GIR Titan-Hykkoris

Reference manual

Version 1.0-3, july 2008

Contents

Introduction

GIR Titan-Hykkoris is a fuel delivery management software.

It is meant to be used with fuel delivery terminals, and offers the following features:

- Define a list of vehicles (and optionally drivers) that are allowed to take fuel.
- Synchronize data with terminals using a network, serial or modem connection.
- Retrieve transactions from terminals.
- Consult the transactions history, generate reports to follow vehicles consumptions.
- Export transactions to files, to reprocess them in third-party applications.
- Import transactions from files generated by third-party applications.

GIR Titan-Hykkoris is a web-oriented application. It is installed on a single computer (the server), and can be accessed from any other computer using a web browser.

Terminology

- GIR Titan-Hykkoris server: The computer where the application is installed.
- Terminal: An equipment that performs vehicles/drivers identification, controls the fuel delivery, and stores transactions. All terminals are connected to the GIR Titan-Hykkoris server.
- Key: An item used to identify vehicles or drivers on the terminal. It can use various technologies (RFID, smartcard, ...).
- Transaction: A collection of data related to a fuel delivery (date, time, vehicle, volume, ...).
- User: A person that connects to the application.
- Driver: A person that uses the terminal to take fuel.

Chapter 1

Getting started

1.1 Installation

GIR Titan-Hykkoris comes with a Windows installer. Start the installation program, select the installation directory, and follow the instructions on screen to finish the installation.

When GIR Titan-Hykkoris is running, it shows an icon in the task tray, next to the clock. Double-click the icon to open a web browser on the login page.

GIR Titan-Hykkoris icon in the task tray

1.1.1 First login

On the first time you start the application, you will be asked if you want to create a new database. Confirm the operation, and wait a few seconds for the database to be created. When this is done, click Connect to display the home page. The application is now ready to use.

1.1.2 Installation directory

GIR Titan-Hykkoris uses only the files in its installation directory. It must have full permissions on this directory and subdirectories. GIR Titan-Hykkoris doesn't use the registry or shared libraries.

The installation directory contains the following files:

- `hl.exe`: The main program. It should always be started using the loader.
- `winhl.exe`: Windows loader, visible in the task tray.
- `winhl.ini`: Configuration file. It defines the TCP ports used by the application.
- `tables` subdirectory: The database
- `temp` subdirectory: Temporary files

- `output` subdirectory: Exported files
- `proxi` subdirectory: EMG keys definition files (See 9.3.1, page 70).

1.1.3 TCP ports

GIR Titan-Hykkoris needs two TCP ports to operate correctly. By default, these ports are 4747 and 8080.

- 8080: Web server port, used by web browsers to establish connections.
- 4747: Port used internally by the application, bound to the local network interface.

If you get an error message "Another application seems to use this port" when starting GIR Titan-Hykkoris, edit the ports value in the `winhl.ini` file.

1.1.4 Application update

To update the application to a newer version, run the installation program with the same installation directory. The program files will be updated, and the existing database will be kept.

A database conversion may occur on the first login that follows an application update. This simply means that the database structure is being updated to support new features. Just wait for the conversion to complete, and click Connect.

1.2 Connection

1.2.1 Local access

Double-click the GIR Titan-Hykkoris icon in the task tray to open a web browser. The page displayed in the browser depends on the configuration of the default user account:

- If the default user account has no password, a session is automatically opened and the home page is displayed. This is meant for use in a single-user environment.
- If a password has been set for the default user account, the login page is displayed. This is meant for use in a multi-user environment.

1.2.2 Remote access

Enter the following URL in the web browser navigation bar:

`http://<server>:8080`

where `<server>` is the hostname or IP address of the GIR Titan-Hykkoris server.

Upon successful connection, the login page will be displayed in the web browser. You can bookmark the above URL for easier access to the application from remote computers.

1.2.3 Login page

The login page prompts for a user name and a password, to connect to the application.

The default user account is `admin` with no password. Additional accounts can be defined, allowing multiple users to use the application simultaneously (See 2.4, page 25).

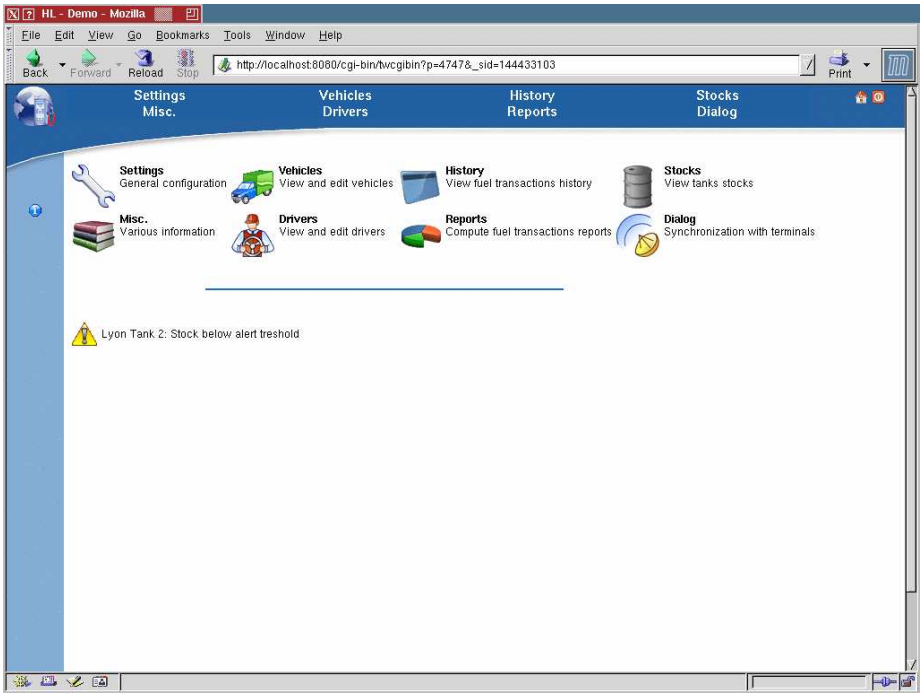
Accounts with an empty password are only allowed to login locally.

To close a session and go back to the login page, click on the Logout icon () at the top right corner of the screen.

1.3 Home page

The home page is the first page displayed when a user logs in. It contains two parts: the main menu and the dashboard.

At any time, the home page can be accessed by clicking on the Home icon () in the top right corner of the screen, or on the application logo in the top left corner of the screen.



1.3.1 Main menu

The main menu is available at any time in the upper part of the screen. It is also displayed on the home page. It contains the following entries:

- ▷ Settings: Shows a submenu related to the application configuration. This menu will typically be used a lot during the initial setup, then occasionally afterwards.
  - Config.: Global configuration
  - Products, Sites and Terminals: Terminals configuration
  - Users: Users accounts
  - Departments, Vehicles types, Drivers types, Providers: Assignments used to classify data. They are all optional.
- ▷ Misc.: Shows a submenu related to ordinary operations.
  - Events: Events retrieved from terminals or generated by the application.
  - Audit trail: History of all user-performed actions that changed the database.
  - Backup: Database backup tool.
  - Purge: Database purge tool.
  - Recompute: Recompute fields in fuel transactions.
  - HLF export: Export transactions data in a file, to process it in another application.
  - Export AEAT, Export MR4G, . . . : Other export formats.
  - Transactions import: Import transactions from files generated by a third-party application.
  - Supplies: Tank supplies history
- ▷ Vehicles: Shows the vehicles list.
- ▷ Drivers: Shows the drivers list.
- ▷ History: Shows an history of all fuel transactions.
- ▷ Reports: Computes reports on transactions.
- ▷ Stocks: Shows tanks stocks and terminals status.
- ▷ Dialog: Retrieves transactions and update data on terminals.

1.3.2 Dashboard

The dashboard is a list of important messages about the application status. The following messages can appear on it:

- License about to expire
- Terminal link error
- Pump blocked or in manual mode
- Tank stock below the alert treshold
- Expired maintenances

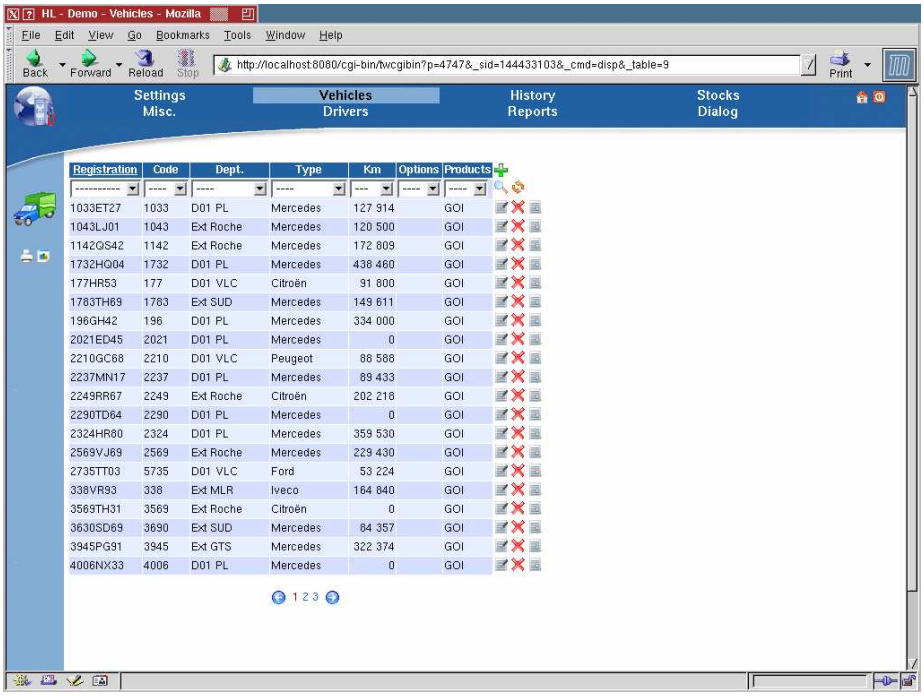
When one of those messages is displayed, the Alert icon () shows up at the top right corner of the screen. Click it to display the dashboard.

When there is nothing special to report, the dashboard is empty, and the home page only displays the main menu.

1.4 User interface

Most pages displayed by the application are about data consultation or modification. For instance, in the Vehicles menu, you can:

- View the vehicles list
- Create new vehicles
- Edit existing vehicles
- Search for a specific vehicle



Actions are performed by clicking on the corresponding icon. The most common icons are listed below.

- Icons available in the left panel:

	Printable version: Display a printable version of the current page.
	CSV file: Download data on the current page as a CSV file.
	Reload: Reload the current page.

- Icons available in a list header:

	Add: Add a new entry.
	Filter: Apply the current filter.
	Remove filter: Clear all active filters.

Note: the Filter button submits the form with the multiple filtering dropdown lists. On web browsers that support javascript, this is done automatically when a list value is changed. The Filter button is only provided for compatibility with other browsers.

- Icons available for each row in a list:

	Modify: Modify an existing entry.
	Delete: Delete an existing entry.
	Detail: View details about the selected entry. (ex: transactions made by a vehicle).
	Sheet access: For some tables, clicking on the element name (ex: Registration in vehicles) shows a detailled sheet on the selected element.

1.4.1 Paging

To limit the size of the generated web pages, the number of entries on a single page is limited, generally to 20 elements. When the total number of entries is greater than this, a navigation row appears at the bottom of the page.

The current page number is highlighted. Click on a page number to view its content, or use the Previous page and Next page buttons to browse the data.

1.4.2 Filtering and sorting

When consulting data, for instance the transactions history, you can perform various queries to get the information you're looking for.

A query is made by combining one or more filtering or sorting criteria.

- To sort data on a specific field, click on the column header. The current sort field appears underlined. Clicking a second time on the current sort field reverses the sort order.

- To filter data on a specific field, select a value in the dropdown list below the column header. Most common values are directly available in the list.

  To filter on a specific value, click the >>> item. This will display a new screen where you can enter detailed information about the filter you want. Validate to apply the filter and go back to the list.

  Click on Remove filter to clear all filter criteria and see all stored entries.

Example

Assume that we want to know what was the largest fuel delivery for the vehicle *1234ZZ69* during September 2007. This can be done by following the steps below:

- Go to the History menu. The page lists all fuel transactions.
- Select *Sep 2007* in the dropdown list under the Time column. The page now only lists fuel transactions made during September 2007.
- Select *1234ZZ69* in the dropdown list under the Vehicle column. The page now only lists fuel transactions made by the vehicle *1234ZZ69* during September 2007.
- Click on the L column header. Transactions are now ordered by increasing volume. Click on L again to order by decreasing volume. The first transaction in the list is the information we were looking for.

1.4.3 Units switching

In the English version of the application, all pages that display a vehicle meter can use either kilometers or miles as the base unit.

To switch between the two modes, click on the *Km* or *Mi* radio button in the left border of the screen.

