



carte de professionnel de santé

CALL FOR COMMENTS

Use of the PC/SC Resource Manager in the French healthcare sector

Version 1.0

Date 23/03/2009

Reference PDT-EB-007-English

Information FREE

Etat REFERENCE

GIP-CPS

8bis, rue de Châteaudun

75009 PARIS

www.gip-cps.fr

GIE SESAM-VITALE

5, boulevard Alexandre Oyon

72 019 LE MANS Cedex 2

www.sesam-vitale.fr

© G.I.E. SESAM-VITALE and GIP-CPS

Conforming to the French law, article L.122-4, this document cannot be reproduced, modified, copied or distributed without written authorization of the G.I.E. SESAM-VITALE and the GIP-CPS, even in electronic format.

TABLE DES MATIERES

1	INTRODUCTION.....	4
1.1	OBJECT OF THE CALL FOR COMMENTS	5
1.2	CONTENT OF THE DOCUMENT	5
1.3	ABBREVIATIONS	6
1.4	DEFINITIONS	7
2	PRACTICAL INFORMATION CONCERNING THE CALL FOR COMMENTS	8
3	PRESENTATION OF THE CURRENT ARCHITECTURE OF A HCP WORKSTATION.....	9
4	CALL FOR COMMENTS – ISSUES TO BE SOLVED	14
4.1	CONFIGURATIONS TO BE COVERED BY THE FUTURE ARCHITECTURE	14
4.2	PC/SC STANDARD AND THE TARGET OPERATIONAL SYSTEMS	15
4.3	CONFIGURATION CASES A AND B	16
4.4	CONFIGURATION CASES C AND D	18
4.5	CONSTRAINTS TO BE RESPECTED	20
4.6	PRESENTATION OF THE PROPOSED ARCHITECTURE FOR PHASE 1	20
4.6.1	<i>Hypothesizes</i>	20
4.6.2	<i>Architecture of the proposed architecture</i>	22
4.6.3	<i>Principles of functioning</i>	23
4.6.4	<i>Updating of a SESAM-Vitale terminal</i>	27
4.6.5	<i>Functioning of autonomous S-V terminals</i>	27
4.6.6	<i>Workstations with « virtual clients »</i>	28
4.7	ARCHITECTURE WITH MODIFIED S-V TERMINALS - PHASE 2.....	29
4.8	MULTIPLE READERS ON A WORKSTATION	31
5	IMPLEMENTATION EXAMPLES.....	33
6	ANNEX 1 : EXTRACTS OF PC/SC V1 & V2 SPECIFICATIONS.....	34
6.1	VENDOR-DEFINES FEATURES	34

TABLE DES ILLUSTRATIONS

DIAGRAM 1 :	WORKSTATION EQUIPPED WITH A SESAM-VITALE TERMINAL.....	10
DIAGRAM 2 :	WORKSTATION EQUIPPED WITH PC/SC READERS (HOSPITALS)	12
DIAGRAM 3 :	CONFIGURATIONS TO BE COVERED BY THE FUTURE ARCHITECTURE	14
DIAGRAM 4 :	CONFIGURATION WITH TWO PC/SC READERS	16
DIAGRAM 5 :	CONFIGURATION WITH ONE SESAM-VITALE TERMINAL	18
DIAGRAM 6 :	THE PROPOSED ARCHITECTURE FOR PHASE 1	22
DIAGRAM 7 :	THE PROPOSED ARCHITECTURE FOR PHASE 2	29
DIAGRAM 8 :	CONFIGURATIONS WITH MULTIPLE S-V TERMINALS.....	31

1 Introduction

The GIE SESAM-VITALE, the GIP-CPS and their members are considering the opportunity to modernize the architecture of the HealthCare (HC) Professional workstation.

The current architecture has been specified in the middle of the nineties and is built on a proprietary Resource Manager, called GALSS, allowing access to cards and readers.

A logical evolution would be to migrate towards the standard PC/SC Resource Manager.

The problem to solve, however, is that over 250.000 workstations in the private sector are equipped with **SESAM-Vitale terminals**. This type of terminal embarks a local application and provides 2 or 3 slots for smart cards, a keyboard and a display.

Furthermore, considering the various actors in the system (about 200 software editors and a dozen of S-V terminal industrials), the updating of the software of all the terminals in the field will take at least 5 years.

This means that the future architecture has to be introduced in 2 phases :

- ⇒ **Phase 1 - short term**, the architecture must be compatible with the current SESAM-Vitale terminals (no software modification), allowing applications to benefit rapidly from the PC/SC architecture,
- ⇒ **Phase 2 - medium term**, prepared in parallel, this architecture is based on S-V terminals with adapted software allowing a more conventional (and elegant) integration in the PS/SC architecture.

The hospitals tend to deploy only PC/SC readers especially for their low TCO (Total Cost of Ownership) ; the readers are cheap and easy to install. With some exceptions, the hospitals do not use SESAM-Vitale terminals.

The target of this Call for Comments is to collect information about the feasibility and the best way of implementing the standard PC/SC Resource Manager for SESAM-Vitale terminals.

If the outcome is positive, we will build a « Proof of Concept » (maquette) which can be industrialized for distribution if the results are positive.

The GIP-CPS

The GIP-CPS is a non-profit organization, created in 1993 in order to emit the national French Healthcare Professional card, called « **CPS** »¹, for all healthcare professionals regulated by law and their employees.

Its members are the major actors in the Healthcare sector : the State (Ministries of Health, Agriculture and Budget), the professional orders (physicians, pharmacists, dentists and midwives), the healthcare-insurances (public and private) and representatives of medical and hospital associations.

The CPS card (with RSA crypto-processor) allows its holder to authenticate himself and to digitally sign the documents he creates.

¹ CPS = Carte de professionnel de Santé = French Healthcare professional card

Beside cards, the GIP-CPS distributes corresponding middleware (cryptographic libraries : CSP & PKCS#11 for Windows - PKCS#11 and Tokend-CDSA for MacOS - PKCS#11 for LINUX). The GIP-CPS maintains a public X.500 LDAP-directory ² and our certificates are compliant with X.509.

The GIE SESAM-VITALE

The GIE SESAM-VITALE has been created in 1993 by the public healthcare-insurances in order to emit the national French Patient Data Card, called « **Vitale** ». Later, the private healthcare-insurances joined the GIE. The GIE is in charge of the national SESAM project for the dematerialization of reimbursement forms and the development of on-line services for HC Professionals on behalf of the healthcare-insurances.

Some figures of the French HC sector :

- ⇒ the CPS card is distributed to over 600.000 HC Professionals (mainly in the private sector), the potential is about 1.5 million cards through the generalization of the CPS within the 3.500 French hospitals.
- ⇒ the Vitale card is distributed to over 50 million persons (all insured persons of 16 years and over) ;
- ⇒ over 250.000 HC Professionals in the private sector are equipped with a SESAM-Vitale terminal on their workstation (they have generated about 1.5 billion electronic reimbursement forms in 2008) ;
- ⇒ **currently, over 300.000 HC Professional workstations are equipped with readers and HC software, the potential number of workstations is over 1 million.**

1.1 Object of the call for comments

The object of this document is to describe an architecture based on PC/SC proposed by the GIE SESAM-VITALE and the GIP-CPS, and to submit a number of questions. The PC/SC experts of the addressed companies are kindly requested to comment the architecture on its feasibility and to answer the questions.

**The proposed architecture reflects our understanding which is quite limited.
Any suggestion to improve the proposed architecture is welcome and will be studied by our project teams.**

1.2 Content of the document

Beside the introduction, the document contains the following chapters :

- chapter 2 presents the practical information concerning the Call for Comments,
- chapter 3 presents the current architecture of the HCP workstation,
- chapter 4 presents the various configurations to be covered, the problems to be solved and their constraints, a proposed architecture and the questions to be answered.

