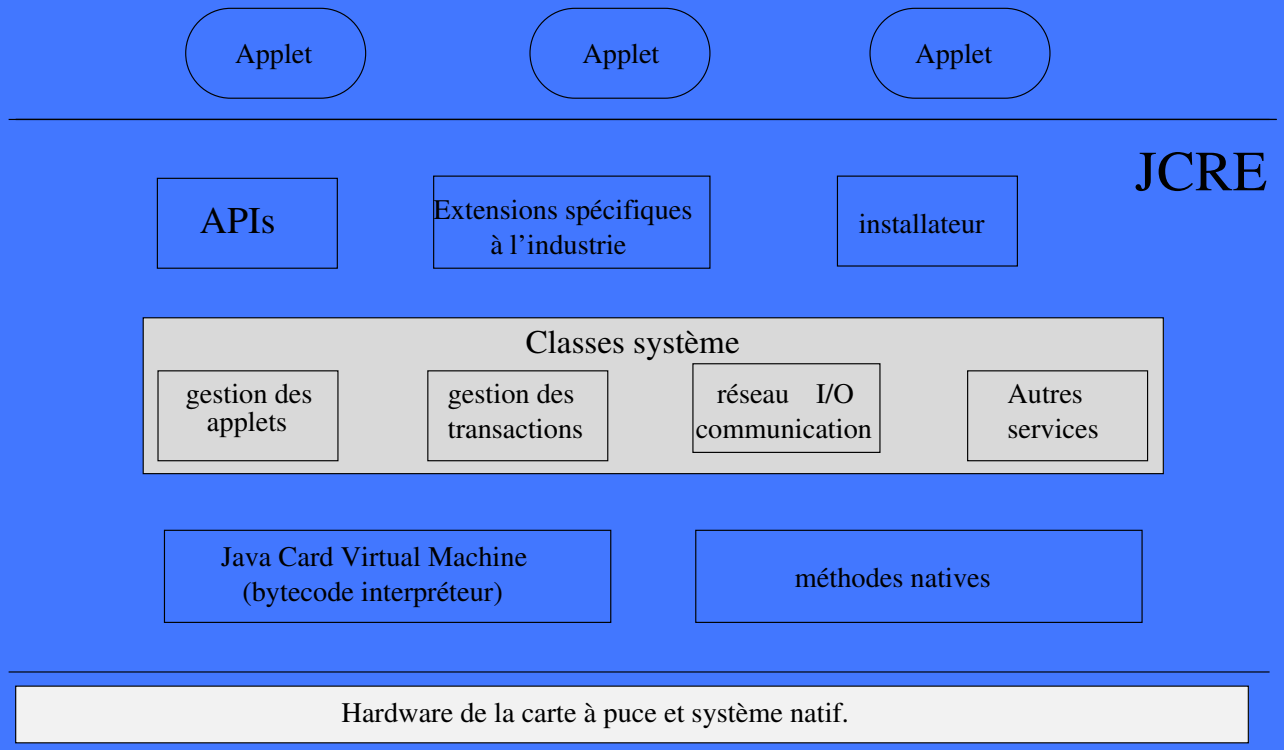
La machine virtuelle de la Java Card : JCVM

Installateur Java Card et programme d'installation hors carte



L'environnement d'exécution de la Java Card : JCRE

Le JCRE est responsable de la gestion des ressources de la carte, communication réseau, exécution des applets, du système de la carte et de la sécurité des applets.



Le cycle de vie du JCRE

Le JCRE est initialisé à la phase d'initialisation de la carte (seulement une fois dans le cycle de vie de la carte).

Le cycle de vie du JCRE est le même que le cycle de vie de la carte.

