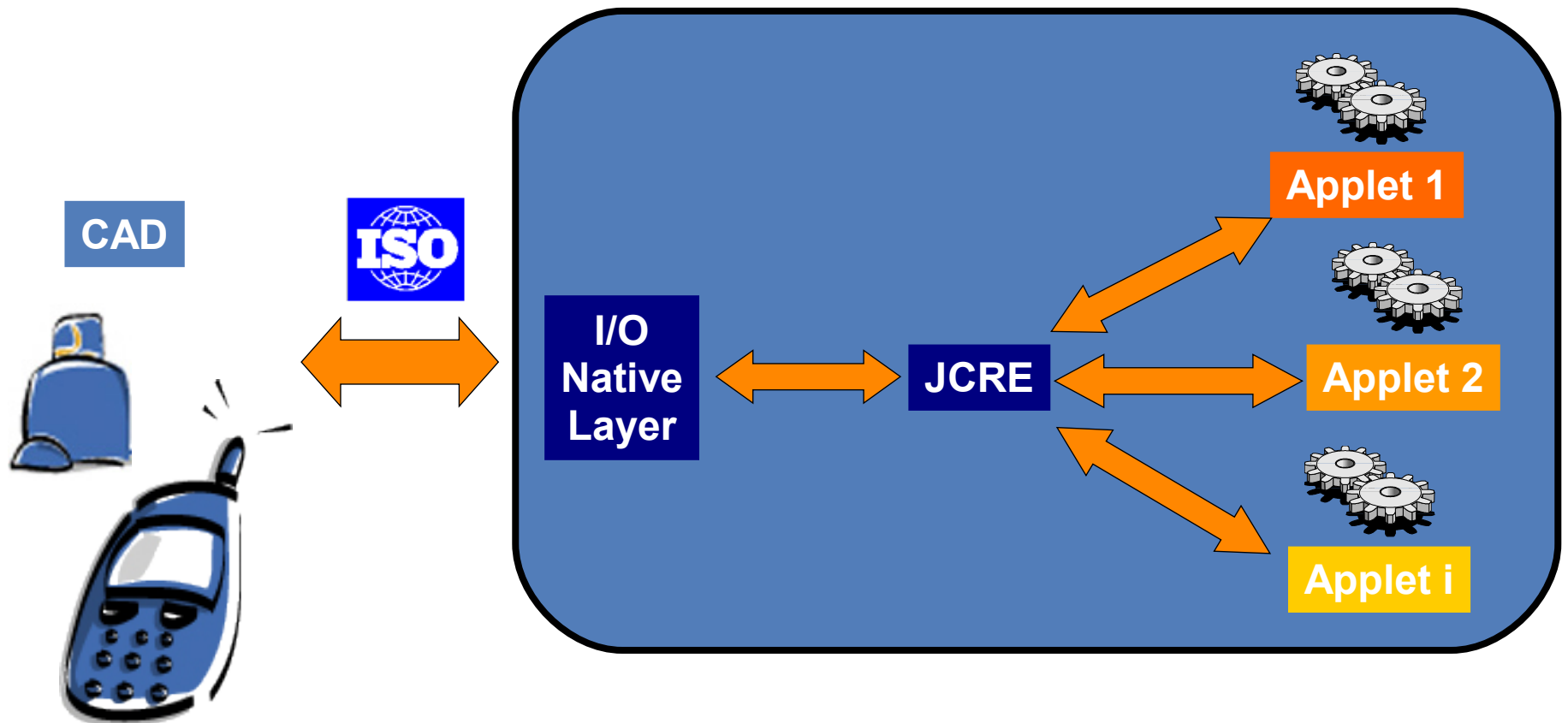
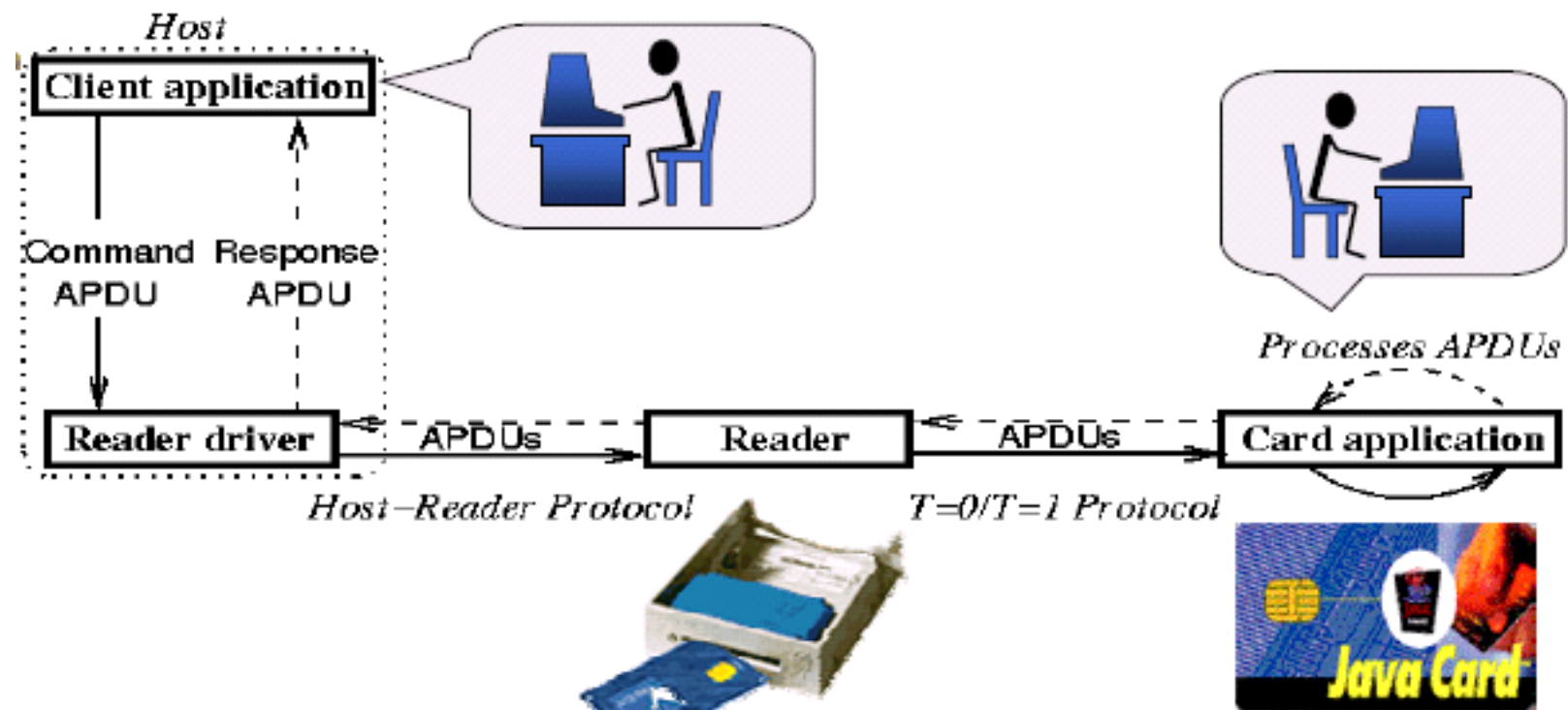