In the *Km* mode, consumptions are expressed in litres for 100 km (*L/100*).

In the *Mi* mode, consumptions are expressed in miles per gallon (*Mpg*).

Chapter 2

Settings menu

2.1 Installer mode

Some data in the settings menu are directly related to the terminals hardware configuration, or have a global impact on the database.

Generally, this information is defined once and for all during the installation, and doesn't need to change afterwards. To prevent inopportune changes in the configuration, the application includes a protection mechanism called "installer mode".

Installer mode is only available for manager accounts. It is enabled by entering a RSC code the first time a protected entry is modified. When installer mode is enabled, a red text is displayed in the left border of the screen. As long as this text is shown, all protected entries can be edited.

Installer mode stays enabled until explicitely stopped, or until the user logs out. To manually stop installer mode without logging out, click on the red text then click on the Stop button.

2.1.1 QSC and RSC codes

Installer mode is enabled by entering the RSC code associated to the QSC code displayed by the application.

The RSC code is either a constant or a random value, depending on the application version.

Constant values are given with the application package.

Random values are solved using a key generator, available on the Internet or through a voice server. The corresponding URL or phone number are specified with the application package, with an access limited to the authorized dealers.

All fields requiring the installer mode are followed by [RSC].

2.2 Global configuration

The global configuration can be edited in ▷ Settings, Config..

It contains several sections:

- Identification: defines how vehicles and/or drivers are identified on terminals.

- Default vehicle type: defines additional options for transactions on terminals (e.g. meter entry, volume limitation, ...).
- Auto backup: configures the automatic database backup.
- Exports: configures files exports.

2.2.1 Identification

Id. mode: Defines what must be identified on the terminal.

- *Vehicle only*: The terminal only identifies the vehicle.
- *Vehicle+Driver*: The terminal identifies the vehicle, then the driver.
- *Driver+Vehicle*: The terminal identifies the driver, then the vehicle.

Vehicle id.: Defines how the vehicle identification is performed.

- *Code (hidden)*: Vehicles are identified by entering a code on the keyboard. The entry on the terminal screen is concealed with the character '*'. The code must be defined in the Code field of the vehicle. The code will not be visible for users with the consultation level. Users with the manager level can see the code by ticking the dedicated checkbox.
- *Code (visible)*: Vehicles are identified by entering a code on the keyboard. The entry is visible on the terminal screen. The code must be defined in the Code field of the vehicle.
- *Registration*: Vehicles are identified by entering a code on the keyboard. The entry is visible on the terminal screen. The code must be defined in the Registration field of the vehicle. If this field contains several space-separated words, only the first word is used. (e.g. a vehicle with "1234 ABCD" in its registration field is identified by typing "1234" on the terminal).
- *Number*: Vehicles are identified by entering a code on the keyboard. The entry is visible on the terminal screen. The code must be defined in the Number field of the vehicle. The difference between Number and Code fields is that Number can also be used in vehicles display (see below).
- ... *key*: Vehicles are identified by placing a key in front of the reader. The key number must be defined in the Key field of the vehicle. Depending on the application version, one or several key types can be available (EMG keys, MFG keys, ...). For more information on each key type, see 9, page 69.
- ... *key + PIN code*: Vehicles are identified by placing a key in front of the reader, then by entering a secret code. The key number must be defined in the Key field of the vehicle. The PIN code must be defined in the Code field of the vehicle, and must be a 4-digits code. The PIN code is displayed under the same conditions as a hidden code.

Driver id.: Same thing as Vehicle id., for drivers identification.

Show vehicles as: Defines which fields are used to refer to a vehicle in the application.

- *Registration*: Only the Registration field is used.
- *Registration + Number*, *Number + Registration*: Fields Registration and Number are used in the specified order.

Show drivers as: Defines which fields are used to refer to a driver in the application.

- *Surname + First name*: Fields Surname and First name are used.
- *Surname + First name + Number*, *Number + Surname + First name*: Fields Surname, First name and Number are used in the specified order.

2.2.2 Default vehicle type

This section defines what a terminal should do after a vehicle is identified. Two main aspects can be configured:

- Optional entries: before starting the delivery, the terminal prompts for additional entries, that will be stored in the transaction (meter, NCE code).
- Volume control: the terminal limits the amount of fuel that a vehicle can take.

The following settings can also be defined in Vehicles types.

When a vehicle with no specified type is identified, the settings from the global configuration are used. Otherwise, the settings from the vehicle type are used.

This allows to adapt the terminal behaviour to the identified vehicle. For instance, you may want to ask for a meter only for some vehicles.

Meter: Defines if a meter entry is asked on the terminal.

- *None*: No meter entry
- *Km (no control)*, *Km (strict range)*, *Km (forceable range)*: A meter entry is asked after the identification (unit: km). There are three control modes to validate the integrity of the entered value:
  - *No control*: any value is accepted.
  - *Strict range*: the value entered must be between the last known meter and the last known meter added to the defined range. Example:

Last meter	Range	Allowed values
10000	2000	10000 to 12000

  - *Forceable range*: same control as above, but when the same value is entered twice, the terminal allows to bypass the control. The meter will be marked as "Forced" in the transaction.
- *Hours*: A meter entry is asked after the identification (unit: hour). The range control modes are the same as for *Km*.
- *Miles*: A meter entry is asked after the identification (unit: miles). The range control modes are the same as for *Km*.

Range: Defines the maximum difference between the last known and the current meter, on meter entry (Only for *Strict range* and *Forceable range* modes).

Vol.max: Defines the maximum volume that a vehicle can take in a single transaction.

Vol.max every: When a Vol.max is defined, defines how often a vehicle can take it.

- *Unlimited*: A vehicle can repeat transactions undefinitely.
- *1h-24h*: A vehicle can take at most Vol.max during the specified period.

Example: Assume that Vol.max is set to 100 L, and Vol.max every to *1h*:

- A vehicle that takes 100 L at 8:00 must wait until 9:00 before being allowed to retake fuel.
- A vehicle that takes 60 L at 8:00 then comes back at 8:15 can take at most 40 L. Its credit will then grow back to 60 L at 9:00, then to 100 L at 9:15, if no transaction occured in the meantime.

Options:

- *NCE code*: A non-checked entry is asked on the terminal, after the identification. Any non-empty value is accepted. This code is then stored in the transaction.

2.2.3 Auto backup

This section configures the automatic database backup. For more information on the backup tool, see 8.1, page 67.

Hour: Defines when the daily database backup should be launched.

Directory: Defines an external directory where backup files will be copied. It is better to define a directory located on a hard drive different from the one where the application is installed.

Copies: Defines how many backup files must be kept in the external directory

2.2.4 Other fields

HLF export: See 6.1, page 53.

AEAT export, MR4G export: Specific export formats settings, see 9, page 69.

CSV file format: Defines the format of downloadable CSV files

2.3 Terminals configuration

A terminal must be defined in the database in order for GIR Titan-Hykkoris to communicate with it. A terminal configuration includes the following steps:

- Products definition in ▷ Settings, Products.

  The products list is global to all terminals. In each vehicle, you can define a list of the products that it can take.

- Sites and tanks definition in ▷ Settings, Sites.

  A terminal is related to a site. The pumps defined on a terminal are related to the tanks defined in its site. Several terminals can share the same site.

- Terminal definition in ▷ Settings, Terminals:

  – Link definition: link type (network, modem, ...) and associated parameters (IP address, phone number, ...).

  – For each pump, definition of the product delivered, and of the tank in which it takes the fuel.

2.3.1 Products

Name: Product name. This field is used to refer to the product in the application. It also appears on the terminal screen.

Options:

- *Checked by default*: When this option is set, the product is checked by default when a new vehicle is created.
- *No meter*: When this option is set, the vehicle meter is never asked on the terminal for this product.
- *No vol.max*: When this option is set, the maximum volume allowed to a vehicle is ignored for this product.

Example: A station is delivering two products, *GOI* and *ADBLUE*. We have 50 vehicles, and we want to enable meter entry for 45 of them, but only for *GOI* deliveries.

Then the following configuration should be defined:

- In the global configuration, set *Km* in the Meter field.
- Create a vehicle type named `Special`, with *None* in the Meter field.
- Assign the vehicle type `Special` to the five vehicles that don't have to enter a meter when taking *GOI*. Leave all other vehicles with the vehicle type *(N/A)*.
- Set the *No meter* option in the *ADBLUE* product.

2.3.2 Sites

Name: Site name. This field is used to refer to the site in the application.

Tanks:

Capacity: Maximum tank capacity.

Alert treshold: Capacity percentage below which an alert is issued. The alert will apear on the dashboard (See 1.3.2, page 12).

For instance, defining an alert treshold of 20% for a 40 000 L tank will show an alert when the tank stock is below 8 000 L.

Stock: Current stock volume. It is decreased when fuel transactions are processed, and increased when a supply is entered.

Unit price: Current stock unit price. It is assigned to fuel transactions when they are processed.

2.3.3 Terminals

A summary of all modules on the terminal, with their addresses, is available in the terminal sheet.

Site: Terminal site. Pumps defined in this terminal can only be attached to tanks from the selected site.

Name: Terminal name. This field is used to refer to the terminal in the application.

Addr: Terminal address, as defined in the terminal internal setup. This is only usefull when multiplexing several terminals over a single RS485 bus, and should be left to 1 for all other cases.

Conn. type: Connection type

- *TCP/IP*: TCP/IP network connection.
- *Direct ...*: Direct serial connection (RS232) at 57600, 19200 or 9600 bauds.
- *Modem ...*: Modem connection at 57600, 19200 or 9600 bauds.
- *USB key*: Connection with a USB key (see below).

Host:Port: Hostname (or IP address) and TCP port for *TCP/IP* connections. Ex.: `192.168.12.34:6501`.

Serial port: Serial port for direct and modem connections

Phone num.: Phone number for modem connections.

Path: Path (generally a drive letter) to use for USB key connections. If this parameter is not specified, the path will be asked when the dialog starts.

S/N: Serial number of the terminal, for USB key connections. It is a 12-digits hexadecimal number.

Auto dial.: Hour of the daily automatic dialog with the terminal (See 5.1, page 41).

Status:

- *Active*: Terminal is available for dialogs and in the Stocks menu.
- *Inactive*: Terminal is not available for dialogs, but appears in the Stocks menu. This can be usefull to manage a "virtual" terminal where all transactions are entered manually.
- *Hidden*: Terminal is not available for dialogs and don't appear in the Stocks menu. This can be used either to pre-define the configuration for a future terminal which is not installed yet, or to hide an old terminal that has been removed, but still has transactions in the history.

Real time:

- *Enabled*: The terminal is updated automatically: data entered in the database are sent a few minutes after their entry, and transactions are retrieved just after their termination.
- *Retrieval only*: Transactions are retrieved just after their termination, data sending requires an explicit dialog.
- *Disabled*: Data sending and transactions retrieval both require and explicit dialog. Terminals using a modem connection always use this mode, no matter what the setting in the Real time field is.

T.vol: Defines how long the volume is displayed on the terminal screen after the end of a delivery.

Pumps access: Defines an access type to restrict the timeslots during which the terminal is available for fuelings (See 2.5.5, page 27).

USB key connection

When the Conn. type field is *USB key*, there is no physical link between GIR Titan-Hykkoris and the terminal. Data synchronization consists of the following steps:

1. Insertion of the USB key in the PC.
2. Launching of a dialog with the terminal (See 5.1, page 41). PC data (terminal configuration, vehicles, drivers, ...) are copied to the USB key. If the key has already been synchronized with the terminal, transactions retrieved on the USB key are stored in the database.
3. Insertion of the USB key in the terminal.
4. The USB key is detected, and the terminal asks to press a key to start the synchronization. During this step, data sent by the PC are copied to the terminal. The transactions in the terminal memory are copied on the USB key.
5. Back to step 1.

A single USB key can be used to synchronize several terminals at the same time.

Pumps

Product: Product delivered on this pump. When this field is *(N/A)*, the pump is not defined.

Tank: Tank associated to this pump. When transactions made on this pump are processed, the associated tank stock is decreased.

Tops/L: Number of pulses per litre for the counting device.

T.begin: Time after which a transaction is stopped if no pulse has been detected since the pump was commanded (See 5.3.5, page 49).

T.end: Time after which a transaction is stopped if no pulse has been detected, after at least one pulse occured (See 5.3.5, page 49).

Options:

- *Block after 3 null deliveries*: Automatically blocks the pump after 3 sucessive null deliveries.
- *Record unauthorised refuelings*: When an unauthorised refueling is detected, a transaction is stored with its date, duration and volume, and the 'Unauthorised refueling' mention.

Gauges

Type: Gauge type. When this field is *(N/A)*, the gauge is not defined. When a gauge is defined, an additional line shows up in the ▷ Stocks menu, with the tank stock returned by the gauge. This value can be clicked to view other gauge information (height, temperature ...).

Tank: Tank associated to this gauge.

