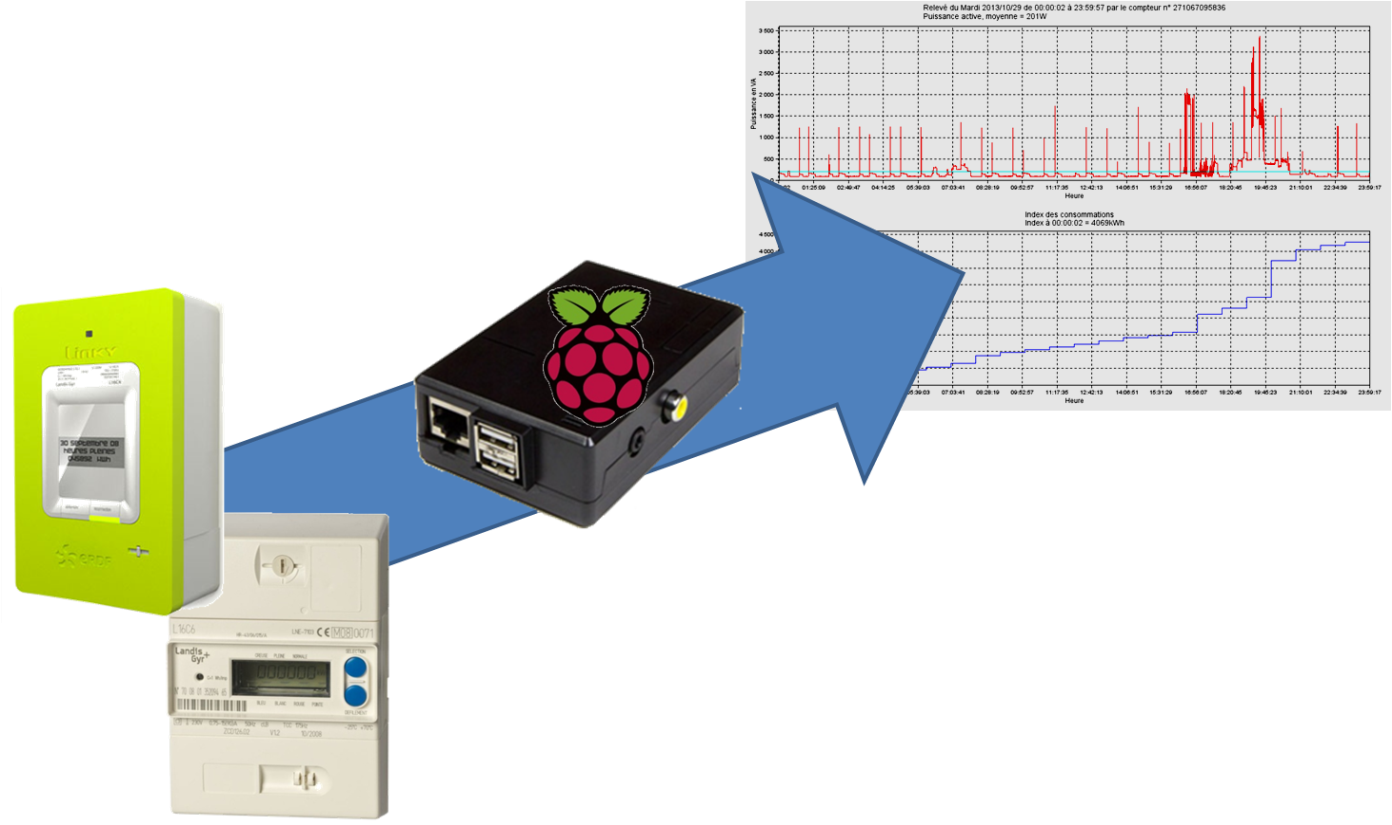
Sébastien LEMOINE <https://github.com/sebastien0/RPi.SuivitTRElec>

**SUPERVISION TEMPS-REEL DES CONSOMMATIONS ELECTRIQUES DOMESTIQUES**



**Date :**

**Révision :**

# HISTORIQUE

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Description** |
| A0 | 08/2014 | Création |
| A1 | 12/2014 | Correction schémas *A2 - Schéma électrique* et *A31 – Algorithme*  Ajout *Documentation des sources* |
|  |  |  |
|  |  |  |
|  |  |  |

# SOMMAIRE

[I. HISTORIQUE 2](#_Toc405543527)

[II. SOMMAIRE 3](#_Toc405543528)

[III. ILLUSTRATIONS 5](#_Toc405543529)

[III.1. Liste des figures 5](#_Toc405543530)

[III.2. Liste des tableaux 5](#_Toc405543531)

[IV. LEXIQUE 6](#_Toc405543532)

[V. DESCRIPTION 7](#_Toc405543533)

[VI. CONFIGURATION DE LA R-PI 8](#_Toc405543534)

[VII. ARCHITECTURE 9](#_Toc405543535)

[VII.1. A-0 - Architecture Générale 9](#_Toc405543536)

[VII.1.1. Schéma de principe 9](#_Toc405543537)

[VII.2. A1 - Compteur ERDF 10](#_Toc405543538)

[VII.3. A2 – Conditionnement 10](#_Toc405543539)

[VII.3.1. Schéma de principe 10](#_Toc405543540)

[VII.4. A3 - Acquisition – Stockage 12](#_Toc405543541)

[VII.4.1. Schéma de principe 12](#_Toc405543542)

[VII.4.2. A31 –Programme en C 12](#_Toc405543543)

[VII.4.2.1. Synoptique d’une trame 12](#_Toc405543544)

[VII.4.2.2. Algorithme 14](#_Toc405543545)

[VII.5. A4 – Transfert 15](#_Toc405543546)

[VII.5.1. Schéma de principe 15](#_Toc405543547)

[VII.5.2. A41 – Fichier csv 15](#_Toc405543548)

[VII.6. A5 – Exploitation 18](#_Toc405543549)

[VII.6.1. Schéma de principe 18](#_Toc405543550)

[VII.6.2. Programme sous Scilab 19](#_Toc405543551)

[VII.6.3. Schéma de principe 19](#_Toc405543552)

[VII.6.3.1. Algorithme 19](#_Toc405543553)

[VII.6.3.2. Liste des fonctions disponibles 20](#_Toc405543554)

[VII.7. A6 – Affichage 21](#_Toc405543555)

[VII.7.1. Schéma de principe 21](#_Toc405543556)

[VIII. ANNEXES 23](#_Toc405543557)

[VIII.1. Bibliographie 23](#_Toc405543558)

[VIII.2. Compteurs Testés 23](#_Toc405543559)

[VIII.3. BOM 23](#_Toc405543560)

[VIII.4. Fichiers générés 23](#_Toc405543561)

[VIII.5. Documentations des sources 23](#_Toc405543562)

[VIII.5.1. Code C 23](#_Toc405543563)

[VIII.5.2. Scripts Scilab 24](#_Toc405543564)

# ILLUSTRATIONS

## Liste des figures

[Figure 1 - A-0 - Schéma de principe 9](#_Toc405543565)

[Figure 2 - A2 - Schéma de principe 10](#_Toc405543566)

[Figure 3 - A2 - Schéma électrique 11](#_Toc405543567)

[Figure 4 - A3 - Schéma de principe 12](#_Toc405543568)

[Figure 5 – A31 - Constitution d’une trame 13](#_Toc405543569)

[Figure 6 – A31 - Algorithme 14](#_Toc405543570)

[Figure 7 - A4 - Schéma de principe 15](#_Toc405543571)

[Figure 8 – A41 - Fichier csv 16](#_Toc405543572)

[Figure 9 – Extrait du fichier csv pour une configuration Base 16](#_Toc405543573)

[Figure 10 – Extrait du fichier texte pour une configuration Base 17](#_Toc405543574)

[Figure 11 – Extrait du fichier texte pour une configuration HCHP 18](#_Toc405543575)

[Figure 12 - A5 - Schéma de principe 18](#_Toc405543576)

[Figure 13 - A6 - Schéma de principe 21](#_Toc405543577)

[Figure 14 – Documentation, configuration 24](#_Toc405543578)

[Figure 14 – Configuration, exemple de balises 25](#_Toc405543579)

## Liste des tableaux

[Tableau 1 – Configuration FTP 15](#_Toc405543580)

[Tableau 2 – Liste des fonctions disponibles 21](#_Toc405543581)

[Tableau 3 – Documentation, liste des balises 25](#_Toc405543582)

