

A dark blue vertical bar runs down the left side of the page. A blue arrow points to the right from this bar, containing the date.

31/03/2017

AppliConso : Manuel Utilisateur

Several thin, curved lines in dark blue and light grey originate from the bottom left and sweep upwards and to the right.

LOPEZ Clément, THIEBAUD Charles, TOCABENS
Emmanuel
IUT BLAGNAC

Table des matières

I.	Introduction	2
II.	Guide Pratique	3
A.	Matériel nécessaire	3
B.	Fonctionnement de l'application	3
1.	Collecte et apprentissage des données	3
2.	Prédiction de la consommation	4
3.	Précision du modèle	4
4.	Gestion des modèles	4
C.	Lancement de l'application	5
1.	Lancement en surveillance générale	5
2.	Lancement en surveillance d'une application ciblée	6
D.	Enregistrement et consultation des évènements : fichier de log	6
E.	Paramètres modifiables	7
1.	Chemin de la batterie	7
2.	Chemin de la luminosité	7
3.	Enregistrement ou non des données dans le fichier de log	7
4.	Nom du fichier de log	8
5.	Seuil minimal de prise en compte des applications	8
6.	Numéro de port de l'interface web	8
F.	Fermer « AppliConso »	8

I. Introduction

L'application « AppliConso » est un outil permettant à l'utilisateur de pouvoir consulter la consommation en temps réel de sa machine ainsi que la consommation d'une application spécifique ou de toutes les applications ouvertes au lancement de « AppliConso ».

Elle est le fruit d'une demande de l'Institut de Recherche en Informatique de Toulouse (IRIT) et s'inscrit notamment dans un objectif pédagogique (permettre à de jeunes programmeurs d'avoir un aperçu direct de la consommation des applications qu'ils ont créés) ainsi que dans un but écologique (permettre aux utilisateurs de se rendre compte de la consommation de leurs machines).

« AppliConso » se veut très simple d'utilisation, aussi, le lancement de cette application et l'affichage de la consommation se font via un terminal de commande.

Ce Guide a par ailleurs été rédigé afin de faciliter la compréhension de l'utilisation et du fonctionnement de « AppliConso » ainsi que de répondre aux éventuelles questions que vous pourriez vous poser au cours de son usage.

II. Guide Pratique

A. Matériel nécessaire

« AppliConso » est une application adaptée aux utilisateurs de PC portables utilisant LINUX comme système d'exploitation.

L'application n'est pas compatible avec un PC fixe puisqu'elle fonctionne suivant 2 phases (quand le PC portable est en charge et quand il ne l'est pas). Pour plus d'informations sur ce fonctionnement en 2 phases vous pouvez vous référer à la partie « Fonctionnement de l'application ».

L'application est également incompatible à tout autre système d'exploitation que LINUX.

Pour de plus amples informations, vous pouvez vous référer au manuel d'installation de l'application.

B. Fonctionnement de l'application

AppliConso fonctionne en plusieurs phases :

Dans un premier temps, AppliConso collecte des données lorsque l'ordinateur est débranché, qui serviront plus tard à TensorFlow pour l'apprentissage et la construction d'un modèle. La précision de ce modèle dépend du temps passé à collecter les données et de la plage de données collectées : il est donc recommandé de faire varier les différentes valeurs de chaque capteur lors de cette phase (Utilisation du CPU, Luminosité) pour obtenir un modèle fidèle à la réalité.

1. Collecte et apprentissage des données

« AppliConso » détecte si une source d'alimentation est branchée, et son comportement diffère :

a) Sur batterie

Par souci de performances, AppliConso ne procède qu'à la collecte de données lorsque l'ordinateur est débranché. Ces données seront ensuite traitées lorsque l'ordinateur sera branché sur secteur.

b) Sur secteur

AppliConso procède alors à l'apprentissage des données collectées pour construire le modèle TensorFlow qui permettra à l'application de prédire la consommation en temps réel. Cette opération est gourmande (énergie et puissance).