Abacus: Abacus of the gauged tank. Abacuses are defined in ▷ Settings, Abacuses.

An abacus is defined by a name, and a list of height/volume couples.

Access

Command duration:

Access

Identification on accesses reuses the parameters defined in the identification section of the global configuration. Only the identification by key is supported.

Hence, vehicles or drivers can use accesses in the following cases:

- *Vehicle only*: Accesses can be used by vehicles if the Vehicle id. field is by key.
- *Vehicle+Driver* or *Driver+Vehicle*: Accesses can be used by vehicles if the Vehicle id. field is by key, and by drivers if the Driver id. field is by key.

An access is defined by the following fields:

Command duration: Access command duration in seconds. When this field is 0, the access is not defined.

Access type: Access type associated to this access. When this field is *(N/A)*, the access can be used by all authorized vehicles (resp. drivers), without any timeslot restriction.

When this field is defined, the access can be used only by vehicles (resp. drivers) for which this access type is checked. It is also possible to defined timeslots restrictions (See 2.5.5, page 27).

Fuelers

The fuelers section defines a list of special keys or codes that can be used to replace a normal identification. When a fueler key or code is used on the terminal, it prompts for a vehicle code. The transaction will then be assigned to the selected vehicle (See 5.4, page 50).

Key: Fueler key (when vehicles or drivers are identified with a key).

Code: Fueler code (only when both vehicles and drivers are identified with a code).

2.4 Users

When the application is used in a multi-user environment, a user account should be defined for each individual user, in ▷ Settings, Users.

Login: Account login, to enter on the login page.

Password: Account password, to enter on the login page. Accounts with a blank password can only login locally, on the GIR Titan-Hykkoris server. A password is required to login from a remote machine.

Level:

- *Manager*: Full access to the application.
- *Consultation*: Limited access to the application. The user can only consult data and launch dialogs to retrieve transactions.
- *Disabled*: Account disabled: user can't log in.

The *admin* login is built-in: the login name and level can't be modified. You can only customize the password.

The application keeps track of all actions performed by a user, in the audit trail (See 4.3, page 38).

For that reason, when a user no longer uses the application, it is better to set it to *Disabled* rather than to delete it. When a user is deleted, all its entries in the audit trail become unassigned.

2.4.1 Forgotten password

Here is the procedure to follow when an account password was forgotten:

1. For a custom user account: Login with the *admin* account, and define a new password for the concerned user.
2. For the *admin* account: On the GIR Titan-Hykkoris server, double-click the task tray icon to open a web browser on the login page. The web browser navigation bar should contain the following URL:

   `http://localhost:8080/cgi-bin/twcgibin.exe?p=4747`

   Ports 8080 and 4747 may vary depending on the configuration.

   Append `&admin=debug` to this URL, to obtain the following:

   `http://localhost:8080/cgi-bin/twcgibin.exe?p=4747&admin=debug`

   Press enter to load the new URL. This will display a special page, with a Password reset menu. Click on Password reset and confirm the operation.

   You can now come back to the login page and connect to the application with the login *admin* and no password.

   Note that this special page is only available from the GIR Titan-Hykkoris server.

2.5 Assignments

Assignments are various entities that are used to classify data. They can be used as filter criteria in several screens of the application. They generally consist of just a name, and are all optional.

They are all defined in the ▷ Settings menu.

2.5.1 Departments

Departments are used to classify vehicles and drivers. A department can be used as filter criterion in vehicles and drivers lists, transactions history, reports, ...

Name: Unique name of the department. This field is used to refer to the department in the application.

Usage example: Define a department for each subsidiary company, to easily generate reports containing only the vehicles from a specific company (Company1, Company2, ...).

2.5.2 Vehicles types

Vehicles types are used to classify vehicles. A vehicle type can be used as filter criterion in vehicles list, transactions history, reports, ...

Name: Unique name of the vehicle type. This field is used to refer to the vehicle type in the application.

Usage example: Define a vehicle type for each manufacturer, and generate a report by vehicle type to compare fuel consumptions between various manufacturers (Mercedes, Renault, Scania, ...).

Vehicles types also define several settings (Meter type, maximum volume, ...). The settings defined in a vehicle type apply to all vehicles assigned to this type. Vehicles that are not assigned to a type use the settings defined in the global configuration.

For a detailed description of these settings, see 2.2.2, page 19.

2.5.3 Drivers types

Drivers types are used to classify drivers. A driver type can be used as filter criterion in drivers list, transactions history, reports, ...

Name: Unique name of the driver type. This field is used to refer to the driver type in the application.

2.5.4 Providers

Providers are used to classify tank supplies. A provider can be used as filter criterion in tank supplies history.

Name: Unique name of the provider. This field is used to refer to the provider in the application.

2.5.5 Access types

An access type allows to restrict the authorized keys on one or several access points. When an access type is defined, a checkbox appears in vehicles and drivers (when identified by a key), allowing to specify their authorization on the access type.

An access type contains the following fields:

Name: Unique name of the access type. This field is used to refer to the access type in the application.

Checked by default: When this option is set, the access type is checked by default when a new vehicle or driver is created.

Authorized timeslots: By default, an access is authorized 24/7. When this option is enabled, timeslots can be defined in the access type sheet. Two timeslots can be defined for each day of the week. Vehicles or drivers with the Authorized 24H/24H option are not concerned with timeslots restrictions.

Forced timeslots: Enables access command forcing. This can be done either automatically, by defining forcing timeslots in the access type sheet, or manually, using remote forcing in the ▷ Stocks menu.

Chapter 3

Vehicles and drivers

Vehicles and drivers are used to know who takes the fuel. They are identified on the terminal either by using a key or entering a code, then are stored in fuel transactions, along with other information on the fuel delivery (date, volume, ...).

Driver identification is optional, depending on the settings in the global configuration.

3.1 Vehicles

Registration, Number: Fields used to refer to the vehicule in the application. For instance, they are displayed:

- In the Vehicle column of the fuel transactions history.
- In the dropdown list of vehicles, in the manual transaction entry form.
- In the Vehicle column of fuel transactions reports.

The Number field is optional, and can be displayed either before or after Registration, as defined in the Show vehicles as field of the global configuration (See 2.2.1, page 18).

Key, Code: Key used for identification on terminals (See 2.2.1, page 18).

Dept.: Asssociated department (See 2.5.1, page 26). It can be used as a criterion in histories and reports.

Usage example: Company1, Company2, ...

Type: Associated vehicle type (See 2.5.2, page 26). It can be used as a criterion in histories and reports, and also defines several settings that will apply to the vehicle.

Usage example: Mercedes, Renault, Scania, ...

Km: Current meter value. This field is automatically updated when a fuel transaction with an entered meter is processed.

Generally, meters should not be modified manually. The only case where this might be necessary is when an invalid meter was entered in a transaction, and that it causes the following meter entries on the terminal to be refused. To manually modify a meter, click on the meter value in the vehicles list.

Options:

- *Forbidden*: The vehicle isn't allowed to take fuel. If such a vehicle is identified on a terminal, the message `VEHICLE FORBIDDEN` will be displayed, and an event will be stored.
- *Synchronized*: The vehicle can be updated by synchronized import (See 7.3.1, page 64).
- *Authorized 24H/24H*: The vehicle ignores timeslots restrictions in access types (See 2.5.5, page 27).

Products: List of all products authorized to this vehicle.

Accesses: List of access types authorized to this vehicle (See **??**, page **??**).

Maintenances: List of maintenances for this vehicle (See 9.1.1, page 69).

3.1.1 Department reassignment

When a fuel transaction is processed, the current vehicle department is stored in the transaction. This way, when a vehicle department changes, existing transactions remain assigned to the previous department, while new transactions will be assigned to the new department.

However, the vehicle department is not always changed exactly at the date of its administrative transfer. Hence, when a vehicle department is changed, the application asks if departments should be recomputed.

Click Continue to proceed with the department modification without changing any transactions.

Click Recompute departments to update the department in all transactions assigned to this vehicle and belonging to a given time range.

3.2 Drivers

Surname, First name, Number: Fields used to refer to the driver in the application. For instance, they are displayed:

- In the Driver column of the fuel transactions history.
- In the dropdown list of drivers, in the manual transaction entry form.
- In the Driver column of fuel transactions reports.

The Number field is optional, and can be displayed either before or after Surname and First name, as defined in the Show drivers as field of the global configuration (See 2.2.1, page 18).

Key, Code: Key used for identification on terminals (See 2.2.1, page 18).

Dept.: Asssociated department (See 2.5.1, page 26). It can be used as a criterion in histories and reports.

Usage example: Company1, Company2, ...

Type: Associated driver type (See 2.5.3, page 27). It can be used as a criterion in histories and reports.

Usage example: Truck drivers, Administrative staff, ...

Options:

- *Forbidden*: The driver isn't allowed to take fuel. If such a driver is identified on a terminal, the message `DRIVER FORBIDDEN` will be displayed, and an event will be stored.
- *Synchronized*: The driver can be updated by synchronized import (See 7.3.1, page 64).
- *Authorized 24H/24H*: The driver ignores timeslots restrictions in access types (See 2.5.5, page 27).

Accesses: List of access types authorized to this driver (See **??**, page **??**).

Chapter 4

Operations history

The application keeps track of various operations:

- Fuel transactions: Data related to fuel deliveries. Fuel transactions can either be retrieved from terminals, or entered manually. They can be consulted in two ways:
  - In the ▷ History menu, for a detailed list of each individual transaction.
  - In the ▷ Reports menu, for a more synthetic view of all the transactions that occured during a given time range.
- Events: General-purpose history, in ▷ Misc., Events. Events can either be retrieved from terminals, or generated by the application.
- Tank supplies: Tank supplies history, in ▷ Misc., Tank supplies.
- Access transactions, in ▷ Misc., History of accesses.
- Audit trail: Users actions history, in ▷ Misc., Audit trail.

4.1 Fuel transactions

4.1.1 History

By default, the history shows all transactions stored in the database, ordered by decreasing date. Various queries can be performed on the history, to view only transactions that match a specific criterion (See 1.4, page 14).

The following fields are shown in the transactions list:

Time: Date and time of the transaction. The time is taken when the pump is commanded, and stored with a one-second precision.

Dept.: Vehicle department at the time of the transaction.

Vehicle: Identified vehicle.

Driver: Identified driver.

Prod.: Product delivered.

Pump: Terminal and pump where the delivery was made.

L: Volume delivered.

€/L: Unit price.

Km: Meter entered (unit may vary).

Cov. km: Distance covered since the previous transaction.

L/100: Fuel consumption, in litres for 100 km (or miles per gallon).

The following fields are only shown in the detailed transaction information, available by clicking on the time value:

Dur.: Transaction duration.

NCE code: Non-checked entry entered on the terminal.

Special notes

Some transactions can contain a special indicator on one or several fields. These indicators are generally shown between brackets, and a hint is displayed at the bottom of the page.

The following list details all possible indicators:

- [M] in the Pump field: Transaction entered manually (See 4.1.2, page 34).
- [F] in the Km field: The meter entry was forced (See 2.2.2, page 19).
- [Max] in the L field: The delivery stopped after reaching the maximum volume (See 2.2.2, page 20).
- [Flr.] in the Vehicle field: A fueler key or code was used (See 2.3.3, page 25).
- [New] in the Cov. km field: New meter (See 4.1.6, page 37).
- (1234) in the Vehicle field: NCE code *1234* was entered.

4.1.2 Manual transaction entry

Transactions can be entered manually. This is generally used in two cases:

- To enter fuelings that were made outside of the stations managed by the application.
- To enter fuel deliveries that were made manually, for instance during a temporary terminal malfunction.

Click on the Manual transaction button in the transactions history to display the manual transaction form.

You will be prompted for the following information:

Type: Transaction type

- *External*: Fuel delivery made in a station that is not managed by the application.
- *Internal*: Manual fuel delivery on a terminal defined in the application.

Time: Date and time of the fueling.

Vehicle: Vehicle that made the fueling.

Driver: Driver that made the fueling (Only when driver identification is enabled).

Volume: Quantity of fuel delivered, in litres.

Depending on the transaction type and the selected vehicle, additional data might be required:

Product: Product delivered (Only for external transactions).

Unit price: Unit price of the delivery (Only for external transactions).

Pump: Pump where the delivery was made (Only for internal transactions).

Meter: Vehicle meter at the time of the delivery (Only when meter entry is enabled for the selected vehicle).

NCE code: NCE code associated to the delivery (Only when NCE code entry is enabled for the selected vehicle).

When all data has been entered, it is shown in a summary screen. The Validate button confirms the entry, and stores the transaction in the history. The Cancel button goes back to the previous step to allow data correction.

4.1.3 Transaction modification

Once a transaction is stored in the history, it can be modified. Transaction modification is usefull in several cases:

- To correct an error in a transaction entered manually
- To correct an error in an entry made on the terminal (meter, NCE code, ...).
- To assign the transaction to another vehicle or driver, e.g. when a vehicle was identified using the key of another vehicle.
- To correct the delivered volume, in case of pump malfunction.