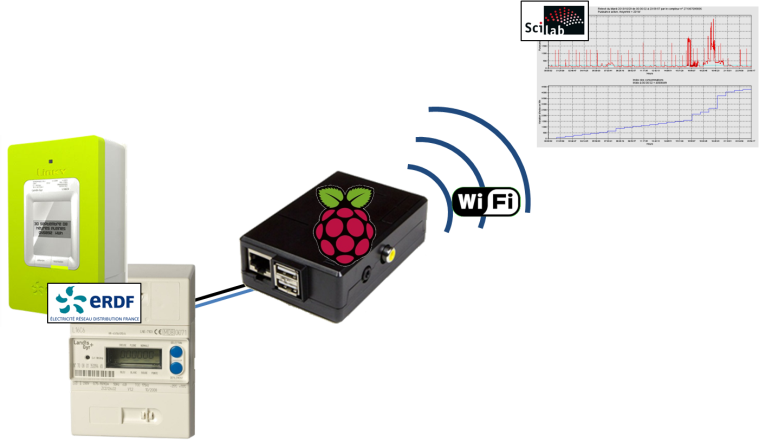
# LEXIQUE

| **Acronyme** | **Description** |
| --- | --- |
| **BOM** | **B**ill **O**f **M**aterials, nomenclature |
| **Checksum** | Somme de contrôle |
| **ERDF** | **É**lectricité **R**éseau **D**istribution **F**rance |
| **Fc** | **F**réquence de **C**oupure |
| **FTP** | **F**ile **T**ransfert **P**rotocol, protocole de transfert de fichiers |
| **OS** | **O**perating **S**ystem, système d’exploitation |
| **R-Pi** | **R**aspberry-**Pi** |
| **SSH** | **S**ecure **SH**ell |
| **TR** | **T**emps **R**éel |
| **UART** | **U**niversal **A**synchronous **R**eceiver **T**ransmitter, émetteur-récepteur asynchrone universel |
| **VNC** | **V**irtual **N**etwork **C**omputing |

# DESCRIPTION

Ce projet permet de consulter les consommations électriques en temps réel depuis un terminal connecté (ex. : ordinateur, Smartphone ou tablette). Une extension permet d’accéder aux fichiers de point et de réaliser des analyses approfondies avec Scilab ([www.scilab.org/fr](http://www.scilab.org/fr)).

Dans l’objectif d’avoir une nomenclature la plus économique possible, le projet se base sur le compteur électrique ERDF en tête d’installation électrique, d’une Raspberry-Pi ([www.raspberrypi.org](http://www.raspberrypi.org)) et de quelques composants électroniques.



Il est possible de se connecter sur tout type de compteur électronique possédant une sortie liaison série. Se reporter au *§VIII.2* pour connaître la liste des compteurs déjà testés.

# CONFIGURATION DE LA R-PI

Libérer UART

Client NTP

Dongle Wifi

Lancer le programme en SSH : ./compteur\_linky >> compteur.log &

# ARCHITECTURE

## A-0 - Architecture Générale

### Schéma de principe

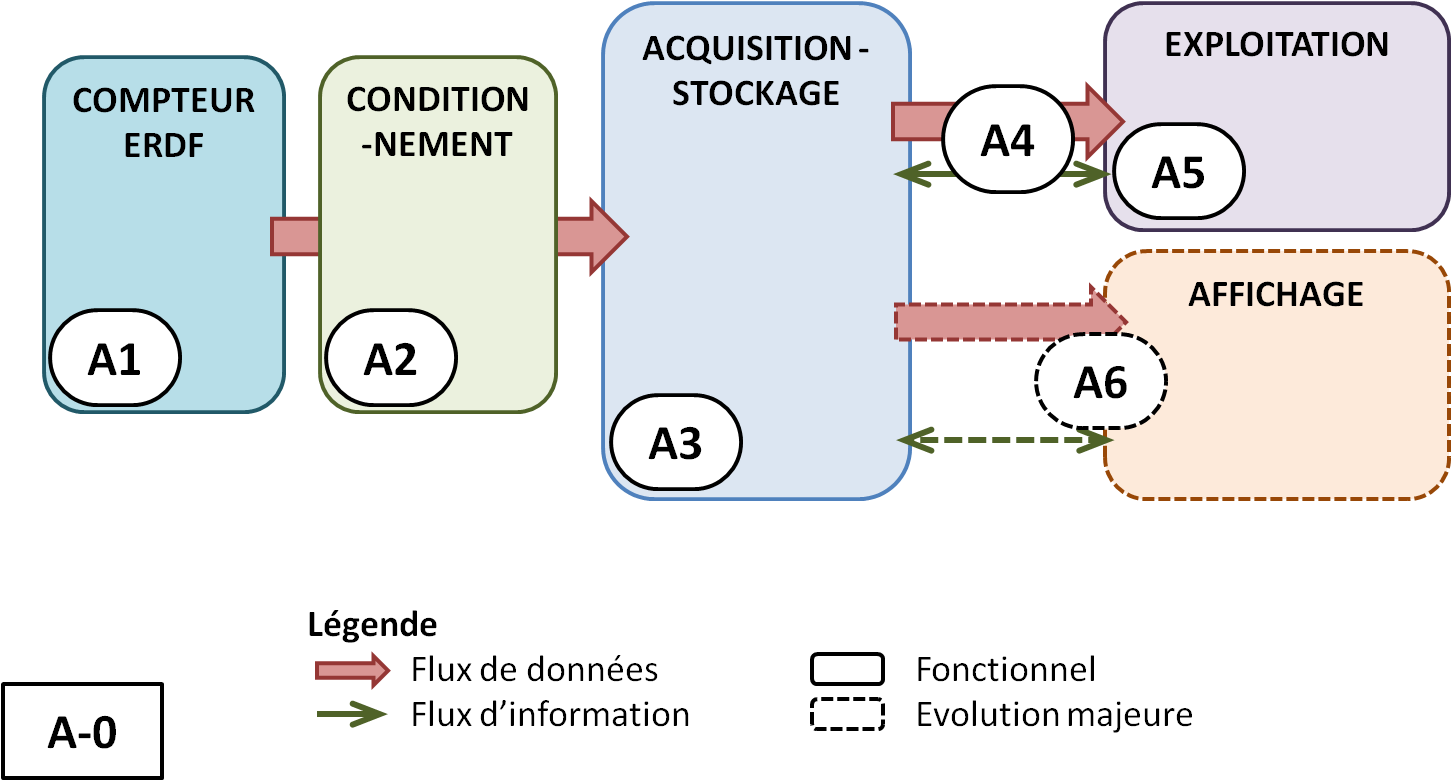


Figure - A-0 - Schéma de principe

|  |  |  |
| --- | --- | --- |
| **Référence** | **Nom** | **Description** |
| A-0 | Architecture Générale | Vue d’ensemble |
| A1 | Compteur ERDF | Acquisition des grandeurs électriques, échantillonnage, envoi sur liaison série |
| A2 | Conditionnement | Mise en forme électrique de la liaison série |
| A3 | Acquisition - Stockage | Décodage, horodatage, stockage, serveurs |
| A4 | Transfert | Transfert ponctuel pour exploitation |
| A5 | Exploitation | Analyses approfondies avec Scilab |
| A6 | Affichage | Affichage basique avec un navigateur web |

## A1 - Compteur ERDF

A partir des documents [D1] et [D2], les caractéristiques physiques des signaux de télé-information client, sous le couvre-bornes, sont les suivantes :

* source de courant,
* binaire,
* unidirectionnelle,
* vitesse de modulation 1200 bauds +/-1% ,
* durée égale des bits à « 0 » et à « 1 » ,
* fréquence de la porteuse 50kHz +/-3% ,
* logique de codage négative
  + un bit émis à "0" correspond à la présence de porteuse pendant le temps correspondant.
  + un bit émis à "1" correspond à l'absence de porteuse pendant le temps correspondant.

Après démodulation, la liaison asynchrone classique a les caractéristiques suivantes:

* vitesse de transmission 1200 bauds
* codage de toutes les informations sous forme ASCII (affichable)
  + 7 bits pour représenter un caractère ASCII
  + 1 bit de parité, parité paire (even)
* un bit de start avant chaque caractère <=> "0" logique
* un bit de stop après chaque caractère <=> "1" logique

Le contenu des trames est décrite dans le §VII.4.2.1.