² <ldap://annuaire.gip-cps.fr/> and <http://annuaire.gip-cps.fr/>

1.3 Abbreviations

Abbreviations	
APDU	Application Protocol Data Unit = card instruction (standard ISO 7816)
API	Application Programming Interface
API-CPS	API offering services for the CPS card
API-SSV	API offering SESAM-Vitale Services
API-Vitale-Reading	API offering SESAM-Vitale Services but limited to the reading of a Vitale card
ATR	Answer To Reset (of the card at power-on)
CB	<i>Carte Bancaire</i> = electronic payment card (Visa, MasterCard, ...)
CDSA	Common Data Security Architecture, specific for MacOS
CPS	<i>Carte de Professionnel de Santé</i> = French HC Professional card
CSP	Cryptographic Service Provider, specific for Windows
EFT	Electronic Funds Transfer
FU	Functional Unit = card slot on a SESAM-Vitale terminal
GALSS	<i>Gestionnaire d'Accès au Lecteur Santé Social</i> = current proprietary French resource manager for cards and readers
HC	Healthcare
HCP	Healthcare Professional
PKCS#11	Public Key Cryptographic Standards, part 11 - standard cryptographic library, allowing the creation of "drivers" for any type of cryptographic token (not OS specific)
PnP	Plug and Play
POS	Point Of Sales
PS	<i>Professionnel de Santé</i> = HC Professional = HCP
PSS	<i>Protocole Santé Social</i> = current proprietary French protocol between workstation and SESAM-Vitale terminal
SSO	Single Sign On
S-V	SESAM-Vitale

1.4 Definitions

Definitions	
GALSS aware Application	Application using the current GALSS in order to access cards (and readers).
PC/SC aware Application	Application using the PC/SC Resource Manager in order to access cards (and readers) through the standard PC/SC architecture.
DUO card	<p>Patient Data Card issued by the private healthcare-insurances. The DUO card is technical identical to the Vitale card. They can be used simultaneously during the creation of an electronic reimbursement form.</p> <p>Note : The DUO card has the same ATR as the Vitale card.</p>
Command for SESAM-Vitale terminal	<p>The commands for the SESAM-Vitale terminal have a proprietary format and are treated by an embedded application in the terminal.</p> <p>Such a command contains :</p> <ul style="list-style-type: none"> ○ either an application command (ex. signature of an electronic reimbursement form), ○ or a card instruction.
Windows registry	The Windows Registry is a database storing settings and options for Microsoft Windows operating systems. It contains information and settings for all the hardware, operating system software, most non-operating system software, and per-user settings.
PC/SC Resource Manager	<p>The PC/SC Resource Manager allows the use of PC/SC readers as « native » resources of the OS.</p> <p><i>Note : In this document we use abusively the expression « PC/SC Resource Manager », it would have been more correct to use the expression « ICC Resource Manager ».</i></p>

2 Practical information concerning the Call for Comments

The present Call for Comments is open to all companies and in particular :

- industrials of smart card readers,
- editors of Operating Systems and « desktop virtualization systems »,
- experts in security and workstation architecture,
- ...

The companies are invited to send their comments before

end of April 2009

to the following mail address :


support.commentaire-pcsc@sesam-vitale.fr

Questions concerning this Call for Comments can be sent to the same mail address.

The GIP-CPS and the GIE SESAM-VITALE guarantee the confidentiality of your comments.

Questions

The questions are numbered and have the following format :

	<i>In italics : introduction / context of the question.</i> Question number n addressed to the PC/SC experts.
---	---

Participation conditions

The answers to this Call for Comments will be delivered free of charge.

The participants agree that their comments can be freely used by the GIE SESAM-VITALE and the GIP-CPS.

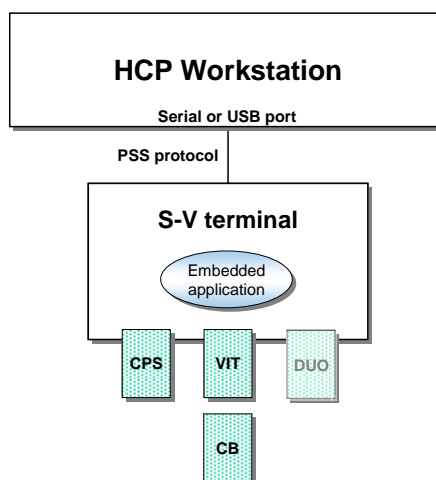
3 Presentation of the current architecture of a HCP workstation

The S-V terminal

The HCP workstations in the private sector are equipped with S-V terminals designed specifically for the SESAM application (dematerialization of reimbursement forms).

These terminals designed in the beginning of the nineties have the following characteristics :

- ⇒ 2 or 3 slots for smart cards, called Functional Units (FU),
- ⇒ a keyboard and display (mainly used for PIN entering),
- ⇒ specific software, mainly composed of the operating system, protocol drivers and an embedded application,
- ⇒ a proprietary protocol, called PSS (only used in the French healthcare sector).
- ⇒ a serial or USB interface,



The commands for the SESAM-Vitale terminals have a proprietary format and are treated by the embedded application on the terminal. They contain :

- either an application (macro) command (ex. signature of a reimbursement form), the command is executed locally with possible access to all inserted cards,
- or a card instruction - power-on, APDU, ... - to be executed by the card in the specified slot, called functional unit (FU).

The slot numbers on a S-V terminal are standardized, they allow applications to specify to which card an APDU has to be directed :

- ⇒ FU = 1 for the CPS card
- ⇒ FU = 2 for the Vitale card,
- ⇒ FU = 3 for a possible other card - ex. another insurance card like DUO
(only some S-V terminals offer a 3rd slot).

The S-V terminal can also contain a stand-alone EFT/POS application. In that case, the payment card (CB) is introduced in the Vitale slot. During the payment transaction, the S-V terminal may ignore all commands coming from the workstation (in general they all ignore them).

Workstations in private sector

The current architecture on the HCP workstation allows the SESAM and other medical applications to access cards and the embedded application in the SESAM-Vitale terminal through the GALSS resource manager. The diagram hereunder is the usual configuration in the private sector.

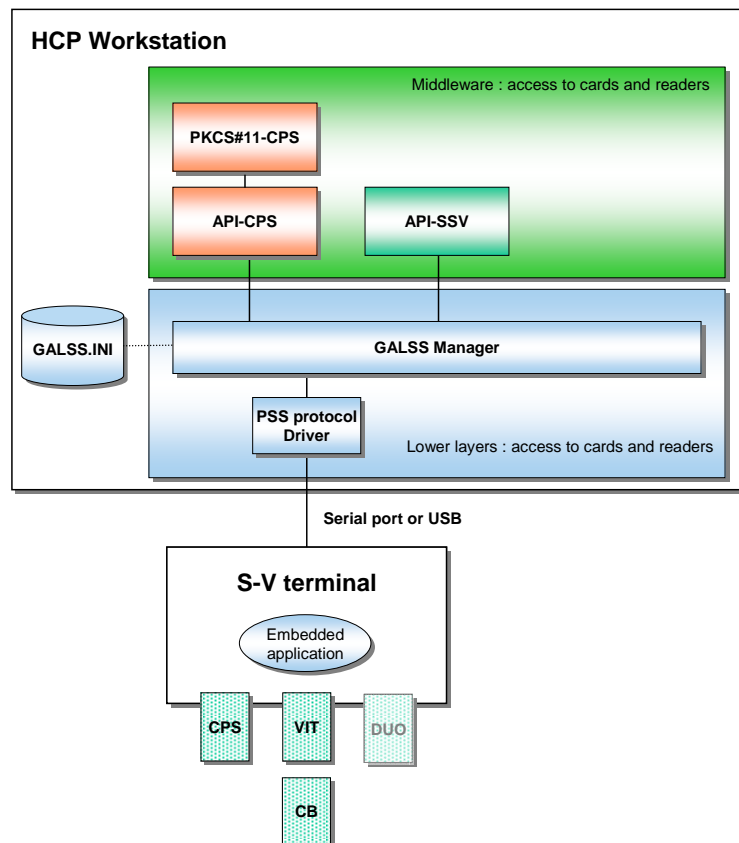


Diagram 1 : Workstation equipped with a SESAM-Vitale terminal

In this configuration, the applications communicate exclusively with the embedded application in the SESAM-Vitale terminal for all accesses to the Vitale and CPS cards.

Middleware for the CPS card

The **API-CPS** offers proprietary verbs for executing basic functions by the CPS cards (each chaining one or more cards APDUs), such as : reading of HCP identity and professional quality, reading of X.509 certificates, providing an authentication value (RSA), calculation of an electronic signature (RSA), ...

The **PKCS#11-CPS** library allows « PKCS#11 standard aware applications » such as **Mozilla based browsers** to work with the CPS card.

Note 1 : For Windows, a **CSP-CPS** library exists (on top of the PKCS#11-CPS library) allowing « CSP aware applications » such as **Internet Explorer** to work with the CPS card.

Note 2 : For MacOS, a **TokenD-CDSA-CPS** library exists (on top of the PKCS#11-CPS library) allowing « CDSA aware applications » such as **Safari** to work with the CPS card.

Middleware for the Vitale card

The **API-SSV** offers verbs for executing application (macro) commands for the SESAM application in the S-V terminal, such as : reading of CPS and Vitale cards, signing reimbursement forms, ...

