

PROGRAMMEZ!

le magazine du développeur

www.programmez.com



Cahier DRUPAL

Installation / Configuration / Mon 1er site / **Drupal 8**

100
PAGES

Coding à la plage

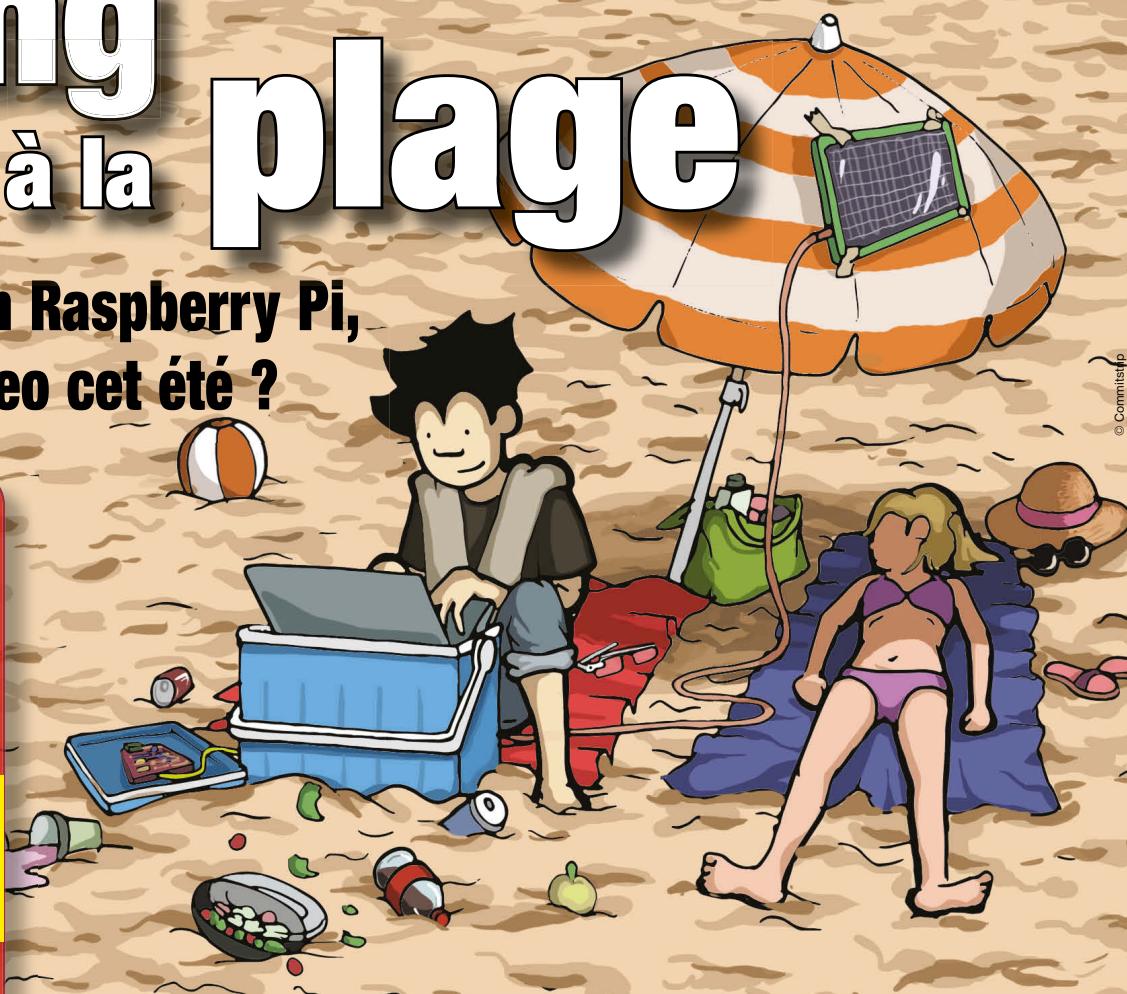
Que faire de son Raspberry Pi, Arduino et Galileo cet été ?

Écologie :
l'écoconception logicielle :
c'est maintenant !

Les nouveautés de MariaDB

Optimiser les performances de Google Maps

Et aussi :
Le langage SWIFT
Visual Studio 2014



La révolution du « maker faire »

Mensuel n°176 - Juillet-Août 2014



WINDEV 19

DÉVELOPPEZ 10 FOIS PLUS VITE

WINDEV : Logiciel professionnel de développement multi-plateformes

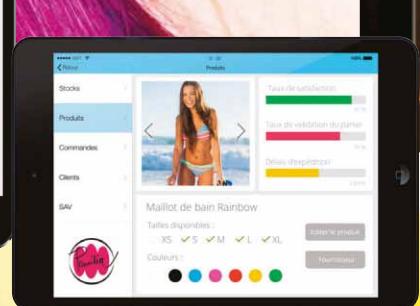
Vous méritez le meilleur, vos équipes méritent le meilleur pour être le plus efficace possible, quel que soit le matériel sur lequel vos applications vont fonctionner : **PC, tablette, smartphone, en application native ou en site Internet**.

Incroyable : grâce à **WINDEV 19, WEBDEV 19 et WINDEV Mobile 19, les applications que vous écrivez sont nativement portables**.

Vous réutilisez votre code, vos fenêtres, vos états... dans tous les environnements: **Windows, Mac, Linux, Android, iOS (iPhone, iPad), Windows Phone**, pour des applications natives et pour des sites, avec les données en local, sur serveur local ou distant, ou encore dans le **cloud**.

Dans l'intérêt de votre entreprise, dans l'intérêt de vos utilisateurs et de vos clients, commandez aujourd'hui votre **WINDEV 19** !

document non contractuel



Environnement de développement professionnel depuis 20 ans
Dossier + DVD + Témoignages sur simple demande (gratuit)

Fournisseur Officiel de la
Préparation Olympique

WINDEV AGL N°1 en FRANCE



www.pcsoft.fr

Des centaines de références sur le site

Devoirs de vacances



©German Federal Archive

“ Je crains le jour où la technologie dépassera l'Homme.
(Albert Einstein)



©Syfy

“ Les Cylons ont été créés par l'Homme... ils se sont rebellés... ils ont évolué... leur apparence, leurs sentiments sont humains... certains sont programmés pour croire qu'ils sont humains...

(Battlestar Galactica)

Deux réflexions, une réelle, une provenant d'une référence absolue de la Science Fiction. Nous vous proposons un été studieux : envoyez-nous, avant le 5 septembre, vos commentaires et analyses argumentés. Les meilleurs seront publiés dans le n°178 !

A envoyer à l'adresse :
redaction@programmez.com

Rendez-vous le 30 août pour Programmez ! n°177.

La rédaction.

sommatoire	4	Swift : le défi d'Apple		90
	37	Cahier de vacances : 100 % DRUPAL		F# 1 ^{re} partie
85	Nokia X			35
	95	Optimiser ses Maps		Les dévs du mois
16	Un peu de regex ?			
	18	Devoxx 2014 2e partie		65
28				Soleil, plage et coding
				Arduino, ampoules HUE, Raspberry Pi, Galileo
31	La révolution de l'écoconception logicielle			
	Wakanda : c'est quoi ?			91
6	22	Maria DB		Xamarin
	26	Visual Studio 2014		
n° 177 en kiosque le 30 août 2014	98	CommitStrip		

À LIRE DANS LE PROCHAIN NUMÉRO

n° 177 en kiosque le 30 août 2014

Dossier PHP

Quel avenir pour PHP ?
PHP en mode cloud
Un outillage de plus en plus complet

3D Babylon.js, la tuerie ! Le moteur 3D surdoué

Formation / carrière

Comment apprendre le code à l'école et aux enfants ?
Un point complet sur des initiatives concrètes.

Un aperçu du langage Swift d'Apple

Beaucoup d'annonces en ce 2 juin du côté d'Apple durant la conférence développeurs : OS X Yosemite et la nouvelle mouture de l'OS d'Apple pour les iPhones et iPad, iOS 8. Toutefois, l'une des plus grandes surprises de « l'Apple Worldwide Developers Conference » 2014 est la présentation d'un nouveau langage de programmation simple, plus pratique, plus rapide : Swift. Ce nouveau langage de programmation de la firme à la pomme introduit une nouvelle syntaxe, plus épurée et moins verbeuse que l'objective-c, qui, lui, reposait sur le modèle du C. On peut dire que ce langage a été bien accueilli lors de son annonce.

La société TIOBE s'attend à voir ce nouveau langage dans le top 20 vraiment rapidement, dans les prochains jours. Cette entreprise mesure la popularité des langages en se basant sur les requêtes dans les moteurs de recherche ou sur les sites les plus fréquentés de la toile.

Pourquoi un nouveau langage pour un système qui fonctionne ?

Tout d'abord, l'objective-C étant un langage datant de 1983, il devient vieillissant. De plus, c'est une extension du C ANSI et il garde une grosse partie des défauts de ce dernier, dont le majeur qui est la gestion des pointeurs. C'est un langage réputé, cependant difficile à prendre en main. Plus votre application sera volumineuse et utilisera de pointeurs, plus il sera difficile d'éviter les fuites mémoire, les failles de sécurité et les plantages dus à une mauvaise gestion de la mémoire.

Craig Federighi, vice-président et responsable de la partie logicielle d'Apple, a promis que « Swift » serait un langage qui faciliterait la vie de ses développeurs. (*Objective-C sans le langage de programmation C*)

Le langage Swift

Le langage Swift est multi-paradigmes : il puise son inspiration autant du langage Objective-C que des langages Ruby ou Python. Il constitue un langage de programmation pour Cocoa et Cocoa Touch, les API d'Apple pour le développement orienté objet de OS X et iOS.

Il sera compatible à partir de la mise à jour iOS 8, pour les appareils mobiles de la Pomme. Il utilise le compilateur LLVM (Low Level Virtual Machine) afin d'être transformé en code natif optimisé.

Selon Apple, le tri d'objets complexes est 3.9 fois plus rapide que Python (pour objective-c: 2.8 fois).

Mais cela sera à confirmer concrètement sur le terrain avec les nouvelles applications.

Un langage plus concis

Plus simple d'apprentissage et plus propre, grâce à son aspect épuré et inspiré de certains langages de script tels que python.

Quand dans Objective-C :

```
#import <stdio.h>
#import <Foundation/Foundation.h>
int main(void)
{
    NSLog(@"Bienvenue sur Programmez !");
    return 0;
}
```

Avec Swift, une ligne suffit :

Println(*«*Bienvenue sur Programmez !*»*)

C'est un langage sécuritaire.

Finis les pointeurs

En effet, il est plus sécuritaire grâce à la gestion dynamique de la mémoire : les variables sont toujours initialisées avant utilisation et les tableaux sont vérifiés pour éviter les débordements.

Il n'est plus nécessaire de penser à la libération de la mémoire grâce à ARC (Automatic Reference Counting de swift), qui permet de suivre et gérer la mémoire de votre application à la manière d'un garbage collector de Java. ARC libère automatiquement la mémoire utilisée par les instances de classe lorsqu'elles ne sont plus nécessaires.

Ce code générera une erreur à la compilation car « maVariable » n'a pas été initialisée :

```
var maVariable :Integer
println(maVariable)
```

Types, contrôle de flux et opérateur

Swift utilise des éléments en standard tel que l'objective-c : primitives de bas niveau comme les types, le contrôle de flux, les opérateurs avec des petits plus (overflow operators) opérateur évitant les dépassements (Overflow Operators)

```
var maxDeInt= Int16.max
maxDeInt += 1
//Erreur dans ce cas là.
maxDeInt = maxDeInt &+ 1
//maxDeInt égale à zero grâce à l'opérateur « &+ » dans ce cas-là plus d'erreur.
```

Class / Structure et héritage

Il reste un langage s'inspirant de l'orientation objet d'Objective-c. Celui-ci comprend la notion class et introduit les génériques. Il intègre également la notion de structure.

Exemple code structure en swift :

```
struct maStructure{
    func machin(){
        println("Hello Structure")
    }
}
var ob = maStructure()
ob.machin()
```

Ce qu'il est possible de faire avec les class et struct

Définir des propriétés pour stocker des valeurs, ajouter des méthodes. Elles peuvent être étendues afin d'obtenir plus de fonctionnalités et permettent

l'implémentation de « protocols ».

Ce qu'il est possible de faire uniquement avec une class: Héritage, type casting, qui permet de vérifier le type d'une instance et de traiter cette dernière comme si c'était une superclass différente ou une sous-classe de sa propre hiérarchie, dé-initialiser afin de libérer de la ressource (ARC pour la libération de la mémoire automatique).

ARC, Class et déinitialisation

```
class LecteurProgrammez{
    let nom:String
    init(nom: String)
    {
        self.nom = nom
        println("Bonne lecture \u202a(nom), initialisation")
    }
    deinit{
        println("Fin de lecture \u202a(nom), en cours de d\u00e9-initialisation ")
    }
    var unLecteur :LecteurProgrammez?
    // notez l'utilisation de ? permet d'initialiser avec nil
    unLecteur = LecteurProgrammez(nom:"roger")
    // Affichage de Bonne lecture roger, initialisation
    unLecteur=nil
    // Affichage de Fin de lecture \u202a(nom), en cours de
    d\u00e9-initialisation, ARC prend le relais ;)
```

Les « protocols », l'interface made in Swift

Ils reprennent la notion de protocols d'objective-c, très similaires aux interfaces en java, qui permettent de définir le modèle de méthodes, propriétés ou autres requis à une tâche particulière d'une fonctionnalité. Un « protocol » est ensuite adopté par une class, structure ou énumération.

Exemple :

```
protocol NomComplet {
    var nomComplet: String { get }
}
struct Person: NomComplet {
    var nomComplet: String
}
let lecteur = Person(nomComplet: "LecteurProgrammez")
```

Closure, fermetures unifiées

Il introduit aussi la notion de « Closure » qui sont des blocs autonomes de fonctionnalités semblables aux « blocks » introduits depuis 2010 dans objective-c ou aux lambdas dans d'autres langages tels que le C#.

Les « closures » permettent d'écrire des fonctions locales, qui peuvent être passées comme arguments ou retournées comme valeurs par des fonctions appelées.

```
let noms = ["Amandine", "Stephanie", "Richard", "Daniel"]
//Exemple de closure————
func inverse(s1: String, s2: String) -> Bool {
    return s1 > s2
}
//Fin de l'exemple————
var triInverse = sort(noms, inverse)
Resultat pour triInverse [Stephanie, Richard, Daniel, Amandine]
```

Structure de données

Swift propose deux voies: les tableaux et les dictionnaires. Les tableaux permettent de stocker de manière ordonnée une liste de valeurs de même type, tandis que les dictionnaires stockent de manière non ordonnée des collections de valeurs de même type.

Dans le second cas, les valeurs peuvent être référencées grâce à une clé ou identifiant.

Il est impossible d'insérer des valeurs de type incorrect sans la génération d'une erreur.

Déclaration d'un Array et comment y accéder :
var monPremierArray : String[] = ["Stones", "Beatles"]
println("nombre de groupes \u202a(monPremierArray.count)")
Itération sur un tableau :
For item in monPremierArray
{
 println(item)
}

Mais ce n'est pas tout...

Swift apporte son lot de nouveautés. Par exemple : les Tuples et valeurs de retour multiple, les génériques, Itération rapide et concise sur une plage ou une collection, les types facultatifs (var pour les variables et let pour les constantes). Il introduit aussi la notion de map et de filter.

Pour développer en Swift

Il faudra utiliser l'IDE XCode 6 (actuellement en version bêta). A ce jour, pour télécharger officiellement cette version, il faut faire prendre un abonnement annuel développeur Apple (OS X ou iOS).

Pour ceux qui souhaitent s'initier rapidement au langage, Apple a mis gratuitement à disposition un manuel permettant d'appréhender rapidement ce nouveau langage (« the swift programming language »).

Swift tête baissée ?

Swift est un grand pas en avant dans l'univers de programmation d'Apple. Il est plus accessible et permettra de séduire les nouveaux développeurs réticents à l'objective-c.

Les développeurs débutants voulant s'initier rapidement à un nouveau langage, uniquement sur les plateformes Apple, pourront sans conteste utiliser Swift.

Pour ceux qui souhaitent aller plus loin, cela ne suffira peut-être pas à l'heure actuelle. Beaucoup de fonctionnalités sont en objective-c donc, lors d'un débogage, la connaissance de ce langage va être un plus indéniable. Le fait qu'il soit moins verbeux peut également être un frein à la compréhension des mécanismes sous-jacents.

Swift est une réponse, attendue par de nombreux développeurs.

Cependant il reste un langage Apple pour l'univers Apple, cela oblige donc à l'apprentissage d'un nouveau langage pour un type de plateforme, alors que nous avons d'un autre côté Xamarin qui nous permet d'être multiplateforme.

A l'époque de l'ouverture, était-il judicieux de la part de la firme à la pomme de sortir un nouveau langage uniquement pour ses produits ?

L'avenir nous le dira.

 Freddy Hebrard

Formateur / Ingénieur d'étude

MCPD Microsoft

freddyhebrard@hotmail.com

Devoxx4Kids : la technologie ce n'est pas que pour les adultes !

Le 22 mars dernier, la première édition de Devoxx4Kids Lille a eu lieu. L'idée est simple : sur une journée, les enfants et les parents peuvent venir toucher, utiliser, coder les dernières technologiques. Ch'ti JUG a fait les choses en grand, la preuve.

L'événement était monté avec Devoxx4Kids France et Devoxx4Kids Belgique qui ont pu apporter des goodies, leur matériel et leur expertise. On n'improvise pas des sessions découvertes de Scratch, de NAO et de Mindstorm comme cela. Il a fallu plusieurs semaines de travail. Une cinquantaine de personnes étaient présentes.

Place aux enfants !

Les enfants ont commencé à arriver à 13h30, à l'heure, et très rapidement, tout le monde était là dans le hall, prêt à s'amuser et à découvrir un nouveau monde. Après une petite introduction pour donner les grandes lignes de l'organisation et quelques règles de sécurité, les enfants ont été répartis en 3 groupes et ont été accompagnés dans les 3 salles des ateliers. Nous n'avions pas assez préparé la répartition des enfants par groupe, on a donc du un peu improvisé, mais au bout du compte, les groupes ont bien fonctionné ! C'est parti pour une heure d'atelier, les photographes passent dans les ateliers pour immortaliser cette journée, les parents observent ce qu'il se passe, eux aussi découvrent tout ça et surtout les enfants sont attentifs et s'amusent de pouvoir créer des jeux en scratch ou à contrôler les robots NAOs et Mindstorm.

On enchaîne

À la fin de la première session, on fait tourner les groupes d'enfants à l'atelier suivant. Pas de pause, seulement quelques minutes pour changer de salle. Les enfants se racontent ce qu'ils ont fait, ils rient et disent quelques bêtises. Le second atelier démarre, on est un petit peu en retard (5 min), mais tout se déroule bien. On est en surnombre dans l'atelier Mindstorm avec 3-4 voir parfois 5 enfants par robot, on voit bien que dans l'atelier Scratch il y a pas mal d'effervescence, mais tout le monde est ravi, les animateurs s'en sortent très bien ! Puis c'est déjà l'heure de la pause goûter !



“On croirait comme si qu'il y a un anniversaire !”

Les enfants arrivent tous dans le hall pour prendre le goûter, une grande table est au centre avec gâteaux, bonbons et boissons. Un enfant dit « On croirait comme si qu'il y a un anniversaire ! ». Tout le monde profite du buffet, le robot NAO fait son show en dansant et tout le monde se met en rond autour du robot Astro, on prend des photos des enfants, des organisateurs et des photos personnalisées des enfants avec le NAO pour le souvenir. Ça passe vite,

très vite, c'est la fin de la pause, tout le monde retourne vers les ateliers pour la dernière session. Le dernier atelier permet à chaque groupe de boucler son programme et les enfants sont une nouvelle fois ravis de découvrir une nouvelle manière d'apprendre. Chaque atelier est très différent des autres. Avec scratch les enfants sont très libres, ils décident de créer un jeu, de raconter une histoire, de faire des bêtises, d'enregistrer des sons... Avec Mindstorm, ils découvrent le montage du Lego puis apprennent à interagir avec le robot pour envoyer des missiles sur des petites cibles ou à se déplacer sur une carte. Avec NAO, on commence par écouter le robot se présenter lui-même puis on écoute Daniel de Luca qui explique avec beaucoup de pédagogie comment utiliser le logiciel pour prendre en main le robot qu'on peut faire s'asseoir, se lever, marcher, parler, détecter la présence d'un obstacle sur son chemin... Et c'est déjà la fin.

Vivement le prochain !

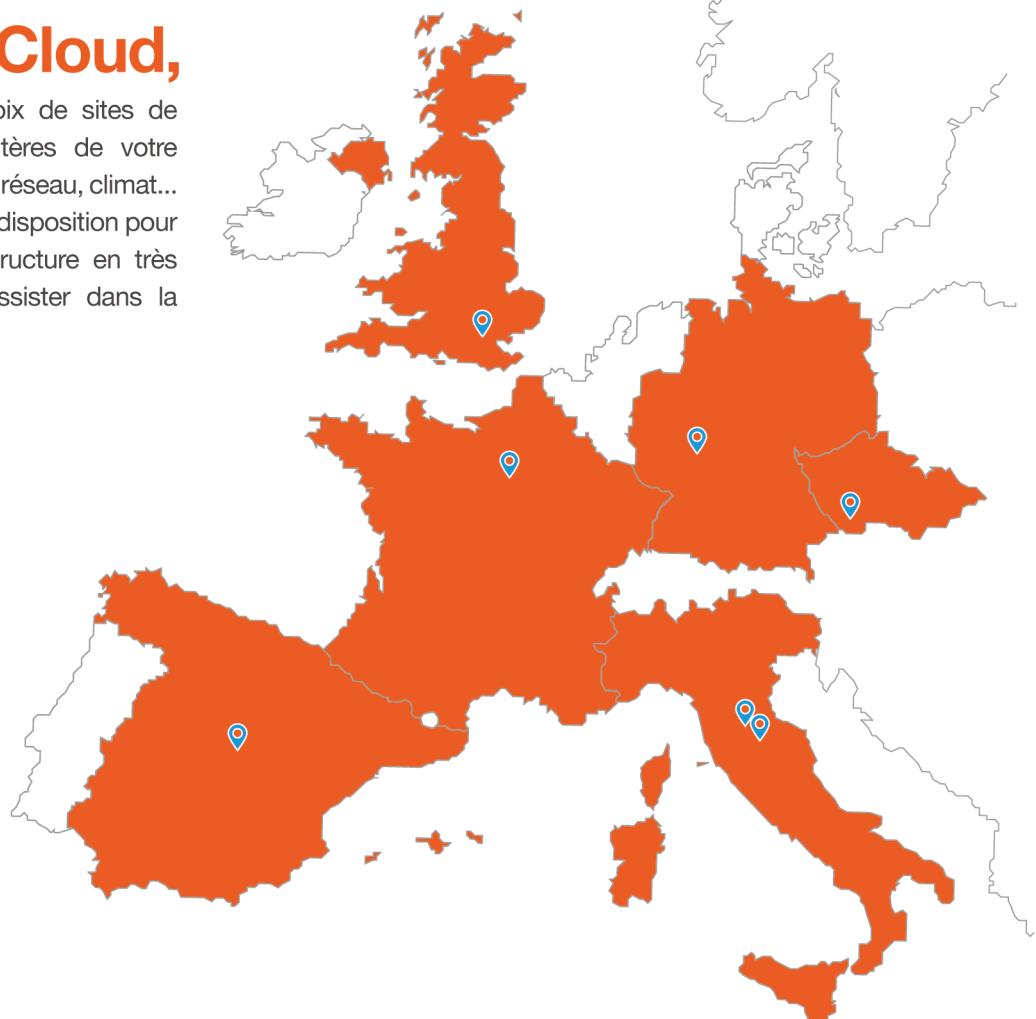
En sortant, on demande aux enfants s'ils ont aimé et quel atelier ils ont préféré. « Légo », « Scratch », « NAO ». Ils ne sont pas d'accord pour choisir leur préféré, mais ils se sont tous amusés. On peut décompresser, tout s'est bien passé. Un enfant vient me demander si le samedi prochain c'est à la même heure... je ris, pas lui, il est très sérieux ! Des enfants, arrivent avec leurs manteaux sur le dos, ils viennent vers moi, j'avoue ne plus savoir comment ils s'appellent, mais ils me font chacun un bisou et me disent à bientôt, ça fait plaisir. Les parents aussi nous remercient à leur façon (sans les bisous) et demandent des infos sur la prochaine édition. On va y réfléchir, car c'était vraiment une journée spéciale !

● Cyril Lakech (@cyril_lakech)
 ● Ch'ti JUG Leader (@chtijug)

Je voudrais bâtir un PRA multi-site ? Je fais comment

Avec Aruba Cloud,

vous disposez d'un large choix de sites de secours, en fonction des critères de votre stratégie de sécurité: proximité, réseau, climat... Nos équipes sont aussi à votre disposition pour vous aider à bâtir une infrastructure en très haute disponibilité et vous assister dans la définition de votre stratégie.



3
hyperviseurs



6 datacenters
en Europe



APIs et
connecteurs



70+
templates



Contrôle
des coûts

“ Nous avons choisi Aruba Cloud car nous bénéficions d'un haut niveau de performance, à des coûts contrôlés et surtout car ils sont à dimension humaine, comme nous. Xavier Dufour - Directeur R&D - ITMP

Contactez-nous!

0810 710 300

www.arubacloud.fr

Cloud Public

Cloud Privé

Cloud Hybride

Cloud Storage

Infogérance

MY COUNTRY. MY CLOUD.*

aruba
CLOUD

Chapitre 8 : des événements hors du commun !

Comme vous l'aurez compris au travers des chapitres précédents, la vie à la Silicon Valley a un goût merveilleux et sucré de Paradis du Geek où il est bon d'être « Hacker » ou « Maker »; une vie dans laquelle vous pouvez à juste titre être fier d'être Hacker et non seulement au niveau software mais hacker de tous les processus existants, tous domaines confondus... Regardez cette merveilleuse vidéo de Logan LaPlante, jeune adolescent de 13 ans revendiquant avoir hacké le système éducatif et pour cause ... voici sa vidéo au TED : <http://youtu.be/h11u3vtcpaY> ou encore son About : <http://about.me/loganlaplante>

Bien que les technologies d'aujourd'hui permettent de suivre les tendances et grandes messes des acteurs majeurs à tout endroit et tout moment sur la planète, il n'en demeure pas moins que le même fuseau horaire et la proximité sont encore et toujours un point clé permettant d'anticiper bien des évolutions du marché et progressions technologiques.

Et je ne vous parle pas de la nécessité de « le Vivre », oui, vivre au cœur de ce tumultueux et formidable tourbillon d'innovations, d'événements, de rencontres, de discussions, d'échanges, de challenges des technologies, d'opinions, de visions et surtout de Missions. La Valley regorge ainsi d'une énergie qu'il est difficile d'expliquer sans y avoir goûté ! Qu'attendez-vous pour réserver un billet d'avion pour vos prochaines vacances ?

La Valley, quoi qu'en dise, est tout sauf les Etats-Unis ! Un peu comme ces zones hors de tout état, pays, frontière... cette partie de la Cali-

fornie dégage une énergie et un esprit de non-conformisme positif que nous ne retrouverons nulle part ailleurs, oui, ici vous avez le droit d'être de ce que vous êtes.... Pas de jugement, le seul jugement viendra de vous, de ce que vous avez décidé d'être ou de ne pas être, de vos conquêtes, de vos envies, de vos passions et de la rage que vous avez chaque matin à vous lever pour atteindre un objectif plus fort que tout le reste : « Changer le Monde » !

Cet « Esprit de la Valley » se rencontre dans les événements de Geeks, les Meetups, ou encore dans les événements de Hackers, de Makers... les Maker Faire ! Et c'est tout le sujet de cet article !

Découvrons ces nouveaux mouvements qui prennent depuis quelques années une ampleur de plus en plus conséquente et mettons le focus sur la « Maker Faire » organisée le mois dernier au sein de la Silicon Valley.

2014 est considérée comme l'année de la Maker Faire ! Imaginez que même la Maison Blanche et le président Barack Obama puisse héberger une Maker Faire afin de célébrer une nation de Makers !



18 Mai). C'étaient plus de 1000 makers exposants et plus de 120.000 personnes sur deux jours dans un parc aussi grand qu'un parc d'attraction.

Et la maker faire de la Valley a pour originalité que si vous êtes exposant, vous pourriez être visité par des géants comme les fondateurs ou dirigeants de Google, Facebook, Yahoo en mal constant d'innovation pour leur stratégie respective. L'un des fleurons français, originaire d'Euratechnologies dans le Nord, qu'est la première caméra 360 degrés conçue par l'équipe de Richard Ollier de la société Gyroptic a clairement vécu ce type de rencontre.

Il vous suffit de regarder le niveau de réussite de leur Kickstarter pour comprendre :

<https://www.kickstarter.com/projects/gyroptic/the-worlds-first-full-hd-360-camera> Une équipe de

Hackers de haut vol qui a trouvé son positionnement au sein de la Valley, et pour être proche d'eux, je peux vous garantir qu'ils méritent leur place au sein de la Maker Faire... Et, pourquoi pas VOUS et votre projet auquel vous rêvez depuis si longtemps ?

Maker Faire, quels Sujets, quels Thèmes ?

Et c'est là où tout devient magique, la Maker Faire de la Valley est tellement annonciatrice de futures vagues technologiques et donc de mouvements de marchés.



Maker Faire vous avez dit ?

Il vous suffira de vous diriger vers le site officiel de la maker faire à Paris pour vous aider à comprendre, en voici un petit extrait : « *La Maker Faire est le plus grand événement au monde qui regroupe ateliers, présentations et conférences autour des thèmes de la créativité, la fabrication et le mouvement Do it Yourself. La Maker Faire est un événement pour tous les adeptes des technologies, artisans, éducateurs, inventeurs, hobbyistes, ingénieurs, clubs de science, auteurs, artistes étudiants et exposants commerciaux. Tous ces "Makers" viennent à la Maker Faire pour présenter ce qu'ils font et partager ce qu'ils ont appris.* » -

<http://www.makerfaireparis.com/presentation/que-cest-ce-quune-maker-faire/>

Ces événements sont organisés maintenant partout à l'échelle mondiale, la plus prestigieuse se trouvant à San Mateo, en pleine Valley (17 et

L'été de tous les codes !

Abonnez-vous !

réserve à la France
Métropolitaine

Spécial étudiant
39€
1 an 11 numéros

1 an 11 numéros
49€
seulement (*)

2 ans 22 numéros
79€
seulement (*)



© Commissip

(*) Tarifs France métropolitaine

Toutes nos offres sur www.programmez.com

Oui, je m'abonne

- Abonnement 1 an au magazine :** 49 € (au lieu de 65,45 €, prix au numéro)
- Abonnement 2 ans au magazine :** 79 € (au lieu de 130,9 €, prix au numéro)
- Abonnement spécial étudiant 1 an au magazine :** 39 €

Photocopie de la carte d'étudiant à joindre

Tarifs France métropolitaine

ABONNEMENT retourner avec votre règlement à
Programmez, 17, route des Boulanger 78926 Yvelines cedex 9

Offre spéciale : abonnement + clé USB Programmez!

- 1 an (11 numéros) + clé USB : 60 €
- 2 ans (22 numéros) + clé USB : 90 €

Clé USB contenant tous les numéros de Programmez! depuis le n°100, valeur : 29,90 €

Tarifs France métropolitaine

M. Mme Mlle Entreprise : _____ Fonction : _____

Prénom : _____ Nom : _____

Adresse : _____

Code postal : _____ Ville : _____

Tél : _____ (Attention, e-mail indispensable pour les archives sur internet)

E-mail : _____ @ _____

Je joins mon règlement par chèque à l'ordre de Programmez !

Je souhaite régler à réception de facture

L'impression 3D

Les grands thèmes cette année étaient l'impression 3D tous types, du plastique au papier en passant par le bois ou encore le métal avec des objectifs tout aussi nobles qu'éducatifs avec l'impression de prothèses, de déguisements, de figurines ou encore de vaisseaux de Star Wars (là, j'adore) pour enfin proposer une application de création de blocs type Lego à envoyer ensuite directement à votre imprimante, des bijoux étaient également au menu et malheureusement des armes, nous n'y couperons pas, nous sommes à la fin de la journée dans les terres du Far West. Et le plus surprenant comme à chaque fois, est de voir l'origine de chacun de ces projets. Prenons la prothèse 3D sous différentes formes. Une jeune équipe d'étudiants venant tout droit d'Amérique Latine s'est attaquée à la mission d'intégrer des moteurs dans l'avant-bras afin de connecter des capteurs sensoriels sur la peau de l'utilisateur ou par casque d'analyse cérébrale et permettre ainsi une meilleure mobilité de la prothèse.

Les objets connectés

Les objets connectés n'ont pas été laissés de côté, bien au contraire, du dongle OBD2 connecté pour votre voiture, aux connecteurs tous types confondus pour votre maison, la gestion de vos plantes en passant par des objets artistiques, rien ni aucun domaine n'est laissé au hasard. L'un des plus surprenants étant Sony qui y exposait un projet appelé Mesh proposant une série de connecteurs de différents types comme des boutons, microphone, speaker, LED ou capteurs de luminosité vous permettant de créer au travers d'une application basée sur votre iPad n'importe quel scénario de surveillance ou de gestion de votre maison.

Très ludique et pédagogique au premier abord ! En quelques minutes, le démonstrateur nous monte un scénario de surveillance d'ouverture de porte avec photo à la clé et envoi par mail sur le portable de votre choix. Assez concluant, le seul regret étant que ce projet Mesh est encore en mode laboratoire et n'a pas de date de commercialisation annoncée. A mon avis ils étaient surtout là pour faire leur AB Testing et valider l'intérêt du grand public pour ce type de produit. Vu le monde qui s'arrêtait sur leur stand, j'ose espérer que nous devrions rapidement voir le produit sortir ou tout du moins un concurrent qui aura été inspiré par les démos. On ne comptait plus le nombre de Raspberry Pi et Arduino tant il y en avait !

La Robotique et les drones

Et justement ceci nous permet la transition vers le second sujet, la robotique et les drones ! Que dire de plus que ce qui n'a déjà été écrit dans les



articles précédents. Le mouvement est lancé, les robots toutes catégories confondues arrivent massivement, les drones ont été domestiqués pour en faire des objets de combats en arènes fermées par des filets... Etrange d'assister à ce type de spectacle où vous avez dans une zone grande comme un terrain de basket entouré par des filets, pas moins de 15 drones qui se battent en duels ponctuels avec un objectif digne d'Highlander.

Tout amateur s'y lance et créé son propre robot utile ou inutile, d'amusement ou de décoration, tout y passe... Il faut dire que les dernières annonces de ce mois de Juin par Aldebaran Robotics au Japon avec le robot Pepper ne font que confirmer la tendance mondiale et l'appétit de plus en plus féroce des consommateurs... Avec un robot annoncé à 2000 dollars, une grande partie de la population mondiale pourra rapidement se l'offrir.

Et encore ...

Je conclurai avec quelques coups de cœur. Tout d'abord, les associations qui y affichaient clairement leurs couleurs comme

<http://www.notimpossiblenow.com>, l'electronic frontier foundation ayant pour slogan de défendre notre liberté digitale <https://www.eff.org>.

Ensuite, Les voitures et autres véhicules électriques qui y étaient bien représentés... Il m'en a fallu de peu pour succomber à la tentation d'achat d'une trottinette électrique vous amenant à pas moins de 20 km/h sur 30 km.

Et enfin, le non moins spectaculaire ArcAttack! avec la fabuleuse interprétation de la marche impériale que je vous invite à regarder sur YouTube : <http://youtu.be/htQXkXGQgFM>

Liens :

Maker Faire : <http://makerfaire.com>

White House Maker Faire :

<http://www.whitehouse.gov/blog/2014/06/04/first-ever-white-house-maker-faire-celebrating-nation-makers>

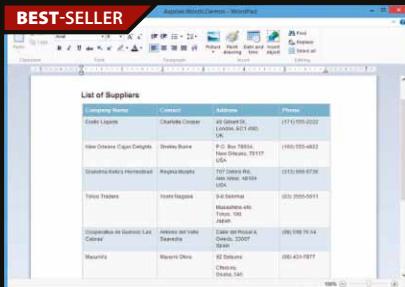
Aldebaran Pepper :

<http://www.aldebaran.com/en/medias/pepper-video>

Gregory Renard (Redo)

Geek in Silicon Valley

<http://gregoryrenard.wordpress.com>

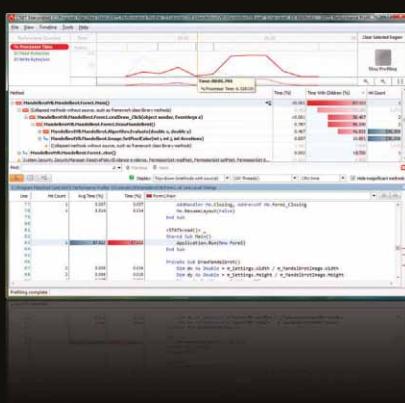


Aspose.Words for .NET à partir de € 727



Lisez, modifiez et écrivez des documents Word sans Microsoft Word.

- Création de documents, manipulation du contenu/formatage, puissante capacité de fusion de courrier et exportation en DOC/HTML
- Accès détaillé à tous les éléments d'un document par programmation
- Support les formats de fichiers: DOC, DOCX, WordprocessingML, RTF, HTML, OOXML, OpenDocument, PDF, XPS, EMF et EPUB

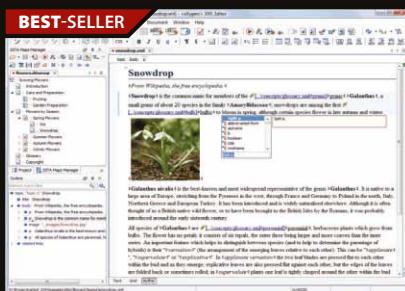


Red Gate .NET Developer Bundle à partir de € 760



Éradiquez et corrigez le code lent, identifiez le code .NET bogué et comprenez les raisons.

- Bénéficiez d'une vue d'ensemble des performances de vos applications et identifiez les goulets d'étranglement, dans le code ou la base de données
- Trouvez rapidement les fuites de mémoire et optimisez la mémoire de vos codes C# et VB.NET
- Comprenez et déboguez le code de tiers, frameworks, composants et bibliothèques inclus
- Standardisez la gestion des performances de votre équipe de développement



oXygen XML Editor Professional à partir de € 355



Éditeur XML multiplateforme supportant la plupart des technologies XML.

- Distribuez/actualisez facilement des plug-in/frameworks pour Oxygen
- Simplifiez la configuration des projets XML avec Master Files
- Affichage des modifications et commentaires de révision dans légendes
- Support d'édition visuelle convivial pour DocBook, DITA, TEI, XHTML
- Validez les documents XML avec les schémas XML, Relax NG, DTD, NVDL et Schematron



DevExpress DXperience à partir de € 1 091



Tous les outils DevExpress ASP.NET, WinForms, Silverlight, WPF et IDE Productivity en un.

- Abonnement de 12 mois pour tous les produits et mises à jour DevExpress et accès aux versions bêta en développement actif
- Modèles et thèmes d'application intégrés exceptionnels
- Support de nouvelle vue Windows 8 UI et panneaux ancrables tactiles
- Support interface codée pour test environnement utilisateur

Les 24h de l'informatique

Organisé par les départements d'informatique des IUT de France, la 4e édition du concours de programmation des étudiants d'IUT informatique s'est déroulée les 16 et 17 mai à l'IUT de Strasbourg.

180 étudiants, venus de toute la France, répartis en 32 équipes, sont venus s'affronter amicalement dans la grande salle de l'ESTII de Strasbourg. Au programme, une véritable épreuve d'endurance : 24h de programmation non-stop. Chaque année cette manifestation, qui s'adresse aux étudiants en IUT informatique (43 départements en France), leur permet de confronter, dans une ambiance détendue, leurs qualités de programmation, de travail en équipe, d'endurance...

Il est 13h lorsque les portes s'ouvrent et que les premières équipes s'installent. 40 minutes plus tard, les compétiteurs sont réunis dans un grand amphithéâtre pour la présentation détaillée de la compétition. Trois épreuves de 8h chacune sont au programme : programmation pure, programmation web et hacking.

14h sonnent et le coup d'envoi de la première épreuve est donné. L'objectif est d'écrire un programme gérant l'intelligence artificielle du jeu « Quoridor » en réseau. L'équipe d'organisation a développé, en Java, le serveur du jeu, chaque équipe d'étudiants doit développer un client (le choix du langage est libre, la communication avec le serveur se faisant par socket) qui joue mieux que celui des concurrents...

Les équipes peuvent travailler avec leur PC, leur langage de développement, leur environnement... Seule contrainte : le client doit pouvoir s'exécuter sur les machines Linux pour la « bataille » finale entre les 32 équipes. Bien entendu, le règlement interdit certaines pratiques... un groupe reçoit d'ailleurs un avertissement pour avoir tenté de perturber les autres équipes en « floodant » les serveurs. L'ambiance est très détendue, autour d'un café ou d'un soda, les équipes n'hésitent pas à bavarder entre elles ou avec les enseignants



Les vainqueurs....



« Cette compétition nous a beaucoup appris sur le travail d'équipe » nous disent les membres de « Hey Team », « (...) c'est un véritable défi non seulement de programmation mais aussi un défi physique, savoir se ménager, savoir faire des pauses pour rester efficaces ».

Les portes se ferment sur la grande salle où, déjà, l'équipe d'organisation s'active pour ranger les serveurs, routeurs, switchs et câbles, pendant que les étudiants reprennent le train ou la voiture... Eh oui, lundi matin il faudra reprendre les cours !

 Mattieu Simonet
Enseignant, IUT DIJON

 Alexandre Guidet
Président de l'assemblée des chefs de département informatique

Sur vos agendas

Droidcon Paris 2014 : l'événement Android est enfin annoncé pour les 22 et 23 septembre. Cette nouvelle édition sera organisée par le PAUG et BeMyApp.

Forum PHP : 23-24 octobre 2014, la référence PHP de l'automne en France !

Open World Forum : 29 octobre au 1er novembre

Drupagora, 14 novembre : la communauté Drupal se réunit pour parler projets, futur de la plate-forme, développement.



Montée en charge linéaire et extrêmement performante



Pour applications .NET et Java
(supporté sur Windows Azure et Amazon AWS)



Les données en cache (via NCache), réduisent les accès coûteux en base, et permettent à vos applications de monter en puissance vers “extreme transaction processing” (XTP). TayzGrid est une implémentation native 100% Java de NCache.

Cache distribué en mémoire

- Extrêmement rapide et montée en charge linéaire avec 100% uptime
- Topologie en Miroir, Répliquée, Partitionnée et Cache Client
- NHibernate et Entity Framework cache niveau 2

Optimisation ASP.NET de Web Farms

- ASP.NET Session State cache
- ASP.NET View State cache
- ASP.NET Output Cache provider

Partage de données en mode Runtime

- Notifications d'événements puissants pour le partage de pub / sub données



Alachisoft

sales@alachisoft.com
US: +1 (925) 236 3830

Siège de l'entreprise

Télécharger un essai GRATUIT!

www.alachisoft.com

RedFabriQ

info@redfabriq.com
Tel: +33 1 40 16 07 89
Distributeur et intégrateur
en France

Pas contents des salaires mais contents du job !

Jobprod publiait au printemps dernier une étude sur les salaires des développeurs. Deux chiffres clés : 57 % des développeurs ne sont pas satisfaits de leur salaire mais 78 % s'épanouissent au travail ! Etonnement, ils ne sont que 19 % à vouloir une augmentation...

Sur les salaires, les moyennes restent assez stables : 30 k€ pour un développement avec - 2 ans d'expérience, 33 k€ pour un lead développeurs et un expert technique / architecte. La région parisienne affiche un salaire moyen de 42 k€ contre 35,8 en province.



Parmi les spécialisations, on constate, selon l'étude, que les profils Python et Ruby connaissent généralement une majorité de 11,6 %, par contre, - 13 % pour un développeur PHP et un stabilité pour les développeurs natifs et mobiles.

Les développeurs sont aussi concernés par la présence d'une part variable dans le salaire, à peu près d'un tiers des développeurs seraient concernés, particulièrement dans les petites structures et les agences.

Jobprod met aussi en avant l'existence d'un salaire différent entre homme et femme, en moyenne 15 %. L'écart est surtout flagrant dans les profils ayant + 5 ans d'expérience. Pour en savoir plus : http://www.slideshare.net/slideshow/embed_code/34958852

Aux jeunes créateurs d'entreprises Innovantes...

Vous êtes un jeune créateur d'entreprise vous avez un bon projet et vous avez des associés potentiels. Vous avez proposé votre projet à OSEO et vous avez eu une réponse positive avec une promesse de subvention ou d'avance à la clef. OSEO vous explique que l'aide est nominative et qu'en tant que porteur du projet, c'est à vous de signer les papiers, ATTENTION ! Lorsque vous présentez votre projet à OSEO, vous préparez une liste de dépenses prévisionnelles conformes à la liste des dépenses prises en charge. Soyez très prudent car l'aide est proportionnelle au montant total des dépenses (environ 50%) du montant TTC puisque vous n'avez pas encore créé votre entreprise. Vous n'avez pas la possibilité de récupérer la TVA puisque vous ne pourrez le faire qu'après avoir créé l'entreprise et surtout, uniquement pour les dépenses postérieures à sa création.

Lorsque vous créez votre projet de dépenses, vous le faites souvent HT. Comme les montants avancés par OSEO sont importants, la différence est loin d'être négligeable. Voici quelques conseils qui pourraient vous éviter de vivre une

expérience assez désagréable. Avant tout, faites en sorte que vos associés potentiels se portent caution solidaire (même s'ils abandonnent le projet ultérieurement) pour la différence entre le montant avancé par OSEO et le montant que vous avez prévu de dépenser. S'ils vous lâchent en route, vous pourriez être seul à devoir déboursier cette somme pour ne pas avoir d'ennuis avec OSEO.

Chassez les étoiles que vous aurez dans les yeux à l'annonce du montant de l'aide car, même s'il est important, il est soumis à une réglementation très stricte et vous ne pourrez pas en faire ce que vous voulez. Les dépenses prises en charges sont souvent inutiles au stade embryonnaire où en est votre entreprise. Ne perdez pas de vue que vous ne servirez que de « passe plats » pour que l'argent soit transféré d'OSEO à vos prestataires.

Enfin, c'est beaucoup d'effort pour obtenir plus de contraintes que d'aide. Accepter l'argent d'OSEO c'est prendre le risque de voir l'Etat chercher à vérifier ce que vous avez fait de l'argent et ce, pas forcément au moment où vous vous y attendriez.

Un répéteur Wifi plus compact

Belkin propose un nouveau répéteur Wifi, très discret. En quelques minutes, il est opérationnel grâce à un assistant de configuration rapide, accessible depuis un mobile ou un PC. Il fonctionne avec la plupart des routeurs du marché.



L'intérêt d'un répéteur est d'amplifier le signal Wifi et d'améliorer la connectivité.

Prix conseillé : 34,90 €

Nouvelle carte PNY

PNY Technologies annonce la PNY GeForce GT 730. Compatible PCI Express 3, elle possède des sorties DVI, VGA et HDMI, 384 coeurs CUDA, 1 Go de mémoire, une fréquence de 902 mHz. Elle suffira pour la bureautique, l'usage courant.

Prix public : 54,90 €

NEW HOSTING

AVEC LES MEILLEURES APPLICATIONS !

WordPress & 140 Apps populaires

- Support dédié 24/7 assuré par les experts 1&1
- **Versions d'évaluation disponibles**
- Notifications de sécurité : mises à jour, patchs...
- Performance optimale : 2 Go de RAM garantis
- Service complet pour 140 Apps & CMS réputés : WordPress, Joomla!™, Drupal™, TYPO3, Magento®

Outils de référence

- PHP 5.5, Perl, Python, Ruby
- 1&1 Mobile Website Builder
- NetObjects Fusion® 2013 – 1&1 Edition

Marketing efficace

- 1&1 Référencement Pro
- 1&1 Newsletter Tool
- 1&1 WebStat
- Crédits Facebook®

Infrastructure High-Tech

- Disponibilité maximale grâce à la **géo-redondance**
- Connectivité > 300 Gbits/s
- 1&1 CDN powered by CloudFlare® (23 PoPs)

Tout inclus

- Nom de domaine : .fr, .com, .net, .eu, .org, .info
- Ressources illimitées : espace Web, trafic, comptes email et bases de données MySQL
- Système d'exploitation Windows ou Linux



PACKS COMPLETS
POUR PROFESSIONNELS

À partir de
0,99
€ HT/mois*

0970 808 911
(appel non surtaxé)

1&1

*30 jours « satisfait ou 100 % remboursé ». Les packs hébergement 1&1 New Hosting sont à partir de 0,99 € HT/mois (1,19 € TTC) la 1ère année au lieu de 2,99 € HT/mois (3,59 € TTC) pour un engagement de 12 mois. À l'issue des 12 premiers mois, les prix habituels s'appliquent. Offres sans durée minimale d'engagement également disponibles. Offres à durée limitée et soumises à conditions détaillées disponibles sur 1and1.fr.

1and1.fr

Petite introduction aux expressions régulières

Rentrons tout de suite dans le vif du sujet avec :

```
^([25[0-5]|2[0-4]\d|[0-1]?\d?\d)(.(25[0-5]|2[0-4]\d|[0-1]?\d?\d)){3}$
```

Qu'est-ce que c'est ? Qu'est ce que ça signifie ? C'est tout simplement un REGEX pour vérifier la validité d'une adresse IP...

Principes fondamentaux

Les expressions régulières, appelées aussi Regex (traduction littérale de REGular EXpression) sont un outil permettant d'agir sur une chaîne de caractère de 3 manières :

- valider une syntaxe (qui est l'utilisation la plus courante) ;
- remplacer une chaîne par une autre ;
- découper une chaîne.

Les Regex sont implémentées dans la plupart des langages de programmation (Python, Java, PHP, C#, etc.). Dans cet article, nous pratiquerons par l'exemple avec des *snippets* (extraits de code) en C#. La syntaxe étant à première vue assez complexe, nous trouverons de nombreux tableaux « aide mémoire » sur le web expliquant les différents caractères utilisés dans les Regex (exemple : site MSDN). L'utilisation du moteur de Regex en C# se fait par l'ajout de l'espace de nommage `System.Text.RegularExpressions` en en-tête de classe.

Méthode `IsMatch()`

La méthode `IsMatch()` est la plus couramment utilisée dans les cas de validation de chaînes de caractères. Voici des exemples très simples d'application : Exemple 1 : Validation d'une adresse email formatée de la façon suivante `XXX@XXX.XXX`

```
public static bool ValidMail(string adresse)
{
    //Note : Placer @ pour éviter de doubler les caractères
    //d'échappement de la chaîne
    var myRegex = new Regex(@"^([\w]+@[.\w]+\.\w+)$");
    return myRegex.IsMatch(adresse);
}
```

Exemple 2 : Validation de la saisie des données dans une zone de saisie de type formulaire (TextBox)

```
// ([0-9])  -> valeur de 0 à 9
// (^)       -> Qui commence par
// ($)       -> Qui se termine par
if (!Regex.IsMatch(textBox.Text, "^[0-9]+$"))
{
    textBox.Text = string.Empty;
    MessageBox.Show("Erreur : valeur attendue de type numérique");
}
```

Méthode `Replace()`

La méthode `Replace()` sert, comme l'indique son nom, à remplacer une chaîne par une autre, dans le même esprit que la méthode « `replace` » de la classe « `String` ». Exemple 3 : En supposant que le path suivant « `H:\app\new\data` » doit être reformaté de cette manière : « `H:\app\new\data\` »

```
//return oldPath.Replace(@"\", @"\");
//équivalent
return Regex.Replace(oldPath, @"\", @"\");
```

Exemple 4 : Avec un pattern de remplacement plus complexe : on recherche des doublons

```
string input = "Les Google Glass Glass sont des lunettes lunettes... ";
```

```
string pattern = @"(\w+)\s(\1)";
// (\w+) -> un ou plusieurs caractères alphabétiques.
// \s    -> un espace blanc.
// (\1) -> la chaîne dans le premier groupe capturé
// \W    -> un caractère non-alphabétique, notamment l'espace
// blanc et ponctuation.

foreach (Match match in Regex.Matches(input, pattern, RegexOptions.IgnoreCase))
{
    Console.WriteLine("Mots dupliqué '{0}' trouvé aux positions {1} et {2}.",
        match.Groups[1].Value,
        match.Groups[1].Index,
        match.Groups[2].Index);

    //Résultats
    //Mots dupliqués 'Glass' trouvés aux positions 11 et 17.
    //Mots dupliqués 'lunettes' trouvés aux positions 32 et 41.
```

Méthode `Split()`

Les REGEX offrent 2 possibilités pour découper une chaîne :

- Le découpage par séparateur ;
- Le découpage par regroupement.

Exemple 5 : on découpe une chaîne en fonction d'un ou plusieurs séparateurs.

```
String s1 = "Oculus Rift : la réalité virtuelle à portée de code ?";
char[] seps = { ' ', '\n', ',' };
// séparateur = espace, retour chariot, virgule
String[] values = s1.Split(seps);
foreach (string value in values)
{
    Console.WriteLine(value);
}
// Résultats
// Oculus
// Rift
// ...
```

Le découpage par regroupement qui permet de ranger dans un tableau les éléments de chaîne qui sont extraits en fonction du pattern et de les réutiliser ensuite. Un `MatchEvaluator` représente la méthode qui est appelée à chaque fois qu'une correspondance avec une expression régulière est trouvée lors d'une opération de méthode `Replace`.

Exemple 6 : A partir d'une chaîne donnée, séparer les lettres et chiffres par un caractère *

```
string input = "23x * y34x2";
// on sépare les groupes de chiffres/lettres par un *
Regex reg = new Regex(@"\d(?=[a-z])|[a-z](?=\d)");
string result = reg.Replace(input, delegate(Match m)
{
    return m.Value + " * ";
    // Résultats: 23 * x * y * 34 * x * 2
});
```

Pour conclure, les REGEX sont un outil très puissant, mais en contrepartie difficile à tester et débuguer à la volée. Il existe des outils en ligne qui aident à valider le pattern.

 Alice Barralon - Chef de projet mobilité
alice.barralon@gmail.com



LE CLOUD GAULOIS, UNE RÉALITÉ ! VENEZ TESTER SA PUISSANCE

EXPRESS HOSTING

Cloud Public
Serveur Virtuel
Serveur Dédié
Nom de domaine
Hébergement Web

ENTERPRISE SERVICES

Cloud Privé
Infogérance
PRA/PCA
Haute disponibilité
Datacenter

EX10

Cloud Hybride
Exchange
Lync
Sharepoint
Plateforme Collaborative

Devoxx France 2014, la suite !

Le mois dernier, nous avions déjà largement abordé la grande conférence Java : Devoxx France. Mais comme il restait encore beaucoup de matière à traiter, nous vous proposons Devoxx, la suite ! Au menu : Linux, JavaScript, des tests, des frameworks. Enjoy ! La Rédaction.

Linux 101

J'ai eu la chance de participer à mon premier Devoxx. Parmi les conférences auxquelles j'ai assisté, Linux 101 m'a beaucoup marqué. J'y ai vu un bel aperçu de commandes Linux directement utilisables au quotidien, grâce au très bon duo Pierre-Antoine Grégoire et Frédéric Weisbecker. Le premier travaille chez Agile Partner, et le second chez RedHat, sur le noyau du système. Même s'ils ont essayé dans un premier temps de nous faire croire qu'ils ne connaissaient rien à Linux, l'assistance n'a pas été dupe.

Nous avons réalisé un voyage de 3h dans l'univers Linux, avec pour seul compagnon le terminal, et quelques slides pour résumer de temps en temps. Notons que les sièges de l'auditorium de l'hôtel Mariott étaient confortables, et l'assistance réactive, ce qui a rendu la présentation encore plus sympathique.

Les thèmes abordés étaient très différents, en commençant par la description de l'architecture du file system, passant par les périphériques de sortie, les droits d'accès et la gestion des utilisateurs, l'historique de commandes ou encore les disques et les points de montage.