## A2 – Conditionnement

### Schéma de principe

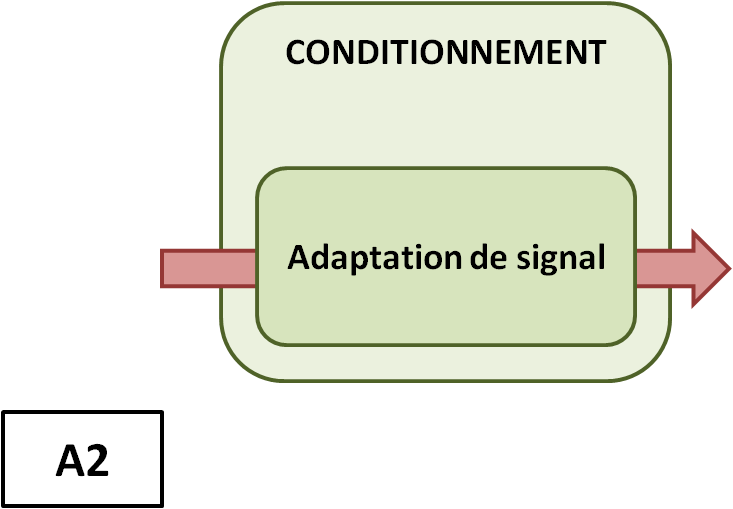


Figure - A2 - Schéma de principe

Cette fonction convertie un signal modulé en amplitude en un signal logique 3,3V. La BOM se trouve dans le §VIII.3.

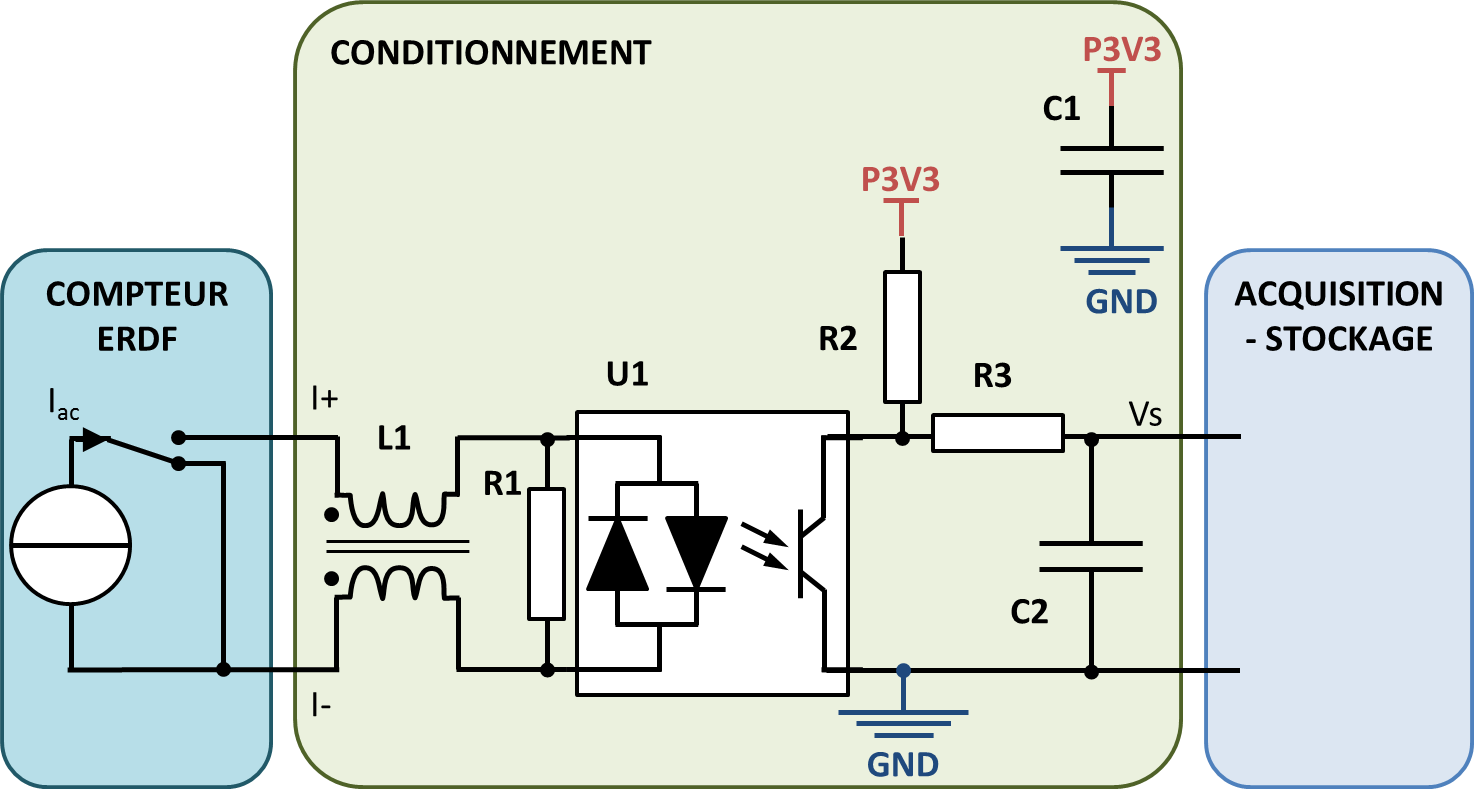


Figure - A2 - Schéma électrique

Nota : Le compteur ERDF est une modélisation électrique.

La fréquence du courant porteur Iac est à 50kHz. L’optocoupleur U1 est naturellement lent à la commutation et filtre le 50kHz d’où Vs un signal carré à logique positive.

L1 est un filtre de mode commun et le filtre composé par R2, R3 et C2 est fixé à fc ≈ 10\*fRS232 = 12kHz

## A3 - Acquisition – Stockage

### Schéma de principe

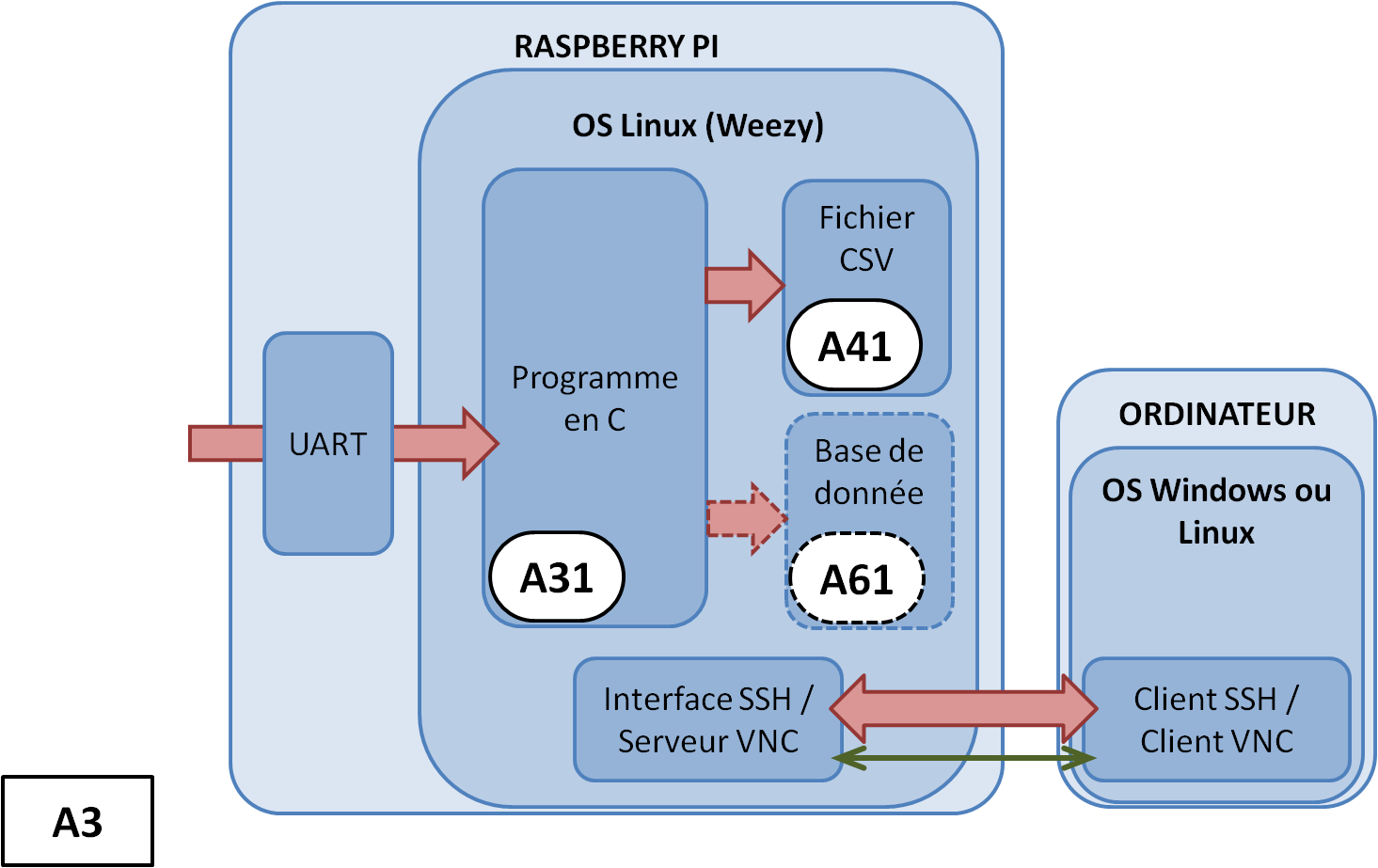


Figure - A3 - Schéma de principe

La R-Pi assure la communication unidirectionnelle avec le compteur, le décodage, l’horodatage et l’enregistrement des points de mesure.