Communication avec une applet

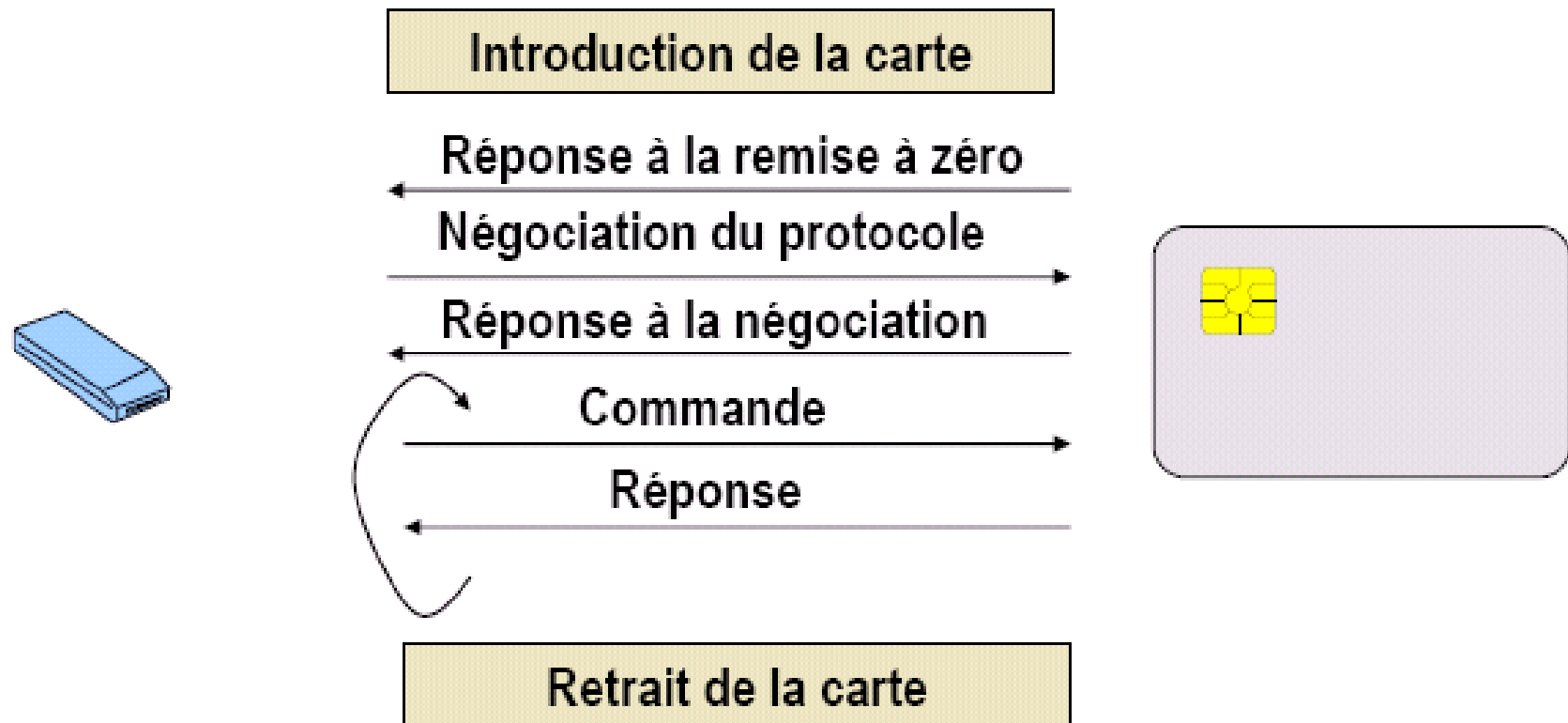


Les communications APDU

- La communication APDU entre l'application host et l'applet
 - Encapsulation du protocole sous-jacent (T=0, T=1, etc.)



Communication APDU (ISO7816-4)



Le protocole APDU (version standard ou étendue)

- 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) : transit de la carte vers le CAD

Corps optionnel	Enqueue obligatoire	
Champ de données	SW1	SW2

- Standardise différents codes de réponse SW1/SW2
 - 90 00: OK
 - 6E XX: Classe inconnue
 - 6D XX: instruction inconnue
 - 9X XX: Erreur applicative

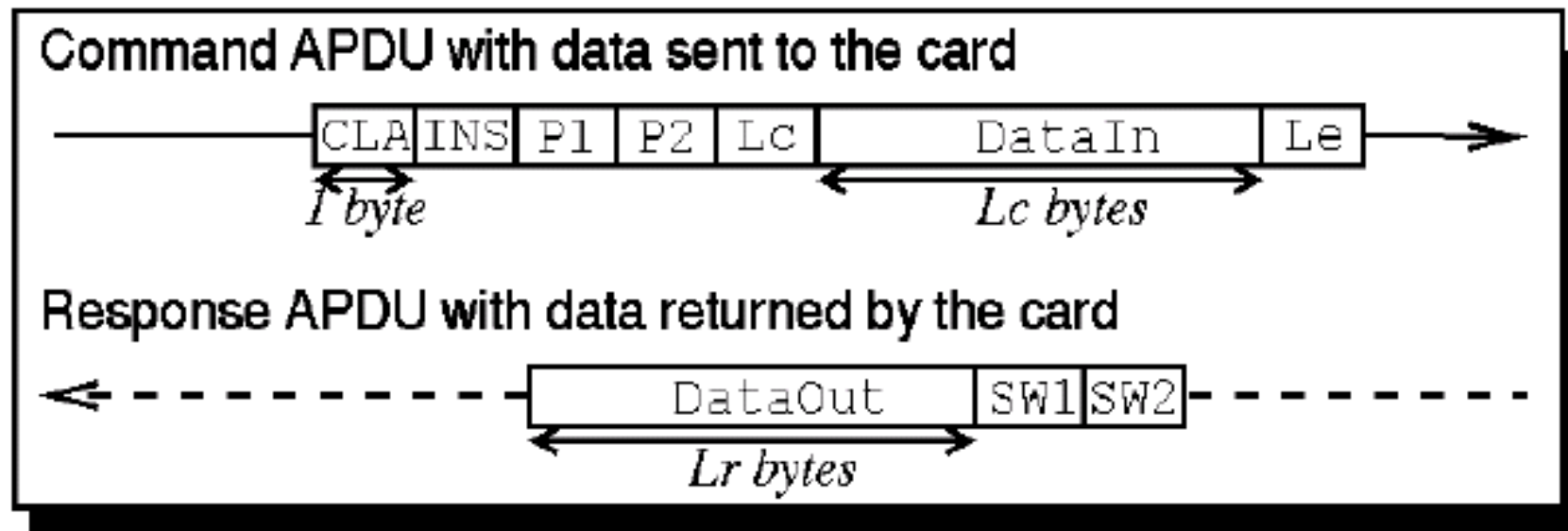
Les différents cas d'échange APDU

	commande APDU	réponse APDU						
cas 1	<table><tr><td>entête</td></tr></table>	entête	<table><tr><td>SW</td></tr></table>	SW				
entête								
SW								
cas 2	<table><tr><td>entête</td><td>Le</td></tr></table>	entête	Le	<table><tr><td>données</td><td>SW</td></tr></table>	données	SW		
entête	Le							
données	SW							
cas 3	<table><tr><td>entête</td><td>Lc</td><td>données</td></tr></table>	entête	Lc	données	<table><tr><td>SW</td></tr></table>	SW		
entête	Lc	données						
SW								
cas 4	<table><tr><td>entête</td><td>Lc</td><td>données</td><td>Le</td></tr></table>	entête	Lc	données	Le	<table><tr><td>données</td><td>SW</td></tr></table>	données	SW
entête	Lc	données	Le					
données	SW							