J'en ai retenu quelques commandes qui peuvent être utiles au quotidien :

► **stat** : afficher des informations sur des fichiers ou répertoires passés en arguments

```
$> stat /tmp
Fichier : "/tmp"
    Taille : 4096    Blocs : 8    ES blocs : 4096    répertoire
Device : 805h/2053d Inode : 3145729      Liens : 9
Accès : (1777/drwxrwxrwt) UID : ( 0/  root)  GID : ( 0/  root)
Accès : 2014-04-22 18:49:45.377529471 +0200
Modif. : 2014-04-22 19:02:48.809503235 +0200
Chgt : 2014-04-22 19:02:48.809503235 +0200
Créé : -
```

► **who** : afficher des informations sur les utilisateurs connectés

```
$> who
lucie    tty7          2014-04-25 09:52 (:0)
lucie    pts/0          2014-04-25 09:54 (:0)
lucie    pts/1          2014-04-25 10:55 (:0)
lucie    pts/7          2014-04-25 11:02 (:0)
lucie    pts/10         2014-04-25 17:03 (:0)
```

► **man** : affiche le manuel pour la commande passée en paramètre

```
$> man ls
LS(1)          User Commands      LS(1)
NAME
    ls - list directory contents
[...]
```

► **apropos** : liste les commandes du manuel qui comprennent dans leur description les mots passés en arguments

```
$> apropos apr
```



```
apropos (1) - Chercher le nom et la description des pages de manuel
ldap.conf (5) - LDAP configuration file/environment variables
```

► **which** : retrouver l'emplacement d'une commande passée en argument

```
$> which java
/home/dev/ouutils/jdk/bin/java
```

► **history** : affiche les dernières entrées de la console, avec leur id

```
$> history
[...]
516 apropos apr
517 stat /tmp
518 which java
519 history
```

► **!(bang)** : exécute la dernière commande comprenant l'argument. Peut également être utilisé avec l'id passé par history

```
$> !518
$> which java
/home/dev/ouutils/jdk/bin/java
```

► **locate** : permet de retrouver tous les fichiers comprenant le mot passé en argument. Attention, cette commande s'appuie sur une base de données qui est générée régulièrement, donc pas forcément à jour. Pour l'actualiser, utiliser la commande **updatedb**.

```
$> locate java
/etc/.java
/etc/java-7-openjdk
```

```
/etc/javascript-common
[...]
```

► **grep** : recherche la chaîne de caractères passée en paramètre dans les fichiers donnés

```
$> grep default index.html
<p>This is the default web page for this server.</p>
```

► **psTree** : affiche le graphe des processus en cours d'exécution

```
$> pstree
init—NetworkManager—dhclient
```

```
|
|           └─dnsmasq
|
└─3* [{NetworkManager}]
|
└─accounts-daemon—2* [{accounts-daemon}]
[...]
```

La liste de toutes les commandes présentées sont disponibles ici : <http://www.slideshare.net/zepag/linux-101-slidesfr>

En conclusion, cette conférence était selon moi très didactique, et orientée à juste titre vers les utilisateurs réguliers mais non experts de Linux.

● Lucie Harivel - Netapsys

SinonJS, l'ami du test unitaire Javascript

Nos *applis web* se veulent toujours plus élaborées, leurs pages privilégiant le confort de l'utilisateur et l'efficacité pour de meilleures interactions et une réactivité accrues. Les écrire, c'est, entre-autres, la tâche qui incombe au développeur "front".

Toujours plus de code JavaScript, davantage de fonctionnalités lui étant dévolues et fatidiquement plus de risques de bugs en tous genres... et de régression. Il est donc vital de nous assurer que ce code est testé, au même titre que celui s'exécutant sur un serveur. Pour ce dernier d'ailleurs, l'intérêt des tests unitaires ne se discute plus. Et pour cela, le développeur back-end Java dispose d'un attirail bien fourni (Mockito, PowerMock, EasyMock ...). Un développeur JavaScript devra-t-il réinventer la roue pour aboutir au même résultat ? Non, bien sûr ! Il y a Sinon.js ! Pour cette édition de Devoxx FR, Florent Duveau s'est proposé de nous faire découvrir cette librairie sous une approche « live coding » des tests unitaires. S'agissant d'un quickie, il disposait de 15 minutes !

Sinon.js : mise en bouche

Sinon.js ne cache pas ses inspirations, celles tout particulièrement du célèbrissime Mockito: son utilisation repose sur les entités suivantes :

- **Le Spy** : espion d'une fonction, s'assurant qu'elle a bien été appellée avec des arguments précis, qu'elle retourne bien la valeur attendue ou qu'elle a bien levé telle ou telle exception.
 - **Le Stub** : simule le comportement d'une fonction en lui faisant retourner une valeur précise. Il est également possible, à la manière du Spy, de vérifier les arguments d'appels, le nombre d'invocations, ...
 - **Le Mock** : provoquera un échec du test si la ou les fonctions associées ne sont pas invoquées de la manière attendue. Les attentes sont ainsi exprimées en amont de l'invocation et en aval de leur vérification.
- Sans oublier des modules prêts à l'emploi faisant gagner un temps précieux, tels le **FakeServer**, capable de simuler un serveur HTTP en quelques lignes de code.

On passe à table !

Pour la petite histoire, il faut savoir que Florent a fait du burger une de ses passions. Vous pouvez d'ailleurs consulter ses faits d'armes du côté de www.mybestburger.fr. Amusante coïncidence, pour un quickie programmé justement à l'heure de la pause déjeuner. L'exemple sur lequel se base la présentation est une petite application Web permettant de noter divers restaurants de burgers. Elle a été développée en HTML5/CSS avec Marionette et s'appuie notamment sur Backbone.js. L'application étant prête, Florent a garanti la stabilité du login tout au long de ce quickie, à grands renforts de tests unitaires - exécutés en ligne de commande avec Grunt.

► Une tentative de login réussie : tout changement de modèle va provoquer un rendu adéquat. Un **Spy** a été positionné sur la fonction *render* du prototype de la vue.

```
1  it('should call render function when model changes', sinon.test(function () {
2     var renderSpy = this.spy(HeaderItemView.prototype, 'render');
3     var header = new HeaderItemView();
4     header.model.set('auth', 'logged');
5     renderSpy.should.have.been.calledOnce;
6     header.model.clear();
7   }));

```

► Un *submit* (équivalent à un clic sur le bouton login) va appeler une seule fois la fonction *login* de la vue. Ici également grâce à un **Spy** sur la fonction du prototype.

```
1  it('should call login function when click on submit', sinon.test(function () {
2     var loginSpy = this.spy(HeaderItemView.prototype, 'login');
3     var header = new HeaderItemView();
4     header.render();
5     header.ui.submit.click();
6     loginSpy.should.have.been.calledOnce;
7   }));

```

► Un élément du DOM généré par l'action de rendu possède bien la valeur attendue. La valeur du modèle aura été préalablement '**Stubbée**' afin de simuler le cas d'un login réussi.

```
1  it('should render Hey Florent when logged = true in model', sinon.test(function () {
2     this.stub(header.model, 'toJSON').returns({logged: true, name: "Florent"});
3     header.render();
4     expect(header.$el).toContain("Hey Florent !");
5   }));

```

► Une erreur retournée par le serveur lors du login entraînera l'appel à la fonction *logFailed*. Ici le **FakeServer** a été paramétré pour répondre 401 (Non autorisé à accéder à la ressource) à la requête POST; avec utilisation de **Mock**.

```
1  it('should call logFailed when server respond error', sinon.test(function () {
2     var mockLogFailed = this.mock(header).expects('logFailed').once();
3     header.render();
4     this.server.respondWith('POST', '/user', [401, {}, '{"logged":false}']);
5     header.login();
6     this.server.respond();
7     mockLogFailed.verify();
8   }));

```

D'ailleurs, grâce aux librairies sinon-chai et chai-jquery, les objets peuvent être assertés directement (*loginSpy.should...* *\$.el.should...*), rendant le code plutôt élégant et lisible.

● Jérôme Canler - Ingénieur Concepteur chez Soat

RxJava et l'enfer des callbacks

Devoxx France propose des sessions courtes, des Quickies, durant le déjeuner. Ces sessions permettent de brièvement présenter des outils, des méthodes de conception ou bien des frameworks. Découvrons RxJava.

Asynchrone

Les applications actuelles sont de plus en plus réparties. Il faut appeler plusieurs webservices, une base de données, et à partir de ces réponses, composer un résultat. Les appels à ces différents services sont bloquants et peuvent donc allonger le temps de traitement : il faut attendre la réponse venant du premier service avant de pouvoir appeler le second. Le code est dans l'obligation d'attendre : il est bloquant **Fig.1**. L'asynchronisme permet de lancer des tâches, tandis que son code continue de s'exécuter. Un exemple typique est d'effectuer l'appel à un webservice, et de continuer un traitement le temps que ce webservice réponde. On exploite alors ce temps d'attente pour effectuer d'autres tâches nécessaires à la réalisation de son service **Fig.2**.

Futures

Une manière d'effectuer un appel asynchrone en Java est de passer par les Futures. Le problème est qu'écrire des Futures dépendant d'autres Futures peut devenir complexe, tout comme il est difficile de créer un Future composé d'autres Futures.

L'API Future expose également une méthode 'get' qui peut bloquer prématurément un traitement, gâchant tous les efforts pour rendre son code non bloquant.

Callbacks

Une alternative possible aux Futures est l'utilisation de callbacks. Ce pattern est familier des développeurs javascript utilisant jQuery pour des appels 'Ajax'.

```
$.get( "/users/JakeWharton/repos", function( data ) {
    $( ".result" ).html( data );
    alert( "Load was performed." );
});
```

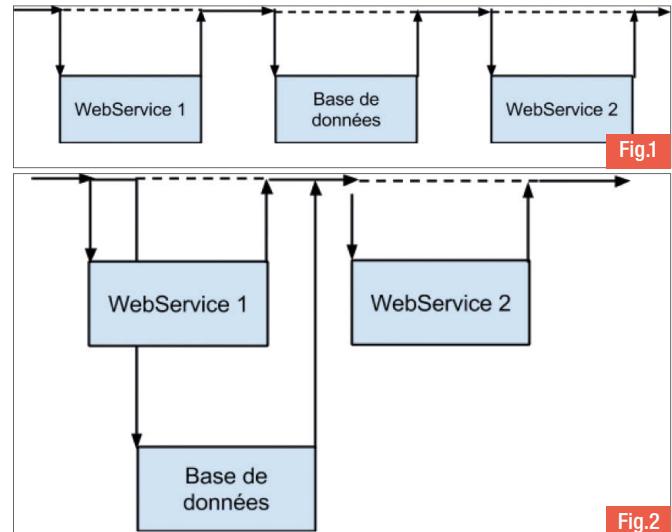
La fonction passée en paramètre de la méthode \$.get sera appellée une fois la réponse du serveur obtenue. Via une callback, il n'est pas possible de bloquer prématurément un traitement. Ce système de callback pourrait être l'API idéale pour gérer des traitements asynchrones. Malheureusement, la dépendance entre callbacks est difficile à gérer dès qu'il y a plusieurs niveaux d'imbriques, c'est à dire quand une callback dépend d'un résultat d'une callback qui dépend elle-même du résultat d'une autre callback.

```
$.get("/users/JakeWharton/repos", function( data ) {
    $.get("/repos/JakeWharton/" + data.name + "/commits", function
( data2 ) {
    $( ".result" ).html( data2 );
    alert( "Load was performed." );
});});
```

RxJava

RxJava est un portage en java du framework Reactive Extensions, qui est à l'origine développé pour C#. Ce portage est réalisé par l'entreprise Netflix, qui utilise ce framework massivement en production.

Le framework s'appuie sur le pattern Observer/Observable pour le support de séquences de données/événements, tout en y apportant des méthodes de compositions.



Présentation de l'API

L'interface Observer permet de définir le comportement à adopter :

- En cas de publication d'une nouvelle valeur via la méthode "onNext(T data)",
- En cas de fin de production d'éléments de la séquence observée via la méthode "onCompleted()",
- En cas d'erreur via la méthode "onError(Throwable t)".

Les `Observer<T>` observent des `Observable<T>` qui sont donc des séquences d'objets de type T. On peut créer des `Observables` via des méthodes déjà fournies par l'API telles que `Observable.from(T... values)`, qui crée une séquence à partir de "values", ou encore `Observable.just(T oneValue)`, qui crée une séquence d'un élément unique. Il est également possible de construire son implémentation via `Observable.create()`.

On branche un `Observer` sur un `Observable` via la méthode `subscribe` de l'`Observable`. C'est donc la séquence observée qui notifie les observers.

```
Observable.range(1, 10).subscribe(l) -> System.out.println
("Data : " + l))
```

Ces `Observable<T>` proposent plusieurs méthodes que l'on retrouve en partie dans la programmation fonctionnelle. On y trouve des méthodes de transformation (map, flatMap), de filtrage (filter, skip, distinct, takeLast, elementAt), de combinaison (merge, zip, reduce), d'actions (groupBy, concat) mais aussi des méthodes de gestion d'erreurs (onErrorReturn, onErrorResumeNext, retry).

Exemple d'utilisation

Pour cet exemple, nous allons utiliser Java 8 pour les lambdas expressions et Retrofit 1.5.0, client REST pour Android et Java, qui permet, à travers son api, d'utiliser directement des observables. La version de RxJava utilisée sera la 0.17.0. Le but sera de récupérer la liste des "forks" de plusieurs utilisateurs de GitHub, puis la liste des commits de ces "forks". Il faut d'abord créer le stub de l'API Rest de GitHub

```
import java.util.List;
import retrofit.http.GET;
import retrofit.http.Path;
import rx.Observable;
```

```

public interface GitHubService {
    @GET("/users/{user}/repos")
    Observable<List<Repo>> listRepos(@Path("user") String user);
}

public class Repo {
    public String id;
    public String name;
    public String description;
    public boolean fork;
    public Owner owner;

    public static class Owner {
        public String login;
    };
}

```

Puis créer l'implémentation de ce service REST via Retrofit :

```

RestAdapter restAdapter = new RestAdapter.Builder()
    .setEndpoint("https://api.github.com")
    .build();

GitHubService github = restAdapter.create(GitHubService.class);

Tout ceci est spécifique à Retrofit. Manipulons maintenant le service en utilisant RxJava en récupérant la liste des repositories de l'utilisateur "JakeWharton" (qui est un des développeurs de Retrofit).

github.listRepos("JakeWharton").subscribe((repos) -> System.out.println("Repository : " + repos));

```

GitHub sera appelé non pas lors de l'utilisation de `github.listRepos("JakeWharton")` mais lors de la souscription à l'Observable, c'est-à-dire quand la méthode `subscribe` est appelée. L'utilisation de l'interface Observable permet de cacher le traitement effectué derrière l'appel : l'est-il d'une manière synchrone ou asynchrone ? Caché par l'Observable, il est facile de manipuler d'une manière uniforme des appels synchrones et asynchrones. Il est également possible de commencer une implémentation synchrone de l'Observable, puis la convertir en asynchrone. Cette modification, même si elle a lieu plus tard dans la vie du projet, n'impactera pas le code utilisant notre Observable. Pour obtenir la liste des repositories de plusieurs utilisateurs, c'est un petit peu plus complexe : nous allons créer un observable (séquence) de nom d'utilisateur (`Observable.from`) que nous allons transformer (via `flatMap`) en `Observable<List<Repo>>`. Pour information, "benjchristensen" est un développeur travaillant sur RxJava.

```

Observable<List<Repo>> reposOfUsers = Observable.from(Arrays.asList("JakeWharton", "benjchristensen"))
    .flatMap((user) -> github.listRepos(user));

reposOfUsers.subscribe((repos) -> System.out.println("Repository : " + repos));

```

La méthode `map` permet de convertir un élément de type A en type B. `flatMap` permet de convertir un élément de type A en 0, 1 ou plusieurs éléments de type B. Cette séquence de 0 ou de plusieurs éléments est représentée via un Observable. C'est pourquoi `flatMap` demande une lambda du type `Observable<A> -> Observable`. Dans l'exemple précédent, `flatMap` est utilisé pour convertir un `Observable<String>` en `Observable<List<Repo>>`. Pour récupérer la liste des "forks", il faut transformer un `Observable<List<Repo>>` en

`Observable<Repo>` qui contiendra 0, un ou plusieurs Repo. Il faut donc à nouveau utiliser la méthode `flatMap`. Sur ces Repo, nous ne garderons que les forks : opération simple à faire via la méthode `filter`.

```

Observable<Repo> forks = reposOfUsers.flatMap((repos) ->
    Observable.from(repos).filter((repo) -> repo.fork));
forks.subscribe((repo) -> System.out.println("Fork : " + repo));

```

La liste des commits d'un repository est obtenue via une autre méthode REST. Il faut modifier notre Stub en conséquence :

```

public interface GitHubService {
    @GET("/users/{user}/repos")
    Observable<List<Repo>> listRepos(@Path("user") String user);

    @GET("/repos/{user}/{repo}/commits")
    Observable<List<Commit>> commits(@Path("user") String user,
        @Path("repo") String repo);
}

public class Commit {
    public String sha;
}

```

Pour obtenir la liste des commits d'un repository, il est nécessaire d'appeler une méthode REST pour obtenir les informations du repository (cf ci-dessus) puis d'appeler une nouvelle méthode REST pour récupérer cette fois-ci les commits. Tout ceci peut être résumé via cette expression : `Observable<Repo> -> Observable<List<Commit>>`. Vous l'aurez compris, pour chaîner deux appels asynchrones, `flatMap` est le compagnon idéal.

```

Observable<List<Commit>> commits = forks.flatMap((repo) ->
    github.commits(repo.owner.login, repo.name));
    commits.subscribe((c) -> System.out.println("Liste des commits : " + c));

```

Au final, nos appels asynchrones peuvent être réduits à ce code :

```

Observable.from(Arrays.asList("JakeWharton", "benjchristensen"))
    .flatMap((github::listRepos)
        .flatMap(Observable::from)
        .filter((repo) -> repo.fork)
        .flatMap((repo) -> github.commits(repo.owner.login, repo.name)))
        .subscribe((c) -> System.out.println("Liste des commits : " + c));

```

Il est aisément, dès lors que l'on a compris le concept de `flatMap`, de chaîner des appels asynchrones, ou synchrones, sans avoir la mélasse de callback habituelle.

Reactive Streams

Ceci était une brève introduction, avec un cas pratique, à l'utilisation de RxJava. Les applications asynchrones deviennent de plus en plus courantes et RxJava est une solution permettant d'écrire ce type d'application plus facilement. Lors de ce Devoxx France 2014, l'initiative Reactive Streams a également été annoncée : l'idée étant de définir une API standardisée pour la manipulation de flux asynchrones. Plusieurs implémentations sont proposées tels que Akka Streams, Reactor Composable mais aussi RxJava. S'intéresser à RxJava aujourd'hui, c'est aussi s'intéresser aux standards de demain.

- ⌚ David Wursteisen – Expert Java JEE chez Soat
- ⌚ Arnaud Jolly – Ingénieur Concepteur chez Soat

MariaDB : rapprocher systèmes critiques et environnements fédérateurs

C'est dans le cadre de l'industrie du logiciel au sens large que j'ai pu observer l'évolution des bases de données. Skysql, The MariaDB Company, participe aux efforts permanents pour développer la technologie de base de données Open Source la plus performante au monde et, en parallèle, ses services de support, de conseil, de formation de Remote DBA répondent aux besoins et aux exigences des entreprises.

Cela m'a fait prendre conscience que les cas d'utilisation qui comportent les plus grands défis nécessitent tout à la fois une capacité de montée en charge sans précédent, une disponibilité continue, une grande souplesse opérationnelle et des performances hors pair. Rien que ça! Pourtant, ses nouvelles fonctionnalités puissantes et sa technologie de pointe permettent à la base de données MariaDB 10 de répondre à toutes ces exigences. MariaDB 10 est l'aboutissement de près de 2 ans de travail et de 10 versions de développement issues des efforts de programmation, de test et d'itération fournis par l'équipe technique et la communauté des utilisateurs en croissance rapide. Si elle s'appuie sur des codes testés en situation depuis des décennies et des millions d'installations MySQL, la base de données MariaDB 10 prend néanmoins de nouvelles orientations. Le cœur de l'équipe de MariaDB travaille sans relâche pour concilier, d'une part, les performances, la disponibilité et la sécurité critiques d'un SGBDR mature et, d'autre part, la capacité de montée en charge et la simplicité de développement des systèmes NoSQL. MariaDB 10 associe la puissance et la fiabilité de MySQL à des fonctionnalités qui permettent aux applications Web actuelles de gérer plusieurs millions de clients et aux équipes DevOps de ramener à quelques heures le cycle de développement. Découvrons ensemble rapidement les nouvelles fonctionnalités de MariaDB 10 et comment elles fédèrent les systèmes SGBDR et NoSQL dans un nouveau type de base de données. Patrik Sallner, CEO de Skysql, précise : «MariaDB vise à devenir la première plateforme de base de données pour le Cloud. En pratique, cela signifie que nous développons MariaDB sur 3 axes: plusieurs modes de réplication, l'interopérabilité avec d'autres bases de données, et la scalabilité nécessaire pour la performance dans le Cloud».

Réplication : les coulisses de la montée en charge et de la disponibilité continue

Alors que le prix du matériel chute et que le coût de l'infrastructure Cloud baisse encore plus vite, la montée en charge est assurée via l'adoption effi-

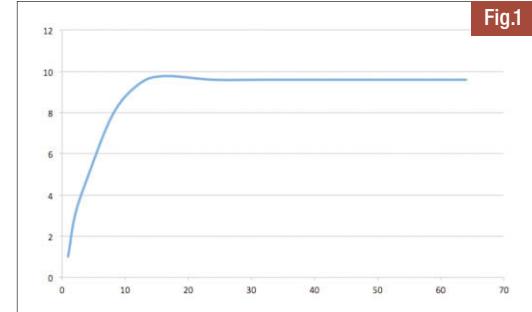
cace de toutes ces ressources peu onéreuses par des architectures informatiques massivement parallèles et capables de s'adapter pour supporter même des charges Web à grande échelle. Les bases de données montent également en charge de la sorte, mais répliquer les données et faciliter le parallélisme tout en assurant la cohérence de l'ensemble représente un enjeu algorithmique énorme. Les innovations les plus prometteuses fournies par MariaDB 10 offrent de nouveaux moyens pour répliquer les données et donner vie à de tout nouveaux cas d'utilisation auparavant hors de portée de l'écosystème MySQL.

Réplication esclave parallèle

Si vous souhaitez de nombreuses répliques de votre base de données pour faire évoluer l'accès aux données, il vous faudra, d'une manière ou d'une autre, maintenir toutes ces répliques à jour. MariaDB 10 offre une nouvelle mise en œuvre d'une idée ancienne: la réplication esclave parallèle. Et si toutes ces répliques esclaves pouvaient traiter en parallèle les mises à jour pour votre instance de base de données maître? Les bases de données esclaves pourraient gérer des charges de mise à jour bien plus conséquentes, tout en continuant de garantir une énorme montée en charge en termes de lecture. Pour déployer ces mises à jour esclaves parallèles, chaque événement de réplication doit être tagué au moyen d'un identifiant unique qui permet aux systèmes esclaves de traiter les mises à jour dans le bon ordre. MySQL 5.6 intègre un identifiant de transaction globale (GTID), ce tag unique que je viens d'évoquer. Mais grâce au nouveau déploiement GTID de MariaDB 10, MariaDB 10 peut désormais répliquer des mises à jour sur les bases de données esclaves avec un parallélisme beaucoup plus important. C'est une façon originale de dire que MariaDB sait gérer un nombre bien plus important de mises à jour par seconde, soit près de 10 fois plus de mises à jour par seconde qu'un processus de réplication (1).

monthread. Fig.1.

Et alors? Eh bien, cela implique moins de répliques, moins de ressources, un débit



Nombre de threads esclaves

supérieur et des connexions plus nombreuses. Autrement dit : vous économisez de l'argent ! Qui plus est, MariaDB peut gérer des charges de travail encore plus importantes. Bref, il s'agit d'une technologie de pointe qui offre des avantages évidents.

Réplication multi-source

Désormais, MariaDB 10 fournit toutes ces bases de données et applications qui traitent d'énormes volumes de transactions par seconde. Si vous êtes une enseigne d'envergure mondiale qui gère des centaines voire des milliers de boutiques physiques équipées de systèmes de point de vente (POS) avec une imposante présence sur le Web en matière d'e-commerce et un système de gestion de contenu sophistiqué, et que vous collectez des volumes incroyablement riches de données chaque seconde, chaque jour, comment diable stockez-vous toutes ces données dans votre entrepôt de données ou votre moteur d'analyse de Big Data afin de pouvoir les exploiter pour votre activité ?

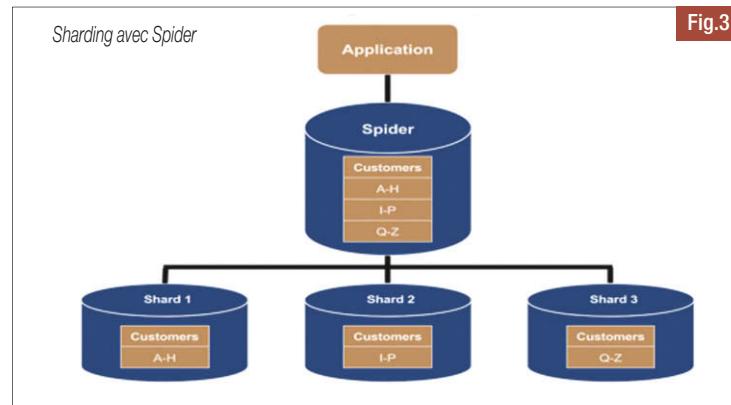
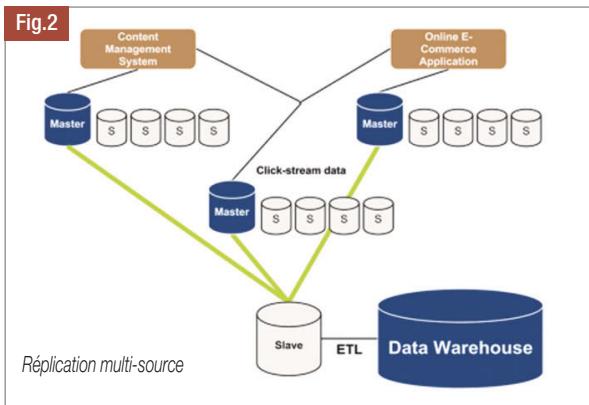
Imaginez que MariaDB 10 puisse tout simplement répliquer ces données dans vos applications analytiques appropriées. Eh bien, c'est le cas! En effet, à l'aide de la réplication multi-source, vous pouvez configurer une base de données esclave

(1) Note

Exécution avec sysbench 0.5 oltp.lua.

Réplication parallèle « In-order ». Table unique, 10 millions de lignes, log binaire anti-crash, transactions de taille moyenne, 1 mise à jour de clés primaires/transaction, 16 Go de pool mémoire.

Pour en savoir plus, rendez-vous sur <http://kristian-nielsen.livejournal.com/18435.html>.



pour qu'elle collecte des données issues de nombreuses applications et de multiples sites. Il est possible de fédérer au sein de tables de bases de données aussi bien des données transactionnelles que des flux de clics sur des sites ou encore des analyses et des évaluations collaboratives et de les transformer, les normaliser et les injecter dans votre application de traitement analytique. Le nouvel identifiant GTID de MariaDB 10 fait partie de la magie de l'ensemble. Il trie les données provenant de tous ces systèmes et applications hétérogènes et assure leur cohérence. La réplication multi-source simplifie également l'administration de toutes ces applications. Elle vous épargne beaucoup de problèmes et fait gagner du temps à votre équipe d'exploitation. Plus d'analyses, plus de précision, moins de problèmes et plus d'avantages grâce à la technologie de pointe intégrée à MariaDB 10, Fig.2.

Partitionnement (sharding) avec le moteur de stockage Spider Storage Engine

Imaginons que vous utilisez une application Web à grande échelle qui gère des millions de clients et d'enregistrements de profil. Cela implique beaucoup de traitement : votre magasin, votre système d'inventaire, votre moteur de recommandation, votre portail de service clients, votre.... Bref, vous voyez ce que je veux dire... Ces tables qui contiennent des millions de lignes se transforment en goulot d'étranglement. Et si vous pouviez partitionner ces tables sur de multiples serveurs et si vos applications pouvaient exécuter en parallèle de nombreuses opérations sur ces tables grâce à la même stratégie de montée en charge que celle déployée pour la réplication? Il s'agit d'une technique connue dans le monde des bases de données sous le nom de sharding ou partitionnement. Et même si elle très efficace pour traiter des tables gigantesques, elle exige souvent de vos développeurs qu'ils conçoivent la logique au sein de votre application. Et c'est là que les choses se compliquent !

Sharding avec Spider (Fig.3)

Et si MariaDB 10 pouvait partitionner ces énormes

tables en arrière-plan afin que les applications puissent continuer d'y accéder à l'aide d'une logique simple? Et si les requêtes exécutées sur ces tables pouvaient être diffusées une par une sur les nombreux serveurs en parallèle afin de supprimer ces goulets d'étranglement? C'est ce que le moteur de stockage Spider Storage Engine vous permet de faire. Grâce à cette option intégrée à MariaDB 10, vous définissez le mode de partitionnement des tables au sein de la base de données, et non pas à l'intérieur de l'application. Tout est transparent pour les développeurs. Mais votre application peut traiter des millions de lignes et des téra-octets de données dans chaque table de manière beaucoup plus performante. À l'échelle du Web et sans contrainte de développement, Spider est en outre une fonctionnalité conçue par un développeur tiers et intégrée à MariaDB 10 par l'équipe. Bref, une innovation plus rapide et développée, aussi, en parallèle !

Connecteurs MariaDB sous licence GPL

MariaDB fournit des connecteurs sous licence GPL. Ceci est très important pour tous ceux qui distribuent des applications embarquant MariaDB ou MySQL. Les connecteurs disponibles sont sous licence GPL :

Connecteur natif C/C++

Ce connecteur est compatible avec le connecteur legacy MySQL.

Il fournit une alternative GPL libre d'usage pour des développeurs C/C++ désirant se connecter à une base MariaDB ou MySQL.

Connecteur Java

Le Connecteur Java MariaDB est un connecteur JDBC de Type 4 qui permet à un développeur d'application Java de se connecter à des bases MariaDB ou MySQL.

Connecteur ODBC.

Le connecteur ODBC MariaDB pour les environnements Windows supporte le standard ODBC 3.5. Il peut être utilisé en remplacement du connecteur ODBC MySQL. Ce connecteur supporte

le mode ODBC ou ANSI.

Ce connecteur est construit au dessus de la librairie MariaDB C GPL. Ce connecteur ODBC est aussi sous licence GPL. Ce connecteur ODBC utilise le protocole binaire de «prepared statements» pour la communication client serveur.

Comment passer de MySQL à MariaDB ?

Pour toutes sortes de raisons pratiques, il est possible d'envisager MariaDB comme une mise à niveau de MySQL :

- Si vous utilisez Windows, reportez-vous aussi à [Upgrading MariaDB on Windows \(Comment mettre à niveau MariaDB sur Windows\)](#).
- Pour une mise à niveau à partir de versions très anciennes de MySQL, reportez-vous à [« Upgrading to MariaDB from MySQL 5.0 » \(Comment passer à MariaDB depuis MySQL 5 ou depuis une version plus ancienne\)](#).
- Dans la même version de base (par exemple 5.1), il vous suffit de désinstaller MySQL et d'installer MariaDB et le tour est joué. Il n'est pas nécessaire de purger ou de restaurer les bases de données. Comme pour toute mise à niveau, nous vous recommandons de sauvegarder vos données avant l'opération.
- Pour finaliser la mise à niveau, exéutez `mysql_upgrade` comme vous le feriez pour MySQL. Ceci est indispensable pour mettre à jour votre privilège `mysql` et vos tables d'événements par rapport aux nouveaux champs utilisés par MariaDB. Remarque : si vous utilisez un package MariaDB, `mysql_upgrade` est en règle générale exécuté automatiquement.
- Après le lancement de `mysql_upgrade`, vous devez redémarrer MariaDB afin que les nouveaux changements prennent effet.
- Tous vos anciens clients et connecteurs (PHP, Perl, Python, Java, etc.) fonctionneront comme avant (inutile de recompiler). En effet, MariaDB et MySQL utilisent le même protocole client tandis que les bibliothèques de clients sont compatibles au niveau binaire. Si vous le souhaitez, vous pouvez aussi utiliser vos anciens paquets de connecteurs MySQL avec MariaDB.

Comment mettre à niveau my.cnf ?

Toutes les options disponibles dans votre fichier my.cnf MySQL d'origine fonctionneront sans problème pour MariaDB.

Cependant, dans la mesure où MariaDB contient davantage de fonctionnalités que MySQL, vous devez tenir compte des facteurs suivants lorsque vous envisagez de modifier le fichier my.cnf.

- ▷ Par défaut, MariaDB utilise le moteur de stockage Aria, plutôt que MyISAM, pour les fichiers temporaires internes. Si vous avez beaucoup de fichiers temporaires, ajoutez et définissez aria-pagecache-buffer-size sur la même valeur que key-buffer-size.
- ▷ Si vous n'utilisez pas de tables MyISAM, vous pouvez définir key-buffer-size sur une valeur très faible, par exemple 64K.

Autres précisions :

- ▷ MariaDB utilise des versions LGPL du connecteur C et de Java. Si vous diffusez une application qui prend en charge MariaDB ou MySQL, vous devriez les utiliser !
- ▷ Pensez à tester le moteur de stockage TokuDB ou d'autres nouveaux moteurs de stockage fournis par MariaDB.

Voir aussi :

- ▷ MariaDB offre beaucoup de nouvelles fonctionnalités intéressantes.
- ▷ De plus amples informations sont disponibles sur la page d'installation de MariaDB.
- ▷ Il existe un screencast pour savoir comment passer de MySQL à MariaDB.

MariaDB Galera Cluster 5.5.37 -

MariaDB Enterprise Galera Cluster est une solution avancée de Base de Données relationnelle en cluster, développée pour la production et les environnements critiques.

Cette solution contient des fonctionnalités avancées comme :

- ▷ auto-provisioning et monitoring pour les entreprises
 - ▷ haute disponibilité
 - ▷ du support en 24x7x365 par l'équipe MariaDB.
- MariaDB Galera Cluster 5.5.37 est une version **stable** (GA). Il s'agit d'une fusion de MariaDB 5.5.37 et de Galera Cluster qui résout de nouveaux bugs.

Différents articles sur MariaDB Galera Cluster, dont les limitations connues, et "comment démarer", se trouvent dans la section **Galera** de la base de connaissances.

Pour la liste des modifications apportées à MariaDB Galera Cluster 5.5.37 avec les liens vers des informations détaillées sur chaque push, reportez-vous à la page de log des modifications MariaDB Galera Cluster 5.5.37 Changelog.

Remarques sur cette version :

- ▷ MariaDB Galera Cluster 5.5.37 comprend

MariaDB 5.5.37 avec des ajouts Codership (lp:codership-mysql/5.5 jusqu'à la révision 3980) et Galera 25.3.5.

- ▷ Cette version de MariaDB Galera Cluster prend en charge l'API wsrep v25, ce qui permet d'utiliser MariaDB Galera Cluster avec un provider wsrep 25.2.x ou 25.3.x Galera. Un provider wsrep 25.3.x est compris dans les référentiels MariaDB et les providers wsrep 25.3.x et 25.2.x sont disponibles sur la page des téléchargements.
- ▷ Cette version comprend une nouvelle méthode pour le transfert d'un état instantané, wsrep_sst_xtrabackup-v2. Cette méthode de transfert d'état instantané peut être configurée à l'aide de l'option –wsrep_sst_method=xtrabackup-v2. Son utilisation nécessite Xtrabackup (>= 2.1.6) et d'autres paquets logiciels tels que socat, nc et tar.

Reportez-vous aux Informations sur la version MariaDB 5.5.37 et au log des modifications pour en savoir plus sur les modifications apportées à MariaDB.

Remarque : si les versions Galera 25.2.x et 25.3.x sont toutes deux utilisées dans le cluster, MariaDB avec Galera 25.3.x doit être démarrée avec wsrep_provider_options='socket.checksum=1' afin d'être rétrocompatible avec Galera v2. Les providers Galera wsrep autres que 25.3.x ou 25.2.x ne sont pas pris en charge.

Résolutions des bugs... Profitez bien de MariaDB Galera Cluster !

Cette version contient des résolutions de bugs, des avertissements sur les erreurs de compilation ainsi que des améliorations apportées aux scripts existants. Une liste de tous les bugs résolus est disponible dans le log des modifications. Vous avez la possibilité d'être automatiquement averti(e) des nouvelles versions en vous abonnant à la liste d'annonce announce@mariadb.org (liste à faible trafic).

MariaDB est peut-être déjà intégrée à votre distribution préférée. De plus amples informations sont disponibles sur la page Distributions comprenant MariaDB (« Distributions which Include MariaDB »).

Log de modifications MariaDB Galera Cluster 5.5.37

Les liens associés aux numéros de révision pointent vers la page de révision sur Launchpad. Sur Launchpad, vous pouvez obtenir plus de détails sur la révision et visualiser les différences au niveau du code modifié dans cette révision :

- ▷ Revision #3486 Fri 2014-04-25 18:23:14 -0400
 - wsrep_sst_xtrabackup-v2 ajouté au serveur deb
 - Fichier .bzrignore mis à jour
- ▷ Revision #3484 Fri 2014-04-25 10:58:51 -0400
 - MDEV-6148 : résultats des tests mis à jour

SkySQL

SkySQL est le premier fournisseur mondial de logiciels de base de données supportant les dernières technologies de Cloud et de développement de pointe, ainsi que le support complet pour les environnements MySQL. SkySQL compte plus de 400 clients dans 30 pays à travers le monde, parmi lesquels : Canal +, ClubMed, Deutsche Telekom, La Poste, Virgin Mobile, Western Digital. Par ailleurs soutenu par un solide écosystème des principaux fournisseurs de technologies, telles que Red Hat et SuSE, SkySQL est parfaitement adapté pour aider les entreprises à faire face à tous les besoins en base de données.

- ▷ Revision #3484 Fri 2014-04-25 10:58:51 -0400
 - MDEV-6171 : message de résolution d'erreur pour split-brain
- ▷ Revision #3483 Thu 2014-04-24 22:52:07 -0400
 - Résolution des chemins de fichiers dans @TOKUDB_DEB_FILES@.
- ▷ Revision #3482 Mon 2014-04-21 21:45:19 -0400
 - MDEV-6148 main.flush_read_lock : échec en raison d'un verrouillage de lecture conflictuel
- ▷ Revision #3481 Fri 2014-04-18 19:16:46 -0400
 - WSREP_PATCH_REVNO mis à jour.
- ▷ Revision #3480 Fri 2014-04-18 16:56:44 -0400
 - * Doubloons inutiles de fichiers de scripts supprimés. * Tests MTR échoués mis à jour.
- ▷ Revision #3479 Fri 2014-04-18 07:51:52 +0300
 - Merge lp:maria/5.5 jusqu'à mariadb-5.5.37 révision 4148.
- ▷ Revision #3478 Wed 2014-04-16 13:08:29 +0300
 - Merge lp:codership-mysql/5.5 -r3961..3980.
- ▷ Revision #3477 [merge] Wed 2014-04-16 12:13:43 +0300
 - Merge de lp:maria/5.5 à MariaDB 5.5.37, révision 4154.
- ▷ Revision #3476 Thu 2014-03-20 18:33:23 -0400
 - L'arborescence mariadb-galera MDEV-5903 crée trop de paquets rpm
- ▷ Revision #3475 Fri 2014-03-14 17:54:53 -0400
 - Scripts debian modifiés pour des paquets galera autonomes.
- ▷ Revision #3474 Thu 2014-03-13 11:51:31 -0400
 - Correctifs pour scripts Debian.
- ▷ Revision #3473 Tue 2014-03-11 23:12:18 -0400
 - MDEV-5812 Fuites de mémoire lors de l'exécution de mysqld en mode bootstrap

● Jean Vidame

Apple veut-il la fin d'OpenGL ?

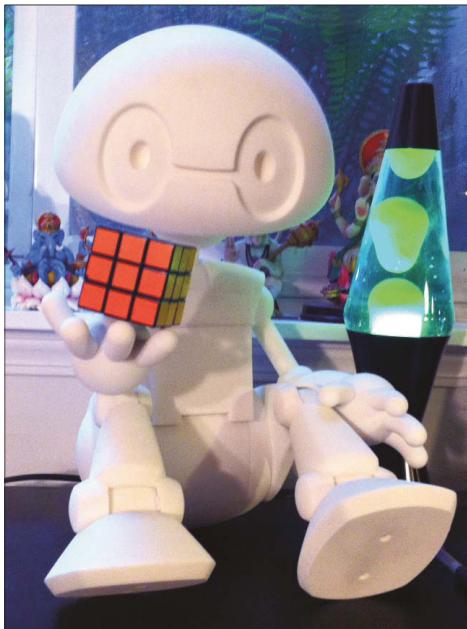
Lors de sa conférence développeur, Apple a dévoilé une nouvelle brique 3D très proche du matériel et du processeur : Metal. OpenGL, fortement soutenu par Apple jusqu'à présent, a un problème : il est souvent lent et d'un haut niveau d'abstraction. De ce fait, il profite mal des capacités CPU – GPU. Apple a montré clairement qu'il veut réduire la couche 3D nécessaire et s'appuyer sur la plate-forme processeur (A7 actuellement). Mais finalement, Apple ne fait que suivre Microsoft avec DirectX et AD avec Mantle. Apple a toutefois un avantage : ne s'occuper que d'une plate-forme, la sienne. Potentiellement, OpenGL sera oublié... Alex St. John, créateur de DirectX, pense que Metal signe la fin d'OpenGL pour Apple. « pourquoi aider Google à siphonner les développeurs de jeux en soutenant OpenGL » dixit Alex ! Metal n'a pas fini de faire parler de lui en attendant sa sortie officielle en automne prochain avec iOS 8. Les sessions techniques Metal sont disponibles sur le site www.developer.apple.com/wwdc

Fire Phone : le smartphone selon Amazon

Amazon a dévoilé le 18 juin dernier son premier smartphone, le Fire Phone. Il sera disponible fin juillet uniquement chez ATT (abonnement 2 ans). Pas de dates pour le reste du monde.

Le Fire Phone embarque : un écran 4,7 pouces, un processeur 4 coeurs, 2 Go de Ram, une puce graphique, un capteur photo de 13 millions de pixels. Il possède un bouton pour l'aide d'urgence et Amazon mise beaucoup sur la qualité et le confort de lecture. Amazon croit beaucoup à Firefly. Cette fonction permet, avec une simple photo, de reconnaître un produit (livre, DVD, etc.) et ensuite

vous pourrez l'acheter sur... Amazon. Amazon a aussi beaucoup travaillé sur les effets 3D, l'utilisation de la 3D dans les applications et l'interface. Fire Phone embarque la technologie Dynamic Perspective. Cette technologie crée à la volée les effets 3D selon la position de l'utilisateur et du téléphone, en plus d'objets 3D par défaut. Un SDK est disponible pour utiliser et intégrer Dynamic Perspective. Le principe est très séduisant. Un SDK Firefly est lui aussi disponible. Fire Phone utilise le système Android. Lien : <https://developer.amazon.com/public/solutions/devices/fire-phone>



Intel veut proposer un robot à imprimer soi-même !

Intel veut proposer un robot fonctionnel complet à imprimer et à monter soi-même : Jimmy. Robot humanoïde, Intel a l'ambition de proposer un robot coûtant environ 1600 \$. Le design matériel sera disponible gratuitement pour imprimer les éléments. Un kit complet sera disponible à la vente comprenant les moteurs, les processeurs, la batterie, etc. Intel fournit la mécanique et l'électronique et à vous d'imprimer le reste. Pour en savoir plus : <http://21stcenturyrobot.com>

Des écrans 4K pour tout le monde ?

C'est la dernière tendance à la mode. C'est au constructeur qui proposera le premier écran 4K le moins cher. Bien entendu, il faut faire très attention : quid de la qualité de la dalle et donc de l'image finale ? Apple avait lancé la guerre du « retina » sur les mobiles et les ordinateurs, maintenant on passe au 4K. Intel et Samsung ont annoncé une collaboration pour favoriser l'adoption du 4K pour les constructeurs ! A quel prix ? 399 \$! Intel parie sur des dalles de meilleures qualités à 999 \$. Cependant, actuellement, il est impossible de produire des dalles à ce tarif. Et surtout, il faut adapter les pilotes graphiques, la qualité des interfaces, etc. Ce qui prendra du temps. Ensuite, quid du contenu 4K qui est aujourd'hui quasiment inexistant !

Kinect est-il un boulet de performances pour la Xbox One ?

Microsoft propose une version Xbox One sans Kinect. Mais quelques jours plus tard, l'éditeur confirme que sa console sans Kinect a un surcroît de performances et de ressources graphiques. Et Microsoft veut que les développeurs en profitent avec de futurs outils de développement... Finalement, 24h après la parution de l'article sur eurogamer.net, Microsoft a respecté son propos : retirer Kinect n'aura pas d'impact sur les performances de la Xbox One, mais le SDK de Juin donnera accès à des ressources GPU réservées jusqu'à présent à la Kinect. Désidément, avec la Xbox One, Microsoft n'a pas de chance sur la communication... Source : <http://goo.gl/w1e38E>

AUTODESK INVESTIT SUR LE MARCHÉ DE L'IMPRESSION 3D...

Autodesk est connu par ses outils de CAO et de 3D. Il maîtrise donc la chaîne logicielle de la 3D mais mi-mai dernier, l'éditeur a annoncé un projet surprenant : Spark. Cette plate-forme comprend une partie logicielle et une partie matérielle. Le logiciel est important dans l'impression 3D pour créer et gérer les modèles 3D des objets à imprimer, mais le constat est qu'aujourd'hui, on trouve tout et n'importe quoi. Avec le logiciel Spark, Autodesk veut proposer une nouvelle approche pour rationaliser la chaîne logicielle. Ces outils seront proposés aux constructeurs. Et l'éditeur en profite pour annoncer la sortie dans les prochains mois de sa propre imprimante 3D...



Aperçu de la préversion de Visual Studio 2014

Visual Studio est entré dans un cycle de releases beaucoup plus rapide qu'auparavant, depuis la version 2012. C'est pour ça qu'un peu plus de 6 mois après la sortie de Visual Studio 2013, Microsoft nous propose déjà une version 14 à tester. Cela confirme la volonté de l'équipe de Visual Studio et du framework .NET d'obtenir du feed-back le plus tôt et le plus souvent possible pour nous fournir un produit d'une qualité toujours meilleure.

Cette CTP nous propose de nombreuses nouveautés dans 3 grands domaines. Le .NET Compiler Platform (aka "Roslyn") est beaucoup mieux intégré à Visual Studio qu'il ne l'était lors de la preview faite à la conférence BUILD. L'équipe C++ fait un bond en avant dans l'intégration du standard C++ 14. Enfin, ASP.NET vNext annonce un futur encore plus souple en terme de déploiement et de développement. Bien qu'il soit trop tôt pour pouvoir évoquer tous les détails, je vous propose de parcourir les nouveautés les plus marquantes.

.NET Compiler Platform (aka "Roslyn")

Lors de la BUILD, ce compilateur a été rendu open source (disponible ici : <https://roslyn.codeplex.com/>). C'est une bonne nouvelle, car cela permet notamment à n'importe qui d'ajouter des fonctionnalités d'analyse de code en profitant des mêmes API que celles utilisées par Microsoft, pour C# ou pour VB. Dans cette CTP, quelques exemples sont proposés, comme un refactoring plus évolué... Fig.1.

... ou encore l'apparition du nouvel opérateur " ? " qui permet de simplifier l'écriture lors de la gestion des références null Fig.2.

Dans l'exemple ci-dessus, si `mavariable` est null le code renvoie null (et ne génère pas une `NullReferenceException`), sinon le code renvoie la valeur de `UnString`. Pour activer cette fonctionnalité, il est nécessaire d'éditer le fichier `.cspoj` manuellement et de préciser qu'il faut utiliser la version expérimentale du langage : Fig.3.

Du côté de Visual Basic, on voit apparaître des nouveautés sur le refactoring et la possibilité d'écrire une chaîne de caractères sur plusieurs lignes. Ce sont de petites modifications, mais c'est surtout une manière de montrer les possibilités de la plateforme. Désormais, si vous pensez à une nouvelle fonctionnalité, n'hésitez pas à l'implémenter !

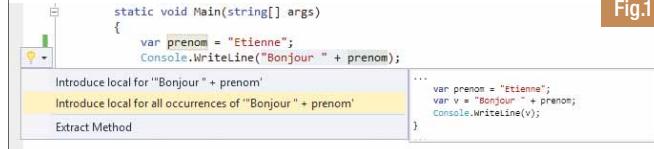


Fig.1

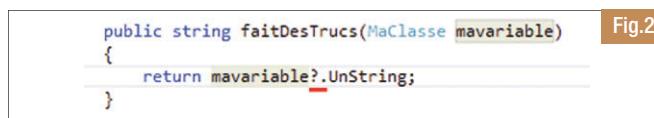


Fig.2



Fig.3

Vers C++ 14

C++ vit vraiment de beaux jours au sein de Visual Studio. Entre les nouveautés déjà disponibles dans cette CTP et celles qui sont à venir, il apparaît clairement que ce langage n'est pas délaissé. La roadmap présentée par Somasegar (VP de la division développeurs de Microsoft) montre un cycle de release rapide pour ce langage. On remarque que les nouveautés déjà nombreuses dans cette CTP seront certainement complétées par d'autres lors d'une prochaine version (peut-être la finale ?) Fig.4.

On y voit très clairement les nouveautés de cette CTP : telles que le support des C++14 generalized lambda capture, inline namespaces, ou encore user-defined literals. À titre d'exemple, certains headers C++ fournis en standard incluent maintenant des littéraux définis de cette manière. Par exemple :

Fig.5.

VC++ Conformance Update

Fig.4

VC++ 2013 Preview	VC++ 2013 RTM	VC++ Nov 2013 CTP	VS "14" CTPs (upcoming)
Explicit conversion operators	Non-static data member initializers	<code>_func_</code> <code>Extended sizeof</code>	Thread-safe function local static <code>init</code>
Raw string literals	<code>= default</code>	Implicit move generation	<code>noexcept (unconditional)</code>
Function template default arguments	<code>= delete</code>	Ref-qualifiers: & and && for <code>*this</code>	<code>alignof</code> <code>alignas</code>
Delegating constructors	"using" aliases <code>C++14 libs: type aliases</code>	<code>C++14 decltype(auto)</code>	Inheriting constructors
Uniform init & initializer lists	<code>C99 variable decls</code> <code>C99 _Bool</code>	<code>C++14 auto function return type deduction</code>	User-defined literals
Variadic templates	<code>C99 compound literals</code>	<code>C++14 generic lambdas (partial)</code>	<code>thread_local</code> Attributes
<code>C++14 libs: cbegin/greater<>/make_unique</code>	<code>C99 designated initializers</code>	<code>C++14 TS? async/await (partial)</code>	<code>C++14 generalized lambda capture</code>
			<code>C++14 libs: std:: user-defined literals</code>
			constexpr (except ctors, literal types)

Notes : plus de 400 bugs du compilateur ont été corrigés dans cette nouvelle version !

On peut également noter, en bonus, une fonctionnalité qui est apparue dans la version de novembre 2013 et incluse dans cette CTP : les méthodes *resumable* et le mot clé *await* qui permettent d'obtenir un résultat similaire à *async/await* en C# ou VB. Cette fonctionnalité ne vient pas du standard C++14, mais a été proposée pour C++17.

En dehors des nouveautés du langage (que vous pouvez retrouver ici : <http://j.mp/vs14cpp>) Visual Studio 14 propose de nouveaux outils, tels que l'analyse dynamique de mémoire native pour les applications Windows Desktop. C'était déjà disponible pour les applications .Net notamment, mais vous pouvez maintenant détecter plus facilement les fuites mémoires ou simplement optimiser la mémoire utilisée ! **Fig.6**.

ASP.NET vNext

Mobile-First, Cloud-First ! C'est ce que titrait le blog de Somasegar dans un billet en mai dernier. Il y annonçait, notamment, le support en preview d'Apache Cordova pour le développement multiplateforme en HTML/CSS et JavaScript. Il parle également d'ASP.NET vNext dont une partie est désormais accessible dans cette CTP et dont un des atouts majeurs sera de permettre de développer pour le cloud et/ou pour l'hébergement local de la même manière, ou en tout cas avec le moins de friction possible.

L'objectif de la prochaine version d'ASP.NET est plus que jamais de nous simplifier la vie. Les frameworks ont été rationalisés et regroupés dans un seul modèle de programmation pour MVC, Web API, Web Pages, SignalR et Entity Framework. Les applications développées avec ASP.NET vNext seront prêtes pour le Cloud sans effort particulier. Les services tels que le service de session state ou le cache s'adaptent automatiquement en fonction que l'application tourne dans le cloud ou dans un serveur classique. Passer d'un hébergement *on-premise* à un hébergement dans les nuages ne sera plus un problème : c'est le code d'ASP.NET qui change, pas le vôtre.

Grâce à l'utilisation de Roslyn notamment, il n'est plus nécessaire de compiler le code dans Visual Studio pour voir les modifications apparaître dans le navigateur quand on développe. Un simple rafraîchissement dans le navigateur et voilà ! Si on regarde de plus près un projet ASP.NET vNext, la première chose que l'on remarque, est que le fichier qui définit le projet n'est pas un standard .csproj ou .vbproj, mais .kproj. Il ressemble fortement à un fichier

csproj classique, mais ne contient aucune référence. Ceci est lié à l'utilisation à de nuget pour distribuer ASP.NET. Le projet ne référence plus un framework, mais un ensemble de dépendances arborescentes. Quand on crée le projet, Visual Studio récupère l'ensemble des dépendances nécessaires. La liste de ces

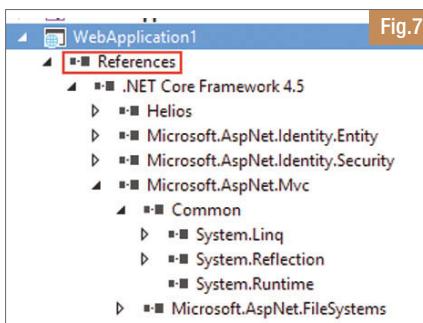


Fig.7

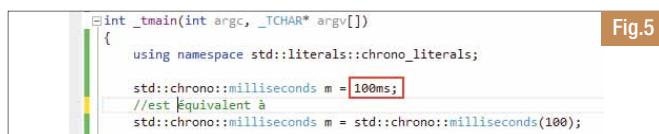


Fig.5

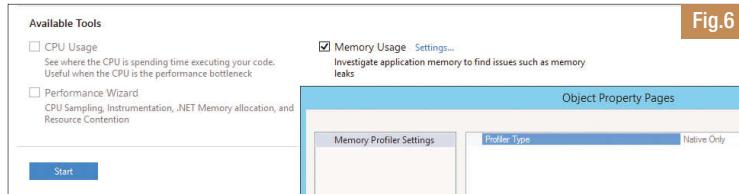


Fig.6

Important :

Visual Studio 14 est actuellement en version **Community Technology Preview (CTP)** ce qui signifie que vous ne devez pas l'utiliser pour écrire du code de production. De plus cette version ne supporte pas l'installation côté à côté avec une autre version : évitez de l'installer sur votre machine de développement. Vous pouvez :

► Télécharger cette version sur le site de Microsoft : <http://go.microsoft.com/fwlink/p/?LinkId=400850>

► Créer une machine virtuelle dans votre souscription Microsoft Azure et utiliser le template Visual Studio Professional 14 CTP 1

note : il n'y a pas de Team Foundation Server 14 CTP à l'heure actuelle. Brian Harry, le Program Manager de TFS a annoncé sur son blog que les nouveautés sont délivrées sur Visual Studio Online comme toujours et qu'il n'y a pas lieu à générer une CTP à ce stade.

dépendances est stockée dans un fichier nommé **Project.json**. Il peut paraître un peu complexe, mais heureusement, Microsoft fournit l'intelliSense pour nous faciliter la vie **Fig.7**. Dès qu'une modification est réalisée dans le fichier **Project.json**, une opération **packages restore** nuget est automatiquement déclenchée pour mettre à jour correctement les dépendances. Au niveau de la publication, deux options sont disponibles dans cette preview : Déploiement vers Azure

► Publication dans le système de fichier

► La publication locale va jusqu'à générer un fichier **web.cmd** que l'on n'a plus qu'à exécuter pour héberger le site localement, dans la console Windows.

Conclusion

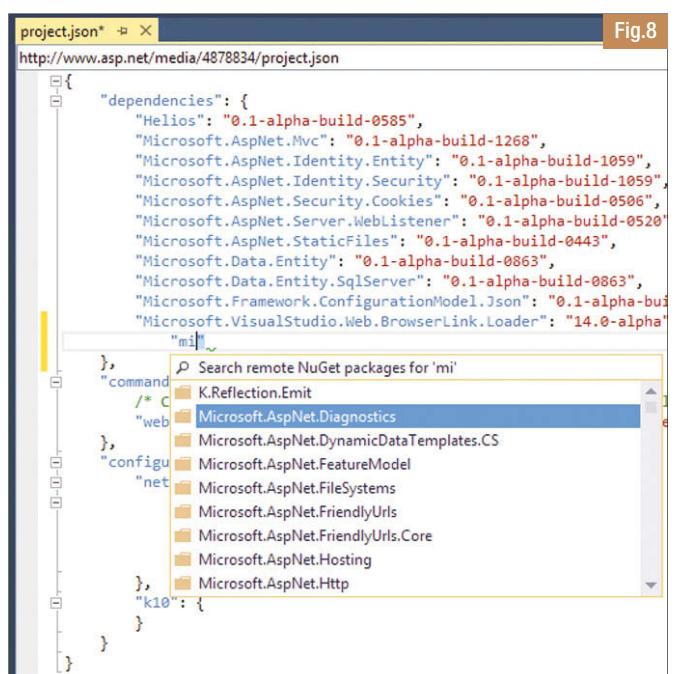
Cette première preview de Visual Studio 14 montre un énorme travail d'amélioration pour certains aspects et une grosse refonte pour d'autres. L'outil est maintenant basé sur un compilateur open source laissant plus de place à la personnalisation. Le futur d'ASP.NET montre une rationalisation et une volonté d'aller vers un modèle plus simple à appréhender, plus passe-partout et plus générique.

► Etienne Margraff

ALM Lead – Infinite Square



Fig.8



Ecoconception logicielle

la chasse aux obésiciels est ouverte !