L’interface SSH/VNC sert à la gestion à distance.

### A31 –Programme en C

#### Synoptique d’une trame

Les trames envoyées par le compteur ne sont pas horodatées et se composent comme décrit dans la . Envoyées en boucle, leur contenu peut varier selon le compteur ou la configuration utilisée (Base, HCHP, …).

L’algorithme du programme est donnée dans le *§*.

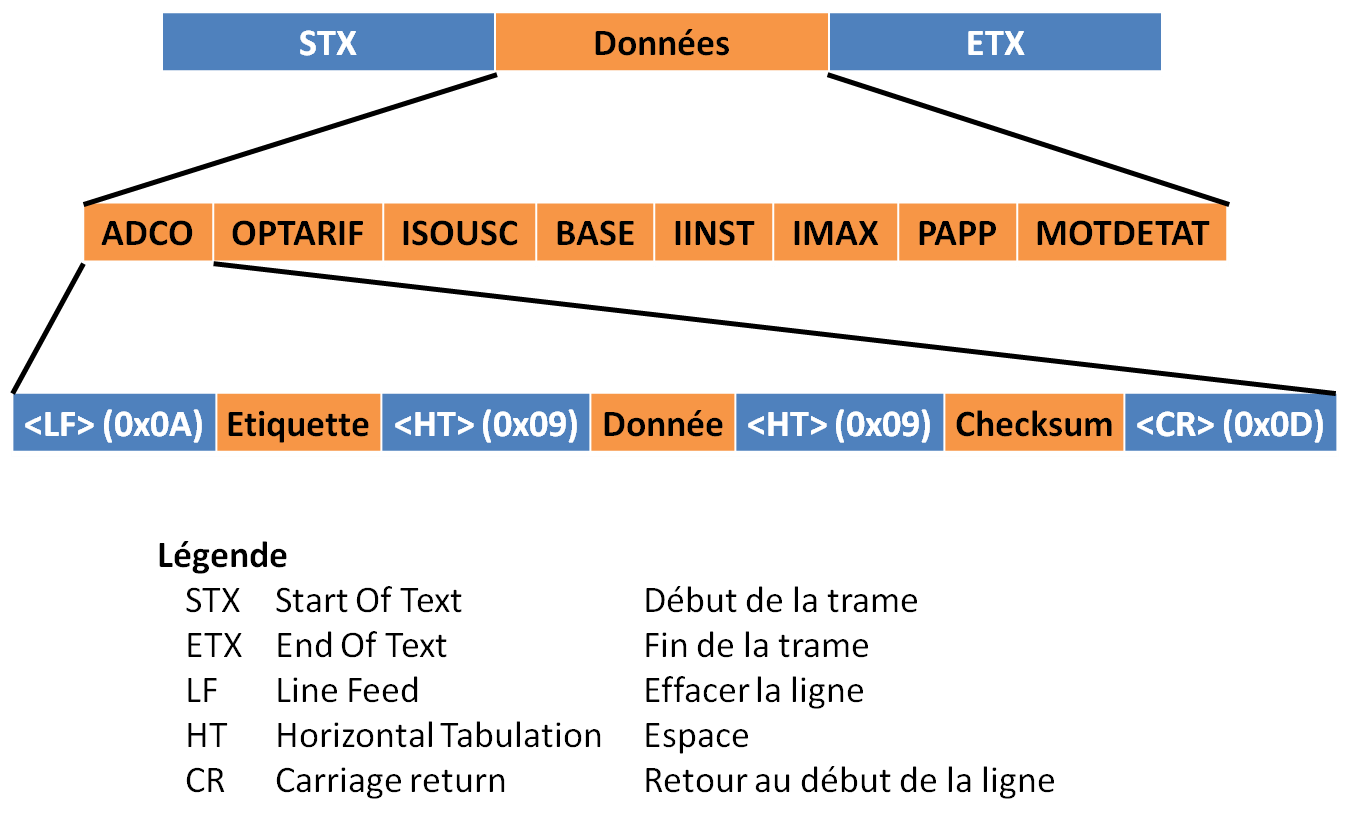


Figure – A31 - Constitution d’une trame

#### Algorithme

Le programme est structuré comme présenté par la .