2. Prédiction de la consommation

a) Sur batterie

AppliConso affiche la consommation réelle de l'ordinateur, obtenue par les fichiers systèmes.

b) Sur secteur

AppliConso fait appel au modèle TensorFlow construit pour estimer la consommation en temps réel de l'ordinateur.

3. Précision du modèle

Le modèle construit a une précision variable, qui dépend de deux facteurs : la quantité et la diversité des données collectées.

Il est donc conseillé à l'utilisateur de faire varier les valeurs des différentes sondes lors de la collecte (charge CPU, luminosité, ect...).

Le temps nécessaire pour construire un modèle relativement fiable est de l'ordre de moins de 5 minutes, en faisant convenablement varier les différents capteurs.

4. Gestion des modèles

Un modèle correspond à une combinaison de sondes/capteurs. Le modèle est importé au lancement de l'application. Si aucun modèle n'existe ou si une sonde est retirée ou ajoutée, le modèle est recréé.

C. Lancement de l'application

Sous LINUX, ouvrez un terminal dans le dossier où se situe « AppliConso ».

Pour lancer l'application l'utilisateur a le choix entre plusieurs options possibles, chacune permettant l'affichage d'un certain type de données. En fonction de l'option de lancement, « AppliConso » affichera la consommation demandée (système, toutes les applications, une application spécifique), il est à noter que la consommation du système est affichée quel que soit le cas de figure.

1. Lancement en surveillance générale

a) *Interface console*

Pour afficher la consommation de toutes les applications, il suffit de lancer « AppliConso » avec la commande :

python AppliConso.py

« AppliConso » affichera alors en continu la consommation du système ainsi que la consommation de chaque application ouverte suivant le modèle :

<PID de l'application> : <consommation de l'application>.

« AppliConso » n'affichant pas directement le nom de l'application ouverte mais son PID, il est recommandé d'utiliser « AppliConso » avec un gestionnaire de tâches d'ouvert permettant de faire la correspondance entre le PID et le nom de chaque application.

Il est également à noter que la liste des applications ouvertes est initialisée au lancement de « AppliConso » et n'est pas réactualiser par la suite. La consommation d'un processus exécuté après le lancement de « AppliConso » ne sera donc pas visible dans la liste affichée.

b) *Interface Web*

L'interface web permet un affichage plus élégant des données reçues, sous forme de courbe. Il est accessible en lançant AppliConso en ligne de commande avec :

python AppliConso.py -w

Il faut ensuite lancer le fichier **graph.html** dans le répertoire d'AppliConso, ouvrant ainsi votre navigateur internet. La consommation du système devrait alors s'afficher sur la page ouverte de votre navigateur sous forme de courbe.

2. Lancement en surveillance d'une application ciblée

a) Interface console

Pour afficher la consommation d'une application spécifiquement, on va utiliser son PID pour la sélectionner.

On va donc lancer « AppliConso » avec la commande :

python AppliConso.py <PID appli à surveiller>

Comme pour l'affichage de toutes les applications, « AppliConso » ne va pas afficher directement le nom du processus exécuté mais son PID et sa consommation.

b) Interface Web

Pour afficher ces données en interface web, il faut lancer « AppliConso » avec la commande console :

python AppliConso.py -w <PID appli à surveiller>

Il faut ensuite lancer le fichier **graph.html** dans le répertoire d'AppliConso, ouvrant ainsi votre navigateur internet. Deux courbes s'afficheront alors sur la page ouverte de votre navigateur : une pour la consommation du système et une pour la consommation de l'application surveillée.

D. Enregistrement et consultation des évènements : fichier de log

La trace des différents évènements advenus lors de l'exécution de « AppliConso » (données des capteurs, machine en charge ou sur batterie, etc...) est conservée dans un fichier de log accessible et consultable à tout moment dans le répertoire de l'application.

L'enregistrement ou non des données d'une exécution de « AppliConso » ainsi que le fichier dans lequel ces données sont enregistrées sont cependant modifiables.