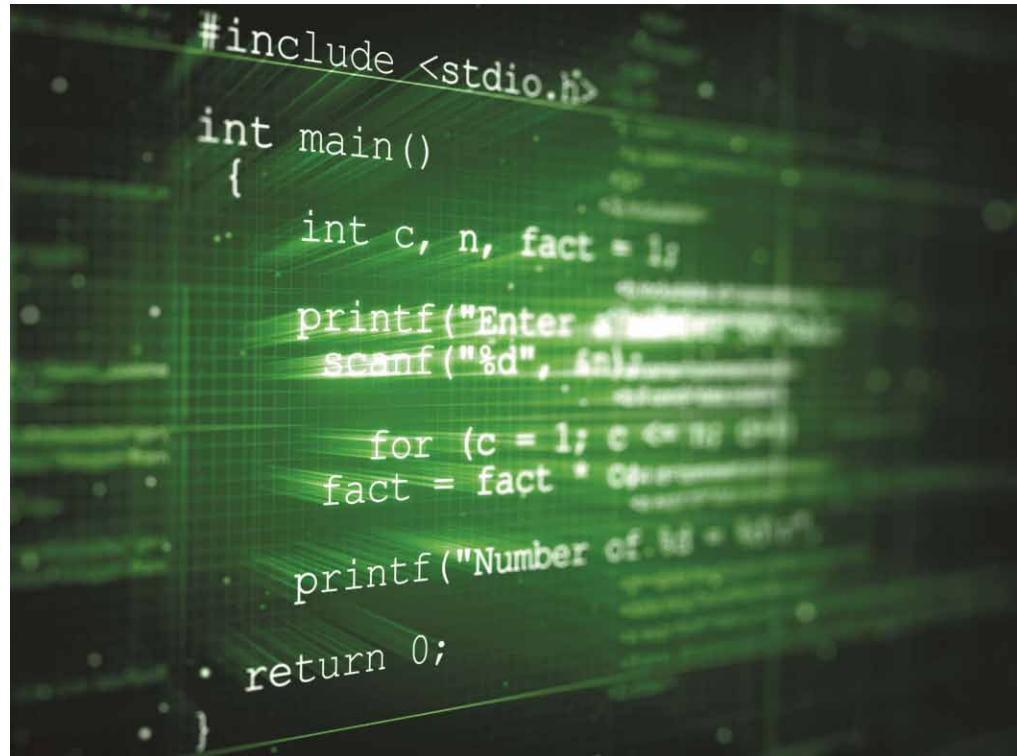
1^{re} partie

De plus en plus d'éditeurs et d'entreprises réagissent enfin face aux surcoûts engendrés par le « gras numérique ». Le mot d'ordre est au régime.

Il est un temps, pas si lointain, où chaque ligne de code était bichonnée par son développeur, optimisée à souhait et vérifiée sous toutes les coutures. Une évidence puisque les ressources informatiques - cycles processeur, octets en mémoire vive, espace disque, puissance de la carte graphique, etc. - étaient comptées. Le développement était alors un sport de haut niveau qui voyait s'affronter les meilleurs athlètes mondiaux dans des compétitions informelles à grand renfort de « démos » dans la catégorie moins de 64 Ko de mémoire vive.

Je ne vous parle pas de ce temps par nostalgie. Non, il s'agit plutôt d'une nécessité. Après avoir vécu dans l'opulence ces 15 dernières années, les informaticiens (re)découvrent à nouveau la finitude des ressources informatiques. C'est qu'entre temps, une crise économique mondiale sans précédent et plusieurs crises écologiques sont passées par là. Bref, la fête est finie et les enfants gâtés de l'informatique doivent revenir rapidement à la raison, sous peine de changer de métier.

L'heure est à l'efficience, c'est-à-dire aux économies de moyens. Après la course aux mégahertz (MHz), les fabricants de microprocesseurs se livrent une bataille acharnée sur le terrain du Thermal Design Power (TDP). La puce qui remporte la mise est celle qui dépense le moins d'énergie pour effectuer un traitement donné. En ligne de mire, l'autonomie des terminaux mobiles connectés (smartphone, tablette, portables, etc.), la facture électrique des data centers et des parcs de postes de travail. Les fondeurs sont plutôt bons dans ce domaine. Ils ont divisé par 40 la consommation électrique de leurs puces depuis 1946, pour un même nombre de traitements. La courbe suit d'ailleurs celle de la loi de Moore. Malheureusement, depuis plus de 20 ans, les logiciels suivent la tendance inverse. Edictée en 1991, la loi de Wirth indique que « *le logiciel ralentit plus vite que le matériel n'accélère* ». Autrement dit, les



istock : © adempercem

“ Grâce à l'application de bonnes pratiques simples, les développeurs du monde entier peuvent réduire bien plus efficacement les impacts environnementaux de l'informatique et du web que Greenpeace et ses millions de supporters. ”

fabricants ont beau proposer des équipements toujours plus puissants tout en étant moins énergivores, ces progrès sont annihilés par la lourdeur croissante des logiciels.

Le coût du « gras » numérique devient exorbitant

« *L'empreinte ressources (cycles CPU, quantité de mémoire vive, bande passante, etc.) d'un logiciel double en moyenne tous les 2 à 3 ans* » confirme une étude menée en 2010 par GreenIT.fr sur des logiciels tels que Windows, Mac OS X, Adobe Reader et la suite Office de Microsoft. Par exemple, il faut 114 fois plus de mémoire vive entre les couples Windows 98 - Office 97 et Windows 8 - Office 2013 pour écrire un texte, effectuer un calcul ou envoyer un e-mail. Ce phénomène n'est pas propre à Microsoft : il touche tous les éditeurs et tous les

développements sur mesure. Or, ce n'est pas parce que l'on travaille avec un logiciel dernier cri que l'on écrit un texte 114 fois plus intelligent 114 fois plus vite...

Pour gagner en portabilité et en productivité, l'industrie du logiciel a empilé les couches d'abstraction et les frameworks. Elle a aussi connecté entre eux des composants génériques qui embarquent souvent plus de fonctionnalités que nécessaire. Ce caractère générique et portable se paie au prix fort : l'inflation des lignes de code se traduit par une dette applicative galopante. Le cabinet d'analyse Gartner estime ainsi que la dette technique mondiale doublera en 5 ans, passant de 500 milliards de dollars en 2010 à 1 000 milliards de dollars 2015.

Dans une étude menée en 2011 sur 365 millions de lignes de code, Cast Software explique en partie ce doublement de la dette technique par le

coût de maintenance exorbitant des dernières générations de langages. Ainsi, maintenir une application écrite en Java coûte 4,3 fois plus cher (5,42 dollars) que maintenir une application écrite en Cobol (1,26 \$). La deuxième loi de la thermodynamique - « Rien ne se perd, rien ne se crée, tout se transforme » - se vérifie parfaitement en programmation : les gains de productivité lors du développement sont complètement absorbés par le coût de la dette technique. CQFD.

Un nombre croissant de responsables informatiques se mettent à raisonner en coût complet et plus seulement en coût de licence. Dès que l'on prend en compte le coût d'infrastructure - investissement initial, coût de fonctionnement, coûts de renouvellement, etc. - et la dette technique, il devient alors évident que le coût de développement ou d'acquisition du logiciel n'est pas significatif. D'où l'intérêt d'investir à nouveau une partie du budget dans une conception de qualité, apte à réduire le coût de fonctionnement sur la durée et, pourquoi pas, les impacts environnementaux associés au logiciel.

Une démarche d'efficience

Ces différentes prises de conscience se traduisent aujourd'hui par une démarche d'écoconception logicielle qui s'applique à toutes les technologies et toutes les architectures logicielles, de la programmation temps réelle, à l'embarqué, en passant par le web, les langages interprétés et compilés, le code SQL, les feuilles de styles, etc. Bref tout ce qui a un ADN composé de 0 et de 1.

L'objectif numéro un de l'écoconception logicielle est de réduire l'empreinte ressources de l'application. Cette dernière est constituée par l'infrastructure (serveurs, réseau, ordinateurs, etc.) et les flux physiques (kWh électriques, bande passante, etc.) associés. « *Plus vous réduisez la quantité de mémoire vive, bande passante, cycles CPU, nombre de serveurs, etc. et plus vous réduisez, mécaniquement, le coût complet de l'application et son empreinte écologique. C'est un tiercé gagnant : empreinte ressources, coût complet, impacts environnementaux* » explique Jérémie Chatard, directeur technique de Breek. Il a notamment participé à la création d'un référentiel de bonnes pratiques d'écoconception web qui a donné naissance au livre « Eco-conception web : les 100 bonnes pratiques » paru chez Eyrolles.

Dans cette rencontre entre le développement durable et le logiciel, on ne se limite pas seulement à la réduction des impacts environnementaux. « *il s'agit également de prendre en compte la dimension sociale, notamment l'accessibilité* » rappelle de son côté Philippe Balin, expert accessibilité au sein de

l'entreprise Solcap21. Le duo social – environnement fonctionne bien d'un point de vue économique. Le coût de l'accessibilité est compensé par les économies générées par l'écoconception logicielle.

Des gains « monstrueux »

Si la préservation de notre belle planète et la responsabilité sociale et sociétale sont louables, aucune entreprise n'est philanthrope. Ce sont donc les économies financières qui les décident à mettre en œuvre cette démarche. Les gains démontrés sur les premiers retours d'expérience sont tellement « monstrueux » qu'il n'est pas difficile de les convaincre.

Il y a 3 ans, Facebook a par exemple investi quelques centaines de jours. homme dans le développement d'un compilateur PHP vers C++. Disponible aujourd'hui sous licence open source, HHVM a permis au réseau social de diviser par 2 le nombre de serveurs nécessaires à la génération des milliards de pages servies chaque mois. Facebook a ainsi évité la construction d'un nouveau data center (quelques centaines de millions de dollars) et récupéré près de 50 % de la capacité dans ses fermes de serveurs web ! Sans compter une facture électrique divisée par 2 et des émissions de gaz à effet de serre réduites d'autant. Au final, les économies se montent à plusieurs milliards de dollars sur 4 ans ! Aussi impressionnant soit-il, cet exemple fait piètre figure comparé aux gains obtenus par IBM et LinkedIn. IBM en premier a démontré que l'optimisation d'un algorithme destiné à la manipulation de données de type Big Data pouvait diviser par 100 le temps de traitement des données tout en ne nécessitant que 1 % de l'énergie initialement consommée, soit quelques centaines de kWh au lieu de plusieurs MWh. Cela signifie qu'à infrastructure (serveurs, etc.) identique, IBM peut désormais réaliser 100 fois plus de traitements à la même vitesse, sans avoir ajouté un seul serveur ! De son côté, LinkedIn s'est aperçu qu'en remplaçant un proxy écrit initialement en Ruby On Rails par un développement basé sur Node.js, il pouvait diviser le nombre de serveurs nécessaire par... 112. Les retours d'expérience de ce type se comptent désormais par dizaines, et pas qu'aux Etats-Unis. De grandes entreprises françaises s'intéressent aussi à cette démarche avec des premiers résultats très encourageant.

Ne pas sous-estimer l'importance du fonctionnel

L'écoconception logicielle est une démarche d'amélioration continue qui s'applique à toutes les étapes du cycle de vie d'un logiciel. Les phases amont – expression du besoin, conception – sont aussi importantes que l'étape de la

Témoignage



« Travailler sur le fonctionnel, en amont du code, est indispensable »

« *Un travail en amont du code est indispensable. En effet, vous aurez beau optimiser le code d'une fonctionnalité pour qu'elle consomme le moins de ressources possible, si cette fonctionnalité n'est pas utilisée, elle consomme toujours trop de mémoire et de cycles CPU. La quantification précise du besoin et une forme de « sobriété » en termes de couverture fonctionnelle sont donc indispensables.* »

Jérémie Chatard,
directeur technique de Breek

programmation, voir plus. En effet, vous aurez beau optimiser le code d'une fonctionnalité pour qu'elle consomme le moins de ressources possible, si cette fonctionnalité n'est pas utilisée, elle consomme toujours trop de mémoire et de cycles CPU. La quantification précise du besoin et une forme de « sobriété » en termes de couverture fonctionnelle sont donc indispensables. De nombreuses études et retours d'expérience le démontrent. Concernant la couverture fonctionnelle, une étude du Standish Group indique que 45 % des fonctionnalités demandées par les utilisateurs ne sont jamais utilisées. Et ce chiffre monte à 70 % si on intègre les fonctionnalités rarement utilisées. Dit autrement, 70 % des exigences fonctionnelles devraient être éliminées lors de leur recueil et de leur formalisation. Ce constat vaut également à l'échelle d'un logiciel tout entier. Plusieurs enquêtes estiment que de 25 % à 64 % des logiciels déployés par les entreprises ne sont jamais utilisés. Cette gabegie se traduit par un surcoût de licences de 100 000 dollars par an en moyenne par entreprise et 27 milliards de dollars à l'échelle de la planète. Sans compter les centaines de milliers de serveurs zombies qui exécutent ce code inutilement. On peut penser par exemple au lecteur PDF d'Adobe tellement lourd qu'il est régulièrement remplacé par un Foxit Reader ou ses acolytes. L'entreprise continue cependant à payer les licences et l'ordinateur à

exécuter le logiciel, ce qui réduit d'autant les ressources disponibles. Concernant la quantification précise des besoins, Microsoft Research a démontré il y a quelques années que le fait d'afficher 10 % de résultats en moins sur le moteur de recherche Bing économisait 65 % de consommation électrique côté serveur. Avec 20 % de résultats affichés en moins, on obtient 80 % d'économie d'énergie côté serveur. Autrement dit, il est primordial de quantifier au plus juste le contenu des logiciels : nombre de résultats retournés par une requête SQL et affichés à l'écran, qualité et taux de compression des images, temps de réponse attendu, etc.

Une mise en œuvre la plus légère possible

La mise en œuvre de l'écoconception logicielle est simple. Il suffit d'appliquer un ensemble de bonnes pratiques spécifiques. Certaines de ces bonnes pratiques sont regroupées dans des référentiels, comme celles du web dans le livre « Eco-conception web : les 100 bonnes pratiques » paru chez Eyrolles. Une fois que l'on fait clairement la différence entre une démarche de performance et une démarche de sobriété et d'efficience, la plupart des bonnes pratiques tombent sous le sens. D'ailleurs, bon nombre de développeurs qui ont connu les débuts de l'informatique les appliquent déjà, du moins, quand on leur en laisse le temps. Il s'agit autant d'éviter la sur-qualité (couverture fonctionnelle, temps de réponse, etc.) que de compresser les images d'un site web. Deux bonnes pratiques évidentes mais que personne n'applique jamais... A moins de représenter un gouffre financier ou d'être face à un grave problème de performance

<les résultats du Green Challenge>

Green Challenge : empreinte ressources divisée par 6 en quelques jours

En juin 2010, Octo Technology et Greenit.fr organisaient le Green Challenge. Objectif : réduire l'empreinte ressources d'un logiciel développé en Java côté serveur (Google App Engine) et XHTML côté client. En quelques jours, hommes, les 3 équipes gagnantes ont réussi à diviser par 6 l'empreinte ressources du logiciel (caractérisée par la consommation électrique du processeur et de la mémoire vive). Leurs optimisations ? Déplacer un maximum de traitement côté serveur, éviter la sur-qualité (carte des Etats-Unis statique plutôt que dynamique), choisir les composants logiciels les plus performants (benchmark des bibliothèques Java), et optimiser le code HTML. Détail : <http://www.greenit.fr/article/bonnes-pratiques/developper-green-les-enseignements-du-green-challenge-3018>

Glossaire

- **Empreinte ressource** : exprime la quantité de cycles processeur, de mémoire vive, et d'espace de stockage nécessaire au fonctionnement du logiciel. Plus cette quantité est faible et plus le logiciel est efficient. Plus le logiciel est efficient et plus ses impacts environnementaux sont réduits.
- **Obésiciel** : contraction de « obèse » et « logiciel ». Le terme anglais est « bloatware ».

et / ou de montée en charge, il ne faut pas viser les projets existants en priorité. Qu'il s'agisse d'un site web ou d'une application client-serveur traditionnelle existante, « l'optimisation de leur niveau d'écoconception coûtera cher » explique Jérémie Chatard. Mieux vaut donc viser les nouveaux développements en intégrant la démarche dès le début du projet. Il s'agit alors de recentrer les demandes des utilisateurs sur l'essentiel (par exemple en valorisant financièrement chaque fonctionnalité demandée pour les aider à arbitrer), de quantifier le besoin au plus juste lors de l'écriture des spécifications, etc. Bien entendu, le choix d'une technologie et d'une architecture adaptées est primordial. On peut par exemple préférer Percona Server à MySQL, PressFlow à Drupal, etc.

Pour le développement, s'il est effectué par un prestataire, il suffit d'ajouter les bonnes pratiques à appliquer dans les exigences du cahier des charges techniques.

Toute l'intelligence de la démarche consiste à sélectionner uniquement les bonnes pratiques clés - celles qui ont la plus grosse capacité à réduire l'empreinte ressources – pour ne pas alourdir inutilement le travail des développeurs. Sinon, gare au retour de bâton ! Des audits réguliers, plus ou moins profonds, permettent de vérifier la bonne application des bonnes pratiques tout au long du projet.

● Françoise Brindille

Pour aller plus loin

« Eco-conception web : les 100 bonnes pratiques », Eyrolles, 128p, 12 euros, ISBN : 978-2-212-13575-6
www.greenit.fr/tag/eco-conception-logicielle

Restez connecté(e) à l'actualité !

- **L'actu** de Programmez.com : le fil d'info quotidien
- **La newsletter hebdo** : la synthèse des informations indispensables.
- **Agenda** : Tous les salons, barcamp et conférences.

Abonnez-vous, c'est gratuit !

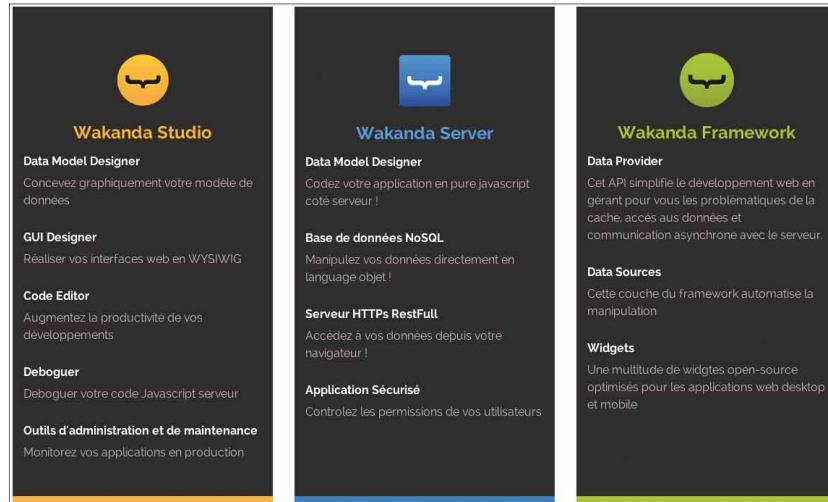
www.programmez.com

The screenshot shows the homepage of Programmez.com. At the top, there's a navigation bar with links for 'Actualités', 'Avis d'experts', 'Livres blancs', 'Boutique', 'Logiciels', 'Tutoriels', 'Barcamp', 'Agenda', and 'Emploi'. The main content area features a large article about using Crosscut to digitize usage data. Below that, there's a 'Page non trouvée' section with a thumbnail of a magazine cover. The sidebar on the right contains links to 'Nouveautés de NI LabVIEW 2013', 'Nouveautés de NI LabVIEW 2012', 'Nokia X', 'Sessions de coaching pour les développeurs à Paris (4 et 5 mars) Lyon (7 mars) et Lille (28 mars)', 'Créez votre 1^{er} serveur cloud gratuitement', and a 'PLUS' section with a gear icon.

Créez et intégrez vos propres widgets à la plateforme Wakanda

Wakanda est une plateforme open-source de développement d'applications web qui fournit des outils et services intégrés de développement, déploiement et gestion de vos applications jusqu'au Cloud. Avec Wakanda, vous créez des applications Web et mobiles respectant les standards du Web : HTML5, CSS3, HTTP/REST, JSON-RPC, et bien sûr JavaScript utilisé aussi bien côté client que côté serveur.

Quelques caractéristiques de la plateforme Wakanda :



Une solution tout-en-un, ouverte sur le monde open-source

Vous pouvez choisir d'utiliser la base de données NoSQL intégrée ou bien connecter une base de données externe telle que MySQL, Microsoft SQL Server ou toute base de données compatible ODBC. Coté client, Wakanda fournit une grande diversité de widgets et grâce aux APIs du framework, vous pouvez développer vos propres widgets et les utiliser dans vos outils de développement de la même façon que les widgets natifs Wakanda. Vous pouvez ainsi profiter des frameworks JavaScript les plus avancés comme KendoUI, Sencha, Select2 JS ou jQuery pour créer vos widgets personnalisés. C'est ce que nous allons voir dans cet article.

PRÉSENTATION DE L'EXEMPLE

Cet article illustre la création d'un custom widget Wakanda basé sur le widget JavaScript externe Signature_Pad. Il s'agit d'un plug-in jQuery permettant d'apposer une signature 'manuelle' sur une page web. Les différentes étapes nécessaires pour créer et intégrer ce widget sont décrites en pas à pas. Pour des informations plus détaillées, n'hésitez pas à vous référer à la documentation de Wakanda (<http://doc.wakanda.org/Wakanda/help/Title/en/page3863.html>).

Démarrer

Pour créer un custom widget, des connaissances de base en JavaScript, CSS et HTML5 sont conseillées.



Pour le widget que nous allons créer, nous utiliserons :

► **OS:** Windows 7 64bit avec 4Go de RAM (Ce tutoriel peut également être réalisé sur Macintosh).

► **Wakanda:** Vous pouvez le télécharger ici :

<http://wakanda.org/downloads>

Après avoir téléchargé et décompressé Wakanda, ouvrez le dossier "Wakanda Studio" et lancez Wakanda Studio en double-cliquant sur *Wakanda Studio.exe*.

1. Créer la solution

Dans l'ongle *Home* de Wakanda Studio, cliquez sur **Create a New Solution** : Fig.1.

Donnez un nom à la solution, par exemple "My_Custom_Widget". Assurez-vous que l'option 'Add a blank project to the new solution', pour créer un projet vide dans la solution, est cochée.

En bas de la page du *Solution Explorer*, vous pouvez voir le dossier **WIDGETS**, destiné à contenir les *custom* widgets installés dans Wakanda Studio : Fig.2.

2. Créer le *custom* widget

Selectionnez **New->Custom Widget** dans le menu **File** ou sélectionnez **New Custom Widget** dans le menu contextuel du dossier **WIDGETS**.

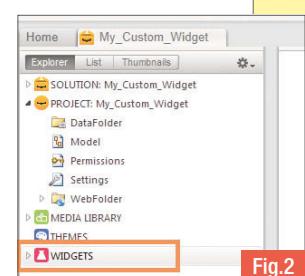
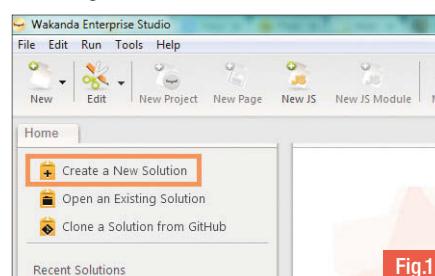
Saisissez un nom pour le *custom* widget, par exemple "My_Widget", et cliquez sur le bouton **Create**.

Un dossier est alors créé dans le dossier **WIDGETS** : Fig.3.

Le dossier créé pour votre widget inclut par défaut les éléments suivants :

► **css** : Dossier contenant le fichier CSS qui définit le style par défaut du widget.

► **designer.js** : Fichier JavaScript vous permettant de définir l'apparence du widget dans Wakanda Studio.



- ▶ **Icons** : Dossier contenant l'icône du widget à afficher dans le *GUI Designer*.
 - ▶ **package.json** : Dans ce fichier JSON, vous définissez les meta data du widget ainsi que les références à tous les fichiers JavaScript et CSS qui doivent être chargés.
 - ▶ **README.md** : Fichier "read me" de votre *custom widget*, que vous pouvez notamment utiliser dans le dépôt GitHub.
 - ▶ **widget.js** : Fichier JavaScript contenant le code de votre *custom widget*, vous permettant de définir ses propriétés, méthodes et événements.
- Pour plus d'informations sur la structure des *custom widgets*, veuillez vous reporter à cette documentation (<http://doc.wakanda.org/Wakanda/help/Title/en/page3863.html>).

Notre *custom widget* est basé sur un plug-in jQuery nommé **[signature_pad]** http://szimek.github.io/signature_pad/. Ce plug-in vous permet de signer une page web (Smartphone, Tablette ou Desktop) comme s'il s'agissait d'une feuille de papier.

En premier lieu, téléchargez la dernière version réduite de ce plug-in jQuery, **signature_pad.min.js**, depuis son dépôt github (https://github.com/szimek/signature_pad).

Créez ensuite un dossier "js" dans le dossier du widget et placez le fichier "signature_pad.min.js" dans ce dossier **Fig.4**.

Ajoutez cette dépendance dans le fichier "package.json" du *custom widget* : **Fig.5**.

3. Créer des propriétés et des méthodes pour le *custom widget*.

Un *custom widget* hérite directement de l'objet *Widget*, qui propose une API commune à tous les widgets de Wakanda. Le fichier "widget.js" contient le code nécessaire pour créer le widget.

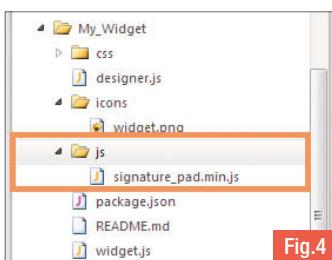
L'intégration d'un widget provenant d'un autre framework nécessite d'analyser les propriétés et méthodes à encapsuler. Nous allons ajouter les propriétés définies dans la documentation de l'API du plug-in JQuery (https://github.com/szimek/signature_pad). Ce plug-in jQuery accepte six options: *dotSize*, *minWidth*, *maxWidth*, *backroundColor*, *penColor* et *velocityFilterWeight*.

Les commentaires dans le code vous guident pour la définition des propriétés et des événements du widget.

Nous allons commencer par ajouter du code pour initialiser le DOM du widget via la fonction **init()** :

[js]

```
init: function() {
```



```
var $node = $(this.node);
// Ajout d'un élément DOM canvas au widget
this.node.innerHTML = '<canvas></canvas>';
// Modification des attributs du canvas à l'aide de jQuery
$node.find(<canvas>).attr('id', 'canvas_' + this.node.id),
    .attr('width', this.node.clientWidth),
    .attr('height', this.node.clientHeight);

if (!window.Designer) {
    var that = this;
    this.signaturePad = new SignaturePad($node.find('canvas')[0], {

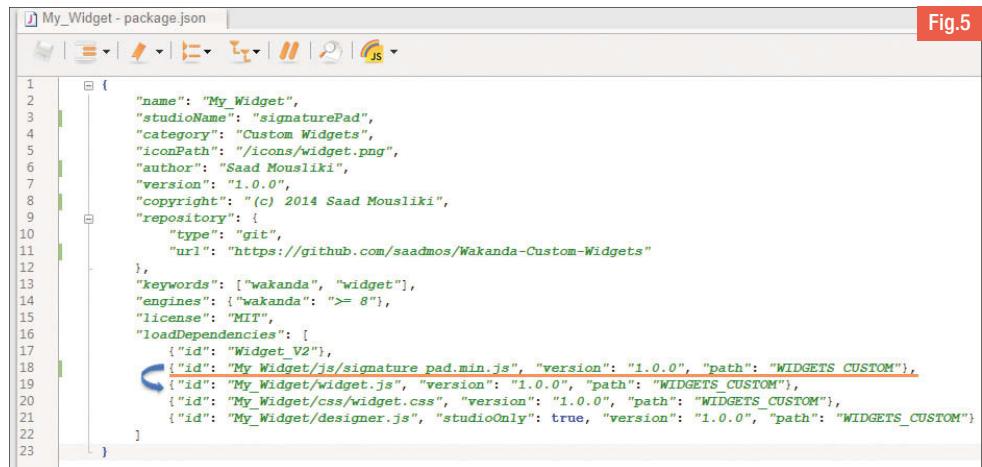
        minWidth: this.minWidth(),
        maxWidth: this.maxWidth(),
        penColor: this.penColor(),
        dotSize: this.dotSize(),
        backgroundColor: this.backgroundColor(),
        velocityFilterWeight: this.velocityFilterWeight(),
        // Ajout d'événements à l'aide de méthode fire()
        // disponible dans l'API des custom widgets
        onBegin: function() {
            that.fire('begin');
        },
        onEnd: function() {
            that.fire('end');
        }
    });
}
```

[/js]

Le code de la fonction **init()** est en charge de la création de l'élément DOM du *custom widget* lorsque vous ajoutez une instance dans votre Page. Dans ce code, nous commençons par ajouter l'élément canvas au DOM de notre *custom widget*, ce qui est nécessaire pour instancier le *custom widget*, comme décrit dans la documentation de l'API du plug-in JQuery (https://github.com/szimek/signature_pad). La documentation de SignaturePad indique également que nous devons d'abord créer une zone de dessin puis passer le canvas HTML5 déjà créé en tant que premier argument à la fonction **SignaturePad()**. Nous devons également passer les autres propriétés, comme *minWidth*, *maxWidth* et *penColor* dans le second argument de cette fonction.

L'ajout des événements **onBegin** et **onEnd** à l'aide de la [fonction **fire()**] est détaillé dans la suite du tutoriel.

Fig.5



Note : Dans le code de notre *custom widget*, nous utilisons l'API jQuery, qui est chargée par défaut dans WAF (<http://doc.wakanda.org/WakandaStudio/help/Title/en/page2136.html>). Pour sélectionner le widget à l'aide de jQuery, vous pouvez utiliser `$(this.node)`, qui équivaut à `document.getElementById(my_Widget_ID)`. Après avoir ajouté des événements à notre *custom widget*, nous pouvons configurer les propriétés qui seront visibles dans la zone **Properties** de Wakanda Studio. Nous pouvons ajouter ces propriétés à notre *custom widget* via le code suivant :

```
[js]
dotSize: widget.property({
  type: 'number',
  defaultValue: 0.2,
  bindable: false
}),
minWidth: widget.property({
  type: 'number',
  defaultValue: 0.5,
  bindable: false
}),
maxWidth: widget.property({
  type: 'number',
  defaultValue: 2.5,
  bindable: false
}),
backgroundColor: widget.property({
  type: 'string',
  defaultValue: '#fff',
  bindable: false
}),
penColor: widget.property({
  type: 'string',
  defaultValue: 'black',
  bindable: false
}),
velocityFilterWeight: widget.property({
  type: 'number',
  defaultValue: 0.7,
  bindable: false
})
})
```

Pour plus d'informations sur l'ajout de propriétés dans les *custom widgets*, reportez-vous à la documentation Wakanda (<http://doc.wakanda.org/Wakanda/help/Title/en/page3871.html>). Vous pourrez modifier les valeurs des propriétés du *custom widget* dans Wakanda Studio une fois que vous l'aurez ajouté dans une Page : **Fig.6**. A ce stade, vous pouvez ouvrir une Page pour vérifier que les propriétés et les événements que vous avez ajoutés correspondent à vos attentes.

Astuces :

► Utilisez la fonction `render()` pour mettre à jour votre *custom widget* à

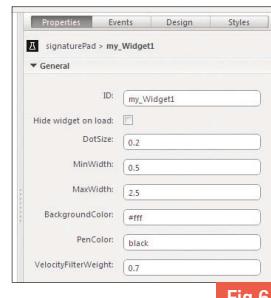


Fig.6

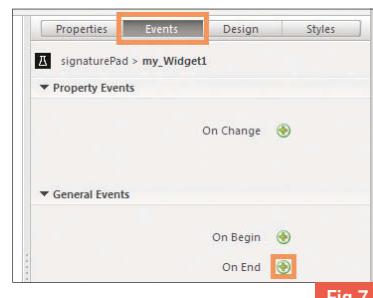


Fig.7

chaque fois qu'une des propriétés du widget est mise à jour par l'utilisateur. L'événement `onChange()` de la propriété est appelé et la fonction `render()` va appliquer la nouvelle valeur de la propriété.

- Si vous voulez adapter le comportement de votre widget selon qu'il est utilisé dans Wakanda Studio ou dans un navigateur (au runtime), vous pouvez utiliser `window.Designer`, qui retourne `true` si vous êtes dans Wakanda Studio et `false` dans le cas contraire. Dans notre version du *custom widget*, nous avons utilisé cette astuce pour ne pas permettre à l'utilisateur d'effectuer une signature dans Wakanda Studio.
- Si vous utilisez dans votre *custom widget* une version de jQuery plus récente que celle de Wakanda, vous pouvez inclure le code suivant dans votre fichier "widget.js" :

```
[js]  var $ = JQuery.noConflict(true); [/js]
```

- Voici un exemple de fonctions *getter* et *setter* d'une propriété :

```
[js]
var propertyName = this.propertyName();
/* lire la valeur de propriété du widget */
this.propertyName(newValue);
/* mettre à jour propertyName avec la valeur newValue */
[/js]
```

- Si vous avez des propriétés et des événements qui sont utilisés uniquement dans le code interne de votre widget et que vous ne voulez pas retourner dans l'objet widget, il peut être utile d'appliquer les conseils de JavaScript Module Pattern (<http://addyosmani.com/resources/essentialjsdesignpatterns/book/>) pour les masquer.
- Vous pouvez utiliser l'événement `onChange()`, qui est généré lorsque la valeur de propriété d'un widget est modifiée au runtime, pour mettre à jour votre widget personnalisé. Pour plus d'informations sur cet événement, reportez-vous à la documentation Wakanda (<http://doc.wakanda.org/Wakanda/help/Command/en/page901009.html>).

4. Définir les événements du *custom widget*.

Dans la fonction `init()`, nous avons défini deux événements pour notre *custom widget*: `onBegin` et `onEnd`. L'événement `onBegin` est généré lorsque l'utilisateur commence à écrire sa signature et l'événement `onEnd` lorsqu'il termine sa saisie.

Pour ajouter un événement à un *custom widget*, vous utilisez la fonction `fire()` (<http://doc.wakanda.org/Wakanda/help/Command/en/page900975.html>). Pour plus de détails sur ces deux événements, veuillez vous référer à la documentation de l'API du plug-in JQuery (https://github.com/szimek/signature_pad).

Vous devez également ajouter les événements personnalisés dans le fichier `designer.js` afin qu'ils apparaissent dans Wakanda Studio :

```
[js]
/* Définition des événements personnalisés pour votre widget */
My_Widget.addEvent('begin');
My_Widget.addEvent('end');
[/js]
```

Vous pouvez alors visualiser ces événements pour chaque instance du *custom widget* dans la page Events. L'utilisateur pourra ajouter facilement le code à exécuter pour chaque événement en cliquant sur le bouton "+" : **Fig.7**. En cliquant sur le bouton +, nous pouvons ajouter le code pour l'évé-

nement On End afin que l'utilisateur soit averti que sa signature a été enregistrée :

```
[js]
my_Widget1.end = function my_Widget1_end (event)
{
  alert(<La signature a été stockée dans un fichier image !>);

}
[/js]
```

Dans le fichier *designer.js*, nous ajoutons également les fonctions *clear()*, *isEmpty()* et *toDataURL()* à notre *custom* widget :

```
[js]
clear: function() {
  this.signaturePad.clear();
},
isEmpty: function() {
  return this.signaturePad.isEmpty();
},
/* Cette méthode convertit le canvas SignaturePad en image
pouvant être ouverte dans un navigateur */
toDataURL : function() {
  return this.signaturePad.toDataURL();
}
[/js]
```

5. Prévisualiser et déboguer votre *custom* widget.

Pour prévisualiser votre *custom* widget, il vous suffit de l'insérer dans votre Page par simple glisser/déposer, comme pour tout widget : **Fig.8.** Ouvrez ensuite la page dans votre navigateur pour tester l'utilisation de votre *custom* widget.

6. Partager votre widget.

Vous pouvez partager votre *custom* widget en le plaçant dans le dépôt GitHub des *custom* widgets de Wakanda. Pour cela, suivez simplement les étapes décrites dans ce guide (<http://forum.wakanda.org/showthread.php?6307-Kick-off-Widgets-and-Wakanda-8>).

Dans la version en développement de Wakanda (<http://download.wakanda.org/DevChannel/Main/>), nous avons inclus la fonction Add-Ons Extension (<http://doc.wakanda.org/WakandaStudio/help/Title/en/page4263.html>), qui vous permet d'installer des thèmes, widgets, modules et extensions Wakanda Studio personnalisés : **Fig.9.**

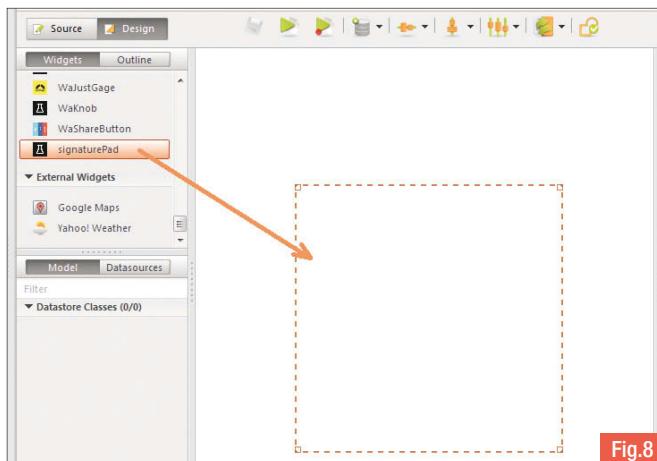


Fig.8

Pour plus d'informations à propos de l'installation d'un *custom* widget à l'aide de l'extension Add-ons, reportez-vous à cette discussion (<http://forum.wakanda.org/showthread.php?6306-Kick-off-Widgets-and-Wakanda-8-The-Add-ons-Extension>).

Conclusion

Si vous avez suivi ce tutoriel, vous avez obtenu un widget personnalisé qui intègre un plug-in jQuery externe. Ce widget peut répondre à divers besoins, comme de stocker la signature d'un utilisateur dans une base distante (http://forum.wakanda.org/showthread.php?6543-Custom-Widget-signature_pad-sign-on-web-page-like-signing-on-paper) ou être utilisé comme application mobile pen du Samsung Note S pour écrire ou dessiner en utilisant différentes couleurs et tailles de stylos.

La communauté JS crée en permanence des bibliothèques surprenantes vous permettant de créer des graphes, sélectionner des cases, plans, grilles, pages, menus, etc. La nouvelle architecture de Widgets de Wakanda vous permet de tirer facilement parti de cette richesse en transformant ces librairies externes en widgets Wakanda. Comme le montre notre exemple, vous pouvez transformer une bibliothèque utile en widget Wakanda que vous pouvez inclure dans tous vos projets Wakanda. Pour étendre les possibilités des bibliothèques externes originales, vous pouvez ajouter vos propres propriétés qui peuvent être liées aux données provenant du datastore.

RESOURCES

“Widgets v2 Creating a Custom Widget”

(<http://doc.wakanda.org/Wakanda/help/Title/en/page3849.html>)

Dépôt GitHub des widgets personnalisés de Wakanda

(<https://github.com/Wakanda-Packages/wakanda-widgets>)

Dépôt GitHub de la version finale du plug-in jQuery SignaturePad :

(<https://github.com/saadmos/SignaturePad>)

Une solution Wakanda qui utilise le widget personnalisé “SignaturePad” pour stocker les signatures sous forme d'images dans le datastore de la solution Wakanda solution :

(https://github.com/saadmos/E-Signature_To_Image_Saver)



Ingénieur en informatique et télécom. Il a plusieurs années d'expérience dans le développement web (html 5, JS, Cloud, mobile, ASP.Net).

(<http://www.sitepoint.com/author/smousliki/>)

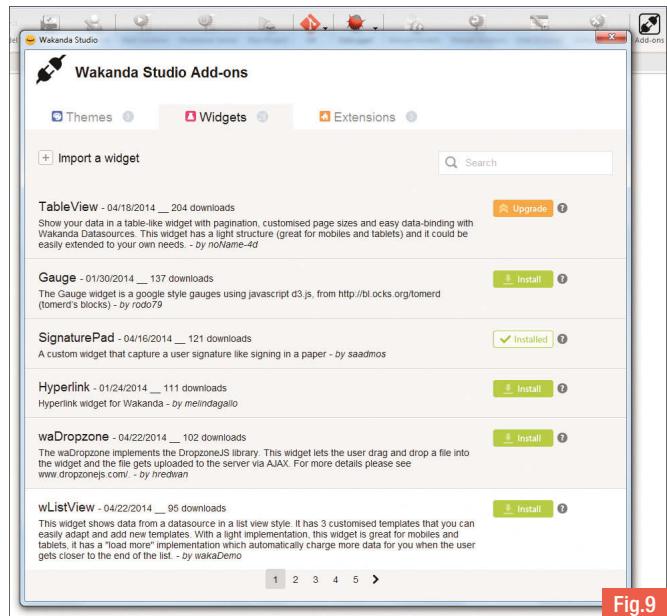


Fig.9

“Avec un simple éditeur et un compilateur, on peut faire des miracles !”



Ayoub Zamouchi a 23 ans. Il a passé avec succès un master en génie des systèmes informatiques. Passionné de nouvelles technologies, il s'intéresse beaucoup aux sports mécaniques et au tennis.

Comment es-tu tombé dans l'informatique et le développement ?

Ta famille, tes amis t'ont soutenu ?

Ma passion pour l'informatique me vient directement de mon père. J'ai eu l'occasion de toucher à un ordinateur dès mon plus jeune âge; notre premier ordinateur était équipé de windows 95, mon père l'utilisait pour saisir des mémoires de thèses et d'autres travaux de bureautique.

A cette époque là en Algérie, les ordinateurs personnels ne se vendaient pas facilement comme c'est le cas aujourd'hui et le fait d'en posséder un à la maison, c'était vraiment génial.

Au cours des années 2000, il y avait les cybers-cafés qui commençaient à ouvrir un peu partout en Algérie; c'était là où je passais la plus grande part de mon temps, je basculais entre les différents sites web, messageries instantanées, forum, etc. Petit à petit j'ai commencé à me poser des questions sur le fonctionnement des logiciels que j'utilisais: comment les jeux vidéos sont-ils réalisés? Comment est-il possible qu'on puisse se connecter à son compte mail à partir de plusieurs ordinateurs et y retrouver les mêmes mails? Où sont stockées toutes ces données? C'était ce genre de questions qui m'a poussé à poursuivre des études en informatique après avoir obtenu mon baccalauréat.

Qu'est-ce que tu aimes dans la programmation : le développement ? la technique ? Réussir à concevoir des applications ? Les défis ?

Ce que j'aime vraiment dans la programmation c'est qu'elle ne

demande pas grand chose : un simple éditeur de texte, un compilateur et on peut faire des miracles. Pour moi chaque nouveau programme ou chaque nouveau projet est un défi qu'il faut relever. Il y a toujours des points qu'on ne maîtrise pas bien; soit on n'est pas habitué au langage, soit on n'est pas habitué à la méthodologie de programmation, etc. Pour moi c'est assez fréquent ce genre de problèmes. Du coup il faut faire face à tout ça. Il faut se débrouiller pour apprendre et réaliser ce qui a été demandé. Tout en respectant les délais.

Pour la technique, je considère cela comme un point très important. A mon avis ce n'est pas une chose qu'on apprend du jour au lendemain, ça m'arrive pas souvent de tomber sur un cas où mes classes n'ont pas été bien conçues. Ou parfois, j'ai créé des méthodes et je n'arrive pas à les généraliser. Du coup je suis obligé de faire du bricolage dans mon code. C'est vrai que ça m'agace de temps en temps mais il faut bien passer par ces étapes sinon on n'apprendra jamais.

Tu as été un des développeurs du nouveau site de Programmez!, est-ce que ce fut un projet normal ? Plus compliqué ? Plus de pressions ?

Ma participation au développement du nouveau site de programmez a vraiment été très bénéfique : travailler avec Drupal au sein d'une équipe avec un autre développeur, un designer et un chef de projet, pratiquer la méthodologie SCRUM, être constamment en relation avec le client. Pour moi, le projet programmez.com a été un cas d'école où j'ai beaucoup appris, comme sur tous les autres

projets.

La pression, oui j'en ai eu pas mal à certaines périodes. Pas forcément de mon boss ou du client mais plutôt des délais et surtout la volonté de maîtriser le planning. Je me suis retrouvé à plusieurs reprises à travailler en pleine nuit et le week-end. Situation que je trouve normale pour n'importe quel développeur. Je fais ça par passion et par amour. Je n'ai jamais senti l'obligation de le faire mais je voulais le faire, comme un défi à relever.

Tu travailles beaucoup avec Drupal, quelle est ton opinion sur l'outil ?

Drupal est un outil très puissant. Son architecture basée sur le principe des Hooks qui sont de simples fonctions invoquées à un moment spécifique du processus de building. Les modules peuvent implémenter ces hooks et apporter leurs modifications au moment nécessaire. Le fait que Drupal soit un projet Open source, on trouve beaucoup de contributions, on peut rapidement dénicher des modules qui facilitent beaucoup de tâches. Sinon Drupal est très bien documenté et il n'est pas compliqué de créer ses propres modules.

L'informatique change vite, très vite. Comment arrives-tu à suivre les évolutions ? À te former ?

Être à jour est une obligation pour chaque développeur. C'est vrai que le domaine informatique évolue d'une manière exponentielle, mais la plupart des nouveaux outils, API, frameworks, etc. sont toujours accompagnés d'une documentation. De plus, on trouve souvent des communautés autour d'une techno, cela m'aide beaucoup pour rester à jour et facilite la

tâche pour quelqu'un qui possède juste la patience de passer un petit moment sur la doc et sur les forums.

Pour se former, il y a aujourd'hui ce qu'on appelle les MOOC (Massive Online Open course), plusieurs plateformes sont aujourd'hui disponibles sur le web, comme edX, OpenClassRooms (le site du zéro), Lynda.com, phpacademy.org, etc. la plupart sont gratuits, d'autres sont payants.

Personnellement je trouve ça plus rapide et plus efficace quand on débute sur une nouvelle technologie que d'utiliser un simple livre; déjà premièrement il faut trouver un livre à jour, les technos évoluent tellement vite que le contenu du livre peut être dépassé au bout d'une année. Mais le fait de voir une vidéo, de voir le code s'écrire, s'exécuter devant ses yeux permet de bien assimiler les choses, surtout si c'est bien expliqué par le formateur et dans la plupart des cas le contenu est rafraîchi fréquemment. Certes il faut s'appliquer pour bien comprendre, d'où le fait que la plupart des plateformes proposent des exercices entre les différents chapitres, des pièces jointes à télécharger, des snippets de codes prêts à s'exécuter et à tester!

Dans ma manière de procéder, je commence toujours par chercher un cours en ligne sur ce que je veux apprendre; puis, je cherche un livre (parfois la doc suffit largement) pour approfondir mes connaissances ou pour me renseigner en cas de besoin.

La vie d'un développeur n'est pas toujours facile : stress, projets, planning serré. Comment concilier le travail, les loisirs, les sorties, la famille ?

A la fin de la semaine, je fais mon maximum pour me changer les idées, faire autre chose. Mais ce n'est pas toujours possible surtout quand les plannings sont serrés, ce qui est souvent le cas. Parfois, je dois coder durant le week-end pour avancer le projet.

Le peu de temps libre que j'ai, je le passe avec des amis. On organise des sorties vers la montagne, parfois la plage. On se met autour d'un barbecue et on passe un bon moment. Ils ont toujours des tas de choses à me raconter. On ne se voit pas beaucoup. Quand on me pose la question « et toi Ayoub, du nouveau ? » Bah je rembobine le film de ma semaine dans ma tête, je me rends compte que j'ai passé 5 jours au bureau, devant mon ordinateur, en train de coder et de résoudre des bugs. Le jour où je trouverai une réponse à cette question, je sera probablement loin du monde des développeurs...

Entre ce que tu as appris en école informatique et le travail « réel », as-tu eu du mal à t'adapter ? Qu'est-ce qui t'a le plus surpris ?

Du mal à m'adapter, non pas du tout ! J'étais ravi de changer d'environnement, de faire la connaissance de nouvelles personnes et de travailler au sein d'une équipe professionnelle.

Une chose qui a vraiment changé entre l'école et le travail c'est que durant ma formation, je ne me préoccupais pas beaucoup de la qualité de mon code.

Dans la plupart des cas il était visualisé soit par d'autres étudiants, soit par un professeur. Leurs remarques et avis à propos de mon code étaient toujours les bienvenus pour moi. Quand j'ai intégré DzMob, on faisait de la prestation. Du coup il

est obligatoire de présenter un code de qualité, donc je faisais de plus en plus attention à la manière de coder, car le fait de recevoir un commentaire du client à propos d'une fonction mal placée ou mal implémentée n'est pas du tout du même niveau que les remontées durant le cursus scolaire. C'est la réputation de l'entreprise qui est en jeu. Cela met plus de pression. J'étais vraiment ravi de découvrir comment les projets sont estimés, comment établir un planning, répartir les différentes tâches sur l'ensemble de l'équipe, etc. On ne peut réellement le pratiquer que dans la "vraie vie".

Où te vois-tu dans 3-5 ans ?
Ce n'est pas facile de le prévoir, si je reviens 5 ans en arrière, quand j'ai commencé mes études en informatique, je n'avais aucune idée de ce que j'allais faire après mes études. Là, maintenant, je me retrouve sur le web et le mobile. C'est en suivant ma passion et mes compétences que je suis arrivé là.

Quels conseils donnerais-tu à des étudiants et aux développeurs qui te lisent ?

Des conseils ! J'en ai toujours besoin moi-même, mais le peu que je puisse dire est que la vraie force d'un développeur c'est sa capacité à s'adapter rapidement, à être autonome, capable d'apprendre de nouvelles choses à n'importe quel moment. Pas la peine de dépenser une fortune pour une formation dans un langage ou un autre, surtout en ce qui concerne le web. On peut trouver beaucoup de choses sur le net. Il suffit d'avoir les notions de base et tout viendra avec l'expérience. Et il n'y a pas mieux que de pratiquer, et surtout d'avoir beaucoup, beaucoup de patience.



“ J'ai eu le coup de cœur durant une séance de programmation HTML ”

L'informatique change vite, très vite. Comment arrives-tu à suivre les évolutions ? A te former ?

Google est mon meilleur ami ! C'est sûr que je travaille dans un domaine très évolutif, donc il faut très régulièrement se remettre en question. Je ne cesse pas de consulter Internet, les forums, de la documentation sur Internet quotidiennement dans le but de faire évoluer ma carrière professionnelle, sans oublier les revues...

La vie d'un développeur n'est pas toujours facile : stress, projets, planning serré. Comment concilier le travail, les loisirs, les sorties, la famille ?

Il est vrai que mon travail m'occupe beaucoup et qu'il empiète largement sur le temps libre et les week-ends. Cela n'est pas dérangeant étant donné que j'aime ce job. Mais il est parfois nécessaire de se couper et se changer les idées : scoutisme, passer du temps avec les enfants, faire des sorties entre amis, découvrir l'Algérie.

Où te vois-tu dans 3-5 ans ?

Question difficile ! Tout d'abord : être expert dans ce que je fais et être spécialisé. Je considère que plus on est spécialisé, et plus on a de chance d'être compétent et expert. Aujourd'hui, mon ambition est de me focaliser sur le mobile.

Quels conseils donnerais-tu à des étudiants et aux développeurs qui te lisent ?

Tout d'abord, je tiens à préciser que je suis fier d'être développeur. Pour moi ce n'est pas une étape vers le poste de chef de projet, il faut être passionné par ce qu'on fait. Il faut beaucoup de patience quand tu te trouves obligé de reprendre plusieurs fois le même problème que tu croyais avoir réglé et qui, en fait, ne l'est pas. A mon avis, je ne pense pas qu'il faut avoir des qualités spécifiques pour faire ce métier. C'est plus une question de compétences.

100% Drupal



Architecture / Installation / Configuration /
Mes premiers projets Drupal /
Découverte de Drupal 8

Pourquoi Drupal ?



Une fois répondu à la question vaut-il mieux coder un site à partir de zéro ou utiliser un CMS, question qui suscite toujours moult débats souvent passionnés, voire houleux, mais qui ne se pose pas vraiment selon moi, arrive la question de savoir quel CMS choisir ?

Assurément choisir un CMS est difficile. Il existe de nombreux

produits, tous avec leurs qualités. Un CMS open source est un bon choix. Souvent, jeter un coup d'œil au code permet de bien comprendre la philosophie de l'outil et donc de l'exploiter au mieux. Les CMS open source sont nombreux eux aussi. Alors lequel choisir ? Ceux qui sont en faveur d'un site codé à partir de zéro argumentent avant tout que cela permet de faire exactement ce que l'on veut, alors qu'un CMS apporte des contraintes et des limitations.

Faire facilement et exactement ce que l'on veut, voilà sans aucun doute le premier critère pour choisir un CMS, et c'est là que Drupal excelle comme vous le montrera ce dossier. Il est vrai que Drupal repose sur une architecture complexe et qu'il est indéniablement plus difficile à prendre en main et à maîtriser que d'autres produits. Mais cette étape passée, la récompense se savoure au quotidien, car la modularité de Drupal lui confère une souplesse remarquable. Des milliers de modules sont là pour vous éviter de réinventer la roue. Certains modules, Views par exemple, sont rien moins que fantastiques ! Sinon en quelques lignes de code vous pouvez écrire le petit module qui ajoutera exactement là où vous le voulez, la fonctionnalité qui vous manque, ni plus ni moins. Car tout peut être customisé dans Drupal, sans toucher pour autant au cœur et ainsi continuer à profiter sans souci de ses mises à jours régulières.

De nos jours la sécurité est un aspect fondamental des sites Internet. Drupal est extrêmement sûr et robuste et bénéficie d'une grande communauté, très réactive. Les problèmes de sécurité sont en général corrigés avant même qu'ils ne soient exploités.

Récemment nous avons passé le site de *Programmez!* sous Drupal. Nous n'avons pas rencontré le moindre incident depuis. Rien n'est venu nous faire regretter ne serait-ce qu'un peu notre choix. Que du bonheur !

 Frédéric Mazué
Journaliste et webmaster de programmez.com

Bien démarrer

Commencer avec un nouveau CMS n'est jamais évident. Cependant la version 7 de Drupal a été améliorée pour mieux appréhender les projets et répondre aux attentes du web dynamique. Avec l'approche de la version 8, il est impératif de ne plus commencer de nouveaux projets en Drupal 6, même si celui-ci est toujours disponible. Il est temps de passer à Drupal 7.

STRUCTURE

La structure du CMS est spécifique à Drupal. Tout d'abord, la gestion modulaire permet de l'utiliser comme un CMF, car il s'appuie sur les concepts de base des frameworks qui lui sont propres. Cependant, il est nécessaire de prévoir un temps d'adaptation à la conception et à son fonctionnement. Drupal est souvent considéré comme un CMS pour développeurs car l'API (application) s'appuie sur les Hooks. C'est pourquoi dans un projet web, la partie développement est mineure, mais provoque des contraintes structurelles plus fortes.

L'approche

Concept

Le concept de Drupal s'appuie sur une utilisation de Noeud, tant pour le contenu, que pour le type de contenu. Il est utilisé pour l'ensemble du CMS et permet de séparer la programmation et la présentation. Or la conception d'un thème est scindée entre les modèles et les templates, permettant de mieux contrôler le contenu.

Architecture

Pour faire fonctionner un site web Drupal, il vous faut un environnement AMP (Apache, MySQL, PHP). Ensuite vous téléchargez la dernière version de Drupal, à partir de l'adresse suivante : <http://www.drupalfr.org>

Avant de commencer un projet, il est important de respecter certaines règles :

Tout d'abord, il ne faut pas toucher à la partie Core. Le seul emplacement que vous pouvez manipuler, se trouve dans le dossier sites [Fig.1](#).

L'emplacement 'ALL' de la partie 'SITES' va contenir l'ensemble des extensions Drupal

► Librairies

► Modules

► Thèmes

Par contre le sous-dossier 'DEFAULT' est important car il contient de nombreuses informations

avec Drupal

Nom	Taille	Type
► includes	47 éléments dossier	
► misc	68 éléments dossier	
► modules	41 éléments dossier	
► profiles	3 éléments dossier	
► scripts	11 éléments dossier	
▼ sites	4 éléments dossier	
▼ all	2 éléments dossier	
► modules	17 éléments dossier	
► themes	1 élément dossier	
► default	3 éléments dossier	
example.sites.php	2,4 ko	script PHP
README.txt	904 octets	document texte brut
► themes	6 éléments dossier	
authorize.php	6,6 ko	script PHP
CHANGELOG.txt	88,7 ko	document texte brut
COPYRIGHT.txt	1,5 ko	document texte brut
cron.php	720 octets	script PHP
index.php	529 octets	script PHP
install.php	703 octets	script PHP
INSTALL.mysql.txt	1,7 ko	document texte brut
INSTALL.pgsql.txt	1,9 ko	document texte brut
INSTALL.sqlite.txt	1,3 ko	document texte brut
INSTALL.txt	18,0 ko	document texte brut

de configurations (ex : connexion à la base de données) et stocke les fichiers uploader (ex : les images)

Développement

Au niveau des développements, il existe certaines règles à respecter. Tout d'abord, il s'agit des normes de codage du CMS à connaître, disponibles sur le lien suivant :

<https://drupal.org/coding-standards/docs>

Le respect de ces règles permet de faciliter la maintenance du code et surtout c'est très utile si vous souhaitez proposer un module à la communauté. Cependant, lors d'un développement, il est important de prendre connaissance de l'API de Drupal (<https://api.drupal.org/>) et d'utiliser les fonctionnalités disponibles, par exemple :

► **t()** : traduit un texte dans la langue courante ou dans la langue proposée

► 10 : formate un lien au format HTML

DÉCOUPES DU DOSSIER SITES

Les 2 dossiers importants dans une architecture d'un projet Drupal à connaître sont les dossiers theme et modules car ils seront pris en compte lors du chargement d'une page.