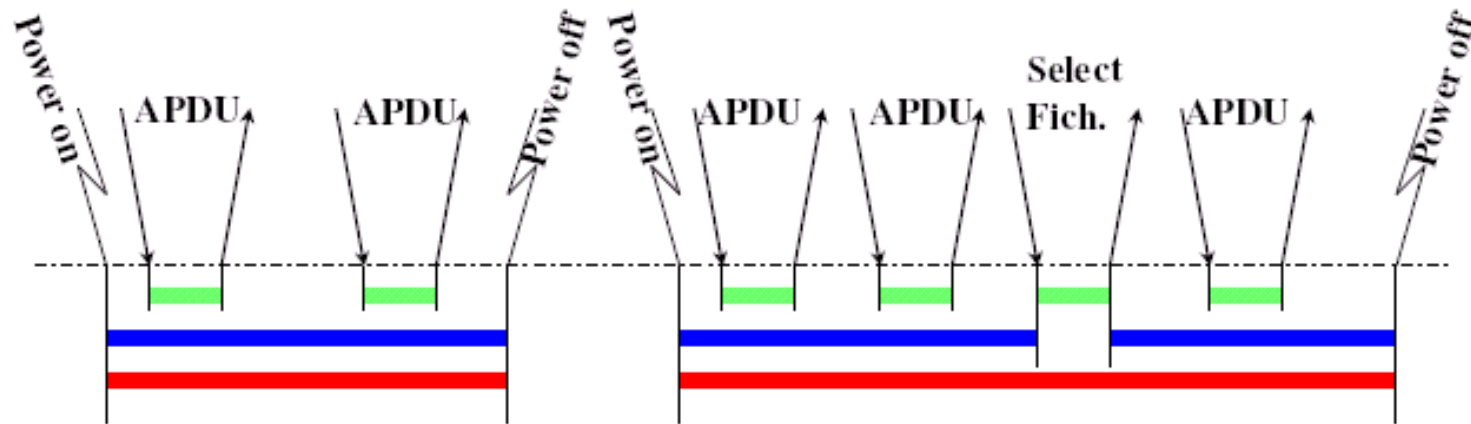
La communication APDU

- Permet tous les types de transfert de données

Cas	Commande	Réponse
1	Pas de données	Pas de données
3	données	Pas de données
2	Pas de données	données
4	données	données