Click on the Modify button in the transactions history to modify a transaction. A form with all editable transaction fields will be displayed. Note that all fields are not always editable:

- For transactions retrieved from a terminal, the date and pump can't be modified.
- For transactions entered manually, any field can be modified.

All fields available when editing a transaction are the same as for a manual entry, and are described in the previous section. The only exception is the *New meter* option, which is explained in the Reports section (See 4.1.5, page 36).

When a transaction has been modified, a new button appears in the history: History of changes. It shows an history of all the modifications that were done on a transaction, as well as the user who made them. The first row, in bold, shows the current transaction. The last row shows the original transaction.

4.1.4 Transaction cancellation

Transaction cancellation is mainly usefull to cancel manual entries that shouldn't have been done. For convenience, the application also allows to cancel transactions retrieved from the terminal. However, there are very few, if any, valid reasons for cancelling a transaction that was retrieved from a terminal.

Click on the Cancel button in the transactions history to cancel a transaction. You will be prompted for confirmation.

Note: Cancelled transactions are NOT deleted from the database. They are still visible by clicking on the Cancelled transactions button in the history.

The only way to permanently remove a transaction from the database is to do a purge (See 8.2, page 68).

4.1.5 Transactions reports

The ▷ Reports menu allows to compute various sums on fuel transactions. For instance, reports can be used to follow the average consumption of a vehicle, or to compare the fuel consumption between several departments.

The report type is defined by a small form at the top of the screen. The first row contains the following fields:

Period: Defines the period during which transactions should be sumed.

Section,Criterion: Defines the elements on which the sum should be computed. For instance, selecting *Department* as section and *Vehicle* as criterion will show the total consumption of each vehicle during the selected period, grouped by department.

Selecting nothing as section and *Driver* as criterion will show the total consumption of each driver during the selected period.

When using the Printable version button of a report, each section begins on a separate page.

The second row contains various filters, that can be used to restrict the sum to transactions related to a specific element (Department, Site, ...).

For each element in the sum, the following data is displayed:

Prod.: Product.

L: Total volume.

Cov. km: Total covered distance.

L/100: Average consumption, computed from the total volume and covered distance. Note that the volume of *New meter* transactions is counted in the L column, but is ignored in the consumption computation. This is why the value in the consumption column doesn't always correspond to the total volume and distance displayed on the same row (See below).

A total is made for each section, as well as a global total.

4.1.6 New meter

Computing an accurate average consumption can be a bit tricky. For instance, consider a new vehicle, that was purchased with a meter of 1234 km. Let's assume that the vehicle makes 3 transactions during its first month of service. Its transaction history will look like this:

Time	L	Km	Cov. Km
29/06/07 08:15	109.00	1924	324
25/06/07 07:32	115.00	1600	366
22/06/07 14:20	150.00	1234	0

If we sum all transactions, we get a total volume of 374 L for a covered distance of 690 km. The average consumption must then be 54.2 L/100km.

But, actually, the volume of the first transaction isn't associated to a covered distance. The vehicle only used 224 L to cover 690 km. Thus, the real average consumption is 32.5 L/100km.

To solve this problem, the application defines the *New meter* option. When a transaction has this option, its volume is ignored in consumption computation.

The *New meter* option is automatically set for the first transaction of a vehicle. It can also be set (or unset) by doing a transaction modification.

This way, when the covered distance in a transaction is not known (e.g. because the previous meter was invalid), just check the *New meter* option to avoid computing a false average consumption.

4.2 Events

The events history is a general-purpose list of various events, that either occured on a specific terminal, or were notified by the application.

Time: Date and time of the event

Type: Event type

- *Link*: Terminal link status change.
- *Ident.*: Unknown or forbidden key on vehicle or driver identification.
- *Pump*: Pump status change: manual/auto mode, blocking or unblocking.
- *Misc.*: Other events.

Term.: Terminal where the event occured.

Message: Event detail.

4.3 Audit trail

The audit trail keeps track of all the actions performed by users that caused a change in the database. It cannot be purged or edited.

Time: Date and time of the action.

User: User who peformed the action.

Operation: Type of action.

- *Login*: User logged in or out.
- *Creation*: Creation of a new element.
- *Modification*: Modification of an existing element.
- *Deletion*: Deletion of an existing element.
- *Dialog*: Entrance or exit from the Stocks menu, or dialog launching.
- *Misc.*: Various actions.

Detail: Detailed information about the action, e.g. what fields were modified, what terminals were synchronized, ...

4.4 Tank supplies

Tank supplies keep track of the tanks stocks evolution.

- The tank supplies history can be viewed in ▷ Misc., Tank supplies.
- New tank supplies can be entered from the ▷ Stocks menu.

4.4.1 History

Time: Date and time of the supply.

Op.: Mode used for data entry: *Supply* or *Regularization*.

Tank: Site and tank where the supply was done.

Provider: Fuel provider.

Old stock, Old U.P.: Tank stock and unit price before the supply was entered.

L: Supply volume. It is always the difference between Old stock and New stock.

New stock, New U.P.: Tank stock and unit price after the supply was entered.

4.4.2 Manual supply entry

To enter a tank supply, go to the Stocks menu, then click the Tank supply button next to the tank name. This will display the tank supply entry form.

There are two entry modes:

- *Supply*: Enter a real tank supply, i.e. a new quantity of fuel that will be added to the current stock. The new stock unit price will be automatically computed using a weighted average method:

$$newprice = (P_s * V_s + P_t * V_t)/(V_s + V_t)$$

  with:

  P_s: supply unit price

  V_s: supply volume

  P_t: tank stock unit price

  V_t: tank stock volume before the supply

- *Regularization*: Enter a Volume and a Unit price that directly replace the current stock and its unit price.

In both modes, an optional Provider can be selected.
After the validation, the new supply is stored in the supplies history.

4.4.3 Supply cancellation

Tank supplies can be cancelled in the history by clicking on the Cancel button. You will be prompted for confirmation.

To preserve stocks consistency, only the last supply on each tank can be cancelled.

Note: Cancelled supplies are not deleted from the database. They are still visible by clicking on the Cancelled supplies button in the history.

4.5 History of accesses

This history shows all access transactions stored in the database, ordered by decreasing date. Various queries can be performed on the history, to view only transactions that match a specific criterion (See 1.4, page 14).

The following fields are shown in the transactions list:

Date: Date and time of the transaction.

Vehicle: Vehicle that made the access.

Driver: Driver that made the access.

Access: Terminal and access point where the transaction was made.

Access type: Access type, as defined in the terminal configurtion.

Status: Authorized or refused.

Chapter 5

Terminals

Terminals are autonomous: they can operate without a permanent link with the GIR Titan-Hykkoris server. To this effect, each terminal has its own local copy of the database, with the list of vehicles and drivers allowed to take fuel. When a transaction is made, it is stored in the terminal memory, waiting to be retrieved by the GIR Titan-Hykkoris server.

Therefore, it is necessary to perform regular synchronizations ("Dialogs") between terminals and the GIR Titan-Hykkoris server.

5.1 Dialogs

5.1.1 Launching a dialog

Dialogs can be launched in several ways:

- Automatically, by defining a daily dialog hour in the terminal configuration.
- Automatically, for terminals using "real-time" mode.
- Manually, in the Dialogs menu. Select a terminal and click the Validate button to launch a dialog. Selecting *All* will launch a dialog with all terminals in the list.

5.1.2 Dialog detail

The synchronization with a terminal consists of several steps. The following steps are always performed:

- Link test: The GIR Titan-Hykkoris server checks the terminal version and its authenticity.
- Transactions retrieval: Transactions are moved from the terminal memory to the application database.
- Time update: The terminal clock is synchronized with the GIR Titan-Hykkoris server clock.

- Data sending: The terminal database is updated to match the application database (list of authorized vehicles and drivers).

Optionally, one or several of the following steps may occur:

- Initialization: The terminal firmware is updated. This step occurs during the first dialog with a terminal, or when a new firmware version is available. When an initialization occurs, it is always preceded by an interactive prompt (See 5.1.5, page 43).
- Configuration sending: The terminal configuration is updated to match the configuration defined in the application (number of pumps, identification mode, ...). This step occurs when a terminal configuration is changed in the application, or when a terminal is replaced with another.

5.1.3 Dialog progress

When a dialog is running, the dialog progress screen is displayed in the Dialog menu.

The icon above the progress bar indicates the dialog status:

- Hourglass: Dialog is in progress.
- Green check: Dialog completed successfully.
- Warning sign: Dialog cancelled or aborted because an error occured. Detail about the error is indicated.

The Cancel button immediately stops the dialog. This should be used with caution, as a cancelled dialog could leave the terminal in a half-synchronized state:

- A dialog cancelled during transactions retrieval can leave transactions on the terminal. These transactions will be retrieved at the next dialog.
- A dialog cancelled during data sending can leave the terminal database partially not up to date. For instance, a vehicle that was just added in the server database may not be able to perform identification on the terminal.
- A dialog cancelled during initialization can leave the terminal in a state where transactions can't be performed. In that case, it is necessary to restart a dialog and to wait for the initialization to complete.

5.1.4 Real time

The bottom part of the screen shows the status of all connections with real-time terminals.

Real-time connections are automatically established a few minutes after the application startup. If an error occurs, GIR Titan-Hykkoris regularly retries to establish the connection.

Click on the icon next to a real-time terminal to launch a dialog with this terminal.

5.1.5 Interactive prompts

In a few cases, a dialog can be suspended, waiting for a the user to make a decision. The progress screen displays an explanatory message, and the list of available choices.

A dialog is suspended in the following cases:

Terminal not initialized: The terminal contains no application. This can be either because the terminal is new, or because a previous initialization was interrupted. The available choices are:

- Initialize the terminal: Performs a new initialization, all data on terminal is deleted.
- Retrieve transactions, then the terminal: Same as above, except that transactions are retrieved before the initialization. This choice is only available when a compatible data format is detected.
- Cancel the dialog: The terminal is left unchanged.

Unknown terminal: The terminal identification data is not what was expected. The generally means that the Vatersay management unit was changed. The available choices are:

- Initialize the terminal: Performs a new initialization, all data on terminal is deleted.
- Retrieve transactions: Forces transactions retrieval. This should be used with caution, as there is no guarantee that transactions stored in the terminal are consistent with the configuration defined in the database.
- Cancel the dialog: The terminal is left unchanged.

New firmware available: A new version is available for the terminal application firmware. The available choices are:

- Upgrade the terminal: Initializes the terminal with the new firmware.
- Continue the dialog: Keeps the previous version. The same question will be asked again on the next dialog.

Notes

If no choice is made within ten minutes, the dialog is cancelled.

Interactive prompts only occur for dialogs launched manually. For automatic dialogs, the following choices are implicitly made:

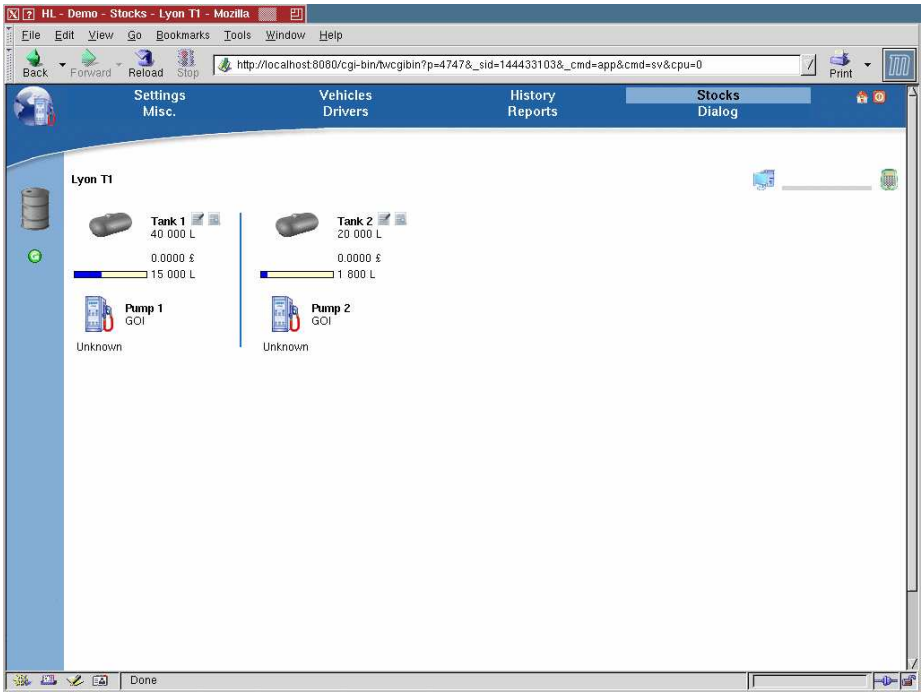
- Unknown or uninitialized terminal: dialog is cancelled.
- New firmware available: the terminal is not upgraded.