Le “theme”

Le dossier thème se compose du template du projet. Il se compose de différents fichiers avec la possibilité de customiser l'affichage du résultat, c'est à dire d'ajouter une ou plusieurs couches supplémentaire à cette vue en JS/CSS. Cette souplesse peut être encadrée directement soit à partir de modules de types 'Panels' ou directement à partir du contenu.

Les modules

Le choix des modules proposés est très important mais une étape préalable au niveau de leur sélection est importante car certains modules offrent des fonctionnalités identiques plus ou moins évoluées ou redondantes.

- ▶ Pour effectuer son choix de modules, 3 pistes :
 - ▶ Il est important de regarder la popularité du module,
 - ▶ Ensuite, si le module subit des mises à jour régulières, c'est un plus dans votre choix,
 - ▶ Enfin, le nombre de fonctions des modules est aussi un critère important.

Il est préférable de ne pas toucher le code existant des modules de la communauté, mais plutôt de réaliser des extensions de celui qui vous intéresse. Le but, c'est que cela permet de faire des mises à jours plus facilement et de ne pas écraser vos modifications à l'intérieur de celui-ci. De plus, pour faciliter la maintenance, il est préférable de séparer dans ce dossier, les modules de la communauté, vos modules et extensions, et les Features.

Spécifications

Lorsque vous aurez choisi les différents modules qui répondront à l'ensemble des fonctionnalités demandées pour votre projet web, vous devez déterminer les contraintes et les limites des modules utilisés. En effet, si vous n'y prenez pas garde, vous rencontrerez des effets de bord non souhaités et cela impactera la performance. L'API de Drupal permet de répondre à cela. Bien entendu, il est important de spécifier les limites du projet pour définir exactement les types de contenus que vous aurez besoin et qui seront proposés aux contributeurs pour effectuer la saisie. Un autre point peut dériver très rapidement : l'utilisation des vues, c'est à dire l'affichage des résultats

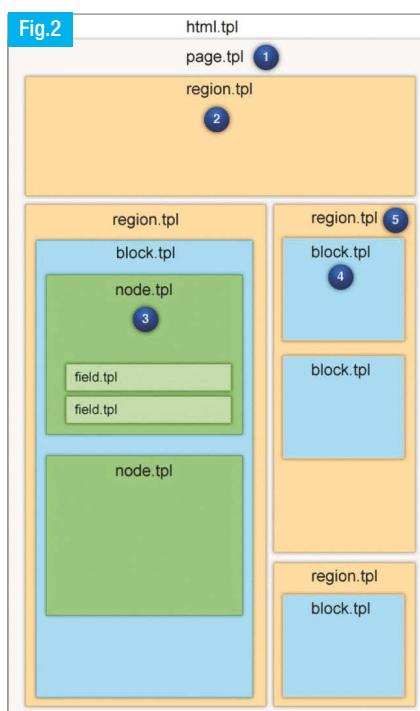
et l'absence des critères d'arguments par rapport à la navigation dans le projet. Ce point est pris en compte lors de la gestion de l'arborescence. Le dernier point est la gestion des utilisateurs et des différents droits 'workflows' car son approche est très souple et ouverte. Cette ouverture permet de définir les droits d'accès très précis et poussés jusqu'à la gestion des droits des blocs et même des champs des types de contenus. Les spécifications sont une étape à part entière dans la réalisation d'un projet Drupal car elles vont vous servir de socles à votre projet.

Graphismes et ergonomie

L'approche graphique d'un projet Drupal est unique pour chaque projet. Elle propose une approche du template différente des autres CMS, en utilisant la notion de blocs comme point de départ. Elle permet ensuite de définir les régions qui elles-mêmes permettront de définir la page HTML. Une région se positionne dans une page sur le même principe qu'une entête, un pied de page, des colonnes à gauche et à droite, etc... **Fig.2.** Ce mode d'affichage permet de contrôler le contenu des blocs par rapport à la position de la navigation. Bien entendu, la conception du Design reste identique à l'ensemble des projets.

Validation & performance

Avant de mettre un projet web en ligne, il est conseillé d'utiliser les outils disponibles du CMS. Ainsi, au cours de votre développement, vous pouvez vous appuyer sur des outils comme Coder, Security Review, Acquia Insight... qui vous aident à améliorer la qualité de votre projet. Cependant, lorsque votre site web sera en production, il est important d'accélérer l'affichage des pages, comme l'activation des caches internes de Drupal. Cette opération est utile au niveau de la performance. Bien entendu, l'utilisation de caches externes comme Varnish ou Akamai sont utiles pour répondre à des besoins très importants au niveau de l'accès au contenu. Toutefois, si vous utilisez des composants externes du type Apache Solr (fonctionnalité avancée de recherche) ou encore APC ou memcache, vous devez penser à les déployer. De plus, les bonnes pratiques web sont nombreuses comme le tuning avec la base de données, l'utilisation des vues... Mais aussi la validation des pages qui doivent être conforme au W3C. Pour cela vous vous appuyez sur les différents outils webs disponibles (Yslow, Pagespeed, webpagetest...)



Christophe Villeneuve

Christophe Villeneuve
Consultant IT pour Neuros
auteur du livre "Drupal7 en mode avancé" aux éditions Eyrolles
Rédacteur pour WebRIVER, membre des Teams DrupalFR,
AFUP, LeMug.fr, Drupagora, PHP7V.

Drupal sur Azure

La plateforme Microsoft Azure est interopérable et Microsoft signe de nombreux accords avec des éditeurs comme Oracle ou SAP. Drupal fut l'un des premiers partenaires supportant leur solution dans le Cloud Azure. Nous allons voir à travers cet article quelles sont les solutions de déploiement possibles à ce jour, quelles architectures sont possibles et comment migrer un projet Drupal existant.

Déploiements

Avant de déployer ou de migrer une solution Drupal, il est important de comprendre l'architecture type. La **figure 1** montre une architecture Drupal on-premise. Drupal est un logiciel multiplateforme, il peut s'installer soit sur un serveur Windows soit Linux, le serveur Web peut être IIS ou Tomcat et la base de données peut être MySQL ou SQL Server. Comme vous le voyez les technologies et combinaisons sont assez variées.

Regardons à présent quelles sont les possibilités offertes par Microsoft Azure. Nous avons en fait trois possibilités comme le montre la **figure 2**. A chaque scénario ses avantages et inconvénients bien évidemment (tableau 1). En tout état de cause, chacun offre des fonctionnalités identiques de construction de fermes web (pour apporter à la fois disponibilités et performances), de scaling automatique, et de monitoring sur l'activité. Nous y reviendrons un peu plus loin dans cet article. L'environnement technologique souhaité (par Exemple : Linux comme OS ou Tomcat comme serveur web) peut contraindre la solution finale. En effet, si vous envisagez d'installer Drupal sous Linux, vous êtes contraint par l'installation via une VM Azure.

Tableau 1 : Avantages et Inconvénients

Solution	Avantage	Inconvénient
WebSite	Rapidité de déploiement	Uniquement OS Windows
CloudService	Possibilité d'installer Tomcat	Architecture Stateless
VM	Linux supporté	Temps d'installation

Côté base de données, nous avons également plusieurs possibilités offertes par la plateforme. Tout d'abord on peut choisir entre deux technologies MySQL ou SQL Server (ce choix peut être influencé soit par un existant à migrer, soit en fonction des appétences de chacun). Ensuite, opter pour une solution as-a-service (on parlera alors de *MySQL Database* ou *SQL Database*) ou pour l'installation sur une machine virtuelle par vos propres soins (Azure IaaS). L'avantage des solutions as-a-service est que la partie Infrastructure, OS et installation logicielle sont pris en charge par le fournisseur, vous vous concentrez sur votre application et ses données J.

« MySQL Database » est proposé par le partenaire **ClearDb** et vous propose une version entrée de gamme limitée mais gratuite (idéal pour vos développements), vous pouvez ensuite facilement migrer vers une autre offre selon vos besoins (**fig 3**) à travers le dashboard de ClearDb (**fig 4**). En cliquant sur « Upgrade », on peut migrer en toute transparence vers une offre supérieure. Techniquement, on peut combiner ces solutions, comme par exemple :

- ▷ WebSite avec une VM SQL Server
- ▷ WebRole avec MySQL Database
- ▷ VM sous Linux avec SQL Database
- ▷ ...

Même si certaines combinaisons sont discutables voire non recommandées.

Drupal sous WebSite

Pour déployer Drupal en utilisant les WebSites et MySQL, il faut suivre les étapes suivantes : instancier les ressources Azure, migrer votre base de données, modifier certains paramètres et publier votre site Drupal.

1. Provisionner l'infrastructure

La première étape consiste bien évidemment à provisionner le site web et la

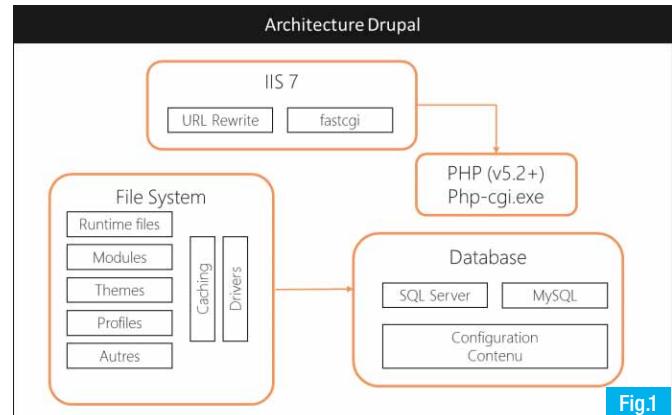
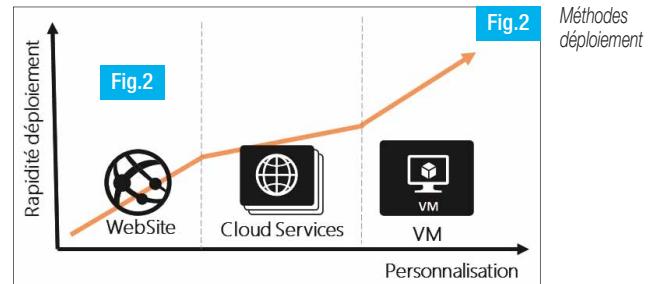


Figure 1: Architecture Drupal



Select ClearDB on Windows Azure

Windows Azure

ClearDB for Windows Azure Service Plans

- Mercury** (FREE): DB Size up to 20MB, Connections up to 4, Performance Low. Features: Daily Backups, Great for getting started and developing your apps.
- Venus** (\$9.99/mo): DB Size up to 1GB, Connections up to 15, Performance Moderate. Features: Daily backups, Excellent choice for light test and staging apps that need a reliable MySQL database.
- Saturn** (\$49.99/mo): Recommended. DB Size up to 5GB, Connections up to 30, Performance Moderate. Features: Daily backups, Great for apps with moderate data requirements that need rock solid MySQL availability.
- Jupiter** (\$99.99/mo): DB Size up to 100GB, Connections up to 40, Performance High. Features: Daily backups, Designed for apps where you need real MySQL reliability, power and throughput.

Offres MySQL Database

Welcome, Michel Hubert | Help & Support | My Account | Sign Out

cleardb

My Dashboard

Here is a list of your current databases. If you want to see more detail about a particular database, click its name.

Create a new database

My Databases

Name	Current DB Size	Maximum DB Size	Status	Message	Database Plan	Actions
celennzastaffdb	0.1 MB	20 MB	Online	All systems are operating normally.	Mercury	Upgrade Delete

Showing 1 of 1 items

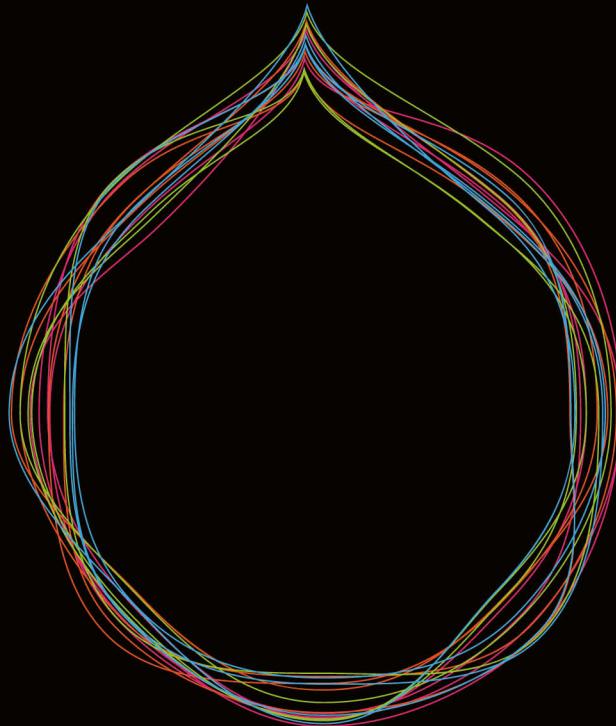
Dashboard ClearDB

base de données MySQL dans notre exemple. Rien de plus, à partir du nouveau portail Azure, il faut faire New/WebSite. (**fig 5**).

Ensuite, on rajoute une ressource de type MySQL (**fig 6**), ainsi les deux ressources sont déclarées comme liées.

Une autre possibilité consiste à créer un site web à partir de la galerie ce qui permet d'instancier un site Drupal préinstallé (**fig7**)

Dès que le site est créé, vous vous connectez sur le site (<http://drupalcelle...>



Une plate-forme digitale unique
qui intègre Commerce, Contenu
et Communautés



Optimisez et Personnalisez
l'expérience utilisateur sur
votre site avec Acquia Lift



Réduisez votre Time to
Market avec Acquia Cloud
Site Factory



Créez des expériences
e-Commerce incomparables
avec Acquia Commerce
Cloud

za.azurewebsites.net dans notre exemple) et vous pouvez lancer l'installation de Drupal (fig 8) via l'interface web, suivie de la configuration de Drupal (connexion base de données, ...)

2. Migrer votre base de données MySQL

Je suis sûr qu'il existe plusieurs façons de copier une base Drupal, mais j'ai l'habitude d'utiliser mysqldump (<http://dev.mysql.com/doc/refman/5.6/en/mysqldump.html>). Pour copier de votre poste de développement vers les WebSite Azure, voici la commande à utiliser :

```
mysqldump -u local_username -password=local_password drupal
| mysql -h remote_host -u remote_username -password=remote
_password remote_db_name
```

Bien entendu, vous remplacez les paramètres par les éléments de connexion de votre base de données MySQL fournis par le portail Azure en cliquant sur : « Afficher les chaînes de connexion ». Vous devez avoir une chaîne de connexion qui ressemble à cela :

```
Database=remote_db_name;Data Source=remote_host;User Id=remote
_username;Password=remote_password
```

Selon la taille de votre base de données et votre bande passante, la copie peut durer un certain temps. A présent votre base de données est prête.

Avant de déployer votre code Drupal, vous devez le modifier pour paramétrer la base de données hébergée dans Azure.

3. Modifier les paramètres bases de données dans settings.php

Vous avez à nouveau besoin de votre chaîne de connexion à votre base de données. Ouvrez le fichier /drupal/sites/default/settings.php avec l'éditeur de votre choix et remplacez les valeurs 'database', 'username', 'password', and 'host' du tableau \$databases avec les bonnes valeurs. Vous devez avoir un code qui ressemble à cela :

```
$databases = array (
  'default' =>
  array (
    'default' =>
    array (
      'database' => 'remote_db_name',
      'username' => 'remote_username',
      'password' => 'remote_password',
      'host' => 'remote_host',
      'port' => '',
      'driver' => 'mysql',
      'prefix' => '',
    ),
  ),
);
```

Après avoir sauvegardé le fichier, vous êtes prêt à déployer le code Drupal.

4. Déploiement du code Drupal avec Git ou FTP

La dernière étape consiste à déployer votre code vers le WebSite Azure en utilisant Git ou FTP.

Si vous utilisez le protocole FTP, vous pouvez récupérer vos paramètres de connexion à partir du dashboard du website (hostname, username). Ensuite vous pouvez utiliser votre client FTP favori comme FileZilla pour uploader vos fichiers Drupal vers le dossier /site/wwwroot de votre site distant dans Azure. Si vous utilisez Git, vous devez configurer votre “repository” Git dans votre WebSite Azure (vous pouvez également utiliser Visual Studio Online, Dropbox ou autres gestionnaires de code source). Ensuite vous devez instal-

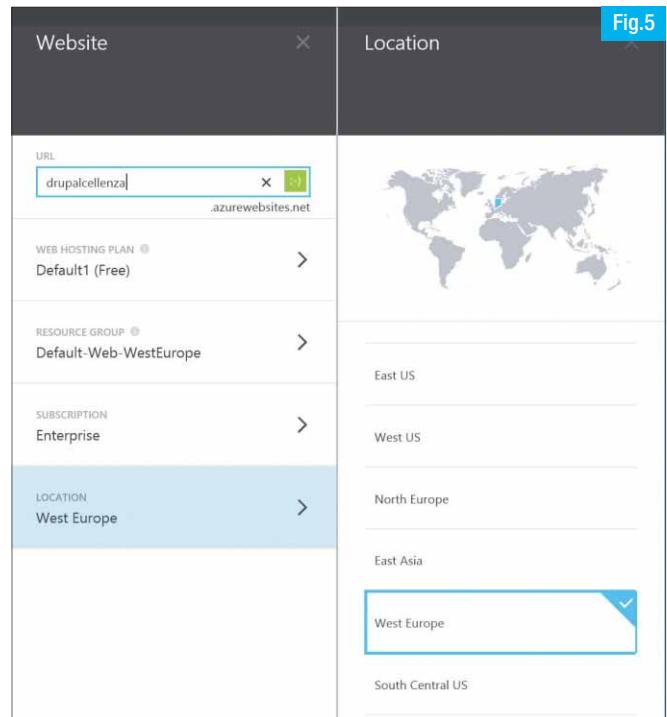


Figure 5 : Création d'un site web

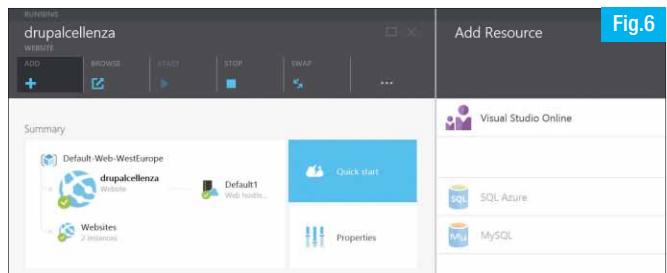


Figure 6 : Ajout MySQL



Figure 7 : Galerie Site Web

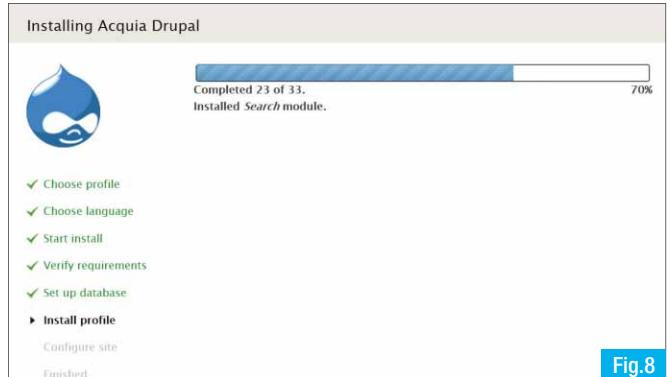


Figure 8 : Installation Drupal



A **CREATIVE AGENCY**
POWERED BY **DRUPAL**



Audit
➤ applicatif
➤ performance
➤ sécurité



Maintenance
➤ préventive
➤ corrective
➤ évolutif



Architecture
technique



Développement



Théming



Support



Formation



Hébergement



PROPULSEZ VOTRE PROJET
AVEC L'AGENCE **100% DRUPAL**

ACTENCY PARIS
82 rue d'Hauteville / 75010 PARIS

ACTENCY STRASBOURG
45 avenue de Colmar / 67000 STRASBOURG

01 47 70 47 70

www.actency.fr
 contact@actency.fr

ler Git sur votre poste de développement. Puis, exécutez les commandes suivantes pour uploader votre code Drupal dès que votre repository est créé :

```
Git init
Git add
Git commit -m « premier commit »
Git remote add <URL de votre repository Git>
Git push azure master
```

Connectez-vous à présent sur l'URL de votre WebSite et votre site Drupal est en ligne !

Drupal sous Linux

Si vous avez l'exigence ou la contrainte d'avoir votre site Drupal sous un serveur Linux, la démarche est différente et un peu plus longue. Vous devez créer votre VM Azure (IaaS) en choisissant l'OS Linux de votre choix puis vous devez configurer le serveur selon vos besoins. Cela nécessite d'avoir les compétences nécessaires et peut prendre du temps.

Il existe un moyen un peu plus rapide dans Microsoft Azure en utilisant un template d'image virtuelle fourni par la communauté, connu sous le nom de VM Depot (<http://vmdespot.msopentech.com>List/Index>).

Pour ce faire, allez dans la rubrique « Virtual Machine » du portail et cliquez sur l'onglet « Images », vous verrez l'option « Parcourir le dépôt d'images virtuelles ». Il existe un grand nombre de templates Drupal selon différents OS (CentOS, Ubuntu) et différentes versions de Drupal. Après avoir renseigné les paramètres de stockage notamment, le template est téléchargé en tant qu'image dans votre espace de stockage.

Dès que le téléchargement est terminé, vous cliquez sur « Incrire » pour enregistrer cette image dans votre bibliothèque d'images. A présent, quand vous créez une image virtuelle, à partir de la galerie, un nouveau template est présent dans la rubrique « Mes Images » (fig 9)

Les médias

Un dernier point à traiter est la gestion des médias (images, fichiers, ...). Dans une architecture on-premise, on utilise un espace de stockage partagé pour y placer les fichiers statiques comme les images, cela évite de devoir les répliquer (et les synchroniser) entre les différents serveurs d'une même ferme. Dans Microsoft Azure, le composant équivalent est le **Blob storage**. Drupal fournit un module qui permet d'utiliser ces blobs au sein de Drupal directement, le module se nomme : Windows Azure Integration Module (<https://drupal.org/project/azure>).

Lorsque vous migrez à partir d'un site existant Drupal sous environnement LAMP, vous pouvez rencontrer un souci avec les fichiers contenant « : » dans le nom de fichier. Cette convention de nom de fichiers n'est pas supportée par Windows, il les remplace par le caractère « _ ». Cependant, les noms de

Figure 9 : Nouveau template

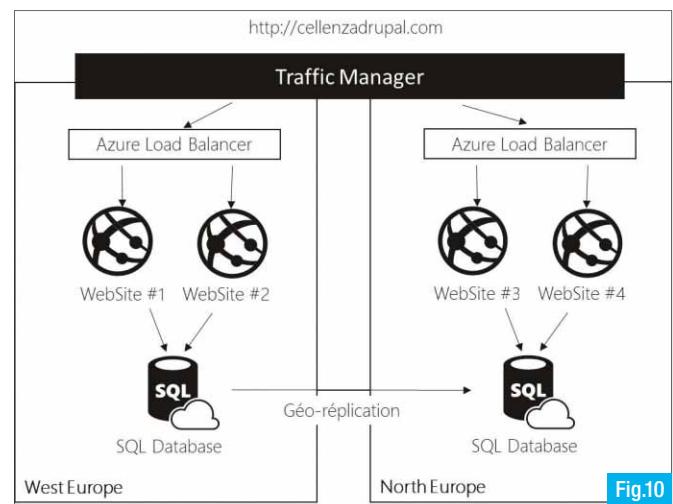
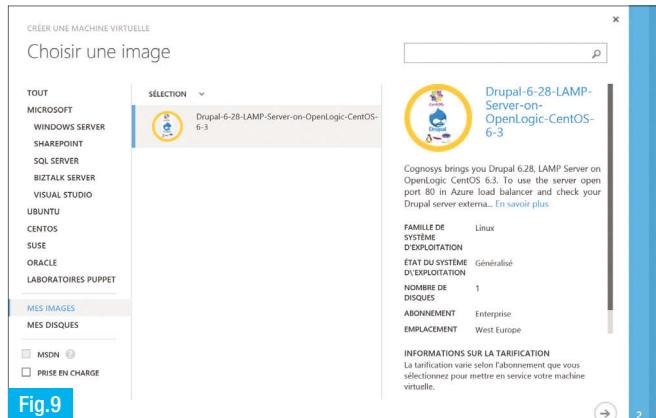


Figure 10 : Haute Disponibilité

fichiers contenant « : » sont bien supportés par les blobs, le problème vient du fait de passer par un poste de développement sous Windows pour effectuer la migration. Vous n'avez dans ce cas pas d'autres choix que de renommer les fichiers concernés directement dans les blobs.

Haute Disponibilité

Si votre site Drupal est jugé comme « Mission Critical », la question de la haute disponibilité se pose. Construire une ferme Web est un premier élément de réponse intra-datacenter. Si on souhaite mettre en place un DRP (Disaster Recovery Plan) pour votre solution Drupal ou accroître votre niveau de service, Microsoft Azure permet de déployer votre solution sur plusieurs datacenters et grâce à Traffic Manager, vous avez à disposition un « load balancer » inter-datacenter. Traffic Manager fonctionne selon 3 modes :

- **Performance** : quand ce mode est sélectionné, Traffic Manager redirige vers le datacenter où l'utilisateur est le proche (meilleures performances), cela évite des situations où un utilisateur aux Etats-Unis est dirigé vers le site en Europe.
- **Round Robin** : ce mode va distribuer le trafic http entre les différents end-points paramétrés dans Traffic Manager.
- **FailOver** : redirige vers le datacenter secondaire si le premier ne répond plus, mode qui nous intéresse dans ce cas précis. Par exemple, vous pouvez avoir un site secondaire qui ne reçoit aucun trafic à moins que le site primaire ne soit en mode Offline pour différentes raisons.

Pour la partie Data, Microsoft Azure propose à présent une nouvelle édition de SQL Database appelée Premium (en preview). Cette nouvelle version inclut des solutions de « business continuity » comme la géo-réplication entre deux datacenters. En combinant ces deux fonctionnalités, on obtient l'architecture dessinée en figure 10, apportant une réelle réponse de business continuity. Cette architecture est complexe à mettre en œuvre on-premise et est extrêmement coûteuse. Le Cloud est une réponse intéressante dans ce cas d'usage !

Conclusion

Comme vous l'avez compris au travers de cet article, il existe plusieurs architectures possibles et différents modes de déploiement d'un site Drupal : du plus rapide avec les WebSites au plus personnalisable avec les machines virtuelles. Cette variété permet de couvrir un grand nombre de besoins en fonction de votre existant, vos compétences et les pré-requis des exploitants.

Microsoft Azure apporte en plus à votre site des fonctionnalités comme le scaling automatique afin d'absorber les pics de charge.



Michel Hubert

Manager Pôle Cloud & Integration - Cellenza



BONJOUR, NOUS SOMMES AMBIKA,
DRUPAL LOVER
DEPUIS 2006.

Développement
Hébergement



ambika.fr

Attention au piège du tout module !

Drupal doit sa renommée en partie à son système modulaire qui permet d'assembler différents modules communautaires afin de construire un projet Web assez complexe sans avoir à écrire une ligne de code (ou peu s'en faut).

Néanmoins, cette approche peut rapidement présenter plus d'inconvénients que de plus-values au regard d'une autre approche qui consiste à se baser d'avantage sur le Framework et les API de Drupal plutôt que sur l'assemblage pur et simple de dizaines, voire de centaines, de modules.

En effet, Drupal est avant tout un Framework qui permet d'implémenter des projets à l'aide de mécanismes tels que des API et des Hooks, le tout en offrant un véritable environnement de développement (séparation de la logique et du rendu, API de tests unitaires, architecture modulaire...) ainsi qu'un package de fonctionnalités de base (gestion des utilisateurs, rôles et permissions, type de contenu, gestion des champs etc.).

2 approches possibles

Deux approches se distinguent donc pour implémenter un projet Drupal : l'approche tout module (ou "Site Building") qui consiste à utiliser majoritairement des modules communautaires pour réaliser les fonctionnalités souhaitées, et l'approche Framework, qui consiste à beaucoup plus utiliser le framework interne de Drupal afin de minimiser le nombre de modules communautaires.

La question est de savoir quelle méthode utiliser pour développer, ou faire développer; votre projet en Drupal est à mettre en adéquation avec les objectifs qu'il adresse ainsi que les compétences techniques des personnes qui l'implémenteront.

Dans le cadre d'un projet pris en charge par des profils ayant peu ou pas de compétences en programmation (PHP, MySQL) et/ou pour un projet de petite à moyenne envergure, l'approche "Site Building" peut s'avérer être un bon compromis pour déployer un projet Web sans avoir à rentrer dans la "mécanique" de Drupal. En revanche, dans le cas d'un projet mis entre les mains de développeurs expérimentés, pour des projets avec des composantes métier plus fortes (Portails Web, portails d'entreprise,

intra/extranet, Job-Board, site médias, comparateurs etc.) l'approche Framework présentera au final beaucoup plus d'avantages sur les plans de la performance, de la stabilité, de la maintenance ou encore de la sécurité.

Construire l'intégralité, ou presque, d'un projet à l'aide de modules communautaires limite certes le nombre de ligne de code à écrire mais impose au système des quantités de lignes de codes supplémentaires, présentes dans les modules (certains comme le module Views, l'un des plus utilisé en Drupal 7, pèse plus de 82 000 lignes de code).

Quand on sait, qu'en moyenne, un module Drupal est installé pour une utilisation de 20 à 30% de ses fonctionnalités, cela représente tout de même de 70 à 80% de code inutilisé dans votre projet.

Multiplié par plusieurs dizaines (et parfois plus d'une centaine) de modules, cela n'est pas sans conséquences sur les performances globales de votre site, malgré tous les mécanismes d'optimisation de chargement du code ou de caches présents dans Drupal.

Ne construire son site qu'avec des modules signifie également de paramétriser leurs fonctionnalités et leurs interactions quasiment exclusivement via le Back-Office, au "clic" de souris. Sachant que chacune de ces configurations est enregistrée en Base De Données, et que deux bases de données ne peuvent pas être fusionnées, se alors pose la question de savoir comment reverser de nouvelles fonctionnalités développées sur un environnement de développement dans un projet qui tourne en production, et dont la B.D.D. ne peut être simplement écrasée.

Bien que des modules d'import, comme Features, existent, ces derniers sont très loin d'être compatibles avec tous les modules et également très loin de permettre un import pur et simple des configurations de tous les modules de votre projet.

La mise en production de nouvelles fonctionnalités se résume donc souvent à une répétition

des configurations sur la version de production, avec tous les risques (oubli, mauvaise manipulation) et tous les inconvénients (double jeux de tests, délais...) que cela représente.

D'autres problèmes se posent à l'utilisation du tout module, comme la sécurité (il faut veiller à mettre à jour tous les modules), la stabilité (une mise à jour peut rendre un module incompatible avec votre projet) ou encore la traçabilité (certaines configurations de modules ne peuvent être connues que des personnes qui les ont paramétrées, QUID alors des personnes chargées de reprendre en main le projet après ?). Reste alors la question de savoir s'il est plus économique et/ou plus rapide de développer un projet Drupal avec les modules communautaires plutôt qu'avec le Framework.

Cela se vérifie dans le cadre de projets de petite envergure, sans grandes complexités techniques et avec assez peu de fonctionnalités métiers.

En revanche, l'expérience tend à montrer que les coûts et les délais d'une implémentation en "tout module" augmentent proportionnellement avec la richesse fonctionnelle et la complexité technique du projet.

Alors faut-il renier définitivement les modules communautaires pour développer un projet Drupal ?

Non, tout simplement parce que certains modules sont développés comme des extensions du framework et livrés avec des API, le tout en offrant des périmètres fonctionnels riches qu'il ne serait pas forcément judicieux de redévelopper.

Pour résumer, plus le périmètre fonctionnel est riche, la complexité technique importante et les exigences du projet élevées (hautes performances, stabilité, montée en charge etc.) et plus il sera judicieux de minimiser l'utilisation de modules communautaires, en favorisant l'utilisation du Framework interne de Drupal.

 Dorian Marchand

Directeur Général de Kernel 42.

À LIRE DANS LE PROCHAIN NUMÉRO

n° 177 en kiosque le 30 août 2014

Formation / carrière

Comment apprendre le code à l'école ?

Un point complet sur des initiatives concrètes.

Dossier PHP

Quel avenir pour PHP ?

PHP en mode cloud
Un outillage de plus en plus complet

3D

Babylon.JS, la tuerie !

Le moteur 3D surdoué

Drupal et ses nombreux modules

Ce n'est pas par hasard que le slogan de Drupal et de sa communauté est « Come for Software, stay for the community ». Une des grandes forces de Drupal est son architecture modulaire et forcément les nombreux modules disponibles. Ainsi lors de la construction d'un projet Drupal et avant d'envisager tout développement spécifique, il peut être pertinent de se demander si un module ne répond pas déjà au besoin exprimé. Pour cela Drupal.org centralise les modules sans toutefois qu'il y ait une quelconque validation de ceux-ci.

L'ensemble des modules est disponible à cette adresse : <https://drupal.org/download>. Avec aujourd'hui près de 9000 modules pour Drupal 7, il peut parfois être compliqué de choisir. Certains modules font la même chose ou vraiment très proche, il est alors difficile de savoir lequel utiliser. Pour trouver un module, n'hésitez pas à fouiller dans les catégories. Si vous préférez, vous pouvez également utiliser la recherche, dans ce cas il faut souvent essayer plusieurs synonymes en anglais pour trouver ce que l'on cherche. Une fois le ou les modules trouvés, voici quelques points à regarder :

- ▶ la version du core de compatibilité : le module existe-t-il pour votre version de Drupal ?
- ▶ Le module est-il stable (en vert), en version de test (aussi appelé bêta en jaune), ou seulement en version de développement (en rouge), dans le dernier cas, il n'est pas recommandé d'utiliser ce module, sauf pour s'impliquer dans la communauté et participer activement à l'évolution de ce module
- ▶ la date de la dernière version permet de savoir si le module est toujours maintenu
- ▶ l'état du développement du module : un ou plusieurs mainteneurs sont présents et toujours actifs
- ▶ Le nombre de téléchargements donne un aperçu des bugs qui auront pu être remontés, plus un module est utilisé et plus il y a des chances pour que les bugs soient rapportés et corrigés (pour cela un tour dans l'issue queue montre l'activité des mainteneurs sur ce module)

Sélection de modules

Afin de démarrer facilement, la liste de modules ci-dessous donne un aperçu des possibilités de Drupal :

Wysiwyg est une librairie de wysiwyg, sera un de vos premiers modules, avec la particularité que celui-ci sert uniquement d'interface avec une librairie de wysiwyg externe comme CKEditor ou TinyMCE. Une fois installé si vous allez sur la page de configuration Administrer > Configuration > Wysiwyg la procédure d'installation de la librairie externe est détaillée. N'oubliez pas de configurer le format d'entrée utilisé par les

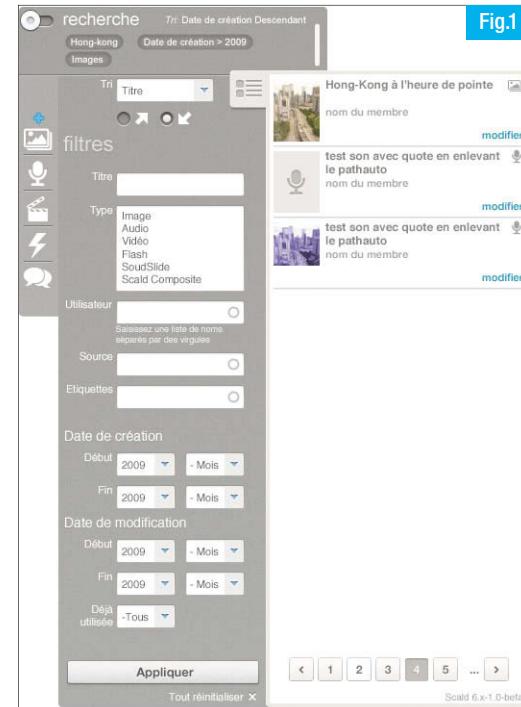
contributeurs de votre site pour leur permettre d'utiliser le wysiwyg, en sélectionnant à la fois les boutons de wysiwyg disponibles, mais aussi si le format est *filtered HTML* en ajoutant les balises HTML nécessaires à celles disponibles par défaut (par exemple les balises pour faire un tableau).

À la fois le wysiwyg à travers la librairie CKEditor sera intégré nativement à Drupal 8.

Views est un outil très puissant permettant de créer des listes de contenus de manière dynamique, il se rapproche d'un requêteur SQL. Cet outil très puissant demande un peu de temps de prise en main et permet avec quelques connaissances de rapidement mettre en place des affichages de pages ou de blocs de contenus. Views s'utilise de deux manières, soit par le code, en déclarant des tableaux associatifs de l'API, ou via l'interface utilisateur. Le `hook_views_query_alter()` va rapidement devenir indispensable lors d'affichage complexe, notamment pour permettre l'utilisation du OR dans les filtres contextuels. Petite astuce : dans les paramètres de Views (dans Administrer > Structure > Views > Paramètre), l'activation de l'affichage SQL permet de se rendre compte de ce que l'interface effectue comme requête. À Views sera aussi intégré au cœur de Drupal 8.

PathAuto : petit module très pratique qui permet de définir des alias d'URL de manière automatique. En choisissant par type de contenu ou vocabulaire de taxonomie ou pour tout le site, la manière dont les alias d'URL du site sont construits.

Rules permet de construire des règles avec le même principe que Actions et Trigger (des modules du cœur de Drupal), mais avec la particularité de pouvoir ajouter des conditions aux déclencheurs et d'ajouter beaucoup plus de déclencheurs disponibles et d'actions. De plus les autres modules du cœur et les modules



contribués utilisent l'API de Rules pour ajouter des actions et des déclencheurs augmentant encore ses possibilités.

Workflow est un module pour gérer finement des étapes de publications. Celui-ci permet de définir différentes étapes dans notre publication, et en s'appuyant sur les rôles de Drupal, de définir quelles sont les personnes autorisées à faire avancer le contenu dans la chaîne de publication. Celui-ci donne la possibilité de définir un workflow différent par type de contenu permettant ainsi d'avoir différents processus de validation du contenu séparés sur le site.

Media/Scald deux modules pour gérer vos médias, le premier étant aussi le plus ancien, il permet plus de choses que le second. Scald présente l'avantage d'avoir été pensé pour une utilisation pour la presse notamment, et permet donc de mettre en place avec une interface très intuitive une bibliothèque de média sur votre site. Dans les deux cas, il existe un ensemble de sous-module vous permettant de vous interfaçer avec des librairies externes comme YouTube ou Tumblr **Fig.1**.

Panels/Display Suite : là encore deux modules qui permettent de définir des affichages avec une interface utilisateur. Display suite dispose d'une interface par glisser-déposer qui est très intuitive, là où Panels permet plus de souplesse sur de nombreux points. N'hésitez pas à les tester, et à utiliser l'un ou l'autre en fonction des objectifs du projet [Fig.2](#).

Localization Update est un module indispensable puisqu'il permet d'importer de manière automatique les traductions mises à disposition par la communauté.

Features sera utile pour un travail en équipe puisqu'il permet d'exporter toute (ou presque) la configuration du site stocké en base de données, sous forme de fichiers php, et donc de gérer ses configurations avec un gestionnaire de source type GIT ou SVN.

Devel et Devel Themer : les deux modules indispensables pour développer sur Drupal et pour *themer* son site. Devel permet notamment de faire du débug (coupler avec Xdebug par exemple), et d'afficher les requêtes SQL exécutées, permettant ainsi d'optimiser son site en terme de performance.

Honeypot sécurise les formulaires du site contre les spams sans demander d'actions supplémentaires aux utilisateurs comme peut le faire un captcha

Security review liste les principales failles de sécurités découvertes sur votre site et le scanne régulièrement pour se prémunir contre les attaques

Search API est un moyen efficace d'obtenir une recherche beaucoup plus efficace que la recherche par défaut de Drupal. Le module s'appuie soit sur une recherche en base de données, soit avec des moteurs de recherche externe comme Apache Solr.

Les dépendances

Un autre aspect des modules de Drupal est qu'il arrive régulièrement que, pour fonctionner, ceux-ci ont besoin d'autres modules, on parle alors de dépendances. Voici les principales dépendances (qui ne sont rien d'autre que des modules) qui sont utilisées :

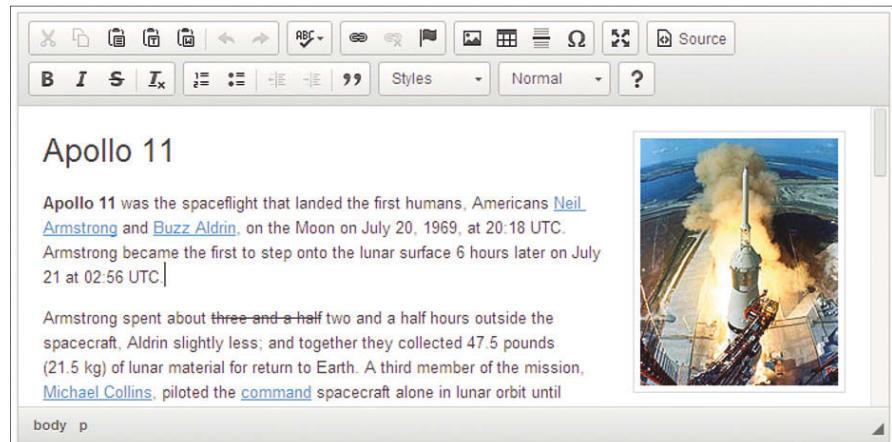
Token permet de variabiliser énormément d'informations du site et de les utiliser ailleurs en fonction du contexte de la page par exemple.

Entity API définit le format utilisé par les entités Drupal et permet ainsi de normaliser un minimum les déclarations de nouvelles entités.

Chaos Tools suite d'outils nécessaires en tant

FIELD	REGION	LABEL	FORMAT
Left			
Author	Left	<Hidden>	Author
Body	Left	<Hidden>	Default
Right			
Tags	Right	<Hidden>	View mode: Taggy
Post date	Right	<Hidden>	Long
Disabled			
Read more	Disabled	<Hidden>	Default
User picture	Disabled	<Hidden>	Thumbnail
Comments	Disabled	<Hidden>	Default

Fig.2



Apollo 11

Apollo 11 was the spaceflight that landed the first humans, Americans [Neil Armstrong](#) and [Buzz Aldrin](#), on the Moon on July 20, 1969, at 20:18 UTC. Armstrong became the first to step onto the lunar surface 6 hours later on July 21 at 02:56 UTC.

Armstrong spent about three and a half hours outside the spacecraft, Aldrin slightly less; and together they collected 47.5 pounds (21.5 kg) of lunar material for return to Earth. A third member of the mission, [Michael Collins](#), piloted the [command](#) spacecraft alone in lunar orbit until

CKEditor

qu'API pour de nombreux modules.

Libraries API sert de déclaration générique aux bibliothèques externes utilisées par le site (comme CKEditor ou les bibliothèques javascript JqueryMulti, le cœur de Drupal embarque une version de jQuery, mais ce n'est pas la dernière version et certains modules ou thèmes peuvent nécessiter une version plus récente de jQuery, ce module donne la possibilité d'embarquer plusieurs versions.

Bonnes pratiques

En plus de sélectionner avec soin les modules à utiliser, il y a quelques règles simples qui facilitent le travail quotidien.

Les modules ou ceux que l'on a soi-même développés doivent être placés dans le répertoire/sites/all/modules ce qui évitera des surprises lors des mises à jour du cœur de Drupal.

Une seule version du module est utilisée en même temps sur le site, cela évitera à Drupal de confondre les fichiers, et d'être perdu pour retrouver les fonctions appelées par le code. Limitez au maximum le nombre de modules utilisés, dans l'idéal un site ne comportera pas plus d'une centaine de modules, sous peine de devoir allouer 1Go de RAM à Drupal et de voir le site mettre plusieurs dizaines de secondes à

s'afficher. Prendre 1h tous les mois pour passer en revue ces modules et désactiver ceux qui ne sont plus utilisés. Cela permet d'optimiser en fonction de l'utilisation réelle de son site et donc d'améliorer l'expérience utilisateur du site.

Un module non utilisé devrait être désactivé ou mieux désinstallé, ce qui permettra de supprimer des informations inutiles en base de données et de gagner en performance.

Comme pour le cœur de Drupal, les modules sont mis à jour régulièrement par leurs mainteneurs, n'hésitez pas à activer l'envoi de mail lors de mise à jour disponible, et à vous abonner au compte Twitter Drupal Security pour savoir très rapidement lorsque des failles de sécurité sont découvertes. En cas de mise à jour, faites celle-ci le plus vite possible pour se prémunir d'éventuelles attaques.

N'hésitez pas à demander de l'aide à la communauté française ou anglophone pour trouver des réponses à vos questions.

 **Vanessa Kovalsky David**,
Lead développeuse Drupal /PHP chez Webnet
<http://vanessakovalsky.net>

Membre actif de DrupalFr, et de la promotion du logiciel libre avec l'ALDI
Lead dev sur le projet Esecouristes :
<https://github.com/vanessakovalsky/esecouristes>

7 ans d'expérience

180 jours de formation par an

600 sociétés nous font confiance

+DE 900 stagiaires ont suivi nos formations

9 formations personnalisables



FORMATION DRUPAL

DÉVELOPPEZ VOS COMPÉTENCES DRUPAL



CONTRIBUTEUR / PUBLISHER



PERFORMANCE & SÉCURITÉ



WEBMASTER



DÉPLOIEMENT



THEMEUR / DESIGNER



RESPONSIVE DESIGN



DÉVELOPPEUR



COMMERCE

8

NOUVEAU

INTRODUCTION À DRUPAL 8

Découvrez les principales nouveautés de Drupal 8, et apprenez les bases pour vous lancer !

ACCOMPAGNEMENT DE PROJET DRUPAL

UNE ÉQUIPE D'EXPERTS À VOTRE SERVICE



CONSEIL



ASSISTANCE
TECHNIQUE



AUDIT



SUPPORT



MAINTENANCE

Plus d'infos sur le site

www.drupalfrance.com

T / +33 1 46 94 69 32

F / +33 1 48 24 95 80

8 rue du Faubourg Poissonnière / 75010 Paris / contact@trainedpeople.com / www.trainedpeople.com

Les modules «clés» de Drupal 7

Avec plus de 30000 développeurs, la communauté Drupal est l'une des plus actives parmi les différents CMS. Chacun d'entre eux peut donc soumettre des modules, thèmes ou distributions sur le site Drupal.org, qui sert de plateforme pour centraliser le travail de la communauté. À ce jour, le site regroupe plus de 25000 modules, 800 distributions et près de 2000 thèmes. Dans cet article, nous allons vous guider pour choisir les modules très utiles, voire indispensables à la création de votre site.

Gestion et remontée de contenu

S'il ne devait y avoir qu'un seul module pour cette catégorie sur Drupal 7, ce serait sans doute Views. Pour preuve, il sera intégré directement dans Drupal 8.

Le module Views permet de créer des listes de remontées de contenu. Grâce à son interface très complète, vous allez pouvoir choisir de créer facilement des pages ou des blocs qui permettent par exemple d'afficher les 3 dernières actualités, ou les contenus taggés avec un certain terme.

Ses possibilités sont quasiment infinies [Fig.1](#).

Plusieurs modules ont été créés pour se greffer à Views, et lui ajouter des fonctionnalités :

- ▷ **Views Slideshow**, qui vous permettra d'afficher vos views sous forme de slideshow entièrement paramétrable,
- ▷ **Footable**, qui intègre le plugin jQuery du même nom à vos views pour créer des tableaux adaptés au responsive,
- ▷ **Views Infinite Scroll** chargera les éléments de votre views au scroll, ce qui permet de ne pas utiliser la pagination.

Fig.1

Fig.2

Gestion des blocs

Si votre site sous Drupal commence à être assez conséquent, alors vous avez sûrement déjà rencontré le problème d'avoir une multitude de blocs et de vous y perdre.

Pour vous aider à mieux gérer les blocs, au moins deux modules très pratiques existent : il s'agit de **Context** et de **Panels**. Ces modules vous permettront d'organiser vos blocs par type de page et ainsi d'avoir une vision plus claire de l'organisation de vos pages.

Mise en forme

Il est pratique de pouvoir mettre en forme le texte directement dans le back-office de votre site. Pour cela, Drupal a des modules qui vous permettent de mettre en place un WYSIWYG (What You See Is What You Get) sur les zones de textes longs (textarea) [Fig.2](#).

Pour mettre ceci en place, vous pouvez installer le module **WYSIWYG** qui vous demandera de choisir l'éditeur de votre choix (par exemple TinyMCE comme sur l'exemple ci-dessus). Il

faudra placer les fichiers sources de l'éditeur dans le répertoire sites/all/libraries. Vous pourrez ensuite le configurer à votre guise, pour choisir par exemple quels boutons sont activ, ou y importer une feuille de style personnalisée. Voici également un add-on pratique au WYSIWYG, il s'agit de **IMCE** (et son sous-module **IMCE Wysiwyg API Bridge**), qui permettent d'ajouter la fonctionnalité d'uploader des images et des fichiers sur le serveur directement via l'interface du WYSIWYG installé.

Référencement

De base, Drupal ne comprend pas de fonctionnalité permettant la gestion des métatags, qui sont indispensables au bon référencement de votre site. Heureusement, un module a été créé spécialement pour cela par la communauté. **Metatags** vous permettra de gérer vos balises meta, y compris les balises de partage Facebook et Twitter.

Le module **Sitemap** vous permettra de créer une

page "Plan du site", également utile pour le référencement. Pour continuer sur le plan du site, vous pourrez créer grâce à **Sitemap XML** sa version XML, que vous pourrez soumettre aux différents moteurs de recherche.

La gestion des redirections et des contenus dupliqués peut-être gérée par le module **Global Redirect**. Pour pousser un petit peu plus loin le référencement de votre site, il est possible d'adapter les différentes pages de votre Drupal aux normes de **Schema.org**, avec le module du même nom. Une fois installé, vous pourrez choisir via l'interface quel champ correspond à quel type d'élément du schéma.

Enfin, pour analyser les résultats de votre référencement, vous pourrez lier votre site à votre compte Google, pour bénéficier des statistiques avec le module **Google Analytics**.

Réécriture d'URL

Le module le plus utilisé pour la réécriture d'URL sur Drupal est **Pathauto**. Il vous permettra

d'automatiser la réécriture des URLs lors de la création de contenu, selon des règles que vous aurez définies auparavant. Petit ajout à ce module, **Transliteration** qui permet de remplacer les caractères spéciaux dans les URLs (par exemple, enlever les accents, etc.).

Formulaires

La création de formulaire sous Drupal est extrêmement simple grâce au module **Webform** Fig.3. Vous pourrez créer un formulaire directement dans l'interface du module, avec les champs dont vous avez besoin. Webform vous offre également la possibilité de configurer l'envoi de mails suite aux soumissions du formulaire. Ceci est très utile par exemple pour envoyer un email de confirmation à une personne ayant répondu à un questionnaire créé par l'intermédiaire de Webform.

De la même manière que Views, plusieurs modules ont été développés pour améliorer et ajouter des fonctionnalités à Webform. En voici quelques uns :

- ▶ **Webform Validation**, qui permet de mettre en place des systèmes de validation de champs poussés,
- ▶ **Webform Hints** qui ajoute la possibilité d'avoir des "placeholders" à la place des labels classiques,
- ▶ **Webform Conditional** offre le choix d'avoir des champs conditionnels en fonction de divers paramètres.

Gestion des images

Le cœur de Drupal inclut un système de recadrage et de modification des images, qui offre déjà des possibilités intéressantes. Il est configurable dans l'interface d'administration via

Configuration > Styles d'image. Chaque style a donc ses spécificités, et vous retrouverez ces modes d'affichages pour les images dans les champs des types de contenus, ou dans les modules comme Views.

Si le système de base de styles d'images vous limite au niveau des possibilités que vous aimeriez avoir voici deux modules qui vont élargir ses fonctionnalités :

- ▶ **Imagecache Actions** Fig.4 va ajouter des filtres au module du cœur de Drupal. Lorsque vous l'aurez activé, vous pourrez ajouter des effets de rotation, de filigrane sur les images, et encore des effets de redimensionnement poussés.
- ▶ **Image Style Quality** lui vous permettra de gérer la qualité des images qui sont altérées par un style. Cela permet de compresser les images pour des raisons de performances, mais aussi de choisir de ne pas avoir de compression, pour avoir une qualité optimale et un rendu visuel agréable.

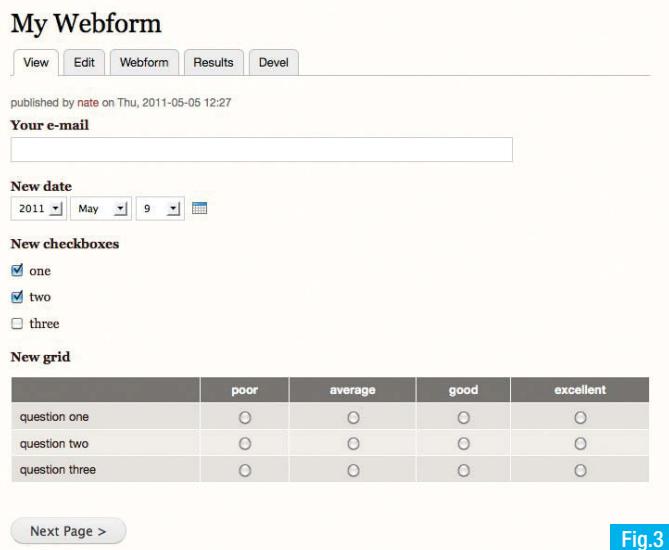
Types de champs

Si vous voulez élargir la possibilité au niveau des types de champs que vous pouvez créer avec Drupal, c'est possible de le faire avec ces modules :

- ▶ **Email** : fournit un champ de type courriel,
- ▶ **Date** : permet d'ajouter des champs de type date et heure,
- ▶ **Link** : créer des champs liens,
- ▶ **Address Field** : crée des champs d'adresses complets (adresse, ville, code postal, pays, état),
- ▶ **Phone** : champs pour les numéros de téléphone, ajoutant la validation de nombreuses syntaxes selon les pays.
- ▶ **Field Group** : Il permet de créer des groupes de champs.

Sauvegarde

Dans certains cas il peut être très utile de sauvegarder sa base de données, afin de pouvoir la restaurer : plantage du serveur, mise à jour de module ratée, mauvaise manipulation dans le backoffice, etc. C'est pourquoi je vous recommande d'installer le module **Backup & Migrate**, qui se chargera de faire des sauvegardes régulières et automatiques de votre



The screenshot shows the 'My Webform' page. At the top, there are tabs for View, Edit, Webform, Results, and Devel. Below the tabs, it says 'published by nate on Thu, 2011-05-05 12:27'. The form fields include 'Your e-mail' (a text input field), 'New date' (a date picker showing '2011 May 9'), 'New checkboxes' (a list with 'one' and 'two' checked, and 'three' unchecked), and 'New grid' (a table with three rows: 'question one', 'question two', and 'question three', each with columns for 'poor', 'average', 'good', and 'excellent').

Fig.3

base. Lorsque vous l'aurez installé, le module permet de paramétriser de faire des backups à intervalles réguliers, sauvegardés automatiquement sur le serveur et/ou envoyés par mail à une adresse spécifiée. Cela permettra donc de remettre sur pied votre site en cas de problème.

Mise en production

Lorsque votre site passe en phase de production, il est essentiel de vérifier quelques points de sécurité. Le module **security_review** liste ces points et vous alerte en cas de problème.

C'est parti !

Un dernier mot concernant l'installation des modules. Ils doivent être présents dans le répertoire "sites/all/modules". Nous vous conseillons vivement de bien distinguer les modules de la communauté et les modules spécifiques à votre site. La bonne pratique est de créer un répertoire "contrib" pour les modules de la communauté et un autre "custom" pour vos modules personnalisés. Cela facilite la maintenance.

Liste de liens utiles :

<https://drupal.org/documentation/install/modules-themes/modules-7>
<https://drupal.org/developing/modules/7>



Maxime Gaul,
Développeur Web /
Feel & Clic
www.feelandclic.com



Maxime Dunand,
Développeur Web /
Feel & Clic
www.feelandclic.com

Action	Settings
Define Canvas	left:4 right:4 top:4 bottom:4 #e5e8a1 under image
Define Canvas	left:4 right:4 top:4 bottom:4 Transparent under image
Overlay: file image to canvas (watermark)	corner-tl.png x:left, y:top alpha:100%
Overlay: file image to canvas (watermark)	corner-tr.png x:right, y:top alpha:100%
Overlay: file image to canvas (watermark)	corner-bl.png x:left, y:bottom alpha:100%
Overlay: file image to canvas (watermark)	corner-br.png x:right, y:bottom alpha:100%



Fig.4

Complétez votre collection

PROGRAMMEZ!

le magazine du développeur www.programmez.com

www.programmez.com



Prix unitaire : 6 € (Frais postaux inclus)

France métropolitaine uniquement.

nouveau

Tout **Programmez!** sur une clé USB

Tous les numéros de Programmez! depuis le n°100.



Clé USB 2 Go. Livrée dans sa boite. Testé sur Linux, OS X, Windows. Les magazines sont au format PDF.