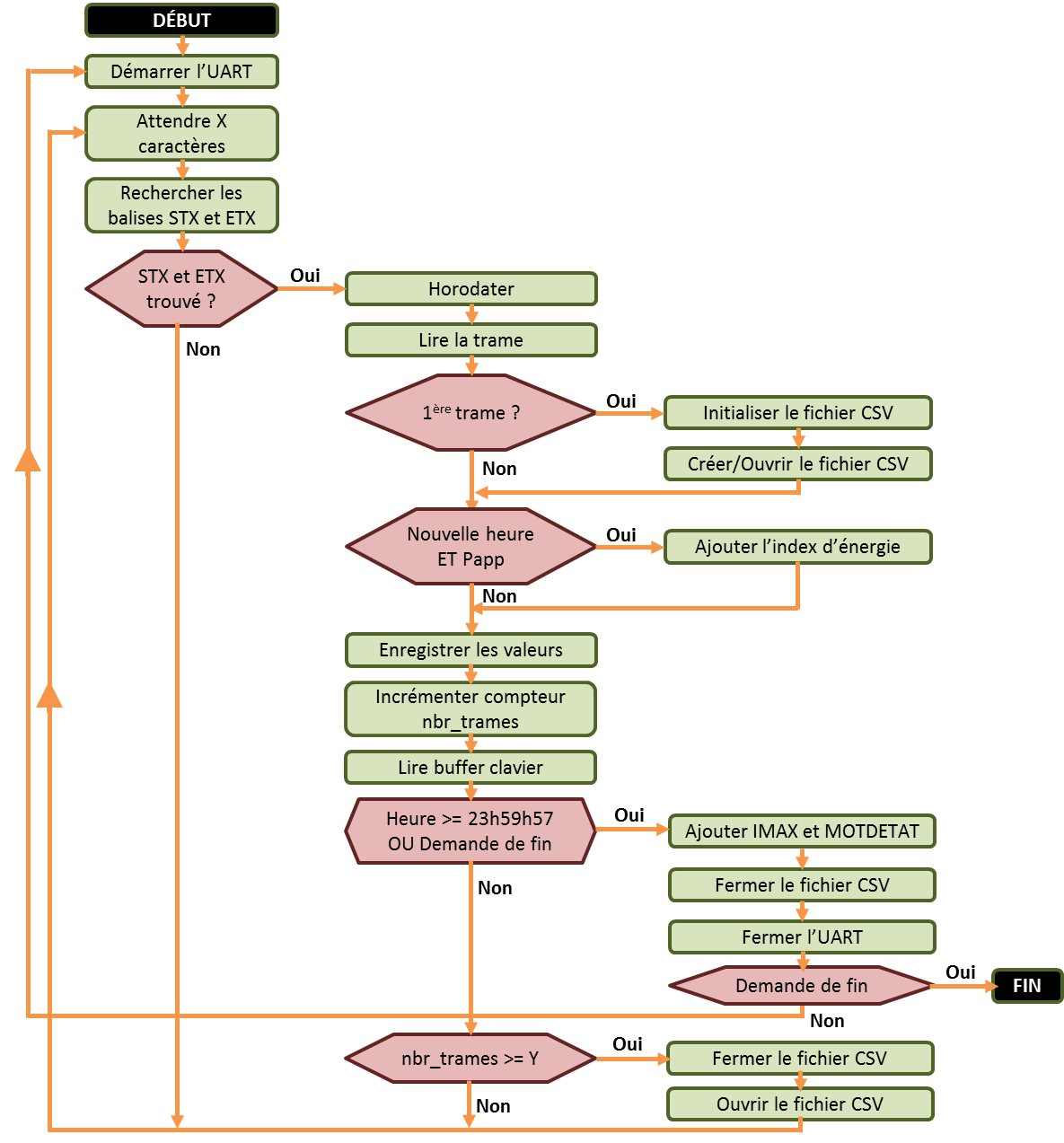


Figure – A31 - Algorithme

## A4 – Transfert

### Schéma de principe

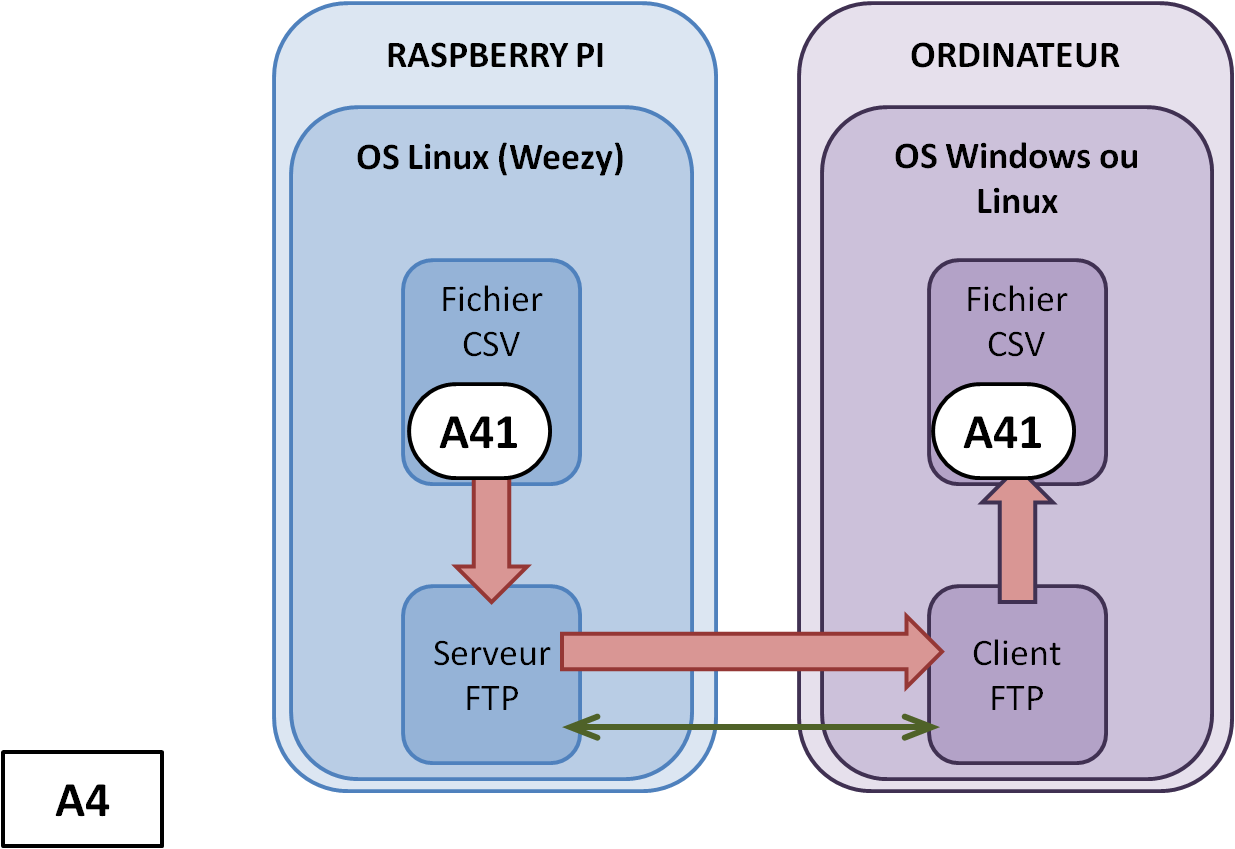


Figure - A4 - Schéma de principe

L’obtention du fichier de point s’effectue par le transfert des fichiers en utilisant le protocole FTP. Il est possible d’utiliser le client FileZilla (<http://filezilla.fr>)

Les informations de connexion sont les suivantes :

|  |  |
| --- | --- |
| **Description** | **Valeur** |
| Adresse IP de la R-Pi | 192.168.1.xxx |
| Nom du compte | pi |
| Mot de passe | xxx |
| Port | 22 |
| Protocole | SFTP |

Tableau – Configuration FTP

### A41 – Fichier csv

Le fichier CSV est un des 2 formats de sortie, le second étant la base de donnée (cf. §VII.7).

Le nom du fichier est composé de la sorte :

Releve\_aaaa\_MM\_jj.csv

Avec aaaa l’année sur 4 chiffres, MM le mois sur 2 chiffres et jj le jour sur 2 chiffres.

Le contenu du fichier varie selon la configuration du compteur mais il peut être illustré par la . Des exemples sont accessibles sous .*\Code\Compteur\_Linky\Releves\Fichiers csv*.

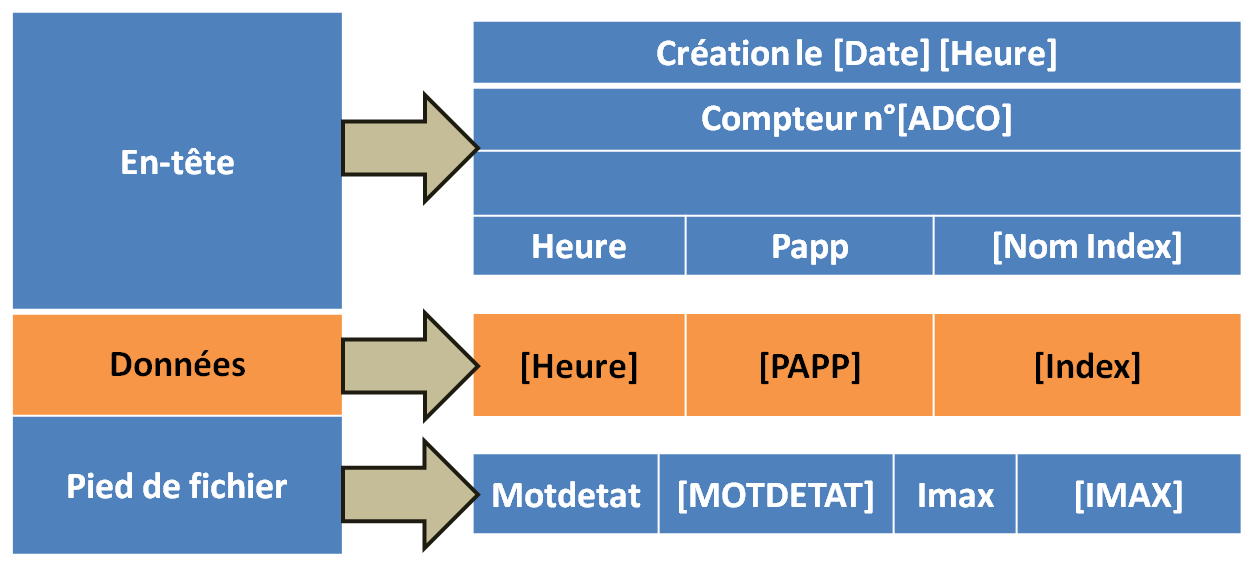


Figure – A41 - Fichier csv

Dans un souci de taille de fichier, les index d’énergie ne sont enregistrés que toutes les heures à xh00m00s ±2s avec x=[0…23]. Lorsque la PAPP n’est pas disponible (i.e. non envoyée par le compteur) l’index est enregistré à chaque trame, le champ Papp est alors vide.

Lorsque l’utilisateur demande l’arrêt du programme, le dernier index, le MOTDETAT et IMAX sont ajoutés.

La est un extrait d’un fichier csv pour un compteur configuré en Base.

Creation;2013/12/15

Compteur;271067095836;Isousc;30

Heure;Papp;Base

00:00:02;00091;004312814

00:00:05;00092;

00:00:08;00092;

00:00:10;00092;

00:00:13;00092;

[…]

00:59:57;00151;

00:59:58;00151

01:00:00;00152;004312900

01:00:01;00153;

01:00:03;00151;

[…]

23:59:56;00093;

23:59:57;00093;

23:59:58;00093;

23:59:59;00093;004316605

Motdetat;000000;Imax;010

Figure – Extrait du fichier csv pour une configuration Base

**ATTENTION** : En attendant une mise à jour, ce sont des *espace-tabulation-espace* et non un *point-virgule* qui sépare les valeurs ; aussi l’extension du fichier est .txt.

La est un extrait d’un fichier texte pour un compteur configuré en base et la pour une configuration HCHP.

[Invalide] est une synthèse sur la validité des trames. Ce calcul se base sur les cheksums reçus et calculés ; cette valeur est uniquement présente et égale à « 1 » ou « -1 » lorsqu’il y a un écart.