5.2 Stocks

The Stocks menu shows an overview of a terminal status, with information on the associated tanks and pumps.

It offers the following features:

- View tanks stocks
- Enter tank supplies
- View pumps status
- Block or unblock pumps
- Remote transaction
- Access forcing or forced stop



Tanks stocks and supply entry use only the database, and are always available.

Pumps status are retrieved in real-time from the terminal. They are only available when the link with the terminal is working.

When using a modem, the connection is established only upon explicit request, by clicking on the Dial button. The connection is automatically closed after a few minutes of inactivity. It can also be closed manually using the Hang up button.

When a terminal status is being retrieved, a small icon at the top right of the screen shows the status of the underlying dialog. When an error occurs, click on it to view detailed information about the error (See 5.1.3, page 42).

5.2.1 Tanks

For each tank, the following information is displayed:

- Tank name.
- Capacity.
- Current stock unit price.
- Current stock volume, with a graphical view in a horizontal bar.
- When a gauge is defined, current stock volume returned by the gauge. This value can be clicked to view other gauge inforamtion (See 2.3.3, page 24).

Two buttons are available next to the tank name. Click on Supply to enter a tank supply. Click on Detail to view the tank supplies history. See 4.4, page 38 for more information.

5.2.2 Pumps

A status is displayed for each pump, below the pump icon. It can be one of the following:

- *Unknown*: Pump status is not retrieved yet. Either a dialog is in progress, or a dialog error occured.
- *Manual mode*: Pump is forced to manual delivery.
- *Blocked*: The pump is blocked. The blocking causes can be:
  - *3 null deliveries*: Pump blocked after 3 deliveries with a null volume.
  - *Manager*: Pump blocked by a user in the Stocks menu.
  - *Module error*: Pump blocked after an error in the module memory.
- *Link error*: Link error between the pump module and the Vatersay unit. Check cable connections and modules addresses.
- *Delivering*: A fuel delivery is in progress. Information on the transaction is displayed (vehicle, volume, ...).
- *Available*: Pump is available for a new delivery.

Only one status is displayed for each pump, with a priority corresponding to the order given above (the higher in the list, the first displayed). That is, if a pump is both blocked and forced to manual mode, it will appear as “Manual mode”.

A button can appear next to the pump status. It depends on the status value:

- If the pump is blocked, an Unblock button is available. Click it to unblock the pump.
- Otherwise, if the pump is not unknown or in manual mode, a Block button is available. Click it to block the pump.

Pump blocking requires a short dialog with the terminal. During this time, the status will be displayed as “Blocking” or “Unblocking”. If the dialog fails, the pump keeps its existing status.

5.2.3 Access

A status is displayed for each access, below the access icon. It can be one of the following:

- *Inconnu*: Access status is not retrieved yet.
- *Link error*: Link error between the access module and the Vatersay unit. Check cable connections and modules addresses.
- *Not available*: Access disabled because of timeslots restrictions.
- *Available*: Access available.
- *Forced*: Access forced (always opened).
- *Forced stop*: Access stopped (always closed, even if forcing timeslots are defined).

5.2.4 Warnings

In the bottom part of the screen, various warnings can appear to indicate a terminal malfunction:

- *Terminal module error*: Link error between the terminal module and the Vatersay unit. Check cable connections and modules addresses.
- *Reader module error*: Link error between the reader module and the Vatersay unit. Check cable connections and modules addresses.
- *Erreur module jauge*: Link error between the gauge module and the Vatersay unit. Check cable connections and modules addresses.
- *Invalid date*: The terminal clock is invalid. Launch a dialog to fix the date and solve the problem.
- *Memory full*: The terminal can’t store any new transaction. Launch a dialog to retrieve transactions and solve the problem.
- *Memory error*: An error occured in the terminal memory. The terminal must be replaced.

5.3 Terminal usage

When a terminal is idle, it displays one of the following messages:

- Message with pump numbers (e.g. `PUMP 1 AVAILABLE`, or `SEL.PUMP 1 2`): one or several pumps are available. Pumps for which numbers are not displayed are either in use or unavailable.
- `DELIVERY IN PROGRESS`: All available pumps are currently in use.
- `OUT OF ORDER`: No pump is available.

The fueling scenario consists of the following steps:

1. Pump selection: selection of a pump number on the keyboard. This step is skipped when only one pump is defined on a terminal.
2. Identification: vehicle and (optionally) driver identification, as defined in the configuration (See 2.2.1, page 18).
3. Optional entries: meter and NCE code entries, as defined in the configuration (See 2.2.2, page 19).

During data entry, the scenario can be cancelled at any time by pressing the DEL key. The message `TRANSACTION CANCELLED` appears, and the terminal returns to the idle state.

A detailed diagram of the fueling scenario is available in appendix to this document (See C, page 83).

5.3.1 Pump selection

```
Mo 03/09/07 08:30:00
PUMP 1 AVAILABLE
```

or

```
Mo 03/09/07 08:30:00
SEL.PUMP 1 2
```

Select the pump number you wish.

```
P1 GOI
Validate your choice
```

The pump and the product name appears, validate with VAL.

Notes:

- When only one pump is defined on the terminal, pump selection is skipped and identification is directly asked.
- When a pump is not available, pressing its number shows the reason.

  Example: Assume that 3 pumps are defined, and the terminal screen is as follows:

  ```
  Mo 03/09/07 08:30:00
  SEL.PUMP 1 2
  ```

  Press 3 to known why pump 3 is not available:

  ```
  Pump 3 GOI
  BLOCKED/MANAGER
  ```

  All possible pumps status are listed in the Stocks menu description (See

5.2.2, page 45).

5.3.2 Identification

A terminal waiting for a driver or vehicle identification displays one of the following messages:

`P1 GOI` `VEHICLE KEY`	Terminal waiting for a vehicle key
`P1 GOI` `VEHIC. CODE`	Terminal waiting for a vehicle code
`P1 GOI` `DRIVER KEY`	Terminal waiting for a driver key
`P1 GOI` `DRIVER CODE`	Terminal waiting for a driver code

Identification can be done in several ways, depending on the configuration:

- by key: a key must be placed in front of the reader.
- by code: a code must be entered on the keyboard.
- by key + PIN code: a key must be placed in front of the reader, then a code must be entered on the keyboard (message `PIN CODE`).

When both vehicles and drivers must be identified, both identifications will be successively requested.

Possible error messages:

- `KEY NOT RECOGN.`, `UNKNOWN KEY`: The key is not defined.
- `UNKOWN CODE`: The code is not defined.
- `KEY EXPECTED`: The code entered resolves to a vehicle or driver which must be identified by key.
- `KEY OUT OF CONTEXT`, `CODE OUT OF CONTEXT`: The key or code identifies a vehicle when a driver is expected, or vice versa.
- `VEHICLE FORBIDDEN`, `DRIVER FORBIDDEN`: The identified vehicle or driver is forbidden.

- `PRODUCT NOT AUTHOR.`: The pump product is not authorized to this vehicle.
- `INSUFFICIENT CREDIT`: The fuel credit is depleted.

5.3.3 Meters entry

```
P1 GOI
Meter km:
```

Enter the odometer value and validate with VAL. Unit may vary.

Possible error messages:

- `INCORRECT VALUE`: The meter entered is out of the validity range. See 2.2.2, page 19.

A meter can be forced even when it is out of the validity range, in the following cases:

- Transaction via fueler (See 2.3.3, page 25). The confirmation message appears after the first meter entry.
- Meter with the *Forceable range* mode. The confirmation message appears when the same value is entered twice in a row.

5.3.4 NCE code entry

```
P1 GOI
NCE code:
```

Enter the NCE code and validate with VAL. Any non-empty value is accepted.

5.3.5 Delivery times

Once the fueling scenario is completed, the delivery starts and the terminal displays `READY FOR DELIVERY`.

The fueling then depends on the pump configuration, based on the following variables:

T_{begin}: Time before first pulse

T_{end}: Time after last pulse

T_{after}: Post-delivery duration

T_{begin} and T_{end} are defined in the pump configuration (See 2.3.3, page 24). T_{after} is fixed to 2 seconds.

A fueling uses the following rules:

- If nothing is delivered for T_{begin} seconds, the delivery stops.
- Once a first output has occured, if no other ouput is made for T_{end} seconds, the delivery stops.
- If a hangup is managed, the delivery stops when the pump nozzle is hanged up. The two time periods above remain valid.
- Once the delivery has been stopped, the volume output is still counted for T_{after} seconds. This allows to take into account compressing phenomena in the delivery pipe.

5.4 Fueler mode

Fueler keys or codes are defined in the terminal configuration (See 2.3.3, page 25). A fueler identification is usefull in two cases:

- As a "joker" key when a key has been lost or forgotten.
- To bypass some restrictions for a specific fueling (forbidden vehicle or driver, product not authorized, ...)

When the vehicle is identified by a key, the code to enter to force the vehicle depends on how the vehicle is displayed in the identification section of the global configuration (see 2.2.1, page 18):

Registration: the first word of the registration.

Registration + Number: the first word of the registration.

Number + Registration: the number.

When the driver is identified by a key, the code to enter to force the driver depends on how the driver is displayed in the identification section of the global configuration (see 2.2.1, page 18):

Surname + First name: the surname.

Surname + First name + Number: the surname.

Number + Surname + First name: the number.

The following sections show how to use a fueler key or code in each possible configuration.

5.4.1 Vehicle code only

Vehicle forcing:

- `VEHIC. CODE`: Enter the fueler code
- `VEHIC. CODE`: Enter the vehicle code

5.4.2 Vehicle key only

Vehicle key lost or vehicle forcing:

- `VEHICLE KEY`: Use the fueler key
- `VEHIC. CODE`: Enter the vehicle code

5.4.3 Driver key then vehicle code

Driver key lost, driver forcing or vehicle forcing:

- `DRIVER KEY`: Use the fueler key
- `VEHIC. CODE`: Enter the vehicle code

5.4.4 Driver key then vehicle key

Driver key lost or driver forcing:

- `DRIVER KEY`: Use the fueler key
- `VEHICLE KEY`: Use the vehicle key

Vehicle key lost or vehicle forcing:

- `DRIVER KEY`: Use the driver key
- `VEHICLE KEY`: Use the fueler key
- `VEHIC. CODE`: Enter the vehicle code

5.4.5 Driver code then vehicle key

Driver forcing: impossible. In that case, it is better to use a "vehicle key then driver code" identification.

Vehicle key lost, or vehicle forcing:

- `DRIVER CODE`: Enter the driver code
- `VEHICLE KEY`: Use the fueler key
- `VEHIC. CODE`: Enter the vehicle code

5.4.6 Driver code then vehicle code

Driver forcing:

- `DRIVER CODE`: Enter the fueler code
- `VEHIC. CODE`: Enter the vehicle code

Vehicle forcing:

- `DRIVER CODE`: Enter the driver code
- `VEHIC. CODE`: Enter the fueler code
- `VEHIC. CODE`: Enter the vehicle code

5.4.7 Vehicle key then driver code

Vehicle key lost, vehicle forcing or driver forcing:

- `VEHICLE KEY`: Use the fueler key
- `VEHIC. CODE`: Enter the vehicle code
- `DRIVER CODE`: Enter the driver code

5.4.8 Vehicle key then driver key

Vehicle key lost or vehicle forcing:

- `VEHICLE KEY`: Use the fueler key
- `VEHIC. CODE`: Enter the vehicle code
- `DRIVER KEY`: Use the driver key

Driver key lost or driver forcing:

- `VEHICLE KEY`: Use the vehicle key
- `DRIVER KEY`: Use the fueler key

5.4.9 Vehicle code then driver key

Vehicle forcing: impossible. In that case, it is better to use a "driver key then vehicle code" identification.

Driver key lost, or driver forcing:

- `VEHIC. CODE`: Enter the vehicle code
- `DRIVER KEY`: Use the fueler key
- `DRIVER CODE`: Enter the driver code

5.4.10 Vehicle code then driver code

Vehicle forcing or driver forcing:

- `VEHIC. CODE`: Enter the fueler code
- `VEHIC. CODE`: Enter the vehicle code
- `DRIVER CODE`: Enter the driver code

Chapter 6

Data export

Data export allows to generate files in a specific format, for further processing in other application.

Several exports are available:

- HLF export: Export fuel transactions in a fixed-format/CSV file.
- Custom HLF export: Export fuel transactions in file with a custom format.
- AEAT export, MR4G export, ...: Specific exports (See 9, page 69).

They are all available in the ▷ Misc. menu.

For all exports, the screen is splitted in two parts:

- New export: Generates a new exported file. Select a time range and validate. Exported files are created in the `output` subdirectory. They can also be downloaded.
- Previous exports: Lists all the exported files in the `output` subdirectory. Click on Download to download the file.