GALSS Manager : French resource manager for cards and readers

The GALSS Manager :

- ⇒ manages the sessions between applications and resources (cards and readers)
 - in principle, the number of readers is not limited,
 - the readers can be S-V terminal type readers or "single slot readers" ;
- ⇒ allows applications to take exclusive access to the resources ;
- ⇒ allows to exchange messages between the application and the local resources in the S-V terminal ;
- ⇒ manages the application contexts for each resource (card presence, ...). These context data are accessible by the applications (via the API-CPS, API-SSV or API-Vitale-Reading) ;

« **GALSS.ini** », the GALSS configuration file :

- ⇒ contains the logical names of all resources : cards (« CPS », « Vitale », ... with possible aliases) and readers/terminals - « GALSS aware applications » read this file in order to get the resource names which they need for the opening of sessions with the current middleware ;
- ⇒ contains information about the physical configuration (characteristics of each reader, associations of cards and slots, I/O port info, ...) allowing the **GALSS Manager and PSS** to manage those resources.
- ⇒ The **GALSS.ini** is a « static » file ; it is not updated when a reader is plugged/unplugged to/from the workstation (we currently do not use « Plug and Play » facilities).

PSS : Protocole Santé Social

This proprietary protocol is used for the communication between the workstation and the S-V terminals.

PSS is conceived for serial interfaces. In the case of an USB terminal, we use an OS native driver which emulates the serial interface.

The PSS protocol is « unidirectional » in the sense that only the workstation can take the initiative of an exchange.

Embedded application

The embedded application in the SESAM-Vitale terminal treats all the commands (card instruction or specific S-V terminal command) sent by the applications

Note : An application on the workstation can send commands to a reader which doesn't contain a card command such as : Get reader configuration, Update terminal software, ...

Workstations in hospitals

In this configuration the SESAM application is not used.

The current architecture allows also covering of workstations with PC/SC readers using a « gateway to PC/SC » instead of the « PSS protocol driver ».

In this configuration, the middleware (API-CPS and the API-Vitale-Reading) gains access to the cards Vitale and CPS cards through SESAM-Vitale terminal commands containing exclusively card instructions, which are transposed to classical card instructions in PC/SC format by the PC/SC gateway.

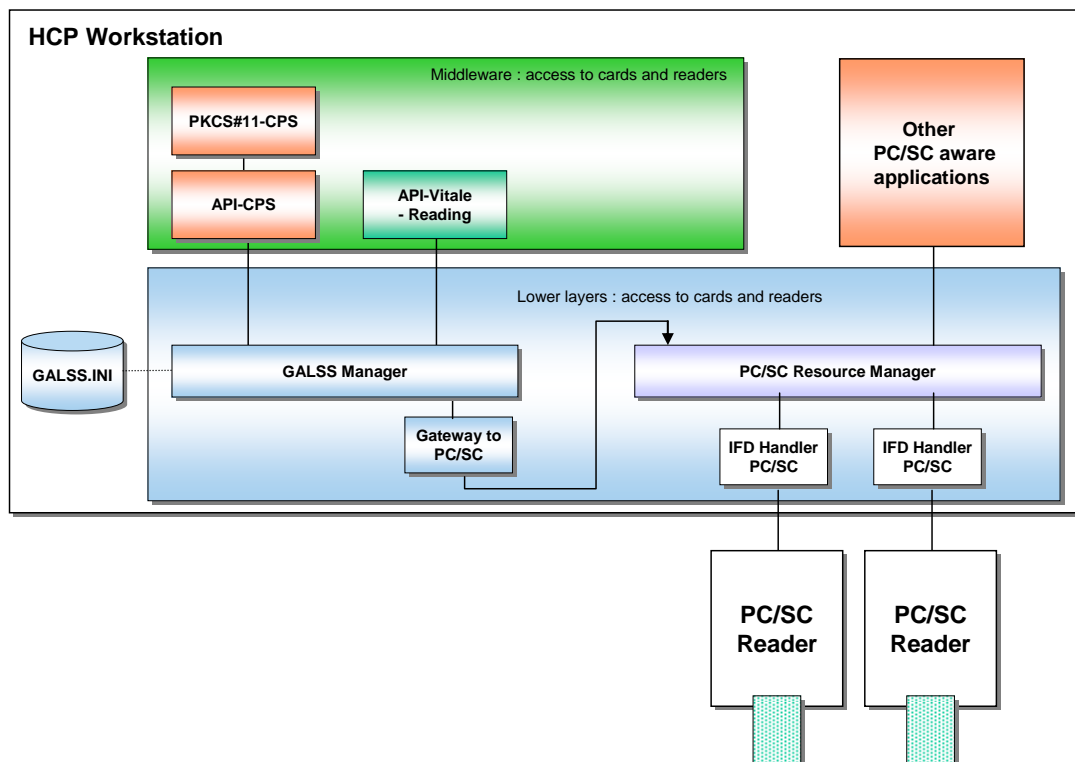


Diagram 2 : Workstation equipped with PC/SC readers (hospitals)

Middleware for the Vitale card

In this configuration, the medical and administrative applications have access to the Vitale card through a simplified library, called **API-Vitale-Reading**. It offers verbs for executing basic reading functions of data in the Vitale card (each chaining one or more cards APDUs).

Gateway to PC/SC

The **Gateway to PC/SC** allows the use of standard PC/SC readers for « GALSS aware applications ». This module replaces the **PSS protocol driver**, it converts the S-V terminal commands in calls to the standard PC/SC Resource Manager. Naturally, in this configuration, only S-V terminal commands containing cards instructions are accepted.

Operating Systems

The following Operating Systems are supported **in common** (they are the **target Operating Systems**) by the GIE SESAM-VITALE and the GIP-CPS :

- Windows (2000, XP, Vista),
- MacOS X,
- Linux,
- Unix SCO.

The GIP-CPS supports also the modules API-CPS, GALSS and the PSS protocol driver on the following Operating Systems :

- Unix AIX,
- SUN Solaris,
- HP-UX.

Virtual clients

The use of the GALSS on workstations with « virtual clients » creates a problem for mobile users because the GALSS module is part of the « user environment ». As a consequence, the user « transports » his physical reader configuration (described in GALSS.ini) from one workstation to another, meaning that, unless the physical configurations are totally identical, his application will not function any more on the new workstation.

This case is treated in § 4.6.6 : Workstations with « virtual clients ».

4 Call for Comments – Issues to be solved

4.1 Configurations to be covered by the future architecture

The future architecture must allow use of the CPS, Vitale, DUO or any other card which might be used in the French HC sector. It must work with independently developed applications which can be active simultaneously on a HCP workstation.

The following configurations have been identified :