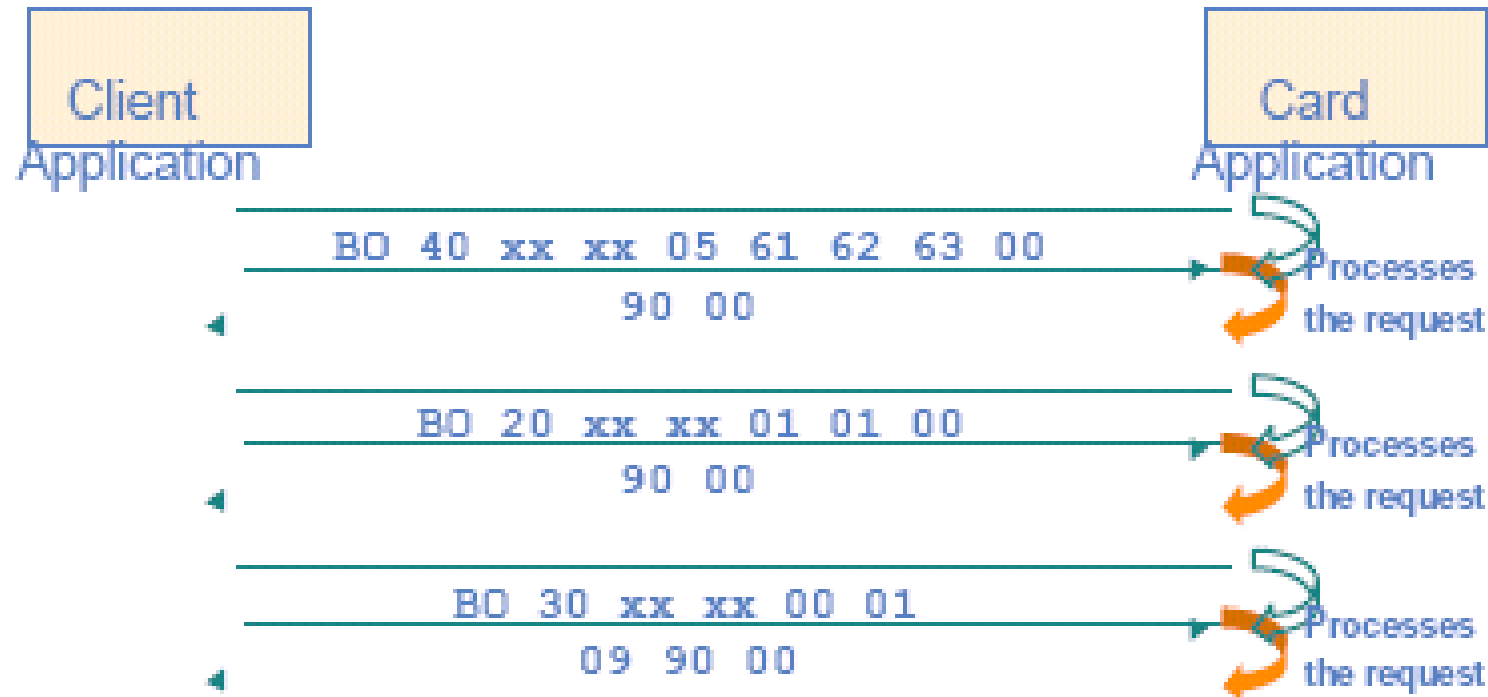


Du point de vue de l'application hôte



- Durée d'une APDU
- Durée d'une session
- Durée d'une connexion

Exemple de communication APDU



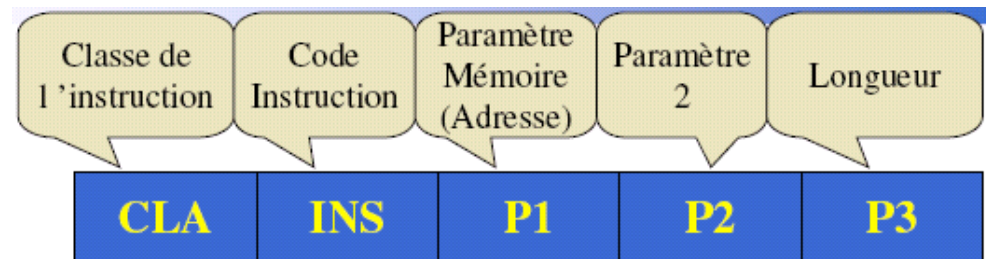
- La transmission d'APDU en T=0 ou T=1 est détaillée dans les annexes de l'ISO7816-4

Complexité