6.1 Export configuration

Format: Defines the file format to use for transactions export. A description of HLF export formats is available in appendix to this document.

Delimiter: Defines the delimiter to use in exported files (except for custom export, see 6.3, page 54).

Hour: A file is exported every day at the specified hour, in the selected format.

Frequency: A file is exported at the selected frequency, if a transaction occured since the last export.

6.2 Export conditions

When exporting transactions, the system uses the following rules:

- One transaction is exported for each transaction that was created during the specified time range.
- Two transactions are exported for each transaction that was modified during the specified time range.
  - The first exported transaction is equal to the transaction before the modification, except that its volume is the opposite of the original volume. When the format allows it, the transaction is flagged as a cancellation.
  - The second exported transaction is equal to the transaction after the modification. When the format allows it, the transaction is flagged as a replacement.

This mechanism guarantees that for a given time range, the exported file will always be the same.

It also allows to export transactions at any time without having to worry about pending modifications. If a modification isn't done at the time of the export on a transaction that is already stored, it will be integrated in the next export.

6.3 Custom export

Custom export allows to export a file in a user-defined format.

6.3.1 Global configuration

Global parameters define fields formats, and date/time formats.

Format type: CSV or fix. CSV format allows variable length fields, whereas fix format requires a length for each field. When using a fix format, CSV parameters are ignored.

Export cancelled or modified transactions: This option defines if cancelled transactions are exported or not. When they are, their volume is negative (See 6.2, page 54).

Separator: Defines the CSV delimiter.

Fields with text separator: Defines the fields that will be surrounded by the text delimiter. This is useful when fields can contain the CSV delimiter.

Text separator: Defines the text delimiter.

Date: Defines the date format. Characters y,m,d are interpreted as year, month and day. Other characters are interpreted as delimiters. Example: yyyy/mm/dd.

Hour: Defines the hour format. Characters h,m,s are interpreted as hour, minute and second. Other characters are interpreted as delimiters. Example: hh:mm:ss.

6.3.2 Fields types

Each field can be one of:

Constant value: Constant characters string.

Date - hour: Date and hour formats are defined in the custom export global configuration.

Transaction field: The field value depends on the transaction. It can be either a characters string, an integer, a decimal number, or a reference (see 6.3.5, page 56).

Options fields: The field has two possible values, depending on an option in the transaction. One value is exported if the option is set, and the other value if it is not set.

6.3.3 Field insertion

1. Click on Add ().
2. Select a field type and a length (optional in CSV), and validate.
3. Depending on the selected field type, additionnal entries may be required:

Constant value: Enter a value.

Option field: Enter two values.

When a length is defined:

Aligning field: Aligns the string in the field.

Cutting field: When a string is too long, defined if it should be truncated left or right.

Padding field: Defines a padding character when the string is too short.

For numerical fields:

Multiplier: Allows to change the unit.

For decimal fields:

Decimal separator: Dot or comma.

Decimal number: Number of digits after the separator.

4. The new field shows up in the export format summary.

6.3.4 Format visualization

When a field is added, it is displayed in the format summary.

Num: Field order.

Position: Position of the field counting from the beginning of the line. *(N/A)* in CSV if there is no length specified.

Length: Field length. *(N/A)* in CSV if there is no length specified.

Field: Selected field type.

Example: Short string describing the field ('A' for text, '1' for numerical, '31/12/2007' for dates, '17:23:54' for hours, 'value1/value2' for options).

To test the format, enter a string in the validation field. The string will then be parsed according to the defined format, allowing to check fields consistency.

6.3.5 References

Some fields can be exported using a map, associating a value in the database with a string to export. References can be defined for the following fields:

- Product
- Department
- Site
- Tank
- Terminal
- Pump

The example below shows an export using a product reference.

6.3.6 Example

- Global format configuration:

Format Type	CSV
Export cancelled or modified transactions	Enabled
Separator	Semi-colon (;)
Fields with text separator	Strings
Text separator	Double quote (")
Date	ddmmyyyy
Hour	hh:mm:ss

This is a CSV export format with ';' as field delimiter. String fields will be sourrounded by "".

- Fields visualization:

Num	Position	Length	Field	Example
1	001 - 011	9	(Constant value)	["monexport"]
2	013 - 020	8	Date	[31122007]
3	022 - 031	10	Volume (L)	[00000001,0]
4	033 - 054	20	Registration - Vehicles	[".....A"]
5	056 - 067	10	Name - Products	[“A ”]
6	*(N/A)*	*(N/A)*	Option : External	[EXT/INT]

  Six fields were defined: a constant value, the date, the volume, the vehicle registration, the product name, and an options field.

  Field 3 was padded with '0', field 4 with '.' and field 5 with ' '. Position and length of the option field are *(N/A)* because the field length was ot specified.

- Reference: The value '78' was defined for the product FUE.

- Exported file

  For a transaction made on february 18th 2008, on an external fuel (FUE) terminal, with a 50L volume by the vehicle registered as 1033ET27, the exported will be:

  "monexport";18022008;00000050,0;".....1033ET27";"78 ";EXT

"monexport": Constant character string, surrounded with double quotes.

18022008: Date with format ddmmyyyy.

00000050,0: 50L volume, padded with '0' on 10 characters, with one decimal digit.

".....1033ET27": Vehicle registration, padded with '.' on 20 characters, surrounded with “.

“78 “: FUE product reference, padded with ' ' on 10 characters, surrounded with double quotes.

EXT: External option. If the transaction had not been external, it would have been 'INT'.

Chapter 7

Imports

GIR Titan-Hykkoris allows to import transactions, vehicles or drivers. Configuration of import formats is similar for all import types, and is described in the next section. Each import has speficities which are detailled in the following sections.

7.1 Import format definition

Import formats allow to import a file in a user-defined format.

They can be configured in the ▷ Settings menu. This page shows the list of all formats currently defined.

The format global configuration can be edited by the Modify button. The format definition is available in the format sheet (by clicking on the format name).

7.1.1 Global configuration

Global parameters define fields formats, and date/time formats:

Name: Name of the import format.

Format type: CSV or fix. CSV format allows variable length fields, whereas fix format requires a length for each field. When using a fix format, CSV parameters are ignored.

CSV-Separator: Defines the CSV field delimiter.

CSV-Text separator: Defines the CSV text delimiter.

Fields specific to transactions import:

Site: Defines the site to which imported transactions will be assigned.

Date: Defines the date format. Characters y,m,d are interpreted as year, month and day. Other characters are interpreted as delimiters. Example: yyyy/mm/dd.

Hour: Defines the hour format. Characters h,m,s are interpreted as hour, minute and second. Other characters are interpreted as delimiters. Example: hh:mm:ss.

Fields specific to vehicles or driver import:

Import mode: Defines how the format is imported (See 7.3.1, page 63).

Reference field: Defines a reference field for synchronized import (See 7.3.1, page 63).

7.1.2 Format definition

Fields types

Each field can be one of:

Constant value: Constant characters string.

Date - hour: Date and hour formats are defined in the import global configuration.

Other fields: Field that will be imported.

Each field can be present at most once for the format to be valid.

Field insertion

1. Click on Add ().
2. Select a field type and a length (optional in CSV), and validate.
3. Depending on the selected field type, additionnal entries may be required:

Ignore 0 beginning field: Import will ignore zeroes at the beginning of the field.

Multiplier: Allows to change the unit.

Filter: Defines the field as a filter. A new form shows up after validation (see below).

Ignore on modification: Synchronized import will use this field only when creating a vehicle or driver.

Filter action: Ignore: on import, if the field value is one of the values in this list, the line is ignored and noted as filtered. Accepter: on import, if the field value is one of the value in this list, the line will be imported. Otherwise, it will be ignored and noted as filtered.

Values: Enter filter values and validate.

Optional values: Enter possible values.

Boolean value: Enter the values to interpred as 'true' and the values to interpred as 'false'.

4. The new field shows up in the import format summary.

7.1.3 Format visualization

When a field is added, it is displayed in the format summary.

Num: Field order.

Position: Position of the field counting from the beginning of the line. *(N/A)* in CSV if there is no length specified.

Length: Field length. *(N/A)* in CSV if there is no length specified.

Field: Selected field type.

Example: Short string describing the field ('A' for text, '1' for numerical, '31/12/2007' for dates, '17:23:54' for hours, '(+) value1/value2/...' for accepting filters, '(-) value1/value2/...' for ignoring filters).

To test the format, enter a string in the validation field. The string will then be parsed according to the defined format, allowing to check fields consistency.

7.1.4 Example: a transactions import format

- Global format configuration:

Name	MyFormat
Site	MySite
Format type	Fix
Date	yyyymmdd
Hour	hhmm

- Fields visualization:

Num	Position	Length	Field	Example
1	001 - 022	22	(Constant value)	[“ ”]
2	023 - 030	8	Date	[20071231]
3	031 - 034	4	Hour	[1723]
4	035 - 038	4	(Constant value)	[“ ”]
5	039 - 042	4	(Constant value)	[“ ”]
6	043 - 044	2	Product ref.	[A]
7	045 - 064	20	(Constante value)	[“ ”]
8	065 - 070	6	Volume (L)	[000.01]
9	071 - 085	15	(Constant value)	[“ ”]
10	086 - 097	12	Price	[0000000.0001]
11	098 - 115	18	(Constant value)	[“ ”]
12	116 - 122	7	Meter	[0000001]
13	123 - 131	9	Registration - Vehicles	[“A ”]
14	132 - 132	1	(Constante value)	(+)[F]

This format defines 7 fields (date, hour, product, volume, price, meter and vehicle registration), and a filter.

The (Constant value) fields define ignored zones in the imported file. Field 8 (Volume) will be imported with a 0.0001 multiplier. If the value in the file is 23145, the value imported in the transaction will be '2.3145'. Field 14 is a one-character filter. Only lines containing a 'F' as this position in the imported file will be processed, others will be filtered out.

- References: The value '10' was defined as reference for product FUE.
- Line of the imported file

Offset	`........10........20........30........40........50........60.......`
	`1...5....0....5....0....5....0....5....0....5....0....5....0....5..`
Line	`467449F ZZZZTOTAL00000199907151542 70FRFFRF1000 000924739500000190006`
Offset	`.70........80........90.......100.......110.......120.......130`
	`..0....5....0....5....0....5....0....5....0....5....0....5....0..`
Line	`3000011152358000000000000247390 00001904963000011102358001234RD69 F`

- Imported transaction: The vehicle with registration 1234RD69 bought 63 litres of FUE on july 15th 1999 at 15h42, for 24.739 euros.

7.1.5 File import

Import can be launched manually in the ▷ Misc. menu.

An import consists of 4 steps:

- Selection of the import format: Select a format and validate.
- Selection of the file to import: Use the file browser to find the file to import on your computer, then select the file and validate.
- Pre-import report: When a format and a file have been selected, a report gives informations on what will be imported. The 5 first errors are displayed, and you can click on (...) to show more errors. On each row, there is the line where the error occured, and the concerned field. Confirm the operation to launch the real import.
- Import report: Once the real import is completed, the final report shows if errors occured, the number of transactions imported, in double, filtered or ignored (with error detail).

7.2 Transactions import

Transactions imported from a file are visible in the ▷ History menu. They are prefixed with [Imp] to distinguish them from transactions retrieved from a terminal.

To compute reports on imported transactions, it can be useful to define a site for each import format (generally with the name of the oil company providing the file), to specify where the transactions were made.

The site can then be used as a filter in ▷ Reports.

7.2.1 Products references

A transaction can be imported only when a product is defined. If no product is defined in the import, the special value '-' is applied to all imported transactions. It is necessary to define at least one product reference for an import format to be valid.

7.2.2 Format validity

A format is valid if:

- A Date field is defined.
- A Vehicle field is defined.
- No field is defined twice.
- At least one product reference is defined.

7.2.3 Transactions import report

The report defines:

- how many transactions will be imported
- how many transactions are in double (already existing in the database)
- how many transactions will be filtered
- how many transactions are ignored because of an error

When importing transactions, the following errors can occur:

- Date: invalid date format.
- Vehicule: vehicle not resolved.
- Product: product not resolved.

7.3 Vehicles/Drivers import

Vehicles (or Drivers) import allows to create vehicles or drivers from a file, or to synchronize their fields, depending on the selected import mode.

Note: in this section, the terms Vehicle and Driver are exchangeable.

7.3.1 Import mode

The import mode is defined in the import format global configuration.

Import mode: defines if the format is an initial or a synchronized import format (only one synchronized format can be defined).

Reference field: defines the field used as reference to resolve the vehicle during synchronization.

Initial import

This import is used to insert new vehicles in GIR Titan-Hykkoris database. Errors generated during such an import can be caused by:

- an incompatibility between the file and the defined format. In that case, the affected line and field are specified.
- an incompatibility with the current database, i.e. a double definition or the same registration (resp. name), badge or code.

Synchronized import

This import allows to update regularly one or several fields in vehicles. Only one synchronization format can be defined at the same time.