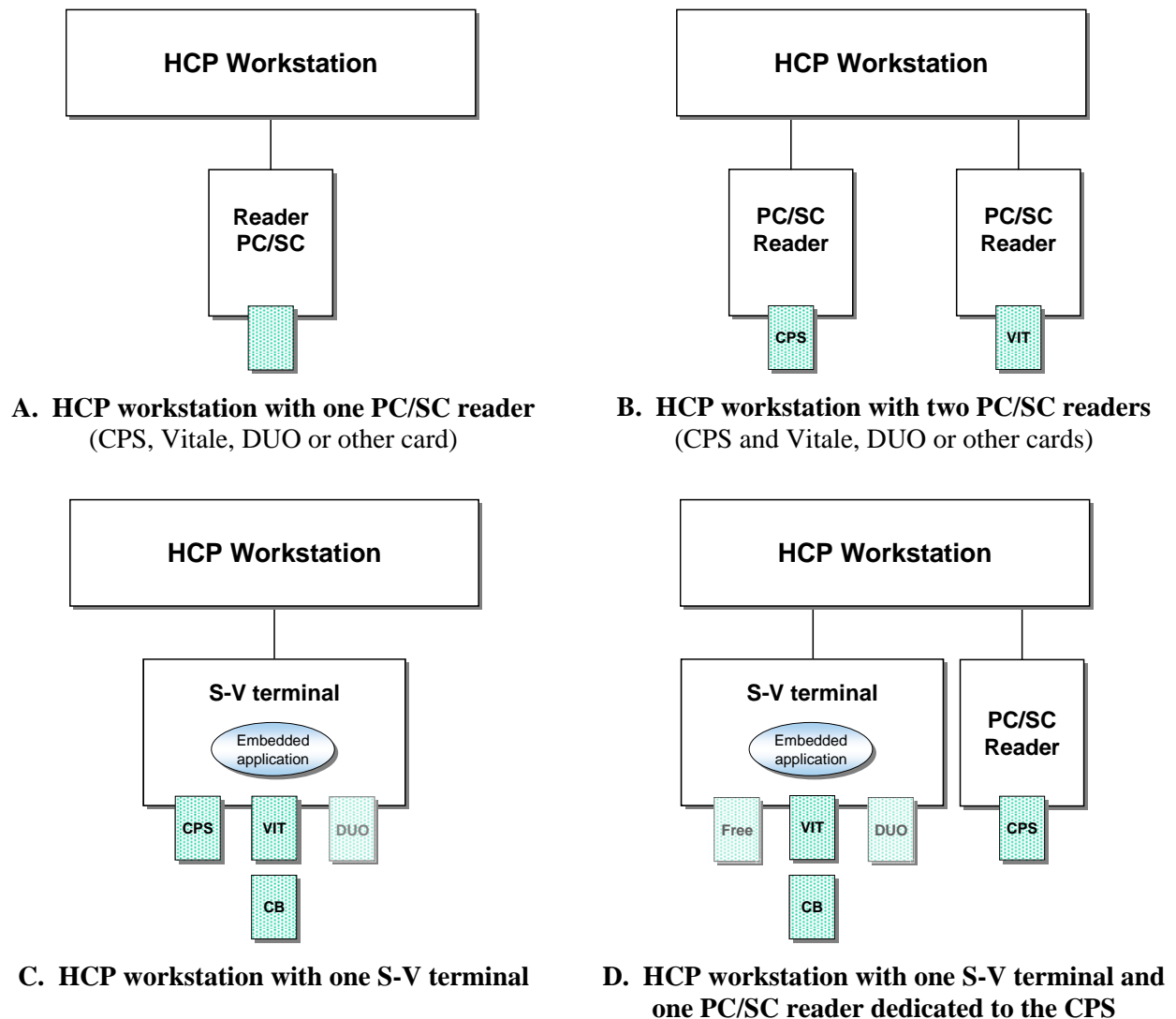


Diagram 3 : Configurations to be covered by the future architecture




1. Configurations **A** and **B** are primarily used in hospitals.
2. Configuration **C** corresponds to the current equipment of the HC Professionals in the private sector.

The S-V terminal can be « multi-application », it can also contain a stand-alone EFT/POS application. For the latter application, the payment card is introduced in the Vitale slot. During the payment transaction, the S-V terminal may ignore all commands coming from the workstation.

There even are S-V terminals with three slots allowing an application to communicate simultaneously with Vitale and DUO cards.

3. Configuration **D** is a hypothetic evolution, not programmed at the moment. Nevertheless, the future architecture must be able to deal with this configuration.

4.2 PC/SC standard and the target Operational Systems

 01	<p>As PC/SC version 2 is not available under Windows (not in the standard installation packages) our proposed architecture is based on PC/SC version 1.</p> <p>Can you confirm that all our target Operating Systems and the other supported Operating Systems support PC/SC version 1 ?</p> <p>The target Operating Systems are the following :</p> <ul style="list-style-type: none"> • Windows (2000 SP4, XP SP3, Vista SP1), • MacOS X (10.3 and 10.4 PPC, 10.4 Intel, 10.5), • Linux (kernel 2.4 and 2.6), • Unix SCO, • « desktop virtualization systems » (like Windows TSE, Citrix, Sun, ...). <p>The other supported Operating Systems are the following :</p> <ul style="list-style-type: none"> • Unix AIX, • SUN Solaris, • HP-UX.
 02	<p>Does PC/SC version 2 guarantee upward compatibility with PC/SC version 1 implementations (PC/SC aware applications and IFD drivers) ?</p>
 03	<p>What is the sustainability of PC/SC for each OS ?</p> <p>Namely, is there a risk that PC/SC would not be supported anymore in the future for some Operating Systems ?</p> <p>Will coming versions of the Operating Systems guarantee backward compatibility ?</p> <p>What are the precautions we will have to take in order to stay compatible with coming versions of the listed Operating Systems ?</p>

4.3 Configuration cases A and B

The future architecture must be able to deal with all described configurations in the preceding chapter.

Configuration A is « classical » for the PC/SC environment ; applications handling only one card with one reader on the HCP workstation will never have problems.

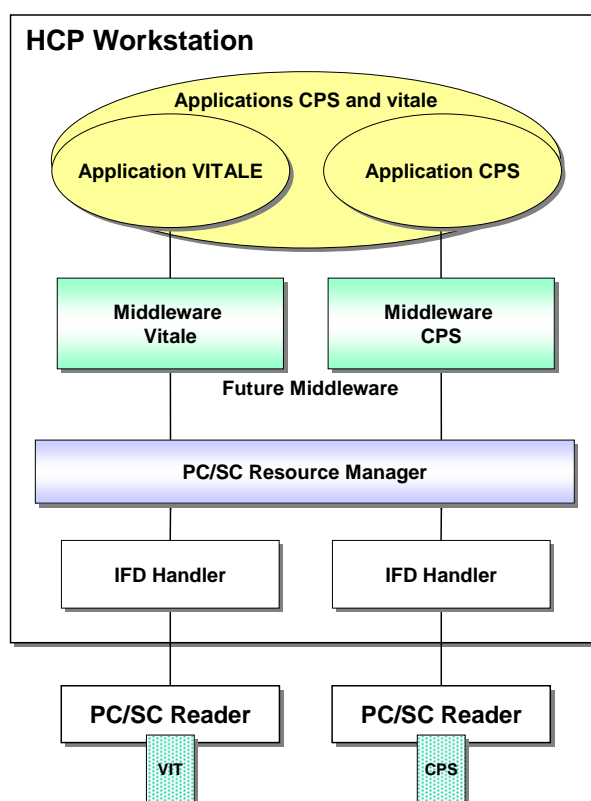


Diagram 4 : Configuration with two PC/SC readers

Configuration B with 2 or more PC/SC readers creates the issue of dynamically associating the various cards - CPS, Vitale, DUO, ... - to their corresponding readers because PC/SC does not offer a (static) configuration file like GALSS.ini which affects each card to a specific reader (or slot).

Most of the available PC/SC readers are "plug & play" (PnP). We can connect 2, or even 3, readers to one workstation which can all be identical (the CPS reader will be oriented to the HC Professional and the other readers will be directed to the patient).

Reminder :

- The CPS and the Vitale card don't have the same ATR.
But the Vitale and the DUO card do have the same ATR.
- There must never be access conflicts to cards between applications which co-exist on the same workstation and want to communicate with the same cards.

For PC/SC aware applications

How can an application manage the association of the readers with the different cards ; or expressed otherwise : how does an application know which reader has to be addressed for a particular card ?

How to treat the following examples (configuration with 2 PC/SC readers) :

- ⇒ The 2 readers are empty and the application asks the user to introduce the CPS card in the corresponding reader. But the CPS can be introduced into the other reader, or even another card can be introduced in any of the 2 readers ;
- ⇒ The reader dedicated to the CPS contains a CPS card, a second application wants to access to the CPS. How does the second application know that the CPS is already present ?
- ⇒ The reader dedicated to the CPS contains a CPS card, a second application wants to access to the Vitale card (or to a DUO card). How does the second application know that the introduced card (CPS) is not a Vitale card (or to a DUO card) and how does it detect that an introduced card in the second reader is the expected one ?



For other applications which use the middleware for the CPS and Vitale cards

Different cases have to be distinguished :

- ⇒ The applications which use a CSP (Windows Cryptographic Service Provider) shouldn't have any problem thanks to the use of Windows events.
Do you confirm this assertion ?
- ⇒ When an application built on PKCS#11 (dedicated either to the CPS card or to the Vitale card) wants to open a session with a « token », how can PKCS#11 associate this request with the correct (dedicated) reader ?
What if the corresponding reader does not yet contain a card ?
Does/can PKCS#11 have to act as a PC/SC aware application ?
- ⇒ For the cases where applications which are built on the API-CPS or the API-SSV, do these respective API have to act as a PC/SC aware applications ?
(Same issue as PKCS#11 ?)

4.4 Configuration cases C and D

The diagram hereunder shows the **Configuration C case** : the workstation is equipped with the current S-V terminal. On that workstation existing « GALSS aware applications » can coexist with newly constructed applications on the future middleware and with other « PC/SC aware applications ».

In fact, the diagram shows the migration from the current « GALSS aware applications » to the future « PC/SC aware applications ».

The issues are similar for the case D configuration.