- Afin de pouvoir rendre transparent la communication APDU vis à vis des différents protocoles sous-jacents, il est nécessaire d'appeler les méthodes de communication dans un certain ordre.
- En effet, $T=0$ par exemple ne permet pas d'avoir des commandes entrantes **et** sortantes ...

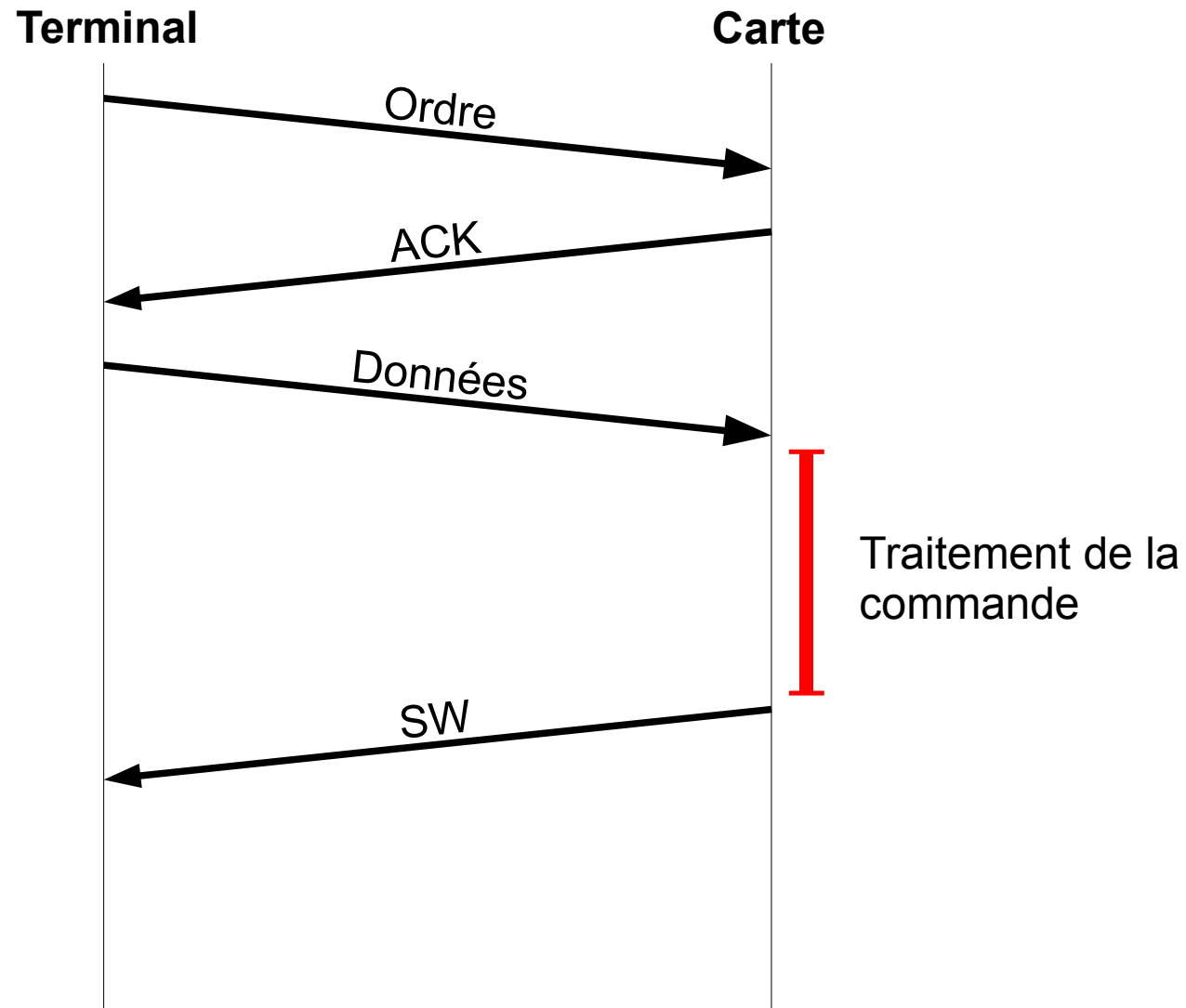
Protocole T=0 (Transmission semi-duplex de caractères asynchrones)