E. Paramètres modifiables

Plusieurs paramètres sont modifiables dans « AppliConso ». Certains le sont par nécessité (sur certains appareils ayant des configurations spécifiques, les paramètres par défaut de « AppliConso » peuvent présenter des erreurs) d'autres permettent une amélioration de l'expérience utilisateur.

Tous sont modifiables via un seul fichier présent dans le répertoire de l'application : **Constantes.py**

1. Chemin de la batterie

En fonction du modèle de votre ordinateur, « AppliConso » est susceptible de rencontrer une erreur avec les configurations par défaut.

En effet, sur certaines machines les données de la batterie sont consultables par le chemin d'accès :

/sys/class/power_supply/BAT1

Tandis que sur d'autres modèles il s'agit du chemin d'accès :

/sys/class/power_supply/BAT0

Par défaut, « AppliConso » va rechercher à l'emplacement du premier des deux chemins. Si une erreur apparaît au lancement de l'application, allez dans le code du fichier **Constantes.py** et modifiez le paramètre **BAT_PATH** en lui donnant le chemin d'accès : **/sys/class/power_supply/BAT0**

2. Chemin de la luminosité

De la même manière que la batterie, le chemin d'accès aux données de la luminosité est modifiable dans le fichier **Constantes.py**, avec le paramètre **BACKLIGHT_PATH**.

Par défaut ce paramètre pointe vers le chemin **/sys/class/backlight/intel_backlight/**

3. Enregistrement ou non des données dans le fichier de log

Garder la trace de l'exécution de « AppliConso » est une option activée par défaut mais que vous pouvez désactiver à tout moment.

Dans le code du fichier **Constantes.py**, l'attribut **LOG** permet de déterminer si oui ou non le fichier de log sera créé/implémenté. Si l'attribut a une valeur à **True** (valeur par défaut), l'enregistrement de la trace se fera, si l'attribut a une valeur à **False** le fichier de log ne sera pas créé/implémenté.

Vous pouvez modifier ce paramètre dans le fichier **Constantes.py**.

4. Nom du fichier de log

Lorsque l'enregistrement de la trace d'exécution de « AppliConso » est activé, les données sont écrites dans un fichier texte qui est créé s'il n'existe pas encore ou implémenté s'il existe. L'utilisateur peut cependant modifier le nom du fichier dans lequel la trace d'exécution sera enregistrée. Ce qui permet de pouvoir, par exemple, créer un fichier de log pour chaque lancement de l'application.

Pour modifier le nom du fichier de log, allez dans le code du fichier **Constantes.py** et modifiez le paramètre **LOG_FILE_NAME**. Par défaut l'attribut a pour valeur **log.txt**, les données seront enregistrées dans un fichier de ce nom dans le répertoire de « AppliConso ».

En cas de changement de nom de fichier de log, il est recommandé de garder l'extension **.txt**

5. Seuil minimal de prise en compte des applications

Lorsque « AppliConso » est lancée en surveillance générale en interface console, l'affichage retourné est la consommation du système ainsi que la consommation de toutes les applications qui consomment au moins un seuil minimal.

Ce seuil peut être changé dans le fichier **Constantes.py**, il s'agit de l'attribut **MIN_CPU_USAGE**. Il représente la part (en %) de consommation du processeur à partir duquel une application est prise en compte par « AppliConso » pour être affichée dans la console.

Par défaut, la valeur de l'attribut **MIN_CPU_USAGE** est à 0, c'est-à-dire que toutes les applications consommant plus de 0% du CPU sont référencées.

6. Numéro de port de l'interface web

Lors du lancement de « AppliConso » en interface web, les données de consommation sont envoyées sur un serveur dont **graph.html** capte les informations pour les afficher.

Ce serveur est ouvert sur un numéro de port qui peut être modifié dans le fichier **Constantes.py** sous l'attribut **PORT**.

Par défaut, sa valeur est de 3035.

F. Fermer « AppliConso »

Appuyer deux fois consécutivement sur CTRL+C ou fermer le terminal fermera « AppliConso »