29,90 €*



* tarif pour l'Europe uniquement. Pour les autres pays, voir la boutique en ligne

vous pouvez aussi commander directement sur notre site internet : www.programmez.com

- | | |
|--|---|
| <input type="checkbox"/> 168 : <u> </u> exemplaire(s) | <input type="checkbox"/> La Clé USB avec les numéros de Programmez depuis le n°100 |
| <input type="checkbox"/> 169 : <u> </u> exemplaire(s) | 29,90 € (Tarif pour Europe uniquement) |
| <input type="checkbox"/> 170 : <u> </u> exemplaire(s) | |
| <input type="checkbox"/> 171 : <u> </u> exemplaire(s) | |
| <input type="checkbox"/> 172 : <u> </u> exemplaire(s) | |
| <input type="checkbox"/> 173 : <u> </u> exemplaire(s) | |
| <input type="checkbox"/> 174 : <u> </u> exemplaire(s) | |
| <input type="checkbox"/> 175 : <u> </u> exemplaire(s) soit <u> </u> exemplaires x 6 € = <u> </u> € | + <input type="checkbox"/> 29,90 € soit au TOTAL = <u> </u> € |

Prix unitaire : 6 € (Frais postaux inclus), France métropolitaine uniquement.

M. Mme Mlle Entreprise : _____ Fonction : _____

Prénom : _____ Nom : _____

Adresse : _____

Code postal : Ville : Tarifs France métropolitaine

Tél : _____ (Attention, e-mail indispensable)

Règlement par chèque à l'ordre de Programmez !

Mon premier projet Drupal

C'est décidé votre projet web s'appuiera sur Drupal. Cependant vous ne connaissez pas encore l'outil et vous voulez partir si possible dans la bonne direction. Il est temps de comprendre comment Drupal fonctionne et comment vous allez l'adapter à votre projet. Pour cela il est nécessaire d'avoir une vision globale avant de se lancer.

Chaque CMS propose sa propre philosophie, ce qui implique que l'on ne peut pas appréhender un projet web de la même façon avec tous ces outils. Drupal propose une approche très structurée et flexible, mais nécessite aussi plus de temps pour maîtriser tous ses concepts. La courbe d'apprentissage est plus longue comparée à d'autres CMS, mais au final on a un outil souple, sur mesure et très puissant. Voyons quelles sont les briques de base du système. Il ne faut pas voir Drupal comme un système d'édition de page web, mais comme un gestionnaire de contenu et de règles de publication de contenu. Drupal applique ce principe à la règle, ce qui peut s'avérer déconcertant au départ car on a peu d'éléments pré-construits. L'image du Lego est bien adaptée : vous avez à votre disposition toutes sortes de pièces prêtes à être utilisées. C'est ensuite à vous de construire quelque chose avec. Dans le cas où il vous faut une pièce qui n'existe pas, alors vous avez différents outils pour la créer.

Pourquoi un site internet ? Cela peut apparaître comme étant une question simple mais la réponse nous donne beaucoup d'informations sur l'architecture que l'on va mettre en place. De façon très globale un site doit permettre de mettre en ligne du contenu (un article, une recette de cuisine...) depuis sa création via un formulaire jusqu'à son affichage sous forme de page web. On a un squelette fonctionnel et esthétique d'un côté, et du contenu de l'autre. Les utilisateurs ajoutent régulièrement du contenu via des formulaires et c'est ensuite Drupal qui gère ces contenus depuis la publication (avec workflow possible) jusqu'à l'affichage.

La vision d'un site web qui consiste à ajouter des pages à une arborescence existante est trop réductrice. Un site n'est pas une collection de pages : c'est un système complexe qui se construit automatiquement en fonction des contributions d'utilisateurs ou de contenus provenant de sources externes.

Lors de la conception d'un site sous Drupal, on réalise toujours un certain nombre d'étapes majeures. Ces dernières sont :

- ▷ La structure des données
- ▷ L'affichage / présentation des contenus
- ▷ Les fonctionnalités
- ▷ La définition des droits utilisateurs
- ▷ La charte graphique
- ▷ Le SEO
- ▷ Le déploiement

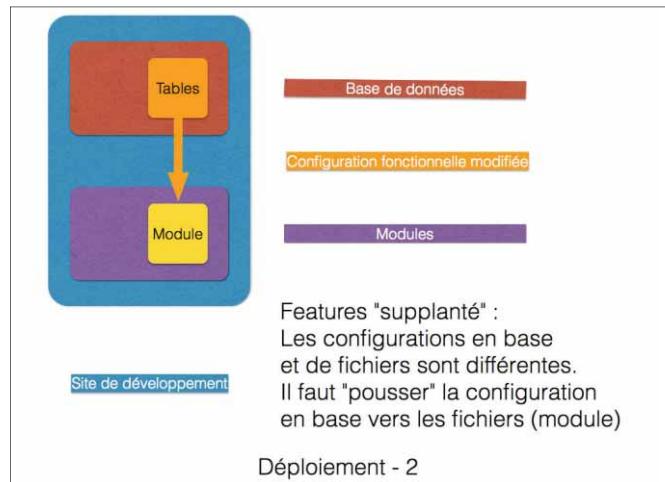
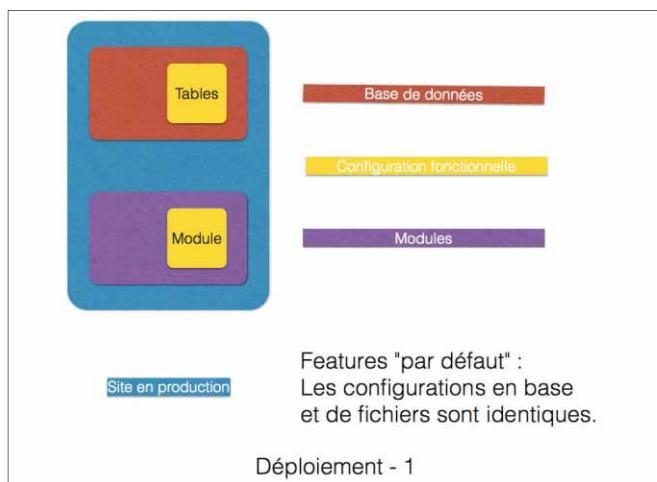
Regardons de plus près comment répondre à ces questions avec Drupal.

La structure des données

Comment structurer ses données ? Un CMS est en premier lieu centré autour des contenus. Il est nécessaire donc de se poser la question de la nature des données que l'on souhaite mettre en ligne. Un blog, par exemple, est un regroupement d'articles qui ont chacun un titre, un corps d'article et des images et vidéos. En revanche une GED (Gestion électronique de documents) a pour but de partager des documents créés en dehors du système, et de ce fait les contenus correspondants sur le site ne nécessitent sans doute qu'un titre et un fichier. Il est primordial de se poser la question "Quels types d'information veut-on mettre en ligne ?". C'est ainsi que l'on va construire petit à petit la

structure de données. Les contenus dans Drupal sont appelés "node". Chaque contenu est un noeud avec des propriétés et des champs que l'on peut ajouter simplement via l'interface du système. Tous les noeuds ont des informations communes : titre, auteur, date de création, date de modification, type, identifiant (nid)... On peut définir des familles de noeuds possédant chacune des champs spécifiques. Ces champs sont de types très variés comme par exemple un email, une adresse, ou bien des images. Chaque famille de noeuds diffère des autres par la nature des champs qu'elle utilise.

De base, Drupal ne propose que deux types de contenu : page de base et article. Ces types de noeuds sont évidemment modifiables et l'on peut surtout en ajouter autant que nécessaire. C'est l'une des forces de Drupal. Par contre cela demande de bien connaître le système afin de définir la structure de données adaptée aux besoins. Afin de pouvoir ajouter des champs à nos types de contenu, il vous faut installer les modules adéquats. Par exemple vous trouverez le module Address Field (sur drupal.org), qui définit les champs d'adresse (numéro, rue, ville, code postal...) en fonction du pays. Si vous souhaitez créer des tableaux de données (un peu comme Excel) vous pouvez utiliser le module TableField. Il existe des dizaines de modules définissant des types de champ particuliers. Pour chaque type de noeud il faut définir les caractéristiques de chaque champ; c'est à dire le type, l'apparence, la cardinalité... Il est préférable de bien mettre à plat la liste des champs nécessaires avant de se lancer dans l'interface de Drupal. Par exemple si l'on choisit un champ



de type fichier, il faut réfléchir à l'emplacement physique de ces fichiers; doit-on utiliser un répertoire par utilisateur ou un seul pour toutes les contributions ? Y a-t-il une limite de taille par fichier ? Il existe parfois des relations entre les types de contenus. Par exemple imaginons que l'on souhaite faire un site référençant les restaurants et hôtels en France. On crée deux types de contenus avec les champs adéquats. Mais dans le cas où un hôtel abrite aussi un restaurant, alors il serait plus intéressant de faire une relation entre les deux. Un hôtel peut faire référence à un restaurant. Cela permet de les lier directement. Ainsi si l'on recherche un hôtel, on saura que cet établissement possède ou non un restaurant, et inversement. On obtient finalement une structure de données relationnelle. Si vous souhaitez ce genre de fonctionnalité sur votre site, utilisez le module Entity References. Afin d'organiser le contenu de son site, Drupal propose un système appelé taxonomie. Ce n'est rien d'autre qu'un ensemble de mots-clés que l'on peut attribuer aux contenus (en réalité à toute entité comme les utilisateurs, les commentaires...). On pourrait imaginer ainsi sur une boutique en ligne que les produits soient organisés suivant différentes catégories. Tout produit est rattaché à une ou plusieurs catégories, permettant alors la recherche et le tri suivant ces critères. On peut ainsi organiser ses contenus autour de termes arbitraires.

La taxonomie est "fieldable"; on peut ajouter des champs comme sur les types de contenus ou les utilisateurs. Ceci est particulièrement pratique lorsque l'on souhaite décrire des termes avec autre chose qu'un champ texte et une description. Prenons l'exemple de la gestion d'un parc automobile. Chaque voiture est un noeud de type "Voiture". On crée alors un nouveau vocabulaire "Marque" contenant un terme par constructeur automobile. Ce vocabulaire est utilisé par le type de noeud "Voiture" et on y ajoute un champ de type image afin d'avoir le logo de chaque constructeur. Chaque voiture

appartient à une marque qui elle-même contient d'autres informations, comme ici le logo. Ainsi on a catégorisé les voitures en marque, mais on a également une description précise de ce qu'est une marque automobile.

L'affichage / présentation des contenus

A partir du moment où la structure de données est définie, il est nécessaire de se poser la question de l'affichage de ces données. Comment sont présentés les contenus aux utilisateurs ? Comment est organisée l'information ? Y a-t-il un système d'archives, un moteur de recherche ? Chaque page web (accessible via son URL) est découpée en différentes zones appelées "régions". On parle du zoning de page, c'est à dire des différents emplacements physiques qui affichent du contenu. On a par exemple typiquement 5 grandes régions : entête, 3 colonnes et pied de page. Evidemment il n'y a pas de limite dans le nombre de régions et leurs positionnements. Ce zoning est la première étape dans la mise en page du site.

Au niveau inférieur il existe le système de blocs permettant d'alimenter les régions en contenus. Chaque bloc doit être positionné dans une région. Ainsi on construit son site en créant des blocs abritant du contenu (de façon générique) que l'on assigne ensuite aux régions. Chaque bloc peut proposer du contenu très varié : du code HTML, une liste de noeuds, les derniers utilisateurs enregistrés sur le site... Quel que soit le contenu du bloc, il est nécessaire de l'assigner à une région. Aucun bloc n'apparaît dans une région par défaut.

Drupal propose en standard un certain nombre de pages pour le front-office. On a à notre disposition une page d'accueil, une page dédiée pour chaque noeud, utilisateur, terme de taxonomie, et une page de maintenance (lorsque le site est hors-ligne). Par ailleurs toute l'administration du site se fait via un grand nombre de pages côté back-office.

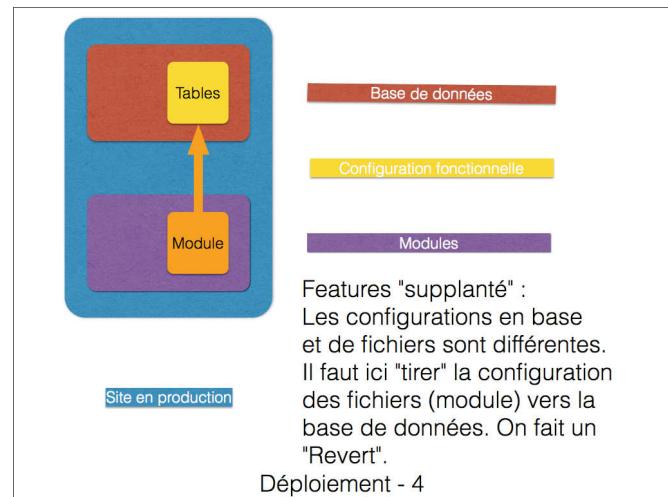
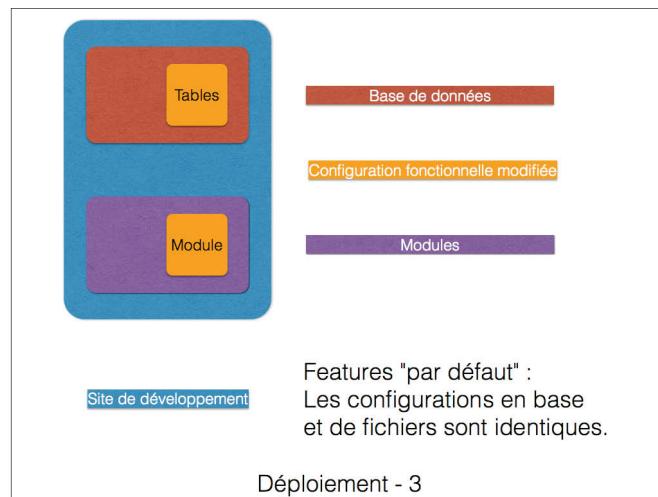
Ces différentes pages sont des pages par défaut que l'on peut personnaliser (même dans le back-office). Mais bien souvent elles ne suffisent pas et on est alors amené à créer ses propres pages. On peut diviser les pages en 2 catégories : pages statiques et pages dynamiques. Une page de "Mentions légales" est considérée comme statique car son contenu n'évolue pas dans le temps (ou alors très rarement) alors qu'une page de listing de produits est dynamique puisque l'on souhaite la faire évoluer automatiquement en fonction de l'arrivée de nouveaux produits.

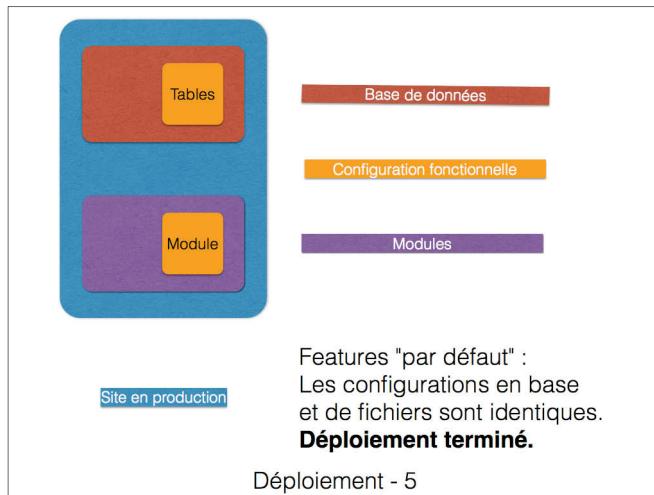
Les pages statiques peuvent être facilement créées en utilisant les noeuds : par exemple pour la page des "Mentions légales", on ajoute un noeud avec comme titre celui de la page désirée, et comme corps, le contenu de la page. Automatiquement si le noeud créé a pour identifiant NID, la page correspondante est accessible au chemin node/NID.

Les pages dynamiques peuvent être créées de différentes manières : utilisation de modules contribués (c'est à dire disponibles sans limites sur drupal.org) ou en codant soi-même ses pages dans un module dit "custom".

Le plus souvent un site consiste principalement en un ensemble de pages listant du contenu. On souhaite par exemple avoir la liste des derniers articles en page d'accueil, la liste des contributions pour chaque utilisateur enregistré sur sa page de profil, mais aussi, pour un contenu, la liste des autres contenus traitant du même sujet. On peut imaginer toutes sortes de listes utilisant des affichages aussi divers qu'un calendrier, une carte interactive ou une animation type slideshow.

La plupart des listes dont on a besoin peuvent être créées à la souris avec le module Views. C'est le module le plus puissant pour les site builders. Il permet de créer des pages ou des blocs listant à peu près tout ce que l'on a en base : noeuds, utilisateurs, terme de taxonomie... En pratique une liste créée avec Views





est appelée une vue. Tout le contenu de la vue est personnalisable; on définit en premier lieu les types d'éléments que l'on souhaite lister (nœuds, utilisateurs...) puis on renseigne comment chacun de ces éléments doit apparaître (avec par exemple son titre, une image ou toute autre propriété ou champ). On a en plus à disposition des critères de filtre et de tri. On peut ainsi faire remonter dans la liste que des éléments qui ont des caractéristiques bien précises dans un ordre déterminé. Il existe de très nombreux réglages disponibles de base, et il est également possible d'ajouter d'autres fonctionnalités via des modules se greffant sur Views.

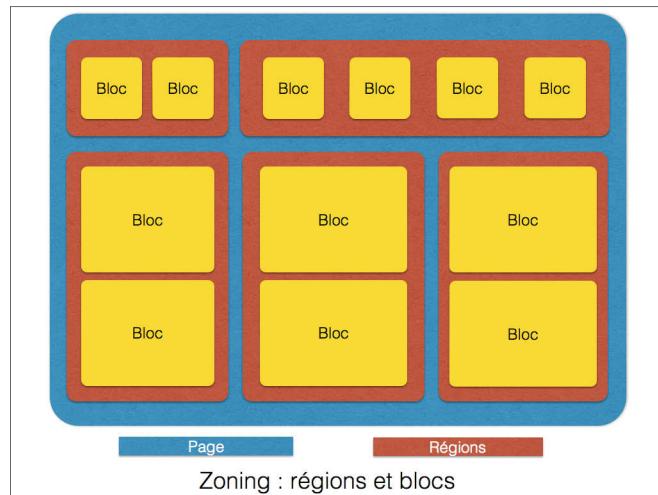
Les fonctionnalités

Jusqu'à présent on a découvert les principes de base indispensables. Cependant comme tout CMS on peut étendre les fonctionnalités du cœur en ajoutant des modules. Tous les sites Drupal partagent la même base et la différence est faite au niveau du choix des modules. Drupal est de ce fait un système très flexible permettant de créer des sites de toutes sortes : site vitrine, réseau social, intranet, e-boutique... C'est la force de la communauté de développeurs qui sont à l'origine du très grand nombre de modules disponibles. Un peu comme un magasin d'applications pour smartphone, on peut dire avec Drupal "il y a un module pour ça !". Il existe de fait des milliers de modules permettant de couvrir un très grand nombre de besoins : SSO, newsletter, QR code, interfaçage avec d'autres systèmes, fonctionnalités communautaires... La liste est presque infinie. Par contre cette multitude de modules n'est pas forcément un avantage pour les novices. Il est difficile de choisir le "bon" module répondant précisément à un besoin. Seule l'expérience permet de tirer partie d'un écosystème aussi foisonnant. Malgré la richesse de modules disponibles on est souvent amené à développer ses propres fonctionnalités sous forme de module custom. Il est alors nécessaire de très bien connaître le système et les modules disponibles, afin d'être

capable de déterminer dans quels cas on va se tourner vers les modules de la communauté plutôt que vers un développement spécifique.

La définition des droits utilisateurs

Le contrôle d'accès proposé par Drupal est à la fois intuitif, simple et efficace. Le système définit un ensemble de permissions (par exemple administrer les utilisateurs, voir le contenu publié, accéder au site en mode maintenance...) et des rôles. Configurer les accès consiste à associer des permissions à chacun des rôles (anonymes, personnes ayant un compte, contributeurs, webmaster, administrateur...). Ensuite chaque utilisateur fait partie d'un ou plusieurs rôles et héritera des permissions correspondantes. En pratique il faut décrire les rôles de chacun dans la "vraie vie" et en déduire les rôles nécessaires sur le site. De façon générale on peut reproduire la structure de l'entreprise, de l'institution ou de l'association au sein de Drupal. On obtient ainsi une hiérarchie naturelle de rôles/personnes, car correspondant à celle existante. Il est ainsi possible d'obtenir parfois jusqu'à 20 ou 30 rôles différents sur un même site. Une fois la liste des rôles établie, il faut leur attribuer les permissions souhaitées. Cette étape n'est en rien compliquée mais nécessite des tests assez longs à mettre en place. Sur un site "classique", il est courant d'avoir 200 ou 300 permissions et entre 5 et 10 rôles. Le nombre de combinaisons est alors très important, entraînant de multiples tests. C'est primordial de bien configurer les rôles, car on peut se retrouver sinon avec des situations de blocage inattendues. On porte une attention particulière à l'accès aux contenus car ils peuvent être sensibles. Les permissions de base sont : "Voir le contenu publié", "Créer", "Modifier", "Modifier ses propres contenus", "Supprimer", "Supprimer ses propres contenus", et ce pour chaque type de contenu. On note au passage qu'il est possible d'avoir le droit de créer du contenu, sans pour autant avoir le droit de le voir. Pour aller plus loin il est possible d'avoir des règles d'accès au niveau de chaque champ, et



de chaque utilisateur. On peut ainsi définir non seulement qui accède à quels contenus, mais également qui accède à quelle information du contenu. Par exemple, si j'ai un site de vente en ligne, je peux décider que certains utilisateurs seront en charge de créer et mettre à jour des fiches produit, mais il ne pourront pas définir le prix de ces produits. Le champ correspondant n'est accessible en écriture que par des utilisateurs en charge de la politique tarifaire.

La charte graphique

Le thème est responsable de l'affichage à l'écran du site. Tout ce que l'on voit dans un navigateur dépend du thème. Il n'y a pas de limites avec Drupal : la seule contrainte est de savoir créer un thème.

Le designer fournit une charte graphique qui doit au final toujours être intégrée sous forme de thème Drupal. Pour ce faire il y a deux approches différentes :

- le designer fournit les maquettes à un intégrateur qui crée les fichiers html, CSS et JS, puis un themeur transforme ces fichiers en thème utilisable par Drupal.
 - le designer fournit les maquettes à un intégrateur qui crée directement un thème Drupal. Le choix de l'une ou l'autre de ces deux approches dépend beaucoup des compétences des intervenants. Il est cependant plus rapide d'intégrer directement la charte graphique fournie par le designer sous forme de thème.
- Il est impossible d'ignorer actuellement le Responsive Web Design. L'idée est d'adapter la présentation d'un site en fonction de l'appareil de consultation. Le designer est donc amené à produire différents gabarits pour chaque page. Les contraintes de temps (et donc de budget) ne sont donc plus les mêmes. De nombreux thèmes comme Omega, Zen ou Bootstrap permettent de gagner du temps, mais ce ne sont pas non plus des solutions miracle. Le travail du themeur est de plus en plus complexe. Au-delà de la mise en page, il est possible d'optimiser la bande passante en allégeant les contenus et le design (moins d'images ou des

images moins lourdes). Ce sont des contraintes techniques plus que graphiques, mais qui doivent être intégrées au design global. Dans tous les cas, pour proposer du contenu adapté à l'appareil, il convient d'avoir un mécanisme de détection des caractéristiques de l'appareil (les médias queries en sont un, mais pas le seul). A ce moment les développeurs doivent intervenir de façon à ce que le contenu affiché dépende de l'écran de visualisation. En allant encore plus loin, il est possible d'adapter le design en contextualisant les conditions locales (éclairage, qualité de connexion...).

Le SEO

Afin d'améliorer le référencement, on peut installer de nombreux modules qui vont permettre d'optimiser des points bien précis. Il n'est pas nécessaire de tous les utiliser d'autant plus que le référencement naturel de Drupal est plutôt bon. Le point de départ est souvent le module SEO Checklist qui recense l'ensemble des points à surveiller en précisant à chaque fois le module concerné. Par exemple le module Pathauto permet de réécrire les URLs en fonction des caractéristiques du contenu affiché (titre, date de publication, auteur...).

Toutes les améliorations de base (fichier robots.txt, réécriture d'URL, duplicate content, metatags...) sont traitées via des modules dédiés. On peut ainsi améliorer son référencement de manière très précise et donc adaptée à ses besoins.

Notons que le SEO n'est pas toujours utile; par exemple dans le cas d'un intranet, on n'est pas indexé par les moteurs de recherche donc on n'a pas besoin d'optimiser son référencement !

Le déploiement

Un site Drupal est constitué des fichiers du cœur et des modules (contribués et custom), ainsi que de la base de données. Lorsque l'on souhaite sauvegarder l'intégralité de son site il suffit de faire un dump de la base et de tous les

fichiers. Imaginons que le site soit en ligne depuis un certain temps et que l'on souhaite le faire évoluer fonctionnellement. Il est hors de question de travailler sur l'instance en production, il faut donc l'installer sur un autre environnement, par exemple sur une machine locale. Pour ce faire on récupère les fichiers et la base en production et on réinstalle le tout sur une autre machine. Lorsque les nouveaux développements sont achevés, comment peut-on alors les passer en production ? Si l'on repart de notre installation de développement, alors on perd tous les changements intervenus entre temps sur la production (par exemple les nouveaux contenus, commentaires, utilisateurs...). Dans le cas où l'on a travaillé qu'avec des fichiers communs aux deux instances, alors on peut mettre en production les nouveaux qui remplaceront les anciens. Mais comment faire lorsque la configuration du site a évolué ? Il est impossible de faire une fusion entre la base de données en production et celle en développement. A partir du moment où l'on active de nouveaux modules ou bien que l'on modifie les paramètres de ceux déjà présents, tous les changements sont stockés en base.

Le module Features est la solution privilégiée pour les déploiements. Il permet d'exporter de la configuration du site en base vers des fichiers. En réalité les fichiers exportés composent un module Drupal que l'on peut alors installer en production et ainsi récupérer la configuration correspondante.

Si l'on a un site en production que l'on veut faire évoluer sans pertes, on commence par exporter avec Features les composants que l'on souhaite modifier. En pratique on sélectionne par exemple un type de contenu, une ou plusieurs vues, des styles d'image etc... Features crée un module contenant la configuration correspondante. Ce module doit être installé sur les sites de production et de développement. Features compare la configuration stockée sous forme de fichier (module) et celle en base. Initialement

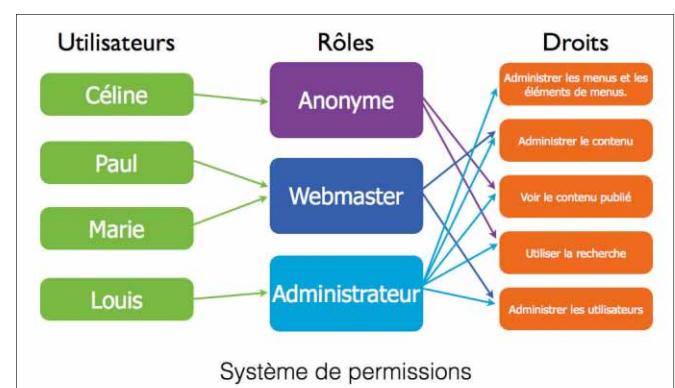
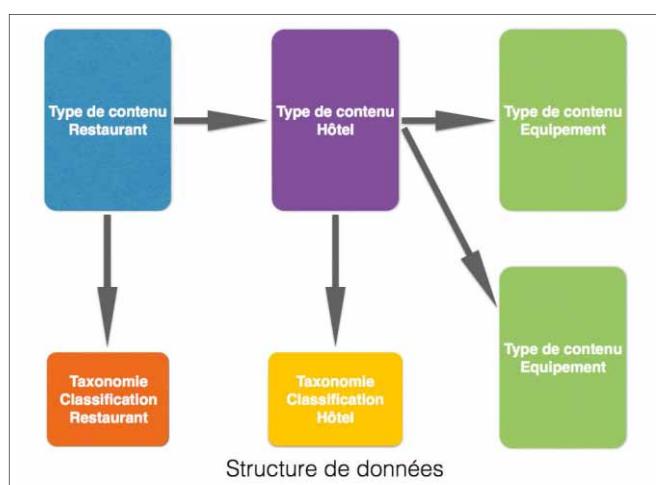
"Par défaut". Ensuite si l'on fait des changements sur notre instance de développement, Features nous indique un état "Supplanté", ce qui signifie que la configuration en base est différente de celle stockée dans notre module. Supposons alors que les modifications soient terminées. Il faut alors recréer le module qui va contenir notre nouvelle configuration. Cette nouvelle version du module doit être installée localement sur notre site de développement. Nous sommes alors dans l'état "Par défaut". Pour transférer nos changements en production, on met à jour notre module en remplaçant les anciens fichiers par les nouveaux. Features indique alors un état "Supplanté". Il faut donc maintenant récupérer la configuration contenu dans notre module pour mettre à jour celle en base. C'est l'opération "Revert" de Features. Une fois fini nous sommes de nouveau dans l'état "Par défaut" sur la production, et tous nos changements ont été pris en compte. Le module Features est extrêmement utile et populaire et permet de presque tout exporter ce que l'on a en base au niveau de la configuration. Bien sûr certains modules que vous pouvez télécharger sont transparents pour Features et il faut alors trouver d'autres solutions. Cependant il est possible pour les développeurs d'utiliser l'API proposée et ainsi de devenir Features compatible.

Pour conclure

Drupal est définitivement de plus en plus destiné à des développements Web complexes. On peut toujours l'utiliser pour faire un blog, mais c'est surtout le public professionnel qui tire partie de sa flexibilité et de sa richesse fonctionnelle. Drupal est à la fois un CMS et un CMF (Content Management Framework). Il peut donc satisfaire aussi bien un sitebuilder qui travaille principalement à la souris qu'un développeur qui ne jure que par le code. C'est bien là la force du système.

● Romain Jarraud

Responsable Formation / Directeur Technique - Trained People



Les thèmes / theming

Avoir son site Internet est aujourd’hui un outil de marketing indispensable et une très bonne source de visibilité pour une organisation. De ce fait, il est primordial d’avoir un site Internet aux couleurs de l’entreprise ou de l’association qu’il présente. Dans le cas d’un site basé sur le CMS Drupal, il est donc essentiel de créer ou adapter un thème qui reprend la charte de l’organisation afin d’avoir une cohérence avec le reste des supports de communication.

Les bases de l'anatomie d'un thème

Un thème Drupal est là pour structurer votre design et son contenu. Il y a donc la possibilité de définir des régions (header, navigation, contenu, footer...), de placer vos fichiers CSS, javascript, images et de créer votre structure HTML via des fichiers de templates. On peut aussi pousser la personnalisation en agissant directement sur le rendu à l'aide de certaines fonctions de pré-traitement.

Pour commencer, tout thème customisé doit se trouver dans le dossier sites/all/themes/ de Drupal. Le nom du dossier sera ensuite le nom de notre thème.

Deux fichiers sont essentiels à la création d'un thème. Le premier d'entre eux est le fichier .info. Placé à la racine du répertoire de votre thème, c'est lui qui définira les régions, et qui va lier les fichiers CSS et Javascript au thème. Sa syntaxe est relativement simple, en voici un exemple :

```
name = Nom du theme
description = Description du thème

stylesheets[all][] = css/style.css
stylesheets[print][] = css/print.css

scripts[] = js/script.js

regions[header] = Header
regions[content] = Content
regions[footer] = Footer
```

Le deuxième fichier est page.tpl.php. C'est dans ce fichier que la structure HTML principale et l'agencement des régions de votre thème va être initiée. Grâce à des fonctions PHP Drupal il est possible dans ce fichier de récupérer tout le contenu d'une région. Par exemple pour récupérer le contenu de la région "content" :

```
<?php print render($page['content']); ?>
```

Notons que les fichiers à l'extension .tpl.php sont des fichiers de template Drupal, et qu'ils peuvent être utilisés pour définir la structure de nombreux éléments. Ils respectent une convention de nommage bien précise. Voici quelques exemples :

- node.tpl.php : Altère l'affichage d'un node
- maintenance-page.tpl.php : Altère la page de maintenance
- field-[nom_du_champ].tpl.php : Altère l'affichage d'un champ

Attention, lors de la modification du fichier.info du thème, ou lors de la création d'un fichier de template .tpl.php, il est nécessaire de vider le cache de votre Drupal pour prendre en compte ces modifications (dans l'administration : Configuration > Performances > Vider le cache).

Arborescence

Tous les fichiers peuvent être placés à la racine de votre thème, cela ne créera pas de problème fonctionnel. Cependant, pour avoir un thème

"propre" nous vous recommandons de l'organiser en sous répertoires : css, js, templates, etc. Cela est utile et vous aidera lorsque vous commencez à avoir un thème qui comporte beaucoup de fichiers.

Méthodes de theming

Plusieurs méthodes sont possibles lors de la création d'un thème. Chacune comporte des avantages et des inconvénients principalement liés au temps de développement et aux performances. Voici les principales techniques qui vous permettront de créer votre thème.

From scratch

Certainement la méthode la plus propre. Le but est de partir de zéro, ce qui nous garantit un thème parfaitement adapté à nos besoins, sans fonctionnalités ou parties dont vous n'auriez pas besoin. L'inconvénient de cette méthode est la perte de temps que vous pourriez avoir à re-commencer un thème en entier.

Modification de thèmes existants

Une autre méthode possible est la modification d'un thème "complet".

Prenons l'exemple du thème Drupal 7 de base, Bartik.

Dans le cadre de cette méthode, nous allons "duplicer" le thème Bartik présent par défaut dans Drupal dans le répertoire "themes", en le copiant dans le répertoire "sites/all/themes". Il faut ensuite renommer le thème. Pour cela, plusieurs étapes :

- Renommer le répertoire "sites/all/themes/bartik"
- Renommer le bartik.info
- Editer le fichier.info et changer le nom du thème (name = ...)
- Editer le fichier template.php et changer tout les "bartik"

Voilà, vous avez maintenant votre thème "clone" de Bartik, que vous allez pouvoir activer dans l'interface d'administration Drupal dans l'onglet Apparence.

Cette méthode de création de thème à l'avantage d'être rapide si vous n'avez pas d'exigences très poussées au niveau du graphisme, par exemple si vous voulez seulement adapter les couleurs, la police d'écriture et quelques tailles de texte. Les éléments des pages (tableaux, formulaires, etc.) sont déjà existants ce qui vous permet d'avoir un thème complet en très peu de temps.

Theme starter

Plusieurs thèmes *starters* existent. Ce terme désigne les thèmes destinés à servir de base de modification pour les personnes désirant créer leur propre thème. Ils n'ont aucune base graphique, mais comportent les fichiers de base et des options de configuration directement administrables via l'interface, comme par exemple des options liées au responsive, ou aux couleurs du site. Certains sont compatibles avec les méthodes de pré-traitement CSS avancées comme Sass ou Less.css.

Cette méthode est l'alternative entre les deux méthodes ci-dessus.

Parmi les plus connus citons Zen, Omega, Adaptive Theme et Basic. Tous sont développés par la communauté et ont l'avantage d'être mis à jour et améliorés régulièrement.

Téléchargement de thèmes

À travers une rapide recherche sur Internet vous pourrez trouver une multitude de sites qui proposent des thèmes pour Drupal, dont certains sont très bien. Le site Drupal.org propose lui aussi une base de données de plus de 1900 thèmes. Nous recommandons de télécharger vos thèmes en priorité sur Drupal.org pour des questions de sécurité. L'inconvénient est que le téléchargement de thème comporte le risque de trouver exactement le même design sur un autre site, voire sur le site de votre concurrent. De plus, ces thèmes même s'ils sont pour la plupart personnalisables, ne seront jamais à la hauteur d'un thème fait sur mesure, adapté à votre besoin. Quelques sites sont également spécialisés dans la revente de thèmes de CMS, qui sont généralement plus exclusifs que les gratuits. Attention cependant à la récupération de thème, qui présentent un sublime aperçu lors de la vente, mais qui sont parfois durs à administrer et à prendre en main pour y mettre son contenu.

Pour résumer ...

Si vous avez peu d'exigence graphique et que votre temps est limité, je vous conseille de partir sur la modification d'un thème existant. Si vous avez beaucoup de temps à consacrer à votre thème et que vous désirez apprendre comment créer un thème de A à Z, alors n'hésitez pas, commencez un thème à partir de zéro. Si vous avez peu de temps pour votre projet mais que vous désirez partir sur une base propre et que vous voulez un site 100% à votre image, commencez à partir d'un thème starter. Si vous n'avez pas du tout de temps pour votre thème et que vous désirez quand même avoir un site professionnel, alors le téléchargement ou l'achat d'un thème se révèle être la meilleure solution.

Aide au theming

Modules

Si vous êtes intégrateur, alors voici quelques modules qui pourront grandement vous aider dans le développement de votre thème :

- ▷ **Style Guide** : Ce module va ajouter une page à votre site Drupal, présentant tous les types d'éléments que votre site pourra afficher, comme les titres, les tableaux, les formulaires, les listes à puces, les styles d'images, les paginations, etc. C'est très utile lorsque vous créez un thème pour avoir une vue d'ensemble sur l'intégralité des éléments, et ainsi les intégrer au plus vite. Grâce à ce module, vous n'avez aussi pas le risque d'oublier d'intégrer un élément.
- ▷ **Block class** : Grâce à ce module vous allez avoir la possibilité d'ajouter des classes aux blocs Drupal
- ▷ **Menu attributes** : Sur le même principe, vous allez pouvoir gérer les attributs d'un menu, et donc leur ajouter des classes également.

Liste de liens utiles

Si vous désirez aller plus loin dans la création et la customisation de thème Drupal, voici quelques liens qui peuvent vous être utiles :

- ▷ <https://drupal.org/node/177868> : Créer une page de paramétrage pour votre thème
- ▷ <http://www.ostraining.com/blog/drupal/views-templates/> : Les fichiers templates du module Views
- ▷ <https://drupal.org/node/1089656> : Le "Template suggestion" ou comment créer/modifier des fichiers tpl.

○ Maxime Gaul,
Développeur Web / Feel & Clic
[www.fealandclic.com](http://www.feelandclic.com)

○ Maxime Dunand,
Développeur Web / Feel & Clic
www.fealandclic.com

LA BOÎTE À OUTILS FEEL & CLIC

Voici 6 thèmes que nous vous recommandons pour Drupal 7.

Bootstrap

- ▷ Inclut le bootstrap de Twitter directement dans Drupal.
- ▷ Avantages : Design de bootstrap de base/Reponsive /Sous-thème facile à mettre en place
- ▷ Inconvénients : Beaucoup d'éléments surchargés pour être compatible avec Bootstrap



Corporate Clean

- ▷ Design de base clair et simple.
- ▷ Avantages : Responsive, plugins jQuery inclus, beaucoup de régions disponibles.
- ▷ Inconvénients : Beaucoup d'options qui peuvent désorienter l'utilisateur débutant.



Bartik

- ▷ Thème de base Drupal 7
- ▷ Avantages : Complet, léger, rapide à customiser.
- ▷ Inconvénients : Design non actuel, non responsive.



Zen

- ▷ Thème starter
- ▷ Avantages : Léger, HTML5, Responsive.
- ▷ Inconvénients : long à customiser



AdaptiveTheme

- ▷ Thème starter
- ▷ Avantages : Très configurable, HTML5, Responsive.
- ▷ Inconvénients : long à customiser



Omega

- ▷ Thème starter
- ▷ Avantages : Très configurable, HTML5, Responsive.
- ▷ Inconvénients : long à customiser



Multilangue sous Drupal 7

Drupal 7 offre un très grand nombre de possibilités de traductions du contenu et des interfaces. Ainsi, il est possible de traduire tous les éléments d'une page dans toutes les langues.

LE MULTILANGUE PAS À PAS

Installation

Deux modules permettent de gérer la traduction d'un site de façon optimum.

► **Internationalization (i18n)** est le module incontournable pour la traduction d'un Drupal 7, il est le module de base pour concevoir une interface multilangue saine.

► **Localization Update (l10n)** permet de mettre à jour les traductions des différents modules installés sur le site de façon à récupérer la traduction officielle réalisée par la communauté.

La plupart des modules connus issus de la base des contributeurs (<http://www.drupal.org>) s'interfacent très bien avec le module i18n, ce qui offre un nombre de possibilités très nombreuses pour la traduction de sites.

Configuration

Vous avez installé les modules requis, vous allez devoir configurer ces modules afin qu'il répondent au maximum à vos besoins. Pour cela, nous allons détailler plusieurs phases de la configuration linguistique.

Ajout d'une nouvelle langue

L'ajout d'une nouvelle langue peut être effectué en quelques secondes seulement. Il suffit de naviguer dans *Configurations > Régionalisation & Langue > Langues > Ajouter une langue*. On nous propose alors de sélectionner une langue pré-configurée ou d'en définir une. Vous venez d'ajouter une nouvelle langue (en quelques secondes). Si vous avez au préalable ajouté le module Localization Update, alors il va se charger de télécharger automatiquement les traductions des modules déjà installés. Si c'est la première nouvelle langue que vous ajoutez, vous allez devoir configurer vos affichages pour qu'ils prennent en compte les contenus de cette nouvelle langue (cf. partie 3).

Définir une langue par défaut

Lorsque vous ajoutez une langue vous pouvez la choisir en tant que langue par défaut. Il suffit de choisir une langue dans l'interface qui liste les langues disponibles sur le site (*Configuration > Régionalisation et Langue > Langues*) Fig.1.

Selectionner la bonne langue pour un utilisateur lambda

Il est possible de choisir plusieurs moyens de

NOM ANGLAIS	NOM NATIF	CODE	DIRECTION	ACTIVÉ	PAR DÉFAUT	ACTIONS
French	FR	fr	De gauche à droite	<input checked="" type="checkbox"/>	<input type="radio"/>	modifier
English	EN	en	De gauche à droite	<input checked="" type="checkbox"/>	<input type="radio"/>	modifier

Fig.1

MÉTHODE DE DÉTECTION	DESCRIPTION	ACTIVÉ	ACTIONS
URL	Déterminer la langue à partir de l'URL (préfixe de chemin ou nom de domaine).	<input checked="" type="checkbox"/>	Configurer
Session	Déterminer la langue à partir d'un paramètre de requête ou de session.	<input type="checkbox"/>	Configurer
Utilisateur	Suivre la préférence linguistique de l'utilisateur.	<input type="checkbox"/>	
Navigateur	Déterminer la langue à partir des paramètres de langue du navigateur.	<input checked="" type="checkbox"/>	
Par défaut	Utiliser la langue par défaut du site (FR).	<input type="checkbox"/>	

Fig.2

détection pour la langue de l'utilisateur. Voici les différentes options disponibles :

- **URL** : permet de déterminer la langue que l'utilisateur souhaite voir s'afficher grâce au préfixe de chemin de l'url ou du nom de domaine. C'est généralement le cas lorsque que l'on fait un site multilangue.
- **Session** : permet de déterminer la langue de l'utilisateur à partir d'un paramètre de requête ou de session.
- **Utilisateur** : si vous avez permis à l'utilisateur de choisir la langue qu'il souhaite utiliser (cf. ci-dessous), cette option permettra de suivre ses préférences linguistiques.
- **Navigateur** : détermine la langue de l'utilisateur à partir des paramètres de langue définis au sein du navigateur.
- **Par défaut** : utilise la langue par défaut du site (généralement anglais).

Il est possible d'ordonner ces options selon votre choix. Cela aura pour résultat de définir minutieusement le choix de la langue de l'utilisateur Fig.2.

Permettre aux utilisateurs de choisir leur langue par défaut

Comme nous avons pu le voir ensemble, Drupal 7 est très abouti en terme de possibilités pour la traduction de contenus.

Il permet en outre certaines fonctionnalités sympas pour l'utilisateur.

En effet un utilisateur peut choisir ou non la langue qu'il souhaite avoir sur l'interface lorsqu'il est connecté, un peu comme ce que l'on retrouve sur n'importe quel site moderne ou réseau social.

Pour activer cette fonctionnalité, il suffit de se rendre dans "Configurations > Régionalisation & Langue > Paramètres régionaux" et de cocher la case "Les utilisateurs peuvent définir leur propre fuseau horaire".

Afficher un bloc de sélection de langue

Il est parfois intéressant de pouvoir laisser choisir l'utilisateur en ce qui concerne la langue qu'il souhaite utiliser dans la navigation sur le site. Il

existe plusieurs modules disponibles dans la communauté permettant d'afficher une navigation qui propose de basculer d'une langue à une autre. Aujourd'hui, nous allons nous concentrer sur un module en particulier : **Language Switcher**.

Ce module permet en effet d'afficher un sélecteur de langue afin de pouvoir changer d'environnement linguistique.

Concevoir les affichages

Utilisation de Views

Lorsque vous utilisez le module Views et que vous souhaitez concevoir des affichages de contenus multilingues, il y a quelques notions à apprendre et comprendre. Tout d'abord, si aucun filtrage n'est fait, les contenus de toutes les langues s'affichent par défaut. Il ne faut donc pas oublier d'ajouter ce filtre.

Pour l'ajouter, ajouter un nouveau filtre et utilisez le filtre "Contenu : Langue". Vous pouvez ensuite le configurer en fonction de vos besoins : langue courante de l'utilisateur, langue par défaut du site, une langue en particulier, ou aucune langue. Vos affichages prendront donc en compte le contexte dans lequel ils sont pour remonter le contenu.

Avec Context

De la même façon qu'en utilisant le module Views, il est possible de filtrer l'affichage de blocs sur certaines pages en fonction de la langue de l'utilisateur, du site, ou une langue en particulier.

Avec des modules custom

Lorsqu'on intitule la conception d'un module de zéro (from scratch), il est important, même primordial, d'utiliser la fonction t() mise à disposition par Drupal afin de traduire tout contenu des pages, blocs et affichages construits et gérés dans ce module.

Pour aller un peu plus loin, la fonction t() accepte des token qui permettent de mettre les variables dans les textes.

Ainsi, il est extrêmement facile de retrouver une portion de texte à traduire directement dans le Backoffice.

Traduire l'interface

Traduction des contenus

Pour pouvoir traduire un contenu sous Drupal, il faut d'abord déclarer son type de contenu ou son vocabulaire comme traduisible. Pour ce faire, il faut aller dans la configuration du type de contenu désiré, et choisir l'option "Traductions : Activée, avec traductions". Une fois ceci effectué, vous trouverez un nouvel onglet "Traduire" lors de l'édition d'un node. Cliquez dessus, et choisissez la langue dans laquelle traduire le contenu. Chaque traduction de contenu crée un nouveau node indépendant; les nodes sont reliés techniquement grâce à un "set de traduction".

Traduction des blocs

La traduction d'un bloc s'effectue aussi facilement que la traduction d'un contenu.

Traduction des menus

Tous les menus d'un site peuvent être traduits dans toutes les langues disponibles au sein de Drupal, et cela très facilement. Allez dans *Structure > Menus > Modifier le menu* que vous avez créé. Deux possibilités s'offrent à vous :

- Traduire et localiser,
- Langue fixe.

Traduction d'un champ entity de type liste

Pour traduire une liste, on peut se baser soit sur une taxonomie, dans ce cas, on traduit directement la taxonomie comme dans la partie traduction de contenu, soit sur une liste basique, lorsque l'on créé le champ, il est possible de traduire toutes les entrées de ce champ dans l'onglet "Traduire". Enjoy!



● Jérémie Allèbe,
Développeur Web /
Feel & Clic
[www.fealandclic.com](http://www.feelandclic.com)



● Chengbo Zhou,
Développeur Web /
Feel & Clic
www.fealandclic.com

100% nouveau,
100% Cloud



CLOUDMAGAZINE.FR

100 % cloud computing

Microsoft tech-days 2014

11,12,13 février au Palais des Congrès de Paris

Je m'inscris

ACTUALITES AVIS D'EXPERT ANALYSES

A la une

SAP : le cloud soutient l'éditeur en 2013
SAP vient d'annoncer les résultats pour le 4e trimestre 2013 et pour l'ensemble de l'année 2013. Le cloud et les services liés ont aidé l'éditeur à se maintenir. Globalement, SAP a généré un chiffre d'affaire de + 5 milliards € pour le dernier trimestre. Les souscriptions cloud / saas pèse 209...

CloudUnit : héberger et déployer des applications Java
un nouveau service Paas Java sera disponible dans les prochains mois : CloudUnit. Il s'agit d'un projet français pour héberger et déployer des applications Java et JEE. Les principales fonctions annoncées sont : montage des environnements Java en quelques minutes, choix technologiques standards (...)

Avis d'expert

Mesurer entre les lignes
Le réseau de compteurs intelligents 4-G sans liens physiques proposé par le Royaume Uni forme une toute nouvelle infrastructure d'importance critique : en tant que telle, elle aura besoin de procédures de tests de haut vol, ...

Analyse

IBM investit dans le Cloud : 1 milliard
Quand les grands acteurs investissent dans le Cloud Computing cela fait mal, très mal. En France, on parle, pour les plus gros projets, de 150 - 200 millions. Quand

Sécurité

Cloud Computing et la sécurité
Les menaces changent avec le Cloud. Cloudwatt apporte sa vision, son analyse. vidéo de 32 minutes.

Cloud Computing et nouvelles menaces

00:00 / 32:33

Newsletters
[Inscrivez-vous](#) à nos newsletter !

Platform.sh : une nouvelle plate-forme d'hébergement et développement pour Drupal

Mon premier site Drupal remonte à 2004. Depuis, j'ai connu pas mal d'autres langages que PHP, je suis un grand fan de Rails, j'aime Erlang pour ses artefacts de concurrence. Mais, Drupal reste adapté à des cas d'usage sur lesquels il est simplement imbattable, que ce soit pour de l'e-commerce avec Drupal Commerce ou simplement pour un bon site de contenu dynamique, multilingue.

Avec Drupal j'ai pu réaliser des réseaux sociaux d'entreprise, même une usine à mini-sites événementiels. Mais la chose qui avait toujours péché, c'était le flux de travail local-dév-staging-production; Et c'est un problème que chaque développeur Drupal rencontre. La communauté l'a souvent abordé, mais le problème reste.

Le problème du Développement / Staging / Production

Selon la volonté de son créateur Dries Buytaert, Drupal est une bête mixte, et il l'a toujours été, entre un "Web Application Framework" permettant à des développeurs de créer de vraies applications, et le "CMS" permettant à un utilisateur final, sans connaissances techniques poussées, non seulement d'écrire du texte, mais encore plus, de remodeler son site, gérer l'affichage, activer et désactiver des fonctionnalités complexes. C'est pour cette raison principalement qu'une partie importante de chaque site Drupal est dans une base de données plutôt que d'être décrite dans des fichiers de configuration ou par du code.

Or ce qu'on sait très bien faire avec des fichiers texte, on sait mal le faire avec une base de données. Suivre les modifications, gérer plusieurs branches parallèles, fusionner des parties de deux branches distinctes. La manière principale de résoudre le problème réside dans le fait que l'intégralité des développements d'un site doit à un moment ou à un autre s'exprimer par des fichiers texte sur le disque. On utilise ainsi les Profiles d'installation comme base, les fichiers make de drush pour décrire les composants du site, incorporer des modules, patcher le cas échéant le core, mais surtout les features qui sont en gros un format d'exportation exposé par de nombreux modules permettant de décrire, en code, ce que l'on a configuré en utilisant l'interface utilisateur de Drupal. La

prochaine version, Drupal 8.0 doit apporter pas mal d'évolution sur ce thème, sans toutefois donner une réponse finale. Or ce mode de développement est loin d'être très productif. Donc, assez souvent des sites qui auront été même "correctement développés" tendent à s'éloigner de ce modèle.

Mais il faut dire aussi que des très nombreux développeurs Drupal, dans des petites agences comme dans des sociétés importantes, considèrent le surcoût du développement structuré (avec l'export de chaque fonctionnalité vers du code) simplement prohibitif. Il ne manque pas de sites développés avec des environnements de développement et des tests qui s'éloignent progressivement de la production, voire des sites qui n'en ont pas du tout. Encore de nos jours, pas mal des sites Drupal sont mis à jour par FTP. Sans le suivi d'un système de gestion de version il est difficile de dire avec précision ce qui tourne réellement en production.

Mais même dans les environnements les mieux structurés, où tout va dans git, où le déploiement est automatique, nous voyons les limites du système "Dev/Staging/Prod": l'environnement de Dev est quasiment toujours cassé (même en ayant un système d'intégration continu) et Staging devient un pot-pourri de mille features différentes, totalement différentes de la production, mais on ne sait pas précisément en quoi. Quand les gens du métier ou les responsables qualité doivent tester, ils ne savent pas ce qu'il faut tester (donc soit on doit faire des scénarios complets très longs, soit on a le risque de ne pas voir les régressions).

Avec le Dev cassé et le Staging dans une situation de flux quantique, non seulement il est difficile de savoir où on en est, mais les données présentent dans Staging sont aussi très souvent obsolètes et encore plus souvent pourries, avec de très nombreux articles appelés "Plop" et des produits d'une grande popularité appelés "qsdqsd".



Une plate-forme cloud dédiée

Nous avons travaillé à une solution permettant de résoudre l'ensemble de ces problématiques, c'est à dire ne rien perdre en qualité tout en gagnant en productivité : <http://platform.sh>. Platform est une solution cloud de développement et d'hébergement. La révolution est d'éliminer les concepts de "Dev" et "Staging", est de ne garder que la prod (comme aux premiers jours de bricolage heureux et sans soucis)... que l'on peut via une simple commande ou un clic de souris, dupliquer intégralement (on clone ainsi les serveurs différents pour le serveur d'application, la base de données, moteur de recherche, cache, etc...), pour créer une copie parfaite à l'octet près, et ceci en quelques secondes. En gros Git, le SCM utilisé ne contrôle plus uniquement le code à déployer, mais la topologie complète du cluster. Chaque branche Git représente un environnement web complet, identique à la prod. On peut le visiter via un navigateur, ou y accéder en SSH. Au lieu d'avoir un seul Dev et un seul Staging, on peut en créer une dizaine dans la journée. On peut ainsi tester chaque nouvelle fonctionnalité en isolation absolue. Une fois le changement testé et approuvé, il suffit de coupler la branche sur master, et voilà, le nouveau code est en production.

Platform utilise une fonctionnalité du Kernel Linux permettant d'avoir quelque chose qui ressemble en tout point à de la virtualisation (faire tourner plusieurs systèmes d'exploitation sur la même machine), mais avec un surcoût minimal. Ils utilisent un projet qui s'appelle LXC et qui gère ce concept de Containers (récemment popularisé par Docker). Là où sur une machine avec disons 20GO de mémoire on peut arriver à avoir une dizaine de VM (et ça va ramer!) avec Platform Commerce Guys, on parvient à en faire tourner plus de 1 000 (et ça va vite!). Chaque service peut-être alloué

dynamiquement à un serveur différent, et on peut contrôler précisément les ressources qui lui sont allouées (et les changer aussi dynamiquement) en termes de processeur, mémoire, disque dur et ainsi de suite. Une branche de développement qui ne va quasiment jamais avoir des visiteurs peut ainsi consommer très peu de ressources, tandis que la branche de production peut verticalement croître pour atteindre la taille de la machine entière. Voir même être distribuée sur de multiples machines et de multiples datacenters donnant une scalabilité horizontale; la version "entreprise" ajoutant la garantie que chaque composant aura un cluster d'au moins trois serveurs dans trois datacenters différents, permettant de garantir une haute-disponibilité réelle.

Pour contrôler Platform, l'interface web a été travaillée pour être extrêmement optimisée :

Fig.1.

Ici, la création d'un nouvel environnement (une nouvelle branche + les services).

On peut également contrôler Platform via un outil en ligne de commande : Fig.2.

On peut également utiliser Git. Par exemple pour prendre un projet Drupal nu, et le transformer en une boutique e-commerce complète vous pouvez...

```
git checkout -b drupal-commerce
```

Ouvrir dans son éditeur de texte préféré le fichier "project.make" et ajouter la ligne:

```
projects[commerce_kickstart][type] = core
```

Puis simplement :

```
git push
```

Fig.3.

Quelques secondes plus tard, vous avez un environnement de Dev nouveau (avec toutes les fonctionnalités e-commerce que l'on peut imaginer...). Ceci ne transforme pas uniquement les relations entre commanditaire et développeurs, ni entre métier et technique, mais aussi pour les techniciens, la relation entre sysadmin et développeurs. Plus besoin de tickets pour provisionner un nouvel environnement, ni de sueurs froides quand le succès est au rendez-vous pendant un weekend et qu'il faut ajouter des ressources à la prod. Plus besoin non plus de migration vers un "serveur plus gros".

Ori Pekelman, Commerce Guys

Fig.1

Platform Cli version 0.1	
Available commands:	
branch	Branch an environment.
build	Builds the current project.
checkout	Checkout an environment.
environments	Get a list of all environments.
get	Does a git clone of the referenced project.
help	Displays help for a command
list	Lists commands
projects	Get a list of all active projects.
ssh-keys	Get a list of all added SSH keys.
environment	
environment:backup	Backup an environment.
environment:branch	Branch an environment.
environment:checkout	Checkout an environment.
environment:delete	Delete an environment.
environment:merge	Merge an environment.
environment:synchronize	Synchronize an environment.
project	
project:build	Builds the current project.
project:delete	Delete a project.
project:get	Does a git clone of the referenced project.
ssh-key	
ssh-key:add	Add a new SSH key.
ssh-key:delete	Delete an SSH key.