- Commande entrante (envoi des données à la carte)
- Commande sortante (récupère des données de la carte)
- Structure de l'ordre
- Octets de procédure

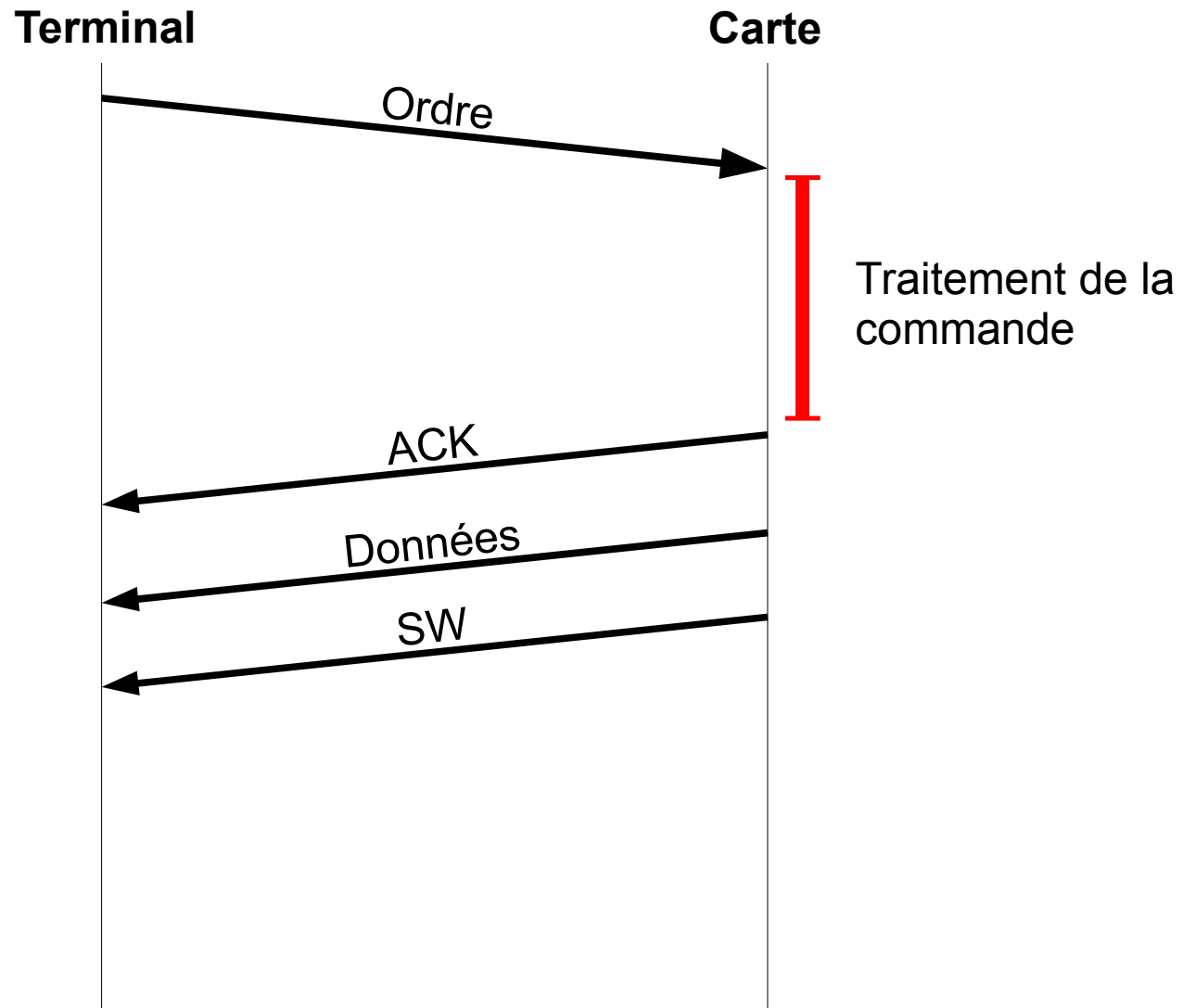


Octet	Valeur	Résultat sur VPP	Résultat en transfert de données	Puis réception de
NUL	'60'	Pas d'action	Pas d'action	Un octet de procédure
ACK	INS	État de pause	Tous les autres octets de données	Un octet de procédure