Creation le 2013/12/15 00:00:02

Compteur n°271067095836

Heure Papp Base Invalide

00:00:02 00091 004312814

00:00:05 00092

00:00:08 00092

00:00:10 00092

00:00:13 00092

[…]

00:59:57 00151

00:59:58 00151

01:00:00 00152 004312900

01:00:01 00153

01:00:03 00151

[…]

23:59:56 00093 1

23:59:57 00093 1

23:59:58 00093 1

00:00:00 00093 004316605 1

Motdetat 000000 Imax 010

Figure – Extrait du fichier texte pour une configuration Base

Creation le 2013/12/28 17:10:49

Compteur n°049701078744

Heure Papp H creuses H pleines Invalide

17:10:49 098437215 131783106 -1

17:10:51 098437215 131783109

17:10:53 098437215 131783111

17:10:54 098437215 131783112

17:10:55 098437215 131783114

[…]

23:59:55 098441046 131797583

23:59:56 098441046 131797583

23:59:58 098441046 131797583

23:59:59 098441046 131797583 -1

Motdetat 000F00 Imax 049

Figure – Extrait du fichier texte pour une configuration HCHP

## A5 – Exploitation

### Schéma de principe

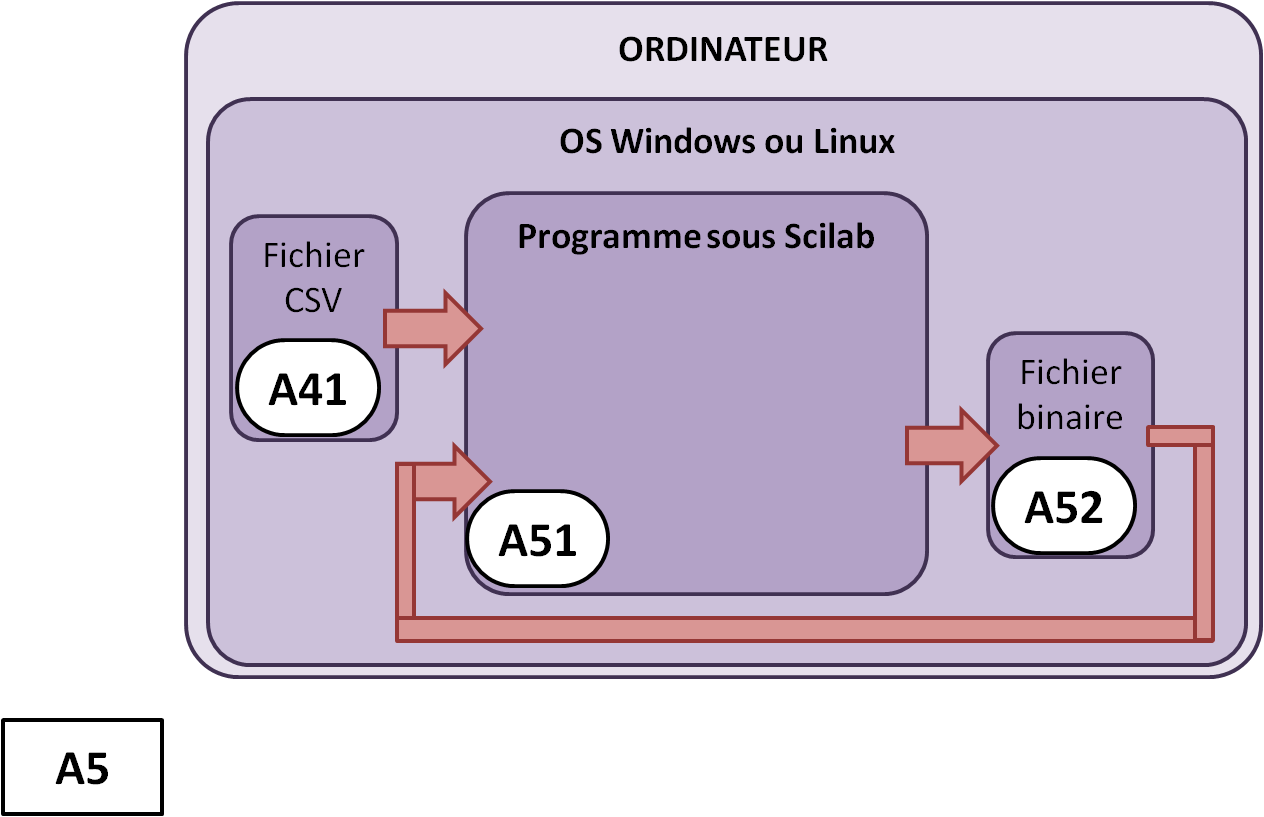


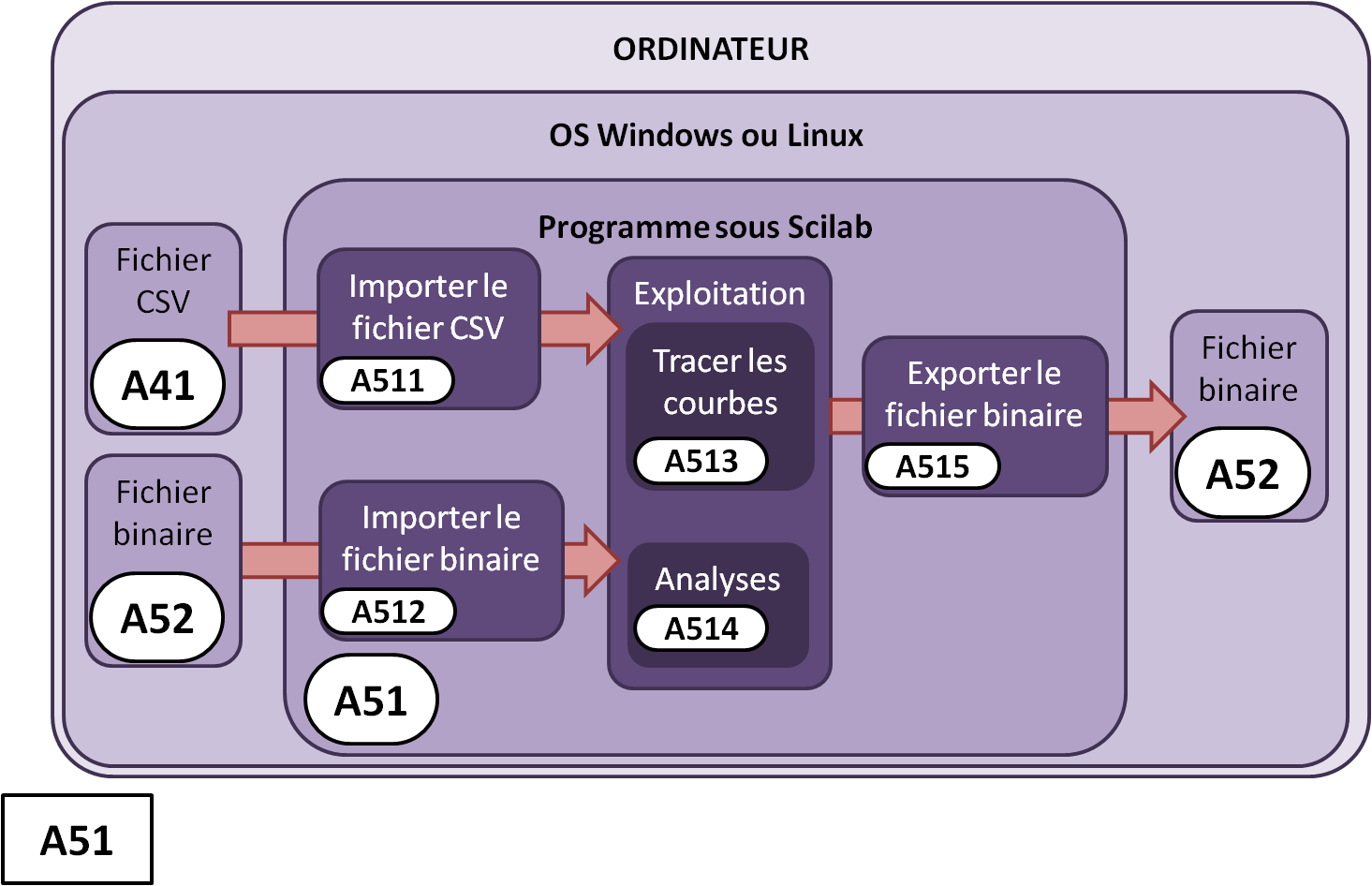
Figure - A5 - Schéma de principe

L’exploitation des relevés se fait à partir de scripts Scilab décrit dans les paragraphes suivants.