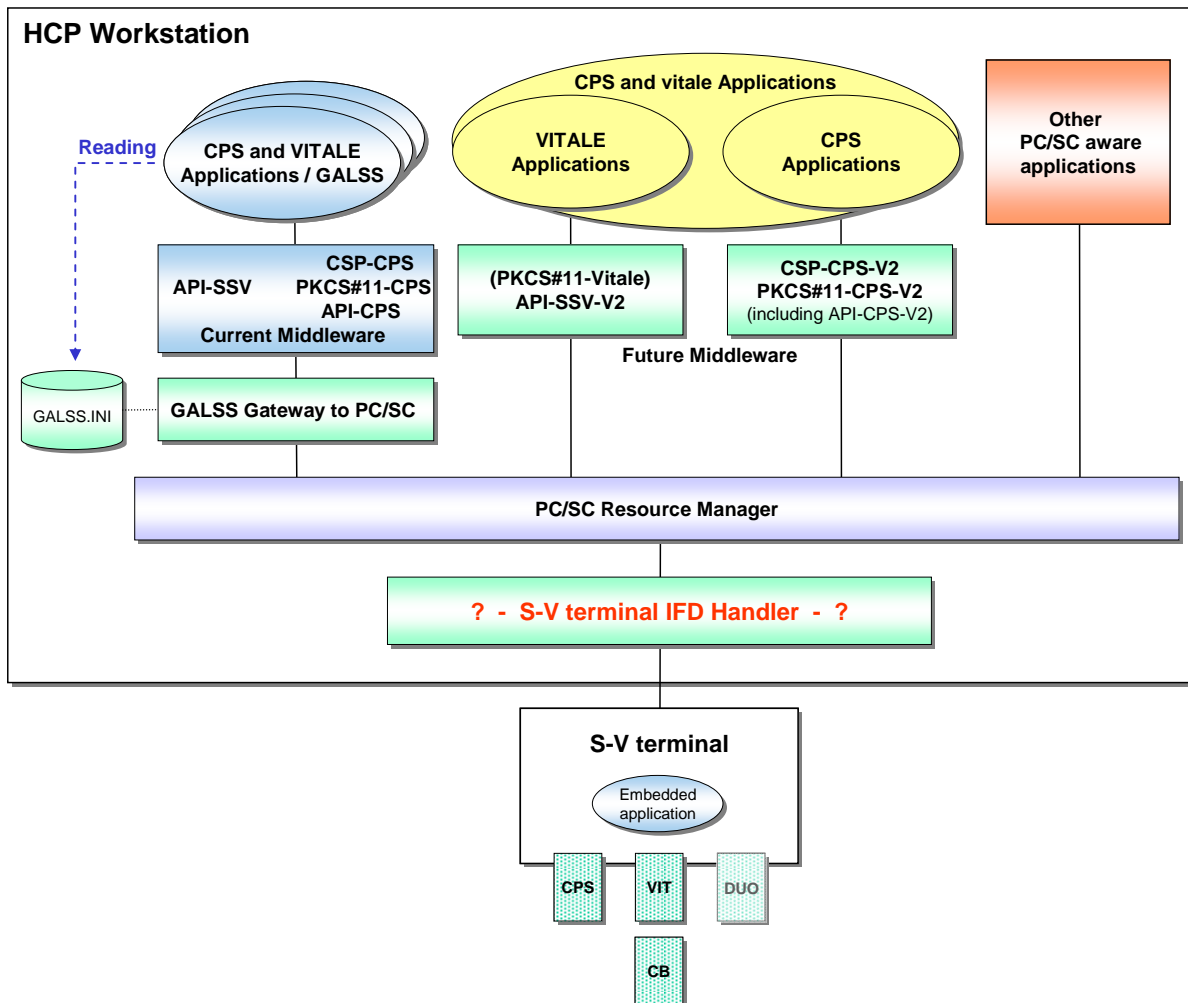


Diagram 5 : Configuration with one SESAM-Vitale terminal

In this configuration :

- The GALSS Manager is replaced by a « GALSS gateway to the PC/SC Resource Manager » which :
 - offers the current GALSS interface for the existing « GALSS aware applications » in order to guarantee upward compatibility ;
 - translates the GALSS commands to standard PC/SC commands.
- The SESAM-Vitale terminal is not managed any more by the GALSS but through one or several IFD Handlers.
(The way how to do that correctly is the main issue of this Call for Comments).
- The local resources of the SESAM-Vitale terminal (cards and slots) are seen by the applications as standard PC/SC readers by the PC/SC aware applications.

In summary :

In blue, the existing (current) middleware and applications :

- medical office software (in the private sector) integrating the SESAM-Vitale reimbursement application,
- the on-line services of the public and private healthcare-insurances,
- other CPS software : secure messaging and medical applications such as access to central patient file servers, ...

In green, the modules to be developed :

- A « GALSS gateway to PC/SC » which communicates directly with the PC/SC Resource Manager,
The « GALSS.ini » configuration file is only maintained in the future architecture for allowing existing « GALSS aware applications » to read the logical names - and their aliases - of the resources of the workstation.
The other information in the GALSS.ini becomes meaningless and can probably be omitted.
- The future middleware of the CPS and Vitale cards will be capable to call directly the PC/SC Resource Manager.
- **An IFD-Handler for the S-V terminal.**

In yellow, the future healthcare applications using the future middleware.

Note : Industrial software products such as internet navigators and SSO products will be able to work indifferently with the current crypto libraries as with the future ones.

Those products know the library to be used via their configuration parameters :

- ⇒ In general, the PKCS#11 libraries are declared as drivers for "security devices" (Mozilla based browsers, ...);
- ⇒ The key-store of Internet Explorer contains for each certificate which has an associated private key the name of the CSP DLL to be used.

The main issue to solve for the future architecture is the functioning of the IFD Handler that has to be developed for the current SESAM-Vitale terminal.

4.5 Constraints to be respected

The future architecture MUST respect the following constraints :

C-1	The future architecture MUST work with the current SESAM-Vitale terminals in the field (250.000 installed !) using the PSS protocol and their proprietary commands.
C-2	The « GALSS gateway to PC/SC » MUST offer an upward compatibility with the current installed applications (constructed on the GALSS and using the GALSS.ini configuration file) in order to allow a « gradual » migration to the future architecture.

4.6 Presentation of the proposed architecture for Phase 1

The hereafter proposed architecture corresponds with **Phase 1 - short term**, « **the architecture MUST be compatible with the current SESAM-Vitale terminals** » as mentioned in the § 1 : Introduction, allowing applications to benefit rapidly from the PC/SC architecture without modifying the software of the S-V terminals.

4.6.1 Hypothesizes

The future architecture will be built on PC/SC version 1, because it is available on all Operating Systems (the PC/SC version 2 offers upward compatibility). PC/SC version 1 is primarily designed for simple readers offering only 1 card slot.

The possibility for vendors to define their own features through the standard PC/SC command « **RESPONSECODE Control ()** » (cf. § 4.8 Annex 1 : Extracts of PC/SC V1 & V2 specifications) allows applications to communicate directly with PC/SC readers on a transparent way for the PC/SC Resource Manager.

This seems to be a proper way to send the SESAM-Vitale terminal commands (containing card instructions or terminal application commands) to the embedded application.

So, the implementation of the proposed architecture is constructed using the following hypothesizes :

H-1	<p>The SESAM-Vitale terminal is seen by the PC/SC Resource Manager as N independent PC/SC readers, N representing the number of slots (2 or 3) offered by the terminal.</p> <p>Each (virtual) reader will be controlled by 1 instance of a unique SESAM-Vitale specific IFD-Handler.</p> <p>When detecting a SESAM-Vital terminal a « BUS Driver » will load dynamically the corresponding instances respecting the Plug and Play rules.</p> <p>The BUS Driver will give "friendly names" to the IFD-Handlers following a predefined naming scheme :</p> <ul style="list-style-type: none"> • SVreader_CPS.x for FU=1, • SVreader_Vitale.x for FU=2, • SVreader_Card3.x for FU=3 (if a 3rd slot is present), • « x » being the index 1, 2, of each connected S-V terminal allowing thus more than 1 S-V terminal to be connected to a HCP workstation). <p>The BUS Driver will also dynamically update system configuration files with the reader information (such as the Windows registry).</p> <p>This hypothesis allows « PC/SC aware applications » to send card instructions to the correct slot without knowing the S-V terminal configuration.</p>
H-2	<p>In order to communicate with the SESAM-Vitale terminal, a S-V-specific feature (for card instructions and application commands) will be specified. The SESAM-Vitale terminal commands will be mapped on the InBuffer of the RESPONSECODE Control () command. The terminal will return the result of the treatment using the OutBuffer.</p> <p>This hypothesis allows applications to send application commands to the embedded application.</p> <p>By convention, application commands used for the electronic reimbursement forms will only be addressed to the Vitale card slot (<i>this can be imposed to the future applications through « best practices »</i>).</p> <p>This approach allows also to seize pincodes locally on the S-V terminal - instead of seizing them on the keyboard of the HCP workstation and sending them in clear to the corresponding card (<i>security issue</i>).</p>

4.6.2 Architecture of the proposed architecture

The proposed architecture is based on the use of one IFD-Handler for each slot in the S-V terminal (= virtual PC/SC reader).