	INS ⊕ '01'	État d'écriture	Tous les autres octets de données	Un octet de procédure
	INS ⊕ 'FF'	État de pause	L'octet de données suivant	Un octet de procédure
	INS ⊕ 'FE'	État d'écriture	L'octet de données suivant	Un octet de procédure
SW1	'6X' (≠'60'), '9X'	État de pause	Pas d'action	Un octet SW2

T=0 : Commande entrante

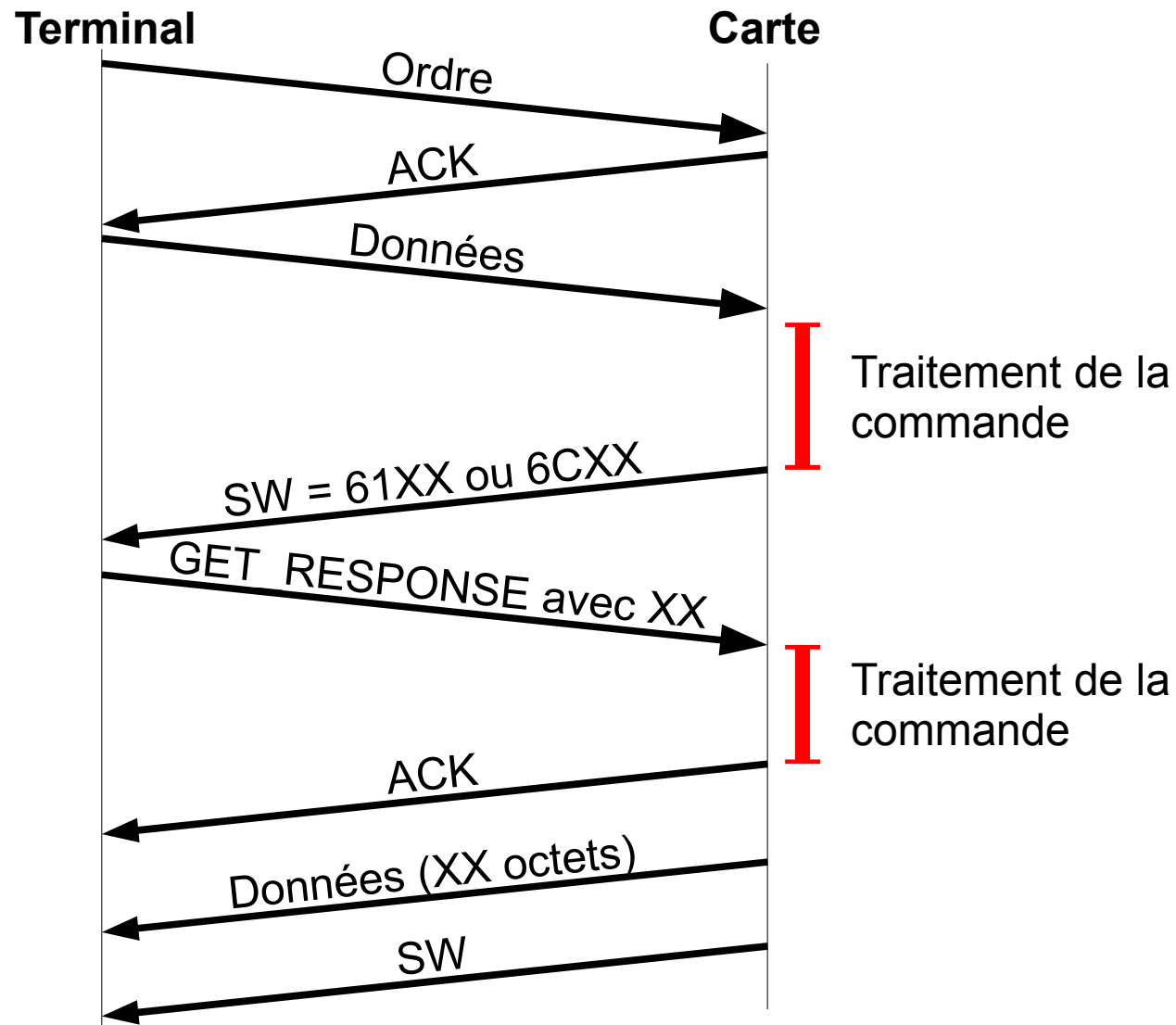


T=0 : Commande sortante



Comment est traité le cas 4 APDU (données entrantes et sortantes) ?