Fig.2

Master		
View this website Access information		
ENVIRONMENTS		
Master		
Showing all types of messages		
Ori Pekelman branched environment drupal-commerce from Master - configure environment Success a minute ago		
Ori Pekelman pushed to Master Success 3 minutes ago		
4d8a563 empty		
Ori Pekelman changed project name to Demo project Success a day ago		
Platform Bot created a new project Project title - configure project Success a day ago		

Fig.3

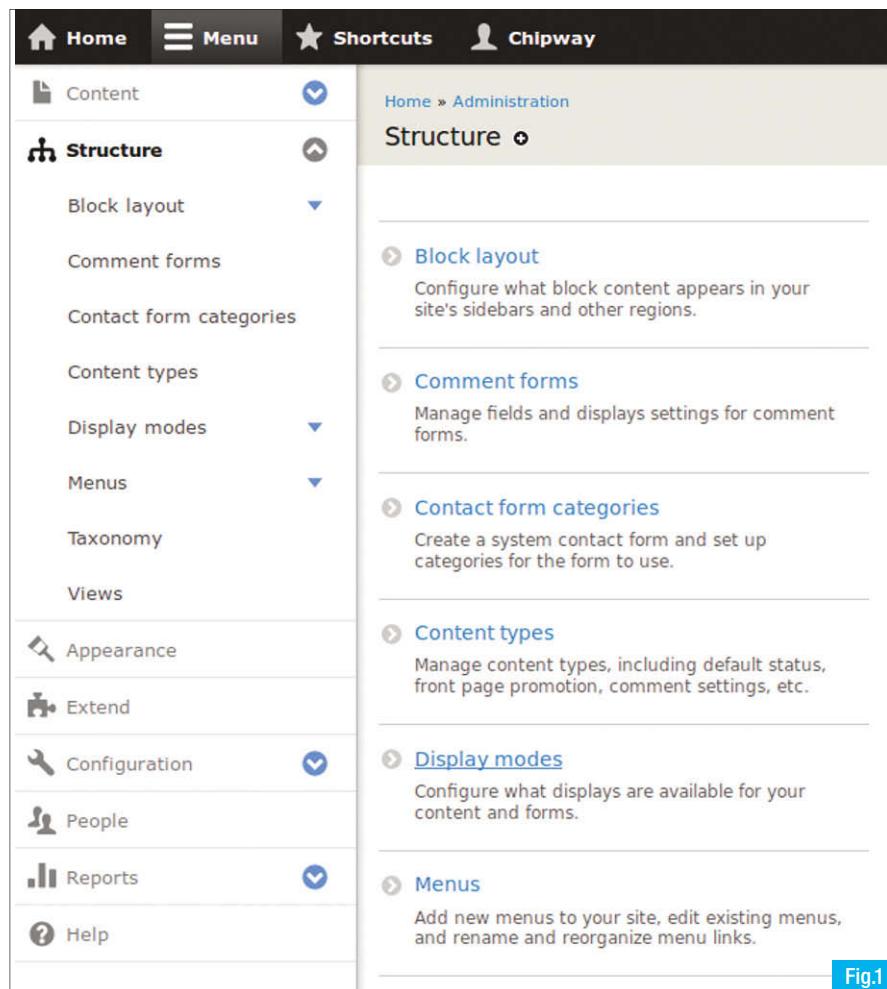
Les nouveautés de la version 8

Alors que la sortie de la version 8 de Drupal approche, voici une synthèse des principales évolutions, améliorations et enjeux liés à cette version majeure qui va révolutionner cet outil et la façon de développer, tout en confortant des évolutions déjà engagées. Cet article vous donnera un aperçu rapide de Drupal 8 en action. Vous comprendrez pourquoi Drupal 8 sera plus facile et agréable à utiliser pour les créateurs de contenu, bien plus simple à installer pour les administrateurs/assembleurs de sites, un plaisir à habiller pour les designers et un challenge plus élevé pour les développeurs.

Drupal est un CMS et framework de développement gratuit et très polyvalent composé d'un Cœur (appelé Core en anglais) auquel des milliers de modules contributifs apportent des fonctionnalités complémentaires selon les cas d'utilisation du site à construire. Son framework permet de développer de nouvelles fonctionnalités (modules) qui interagissent avec le Cœur et les modules contributifs en place. Des modules peuvent également être développés pour modifier le comportement du Cœur ou de modules contributifs sans les modifier (sans les hacker selon l'expression consacrée). Drupal, dont le modèle de développement est contributif et collaboratif, est supporté par une communauté organisée d'un million de membres dont plus de 30 000 développeurs. Drupal fait tourner plus d'un million de sites web dans le monde parmi lesquels whitehouse.gov, gouvernement.fr, lefigaro.fr, louvre.fr, cartier.com, guerlain.com, ec-lyon.fr, franceinter.fr, ...

Drupal permet de développer des sites internet, des portails, des intranets ou extranets, des boutiques ou des applications métier.

Depuis sa première version 1.0 sortie en 2001,



The screenshot shows the Drupal 8 administration interface. The top navigation bar includes 'Home', 'Menu', 'Shortcuts' (with a 'Chipway' profile icon), and a user menu. The left sidebar is titled 'Structure' and contains links for 'Content', 'Block layout', 'Comment forms', 'Contact form categories', 'Content types', 'Display modes', 'Menus', 'Taxonomy', 'Views', 'Appearance', 'Extend', 'Configuration' (with a checked icon), 'People', 'Reports', and 'Help'. The main content area is titled 'Structure' and lists 'Block layout', 'Comment forms', 'Contact form categories', 'Content types', 'Display modes', and 'Menus', each with a brief description and a link to its configuration page.

Fig.1

Responsive Design y compris l'interface d'administration

Drupal a connu 6 versions majeures. La version courante est la version 7. Chaque version majeure apporte des améliorations et changements significatifs quant aux possibilités incluses de base (dans le Cœur) et à la façon de développer un site web. La version 8 qui est en développement sera probablement celle qui va apporter les plus grands changements dans Drupal. Nous allons voir quels sont ses principaux changements, ainsi que leurs impacts prévisibles ou attendus pour différents profils ou cas d'utilisation. Le site officiel de Drupal est géré par la « Drupal Association » internationale : <http://drupal.org>. Le site officiel de Drupal pour la France est le site de l'Association Drupal France et Francophonie (ADFF) : <http://drupalfr.org>.

Principales nouveautés de Drupal 8

Toutes les nouveautés évoquées ici concernent le Cœur de Drupal 8. Views est maintenant intégré au Cœur. C'était jusque là

le module contributif le plus utilisé (environ 800 000 sites) car il permet de définir facilement et rapidement des requêtes sur tout type de contenus ou d'objet Drupal, et d'afficher les résultats sous forme de listes, tableaux, grilles, diaporamas, flux RSS, ... Ces listes peuvent être triées et filtrées pour obtenir par exemple des résultats contextualisés (ex : la liste des ouvrages d'un écrivain dont nous sommes en train de consulter le profil).

Parallèlement Field, apparu avec Drupal 7, est largement réécrit et supporte de nouveaux champs. Pour mémoire, Field permet de créer des structures de contenus ou autres types d'entités sophistiqués comportant des champs de types très variés.

Par défaut Drupal utilise un thème en Responsive Design et HTML5 pour répondre aux besoins d'accès via des terminaux mobiles tels que smartphones ou tablettes, y compris pour l'interface d'administration Fig.1.

La gestion de la disposition des éléments sur la page est harmonisée et évolue vers le "tout sous forme de blocs", blocs qui seront plus

faciles à contextualiser.

La configuration de Drupal et des modules, précédemment stockée dans la même base de données que les contenus, est stockée séparément sous forme de fichiers au format YAML. YAML acronyme récursif de YAML Ain't Markup Language, est un format de représentation de données par sérialisation Unicode. Or la configuration d'un site Drupal comporte de très nombreux éléments générés par les développements réalisés via l'interface d'administration (sans programmer). Ce nouveau mode de stockage des configurations les rend faciles à lire, à gérer avec des outils de gestion de versions de code et à déployer. Le multilingue devient natif et concerne tous les éléments qu'il s'agisse de contenus, d'interfaces ou de configurations. Avec une architecture clarifiée, il devient plus simple à mettre en œuvre tout en étant plus complet. Par exemple, dès l'installation, le site peut passer dans une autre langue que l'anglais avec des traductions immédiatement récupérées depuis « localize.drupal.org ». Des web services complets sont directement supportés par le Cœur (REST, JSON, XML, intégration avec Entity API et Views, ...). Ces web services permettent d'interfacer Drupal avec tous types de services externes et systèmes d'information.

Le développement de Drupal s'appuie maintenant sur des composants Symfony2, et le système de thèmes et templates (« gabarits ») propre aux versions précédentes, phptemplate, est remplacé par Twig, plus sécurisé, facile à lire et à apprendre, utilisé ailleurs et extensible.

L'installateur de Drupal est simplifié.

Drupal intègre maintenant un éditeur HTML par défaut, CKEditor version 4, ainsi que la

possibilité d'édition directe d'éléments visibles sur le site géré (« edit in-place ») **Fig.2**.

Apports de Drupal 8

Drupal 8 est directement utilisable grâce à Views et à l'éditeur intégré. Il permet de créer des sites avec son seul Cœur, sans attendre le portage de modules contributifs vers cette version. Drupal 8 supporte les terminaux mobiles et le HTML5 nativement. Il parle plusieurs langues plus simplement et fournit des services web nativement. Le montage de la page « tout sous forme de blocs » augmente la souplesse de la disposition des éléments sur la page et rend le résultat plus prédictible. L'installation de Drupal est simplifiée et il est facile à déployer entre les différents serveurs tout au long du cycle de développement/test/intégration/recette/pré-production/production. Le design est facilité car les thèmes Twig sont plus simples à créer et à modifier en particulier si l'intégrateur connaît déjà ce système de gabarits utilisé sur d'autres CMS.

Challenges et opportunités

La création et l'habillage d'un site sont simplifiés sous Drupal 8 grâce à tous les éléments intégrés dans son Cœur. Il devient donc abordable par un plus grand nombre d'assemblateurs, designers ou intégrateurs bien que la compréhension de son fonctionnement reste un atout dans la réalisation de sites pérennes et évolutifs. La possibilité pour chaque métier (développeur, intégrateur, graphiste) d'intervenir à son niveau sans interférence sur le travail des autres est encore plus claire qu'avec les versions précédentes. En contrepartie, le développement par programmation nécessitera des notions de Symfony2 en particulier la connaissance des

composants utilisés par Drupal.

Globalement, la version 8 devrait contribuer à élargir rapidement l'écosystème des Drupal installés un peu partout dans le monde, car elle renforce les forces (polyvalence, flexibilité, évolutivité, qualité, sécurité et pérennité...) et améliore ou fait disparaître quelques faiblesses (délai d'adoption des nouvelles versions majeures, site directement utilisable sans avoir à ajouter de nombreux modules, gestion des configurations...).

Oui mais quand ?

Si ces avantages vous ont mis en appétit et qu'il vous tarde de réaliser votre premier site sous Drupal 8, vous poserez forcément deux questions : ▶ « quand sortira-t-il ? » : Bien qu'il soit difficile d'annoncer une date de sortie pour Drupal 8.0 qui sortira « lorsqu'il sera prêt » comme a coutume de dire Dries Buytaert le leader du projet depuis sa création, il est probable qu'elle se situera autour du printemps 2015.

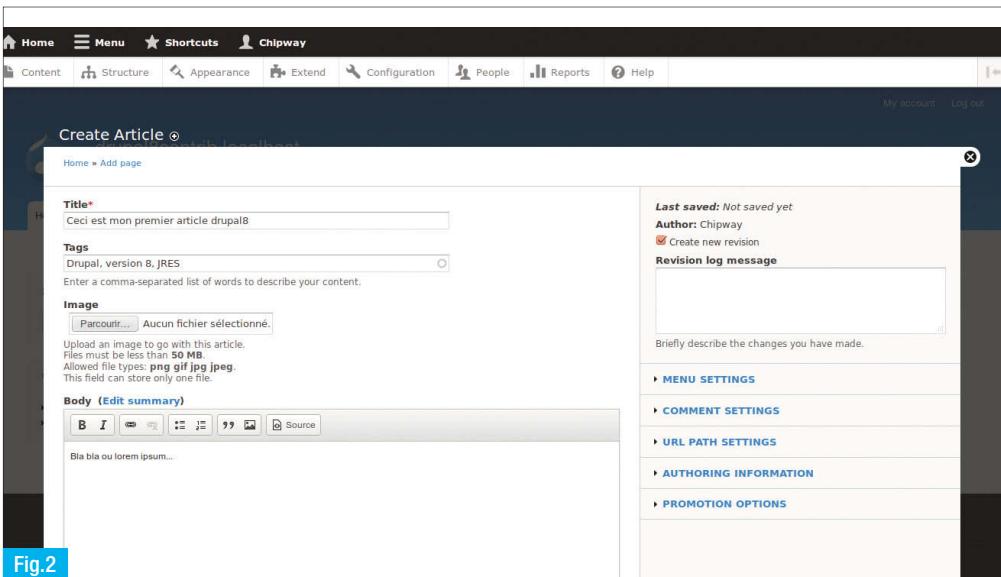
▶ « si j'ai un projet à réaliser sous Drupal en 2014, est-ce que j'attends Drupal 8 ou est-ce que je me lance en Drupal 7 ? » : S'il s'agit d'un site de test ou pour participer au développement de Drupal 8 (debuggage, patch, ...), vous pouvez utiliser Drupal 8, dès aujourd'hui. Mais s'il s'agit de développer un site qui devra être en production d'ici fin 2014, utilisez la version stable actuelle Drupal 7.

Notez bien que Drupal n'est pas une application Symfony2 mais un système Drupal qui embarque des composants Symfony. Si vous ne connaissez pas Drupal, le démarrage d'un site sous Drupal 7 vous permettra de vous initier et familiariser avec les concepts Drupal qui perdurent en Drupal 8, et dont certains portent des appellations que l'on retrouve en Symfony pour des concepts très différents.

Les concepts et fonctionnements que vous apprendrez de Drupal 7, vous permettront d'utiliser Drupal 8 directement pour la partie assemblage de site. D'autre part, la migration entre versions majeures Drupal sera améliorée, par exemple de Drupal 7 à 8, mais aussi de Drupal 6 à 8. Donc, n'attendez pas Drupal 8 pour vous mettre à Drupal et n'hésitez pas à rejoindre les 2068 contributeurs qui ont déjà fourni du code pour Drupal 8.

● Léon Cros

Président de l'Association Drupal France et Francophonie et Directeur Technique de Chipway (@chipway)

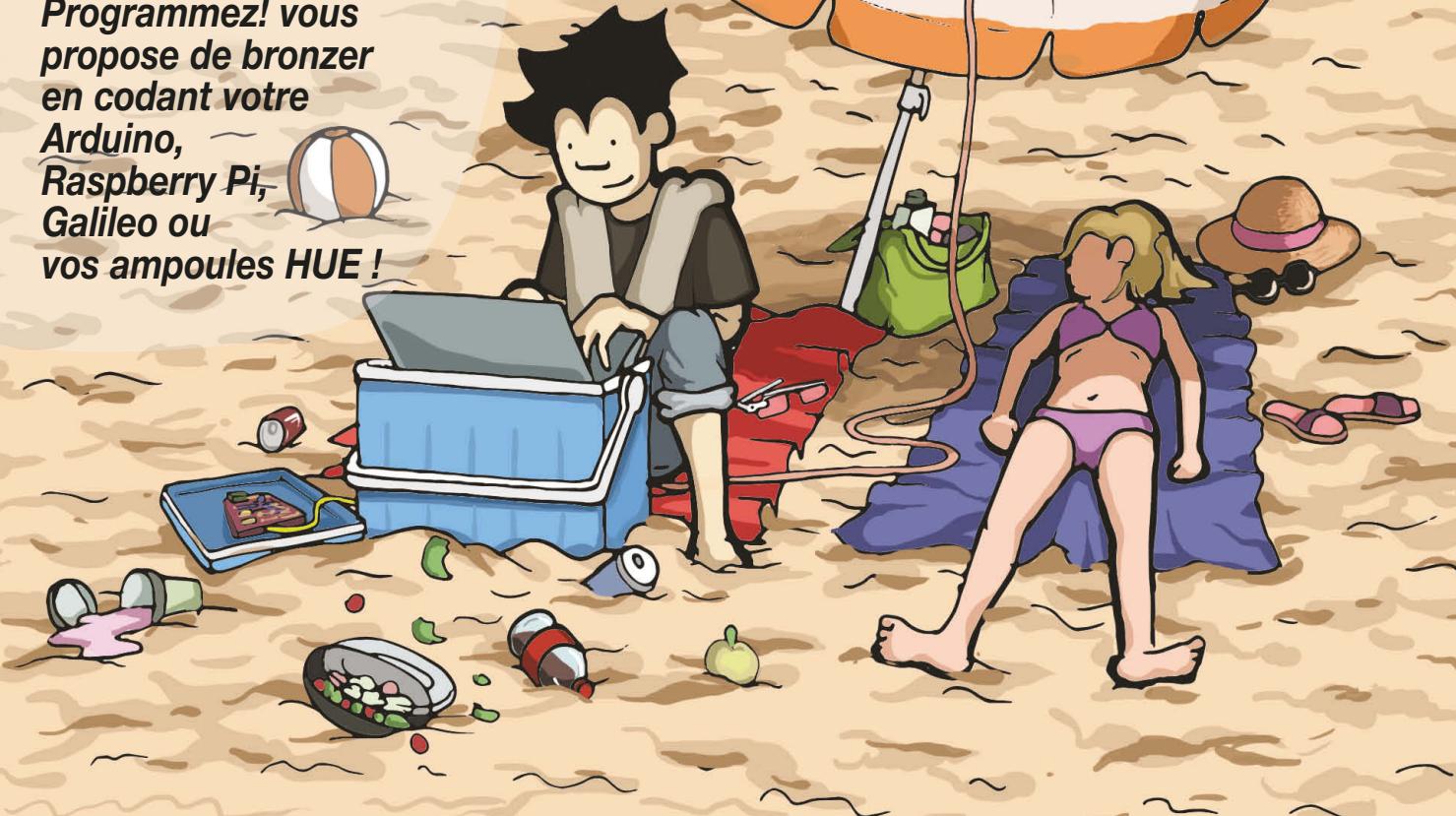
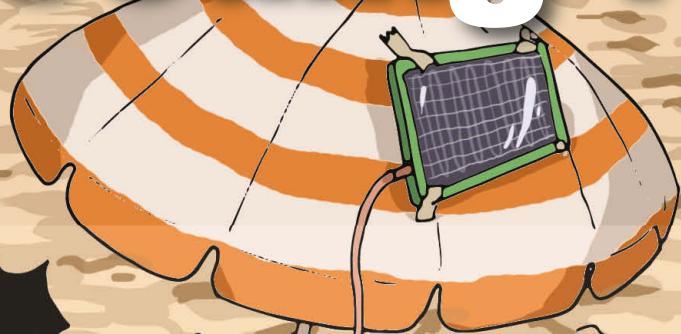


Dès son installation, Drupal 8 propose un éditeur HTML opérationnel

Soleil, Sable et Coding !

La plage c'est bien.
Mais la plage avec son portable, son IDE préféré,
c'est encore mieux.

Pour le numéro d'été,
Programmez! vous
propose de bronzer
en codant votre
Arduino,
Raspberry Pi,
Galileo ou
vos ampoules HUE !



N'oubliez pas votre chargeur solaire ou une très très longue rallonge électrique. Fonctionne aussi à la montagne et dans son jardin.

Coding4Fun - La rédaction.

Faire sa domotique soi-même avec les lampes HUE

Quel bon geek n'a pas rêvé d'avoir sa maison connectée ? Franchissons ensemble une première étape en automatisant de bout en bout une partie essentielle de notre habitat : l'éclairage.



Fig.1

L'objectif de cet exercice est de se passer de nos vieux interrupteurs en laissant nos lampes s'allumer et s'éteindre d'elles-mêmes, en fonction de notre présence dans la pièce. Pour cela nous n'aurons besoin que d'objets disponibles au grand public. Du côté technique nous allons utiliser un objet utilisant la technologie Bluetooth Low Energy : le CC2541 de Texas Instrument, un ordinateur sous Windows 8.1, et des lampes Hue.

Installer et appréhender les lampes Hue

Pour automatiser l'éclairage, il nous faut des lampes connectables. J'ai à disposition des lampes Hue de chez Philips qui, malgré leur prix qui peut être dissuasif (60€ la lampe environ, 200€ le kit de 3 lampes et d'un routeur), sont très faciles d'accès au niveau code et permettent des scénarios complexes en jouant sur l'intensité et la couleur de l'éclairage ainsi que sur la création de groupes de lampes pour les grandes pièces par exemple [Fig.1](#).

Une fois les lampes installées et le bridge connecté à notre réseau, il nous faut rentrer l'adresse <http://www.meethue.com/api/hupnp> dans notre navigateur préféré pour voir le résultat suivant :

L'API de nos lampes sera donc disponible à l'adresse indiquée par la propriété « `internalipaddress` » et une interface de test est disponible ici : <http://<internalipaddress>/debug/clip.html>

Nous ne nous étendrons pas plus sur la découverte des lampes et de l'API puisque la documentation officielle est de qualité. Rendez-vous sur <http://developers.meethue.com/gettingstarted.html>

Commander notre éclairage

Créons notre application Windows 8.1 vide et dans le code-behind de notre MainPage et ajoutons le code suivant :

```
        }  
        catch (Exception e)  
        {  
            Debug.WriteLine(e)  
        }  
    }  
}
```

Vous remplacerez bien évidemment l'adresse du bridge et l'id de la lampe avec les infos propres à votre installation.

Créons ensuite 2 boutons « On » et « Off » dont nous lierons les événements de clic à la méthode `TurnLightAsync` avec les paramètres d'id de la lampe ciblée et de l'état souhaité.

A ce stade nous avons un programme qui nous permet d'allumer/éteindre une lampe. Passons à l'automatisation.

Appairage et détection de notre appareil Bluetooth Fig 2

L'allumage des lampes dépend de la détection de notre présence dans la pièce. Pour détecter à moindre coût cette présence, l'utilisation d'un module Bluetooth est une des solutions les plus avantageuses. Avec sa faible consommation d'énergie, cet appareil pourra être porté directement par la personne à t

faible portée. Et oui, si l'on veut que notre absence soit détectée, il faut bien éviter que notre appareil soit visible depuis l'autre côté de notre appartement. En outre, de nombreux appareils comme les montres connectées sont déjà équipées de cette technologie.



Fin 2

Pour l'exercice j'ai utilisé le CC2541 de Texas Instruments qui est un capteur autonome de température, humidité, etc. qui fonctionne avec une pile de montre et transmet ses données en Bluetooth. Il a aussi la faculté de se reconnecter automatiquement lorsqu'il se retrouve près d'un appareil auquel il a été appairé. Vous l'aurez compris, c'est cette fonctionnalité



qui nous sera utile.

Commençons par appairer notre appareil à notre ordinateur/tablette via le menu de gestion Bluetooth de Windows 8.1, **Fig.3.**

Une fois l'appairage effectué, nous allons lister les appareils connus par notre ordinateur :

66 Programmez! < Juillet-Août 2014

```
var devices = await Windows.Devices.Enumeration.DeviceInformation.FindAllAsync();
```

Cette méthode nous fournit une liste très importante d'appareils, mais nous pouvons retrouver notre appareil par son nom :

```
DeviceInformation device = devices.FirstOrDefault(d => d.Name == "TI BLE Sensor Tag");
if (device == null) return;
```

Nous pourrions également filtrer la liste des appareils par leur capacité à fournir le service GenericAccess que notre CC2541 fournit :

```
//Find the devices that expose the service
var devices = await DeviceInformation.FindAllAsync(GattDeviceService.GetDeviceSelectorFromUuid(GattServiceUuids.GenericAccess));
```

Une fois notre appareil ciblé, il ne nous reste plus qu'à traquer l'état de sa connexion.

Pour cela ajoutons le code suivant :

```
private PnpObjectWatcher _watcher;
private async void InitDeviceWatcher(DeviceInformation device)
{
    _deviceContainerId = "{" + device.Properties["System.Devices.ContainerId"] + "}";

    _watcher = PnpObject.CreateWatcher(PnpObjectType.DeviceContainer,
        new string[] { "System.Devices.Connected" }, String.Empty);

    // Add event handlers
    _watcher.Updated += watcher_Updated;
    _watcher.Start();
}
```

A chaque fois que l'état de connexion du device sera changé, la méthode `watcher_Updated` sera appelée.

Pour que cette méthode fonctionne, il faut avoir demandé dans la liste des devices de nous fournir la propriété `ContainerId`. Pour cela, modifions notre méthode de récupération des objets compatibles et n'oublions pas d'initialiser notre `watcher`:

```
var devices = await DeviceInformation.FindAllAsync(GattDeviceService.GetDeviceSelectorFromUuid(GattServiceUuids.GenericAccess), new String[] { "System.Devices.ContainerId" });
DeviceInformation device = devices.FirstOrDefault(d => d.Name == "TI BLE Sensor Tag");
if (device == null) return;
InitDeviceWatcher(device);
```

Nous pouvons ensuite gérer l'évènement de connexion/reconnexion de notre appareil :

```
private async void watcher_Updated(PnpObjectWatcher sender, PnpObjectUpdate args)
{
    var connectedProperty = args.Properties["System.Devices.Connected"];
```

```
bool isConnected = false;
if (_deviceContainerId == args.Id) && Boolean.TryParse(connectedProperty.ToString(), out isConnected)
{
    if (isConnected)
    {
        Debug.WriteLine("Connected");
    }
    else
    {
        Debug.WriteLine("Disconnected");
    }
}
```

A cette étape, nous arrivons à détecter notre appareil. Lorsque nous le portons et nous nous déplaçons dans d'autres pièces, notre programme est alerté de sa déconnexion et sa reconnexion. Il ne nous reste plus qu'à automatiser les lampes sur ces évènements.

Automatisation de l'allumage et de l'extinction de l'éclairage

Le branchement des deux étapes de développement que nous avons réalisées se fait assez simplement en appelant la méthode `TurnLightAsync` depuis la méthode `watcher_Updated` de la seconde étape.

```
private async void watcher_Updated(PnpObjectWatcher sender, PnpObjectUpdate args)
{
    var connectedProperty = args.Properties["System.Devices.Connected"];
    bool isConnected = false;
    if (_deviceContainerId == args.Id) && Boolean.TryParse(connectedProperty.ToString(), out isConnected)
    {
        if (isConnected)
        {
            Debug.WriteLine("Connected");
            TurnLightAsync(LIGHT_ID, true);
        }
        else
        {
            Debug.WriteLine("Disconnected");
            await TurnLightAsync(LIGHT_ID, false);
        }
    }
}
```

Magique, non ?

Conclusion

Dans le cadre de cet article, nous avons démontré qu'avec des appareils destinés au grand public et quelques lignes de code nous pouvions automatiser notre éclairage. Il ne vous reste plus qu'à imaginer des scénarios plus poussés utilisant des données de température, des couleurs d'éclairage et d'autres pour construire votre domotique.

Raspberry Pi : première installation

Le Raspberry Pi est un micro-ordinateur à processeur ARM développé par la fondation Raspberry Pi. Bien que principalement destiné à l'apprentissage de la programmation, il est suffisamment puissant pour permettre une multitude de fonctionnalités. Il fonctionne grâce à des systèmes d'exploitation libres tels que GNU/Linux ou FreeBSD.

En 2006 les premiers prototypes voient le jour, mais c'est en 2012 que la première version grand public est mise en vente, le Raspberry Pi A, disposant de 256 Mo de RAM. Très vite une autre version, le Raspberry Pi B est commercialisée. Dans sa deuxième révision, il dispose de 512 Mo de RAM mais également de deux ports USB et d'une prise Ethernet, absente de la version précédente. L'amélioration de ses capacités lui a permis notamment de gagner en performance graphique puisqu'il peut désormais lire des vidéos HD. Une autre version est aujourd'hui en cours de production, le Raspberry Pi Computer, plus modulable. Mais nous nous intéresserons aujourd'hui plus particulièrement au Raspberry Pi B 2 ou plus couramment appelé le Raspberry Pi 512. Son prix avoisine les 35 euros. Afin de baisser son coût au maximum, le raspberry Pi est livré seul, sans accessoire. Il vous faudra donc, pour pouvoir l'utiliser, un clavier et une souris usb,

un câble HDMI ou RCA, un câble microSD et une carteSD.

Voici le schéma de ce modèle : Fig.1.

Nous pouvons voir qu'il dispose d'un assez grand choix de connectiques. Pour la vidéo, une prise HDMI est disponible mais également une prise RCA, pour compatibles avec les anciens postes de télévision.

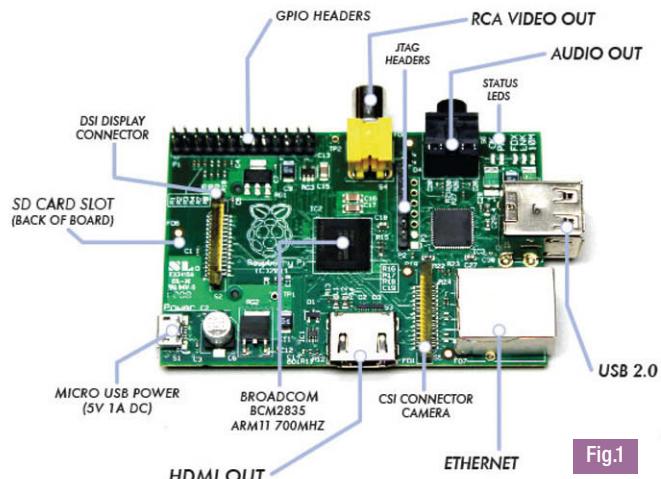


Fig.1

Installation

Le site Raspberry propose plusieurs distributions adaptées au Raspberry Pi notamment Raspbian (debian), ElecOs (un système basé sur l'environnement graphique XBMC), Pidora

(Fedora), Risc (une distribution non Linux).

Nous allons utiliser le dossier «Noobs» - New Out Of Box Software - qui comprend toutes les distributions proposées. Ce dossier, comme son nom l'indique, est à destination des débutants. Il facilite grandement l'installation du système de son choix. Il suffit de disposer d'une carte SD (dans notre cas 8 Go) et de télécharger le dossier noobs à cette adresse :

<http://www.raspberrypi.org/downloads/>. Deux versions sont disponibles, une version netinstall (installation par le réseau) et une version complète. Nous allons prendre la version complète.

Que ce soit sur Mac ou sur un PC (Windows, GNU/Linux, BSD), la démarche reste la même :

- ▶ Formatez votre carteSD au format FAT.
- ▶ Dézipez le fichier NOOBS précédemment téléchargé.
- ▶ Copiez le contenu du dossier NOOBS à la racine de votre carte SD.
- ▶ Positionnez votre carte SD dans le Raspberry.
- ▶ Connectez le Raspberry et tous ses accessoires et mettez-le sous tension grâce au câble microSD.
- ▶ Vous arriverez normalement sur cette page, vous présentant les systèmes que vous pouvez installer Fig.2.

Comme nous pouvons le constater, le système recommandé est Raspbian, premier de la liste. Vous pouvez installer plusieurs systèmes sur la même carte SD. Dans ce cas, un choix vous sera demandé à chaque démarrage.

Nous allons donc choisir Raspbian et configurer le langage et le clavier en français (menu déroulant en bas). Cliquez sur install en haut. Pendant l'installation, vous n'aurez plus d'interface graphique.

Raspbian démarre et vous arriverez sur l'écran ci-contre Fig.3.

Pour naviguer, utilisez les flèches du clavier, la touche entrée pour valider ou entrer dans les menus.

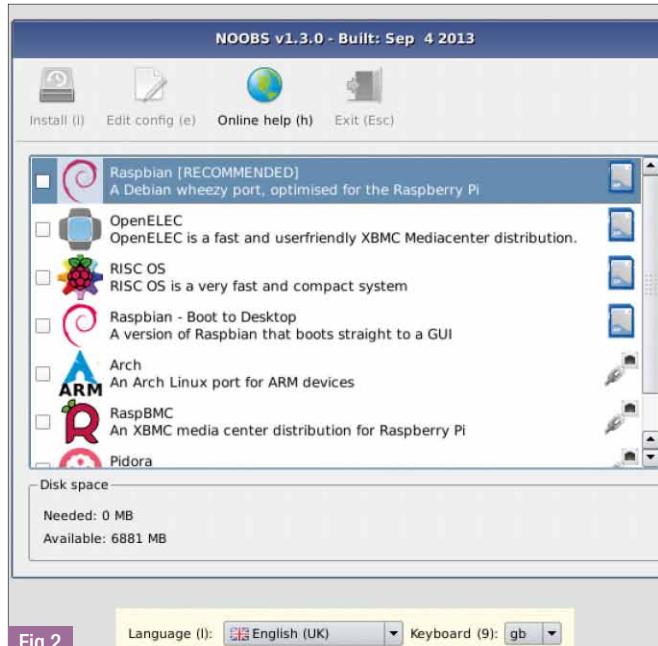


Fig.2

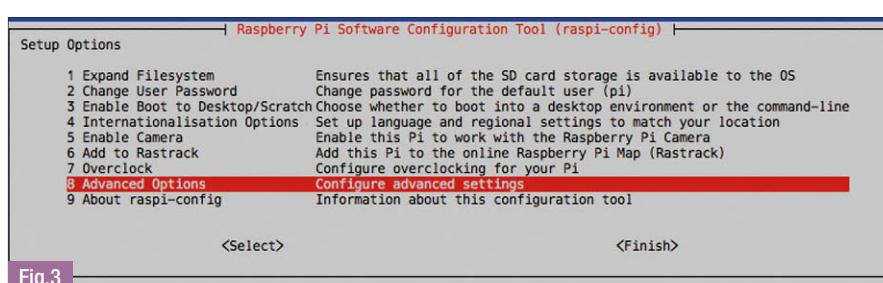


Fig.3

La tabulation permet de changer la zone du curseur (par exemple ici la tabulation permet de se positionner sur Select puis sur Finish et de revenir sur Setup Options). On se servira de la touche espace pour cocher/décocher des options. La plupart des paramètres renseignés ici pourront être modifiés après l'installation.

Voici le détail du menu d'installation :

1 - Expand Filesystem :

Permet d'utiliser la totalité de la carte SD pour l'OS. NOOBS le fait à notre place, vous n'avez pas besoin de vous en occuper.

2 - Change user Password :

Changer le mot de passe de l'utilisateur pi (l'utilisateur par défaut). Au moment où vous entrerez le mot de passe rien ne s'affichera pour des questions de sécurité.

3 - Enable Boot to Desktop/Scratch :

Option pour démarrer directement en graphique. Par défaut, Raspbian démarre en mode console. La commande startx permet de démarrer le serveur graphique.

4 - Internationalization Options :

Menu pour définir les langues prises en charge, la zone géographique et l'agencement du clavier.

5 - Enable Camera :

Permet d'utiliser la caméra Raspberry Pi.

6 - Add to Rastrack :

Permet de recenser son Raspberry sur la carte interactive officielle des Raspberry (Rastrack).

7 - Overclock :

Configuration de l'Overclocking.

5 choix nous sont proposés : None, Modest, Medium, High, Turbo. L'Overclocking permet d'augmenter la cadence du processeur et par conséquent les performances mais peut rendre votre système instable. Ceci réduit également la durée de vie de votre Raspberry. Si vous ne savez pas quoi mettre, sélectionnez

None ou Modest. Medium permet un compromis entre performance et sauvegarde du système.

8 - Advanced Options :

- ▶ **Overscan**: Permet de moduler l'affichage en fonction de la taille de l'écran connecté.
- ▶ **Hostname** : Permet de modifier le nom de l'ordinateur sur le réseau.
- ▶ **Memory split** : Permet de changer la quantité de mémoire

allouée à la partie graphique (GPU). Modifiez cette variable (16/32/64/128/256 Mo) en fonction de l'usage que vous réservez à votre machine. Si vous ne comptez pas beaucoup vous servir du serveur graphique, 64 ou 32 voire 16 Mo seront suffisants. Si vous comptez lire des vidéos, vous devriez avoir un minimum de 128 Mo réservés au GPU. Par défaut nous mettrons 128 Mo.

- ▶ **SSH** : Permet la connexion en SSH. Utile uniquement si vous désirez utiliser votre Raspberry à distance.
- ▶ **SPI** : Active la connexion SPI (Serial Peripheral Interface). Permet la communication avec un autre matériel électronique.
- ▶ **Audio** : Permet de forcer l'utilisation de la sortie Jack ou HDMI pour le son.
- ▶ **Update** : permet de mettre à jour l'outil d'installation.

9 - About Raspi-config :

Information sur l'outil de configuration.

Sélectionnez Finish et validez le redémarrage. Votre Raspberry Pi devrait maintenant redémarrer et soit vous afficher un prompt soit démarrer sur l'environnement graphique LXDE en fonction de ce que vous avez sélectionné **Fig.4**.

Et maintenant ?

Le Raspberry a d'abord été conçu pour la programmation. Vous pourrez donc vous amuser à créer divers programmes dans le langage de votre choix. De nombreux projets existent dans cet environnement. Nous avons en exemple Minecraft Pi, développé par Aron Nieminen et Daniel Frisk, qui permet de jouer à Minecraft via son Raspberry Pi mais aussi d'expérimenter avec le code directement via l'API du jeu. De quoi s'initier en s'amusant à la programmation ! Une utilisation courante est aussi l'adaptation du Raspberry en serveur web ou ftp, pour stocker ou transférer des données facilement. Une adaptation très accessible, réalisable en quelques lignes de commandes avec du matériel que vous avez déjà probablement chez vous (quelques disques durs externes, un hub USB, de quoi

alimenter tout ce joli monde et le tour est joué ! D'autres sont allés plus loin en utilisant leur Raspberry Pi pour créer leur propre Smart TV, des cadres photos numériques ou même une console de jeu portable ! Bien sûr, tous ces exemples ont des niveaux très variables de difficultés, allant du simple branchement de connectiques à la maîtrise d'un fer à souder en passant par une certaine intrépidité, lorsque la garantie d'un écran télé LCD 40 pouces entre en jeu par exemple. Mais ce qui est sûr, c'est qu'avec la quantité du tutoriels et pas-à-pas très bien réalisés et qui foisonnent sur internet, et ce dans toutes les langues, vous pourrez très facilement vous lancer. La seule barrière est réellement la créativité de chacun, mais après avoir consulté tout ce qui se fait actuellement (parfois même par des enfants), ce n'est plus l'inspiration qui vous manquera. Et c'est tout naturellement qu'on le retrouve régulièrement comme outil privilégié pour la réalisation de projets de fin d'études de grandes écoles d'informatique, qui poussent leurs élèves à l'innovation, tels les projets EIP de l'ETNA. En effet, un groupe d'étudiants a conçu pixL, une plateforme permettant de jouer aux jeux de nos anciennes consoles par l'intermédiaire du Raspberry Pi. Pour suivre leur projet <http://pixl-project.com/>.

Nous pouvons donc affirmer que le Raspberry Pi a su arriver à point dans un environnement numérique de plus en plus tourné vers le partage et le DIY (Do It Yourself). Le hack, non pas au sens piratage mais au sens plus global de détournement des limites d'un objet normal, n'est certes pas un concept récent, mais grâce au Raspberry Pi, il devient ludique et à la portée du plus grand nombre. Pour aller plus loin, nous vous proposons une sélection de liens :

Fabriquer son NAS portatif grâce au Raspberry Pi (en anglais) (débutant) :

<http://benperove.com/howto/raspberry-pi-nas-for-travelers/>

Fabriquer sa console portative grâce au Raspberry Pi (en anglais) (avancé) :

<http://www.thingiverse.com/thing:321624>

Le site officiel de Raspberry Pi :

<http://www.raspberrypi.org/>

La page d'Adafruit consacrée à divers projets Raspberry :

<https://learn.adafruit.com/category/raspberry-pi>

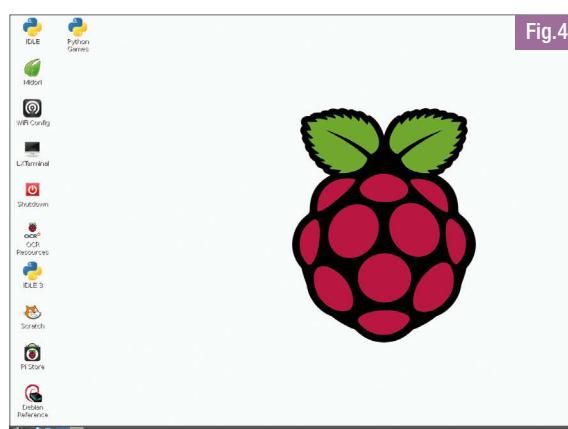


Fig.4



Chloé Liban, développeur d'applications web et Jean-Baptiste Raffin, administrateur systèmes et réseaux à l'ETNA

Pourquoi oser le Raspberry Pi ?

Apparu en février 2012 après une longue gestation, le Raspberry Pi a dépassé tous les espoirs de vente de son créateur, la Fondation Raspberry Pi. En deux ans 3 millions d'exemplaires ont été vendus dans le monde ce qui représente plus de 3600 unités par jour... Initialement conçu pour faciliter l'apprentissage de la programmation aux jeunes générations, le nano-ordinateur a rapidement été adopté par de nombreux constructeurs de systèmes nécessitant une intelligence à moindre coût, avant de pénétrer le monde de l'entreprise.

La disponibilité

Distribué au niveau mondial par Farnell Element 14 et RS Components, le Raspberry Pi (carte de circuit imprimé verte) est fabriqué dans une usine Sony du Pays de Galles, ce qui a permis de créer 30 emplois. En Chine, Egoman Digital Corp. fabrique un Raspberry Pi distribué en Chine, Hong Kong, Macao et Taiwan. Pour le distinguer du modèle européen, la carte de son circuit imprimé est rouge. La production est telle que les commandes sont livrées rapidement, sans les délais qui ont pu exister durant les premiers mois de la mise en vente.

Le Prix

Un des atouts majeurs du Raspberry Pi est son prix. Proposé sous forme de circuit imprimé prêt à l'emploi mais nu, il coûte moins de 32€ hors frais de port (Kubii – avril 2014).

Utilisé comme système embarqué (headless), il se suffit à lui-même. Pour une utilisation en poste de travail ou de développement, il faudra lui adjoindre une alimentation 5volts/2A, une carte SD, un clavier, une souris (si le mode graphique est utilisé), un écran HDMI (mais d'autres possibilités existent : vidéo composite, VGA ou DVI avec adaptateurs) et un boîtier adapté. La plupart de ces accessoires sont disponibles chez les informaticiens: un kit comprenant le Raspberry Pi, une alimentation, une carte SD prête à l'emploi et un boîtier transparent est disponible pour 59€ (Kubii – avril 2014).

Pour tous les projets «à risque» le Raspberry Pi est souvent le bon compromis performances-prix. Il a par exemple permis de réaliser un ballon qui a atteint les limites de l'atmosphère (40 km). Alimenté par quatre piles LR6 – AA et équipé du module caméra développé par la Fondation Raspberry Pi

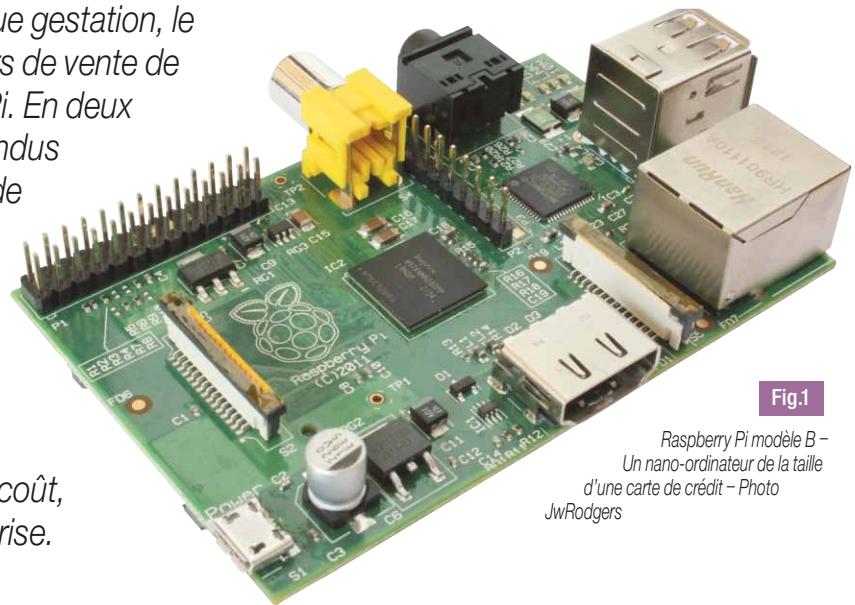


Fig.1

Raspberry Pi modèle B –
Un nano-ordinateur de la taille
d'une carte de crédit – Photo
JwRodgers

(photo 2592x1944 pixels – vidéo 1080p et 720p) le Raspberry Pi a permis la réalisation de photos stratosphériques montrant la courbure de la terre pour... une soixantaine d'euros !

Au vu du résultat, le risque financier était minime.

Le silence

Une des caractéristiques intéressantes du Raspberry Pi, surtout quand on l'utilise dans un salon pour créer un média-center, c'est son silence. La plupart des disques durs multimédias ou PC montés en multimédia possèdent des systèmes de ventilation. Sans importance dans une ambiance bruyante, ce bruit de ventilation devient vite insupportable au cours de la projection d'un film, pendant des séquences calmes. Avec le Raspberry Pi, le problème est complètement éliminé, puisque le bruit est réduit à... rien.

L'absence de ventilateur permet de déployer le Raspberry sans s'inquiéter de son impact sur l'ambiance sonore.

Les caractéristiques physiques

Avec ses dimensions réduites (86 mm x 54 mm), son poids plume (45g) et sa consommation infime (2,5 W) le Raspberry Pi peut s'intégrer facilement dans tous les environnements, s'adapter à une foultitude d'applications.

La vidéo HD

Doté d'un GPU VideoCore IV, le Raspberry Pi est capable de diffuser de la vidéo en full HD – 1080p à 30 trames par seconde (l'écran affiche une image de 1 920 pixels de large et 1 080 pixels de haut). Cela lui donne la possibilité

de produire une image de qualité Blue Ray transmise à l'écran par une prise HDMI. C'est la garantie de pouvoir utiliser des écrans de grande dimension en toute sérénité.

Un accès au monde réel

La présence d'un GPIO sur le SoC BCM2835 permet d'interfacer le Raspberry Pi au monde réel. Des entrées sorties tout-ou-rien sont accessibles en ligne de commande ou via des bibliothèques en Python (RPi.GPIO) ou en C (WiringPi). Les bus SPI et I2C ouvrent l'accès à toute une gamme de capteurs et de convertisseurs analogiques. Depuis la sortie du Raspberry Pi, le développement de cartes répondant à des besoins spécifiques a permis d'étendre les possibilités du nano-ordinateur. Convertisseurs A/D et D/A, modules GPS, GSM, capteurs de température, horloge temps réel, cartes comportant des dizaines d'E/S numériques... se disputent les faveurs du marché.

Un système ouvert

Depuis la publication par Broadcom de la documentation complète du GPU VideoCore IV et du code source complet de la pile graphique sous licence BSD en février 2014, on peut considérer que le Raspberry Pi est libre. (<http://www.raspberrypi.org/a-birthday-present-from-broadcom>)

Les limitations dues à l'utilisation obligatoire du firmware fourni par Broadcom ne devraient bientôt plus exister. Le GPU va pouvoir être utilisé au mieux, et des développements prometteurs sont en cours. On peut en attendre une amélioration substantielle des performances,

Fig.2



Journée Raspberry Pi du CERN à Genève le 12 avril 2014. Le public est varié. Adultes, ados et enfants s'amusent avec les Raspberry Pi.

ainsi que l'ajout de nouvelles fonctions, puisque les capacités de calcul du GPU deviennent accessibles. Un concours lancé par la Fondation Raspberry Pi, doté de 10 000\$ de prix, a d'ores et déjà permis de démontrer la possibilité de jouer à Quake III Arena en full HD 1080p, à 20fps en développant un driver reposant sur le GPU (<https://github.com/simonjhall/challenge>). Les prochaines étapes vont certainement être passionnantes.

Linux & Co

L'utilisateur de GNU/Linux est souvent attaché à sa distribution habituelle. Plus de 40 distributions sont disponibles pour le Raspberry Pi (http://elinux.org/RPi_Distributions#FreeBSD). La distribution préconisée par la Fondation Raspberry Pi est Raspbian, une Debian adaptée au Raspberry Pi. Mais les fans de Arch Linux, Fedora, Gentoo, RISC OS, BSD, OpenSuse ou encore Puppy Linux retrouveront leurs marques.

Une grande souplesse

Une des caractéristiques du Raspberry Pi est d'utiliser une carte SD en guise de moyen de stockage, aussi bien pour le système que pour les données. L'utilisation d'un même Raspberry Pi pour diverses applications peut se faire par un simple remplacement de la carte SD et un redémarrage du Raspberry Pi. Pour les expérimentations, NOOBS (New Out Of the Box Software) est parfait. C'est un système qui peut installer la ou les distributions de votre choix (en français). NOOBS gère le multiboot, ce qui facilite le passage d'un système à l'autre.

Sont disponibles dans NOOBS : Raspbian (une Debian adaptée), OpenELEC et RaspBMC des média-centers basés sur XBMC, RISC OS, ARCH LINUX et Pidora, une Fedora adaptée au Raspberry Pi. Il est possible de démarrer

Raspbian en ligne de commande ou en bureau LXDE mais aussi directement en Scratch. Le Raspberry Pi est très souvent utilisé en centre multimédia grâce à ses capacités Full HD. De plus, connecté en HDMI à un téléviseur, il respecte la norme CEC, qui permet de commander le média-center directement avec la télécommande de la télé. On élimine ainsi le clavier et la souris qui ne sont ni pratiques ni esthétiques dans un salon. Mais avec son processeur ARM et ses 512 Mo de mémoire, parfois avec un disque dur USB en complément, on le retrouve comme serveur web, cloud personnel, ordinateur d'appoint pour la bureautique (LibreOffice), pilote d'une installation domotique (commande de volets, alarme, régulation de température, télésurveillance...), outil d'apprentissage de la programmation...

Une communauté active

On constate l'intérêt porté au Raspberry Pi par l'activité de sa communauté. Le forum <http://www.raspberrypi.org/forums> est un lieu d'échange très actif qui propose un espace en français. Une association créée en France <http://www.raspfr.org> met également en avant la promotion du Raspberry Pi et son forum traite de la robotique, de la domotique, des logiciels... Un groupe du CERN, inquiet de la dégradation des compétences en informatique de la génération actuelle, et s'appuyant sur le rapport alarmant de l'Académie des Sciences publié en mai 2013 : **L'enseignement de l'informatique en France : Il est urgent de ne plus attendre** (http://www.academie-sciences.fr/activite/rapport/rads_0513.pdf) se propose de promouvoir l'étude de l'informatique et de la programmation auprès des établissements scolaires et du grand public. Le Raspberry Pi pourrait être un support peu coûteux pour ce programme.

L'apprentissage de la programmation dès 7 ans

Fig.2.

Conçu dès le début dans un but éducatif, le Raspberry Pi embarque le langage Scratch, développé par le MIT. Ce logiciel libre orienté multimédia permet de programmer des histoires interactives, des jeux et des animations puis de les partager en ligne. Riche de plus de 5 millions de projets, le site <http://scratch.mit.edu> est une mine d'idées et de références. De nombreuses ressources sont disponibles en ligne ou au format papier pour faciliter le démarrage en Scratch et accompagner les enseignants. Accessible à partir de 7 ans, Scratch s'interface aussi avec le monde réel. Il est capable de recevoir des informations provenant de capteurs, et de réagir en pilotant des actionneurs connectés aux sorties du Raspberry Pi, ou à une carte d'extension. Le code est écrit sous une forme visuelle, dans la langue maternelle de l'enfant (une vingtaine de langues européennes disponibles). Des briques de couleurs (contrôles en jaune, variables en rouge, mouvements en bleu...) facilitent la création des programmes et leurs formes rendent impossibles les erreurs de syntaxe. Scratch permet d'aborder l'algorithmique très tôt mais est utilisable aussi bien en collège, lycée, qu'à l'université. Lors de la journée Raspberry Pi du CERN, j'ai pu observer la facilité avec laquelle des adolescents qui n'avaient jamais pratiqué Scratch étaient capables de prendre le langage en main et de piloter des équipements (véhicule, bras robot...) Fig.3.

Pour des applications professionnelles

Destiné à l'éducation mais très vite adopté pour d'autres usages, le Raspberry Pi a su petit à petit se faire une place dans le monde professionnel.

Enseignement : Olivier, enseignant dans un collège de l'Ain a découvert le Raspberry Pi grâce à un ami enseignant. Il l'utilise (ou l'utilisera) pour la formation des élèves à l'informatique, mais aussi pour des applications de mesures physico-chimiques, comptage... Il envisage également des applications robotiques et multimédia. Olivier qui a déjà écrit des programmes (assembleur, C++, Lego Mindstorm) envisage bien entendu de développer des applications avec le Raspberry Pi. Le prix du Raspberry Pi a été un facteur déterminant lors du choix.

Domotique : La société **Oyoma** (94 Saint-Maur des Fossés) a créé un portier vidéo open source, basé sur un Raspberry Pi et sa caméra, baptisé **NovoSip**. Un client SIP est installé dans le Raspberry. Il est mis en relation avec une tablette ou un smartphone sous Android lorsqu'un visiteur appuie sur le bouton d'appel, par un serveur Asterisk embarqué. L'application Android est fournie gratuitement. Elle permet de voir le visiteur et d'ouvrir la porte (ou pas !) en composant un numéro en fréquences vocales (DTMF) sur le clavier du smartphone. (<http://www.novosip.com>)

Capteurs et interfaces industriels : Présentée par **RoboteQ** (Scottsdale, Arizona) la carte **RIO-IO** se connecte sur le GPIO du Raspberry Pi. Elle permet d'alimenter l'ensemble Raspberry Pi + carte d'extension sous une tension comprise entre 10 et 40 volts. Dotée de son propre processeur ARM 32 bits, elle décharge le Raspberry Pi de la gestion des entrées sorties et gère de façon autonome la capture, la conversion, le filtrage et la mise en forme des informations recueillies sur les E/S. La carte est dotée d'un connecteur JAE-MX34 à verrouillage, adapté à des applications embarquées soumises à des vibrations,

changements de températures... comme l'automobile, la robotique, les chaînes de production automatisées. Elle intègre des interfaces CAN et RS232 et peut être équipée en option d'une carte intégrant un accéléromètre 3 axes, un gyroscope 3 axes et un magnétomètre 3 axes. Les algorithmes fournis donnent accès aux informations de cap et d'assiette.

(<http://www.roboteq.com/index.php/roboteq-products-and-services/navigation-computers>)

Développement, programmation : La startup **Weenove** (33 – Bordeaux) développe le logiciel décisionnel **Biwee** qui permet à l'entreprise d'exploiter toutes les données issues de ses applications internes (gestion commerciale, CRM, comptabilité, ERP...). La problématique de Weenove était la mise en place d'un parc informatique pour réaliser du développement sous Visual Studio 2013, SQL Server. Ces activités nécessitent une certaine puissance et de l'espace mémoire lors des compilations. L'objectif de l'étude dans laquelle le Raspberry figurait était de réduire :

- ▷ les coûts matériels
- ▷ le temps d'installation du système et de sa maintenance
- ▷ la consommation énergétique
- ▷ la nuisance sonore

La solution retenue devait également occuper peu d'espace et garder un style visuel épuré sur les bureaux **Fig.4**. Ayant eu l'occasion de travailler auparavant avec des terminaux légers, c'est vers cette solution que les créateurs de l'entreprise se sont tournés, en incluant le Raspberry Pi dans leur étude. L'idée de base repose sur un serveur puissant tournant sous Windows 2012 et permettant l'accès multisessions, sur lequel des clients passifs se connectent en bureau à

distance. Finalement, la solution retenue pour chaque poste de travail est la suivante :

- ▷ Raspberry Pi Model B overclocké à 1GHz
- ▷ Carte mémoire SD – 8Go – SanDisk Ultra - Classe 10
- ▷ Boîtier transparent SB components
- ▷ Ensemble souris-clavier sans fil Logitech MK270
- ▷ Ecran Full HD HP 23» – Modèle 23xi
- ▷ Alimentation 5V/1,5A micro-USB pour Raspberry Pi
- ▷ Câble HDMI 1m
- ▷ Câble RJ45 2m

Après plusieurs essais, c'est la distribution Raspbian qui a été choisie. L'installation a été faite avec NOOBS. Un script se lance au démarrage. Il demande le login et le mot de passe de l'utilisateur, puis lance XfreeRDP en plein écran, avec les options adéquates. Pour l'utilisateur, tout se passe comme si son ensemble écran/clavier/souris était directement relié au serveur. Une image de la carte SD prête à fonctionner permet, en quelques minutes, d'avoir à disposition un nouveau poste de travail. Pour faciliter la maintenance, chaque Raspberry Pi est dans un boîtier de couleur différente et il est relié au réseau par un câble de la même couleur que le boîtier **Fig.5**.

Le coût de cette infrastructure est de 960€ TTC pour le serveur, et de 250€ TTC pour chaque poste de travail. Pour une installation de 8 postes, le coût de chaque poste de travail, serveur compris, est de 370€. Le serveur a de la puissance en réserve et pourrait supporter plus d'utilisateurs, d'autant que les compilations ne se produisent pas simultanément.