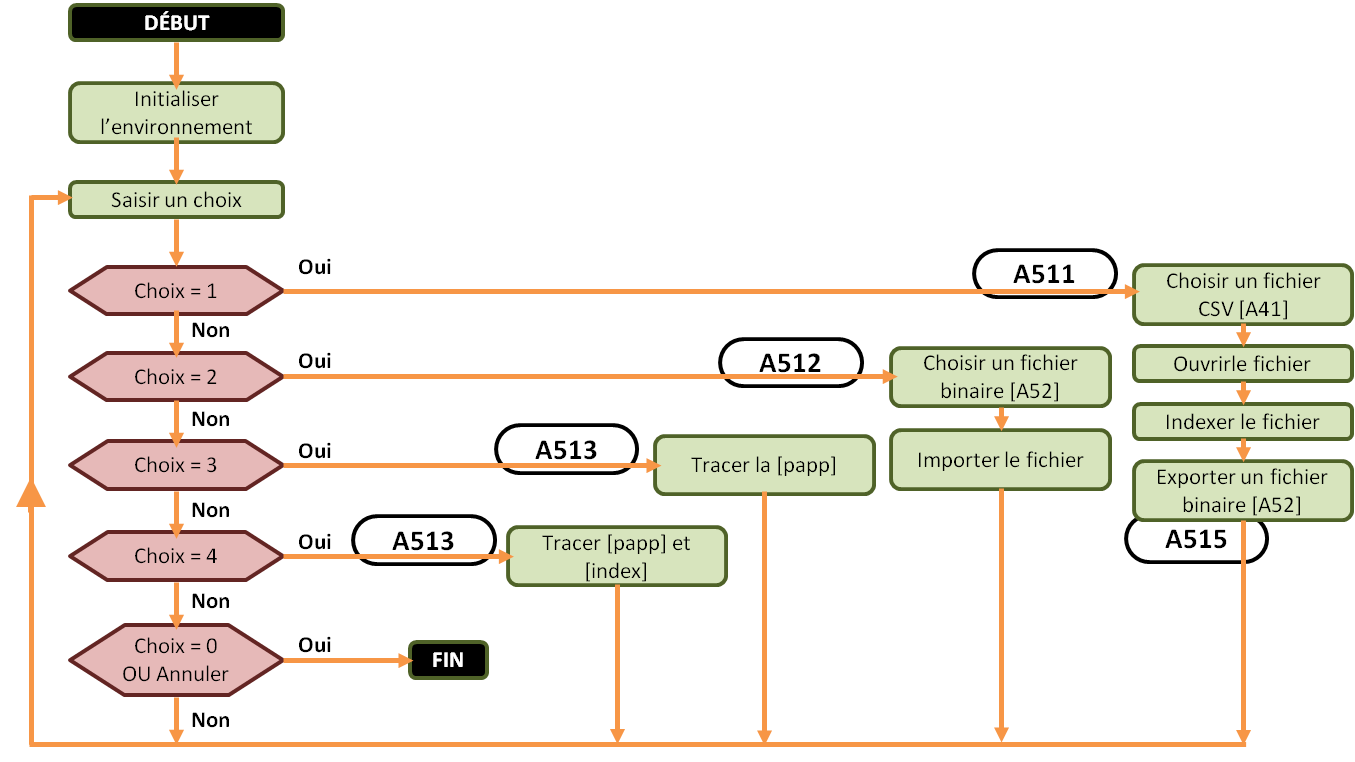
Le fichier [A52] contient les mêmes informations que [A41] mais dans un format binaire (extension .sod) qui est plus rapidement importé dans l’environnement Scilab.

### Programme sous Scilab

### Schéma de principe



#### Algorithme



#### Liste des fonctions disponibles

Le liste les fonctions présentes, une liste exhaustive et à jour est fournie dans [D3].

| **NOM** | **DESCRIPTION** | **PROTOTYPE** |
| --- | --- | --- |
| **[A512] Importer le fichier binaire** | | |
| charger\_variables | Importe les variables depuis le fichier binaire [A52] | charger\_variables(dataPath2Save) |
| **[A513] Tracer les courbes** | | |
| couleur\_plot | Retourne un tableau comprenant toutes les couleurs disponibles pour tracer | couleur = couleur\_plot() |
| heures\_Abscisses | Afficher les abscisses en heure | heures\_Abscisses(nbrLignes, fenetre, graphique, temps) |
| mise\_en\_forme | Met en forme le graphique (titre, légende, échelles, couleur de fond) | mise\_en\_forme(graphique, fenetre, opt\_BackgndCouleur) |
| tracer\_2\_Graph | Tracer 2 graphs ayant des unités différentes (puissance et énergie) | tracer\_2\_Graph(stcReleve) |
| tracer\_D\_Graph | Tracer plusieurs courbes de différentes couleurs, dans 1 graph | tracer\_D\_Graph(data2plot, jour, heure) |
| tracer\_Graph | Tracer 1 graph | tracer\_Graph(data2plot, NumCompteur) |
| **[A514] Analyses** | | |
| configuration | Affiche en console et retourne des booléens avec la configuration du compteur | configuration(donnee) |
| conversion\_temps | Converti un temps en secondes en heures, minutes, secondes et affichage en console (optionnel) | duree = conversion\_temps(tempsSecondes, opt\_affichage) |
| difTemps | Calcule la différence de temps entre 2 instants | Dtemps = difTemps(heure1, heure2) |
| dimensions | Retourne la longueur ou la largeur du tableau | nombre = dimensions(data,choix) |
| energie | Retourne une chaine de caractères les énergies de début et de fin du relevé et leur unité | energieStr = energie(obs\_nbrLignes, obs\_config) |
| GlrBrandtMoy | Retourne le signal filtré par une détection de rupture, basé sur un saut de moyenne | [g,mc,kd,krmv]=GlrBrandtMoy(x,h,Nest,Ndmax) |
| HeuresFonctionnement | Comptabilise un temps (de fonctionnement) où Papp(i) >= moyenne(Papp) | [duree, moyenne] = HeuresFonctionnement(opt\_moyInact) |
| info\_compteur | Affiche en console les informations du compteur | info\_compteur(stcReleve) |
| matrice | Retourne une matrice de valeur constante | tab = matrice(nbrLignes, nombre) |
| Modifier\_Horodatage | Modifier l’horodatage | Modifier\_Horodatage(Heure, offset) |
| moyenneGlissante | Retourne le signal filtré avec un filtre à moyenne glissante | signal\_f = moyenneGlissante(signal, fenetre) |
| nom\_compteur | Retourne le nom du compteur à partir de son numéro | nom = nom\_compteur(numCompteur) |
| nom\_jour | Retourne le nom du jour | nom = nom\_jour(dateReleve) |
| nombre\_2\_Chiffres | Retourne un nombre sur 2 chiffres | strNombre = nombre\_2\_Chiffres (nombre) |
| puissMoyStr | Retourne une chaine de caractères de la puissance moyenne avec son unité | puissMoyStr = puiss\_Moyenne() |
| reglerFctDeci | Affichage du signal avant et après filtrage, pour régler les coefficients du filtre [GlrBrandtMoy] | reglerFctDeci(x, h, g, mc) |
| **[A515] Exporter le fichier binaire** | | |
| Sauve\_Variables | Sauvegarder les variables dans un fichier binaire [A52] | Sauve\_Variables (filePath, stcReleve, stcStatistiques) |