Synchronized import automatically imports the file `[hl]\input\vehicles.txt`. The import starts only when the file `[hl]\input\vehicles.ok` exists. This allows to be sure that `vehicles.txt` is complete. [hl] is the default directory in which GIR Titan-Hykkoris is installed (`c:\hl` by default).

Vehicles can be updated by a synchronized import when the Synchronized option is checked. In that case, all synchronized fields are read-only in GIR Titan-Hykkoris. To edit them, the Synchronized option must be unchecked.

Synchronized vehicles that are not found in the imported file are automatically forbidden. If they come back in a later file, they are authorized again (unless a manual change occured on the vehicle authorization).

7.3.2 References

References can be defined for all vehicles fields referencing another table (e.g. department, vehicle type, custom fields...).

If a reference is not found during the import, an element is automatically created with the imported value.

If two fields Department and Department ref. are defined, only the first field in the format order will be imported.

7.3.3 Format validity

An invalid format is displayed in red in the formats list. The error detail is visible in the format sheet.

A format is valid when:

- A reference field is defined.
- The Registration (resp. Name) field is defined.
- No field is defined twice.

7.3.4 Default format

In ▷ Settings, Vehicles import formats, click on Create a default format (). A format will be created with all fields visible in the vehicles list.

7.3.5 Vehicles import report

The report defines:

- how many vehicles will be created (or synchronized)
- for synchronized import: how many vehicles will be forbidden
- how many dependencies will be created (Departments, Vehicles types, ...).
- lines that caused a warning or an error

During vehicles import, the following errors can occur:

- Vehicle: the reference field is empty.
- Vehicle: the registration (resp. name) is empty.
- Key: the key already exists.
- Code: the code already exists.
- Number: the number already exists.

Chapter 8

Database management

GIR Titan-Hykkoris uses an ISAM database format. The database is made of multiple tables, located in the `tables` subdirectory.

Several tools are available for database management:

- A backup tool, in ▷ Misc., Backup.
- A purge tool, in ▷ Misc., Purge.

8.1 Backup

This menu is used to manage database backups. It offers the following features:

- Create a manual backup
- Trigger an automatic backup
- View and download previous backups
- Restore a previous backup

A backup file has the extension `.HLB`. The typical size of a backup file is 1 to 10 MB. It can be more for large databases.

Backup files can be created:

- Manually, by clicking on Create a new backup file. The file is created in the `temp` subdirectory.
- Automatically, by defining a daily backup hour in the configuration (See 2.2.3, page 20). The file is created in the `temp` subdirectory, and copied to the copy directory, if it is defined in the configuration.

  When a copy directory is configured, the automatic backup can be triggered manually at any time, by clicking on Trigger automatic backup.

The backup menu shows all backup files found in the `temp` subdirectory, most recent first. Click on Download to download a backup file. Click on View to view a summary of the backup file content.

To restore a backup, you must be connected locally on the GIR Titan-Hykkoris server. When it is so, click on View to show the backup file summary, then click on Restore and confirm to restore the database with the selected backup file.

CAUTION: The existing database is entirely overwritten by this operation.

8.2 Purge

The purge operation can be used to permanently remove old data from the database. It is only available to manager users.

Purge can be performed on two types of tables:

- History tables: The purge will delete all records older than the selected date, in the selected tables (Transactions history, Events, ...).
- Data tables: The purge will delete all records that are marked as *Forbidden* in the selected tables (Vehicles, Drivers, ...), and that have no transactions left.

CAUTION: Purged data can't be recovered.

The only way to restore purged data is to restore the whole database from a previous backup. An automatic database backup is done before any purge operation.

Example: In January 2008, we want to delete all transactions made during the year 2006 and before. This can be done by entering `01/01/2007` in the Time field, and ticking the *Fuel transactions* item.

If we also tick the *Vehicles* item, this will delete all vehicles that have the *Forbidden* option set, and that don't have any transaction after the `01/01/2007`.

Chapter 9

Specific features

9.1 Maintenances

9.1.1 Maintenance types

▷ Settings, Maintenance types

Maintenance types allow to classify vehicles maintenances.

Name: Unique name of the maintenance. This field is used to refer to the maintenance type in the application.

Type: Defines on what the maintenance is done: date, odometer or hour meter.

9.1.2 Maintenances

▷ Misc., Maintenances

This page shows maintenances according to various criterions. By default, only expired maintenances are displayed, sorted by maintenance type. Several queries can be made, to show maintenances by department, vehicle type, or maintenance type.

If a maintenance is defined on a vehicle which doesn't have a compatible meter, a message is displayed and invalid maintenances are shown in red, with empty Current value and Term cells.

9.2 MFG keys

MFG keys are identified by an 8 digits number, beginning with 7, to enter in the Key field of vehicles or drivers. This number is both written on the key an stored in the chip.

9.3 EMG keys

EMG keys are identified by an 8 digits number, beginning with 2 or 3, to enter in the Key field of vehicles or drivers.

This number is written on the key, but not in the chip. Thus, there must be a way to translate a key number into the corresponding chip code. This is achieved by proxi files.

9.3.1 Proxi files

Proxi files are text files, that contain on each line a key number and the associated chip code. They must be put in the `proxi` subdirectory.

GIR Titan-Hykkoris has an internal list of key numbers and chip codes that covers a large number range. Hence, installing proxi files is generally not required.

The Check keys submenu in the support menu allows to control the key number / chip code associations. It shows the following information:

- Which key numbers have are supported internally
- The list of detected proxi files
- The list of discrepancies between proxi files and the database, in Vehicles consistency and Drivers consistency.

9.4 ISO2 keys

ISO2 keys are magnetic keys that can be configured in ▷ Settings, Keys. The key length depends on the configuration.

9.5 BS125 keys

BS125 (ground loop) keys are identified by a 10 digits hexadecimal number. Theses keys can by used in addition to EMG or MFG keys.

Using BS125 keys adds new fields in ▷ Settings, Terminals:

Command duration: Command duration to signal the detection of a BS125 key.

Memo duration: Maximum duration between to detection of the key an its usage on the terminal.

***EMG/MFG compatible*:** When this option is checked, the terminal allows to use both BS125 and EML (resp. MFG) keys. Fuelers keys are the considerd as EMG (resp. MFG) keys.

***Loop*:** Loop number associated to a pump.

When both BS125 and EMG/MFG keys are used, a new field appears in vehicles or drivers, allowing to select the key type.

9.6 Temporary license

Some versions of GIR Titan-Hykkoris can have a time-limited renewable license. In that case, the license validity date is displayed in the About screen.

Once the license has expired, the application becomes unusable until a new license code is entered. A warning is displayed on the dashboard two weeks before a license expires.

To enter a new license code, go to the About menu, and click on the validity date.

9.7 Custom fields

GIR Titan-Hykkoris allows to define custom fields for vehicles and drivers.

In ▷ Settings, Custom fields, 3 custom fields can be defined for both vehicles and drivers. Each custom field consists of a name, and a list of values.

When a vehicle (or driver) custom field has a name defined and at least one element, the field shows up in the vehicle (resp. driver).

Custom fields can also be imported in vehicles or driver import (See 7.3, page 63), and used as criterions in reports (See 4.1.5, page 36)

9.8 Site access

This feature allows to restrict a vehicle or driver to a single site.

9.9 AEAT export

GIR Titan-Hykkoris can generate files for Spanish fuel tax partial refund (AEAT: Agencia Estatal de Administración Tributaria).

9.9.1 Configuration

First, AEAT export must be enabled in the ▷ Settings, Config. menu.

The following fields are available:

Format: Defines the format to use

- (N/A): Export disabled
- *CAE (.csv)*: CSV file export, for fuelings in private stations.
- *CIM (.xml)*: XML file export, for professional gasoil card emitting entities.

NIF: NIF code of the company that owns the fuel stations. This field is only used by CIM export.

CodEE: Card emitting entity code of the company that owns the fuel stations. This field is only used by CIM export.

Enabling AEAT export causes the following fields to appear:

- In Products:
  - Code: Product code to use in the exported file
- In Sites:
  - CAE or CIM (depending on the export type): Site code to use in the exported file
- In Vehicles:
  - NIF (CIM export only): NIF code of the company that owns the vehicle. When this field is not defined, the NIF code is inherited from the global configuration.
  - Option *AEAT export*: Enables export of the transactions made by this vehicle.

9.9.2 File generation

File can be generated in the ▷ Misc., AEAT export menu.

Select a time range, then validate. The file is generated in the `output` subdirectory, and can also be downloaded.

A fueling is exported when it matches the following conditions:

- the transaction was stored during the selected time range
- the volume is not zero
- the product has a code defined
- the vehicle option 'AEAT export' is checked
- for CAE export: the site has a CAE defined
- for CIM export: the site has a CIM defined

9.9.3 CAE file format

CSV file with 7 fields delimited by ';'

Position	Description
1	Transaction unique identifier
2	Site CAE code
3	Fueling date (format `AAAAMMDD`)
4	Fueling hour (format `HHMM`)
5	Vehicle registration
6	Product code
7	Volume

9.9.4 CIM file format

XML file with the following structure:

```
<SuministrV2Ent ... CodEE="" ...>
  <Suministro>
    <IdMovCont>...</IdMovCont>
    ...
    <Matricula>...</Matricula>
  </Suministro>
</SuministrV2Ent>
```

where `CodEE` is defined in the global configuration.

Each transaction is a `<Suministro>` markup, and contains the following elements:

Name	Description
`IdMovCont`	Transaction unique identifier
`CIM`	Site CIM code
`Fecha`	Fueling date (format `AAAAMMDD`)
`Hora`	Fueling hour (format `HHMM`)
`CodPro`	Product code
`Lit`	Volume
`NIF`	Vehicle NIF code if defined, or global NIF otherwise
`Matricula`	Vehicle registration

Appendix A

System prerequisites

A.1 Overview

GIR Titan-Hykkoris is installed on a single computer, which will be refered in this document as "GIR Titan-Hykkoris server". GIR Titan-Hykkoris is used through a simple web browser.

Communication with terminals always occurs between the GIR Titan-Hykkoris server and terminals, independently of the computer on which the web browser is installed.

GIR Titan-Hykkoris can be installed:

- On a server (dedicated or not): multiple workstations can access the application with a web browser.
- On an isolated workstation: the application is still usable by accessing the web server through the loopback interface (127.0.0.1).
- On a networked workstation: the application can be used on the computer where it is installed as well as on other computers on the same network.

Under Windows, an icon located near the clock in the taskbar makes it easier to access GIR Titan-Hykkoris from the computer on which it is installed. A double click on this icon directly launches a web browser with the appropriate location.

A.2 Database

GIR Titan-Hykkoris is written using C language, and uses an ISAM database format. It includes an automatic backup tool which can be configured to run at specific hours.

The size of a backup file varies from 500 KB to several MB (about 3 MB for a database with 1000 vehicles and 100 000 transactions).

Data export from GIR Titan-Hykkoris can be achieved in several ways:

- Export of fuel transactions in a HLF file.

- Export of any table directly from the user interface, by downloading a CSV file.

A.3 Minimal configuration

The GIR Titan-Hykkoris server requires the following minimal hardware configuration:

- Intel Pentium III or equivalent processor.
- 250 MB available on hard drive.
- 80 MB of RAM available for the application.
- Operating system:
  - Microsoft Windows 2000, XP, 2003 or Vista (Microsoft Windows 95, 98 and Me are also supported)
  - GNU/Linux kernel 2.2, 2.4 ou 2.6

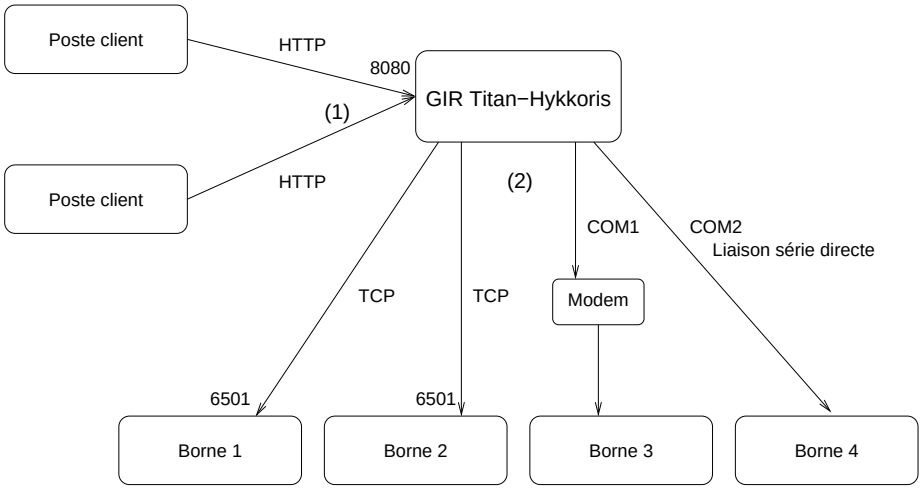
The client workstations must have a web browser:

- Microsoft Internet Explorer version 5 or above
- Mozilla Firefox version 1 or above

A.4 Storage capacities

- 16 products
- 3000 vehicles
- 3000 drivers
- 50 terminals
- 150 000 transactions (Terminal autonomy: 2500 transactions)

A.5 Network access



All the network traffic generated when using GIR Titan-Hykkoris can be classified among two types of queries:

1. Connections of client workstations to the GIR Titan-Hykkoris server: those are HTTP queries to TCP port 8080 of GIR Titan-Hykkoris integrated web server.