L'étude a également porté sur la consommation électrique, qui entraîne des coûts de fonctionnement conséquents. Des mesures précises ont été réalisées : Chaque Raspberry Pi consomme 2,5 watts (à comparer à la centaine de



Fig.3

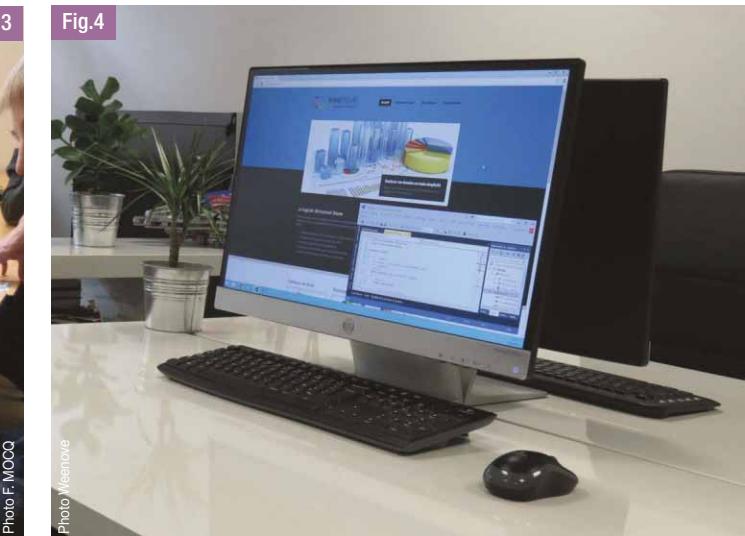


Fig.4

Commande de véhicule ou de bras robot, le langage Scratch tenait la vedette lors de la journée Raspberry Pi du CERN le 12 avril 2014

Chez Weenove, les bureaux sont totalement exempts des fils qui encombrent habituellement les espaces de travail. L'alimentation de l'écran et son câble HDMI sont dissimulés par le support de l'écran, laissant le bureau totalement libre.

watts consommés par un PC de bureau) et la consommation mesurée des écrans est de 24 watts. Les Raspberry Pi dépourvus d'interrupteur sont alimentés en permanence, alors que les écrans sont éteints lorsqu'ils ne sont pas utilisés.

Les limites notées par les utilisateurs sont en premier lieu l'absence du dual-screen habituel pour les développeurs ainsi que quelques saccades peu gênantes lors des défilements d'écrans. Les vidéos YouTube sont saccadées lorsqu'on les affiche en plein écran.

Après 3 mois d'utilisation, les utilisateurs de Raspberry Pi chez Weenove se déclarent très satisfaits de cette solution. La mise en place d'un poste de travail est simplifiée à l'extrême et la maintenance pratiquement inexistante. L'absence de ventilateur sur le Raspberry Pi assure une ambiance calme dans les locaux malgré la présence de plusieurs postes de travail. Les startups voisines de Weenove qui ont eu connaissance de ce déploiement se montrent très intéressées et la solution Raspberry Pi devrait faire des émules prochainement.

Banc test industriel :

La société BSE Electronic (<http://www.bse-electronic.com> - 71 -

Le Creusot) conçoit et produit des cartes et équipements électroniques, en particulier pour le médical. Elle a récemment investi 1.3M€ dans un outil industriel de dernière génération capable d'insérer jusqu'à 8 millions de composants par mois, et continue en 2014 les investissements tant en production qu'en R&D. Cet investissement a permis de relocaliser au Creusot des productions fabriquées en Chine, tout en réalisant des gains substantiels pour le client concerné. La stratégie de l'entreprise a permis la création d'une vingtaine d'emplois locaux, et une vingtaine de nouveaux emplois en R&D doivent être créés dans les prochaines années. Le département R&D recrute actuellement 3 ingénieurs, deux en Software (C embarqué, java...) et un en hardware. BSE Electronic vient de recevoir le label **PME Attractive** (<http://www.pme-attractive.fr/pme/16-BSE-Electronic.html>).

Parmi les productions de BSE Electronic figurent des cartes de gestion d'automatisme de couverture de piscine. Ces cartes sont construites autour d'un microcontrôleur NXP. BSE Electronic est l'une des rares entreprises françaises certifiées **Independent Design House** par NXP et la seule disposant d'une unité de production. Les cartes à base de microcontrôleur facilitent les opérations en apprenant le nombre de tours moteur nécessaires pour couvrir la piscine. Trois boutons poussoirs permettent ensuite de commander facilement le dispositif. Pour tester ces cartes de



Fig.5

Dissimulé sous la table le Raspberry Pi est maintenu par un adhésif double face

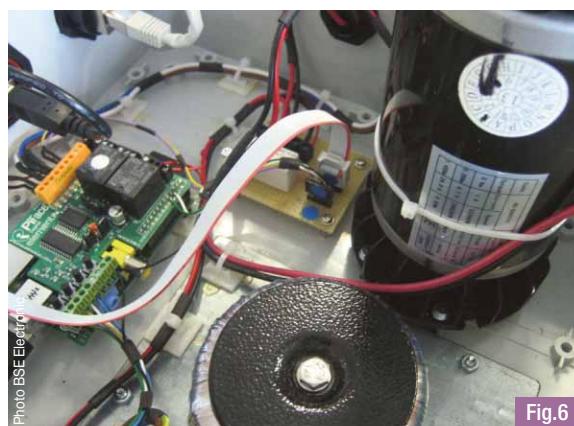


Fig.6

Raspberry Pi dans le banc test de BSE electronic

commande moteur, BSE Electronic a été amenée à réaliser un banc test avec des contraintes particulières : en raison d'un nombre important d'opérations manuelles réalisées sur les cartes, le montage est effectué sur le site Tunisien de la société. Le banc test devait pouvoir être facilement exporté et sa maintenance rester très simple. Ses composants devaient être marqués CE pour permettre le retour du matériel en France. Le client ne souhaitait pas investir dans la conception du banc test et voulait un matériel simple et peu onéreux. Alors que certains bancs tests réalisés par BSE dépassent les 100K€ celui-ci devait être le moins cher possible. Le choix du Raspberry Pi a permis d'aboutir à une solution extrêmement compacte. Il est notamment possible de renvoyer le système informatique complet... dans une enveloppe !

Ces critères ont amené à la réalisation d'un banc test basé sur un Raspberry Pi qui gère la totalité du système. Il se connecte en SSH sur le serveur situé en France pour reporter les informations disponibles (nombre de tests, identifiant des cartes testées, défauts détectés...). Un serveur web permet de se connecter sur le banc test en cas de besoin, pour une intervention plus pointue sur une carte défectueuse. Le Raspberry Pi pilote également l'imprimante Zebra chargée d'imprimer les étiquettes permettant la traçabilité des opérations (Tests OK ou erreurs relevées, N° de série unique de la carte testée...).

Le test complet d'une carte comprend 3 étapes :

► Le programme de test est chargé dans le microcontrôleur de la carte et il est exécuté.

► Si la première étape s'est bien déroulée, la carte passe au résinage pour garantir sa tenue dans les conditions d'utilisation (à proximité d'une piscine). Elle revient ensuite sur le banc pour vérifier que le résinage n'a pas provoqué de dysfonctionnement.

► Le programme définitif est chargé dans le microcontrôleur puis le test final est réalisé.

Chaque test est lancé par appui sur un bouton poussoir et deux LED indiquent le déroulement puis le résultat de chaque test. L'interface côté opérateur reste donc très simple Fig.6.

Le banc test a été réalisé avec :

- un Raspberry Pi Model B (512Mo de RAM)
- une carte PiFace
- système d'exploitation Raspbian
- programme développé en Python en moins de 3 jours (1500 lignes)
- serveur web en Flask, un microframework pour Python

► programmation des chips NXP avec Ipc21isp (http://sourceforge.net/projects/_ipc21isp)

Dans ce cas le Raspberry Pi a permis de répondre au souhait du client qui voulait que le banc test conserve un coût faible. Malgré cette limitation, les possibilités du Raspberry Pi alliées à la carte PiFace ont permis d'assurer la qualité de la réalisation et la traçabilité de toutes les opérations de test. Le matériel est opérationnel depuis presque un an en Tunisie, dans l'environnement d'un atelier de production, sans aucune défaillance.

Client léger : A l'hôpital **Jean Bouvery** (71 - Montceau les Mines) ce sont actuellement 70 Raspberry Pi qui sont déployés en clients légers. En proie à de sérieuses difficultés financières, l'établissement a dû «trouver une solution pour remplacer économiquement le parc informatique, tout en garantissant un outil performant pour les utilisateurs», a déclaré Sylvie Ducharme, directrice du système d'information du groupement hospitalier Le Creusot-Montceau, lors d'un atelier de retour d'expérience sur le thème «système d'informations et développement durable» aux journées Athos. (source : directhopital.com - <http://bit.ly/RXuK4J>). Ce sont donc les considérations financières qui ont conduit à la mise en place de Raspberry Pi comme terminaux légers. Chaque poste coûte environ 170 € alors qu'une solution mettant en œuvre des clients légers classiques ou des PC varie de 250 € à plus

de 500 € pièce. Le service des urgences a été équipé en premier, ce qui a représenté une dizaine de postes en juillet 2013. Au vu des résultats il a été décidé de généraliser cette solution. Les autres services sont équipés au fur et à mesure que le matériel doit être remplacé. Aux 70 équipements en service actuellement viendront bientôt s'ajouter 35 autres destinés à équiper des chariots autonomes. Les chariots utilisés aujourd'hui dans les services de l'hôpital par le personnel chargé des soins et de la distribution des médicaments sont équipés d'un ordinateur portable. Les batteries ont une autonomie de 4 à 5 heures. Elles durent environ 6 mois. L'écran de petite taille est une source d'inconfort pour les utilisateurs. Chaque chariot coûte environ 4000 € sur 3 ans. Les nouveaux modèles (en cours de développement) seront équipés d'un Raspberry Pi et d'un écran de grande dimension. Une batterie lithium fer phosphate (LFP) de dernière génération (garantie 5 ans) fournira une autonomie de 10 à 12 heures. Elle présente des caractéristiques de charge bien plus intéressantes que les modèles actuels (charge à 80% très rapide, pas d'effet mémoire). La réduction du poids des éléments supportés par le chariot permet de choisir pour celui-ci un modèle plus «light» donc plus économique. Au final le coût du chariot est ramené à 1500€ sur 3 ans avec un confort d'utilisation supérieur. Un client sera installé sur le Raspberry Pi pour remonter des informations sur les cycles charge/décharge et sur l'état des batteries afin de mettre en place une maintenance préventive. Ces informations seront traitées par un outil de supervision type Nagios. L'établissement hospitalier utilise **AppliDis Fusion de Systancia** (<http://www.systancia.com/fr>), qui virtualise poste de travail et applications. Le prestataire a dû recompiler le client léger pour le Raspberry Pi. Dans cette utilisation, les serveurs gèrent les applications et le Raspberry Pi ne fait que recevoir les fenêtres avant de les afficher. Lorsqu'un Raspberry Pi se connecte, un premier serveur de load-balancing lui retourne le numéro du serveur d'applications auquel il doit se connecter. Ceci permet de gérer l'équilibrage des charges entre les serveurs. Le client léger accède alors au serveur qui lui a été attribué. C'est actuellement **rdesktop** qui équipe les Raspberry Pi. Une migration vers **FreeRDP** est prévue. Les affichages sont plus fluides, mais surtout **FreeRDP** accepte en natif le flux USB, ce qui permet par exemple d'utiliser directement les pédales utilisés par les secrétaires chargées de la rédaction des comptes-rendus à partir d'enregistrements vocaux. Toutes les applications utilisées dans le centre hospitalier sont supportées sans que les utilisateurs n'éprouvent de gêne. En particulier les applications d'imagerie médicale ne posent aucun problème, que ce soit avec la

radiologie, l'IRM ou les scanners. Le défilement des coupes de scanner avec la roulette de la souris est normale. Comme il n'y a pas de vidéo, les saccades constatées sur cette dernière n'ont pas remis en cause le choix du client léger. Seuls quelques PC sont encore en service car des patients extérieurs amènent parfois les images sur DVD et il faut pouvoir les lire. La phase de préparation des terminaux est réduite à sa plus simple expression. La carte SD est créée à partir d'une image puis elle est personnalisée (nom d'hôte). En quelques minutes le nouveau poste de travail est opérationnel. La maintenance a également été simplifiée. Si le démarrage a été un peu houleux, cela provenait surtout d'une méfiance à l'égard de ce «petit boîtier» manifestée par des personnes qui ne faisaient pas la différence entre la partie applicative et l'installation du Raspberry Pi. En fait une minorité des défauts relevés étaient dus au client léger. Les interventions sont simplifiées. Les problèmes sont souvent fonctionnels et pris en charge au téléphone. Un client VNC autorise la connexion au Raspberry Pi en partage d'écran avec l'utilisateur. Il est également possible de se connecter en SSH pour lancer une mise à jour par exemple. Pour les mises à jour importantes, le service informatique dispose de cartes SD qui sont préparées à l'avance et simplement échangées sur les clients légers, ce qui est très rapide. Les cartes SD récupérées permettent de renouveler l'opération jusqu'à mise à jour de tous les Raspberry Pi. Si un technicien doit intervenir sur site, il part en intervention avec une petite mallette contenant un Raspberry Pi, une alimentation, des câbles et quelques cartes SD. Les temps d'interventions sont réduits et la réussite des interventions est pratiquement assurée contrairement à ce qui se passait avec les PC. Les utilisateurs ont constaté une amélioration des conditions de travail car le bruit de ventilation a complètement disparu, ce qui est surtout notable dans les endroits où fonctionnaient de nombreux PC. La surface des bureaux est libérée : les Raspberry Pi sont fixés sous les bureaux. Ceci a également éliminé le problème des vols d'unités centrales... Avec une consommation de 2 watts chacun, les 100 terminaux légers consomment 200 W au total soit l'équivalent de 2 PC. Les 100 PC remplacés consommaient quant à eux 10 kW. La facture annuelle d'électricité liée au parc informatique devrait donc décroître fortement. L'objectif fixé au départ a été atteint :

- ▶ Réduction des coûts
- ▶ d'achat du matériel
- ▶ des stocks
- ▶ de la maintenance
- ▶ de la consommation électrique

sans que la qualité du service rendu aux utilisateurs soit impactée. Au-delà de la mise en

service des chariots autonomes pour le personnel soignant, le service informatique envisage de mettre en œuvre le Raspberry Pi pour des systèmes de contrôle d'accès RFID utilisant les cartes professionnelles dont dispose le personnel. (<http://www.ch-montceau71.fr/raspberry>).

Ce dernier exemple montre qu'il est possible de déployer le Raspberry Pi dans le monde professionnel avec des objectifs d'économie, tout en préservant la qualité de service.

Une nouvelle version tournée vers l'industrie

Annoncée début avril, la nouvelle version du Raspberry Pi, appelée **Compute Module** cible une clientèle différente. De la taille d'un module mémoire équipant les ordinateurs portables ou les cartes mères micro-ATX, cette carte est au format DDR2 SODIMM (*small outline dual in-line memory module*). Elle mesure 67,6 x 30 mm et embarque uniquement le SoC BCM2835 avec sa mémoire de 512 Mo, ainsi qu'une mémoire flash eMMC qui est l'équivalent de la carte SD du Raspberry Pi classique. L'avantage de cette carte est qu'un nombre bien plus grand de ports et d'interfaces du SoC sont disponibles par rapport à la version actuelle. Une carte d'entrée/sortie de démonstration est prévue également mais elle est plutôt minimalistique. Cette carte *Raspberry Pi Compute Module* au format SODIMM est avant tout destinée à ceux qui souhaitent développer leur propre carte d'entrée/sortie, c'est-à-dire les industriels qui vont pouvoir intégrer la plateforme Raspberry Pi très simplement dans leurs applications, tout en bénéficiant des ressources existantes dans cet environnement.

Une seule limite : l'imagination

Grand comme une carte de crédit – 86 x 54 mm, aussi lourd que 9 feuilles de papier A4 (80g/m²) – 45 grammes, plus silencieux qu'une carpe - 0 dB, le Raspberry Pi nous réserve certainement encore bien des surprises. En deux ans d'existence, il a déjà été mis à bien des sauces, mais la liste n'est certainement pas exhaustive. Les exemples ci-dessus montrent que la réduction des coûts est souvent un facteur important dans le choix du Raspberry Pi par l'entreprise. Mais son silence et sa facilité de mise en œuvre sont également des facteurs à prendre en compte lors du choix.

Et vous, qu'en ferez-vous?



○ François Mocq
Formateur Réseau-Télécom
AFPA Le Creusot -71
Site web : www.framboise314.fr
Auteur du livre : *Raspberry Pi : Exploitez tout le potentiel de votre nano-ordinateur* – Edts ENI

Comprendre les objets connectés avec Arduino

De nos jours, le concept d' "objets connectés" a envahi une part importante des innovations technologiques. Ces nouveaux systèmes embarqués et interconnectés donnent vie à bon nombre de nos objets quotidiens pour nous apporter confort, sécurité, loisir... Tour d'horizon des outils et techniques sur lesquels ils reposent.

Arduino est une famille de cartes électroniques opensource (libres) destinées à l'apprentissage et au prototypage. Elle offre une grande variété de puissances, tailles et usages afin de s'adapter à tous types de montage. La plus connue est l'Arduino Uno : elle se compose de 14 entrées/sorties numériques et 6 analogiques. Cadencée par un cristal à 16MHz, elle permet déjà de réaliser des opérations à haute fréquence. Son cerveau est un microcontrôleur fabriqué par la société Atmel, leader dans le domaine, offrant 32Ko de mémoire pour le programme et 2Ko de mémoire SRAM (équivalent de la mémoire vive). Elle est alimentée par une entrée régulée à 5V : n'importe quelle pile plate 9V ou assemblage de piles AAA suffit à la rendre opérationnelle. Si ces caractéristiques ne sont pas suffisantes pour les besoins d'un projet, il existe aussi l'Arduino Mega : 54 entrées/sorties numériques, 16 analogiques, 256Ko de mémoire pour le code et 8Ko de mémoire vive. Equipé d'une interface USB pour la programmation, tout comme le Uno, il dispose cependant de 4 ports d'entrée/sortie série (Rx/Tx) contre 1 seul pour le Uno Fig.1. Besoin de plus de puissance? Il est possible de passer dans la cour des grands grâce à l'Arduino Due : un vrai processeur ARM 32 bits (nous reviendrons là-dessus plus tard), également utilisé par exemple dans les smartphones ou tablettes, cadencé à 84MHz! Ces processeurs, toujours fournis par Atmel, sont spécialement optimisés pour consommer moins d'énergie, d'où une alimentation de la carte en 3.3V. Enfin la mémoire de 512Ko pour le code et 96Ko de SRAM offrent de vraies perspectives. Toujours pas convaincu? Un besoin spécifique? Pas de problème : miniaturisme avec le Nano, vêtement connecté avec le Lilypad, système d'exploitation embarqué avec le Yún, une version motorisée avec le Robot ou encore une carte préparée pour le jeu ou tout type de commande (joystick) avec le Esplora... Presque tout est prévu pour accélérer et simplifier la mise en route d'un projet.

Un peu d'électronique 1/2 : Interfaces de communication

Il est important d'appréhender quelques concepts inhérents à la gestion de la communication entre

tous les composants d'une carte telle que présentée précédemment afin de comprendre comment les utiliser efficacement.

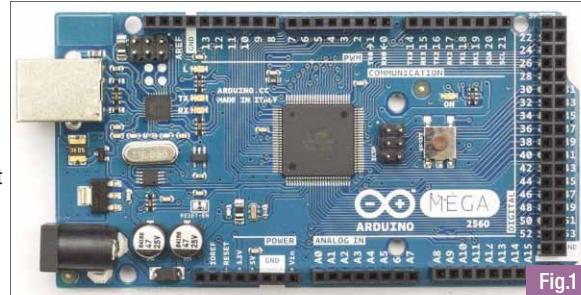


Fig.1

Bus I²C

Cette norme (Inter Integrated Circuit) désigne un bus de données série synchrone bidirectionnel half-duplex (échange dans les deux sens alternativement). Tous les équipements qui lui sont reliés sont tour-à-tour Maître ou Esclave. L'ouverture d'un échange est toujours à l'initiative du Maître, qui impose donc la vitesse d'horloge, vers un seul Esclave (les rôles peuvent être inversés à tout moment). Le dialogue est réalisé au travers de 2 lignes : ▶ SDA (Serial Data Line) : ligne de données bidirectionnelle ▶ SCL (Serial Clock Line) : ligne d'horloge de synchronisation bidirectionnelle La vitesse de communication autorisée se situe généralement dans une fourchette assez large d'environ 50Kbit/s jusqu'à 5Mbit/s. Ainsi, sur une carte Arduino par exemple, ce bus sert à relier au microcontrôleur différents périphériques : carte SD, écran LCD, moteur, un autre Arduino...

Bus SPI

Sur le même principe que l'I²C, ce bus (Serial Peripheral Interface) permet cette fois-ci une communication full-duplex (échange dans les deux sens simultanément). Pour ce faire, il utilise 4 lignes : ▶ SCLK (Horloge) : géré par le maître ▶ MOSI (Master Output, Slave Input) : géré par le maître ▶ MISO (Master Input, Slave Output) : géré par l'esclave ▶ SS (Slave Select) : géré par le maître Cette liaison peut être utilisée pour programmer un microcontrôleur grâce à un câble ISP. Sur les cartes Arduino, le protocole debugWire (créé par Atmel) permet quelques fonctions de debug via cette interface, bien que limitées.

Bus JTAG

Similaire au bus SPI, il permet cependant un accès auxiliaire aux broches d'entrées/sorties, afin de pouvoir observer leur état en cours d'utilisation. Il se base sur 5 lignes :

- ▶ TMS (Test Mode Select) : signal d'activation de la communication JTAG
 - ▶ TCK (Test Clock) : horloge
 - ▶ TDI (Test Data Input) : entrée des données
 - ▶ TDO (Test Data Output) : sortie des données
 - ▶ TRST (Test ReSet) : réinitialisation
- Les broches TDI et TDO peuvent être branchées en série afin de réaliser des chaînes de composants à observer (appelées "Daisy Chain") à partir d'une seule interface JTAG. En plus de permettre la programmation du microcontrôleur, cette interface est largement préférée pour le debug car beaucoup plus réactive et étudiée pour ce besoin. Là encore, un câble spécifique sera nécessaire.

Un peu d'électronique 2/2 : Architectures des microcontrôleurs

Avant d'aller plus loin, il est recommandé de se documenter sur quelques concepts fondamentaux impliqués dans le fonctionnement d'un microcontrôleur.

Les interruptions

L'intérêt principal d'un système embarqué n'est pas seulement d'exécuter séquentiellement un programme, mais bien de savoir réagir à tout type de stimuli provenant de son environnement. Pour ce faire, la grande majorité des microcontrôleurs fournissent des mécanismes d'interruption : le programme en cours est stoppé pour exécuter, de manière parallèle, une routine (ISR - Interrupt Service Routine) spécifique au stimulus reçu. L'état du processus principal est sauvegardé en mémoire avant l'entrée dans la sous-routine pour permettre de retrouver le contexte initial lors de la reprise de l'exécution normale Fig.2.

Watch Dog Timer

Comment réagiraient les ingénieurs de la NASA si le robot Rover venait à s'immobiliser, sur Mars, victime d'un bug bloquant? Bien sûr, tout est



prévu : c'est le rôle de ce timer, interne au microcontrôleur et complètement indépendant, qui détecte les temps morts dans l'exécution du programme et déclenche, si besoin, la réinitialisation du système. En exécution normale, le timer principal (System Tick) est chargé de remettre à zéro le Watch Dog à chaque tour d'horloge.

Les registres

Ils représentent des emplacements mémoire pré-définis contenant l'état des différents composants du système (périphériques, mémoire, configuration...). Par exemple, un port de broches d'entrées/sorties (comme les ports B, port C...) est contrôlé par plusieurs registres Fig.3 :

- ▷ **PIO_PER** (Enable register) : pour activer les broches
- ▷ **PIO_PDR** (Disable register) : pour désactiver les broches
- ▷ **PIO_OER** (Output Enable register) : pour définir les broches en sortie
- ▷ ...

Chacun possède environ 30 bits permettant d'accéder à chacune des broches qui lui sont reliées : Fig.4. Ainsi pour allumer, par exemple, une LED branchée sur la broche 27 du port B, il faut écrire "1" sur le bit 27 du registre Enable du port B :

```
PIOB->PIO_PER = 1 << 27
```

(on trouve parfois : `1 << 27` désigné par `PIO_PB27`)

De même, pour activer plusieurs broches (27 et 29) :

```
PIOB->PIO_PER = 1 << 27 | 1 << 29
```

Les fusibles

Dans le monde des microcontrôleurs, les fusibles (fuses en anglais) sont semblables au BIOS d'un

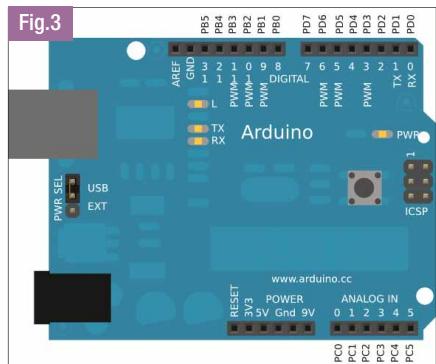
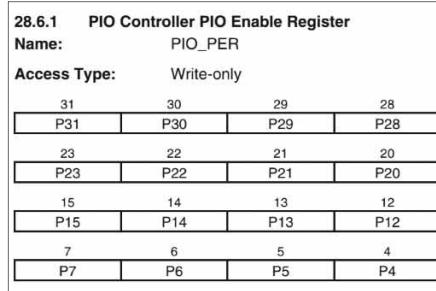


Fig.3



ordinateur : une configuration persistante (hors tension) sur quelques octets qui permet de déterminer l'état de la carte au démarrage (vitesse d'horloge, tension de fonctionnement, interfaces actives...). Ces informations bas-niveau peuvent être modifiées pour adapter le comportement de la carte à un usage particulier, mais avec précaution cependant.

Différenciation AVR/ARM

La plupart des cartes Arduino sont équipées d'un microcontrôleur utilisant une architecture AVR 8 bits. Elle fonctionne à partir d'instructions exécutées sur un seul cycle d'horloge et fournit les interfaces vers les composants de base : registres, mémoire, timers, ports d'entrées/sorties, communication série JTAG ou SPI, synchronisation d'horloge avec un oscillateur interne ou externe... La fréquence de fonctionnement de ces puces atteint le plus souvent 20MHz Fig.5.

L'architecture ARM quant à elle vise à la réalisation de microprocesseurs 32 bits (voire 64 bits), déclinées sous différentes familles selon les usages : par exemple, les Cortex-R destinées aux systèmes temps-réel, les Cortex-M pour le couplage avec un microcontrôleur... Un processeur graphique (GPU) est également intégré, offrant une puissance de calcul plus importante. Leur fréquence de fonctionnement est bien plus élevée, située entre 50MHz et 400MHz, tout en étant optimisés pour réduire leur consommation, de l'ordre de 0.1mA/MHz (notamment via des modes de veille, idéal pour les systèmes embarqués). Ces microprocesseurs sont donc plus indiqués pour la réalisation de systèmes complexes comme un serveur web, une unité de calcul, un afficheur sur grand écran...

Côté programmation

Ces microcontrôleurs sont optimisés pour exécuter du code en langage C. Il en effet idéal pour ce genre de plateforme : plus haut niveau et plus

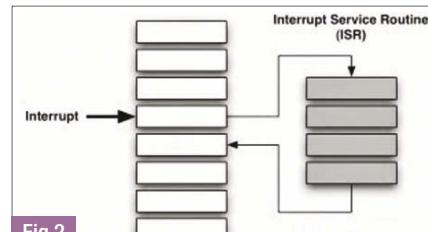


Fig.2

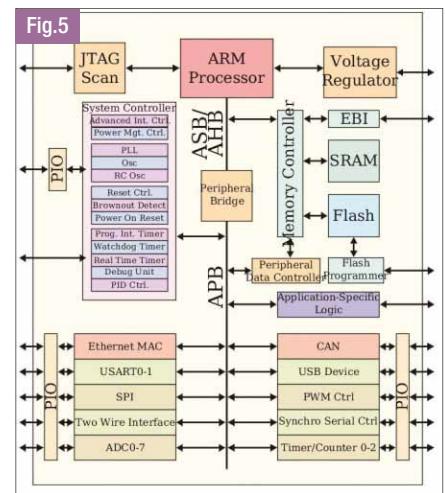
Fig.4

accessible que l'assembleur, il offre des facilités pour manipuler directement les emplacements mémoire, indispensable pour accéder aux registres et maîtriser l'état du microcontrôleur. De plus, toutes les librairies ou nouveautés apportées par le C++ restent superflues dans les environnements embarqués. Ces derniers requièrent également des chaînes de compilation (toolchain) spécifiques : en plus de compiler le code C en langage machine compréhensible par le microcontrôleur, elles permettent de configurer finement l'environnement cible (type de processeur, mémoire disponible...) et de connaître l'espace final occupé par le code compilé. Il peut également être nécessaire de paramétriser le type de programmeur utilisé et son interface (SPI, JTAG...) pour installer le code sur le microcontrôleur.

Présentation de l'environnement de développement Atmel Studio

La société fabricant les puces fournit également un IDE gratuit basé sur Visual Studio (donc destiné exclusivement à Windows) entièrement dédié au développement pour microcontrôleurs. Pour les autres systèmes d'exploitation, la meilleure alternative reste Eclipse et ses plugins pour les toolchains GNU.

L'intérêt principal est de pouvoir créer des projets, dans différents langages (C, C++, assembleur), adaptés à chaque système (AVR ou ARM). Ainsi, toute la chaîne de compilation est automatiquement optimisée pour coller aux spécifications techniques du matériel utilisé. Cet IDE intègre également un framework (ASF - Atmel Studio Framework) dont le rôle est de fournir toute la couche d'abstraction pour l'accès aux modules du microcontrôleur. Une large sélection d'outils pré-codés est donc disponible pour accélérer la réalisation de différentes fonctionnalités : accès à une carte SD, échanges de données par Ethernet, manipulation des timers... Comme tout bon IDE ne serait rien sans outil de debug, Atmel Studio apporte réellement dans ce domaine le savoir-faire de l'entreprise et sa connaissance du matériel. Outre bien sûr la mise en place de points d'ar-



rets, il est possible d'observer en détail l'état de la mémoire, la valeur de tous les bits dans les registres etc Fig.6. Il est donc possible de changer les valeurs à la volée pour modifier le comportement du système à chaud (pour simuler une interruption par exemple).

Outils pour debugger

Malgré tout cet attirail logiciel, debugger du code embarqué nécessite l'utilisation de quelques éléments matériels : en effet, ces fonctions de "supervision" ne peuvent être assurées que par un outil indépendant du système à tester. Et comme l'on peut s'en douter, ce matériel va varier en fonction notamment du microcontrôleur.

Debugger un AVR

Pour observer le déroulement d'un programme sur ce type de plateforme, il existe l' "AVR Dragon" de Atmel : il se branche au PC via une prise USB classique et à la carte cible par une interface SPI. Il sert à la fois de programmeur et de debugger : il autorise la création d'une trentaine de points d'arrêts dans le code et accélère son téléchargement vers la carte cible. Son point faible : le debug n'est optimal que jusqu'à un quart de la fréquence d'horloge du microcontrôleur observé, ce qui peut provoquer quelques lenteurs lors de l'exécution pas-à-pas.

Debugger un ARM

Pour cette architecture, il est possible d'utiliser par exemple le boîtier J-Link de Segger. Basé sur les mêmes principes que l'AVR Dragon, il sert d'intermédiaire entre le PC et la carte cible; mais cette fois l'interface avec cette dernière se fait par un câble JTAG. Le J-Link affiche des caractéristiques très séduisantes : support des processeurs multi-coeurs, un nombre illimité de points d'arrêt...

Quant à sa fréquence de fonctionnement, elle

peut atteindre sans problème plusieurs MHz sans perturber l'observation des variables ou des registres.

Enfin, il fonctionne quel que soit le système d'exploitation du PC ou l'architecture du système cible Fig.7.

Inspirations

Depuis plusieurs années, la baisse du prix et la variété du matériel électronique ainsi que l'accroissement des communautés de "bricoleurs" permettent à tout un chacun de se lancer et de créer des projets innovants.

Du matériel abordable

Que ce soit au niveau des cartes de prototypage toutes prêtées comme présentées dans cet article, du matériel élémentaire (fer à souder, câbles, résistances...) ou encore des périphériques divers

(capteurs, moteurs, diodes, écrans...), tout est aujourd'hui facilement accessible en magasin ou sur internet pour des budgets très raisonnables. De plus, les interfaces avec les moyens de communication courants (Ethernet, Wifi, Bluetooth...) et les accessoires utiles (cartes SD, stations météo, cartes sons, robots...) fournissent pléthore de services pour faciliter la réalisation de systèmes complexes avec très peu d'effort.

Des communautés actives

Comme dans tout autre domaine, chaque fabricant de puces s'est doté d'une forte communauté d'amateurs et professionnels apportant leur passion, et/ou savoir-faire, pour promouvoir le développement et participer à l'adoption du matériel et des outils. Ainsi, le nombre de forums et blogs spécialisés va crescendo, offrant à tous les autodidactes dans l'âme l'accès à des connaissances remarquables. Mais puisque rien ne vaut l'échange direct entre passionnés, le concept de FabLab (laboratoire de fabrication) vise à implanter un peu partout des ateliers bénévoles (parfois sponsorisés) de proximité, pour que tous ceux qui le désirent aient accès à du matériel professionnel et puissent partager lors de sessions de création publiques.

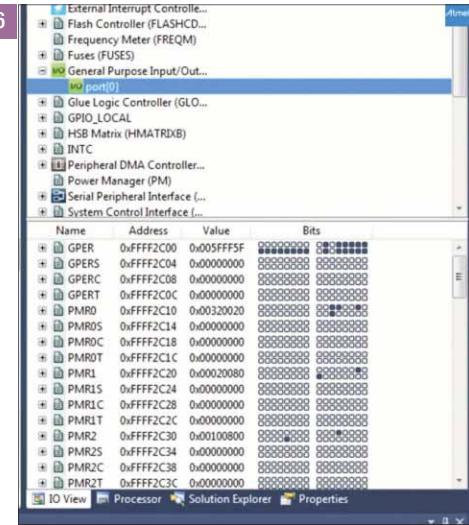
Vers l'Internet des objets

Depuis plusieurs mois, le terme "Internet of things" (IoT) occupe de plus en plus d'espace dans les colonnes des blogs ou magazines de tendances technologiques. Par exemple, la ville de Santander, en Espagne, s'est lancée dans le concept de "smart city" : capteurs de présence pour les éclairages publics, capteurs de proximité pour déterminer les emplacements de parking libres, mesure du poids des conteneurs à déchets pour optimiser les circuits de ramassage... Autre secteur qui a le vent en poupe : le jardin connecté (et par extension l'agriculture). Avec des mesures de paramètres vitaux (humidité, ensoleillement...) et de composition physiologique (pH, nutriments, maturation...), les cultivateurs amateurs ou professionnels disposent d'une nouvelle dimension révolutionnaire dans la pratique de leur activité.

Nouveaux défis

Mais étant encore très récentes, l'effervescence autour de ces nouvelles technologies conduit actuellement à de grandes divergences, dans les protocoles de communication (IPv6, Zigbee pour l'adressage réseau sur ondes radios, SensorML pour la description des données de mesures de capteurs, REST/SOAP pour les services web) ou même les systèmes d'exploitation : Android, Windows for IoT, LynxOS, FemtoOS... Quelques initiatives émergent des grandes entreprises pour essayer de créer des normes : "Industrial Internet

Fig.6



Consortium"(1) porté par Intel, IBM, AT&T, Cisco, General Electric etc. dont le but est d'accompagner les industries dans la création de processus et machines intelligents (voitures et chaînes de fabrication autonomes, médecine...); ou encore la DLNA(2) (Digital Living Network Alliance), un "futur" standard d'échange et de description de contenus multimédia entre divers appareils (tablette, TV, smartphone...). Ainsi, le développement des applications à base d'objets connectés va s'accompagner d'un grand nombre d'innovations dans l'industrie, visant à résoudre les nouvelles problématiques induites par de nouveaux usages : allier performance et miniaturisation, accroître l'autonomie des composants (notamment par de nouveaux moyens de récolte d'énergie - *Energy harvesting* - solaire, éolienne, vibratoire, cinétique...) et sécuriser les échanges d'information (cryptage et interception des paquets transmis à distance, protection du matériel en toute circonstance...). De nouvelles réglementations vont également voir le jour afin d'encadrer l'utilisation de ces objets. On a déjà pu le voir avec l'arrivée des drones et des premiers "accidents" qui ont déclenché bon nombre d'interrogations(3) jusqu'aux plus hautes instances du gouvernement. L'essor de la domotique est également actuellement dans le viseur des institutions(4).

Conclusion

Avec des domaines d'applications vastes et variés, porteurs à la fois d'ambitions et de questionnements, l'Internet des objets reste aujourd'hui un monde à inventer dans lequel chacun peut apporter son savoir ou son savoir-faire. Il ne vous reste plus qu'à choisir par où commencer!

● Paul-Emmanuel Faidherbe

Développeur Java curieux par nature, chez Valtech.fr
developpef.blogspot.fr

1 <http://www.iiconsortium.org/>

2 <http://www.dlna.org/>

3 <http://www.developpement-durable.gouv.fr/Drones-civils-loisir-aeromodelisme>

4 <http://www.ffdomotique.org/missions/>



Fig.7

Arozz'duino : Le système d'arrosage robotisé de vos plantes

Quoi de plus sympa que de retrouver vos plantes en pleine santé lors de votre retour de vacances. A la fin de cet article, vous saurez comment réaliser un système embarqué compact, extensible et à bas coût, qui palliera les besoins de vos plantes pendant votre absence.

Un système qui est centré ou qui repose sur une carte Arduino

Arduino est un petit circuit imprimé open hardware, c'est-à-dire que le schéma de routage de la carte est disponible gratuitement en licence libre. Sur cette carte, vous trouverez un microcontrôleur, l'unité programmable de cette carte. De plus, vous pouvez télécharger gratuitement l'environnement de développement à l'adresse suivante : <http://www.arduino.cc/>.

Le matériel nécessaire

- ▶ Un pot à réserve d'eau qui vous permettra de contenir l'eau nécessaire à l'arrosage des plantes pendant votre absence, pour cet exemple, nous avons choisi un pot d'une contenance de 32 litres.
- ▶ Un tuyau fin d'irrigation, des goutteurs en plastique, et des embouts de jonction, que vous trouverez dans des surfaces spécialisées en bricolage ou jardinage (le tout pour un budget d'environ 15€).
- ▶ Une pompe ou une électrovanne, que vous pourrez trouver dans n'importe quelle surface spécialisée en bricolage et jardinage pour moins de 20€. Pour cet article nous avons choisi une électrovanne. C'est cette dernière qui permettra l'arrosage de vos plantes Fig.1.
- ▶ Un capteur de niveau d'eau, qui nous permettra de surveiller le volume d'eau contenu dans le réservoir, pour éviter de faire tourner la pompe dans le vide, auquel il faudra ajouter une résistance d'1 kOhm, au total environ 10€.
- ▶ Une carte Arduino UNO qui sera le système intelligent de contrôle de notre dispositif. Elle devra contrôler l'humidité du sol et l'arroser en fonction des besoins de la plante. Il faudra simplement vérifier que le niveau d'eau dans le réservoir est suffisant pour autoriser l'irrigation Fig.2.
- ▶ Une shield Arduino Grove qui va nous permettre de connecter facilement nos sondes d'humidité et notre relais à notre carte Arduino Fig.3.
- ▶ 4 sondes d'humidité que l'on peut planter dans le sol afin d'analyser la teneur en eau du sol et ainsi diagnostiquer les besoins en eau des plantes Fig.4.
- ▶ Un relais, qui fera office d'interrupteur, contrôlé électroniquement par la carte Arduino afin de déclencher ou non le système d'arrosage
- ▶ Un transformateur 12V qui permettra d'alimenter notre système embarqué Arduino et le capteur

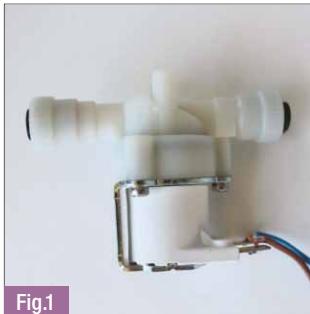


Fig.1



Fig.2

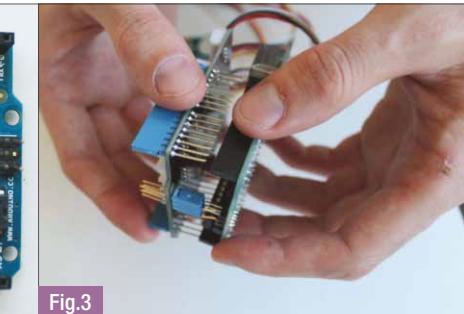


Fig.3

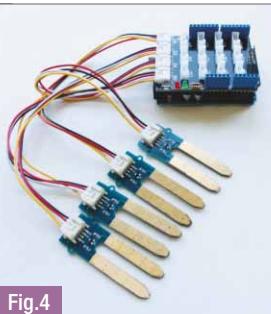


Fig.4

- ▶ Une multiprise afin d'y brancher la pompe ou l'électrovanne et le transformateur 12V.

Descriptif électronique des composants et montage électronique du système

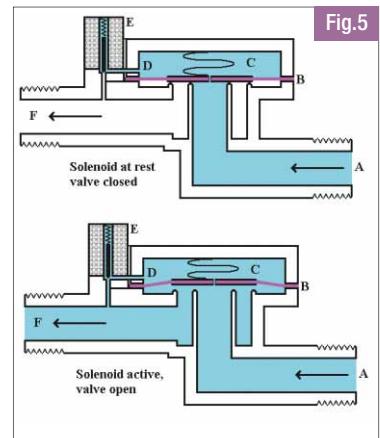
Avant de se lancer dans la programmation du système, intéressons-nous aux capteurs et actionneurs que nous allons utiliser pour en comprendre le fonctionnement.

Electrovanne :

Un petit pointeau poussé par un petit ressort appuie sur une membrane afin d'empêcher l'eau de passer et ainsi de fermer le circuit. Afin d'ouvrir le circuit et permettre à l'eau de s'écouler, il suffit d'alimenter la bobine afin de faire remonter le pointeau, celle-ci va l'attirer comme un aimant en ouvrant ainsi le circuit et permettant à l'eau de s'échapper.

Voici une coupe transversale

d'une électrovanne qui schématise le fonctionnement : Fig.5.
(Source : Wikipedia)



A- Arrivée B- Diaphragme C- Chambre de pression D- Conduit E- Électro-aimant F- Sortie

Le relais :

Le relais est un interrupteur électrique commandé électroniquement. Au lieu d'appuyer avec les doigts pour alimenter ou non votre système, un signal électrique (de 5 volts dans notre cas) permettra d'ouvrir ou fermer votre circuit électrique.

Lorsque nous décidons que nos plantes ont besoin d'eau, nous allons fermer le circuit électrique par l'envoi d'un signal 5V au relais ce qui mettra l'électrovanne en état de marche et permettra l'arrosage de nos plantes.

Une fois l'arrosage terminé nous ouvrirons le circuit électrique en cessant d'émettre notre signal 5V depuis la carte Arduino.

Sonde d'humidité

Cette sonde d'humidité est un capteur analogique, c'est-à-dire que la tension à ses bornes varie en fonction d'une grandeur physique, par exemple l'humidité du sol. Ce capteur mesure l'humidité du sol à partir des changements de conductivité électrique de la terre (la résistance du sol augmente avec la sécheresse). La carte Arduino UNO étant équipée de convertisseurs analogique vers numérique (CAN en français, ADC en anglais), nous allons pouvoir utiliser ces derniers afin d'analyser la teneur en eau du sol. Un CAN est un montage électronique permettant de transformer une tension, en une entité numérique facilement manipulable. La tension Vcc sous laquelle évolue ce capteur est de 5 volts, ce qui signifie que la tension aux bornes du capteur d'humidité va varier de 0V minimum à 5V maximum. Grace aux CAN de la carte Arduino, nous allons pouvoir manipuler très facilement cette grandeur. En effet cette dernière est équipée de CAN 10 bits, donc il nous suffit de récupérer la tension aux bornes de la sonde telle que 0V corresponde à 0 et 5 Volt corresponde au nombre 1023 [Fig.6](#). Le tableau ci-dessous fourni par le constructeur nous donne de bonnes indications quant à l'utilisation de ce capteur. Il nous dit que la valeur récupérée depuis nos fameux CAN est comprise entre 0 et 300 en milieu sec, 300 et 700 en sol humide et 700 et 950 en milieu aqueux. On comprend ainsi de mieux en mieux le principe de régulation de notre système. En effet, nous allons analyser en temps réel la teneur en eau du sol grâce à nos sondes d'humidité, si la valeur récupérée aux bornes de nos sondes nous semble trop faible par rapport à une valeur que l'on juge consigne (humidité minimum dans le sol), alors notre carte Arduino enverra un signal de 5 volts vers notre relais, ce dernier fermera le circuit électrique auquel est branchée notre électrovanne, permettant ainsi l'irrigation de nos plantes, jusqu'à un seuil consigne que nous estimons correct (humidité maximum du sol) [Fig.7](#). Même si le système est quasiment complet avec les composants précédents, je vous conseille d'ajouter un détecteur de niveau d'eau afin de diagnostiquer quand le niveau d'eau est trop faible dans notre réservoir pour pouvoir déclencher l'électrovanne ou la pompe. En effet vous risquez d'endommager la pompe si celle-ci se déclenche quand il n'y a plus d'eau dans le réservoir.

Capteur de niveau d'eau

Le capteur de niveau que je vous propose d'utiliser se comporte comme un simple interrupteur puisque la partie plastique libre est en fait un

Item
Voltage
Current
Output Value

flotteur. Quand celui-ci est plaqué le long de la tige, l'interrupteur est fermé et le courant peut passer au travers. Quand celui-ci est en position basse, l'interrupteur est fermé. L'idée ici est donc de détecter si le niveau d'eau est suffisant. Voici le montage électronique que je vous propose de réaliser afin de détecter l'évènement Niveau d'eau OK niveau d'eau NOK (not ok). Quand le niveau d'eau est bon, l'interrupteur est fermé et nous pouvons capter un niveau haut (5V) depuis notre carte Arduino. Si au contraire le niveau d'eau est faible, alors le flotteur est en position basse. Nous détectons un niveau bas (GND) depuis notre carte Arduino [Fig.8, 9 et 10](#). Les quatre sondes d'humidité sont reliées à 4 des pins analogiques de la carte Arduino. Le relais au pin Digital 3 de notre carte et le détecteur de niveau d'eau au pin numéro 10.

ARDUINO

Maintenant que nous avons le matériel configuré et mis en place, intéressons-nous à la partie programmation de ce merveilleux objet de bidouilleur qu'est la carte Arduino UNO.

L'Arduino UNO est un petit système embarqué, facilement programmable, basé sur un microcontrôleur ATMega328p. Si on s'attarde sur ses spécifications, on lira que sa tension de fonctionnement est de 5V c'est à dire qu'elle pourra assurer un niveau de tension de 5V sur ses broches numériques, et convertir une tension 5V sur ses broches analogiques. Le contrôleur possède une mémoire flash de 32KB, c'est l'espace maximal que pourra occuper le programme que vous allez y écrire. 2 KB de RAM, afin de stocker en mémoire les variables de votre programme. Il possède 1 KB d'EEPROM, une mémoire non volatile qui vous permet de sauvegarder en dur des informations qui vous sont nécessaires et une cadence de 16MHz, c'est à dire que l'ATMega328p est capable d'exécuter 16 millions

Fig.6

d'instruction en une seconde. Certes, on est bien loin de nos processeurs de PC multi-cœurs de plus forte puissance, mais son domaine d'application est tout autre. Nous allons effectivement pouvoir récupérer des informations de notre environnement (niveau d'eau et humidité dans le sol), les traiter et les contrôler (arrosage du sol). Pour réaliser cela, il vous suffit de télécharger l'EDI Arduino à l'adresse suivante : <http://arduino.cc/en/Main/Software>

INTERFACE DE L'EDI Fig.11.

Intéressons-nous à la manière dont nous allons réaliser le système d'arrosage dont nos plantes ont besoin afin de ne pas mourir de soif pendant notre absence. Pour cela nous allons avoir besoin de connaître quelques fonctions. Par exemple imaginons-nous un instant que nous ayons déclaré deux valeurs :

```
const int analogInPin_A0 = A0;
unsigned int A0_value=0;
```

Alors l'instruction suivante :

```
A0_value=analogRead(analogInPin_A0);
```

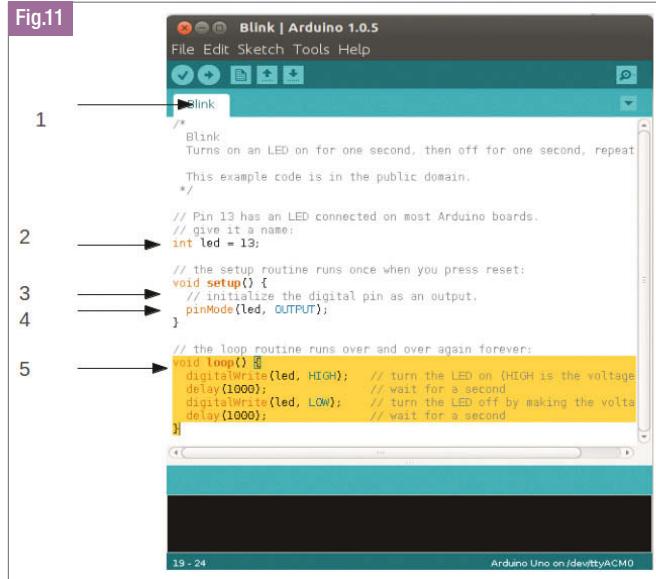
Aura pour effet de stocker dans la variable A0_value la tension entre A0 et GND, c'est à dire une valeur entre 0 et 1023, 0 correspond au même potentiel que la terre (en anglais GND - 0V) et 1023 à Vcc (5V).

Alors la solution à notre problème a du parvenir à votre esprit : en effet, si nous branchons nos 4 sondes d'humidité sur 4 des 6 broches analogiques que possède notre Arduino, alors il nous sera extrêmement facile de connaître en temps réel le taux d'humidité du sol. La base de notre programme est alors posée. La fonction loop étant répétée tant que le microcontrôleur est sous tension, l'appel au début de celle-ci d'une instruction telle que vue plus haut nous permettra de rafraîchir à chaque tour de boucle la variable contenant la quantité d'eau présente dans le sol de nos plantes. Nous pourrons ainsi subvenir à leur besoin en eau en appelant simplement l'instruction.

```
digitalWrite(pumpPin, HIGH);
```

Qui au lieu d'allumer une led, activera notre relais faisant ainsi passer la tension de 230 volts dans notre électrovanne et arrosera la terre. Quand nous aurons estimé que l'humidité du sol est revenue à un degré convenable l'appel de digitalWrite(pumpPin, LOW); coupera l'alimentation et arrêtera notre arrosage. Simple, vous ne trouvez pas ? Les plus attentifs d'entre vous se disent sans doute : « Qu'en est-il du risque d'endommager l'électrovanne si on essayait d'arroser avec un niveau d'eau insuffisant ? »

Fig.11



He bien oui, ce risque subsiste et c'est pour cela que nous allons avoir besoin d'une autre instruction :

Supposons que nous avons déclaré :

```
const int waterLevelPin=10;
```

et que nous avons appelé dans le setup() :

```
pinMode(waterLevelPin, INPUT);
```

Alors l'instruction suivante :

```
boolean tankIsFull=digitalRead(waterLevelPin);
```

sauvegardera en mémoire un booléen pouvant prendre deux valeurs : « true »(1) ou false(0). Dépendant de l'état en entrée de la broche numérique 10.

Si vous reprenez le schéma du détecteur de niveau d'eau, alors la solution saute aux yeux :

Quand le niveau d'eau est bon, l'interrupteur de niveau d'eau est fermé (flotteur à l'horizontal), la broche numérique 10 est donc reliée directement à Vcc, digitalRead(waterLevelPin) = HIGH=true=1.

Si le niveau d'eau est trop bas, le flotteur n'est plus horizontal, l'interrupteur de niveau d'eau est ouvert. La broche numérique 10 est donc reliée à la masse dans ce cas digitalRead(waterLevelPin) = LOW=false=0. C'est aussi simple que cela. Maintenant avant d'activer notre relais qui contrôlera l'électrovanne, il nous suffit de vérifier le flag tankIsFull. Si celui-ci est false alors on n'active surtout pas notre relais. Maintenant que nous avons vu les principales instructions dont nous aurions besoin, je vous propose de passer à la partie programmation.

PROGRAMMATION

Maintenant je vous propose de réaliser notre système d'arrosage automatique. Nous avons à notre disposition :

- 4 sondes d'humidité du sol, qui vont donner une valeur comprise entre 0 et 1023, correspondants au taux d'humidité présents dans le sol.
- Un relais; une fois activé ce dernier alimentera en tension notre électrovanne et nous permettra d'irriguer notre sol
- Un capteur de niveau d'eau qui nous indiquera en temps réel si notre réservoir contient assez d'eau pour pouvoir activer notre relais sans risque.

Notre but ici est donc de réaliser un algorithme de ce type :

Mes plantes ont-elles besoin d'eau ?

_OUI :

Le niveau d'eau dans mon réservoir est-il suffisant?

_oui: arroser()

_non : nothing()

_NON:

nothing()

A ce stade, je vous propose de réaliser des successions d'arrosage de 30 secondes (paramétrables), plutôt que d'arroser tant que le taux d'humidité voulu n'est pas atteint. Afin de ne pas biaiser nos mesures (laissons le temps à l'eau de pénétrer dans le sol avant de continuer d'arroser).

Voici le programme à réaliser :

```
#include <ColorLCDShield.h>
////////////////////////////
// variable           //
////////////////////////////
#define waterTimeOut 30 // le temps en secondes entre deux
arrosages consécutifs.
```

```

#define waterLevelH 600 // valeur d'humidité que nous
souhaitons atteindre en arrosant
#define waterLevelL 400 // valeur à partir de laquelle
nous souhaitons arroser
//Moisture sensor pin
const int analogInPin_A0 = A0; //Nos quatres sondes sont sur
les broches A0,A1,A2,A3
const int analogInPin_A1 = A1;
const int analogInPin_A2 = A2;
const int analogInPin_A3 = A3;

//digital pin
const int pumpPin=3; //Notre relais est relié à la broche 3
const int waterLevelPin=10; //Notre capteur de niveau à la 10

//environnement variable
unsigned int A0_value=0; //4 variables qui contiendront
la valeur
unsigned int A1_value=0; // pour chaque sonde d'humidité
unsigned int A2_value=0;
unsigned int A3_value=0;
unsigned int moy_value=0; // la valeur moyenne de nos 4 sondes

//prog variable
unsigned long pumpStartTime=0; // temps à partir duquel nous
aktivons le relais
unsigned long pumpStopTime=0; // temps à partir duquel nous
désaktivons le relais

boolean pumpFlag=false; // relais activé ou pas, faux à l'init

int currentWaterLevelLim=waterLevelL; //sur quelle valeur de
régulation sommes-nous.

    //Au début, nous souhaitons diagnostiquer la
    demande d'eau

void setup() {
  pinMode(pumpPin, OUTPUT);
  pinMode(waterLevelPin, INPUT);
  digitalWrite(pumpPin, LOW);
}

```

La broche du relais est déclarée en sortie, et ce dernier est désactivé au démarrage du programme. La broche reliée à notre capteur de niveau d'eau est déclarée en entrée. Tout d'abord, les 3 #define:

```

#define waterTimeOut 30
#define waterLevelH 600
#define waterLevelL 400

```

Celles-ci sont paramétrables et vont dépendre de vous. Après avoir testé mes sondes d'humidité, j'ai pour ma part estimé que le sol était suffisamment humide quand la valeur moyenne en sortie de mes 4 sondes d'humidité était de 600. J'ai également estimé que le sol était trop sec quand la valeur tombait en dessous de 400. Ces deux valeurs doivent d'être paramétrées en fonction de vos estimations personnelles, et non des miennes. Enfin j'ai choisi de réguler l'arrosage de façon à ne pas arroser pendant plus de 30 secondes. A la fin du temps imparti, un break de 30 secondes minimum est imposé qui recommencera jusqu'à ce que le taux d'humidité remonte. Voici la fonction loop telle que je me propose de vous la présenter :

```

void loop () {

  boolean tankIsFull=digitalRead(waterLevelPin);
  updateSensorValue();

  if(tankIsFull==true){
    waterManager();
  }
  else {
    digitalWrite(pumpPin, LOW);
    pumpFlag=false;
  }
}

```

On récupère dans tankIsFull le booléen représentant si le niveau d'eau est bon ou pas. On analyse les valeurs de nos sondes. Si le niveau d'eau est ok, alors on peut se permettre de contrôler notre électrovanne. Sinon, on la désactive.

```

updateSensorValue() :
void updateSensorValue(){
  A0_value=analogRead(analogInPin_A0);
  A1_value=analogRead(analogInPin_A1);
  A2_value=analogRead(analogInPin_A2);
  A3_value=analogRead(analogInPin_A3);
  moy_value=(A0_value+A1_value+A2_value+A3_value)/4 ;
}

```

On stocke dans nos variables les informations relatives à la teneur en eau du sol grâce aux CAN. moy_value est la valeur moyenne issue des 4 sondes

```

waterManager() :
void waterManager(){
  if (moy_value<=currentWaterLevelLim) {
    currentWaterLevelLim=waterLevelH;
    if(millis() - pumpStopTime > waterTimeOut*1000 && pumpFlag==false) {
      digitalWrite(pumpPin,HIGH);
      pumpFlag=true;
      pumpStartTime=millis();
    }
    if(millis() - pumpStartTime > waterTimeOut*1000 && pumpFlag==true) {
      digitalWrite(pumpPin,LOW);
      pumpFlag=false;
      pumpStopTime=millis();
    }
  }
  else{
    currentWaterLevelLim=waterLevelL;
    digitalWrite(pumpPin,LOW);
    pumpFlag=false;
  }
}

```

A t=0, currentWaterLevelLim= 400, nous sommes donc en train de

vérifier si nos plantes ont besoin d'eau ou pas. Si ce n'est pas le cas alors on passe dans le else de fin : On reste dans le même état, la pompe est éteinte, et nous souhaitons vérifier que nos plantes ne manquent pas d'eau. Si au contraire c'est le cas, si la valeur moyenne moy_value est $<=400$ dans ce cas nos plantes ont besoin d'eau.

```
CurrentWaterLevelLim=600
```

Désormais, nous vérifions grâce à l'instruction suivante :

```
if (moy_value<=currentWaterLevelLim)
```

que le seuil d'arrosage maximal n'est pas atteint. Si c'est le cas, si moy_value > 600 alors dans ce cas on repasse dans le else de fin de fonction et on éteint l'électrovanne. La valeur en dessous de laquelle nous ne voulons pas tomber est à nouveau 400 unités.