- Utilisation de la commande spéciale GET_RESPONSE



Problème des conditions limites

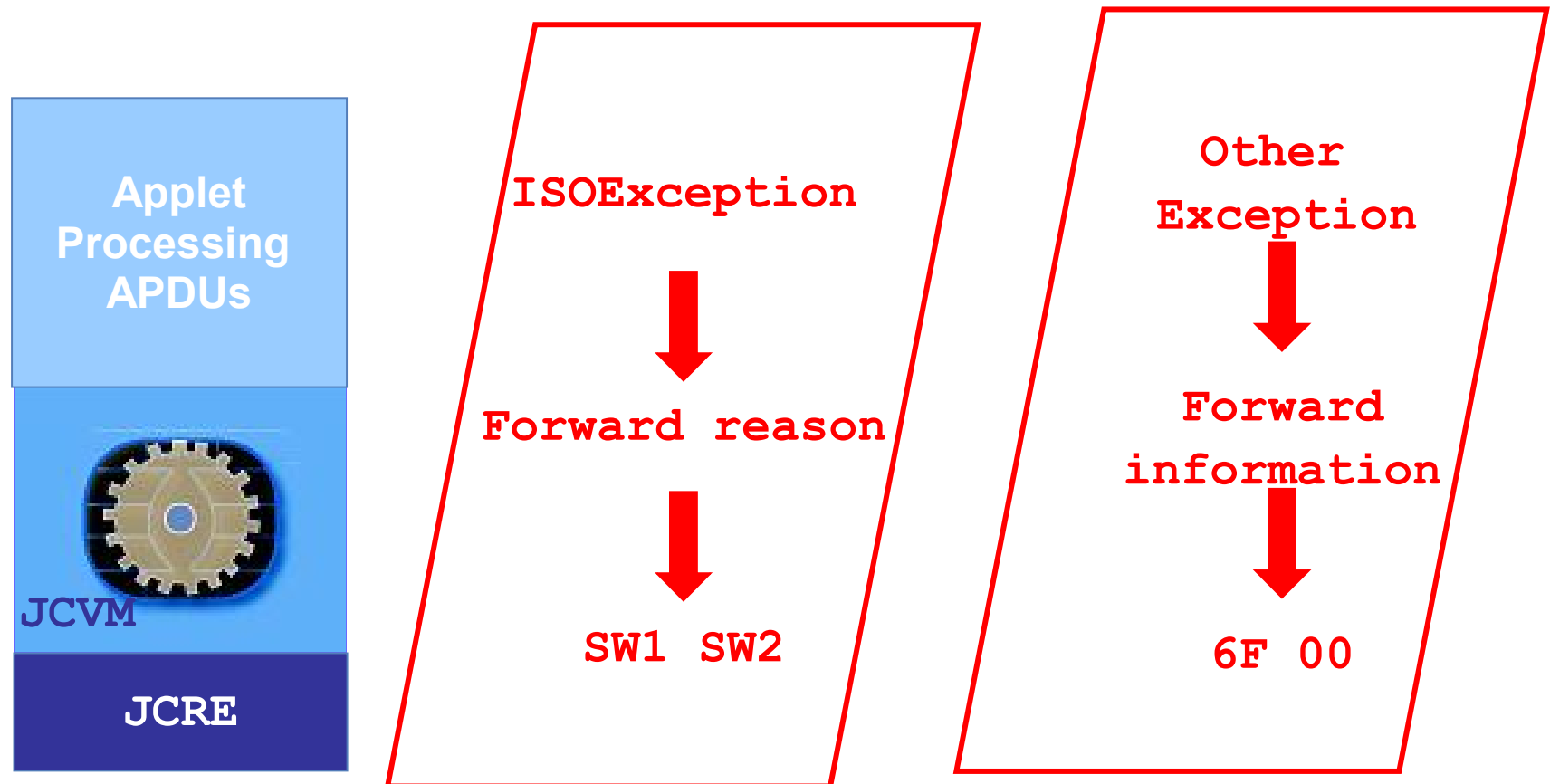
- Cas des cartes à petit buffer APDU et/ou traitement des APDUs étendus
- Taille du buffer « APDU » (un des problèmes pour l'interopérabilité) :
 - Une vingtaine de bytes sur des cartes « anciennes »
 - 261 (5+256) sur les cartes d'aujourd'hui

Pour finir sur les communications, attention ...

- Certaines cartes respectent très strictement les spécifications ISO7816-3,4 etc.
- L'octet **INS doit être pair** ! Il doit être différent de 0x6X et 0x9X
- Idem pour les CLA et les logical channels, les secure channels, etc.

Communication en Java Card (1/2)

- `ISOException.throwIt(short reason)` : pour renvoyer des status words différents de 0x9000



Communication en Java Card (2/2)

- La classe `APDU`
 - `getBuffer` : **accès aux 5 premiers octets**
 - `setIncomingAndReceive ()`
 - `receiveBytes ()`
 - `waitExtension ()` : **octet de procédure**
 - `setOutgoing ()`, `setOutgoingLength ()`
 - `sendBytes ()`, `sendBytesLong ()`
 - `setOutgoingAndSend ()`