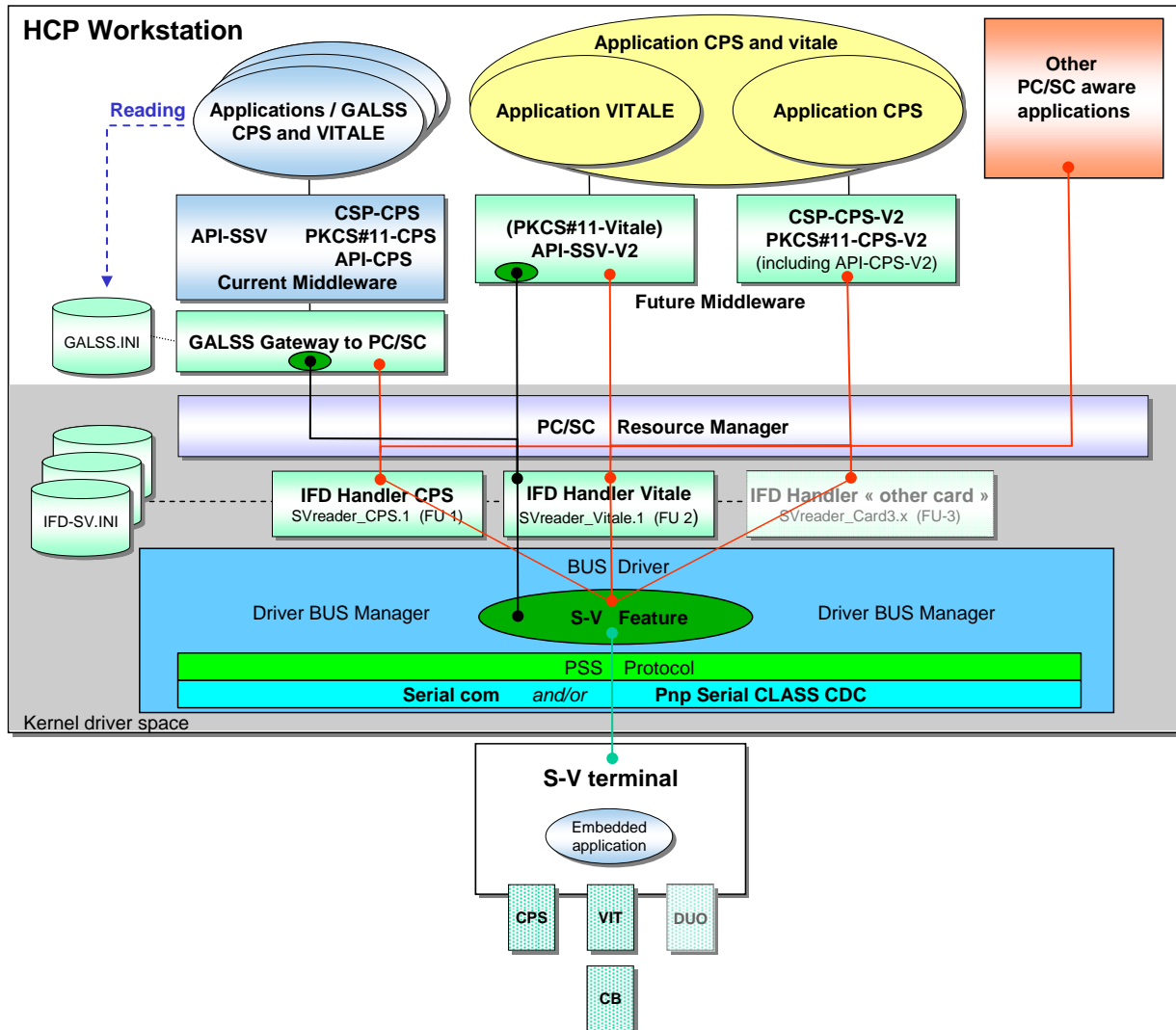


Diagram 6 : The proposed architecture for Phase 1

The links between the different modules are colored according to the type of the exchanged data :

- Red** : standard PC/SC command **not** using the S-V-specific feature,
- Black** : standard PC/SC commands using the S-V-specific feature, *they use the same logical channels as the Red links,*
- Green** : S-V terminal command containing either a card instruction or an application command for the embedded application. The current PSS protocol is used.

4.6.3 Principles of functioning

Current middleware

The current middleware calls the GALSS « as usual » (upward compatibility) in order to send commands to a S-V terminal or a standard PC/SC reader. It is the « GALSS gateway to PC/SC » who will treat the call instead of the current GALSS.

Before the first call the application will have to open a session indicating the name of the card to be addressed.

Note : The configuration file GALSS.ini might be consulted at this occasion.

GALSS gateway to PC/SC

The gateway will receive the call from the current middleware and will treat it in function of the addressed reader (look-up in the configuration file GALSS.ini) :

- **If the addressed reader is a standard PC/SC reader** and if the command contains a card instruction, it will translate it in a standard PC/SC reader command and send it to the PC/SC Resource Manager. The answer will be re-translated to a « GALSS-answer » and returned to the calling application.

If the command contains a S-V terminal command for the embedded application, it will be refused and an error code will be returned to the calling application.

- **If the addressed reader is a S-V terminal**, the gateway will use the S-V-specific feature in order to send the command to the corresponding Vitale reader. The answer will be translated to a « GALSS-answer » and returned to the calling application.

Future Vitale middleware

The future Vitale middleware can generate 2 types of calls :

- either standard PC/SC calls (no use of the S-V-specific feature),
- or calls using the S-V-specific feature when it wants to send an application command to the S-V terminal.

Future CPS middleware

The future CPS middleware will only generate standard PC/SC calls (no use of the S-V-specific feature).

IFD-Handlers for CPS and Vital cards

The specific IFD S-V handlers (multiple instances of one single IFD-handler) will receive standard PC/SC calls using possibly the S-V-specific feature. It will send it to the BUS Driver without any particular treatment.

As defined in hypothesis 1 (H-1), each (virtual) reader has a "friendly name" referencing a specific card slot in a physical S-V terminal (SVreader_CPS.1, SVreader_Vitale.1, ...).

BUS Driver

The BUS Driver is loaded during system start up. At the detection of one or more S-V terminals, it will load dynamically the specific IFD S-V handlers (multiple instances of one single IFD-handler respecting the above described naming scheme). When a S-V terminal is unplugged, the BUS driver will kill the corresponding IFD-handlers.

When the BUS Driver receives a call from an IFD-handler it will treat it in function of the addressed reader (it knows all IFD-Handlers corresponding to connected S-V terminals because it did load them itself using the predetermined naming scheme) :

- **If the addressed reader is a standard PC/SC reader** and if the call contains a card instruction, it will send it to the PC/SC reader without any particular treatment. The answer will be returned to the calling application.

If the command contains a S-V terminal command (use of S-V-specific feature), it will be refused and an error code will be returned to the calling application.

- **If the addressed reader is a S-V terminal**, the BUS Driver will translate the call in a S-V terminal command and send it to the corresponding terminal. The answer will be retranslated to a « PC/SC-answer » and returned to the calling application.

The BUS Driver will communicate via :

- the Serial com for serial S-V readers,
- the PnP Serial CLASS CDC for S-V readers with a USB interface.

In order to avoid two versions of the BUS Driver it can be worth studying the feasibility of developing a single BUS Driver able to handle both cases.

Observations :



1. What if the BUS Driver receives simultaneously calls for different slots corresponding to the same S-V terminal ?









Because the management of exclusive access is the responsibility of the applications (or the middleware of the CPS and Vitale cards) such a situation can only occur when 2 different applications obtained exclusive access to 2 different cards. So, in that case, the BUS Driver can « serialize » the commands respecting the following rules :

- a command can only be sent after having received the answer of a previous command,
- the FIFO principle (First In First Out) has to be respected.

(This situation will never occur when a call is sent to the embedded application in the S-V terminal ; the calling application must first obtain the exclusive access to all slots of the addressed S-V terminal.)

2. By polling the S-V terminals in idle time (with a frequency to be fixed), each IFD-Handler corresponding to a slot of a S-V terminal can generate standard PC/SC events corresponding to the introduction or withdrawal of cards.

 05	Does the proposed architecture and its functioning seem realistic to you ?
 06	Can the proposed architecture function in all mentioned Operating Systems ? If you are an OS editor, how should the architecture be implemented in your particular environment ?