Tableau – Liste des fonctions disponibles

## A6 – Affichage

### Schéma de principe

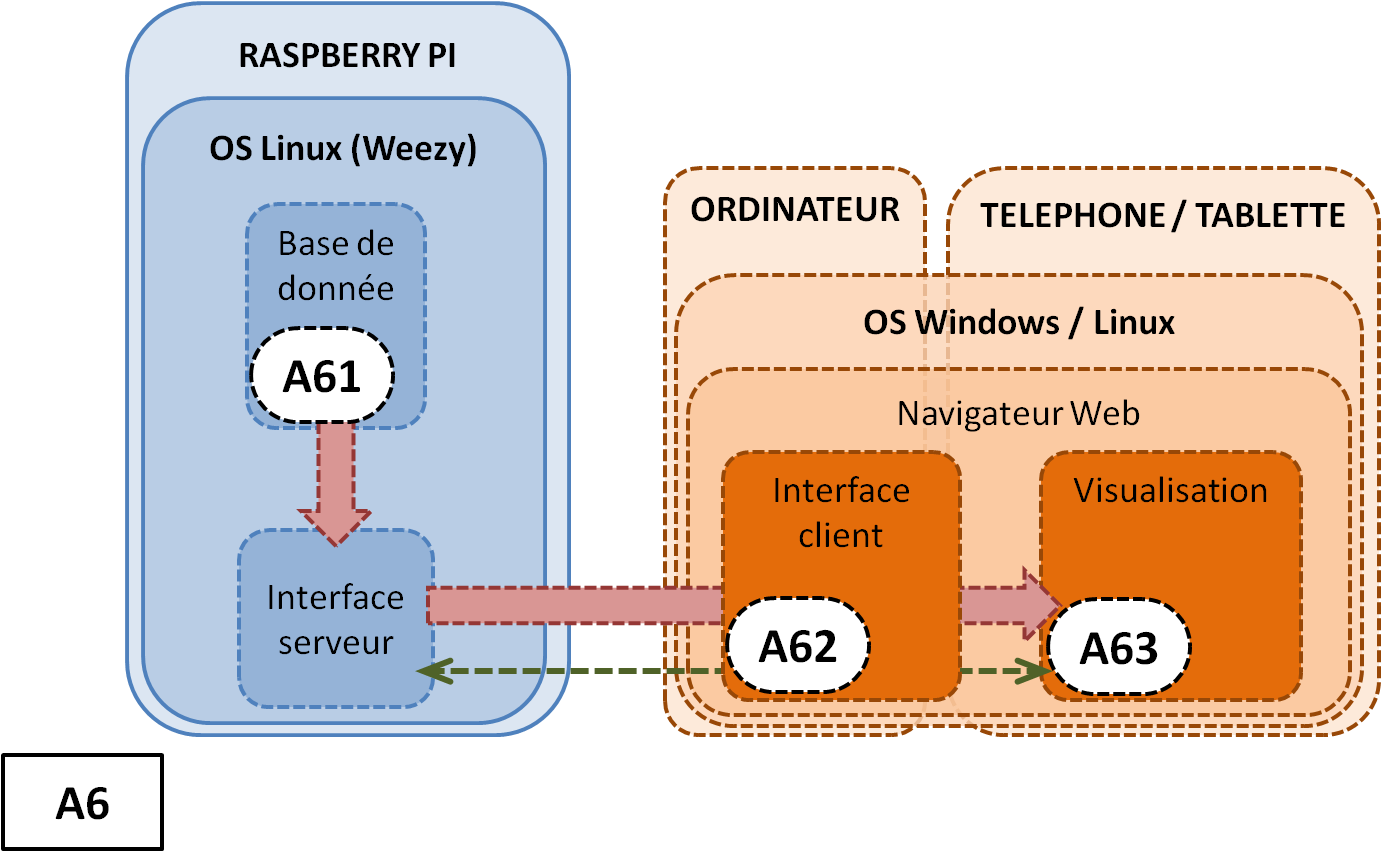


Figure - A6 - Schéma de principe

# ANNEXES

## Bibliographie

1. ERDF-NOI-CPT\_44E [www.erdf.fr/medias/DTR\_Generalites/ERDF-NOI-CPT\_44E.pdf](http://www.erdf.fr/medias/DTR_Generalites/ERDF-NOI-CPT_44E.pdf) Septembre 2013
2. Document Compteur\_energie\_zcd126\_02 <http://www.france-electric.com/compteur-electrique-monophase-edf-chauffage-90a-20kwh-c2x2206265> Septembre 2013
3. Liste exhaustive des fonctions disponibles sous Scilab

## Compteurs Testés

Les compteurs électroniques monophasés déjà testés sont les suivants :

|  |  |  |
| --- | --- | --- |
| **Référence compteur** | **Configuration** | **Remarque** |
| Landis & Gyr ZCD126.02 |  |  |
| Linky | Base |  |
| Mt Sax **(TBD)** | Heures Pleines - Heures Creuses | Pas de Papp dans les trames |
| Claix **(TBD)** | Base |  |

## BOM

## Fichiers générés

## Documentations des sources

### Code C

Installer Doxygen ([www.doxygen.org](http://www.doxygen.org)) et Graphviz (<http://www.graphviz.org>). Dans le fichier de configuration *.\Code\Compteur\_LinkyConfiguration.dxgn*, editer le chemin de GraphViz comme illustré par la .

Exécuter ce même fichier de configuration pour documenter le code présent dans les répertoires *.\Code\Compteur\_Linky\Sources\src* et *.\Code\Compteur\_Linky\Sources\inc.*

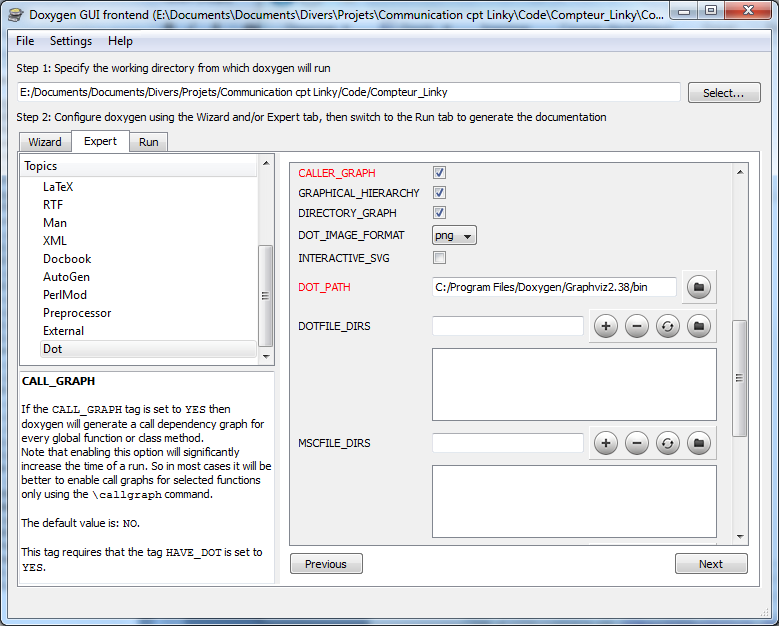
**

Figure – Doxygen, chemin de GraphViz

Le format de sortie est un ensemble de fichiers .html comprenant la liste des *bugs*, des *todos*, des *fichiers* et des *fonctions* comme il est possible de voir dans le fichier ci-dessous. Le point d’entrée étant le fichier *index.html*.



Pour d’informations sur <http://www.stack.nl/~dimitri/doxygen/manual/commands.html>

### Scripts Scilab

L’exécution du script *.\Scilab\DOCUMENTER.sce* permet de documenter les scripts présents dans le répertoire *.\Scilab\Sources* à la manière de Doxygen.

Le format de sortie est un fichier .txt comprenant la liste des *bugs*, des *todos*, des *structures*, des *fichiers* et des *fonctions* comme il est possible de voir dans le fichier ci-dessous. L’affichage en console permet de suivre l’exécution et de voir les éventuelles erreurs.



La configuration s’effectue dans le script *DOCUMENTER.sce.* Comme décrit par la ,il faut renseigner le *chemin du répertoire cible*, le *nom du fichier* *de sortie* et le *nom du projet*.

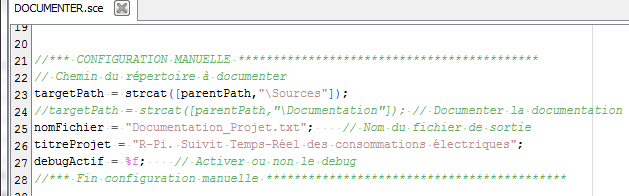


Figure – Documentation, configuration

Les balises prises en charge sont données dans le , quelle que soit leur casse. La illustre comment intégrer les balises dans un fichier ou une fonction.

| **Balise** | **Description** | **Fichier** | **Fonction** | **Restriction** |
| --- | --- | --- | --- | --- |
| todo | A faire | X | X |  |
| bug | Problèmes | X | X |  |
| version | Version | X |  |  |
| author | Auteur | X |  |  |
| date | Date du fichier | X |  |  |
| brief | Résumé | X | X |  |
| fn | Prototype |  | X | Cette balise doit être la première ligne d’une fonction |
| param | Paramètre |  | X |  |
| return | Retour |  | X |  |
| stc | Structure | X |  |  |

Tableau – Documentation, liste des balises

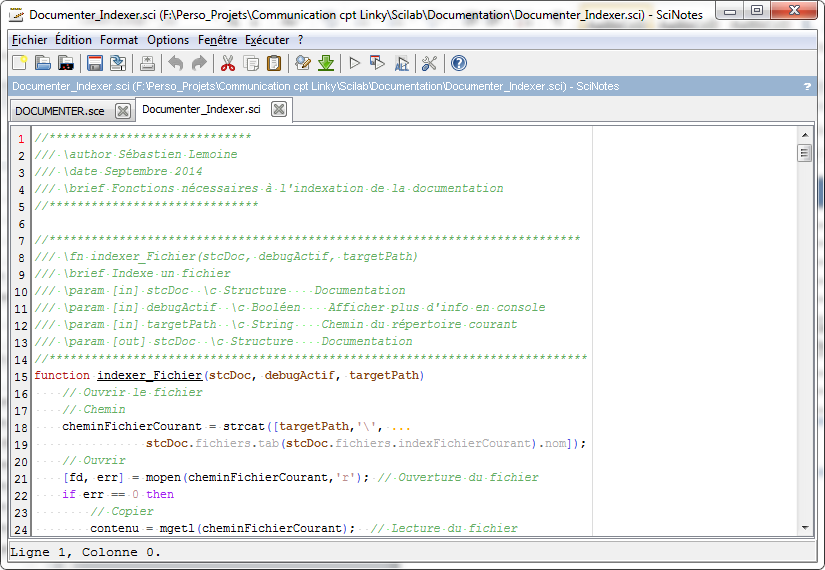


Figure – Configuration, exemple de balises