2. Connections to networked terminals: GIR Titan-Hykkoris connects to the terminal network interface (custom IP address or DNS name) on the TCP port 6501. Communications with terminals can be launched manually by the user, scheduled to be automatically runned at a given hour, of automatically launched when using "real time" mode.

   When the network link between terminals and GIR Titan-Hykkoris is down, fuel delivery is still working: each terminal is autonomous[1], and data is synchronized between terminals and GIR Titan-Hykkoris during dialogues. If a dialogue can't be run because of a bad network, synchronization will be postponed but the fuel delivery remains usable.

A.5.1 Traffic volume

Connection between client workstations and web server : similar to any standard web traffic. The size of HTML pages can vary from 10 KB to 1 MB (generally less than 100 KB).

Connection to terminals :

1. Autonomous mode
   - About 20 KB to retrieve 100 transactions.
   - About 200 KB to send 500 vehicles and 500 drivers.
   - Up to 2 MB for a full initialization.

[1] Terminals have a storage capacity of 2500 transactions

- GIR Titan-Hykkoris handles a synchronization with terminals to minimize the amount of data to send. Hence, traffic volume during automatic dialogues will generally be limited to some KB. For example, a synchronization dialogue with a terminal containing 100 transactions will transfer around 20 KB during 30 seconds.

2. Real time mode Daily traffic (in KB):
   $15 + 0.3 * N * B$
   where N is the number of transactions per terminal and per day, and B the number of terminals

A.6 System access rights

- GIR Titan-Hykkoris doesn't require any software other than the operating system: il completely stands alone, and integrates its own web server [2].

  The application is launched by winhl.exe, which is displayed as an icon in the taskbar. TCP ports used are configurable in the winhl.ini file.

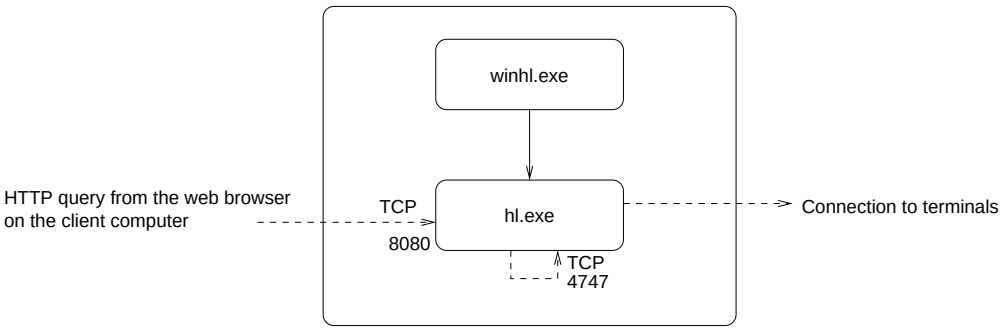
- GIR Titan-Hykkoris doesn't need any particular system right other than total access to its installation tree (`c:\hl` by default). It doesn't use the registry nor any shared DLL.

- Dialogue with terminals:

  for serial links or modems, GIR Titan-Hykkoris must be allowed to open the corresponding serial port.

  for network links, GIR Titan-Hykkoris must be allowed to establish a TCP connection to the terminal network interface.

Under Windows, GIR Titan-Hykkoris is composed of the following processes:



- winhl.exe: Launcher allowing to start hl.exe from an icon in the taskbar.
- hl.exe: Listens locally on TCP ports 8080 and 4747, connects to terminals using TCP or a serial port.

A Linux version is also available.

[2] doesn't require to install Apache, IIS or any other web server. If such a server is installed on the same computer as GIR Titan-Hykkoris, it will run independently (Both servers must listen on different TCP ports, the default port for the GIR Titan-Hykkoris server is 8080)

Appendix B

HLF formats

This appendix describes the HLF exported file formats.

B.1 HLF1 format

Fixed-size format, 617 characters per line. Each line ends with CRLF (0x0D, 0x0A) for a total of 619 characters per line.

All fields are enclosed in double quotes ("), and separated by a special delimiter, defined in the global configuration (',' or ';').

This file can be parsed either as a fixed or CSV format.

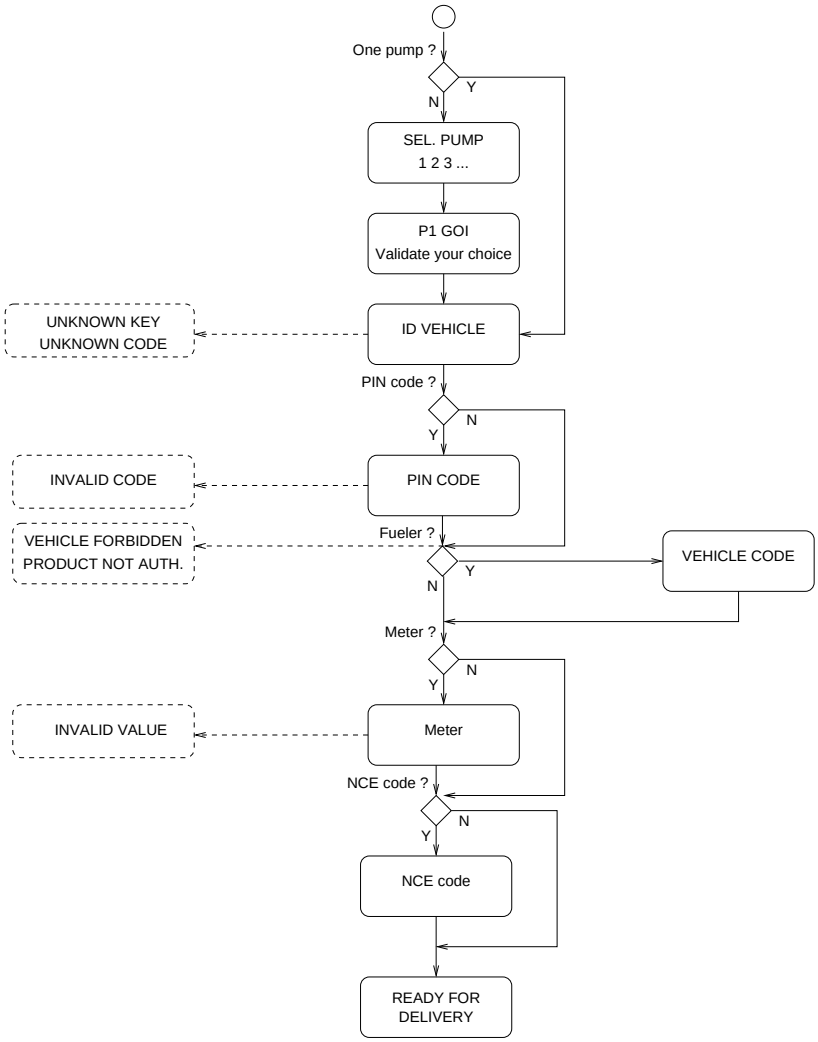
Num.	Offsets	Length	Description
01	000-010	9	Transaction id
02	012-021	8	Date (format `YYYYMMDD`)
03	023-030	6	Hour (format `HHMMSS`)
04	032-063	30	Vehicle registration
05	065-082	16	Vehicle key
06	084-093	8	Vehicle code
07	095-116	20	Department name
08	118-139	20	Vehicle type name
09	141-172	30	Driver name
10	174-195	20	Driver firstname
11	197-214	16	Driver key
12	216-225	8	Driver code
13	227-248	20	Driver department name
14	250-271	20	Driver type name
15	273-284	10	Product name
16	286-307	20	Terminal name
17	309-312	2	Pump number
18	314-335	20	Site name
19	337-340	2	Tank number
20	342-353	10	Volume (L * 100)
21	355-366	10	Unit price (Currency unit * 10000)
22	368-375	6	Duration (s)
23	377-379	1	Meter type ("K" or "H")
24	381-392	10	Meter value (km or h)
25	394-405	10	Covered value (km or h)
26	407-418	10	NCE code
27	420-437	16	Options 1: 'C' cancellation 2: 'R' replacement 3: 'M' manual entry 4: 'N' new meter 5: 'V' maximum volume 6: 'F' meter forced 7: 'E' external fueling
28	439-460	20	unused (reserved for future use)
29	462-483	20	unused (reserved for future use)
30	485-506	20	unused (reserved for future use)
31	508-529	20	unused (reserved for future use)
32	531-552	20	unused (reserved for future use)
33	554-565	10	Vehicle number

34	567-578	10	Driver number
35	580-591	10	unused (reserved for future use)
36	593-604	10	unused (reserved for future use)
37	606-617	10	unused (reserved for future use)

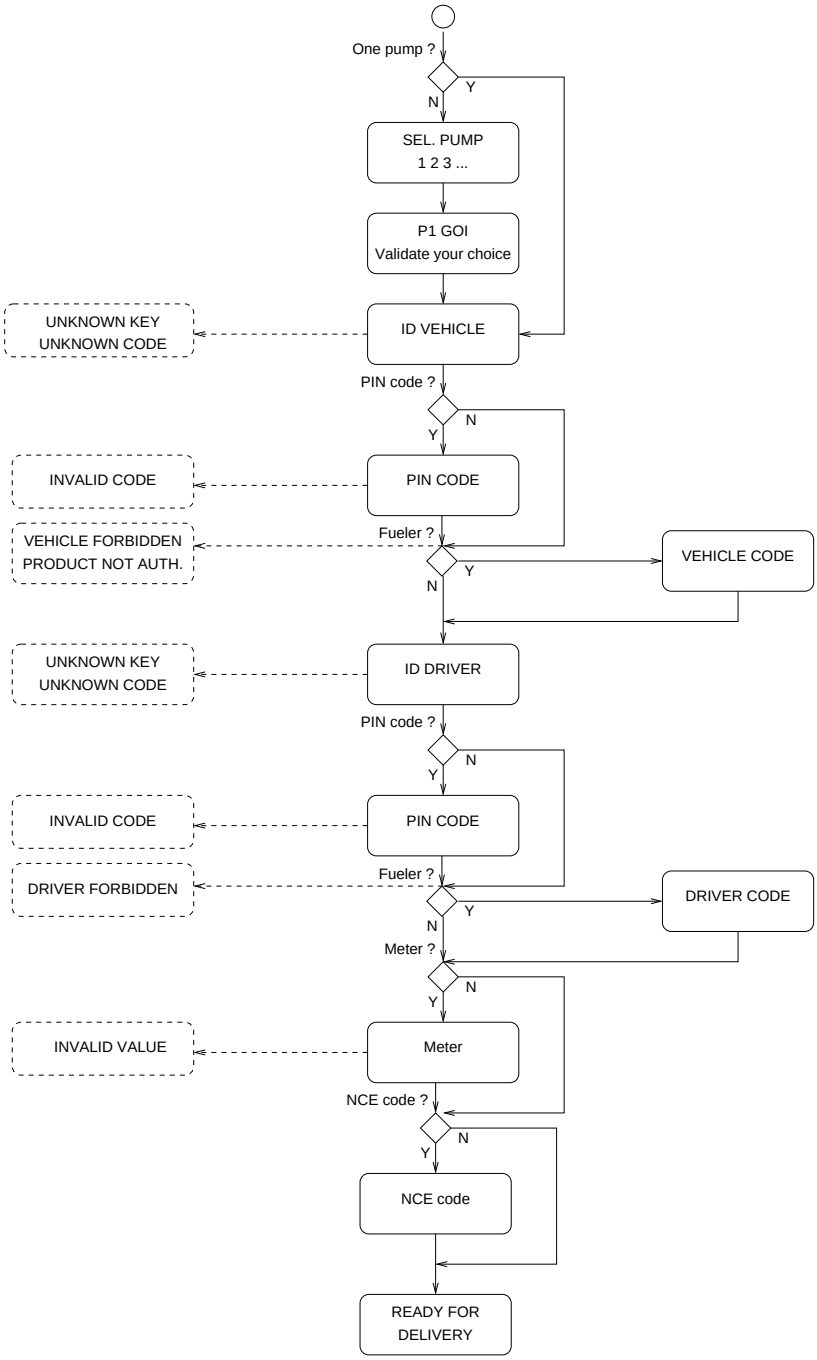
Appendix C

Fueling scenario

C.1 Vehicle only



C.2 Vehicle + driver



C.3 Driver + vehicle