 07	<p>Can the fact that we use a BUS Driver and « virtual slots » generate problems under Windows or the other target Operating Systems ?</p>
 08	<p>Does the described management of card events (introduction and withdrawal) seem correct ? Could we have performance problems ? Do you have other suggestions ?</p>
 09	<p>With Windows, the BUS Driver has to update dynamically the Windows registry with the reader configurations. Can you confirm that assertion ?</p> <p>How can the BUS Driver update the system data with the reader configurations for the following Operating Systems ?</p> <ul style="list-style-type: none"> ⇒ Windows TSE ? ⇒ MacOS ? ⇒ How does it work on workstations with Linux ? ⇒ How does it work on workstations with the other supported Operating Systems ?
 10	<p><i>The volume of the proprietary application commands of the S-V terminal can represent several dozens of Kbytes (input and output).</i></p> <p>We heard that the buffer sizes must not be over 64 Kbytes. What are the maximum buffer limits of vendor-defined features for each of the target Operating Systems ?</p>
 11	<p><i>The proposed architecture uses 1 IFD-Handler for each slot of a S-V terminal.</i></p> <p>Do you think it is possible, or preferable, to use a single IFD-Handler for each S-V terminal ?</p>
 12	<p><i>The current GALSS aware applications and the current middleware use the GALSS.ini configuration file.</i></p> <p>Can we work with a fixed installed GALSS.ini file, configured during application installation, containing all possible resources names ?</p>
 13	<p>Can a IFD-Handler, loaded in the « Kernel driver space », communicate through the standard communication drivers of the target Operating Systems and so allow the use of « IP readers » remotely connected (via Ethernet, Wifi, Bluetooth,...) to a workstation ?</p>
 14	<p>What happens when, in the case of a reader configuration error, a S-V-specific feature is sent to a standard PC/SC reader ? Will it be refused and will the reader send an error code to the calling application and so not disturb the PC/SC Resource Manager ?</p>

Management of exclusive access

Rules for the management of exclusive access

- Each application, or middleware, is responsible for obtaining exclusive access to the resources with which it wishes to communicate ;
- Each application, or middleware, opens a connection to the card located in the reader identified by the *ReaderName* parameter (PC/SC command RESPONSECODE Connect ()). The connection is always opened in « shared access mode ».
- When an application wants to communicate with a card, an exclusive access is obtained by the PC/SC command RESPONSECODE BeginTransaction (). When having exclusive access to a card, an application must limit itself to a simple and short logical transaction, for example :
 - selection an application,
 - selection a signature key (set security context),
 - send data to be signed,
 - perform security operation,
 - get result = signature,
- The application, or middleware, MUST release the exclusive access immediately at the end of each logical transaction by using the PC/SC command RESPONSECODE EndTransaction ().
- When obtaining exclusive access to a card, an application, or the middleware, must check the card state - another application might have changed the card state during a preceding exchange ;

The current CPS and Vitale Middleware (API-SSV and API-CPS) respect equivalent rules. Actually, the GALSS allows to get exclusive access to one or more cards or even to all the resources of the S-V terminal ; in fact, when the Vitale middleware needs access to a S-V terminal it takes systematically the exclusive access to all its resources.

In the PC/SC architecture it is impossible to obtain exclusive access to more than one slot with a single command. An application which wants to use the S-V-specific feature has to obtain the exclusive access for each slot by executing one separate command for each slot (SVreader_????.x, « x » being the index 1, 2, ...) of the corresponding S-V terminal. The sequence of those requests has to respect additional rules in order to avoid « deadlock »³.

As a conclusion, the translation of the management of exclusive access with the current GALSS to the future architecture should not be an issue for harmonious coexistence of applications on a workstation with the proposed architecture.

³ Proposal for the management of multiple requests for exclusive accesses.

When an application, or middleware, would like to obtain exclusive access to several resources, it starts asking exclusive access to the first resource. When obtained, it can ask exclusive access to the second resource and so on until it obtains the exclusive access to all required resources ;

In case of a refusal, the application MUST release all obtained exclusive accesses starting by the last one obtained until the first one. Then, the application waits for a delay (to be fixed, eg. 100 ms) after which it can start trying to get again exclusive access to all required resources.

In case of a number of unsuccessful trials without success (to be fixed, eg. 3 trails), the application will abandon and inform the calling application which, on his turn can inform the user about the unsuccessful ending.

4.6.4 Updating of a SESAM-Vitale terminal

Most of the SESAM-Vitale terminals allow the updating of the terminal software through a specific program on the HC Professional workstation.



Does the proposed architecture allow the updating of the software in a S-V terminal knowing that its size can be over 1 Mbytes ?

What would be the expected time needed to download a software update of 1 Mbytes (without counting the local treatment in the S-V terminal) ?

4.6.5 Functioning of autonomous S-V terminals

Some S-V terminals can function in 2 modes :

- ⇒ as a standard S-V terminal connected to the HCP workstation,
- ⇒ as an autonomous terminal ; HC Professional uses his terminal during his visits of his patients and, when back in his office, reconnects it to his workstation in order to synchronize their contents.



What are your recommendations for the management of frequent plugging and unplugging of a S-V terminal in order to avoid disturbing the PC/SC Resource Manager and the active applications ?

What about the "plug and play" facilities which reconfigure dynamically the configuration data ?

4.6.6 Workstations with « virtual clients »

A large number of hospitals use « virtual clients » on their workstations based on solutions like Windows TSE, Citrix, Sun, ...

In that case, a HC professional can open a session on one workstation, suspend his session and resume it later on another workstation (fortunately patients don't open sessions ...).



Can those environments handle the configurations B, C and D (*Configurations with more than 1 reader connected to each workstation !*) in the proposed architecture ?

How does the system know which reader should be used for the session management ?

Use of the first one in which a card is introduced through the detection of PC/SC events ?

What if the configurations of the workstations in a hospital have different configurations ?

⇒ How does it work with Windows TSE ?

⇒ How does it work on workstations with MacOS ?

⇒ How does it work on workstations with Linux ?

⇒ How does it work on workstations with the other supported Operating Systems ?

How does the system know that a user "switched" from one workstation to another in the middle of a session ? Does it inform the running applications ?

How does an application, which opened sessions with cards on one workstation, manage the fact that a user can "switch" to another workstation with other local readers ?

Are there any limitations / recommendations and, if so, what are your suggestions to deal with them ?

4.7 Architecture with modified S-V terminals - Phase 2

The hereafter proposed architecture corresponds to **Phase 2 - medium term**, mentioned in the § 1 : Introduction, offering a more conventional (and elegant) integration in the PS/SC architecture.

In this architecture, the S-V terminal with USB interface acts like a (virtual) Hub with 2 or 3 PC/SC readers connected to it which can be piloted through the generic USB CCID (Chip/Smart Card Interface Devices) driver. The naming scheme specified for Phase 1 will be respected offering thus a transparent migration from the Phase 1 to the Phase 2 architecture for all applications.

The virtual Hub inside the S-V terminal allows to benefit fully from the Plug and Play facilities and still allows the use of the S-V-specific feature.

For S-V terminals with other interfaces - serial, IP connections, ... - the CCID driver will have to be adapted by the S-V terminal industrial.