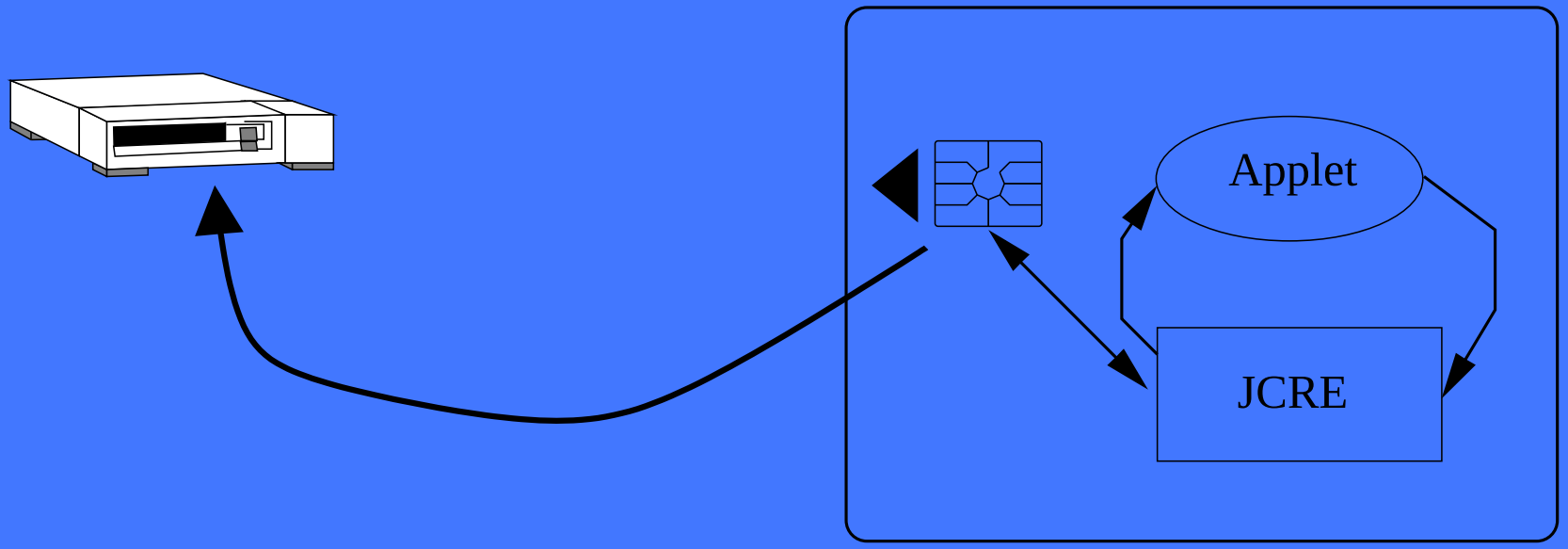
L'environnement d'exécution de la Java Card : JCRE

- Quand la ***tension est enlevée***, la ***machine virtuelle*** est seulement ***suspendue***.
  L'état du JCRE et des objets créés sur la carte est préservée.
- A la prochaine ***remise sous tension***, ***le JCRE relance l'exécution de la machine virtuelle*** en chargeant les données depuis la mémoire persistante.
- Durant le reset, si une transaction n'a pas été terminée, le JCRE réalise tous les nettoyages nécessaires pour remettre le JCRE dans un état consistant.

L'environnement d'exécution de la Java Card : JCRE

Comment le JCRE opère durant les sessions CAD ?

le JCRE opère comme une carte à puce typique.



L'environnement d'exécution de la Java Card : JCRE

Les caractéristiques du JCRE

- ***Objets persistants et temporaires***
  Objets persitants existent à travers les sessions CAD.
  Objets temporaires contiennent des données temporaire qui ne persistent pas au travers des sessions CAD.
- ***Atomicité et transactions***
  Chaque opération d'écriture de la JCVM est ***atomique***.
  Le champ mis à jour prend soit la nouvelle valeur soit il est restauré à la valeur précédente.
  Une transaction est un bloc d'opérations atomiques.

L'environnement d'exécution de la Java Card : JCRE

- ***Firewall des applets et les mécanismes de partage***

  Le firewall isole les applets à l'intérieur de leur espace (contexte).

  Si les applets ont besoin d'accéder à des données ou à des services du JCRE, la machine virtuelle autorise de telles possibilités à travers des ***mécanismes sécurisés de partage.***

Les APIs Java Card

Les APIs Java Card sont un ensemble de classes optimisées pour la programmation des cartes à puce en accord avec le modèle ISO7816.

Le paquetage `java.lang`

Le paquetage Java Card java.lang est un sous ensemble strict de son équivalent le paquetage java.lang sur la plateforme Java.

Le paquetage `javacard.framework`

Ce paquetage fournit la structure des classes et des interfaces pour le noyau fonctionnel des applets Java Card.

Les APIs Java Card

Le paquetage `javacard.security`

Le paquetage javacard.security fournit une architecture aux fonctions cryptographiques supportées sur la plateforme Java Card.

Le paquetage `javacardx.crypto`

Le paquetage javacardx.crypto est un paquetage d'extension.

Conclusion et perspectives

- La carte tente de s'imposer partout avec les cartes multiapplicatives
- La carte possède une très grande sécurité au niveau hardware
- Qu'en est-il du software ?

Références

- ZhiqunChen Zhiqun Chen.
  Java Card™ Technology for Smart Cards.
  Addison-Wesley - ISBN 0-201-70329-7.
- LudovicRousseau Ludovic Rousseau.
  La sécurité des cartes à puce.
  Cours de cryptographie et sécurité. Janvier 2001, Bordeaux I.
- Nathalie Feyt Nathalie Feyt.
  La carte à puce.
  Séminaire ENSEIRB. Janvier 2001, Bordeaux I.
- JCTHP Java Card Technology Home Page.
  http ://java.sun.com/products/javacard/
- ISO International Standards Organisation.
  http ://www.iso.ch