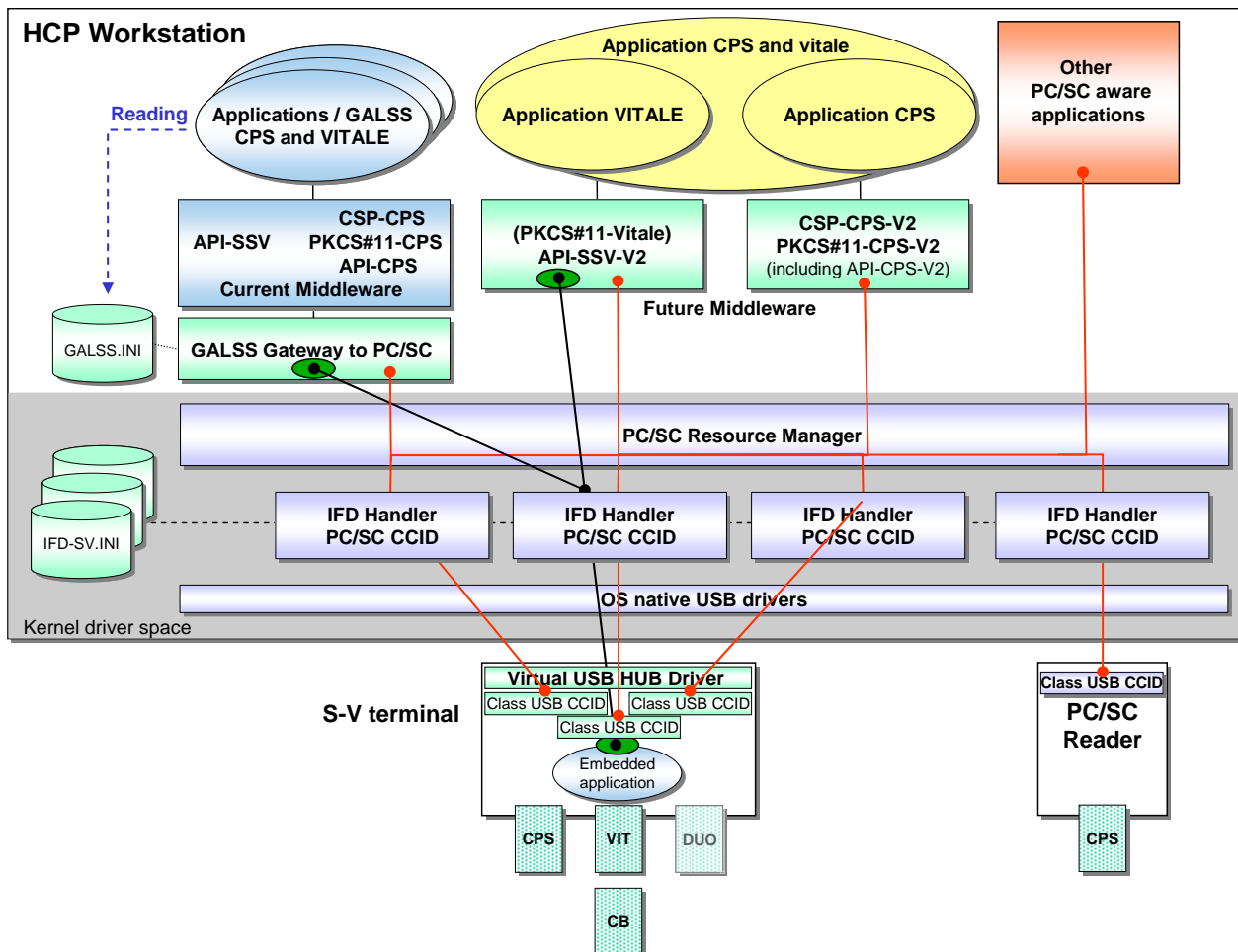


Diagram 7 : The proposed architecture for Phase 2



Does this « Phase 2 architecture » and its functioning seem realistic to you ?

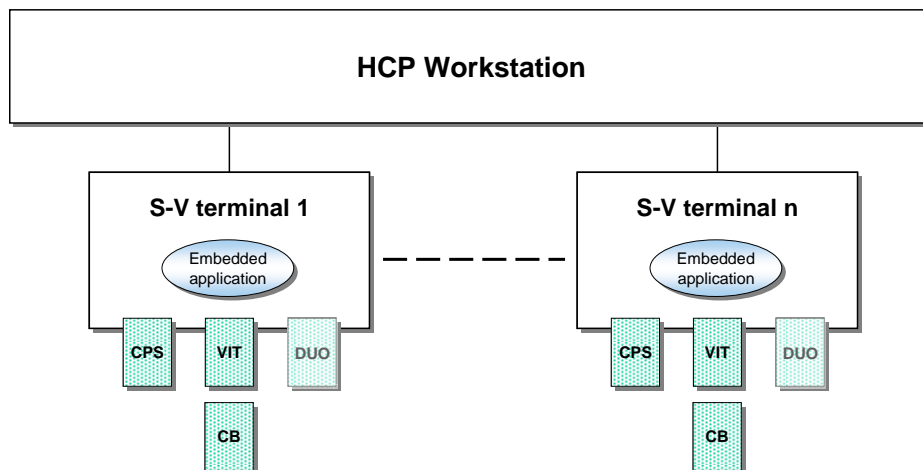
Is it compatible with PC/SC versions 1 and 2 ?

Are there any limitations / recommendations and, if so, what are your suggestions to deal with them ?

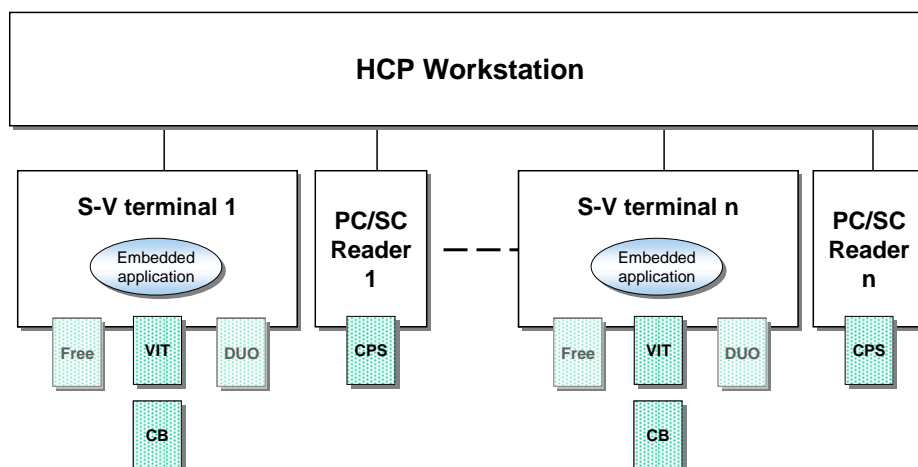
4.8 Multiple readers on a workstation

In the private sector there exist workstations with more than one S-V terminal connected to them. These "heavy" configurations (Windows NT, Linux, SCO, ...) can be mainly found in pharmacies and laboratories.

This gives us 2 additional configurations to be covered :



E. HCP workstation with several S-V terminals





F. HCP workstation with several S-V terminals and as many PC/SC readers dedicated to CPS cards

Diagram 8 : Configurations with multiple S-V terminals



These configurations should not generate additional issues as both phases use a dynamic naming scheme allowing multiple S-V readers to be identified on a HCP workstation :

- SVreader_CPS.1 + SVreader_Vitale.1 + SVreader_Card3.1 for the first detected S-V terminal,
- SVreader_CPS.2 + SVreader_Vitale.2 + SVreader_Card3.2 for the second detected S-V terminal,
- and so on ...

The management of exclusive access won't be impacted either because each process on the workstation can clearly identify the IFD-Handlers for each S-V terminal.

 19	<p>Can these configurations be handled by the « Phase 1 and 2 architectures » ?</p> <p>Is it compatible with PC/SC versions 1 and 2 ?</p> <p>Are there any limitations / recommendations and, if so, what are your suggestions to deal with them ?</p>
 20	<p>What is the maximum number of readers the PC/SC manager can deal with for the following Operating Systems :</p> <ul style="list-style-type: none">⇒ Windows ?⇒ Linux ?⇒ SCO ?⇒ Other supported Operating Systems ?

5 Implementation examples

 21	Can you provide us with examples of source codes handling more than one card ?
 22	<p>Can you provide us with more precise guidelines concerning the following PKCS#5 chapters :</p> <ul style="list-style-type: none">⇒ § 2.3 User Interface Elements ?⇒ § 2.4 Installation and Configuration ?⇒ § 2.5 Runtime Considerations ?⇒ How does it work on workstations with Linux ? <p>How do we have to decline those guidelines for the following Operating Systems ?</p> <ul style="list-style-type: none">⇒ Windows / Windows TSE ?⇒ MacOS ?⇒ Linux ?⇒ The other supported Operating Systems ?

6 Annex 1 : Extracts of PC/SC V1 & V2 specifications

6.1 Vendor-defines features

PC/SC specifications Part 5 « ICC Resource Manager Definition » contain in § 3.2.5.2 « Methods » the definition of the command **RESPONSECODE Control ()** allowing applications to directly communicate with a PC/SC reader (same chapters for both PC/SC versions).

RESPONSECODE Control(

```
IN      DWORD ControlCode    // Vendor-defined control code
IN      BYTE[] InBuffer      // Input data buffer
IN OUT  BYTE[] OutBuffer     // Output data buffer
OUT     DWORD OutBufferLength // Length of data in output data buffer
)
```

This method supports direct communication with the Reader device. Its primary intent is to provide a mechanism to communicate with vendor-defined features. It is the responsibility of the vendor to define *ControlCode* values and input/output data associated with these features.

The *ControlCode*, *InBuffer* data and *OutBuffer* are sent directly to the reader device driver. The response from the device driver is returned in *OutBuffer* and the valid data size indicated in *OutBufferLength*.