Mais dans le cas contraire, tant que nous n'avons pas atteint ce seuil : Alors si la pompe est éteinte et si la différence entre le temps auquel le programme est lancé, et la valeur de temps où nous avons éteint la

pompe (initialisée à 0) est supérieur à 30 secondes, alors on active le relais, le temps auquel nous avons allumé la pompe est =millis(). millis() vous renvoie un "unsigned long" représentant le nombre de millisecondes depuis le début de l'exécution du programme. Ainsi au bout de 30 secondes de non arrosage, si la valeur consigne d'humidité maximale n'est pas atteinte, nous allumons l'électrovanne, le temps auquel nous avons allumé la pompe = millis(). Par l'instruction :

```
if(millis() - pumpStartTime > waterTimeOut*1000 && pumpFlag==true)
```

Si le seuil maximal n'a pas été atteint, alors on éteint la pompe, ce qui laissera du temps à l'humidité de pénétrer dans la terre. Et voilà c'est aussi simple que cela, n'hésitez pas à télécharger le source complet sur : <http://www.humarobotics.com/downloads/ArrozDuino.rar>

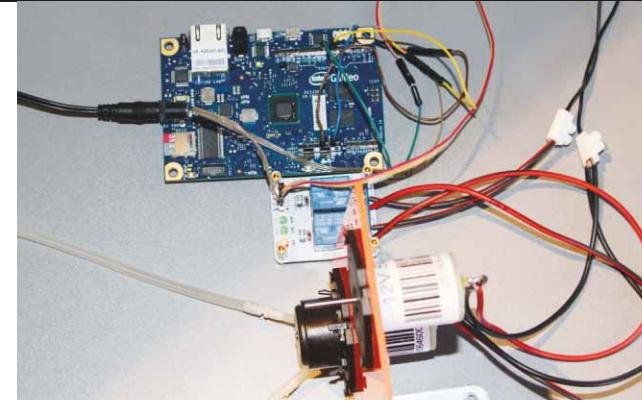
⌚ Joris Laborde

Ingénieur chez Génération Robots/HumaRobotics.

www.generationrobots.com

Galileo : créer son Barbot

Au sein du Hub Innovation de l'Epitech, nous aimons développer des projets sérieux en collaboration avec des partenaires industriels, mais également faire découvrir les nouvelles technologies aux étudiants par le biais de projets fun, dit «Sandbox». Le Barbot est un de ces projets, sorti de l'esprit d'un des membres du hub de manière aléatoire mais ayant conquis l'équipe après 4,2s de réflexion.



Liste du matériel

- Intel Galileo
- pompes péristaltiques 12V DC
- relais 12V
- alimentations 12V DC
- câbles divers

L'idée? Quand des projets sur Kickstarter et autres plateformes de crowdfunding lèvent des fortunes pour des bars automatisés, nous avons voulu démontrer qu'il était possible de faire pareil en DIY (Do It Yourself) et les moyens du bord (ou presque).

Nous allons au fil de cet article vous expliquer, couche par couche comment nous avons créé notre barbot, capable de nous préparer des cocktails «as demand», (le concept du CaaS) et de le rendre connectable de partout par le biais d'une API.

Description du matériel

► «Arduino made by Intel.»

Les cartes Arduino sont équipées de microcontrôleurs Atmel AVR. Il s'agit d'une architecture de type Harvard 8 bits (RISC[1]) modifiée. Cette carte est un peu spécifique vu qu'elle permet de faire fonctionner des programmes pour AVR ainsi que pour x86 tout en conservant la notion de shields pour le prototypage et les évolutions matérielles.

► «Il vaut mieux pomper même s'il ne se passe rien que risquer qu'il se passe quelque chose de pire en ne pompant pas.»

Pour pomper les liquides, nous utilisons des pompes péristaltiques. Ces pompes, habituellement utilisées en milieu pharmaceutique, ont l'avantage de ne pas être en contact direct avec les liquides. De ce fait, l'hygiène est mieux respectée et en cas de changement de liquide, il suffit de remplacer le tuyau et non la pompe entière. Il en va de même lors du nettoyage/rinçage. Ces pompes étant de simples moteurs DC, le contrôle est très aisément, contrairement à un moteur pas-à-pas (plus précis mais un peu plus délicat à piloter au premier abord).

Toutefois, elles présentent quelques inconvénients :

- usure du tube à l'intérieur du corps de la pompe, - effet de «pulsations», - coûts à l'achat (mais fabrication possible du corps de la pompe avec une imprimante 3D).

► Relais

Des relais 12V sont utilisés pour manipuler les pompes (On/Off).

► Alimentation 12V

Une alimentation 12V continue est requise. Pour nos tests, une alimentation 3 à 12V par pompe a été utilisée.

Bien penser à vérifier la tension totale des pompes et dimensionner l'alimentation en conséquence si elle est commune !

Schéma de câblage

La Figure 1 est un exemple de câblage du Barbot. Dans notre cas, 4 sorties digitales du GPIO de la Galileo sont utilisées pour contrôler nos quatre relais. Lorsque le signal HIGH est envoyé sur un GPIO, le relais est un interrupteur qui ferme le circuit et donc fournit 12Volts continus à la pompe que l'on souhaite contrôler. L'électronique ici, si l'on fait abstraction de la Galileo, est très simple, si ce n'est simpliste. Dans les prochaines versions, l'ajout d'un bouton d'amorçage, d'un voyant d'activité ainsi que d'un bouton d'arrêt avec retour à la Galileo sont prévus sur chacune des pompes. On peut ainsi s'assurer que les mesures de liquides seront plus justes, facilement observer l'activité de loin et stopper une pompe défectueuse ou dont la bouteille est vide ou absente.

Notes : Dans cet exemple, nous utiliserons 4 pompes, donc 4 sorties digitales. HIGH active le relai et alimente la pompe, et bien entendu LOW coupe l'alimentation.

Notes : [1] Reduced Instruction Set Computer ou microprocesseur à jeu d'instructions réduit.

I/O GALILEO CONTRÔLE DES POMPES

Notre objectif ici, va être d'utiliser notre Intel Galileo pour contrôler le déclenchement des pompes - et donc permettre de servir les boissons ! Pour cela, nous allons avoir besoin d'interroger les GPIO de la carte pour envoyer le signal. Deux étapes seulement nous suffiront à arriver à nos fins.

GPIO ?

Pour bien comprendre ce qui va suivre, il est nécessaire d'expliciter ce que sont les GPIO, et la manière de les exploiter. Un GPIO, General Purpose Input / Output, est tout simplement un port d'entrée / sortie sans utilisation prédefinie. Ces GPIO peuvent être associés à des connecteurs physiques (ou «pin»), disposés directement sur le circuit - ici, l'Intel Galileo. Dans notre cas, ces pins sont au nombre de 14, ce qui nous permettra largement de subvenir à nos besoins. Nous utiliserons ces GPIO uniquement en mode Output simple, c'est à dire que nous aurons uniquement besoin de faire sortir des valeurs binaires (0, LOW ou 1, HIGH) sur les pins. D'autres modes d'utilisation sont également disponibles pour une utilisation plus fine. Certains peuvent également être utilisés en mode PWM, Pulse Width Modulation, permettant de faire facilement varier la valeur associée. Nous ne nous en servirons pas de cette fonctionnalité pour ce que nous avons à faire, mais elle pourra nous être utile lors de futures améliorations; pour faire varier le débit de liquide, ou utiliser un débitmètre (en mode lecture) par exemple.

First step : accès via le système

Entrons maintenant dans le vif du sujet. Le système Linux embarqué nous permet, via son système de fichiers, d'accéder à toutes les fonctions nécessaires à la configuration et l'exploitation de nos fameux connecteurs. Ce dont nous avons besoin se trouve à l'emplacement `/sys/class/gpio`. Tout d'abord, nous allons avoir besoin d'ouvrir l'accès à notre connecteur avant de pouvoir en faire quoi que ce soit. Via une commande système, cela donne, pour exporter le connecteur numéro 7 (indiqué comme tel sur la Galileo) : `echo -n "27" > /sys/class/gpio/export`

Première surprise : l'identifiant du connecteur physique ne correspond pas à celui du GPIO correspondant ! Inutile de jouer aux devinettes, il va falloir se renseigner dans la documentation Intel pour trouver les correspondances exactes. Gardons cela en tête, et avançons. Ce que nous pouvons immédiatement remarquer est que la commande précédente a eu pour conséquence l'apparition de nouveaux nœuds dans le système de fichiers. Voici ceux qui nous intéressent :

```
/sys/class/gpio/gpio27
/sys/class/gpio/gpio27/direction
/sys/class/gpio/gpio27/drive
/sys/class/gpio/gpio27/value
```

La prochaine étape consiste à configurer le GPIO pour accéder en écriture au connecteur. Pour cela, il nous suffit de l'écrire sur le nœud du système de fichiers créé après l'exécution de la commande précédente : `echo -n "out" > /sys/class/gpio/gpio27/direction`

Une fois cela fait, passons au mode d'utilisation. Celui qui nous intéresse correspond à l'émission binaire d'information. Autrement dit, soit nous voulons envoyer un signal fort (1), soit aucun signal (0). Cela se traduit simplement par : `echo -n "strong" > /sys/class/gpio/gpio27/drive`

C'est tout pour la configuration, il ne nous reste plus qu'à l'exploiter, toujours de la même manière. Pour envoyer un signal : `echo -n "1" > /sys/class/gpio/gpio27/value` Et pour arrêter d'émettre : `echo -n "0" > /sys/class/gpio/gpio27/value`

Nous pouvons enfin fermer le canal vers ce pin de cette manière : `echo -n "27" > /sys/class/gpio/unexport`

Nous savons donc maintenant quoi faire exactement à notre code JavaScript pour permettre de reproduire cela avec notre service en Node.js.

Second step : Helper JavaScript

Nous allons travailler sur un objet GalileoGpio que nous pourrons utiliser facilement :

```
var GalileoGpio = function()
{
}
```

Commençons par simplifier un peu l'utilisation des GPIO en mappant le numéro de pin à l'identifiant de son GPIO :

```
this.pin =
{
  7: "27",
  8: "26"
};
```

Il nous suffit maintenant d'implémenter les méthodes correspondant aux commandes système précédemment décrites. Prenons comme exemple la commande permettant d'exporter, et généralisons-la :

```
GalileoGpio.prototype.openPin = function(pin_number)
{
  fs.writeFileSync("/sys/class/gpio/export", this.pin[pin_number]);
};
```

Ce code d'exemple peut être reproduit pour implémenter l'écriture sur `direction`, `drive` et `value` via des méthodes `setPinDirection`, `setPinPortDrive` et `writePin`. Exposons notre objet pour le rendre accessible dans le contexte Node.js :

```
module.exports = new GalileoGpio;
```

Et nous pouvons maintenant l'utiliser ! Prenons comme exemple la surcouche utilisée pour contrôler les pompes de notre BarBot.

Un peu de surcouche

Maintenant que nous avons fait le plus gros du travail, notre contrôleur de pompes est trivial à construire. On commence par instancier un GalileoGpio :

```
galil = require('./gpio-galileo-helper');
```

Puis nous pouvons commencer à définir notre contrôleur, en profitant de l'occasion pour permettre de travailler uniquement sur des pompes, et non plus sur des connecteurs :

```
var PumpController = function()
{
  this.pump =
```

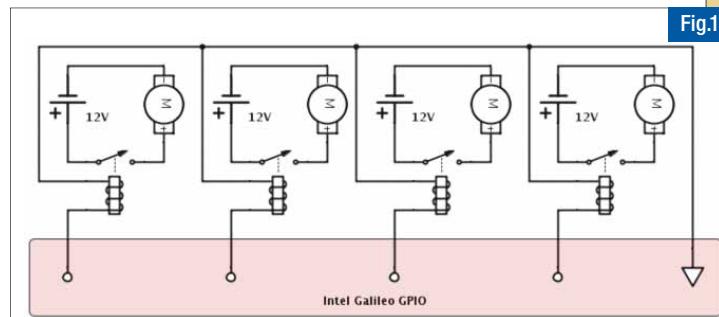


Fig.1

```

{
  1: 7,
  2: 8
};

};

Il ne nous reste plus qu'à implémenter la méthode de contrôle :

PumpController.prototype.drainIndex = function(pump_index,
delay, callback)
{
  var that = this;

  galil.openPin(this.pump[pump_index]);
  galil.setPinDirection(this.pump[pump_index], "out");
  galil.setPinPortDrive(this.pump[pump_index], "strong");
  galil.writePin(this.pump[pump_index], this.state['ON']);

  setTimeout(function()
  {
    galil.writePin(that.pump[pump_index], that.state['OFF']);
    galil.closePin(that.pump[pump_index]);
  }, delay);

  if (callback)
    callback();
};

}

```

Les commandes permettant d'exploiter les pompes devraient maintenant vous être familières, nous n'avons fait que les réutiliser. La méthode ci-dessus automatise leurs appels, et, via l'emploi d'un `setTimeout`, permet de déterminer le temps pendant lequel celle-ci va pomper via le paramètre `delay`.

L'appel d'une `callback` va nous permettre de chaîner nos commandes. L'exemple ci-dessus, lors d'un éventuel chaînage, déclenche les pompes simultanément. Nous pouvons définir son équivalent synchrone `drainIndexSync` simplement en déplaçant l'appel de cette `callback` dans le `setTimeout`. C'est terminé, nous pouvons maintenant utiliser notre `PumpController` dans notre web service !

Références

Documentation : <https://communities.intel.com/community/makers/documentation/galileodocuments>

Programming GPIO from Linux : <http://www.malinov.com/Home/sergey-s-blog/intelgalileo-programmingpifromlinux>

API

Maintenant que nous avons construit l'ensemble de notre barbot, que ce soit la partie mécanique mais également contrôleur, nous pouvons réfléchir à comment offrir la possibilité à l'utilisateur de passer une commande. Pour obtenir un système le plus ouvert possible et qui pourra être utilisé par l'ensemble le plus large possible d'appareil/interface, nous avons implémenté une couche dite d'API permettant de récupérer l'ensemble des recettes connues par le barbot puis de passer commande facilement. Ces deux actions permettent de créer tout type d'interface permettant d'offrir une utilisation facile de notre bar automatisé. Nous pouvons imaginer brancher un client mobile qui fera des appels vers le bot, une interface dite lourde ou embarquée sur tablette pour le serveur, par exemple, mais également, et pourquoi pas, passer commande de manière automatique dès que l'on sort de sa voiture connectée le soir en rentrant chez soi pour avoir un superbe cocktail tout prêt quand on va franchir la porte de la maison !

Explications

L'API offerte par le barbot est assez simple, elle se compose en deux parties : * Une action [GET] `/api/recipes` permettant de récupérer la liste des recettes * Un ensemble d'actions [POST] `/api/recipe/{name}` permettant de commencer le cocktail portant le nom indiqué par le champ `{name}`. Il est donc possible d'effectuer un premier appel, par exemple au lancement de l'application du barman pour récupérer la liste des recettes. On va recevoir un flux JSON du type :

```
{
  "name": "Black Russian", "endpoint": "/api/BlackRussian"},
  "name": "White Russian", "endpoint": "/api/WhiteRussian"
}
```

Chaque recette étant contenue dans un fichier `.json` dans le dossier `/recipes` du barbot avec le schéma suivant :

```
{
  "name": "White Russian",
  "api": "WhiteRussian",
  "ingredients": [
    {"index": 1, "time": 5000},
    {"index": 2, "time": 2000}
  ]
}
```

Une fois la liste reçue, libre à l'application de proposer un système permettant la sélection d'un cocktail puis sa demande, il suffit pour cela d'effectuer un appel, cette fois de type `POST` sur l'adresse de endpoint du cocktail, par exemple `'/api/WhiteRussian'`. Le barbot va réceptionner la commande et la traiter en actionnant les pompes nécessaires pour obtenir le cocktail commandé.

Technique

Maintenant que nous avons vu le principe de l'API exposée par notre Barbot, regardons comment celle-ci a été développée. Le choix de Node.js a été des plus logiques car d'une part c'est un Framework qui tourne de base sur les cartes Galileo, et d'autre part parce qu'il s'utilise très bien dans le cadre de la création d'API. Nous avons ajouté le package Express qui rend la création de site web et donc de service web plus facile avec Node.js, il nous a suffi de faire un script capable de charger une collection de recettes puis d'exposer les deux endpoints vus ci-dessus. Regardons maintenant le code, il est très court car l'API a pour seul rôle de faire proxy vers le contrôleur natif que nous avons détaillé dans la partie précédente.

```

var fs = require('fs');
var async = require('async');

// Load the native controller to manipulate pumps
var bar = require('../pump-controller');

var port = 4242;
var express = require('express');
var app = express();
var server = require('http').createServer(app, { log: false })
app.use(express.json());
app.use(express.urlencoded());
app.use(express.compress());
app.use(express.static(__dirname + '/public'));

// in-memory "database"
var recipes = [];

```

Le début de notre fichier est surtout là pour charger les différents packages utiles pour notre API dont le fameux 'pump-controller', puis on configure le serveur web par le biais du package express'. On va ensuite faire un morceau de code capable de lire l'ensemble des recettes disponibles dans le dossier /recipes. On va les sauvegarder en mémoire mais également créer un appel POST pour chaque cocktail permettant de donner vie à notre api.

```
// we read all recipes inside the ./recipes directory to load them
// in the in-memory "database" and prepare a POST query to be able to
// create cocktail "on demand"
fs.readdir('./recipes/', function (err, files) {
  files.forEach(function (f) {
    if (f.endsWith('.json')) {
      fs.readFile('./recipes/' + f, function (err, data) {
        if (err) {
          console.log('Error: ' + err);
          process.exit();
        } else {
          var r = JSON.parse(data);
          recipes.push(r);
          console.log('cocktail ' + r.name + ' added');
        // here we create a post query for each recipes
        app.post('/api/' + r.api, function (req, res) {
          prepareDrink(r.ingredients, function (err) {
            res.send('ok');
          });
        });
      });
    }
  });
});
});
```

Comme nous avons expliqué, il nous faut aussi une requête de type GET permettant de récupérer la liste des recettes, voici son code :

```
// when users call the GET /api/recipes url, we return the list
// of all recipes
app.get('/api/recipes', function (req, res) {
  var re = [];

  recipes.forEach(function (r) {
    re.push({name: r.name, endpoint: '/api/' + r.api});
  });
  res.send(re);
});
```

Une fois l'ensemble du code écrit, il ne reste plus qu'à utiliser express pour démarrer le serveur web pour rendre l'api disponible à tous les clients.

FRONT WEB

Dans le cas de notre prototype, une mini-interface web a été réalisée pour nous permettre de tester la création de nos premiers cocktails. Pour ce faire le choix s'est porté vers AngularJS hébergé sur un programme en node.js avec Express pour rester sur une stack très Javascript dans l'ensemble de notre projet. L'interface est capable d'utiliser l'API pour récupérer la liste des cocktails et permet de commander un cocktail par un simple clic sur le bouton associé.

Des clients mobiles, desktop,... sont prévus, on peut aller jusqu'à imaginer des scénarios pouvant faire rêver tous les geeks que nous sommes tels que la création d'un cocktail lors d'un déploiement réussi en production...

Et si cela ne motive pas les équipes !

ÉVOLUTIONS ET AMÉLIORATIONS POSSIBLES

Aujourd'hui, la mode est à la simplification des méthodes de paiement. Les méthodes suivantes peuvent avoir leur intérêt selon le milieu où elles sont utilisées

Paiement par mobile

Une application mobile de commande et statistiques sur la soirée ? Et pourquoi ne pas avoir le coût total de la soirée, le nombre de verres consommés ainsi que ses boissons préférées, ou encore des ingrédients indésirables. On peut aisément imaginer avoir un compte client sur le Barbot et à chaque demande d'un client mobile décompter la consommation du compte client associé sur le Barbot.

Paiement par RFID

Même principe que le paiement par mobile, sauf qu'ici on remplace le smartphone par un porte-clés RFID.

Paiement par monnaie virtuelle

Paiement via Bitcoin, Litecoin, ...

Contrôle par RFID

Des badges RDIF autocollants sont disposés sous des verres, permettant de savoir quelle contenance ils représentent et quel cocktail servir. Lorsqu'un utilisateur veut un cocktail, il pose le verre correspondant au cocktail sur un support. Un lecteur RFID relié au Barbot lit le badge. De ce fait, le barbot sait si un verre est présent, quel cocktail servir et sait, par exemple, s'il s'agit d'un verre de taille normale ou alors la version grand format. Hormis les solutions de paiement, d'autres possibilités d'évolutions se présentent.

Prévention

Un éthylomètre peut être interfacé assez facilement avec la Galileo. Avec un badge unique, de la reconnaissance faciale par exemple, l'historique de commande et l'éthylomètre, on peut déterminer si la consommation d'alcool devient excessive. De ce fait on peut ensuite imaginer avertir l'utilisateur, arrêter de le servir ou effectuer des actions prédéterminées (envoyer un SMS à un proche par exemple avec le Shield GSM pour Arduino), ...

Agrémentations diverses

Un distributeur automatique de glaçons ou d'ombrelles est envisageable. La problématique majeure demeure la conservation des glaçons dans un état solide. Suite à la fabrication du cocktail, des alertes visuelles et sonores sur le barbot peuvent être un plus.

Automatisations plus complexes

- Distribution de sucres
- Monitoring en temps réel concernant les niveaux des liquides
- Chargement automatisé des verres
- Suggestion de cocktails selon les goûts (amertume, doux, ...) et dosage du cocktail
- Création/sélection de cocktails selon des choix de couleurs de l'utilisateur
- Gestion des allergies ou interdiction de certains ingrédients
- Cocktails populaires et retours d'impressions des utilisateurs
- Servir l'Absinthe correctement :)

Adresse du projet : <https://github.com/Innovation-Hub/barbot>

 Pierre-Alexandre Louyot

 Thibaut Royer

 Niels Freier

Tous les trois sont Tech'Mentor au sein du Hub Innovation de l'école Epitech.

Intégrer la solution HERE Maps dans une application Nokia X

Unanimement reconnue pour la qualité de son service de cartographie, la solution HERE de Nokia est disponible sous Nokia X via l'API HERE Maps. Elle met à disposition du développeur Android une solution de cartographie clé en main remplaçant plus qu'efficacement la solution Maps de Google. Tour d'horizon de l'API HERE Maps au travers de la réalisation d'une application Nokia X utilisant la cartographie.

Basée sur feu Nokia Maps, HERE est une solution de cartographie reconnue à travers le Monde et utilisée quotidiennement par des millions de personnes que ce soient des grands groupes industriels comme BMW, Garmin, Toyota ou Yahoo mais également des développeurs travaillant sous Windows Phone. L'API HERE Maps mise à disposition sous Nokia X offre au développeur Android le même niveau de qualité et s'avère remplacer plus qu'avantageusement la solution Google Maps habituellement employée sous Android. D'un point de vue technique, HERE Maps se décompose en 3 grandes parties :

- ▶ Cartographie qui comprend notamment le rendu dynamique des cartes vectorielles pour plus de 190 pays avec différents types de cartes : carte normale comprenant les rues et quartiers, carte satellite ou encore carte affichant le trafic en temps réel.
- ▶ Recherche qui permet la recherche de villes, de rues ou de lieux avec ajout des points d'intérêt pour les zones recherchées. En outre, des services de géocodage et de reverse géocodage sont également proposés.
- ▶ Directions pour la gestion des directions entre 2 points que ce soit en voiture ou à pied. De nombreuses options sont à disposition pour ce service. Outre la qualité de ses cartes, HERE Maps présente 2 avantages majeurs pour le développeur et l'utilisateur. La solution utilise un rendu vectoriel pour les cartes normales ce qui permet une vitesse d'affichage accrue et supporte le mode offline pour un grand nombre de ses fonctionnalités. Cette dernière possibilité se révélant être un avantage non négligeable en regard de la solution Google Maps par exemple.

Environnement de travail

Dans la suite de cet article, nous allons prendre en main l'API HERE Maps en réalisant une application s'appuyant sur le service de cartographie. La première étape va donc consister à avoir un environnement de travail adéquat. Le pré-requis initial est de disposer de l'IDE Eclipse avec la dernière version du plugin ADT installé ainsi que le SDK Android. La plateforme Nokia X étant basée sur Android 4.1.2 avec une API en version 16, il est également nécessaire d'avoir installé ces outils depuis le SDK Manager d'Android. La seconde étape va consister à installer les services spécifiques Nokia X. L'installation se fait de nouveau via le SDK Manager au sein duquel on ajoute le site d'add-on suivant : <http://tools.nokia.com/nokia-x/repository/addon.xml>. Une fois le site ajouté, il faut installer les 4 packages listés ci-dessous :

- ▶ Android 4.1.2 (API 16) / Nokia X System Image
- ▶ Extras / Nokia X Device Definitions
- ▶ Extras / Nokia X Services
- ▶ Extras / Nokia X USB Driver

Pour les tests en phase de développement, il sera intéressant de disposer d'un émulateur du Nokia X à défaut de posséder un terminal physique. Les packages installés précédemment nous permettent de disposer désormais d'un périphérique de type Nokia X dans l'écran des terminaux virtuels du manager de périphérique Android (Fig.1).

Lors de la création du périphérique Nokia X virtuel, il conviendra bien entendu de veiller à sélectionner une API en version 16.



Fig.1

Emulateur du Nokia X

Création du projet

L'application utilisant l'API HERE Maps que nous allons développer va dans un premier temps proposer des fonctionnalités élémentaires à l'utilisateur. Ainsi, il pourra se déplacer sur la carte pour voir différentes zones géographiques, centrer la carte sur un point spécifique en s'appuyant sur les fonctionnalités de gestion des gestes de la plateforme Nokia X, zoomer ou dézoomer sur une zone en utilisant le pinch-to-zoom ou via l'écartement de 2 doigts ou encore faire tourner la carte. La création du projet se réalise de la même manière que pour une application Android habituelle sous Eclipse en utilisant le menu New puis «Android Application Project». Rien de particulier dans les étapes de création du projet si ce n'est qu'il faut sélectionner un SDK cible correspondant au moins à Ice Cream Sandwich 4.0.1 soit une API en version 14. La version minimum du SDK supportée par l'API HERE Maps est Android 2.3.3. Le projet est alors créé correctement dans votre workspace Eclipse.

Configuration

Le projet créé, nous devons maintenant ajouter la dépendance à la bibliothèque HERE Maps qui fait partie des services Nokia X que nous avons précédemment installée lors de la préparation de l'environnement de travail. Cet ajout se fait en allant dans l'onglet «Libraries» de la partie «Java Build Path» des propriétés du projet. A partir de cette fenêtre, il reste à cliquer sur «Add External JARs» et à sélectionner le JAR de l'API HERE Maps présent à l'emplacement suivant sur votre ordinateur : [Android SDK]\extras\nokia\nokia_x_services\libs\here\com.here.android.sdk.jar. Dans l'onglet «Order and Export», il faut bien préciser que la bibliothèque ne doit pas être exportée au moment de la création de l'APK de l'application.

Manifest

Au sein du projet, nous allons déclarer dans le manifest que la bibliothèque HERE Maps est utilisée et requise par l'application de la façon suivante :

```

<application ...>
    <uses-library android:name="com.here.android" android:
    required="true" />
    ...
</application>

```

L'API HERE Maps requiert un certain nombre de permissions que nous déclarons au sein du manifest afin que l'application puisse utiliser correctement l'API :

```

<uses-permission android:name="android.permission.ACCESS_FINE
    _LOCATION"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL
    _STORAGE"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK
    _STATE"/>
<uses-permission android:name="android.permission.CHANGE_NETWORK
    _STATE"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_WIFI
    _STATE"/>

```



Fig.2

Afin de fonctionner, le service HERE Maps nécessite de créer un APP ID et un APP TOKEN pour l'application sans quoi l'application ne pourra utiliser le service de cartographie correctement. La génération d'une clé de test se fait instantanément à l'adresse suivante :

<http://developer.nokia.com/nokia-x/nokia-apis/here-maps#keygen>.

La clé créée (Fig.2), nous pouvons éditer le manifest en renseignant l'APP ID et l'APP TOKEN au sein de la balise application comme suit :

```

<meta-data android:name="com.here.android.maps.appid" android:
    value="RntXgSL6TAZaZqm8v6Ac"/>
<meta-data android:name="com.here.android.maps.apptoken" android:
    value="a8nz9XJZ_mgMq44jD4idMQ"/>

```

Affichage de la carte

L'affichage de la carte se fait au sein d'une activité MapActivity et de son layout associé activity_main.xml au sein duquel on définit un fragment qui servira d'hôte pour la carte. Le contenu du layout est le suivant :

```

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <fragment
        class="com.here.android.mapping.MapFragment"
        android:id="@+id/mapfragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

</LinearLayout>

```

La carte HERE Maps est modélisée au sein du fragment MapFragment. Attention, pour des applications souhaitant supporter des versions d'Android inférieures à la 3.x, il faudra utiliser la classe MapCompatibilityFragment qui hérite de la classe de compatibilité Android Fragment. Ici, cela n'est pas notre souhait pour notre application et nous pouvons utiliser

MapFragment. A la création de l'activité, nous allons récupérer l'instance du fragment MapFragment et invoquer la méthode init avec en entrée un listener permettant de savoir quand l'initialisation du fragment sera terminée. Si celle-ci se termine correctement, on pourra alors récupérer l'objet Map associé au fragment et centrer la carte en un point de notre choix via ses coordonnées GPS puis définir le niveau de zoom. Ces actions sont implémentées dans le code suivant :

```

public class MapActivity extends Activity {
    // ...
    private Map map;
    private MapFragment mapFragment;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mapFragment = (MapFragment) getSupportFragmentManager().find
        FragmentById(R.id.mapfragment);
        mapFragment.init(new FragmentInitListener() {
            @Override
            public void onFragmentInitializationCompleted(InitError
            error) {
                if (error == InitError.NONE) {
                    map = mapFragment.getMap();
                    map.setCenter(MapFactory.createGeoCoordinate(43.17,
                    5.22, 0.0), MapAnimation.NONE);
                    map.setZoomLevel((map.getMaxZoomLevel() + map.get
                    MinZoomLevel()) / 2);
                } else {
                    Log.i(TAG, "Erreur à l'init de la carte");
                }
            }
        });
    }
}

```

L'exécution du projet de notre application permet de valider le fonctionnement (Fig.3) puisque l'on obtient bien une carte centrée sur la ville de coordonnées GPS (43.17, 5.22), à savoir Marseille.

Un peu d'interactivité

Fonctionnelle, notre application reste cependant assez rudimentaire. Nous allons donc l'étoffer en lui ajoutant 2 nouvelles fonctionnalités représentées sous la forme de contrôles au niveau IHM. La première sera un bouton permettant de revenir sur la position initiale sur la carte et la seconde permettra de switcher entre les différents schémas de cartes proposés par HERE Maps. Nous commençons par ajouter les 2 widgets à notre activity_main.xml juste au-dessus du fragment :

```

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_gravity="center_horizontal" >

    <Button
        android:id="@+id/homebutton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="goHome"

```

```

    android:text="@string/label_button_gohome" />
<Button
    android:id="@+id/schemebutton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="changeScheme"
    android:text="@string/label_button_changescheme" />
</LinearLayout>

```

Le bouton home déclare la méthode listener de click goHome au sein de laquelle nous allons remettre la carte dans son état initial, c'est-à-dire centrée sur le même point GPS avec le même niveau de zoom et en remettant à zéro toutes les modifications que l'utilisateur aurait pu faire entre temps. En outre, on souhaite garder le schéma de carte choisi par l'utilisateur et c'est pour ça que l'on applique le schéma précédemment stocké dans initialScheme. Le code de la méthode goHome est le suivant :

```

public void goHome(View view) {
    if (map != null) {
        map.setCenter(MapFactory.createGeoCoordinate(43.17, 5.22,
0.0), MapAnimation.NONE);
        map.setZoomLevel((map.getMaxZoomLevel() + map.getMinZoom
Level()) / 2);
        map.setOrientation(0);
        map.setTilt(0);
        map.setMapScheme(initialScheme);
    }
}

```

Au niveau de la méthode de changement de schéma de carte, on va simplement récupérer les différents schémas à disposition pour la carte via la méthode getMapSchemes de l'objet Map puis afficher les différents schémas les uns à la suite des autres lors du click sur le bouton de changement de schéma. Ceci nous donne le code suivant :

```

public void changeScheme(View view) {
    if (map != null) {

```



Affichage de la carte

```

        String current = map.getMapScheme();

        if (initialScheme == null || initialScheme.isEmpty()) {
            initialScheme = current;
        }

        List<String> schemes = map.getMapSchemes();
        int total = map.getMapSchemes().size();

        if (schemes.get(total - 1).equals(current))
            map.setMapScheme(schemes.get(0));
        else {
            for (int count = 0; count < total - 1; count++) {
                if (schemes.get(count).equals(current))
                    map.setMapScheme(schemes.get(count + 1));
            }
        }
    }
}

```

L'exécution de l'application sur notre émulateur Nokia X nous permet de valider le fonctionnement et d'afficher un autre schéma de carte sur la même position (Fig.4).

Calcul de directions

Fonctionnalité incontournable de toute solution de cartographie, le calcul de directions est bien entendu disponible avec HERE Maps. Nous allons donc ajouter le calcul de directions entre 2 points à savoir Marseille et Nice. Pour ce faire, on modifie quelque peu notre layout en ajoutant le bouton de directions ainsi qu'un TextView qui servira à afficher le nombre de manœuvres à effectuer pour la route sélectionnée :

```

<TextView
    android:id="@+id/title"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/textview_routecoordinates_2waypoints" />

```

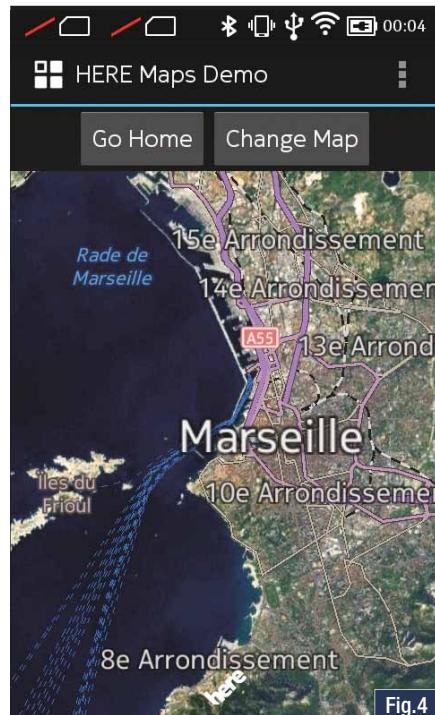
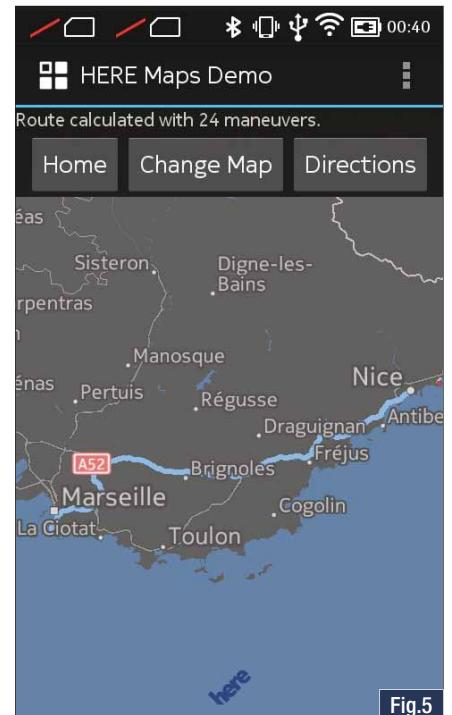


Schéma de carte différent



Calcul de directions

```

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:orientation="horizontal" >

    // home button et scheme button ...

<Button
    android:id="@+id/directionsbutton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="getDirections"
    android:text="@string/label_button_getdirections" />
</LinearLayout>

```

Une fois le layout modifié, nous allons créer une implémentation de RouteManagerEventListener qui est l'interface listener pour les évènements lors du calcul de directions :

```

private RouteManagerEventListener routeManagerEventListener =
    new RouteManagerEventListener() {
    public void onCalculateRouteFinished(RouterError errorCode,
    List<RoutingResult> result) {
        if (errorCode == RouterError.NONE && result.get(0).get
        Route() != null) {
            mapRoute = MapFactory.createMapRoute(result.get(0).get
        Route());
            map.addMapObject(mapRoute);
            GeoBoundingBox gbb = result.get(0).getRoute().get
        BoundingBox();
            map.zoomTo(gbb, MapAnimation.NONE, Map.MOVE_PRESERVE_
        ORIENTATION);
            textViewResult.setText(String.format("Route calculated
        with %d maneuvers.", result.get(0).getRoute().getManeuvers()
        .size()));
        } else {
            textViewResult.setText(String.format("Route calculation
        failed: %s", errorCode.toString()));
        }
    }

    public void onProgress(int percentage) {
        textViewResult.setText(String.format("... %d percent done
        ...", percentage));
    }
};

```

Si le calcul de routes se termine correctement, alors on crée une route en récupérant la première route disponible. Pour ce faire, on utilise la méthode statique createMapRoute de la factory MapFactory. L'objet MapRoute obtenu est ensuite ajouté à la Map pour visualisation par l'utilisateur.

Il nous reste maintenant à implémenter le code de la méthode getDirections chargée de lancer le calcul de directions. La première étape consiste à retirer les résultats du calcul précédent. On initialise ensuite un RouteManager obtenu via la méthode statique getRouteManager de MapFactory. L'étape suivante permet de définir les paramètres du calcul de route via l'objet RoutingMode. On définit les points de départ et d'arrivée de la route et on les ajoute à l'objet liste dédié WaypointParameterList. Enfin, il ne reste plus qu'à lancer le calcul de routes de manière asynchrone via

l'appel à la méthode calculateRouteAsync de l'instance de RouteManager créée en passant en entrée les différents éléments paramétrés ainsi que le RouteManagerEventLister détaillé précédemment. Ceci se traduit par le code suivant :

```

public void getDirections(View view) {
    textViewResult.setText("");

    if (map != null && mapRoute != null) {
        map.removeMapObject(mapRoute);
        mapRoute = null;
    }

    RouteManager routeManager = MapFactory.getRouteManager();
    RoutingMode routingMode = MapFactory.createRoutingMode();
    routingMode.setRoutingOptions(EnumSet.noneOf(RoutingOptions.
    class));
    routingMode.setTransportMode(TransportMode.CAR);
    routingMode.setRoutingType(RoutingType.FASTEST);

    WaypointParameterList waypointParameterList = MapFactory.
    createWaypointParameterList();
    waypointParameterList.addCoordinate(MapFactory.createGeo
    Coordinate(43.17, 5.22));
    waypointParameterList.addCoordinate(MapFactory.createGeo
    Coordinate(43.69, 7.26));

    routeManager.calculateRouteAsync(waypointParameterList,
    routingMode, routeManagerEventListener);
}

```

Le résultat du calcul de directions est présenté à la Fig.5.

Portage depuis Google Maps

Jusqu'à présent, cet article aura fait le tour de la mise en oeuvre de HERE Maps pour une nouvelle application Android pour Nokia X. Néanmoins, dans de nombreux cas d'utilisation, il sera nécessaire pour les développeurs Android de migrer une application existante utilisant Google Maps vers les services de HERE Maps. Cette migration est facilitée par le travail des équipes de Nokia qui ont créé une API wrapper dans HERE Maps proposant un certain nombre d'interfaces et de méthodes calquées sur celles de l'API Google Maps. Dès lors, il suffit quasiment de remplacer les imports des objets et l'on obtient un code fonctionnel en l'état. Certains cas très poussés pourront présenter plus de difficultés de migration mais la majorité des applications existantes verront cette migration vers HERE Maps se réaliser en douceur.

Conclusion

Reposant sur l'excellent service de cartographie HERE de Nokia, l'API HERE Maps offre aux développeurs Android toutes les fonctionnalités de cartographie pour réaliser des applications puissantes sous Nokia X. Le développeur découvrira avec plaisir un certain nombre d'avantages par rapport à la solution Google Maps habituellement utilisée sous Android tel que le mode hors-ligne. En sus, l'API mise à disposition est simple d'emploi et permet d'obtenir un code concis. Enfin, le portage des applications existantes utilisant Google Maps n'a pas été oublié puisque l'API propose un certain nombre d'interfaces et de méthodes dédiées se calquant sur ce qui se fait dans l'API Google Maps. Au final, on obtient une solution performante s'intégrant parfaitement dans la plateforme Nokia X.

● Sylvain Saurel - Ingénieur d'Etudes Java / Java EE
sylvain.saurel@gmail.com

La programmation fonctionnelle en F# #1

Quand on parle de programmation fonctionnelle entre développeurs, il n'est pas rare d'entendre des commentaires du type : « j'en ai fait un peu à l'école » ou « c'est un langage pour faire du calcul scientifique, n'est-ce pas ? ». C'est malheureusement cette perception parfois un peu limitée sur la programmation fonctionnelle qui empêche ce paradigme de se positionner au même niveau que d'autres paradigmes de programmation plus courants tels que la programmation impérative ou orientée objet.

Malgré cela, la programmation fonctionnelle commence à devenir de plus en plus populaire. Nous verrons que les langages fonctionnels nous permettent d'écrire des programmes plus concis et élégants. De par leur nature immuable⁽¹⁾, ces langages facilitent le développement de certaines tâches comme la programmation parallèle.

Nous ferons un tour des principaux concepts de la programmation fonctionnelle en prenant comme base d'exemple le langage F#. Nous verrons que les langages fonctionnels peuvent être considérés comme des vrais choix de langage. Nous verrons comment certains concepts des langages fonctionnels ont joué un rôle sur l'évolution d'autres langages de programmation tels que C# ou Java.

La programmation fonctionnelle

La programmation fonctionnelle est un paradigme de programmation, qui comme son nom l'indique, est centrée sur la notion de fonctions, fonctions qui ne dépendent que des données en entrée et non de l'état global du programme. Cette notion d'immuabilité prévient ainsi les effets de bord, c'est-à-dire, qu'une fonction ne peut pas intervenir au-delà de sa portée (scope).

La programmation fonctionnelle s'oppose souvent à la programmation impérative (comme C#, C++ ou Java) où le changement d'état des données est mis en avant. Mais nous verrons que cette distinction n'est pas vraie dans l'absolu et que la plupart des langages fonctionnels permettent la présence de données mutables.

Les langages fonctionnels

Pour la petite histoire, les langages fonctionnels s'inspirent des notions du lambda-calcul (ou *-calcul*) qui pose les bases sur les notions de fonction et leur application. Cela reste une abstraction mais définit les principes de la programmation fonctionnelle.

Le premier langage fonctionnel apparaît vers la fin des années 1950 et se nomme *Lisp*. Depuis, plusieurs variantes ont vu le jour, et, aujourd'hui nous pouvons trouver une quantité considérable de langages fonctionnels. Les langages fonctionnels peuvent se diviser en deux grands groupes : purs et impurs.

On parle de pureté d'un langage fonctionnel en faisant référence aux fonctions pures. Une fonction pure est une fonction qui satisfait deux conditions : elle retourne toujours le même résultat pour des arguments de même valeur, elle ne dépend d'aucun état ou information externe ; l'évaluation de la fonction est dénouée de tout effet de bord et elle ne provoque aucun changement externe, par exemple l'affectation de la valeur d'une variable externe.

Dans la catégorie des langages purs, les représentants les plus connus sont Haskell (1990), Miranda (1985 et qui n'est plus en développement) et Clean (1987).

En pratique, la plupart des langages fonctionnels sont impurs et nous avons une grande variété de représentants.

(1) On dit qu'une donnée est mutable quand il est possible de changer sa valeur. Dans le cas contraire, on parle de donnée immuable.

F# sur lequel nous baserons nos exemples est apparu en 2002 et est développé par Microsoft Research. Il repose sur la plate-forme .NET. F# est plus qu'un langage fonctionnel, c'est un langage multi paradigme. Étant donné que la plate-forme .NET est de nature objet, F# doit pouvoir s'adapter à ce paradigme. Cela n'empêche que F# supporte également l'ensemble des notions de la programmation fonctionnelle.

Nous avons également : Caml (1995) et OCaml (1996) créés et développés par l'INRIA (Institut national de recherche en informatique et en automatique) en France ; Scala (2003) prévu pour être exécutable dans la JVM principalement ; Erlang (1987), développé par Ericsson, est très connu par sa tolérance aux pannes et sa capacité de mise à jour « à chaud », il est utilisé chez WhatsApp pour la gestion des messages.

Les concepts clés de la programmation fonctionnelle

La Syntaxe

La syntaxe des langages fonctionnels est très succincte, absente de bruit mais sans manque d'information. Un terme récurrent dans la littérature parlant de programmation fonctionnelle est l'élégance. On dit en effet que les langages fonctionnels sont élégants. Les exemples de code seront complétés par l'information concernant le type des valeurs en commentaires. On déclare des valeurs avec le mot clé *let* :

```
let reponse = 42
//val reponse : int = 42
```

Les fonctions sont déclarées de la même façon :

```
let ditSalut nom = printf "Salut %s !" nom
//val ditSalut : nom:string -> unit

let ageChienEnHuman ageChien = ageChien * 7
//val ageChienEnHuman : ageChien:int -> int
```

Dans une fonction, il n'est pas nécessaire de renvoyer explicitement une valeur de retour (comme dans d'autres langages avec le mot clé *return*), la valeur de retour sera la dernière expression d'un bloc.

Tout est expression

En programmation fonctionnelle tout (ou presque tout) est une expression, cela veut dire que toute évaluation renvoie une valeur et qui peut donc être affectée à un identifiant (ou variable). On parle d'ailleurs de rattachement de valeur (value binding) et non d'affectation de valeur à une variable. Même le type considéré comme *void* en programmation impérative est une valeur (appelé *unit* en F#). Quel intérêt ? Cela nous permet d'écrire des expressions de la sorte :

```
let unChiffre = 9
//val unChiffre : int = 9
```

```

let bonneReponse = if unChiffre = 42 then true else false
//val bonneReponse : bool = false

let maListe = [for i in 1..10 -> i*i]
//val maListe : int list = [1; 4; 9; 16; 25; 36; 49;
64; 81; 100]

```

Tout comme en JavaScript, les fonctions sont traitées comme n'importe quelle valeur. On déclare donc une fonction comme n'importe quelle autre valeur :

```

let typeDeMechant nom =
  if nom = "Ernst Stavro Blofeld" then
    "Super méchant"
  elif nom = "Lex Luthor" then
    "Plutôt méchant"
  else
    "On ne sait pas"
//val typeDeMechant : nom:string -> string

```

Nous notons également le type de chaque valeur (en commentaire dans le code). Dans les cas des fonctions, en F# la notation est la suivante : type $p_1 \rightarrow$ type $p_2 \rightarrow \dots$ type $p_n \rightarrow$ type retour. Par exemple, en F# la signature de la fonction qui nous permet de filtrer une liste – correspondant à la méthode *Where* en LINQ pour C# – est la suivante :

```
List.filter : ('T -> bool) -> 'T list -> 'T list
```

Cette fonction reçoit deux paramètres : ($T \rightarrow \text{bool}$) qui est donc à son tour une fonction qui prend un paramètre de type ' T et qui retourne un booléen; et une liste d'éléments de type ' T . *List.filter* renvoie une nouvelle liste de type ' T comme résultat. La notation ressemble beaucoup à la notation mathématique des fonctions. Cela peut paraître déroutant au début mais ce n'est pas plus compliqué que d'assimiler les signatures des méthodes en C# :

```
public static IEnumerable<TSource> Where<TSource>(this
  IEnumerable<TSource> source, Func<TSource, bool> predicate);
```

L'immuabilité (éviter les effets de bord)

L'immuabilité est un concept clé en programmation fonctionnelle qui change la manière dont les programmes sont écrits. Par défaut, toute valeur est immuable en F#, cela veut dire qu'une fois la valeur définie elle ne peut pas changer.

Cette notion n'est pas restreinte aux langages fonctionnels, en C# on trouve également des structures de données immuables : le type *string* ne peut pas changer de valeur ; les méthodes d'extension LINQ (*Where*, *Select*, *GroupBy*...) renvoient des séquences immuables ; le modificateur *readonly* permet de déclarer un champ comme immuable.

Il est possible de créer des valeurs mutables en F# en utilisant de manière explicite le mot clé *mutable*. Cela devient nécessaire quand on veut interagir avec (ou exposer) du code mutable.

```

// affectation de valeur
let form = new System.Windows.Forms.Form()
form.BackColor <- System.Drawing.Color.Azure
form.Text <- "Bonjour le monde !"

```

```

// somme en utilisant une variable mutable
let sommeJusqua limite =
  let mutable somme = 0
  for i in [1..limite] do somme<-somme+i
  somme

```

Quel intérêt ? Des données immuables facilitent la programmation concurrente en prévenant les effets de bord.

Le parallélisme

L'absence d'effets de bords dans l'exécution d'un programme nous permet de pouvoir composer des fonctions de manière sûre. Cela facilite ainsi la possibilité de paralléliser des traitements informatiques. Quand on est sûr qu'une fonction est indépendante et n'a pas la possibilité de changer l'état global de notre programme il devient assez simple et surtout sans risque de la paralléliser.

La récursivité

Les algorithmes récursifs sont assez courants quand on traite avec certains types de structures de données. En F# une fonction récursive doit être explicitement déclarée avec le mot clé *rec*. En programmation fonctionnelle les appels récursifs sont plus courants qu'en programmation impérative. Ainsi, la manipulation de données immuables nous incite à penser autrement nos algorithmes.

```

// somme sans valeurs mutables
let sommeJusquaRec limite =
  let rec sommeInterne somme valeur = //fonction accessible
  uniquement dans le scope sommeJusquaRec
    if valeur = 0 then somme
    else sommeInterne (somme + valeur) (valeur - 1)
  sommeInterne 0 limite

```

Rien de nouveau par rapport à ce que l'on peut faire en C#. Mais nous savons bien que nous ne pouvons pas enchaîner des méthodes récursives indéfiniment puisque nous risquons de dépasser la capacité maximale de la pile d'exécution et ainsi déclencher une *StackOverflowException*. Même si notre méthode a une récursivité terminale (2) (tail-recursive), le compilateur C# ne fait pas d'optimisation sur ce type de méthode. Et ce malgré le support au niveau de la CLR d'une instruction d'optimisation des appels terminaux (tail call optimization). Le compilateur F# fait de l'optimisation de récursivité terminale.

La testabilité

Des fonctions sans effet de bord sont facilement testables. C'est d'ailleurs une pratique courante dans le développement fonctionnel de tester au fur et à mesure les fonctions que l'on définit. Les fonctions, mises bout à bout par des techniques de composition ou pipelining – notions que nous explorerons plus loin – forment un ensemble cohérent. On entend parfois dire à propos des programmes fonctionnels : « mon programme compile donc il marche ». La suite de F# dans *Programmez !* n°177.

● Pablo Fernandez Duran, *SoftFluent*

<http://softfluent.com>
<http://blogs.softfluent.com/>

(2) Un appel dit de type terminal peut être optimisé afin d'éliminer d'une certaine manière la récursivité.

XAMARIN, l'extension multiplateforme de .Net

2^{ème} partie

XAMARIN est une technologie qui a le vent en poupe ! Sur un marché ultra dynamique tel que celui du mobile, disposer d'une solution multiplateforme performante est devenu incontournable. Ce tour d'horizon synthétique à 360 degrés, avec un focus particulier sur l'implémentation des UI représentant le premier challenge pour le développeur, vous permettra d'en mesurer tout le potentiel. Passons à la pratique !

Pour illustrer l'exemple et afin que vous puissiez récupérer le code, je vous recommande l'excellent article de SLODGE sur CodeProject. Nous allons

Créer le CORE

- ▷ Création d'un service et de son interface
- ▷ Création du VIEW MODEL
- ▷ Création du MODEL

Créer une UI Android

- ▷ VIEW Android (fichier AXML)
- ▷ Binder la VIEW au VIEW MODEL du CORE

Créer une UI iOS

- ▷ VIEW iOS (fichier XIB)
- ▷ Binder la VIEW au VIEW MODEL du CORE

Dans votre librairie Core : Création d'un service

Création d'une interface de service :

```
public interface ICalculation
{
    double TipAmount(double subTotal, int generosity);
}
```

Définir l'implémentation :

```
public class Calculation : ICalculation
{
    public double TipAmount(double subTotal, int generosity)
    {
        return subTotal * ((double)generosity)/100.0;
    }
}
```

Création du ViewModel :

Les viewmodel doivent hériter de MvxViewModel (classe générique de base de MVVMCross).

```
using Cirrious.MvvmCross.ViewModels;

namespace TipCalc.Core
{
    public class TipViewModel : MvxViewModel
    {
        private readonly ICalculation _calculation;
        public TipViewModel(ICalculation calculation)
        {
            _calculation = calculation;
        }

        public override void Start()
        {
```

```
        _subTotal = 100;
        _generosity = 10;
        Recalcuate();
        base.Start();
    }

    private double _subTotal;

    public double SubTotal
    {
        get { return _subTotal; }
        set { _subTotal = value; RaisePropertyChanged(() => SubTotal); Recalcuate(); }
    }

    private int _generosity;

    public int Generosity
    {
        get { return _generosity; }
        set { _generosity = value; RaisePropertyChanged(() => Generosity); Recalcuate(); }
    }

    private double _tip;

    public double Tip
    {
        get { return _tip; }
        set { _tip = value; RaisePropertyChanged(() => Tip); }
    }

    private void Recalcuate()
    {
        Tip = _calculation.TipAmount(SubTotal, Generosity);
    }
}
```

Création de notre objet Application qui sera responsable du chargement des viewmodel et services :

L'objet App doit hériter de MvxApplication.

```
using Cirrious.CrossCore.IoC;
using Cirrious.MvvmCross.ViewModels;

namespace TipCalc.Core
{
```

```

public class App : MvxApplication
{
    public App()
    {
        Mvx.RegisterType<ICalculation, Calculation>();
        Mvx.RegisterSingleton<IMvxAppStart>(new MvxAppStart<Tip
ViewModel>());
    }
}

```

A ce stade du développement nous disposons des fondations de notre application qui contient nos modèles de vues, nos services et nos modèles. Nous allons maintenant étendre la vue aux plateformes mobiles cibles type Android et iOS.

Création d'un projet client pour Android

Maintenant on va procéder à la création de notre projet Android qui va utiliser le projet Core et définir les vues de notre application

Création du Setup qui va démarrer notre application Android :

```

using Android.App;          // API XAMARIN
using Android.Content;       // API XAMARIN
using Cirrious.MvvmCross.Droid.Platform; // Framework MvvMCross
using Cirrious.MvvmCross.Droid.Views;    // Framework MvvMCross
using Cirrious.MvvmCross.ViewModels; // Framework MvvMCross
using TipCalc.Core;          // Votre implémentation

namespace TipCalc.UI.Droid
{
    public class Setup : MvxAndroidSetup
    {
        public Setup(Context applicationContext) : base(applicationContext)
        {
        }

        protected override IMvxApplication CreateApp()
        {
            return new App();
        }

        protected override IMvxNavigationSerializer CreateNavigation
Serializer()
        {
            Cirrious.MvvmCross.Plugins.Json.PluginLoader.Instance.
EnsureLoaded(true);
            return new MvxJsonNavigationSerializer();
        }
    }
}

```

Ajout des vues :

Les vues sont définies avec un langage XML (AXML comme vu dans le tableau des UI Mobiles).

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/
res/android"
    xmlns:local="http://schemas.android.com/apk/res/TipCalc.UI.Droid"
    android:orientation="vertical"

```

```

        android:layout_width="fill_parent"
        android:layout_height="fill_parent">
        <TextView
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:text="SubTotal" />
        <EditText
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            local:MvxBind="Text SubTotal" />
        <TextView
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:text="Generosity" />
        <SeekBar
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:max="40"
            local:MvxBind="Progress Generosity" />
        <View
            android:layout_width="fill_parent"
            android:layout_height="1dp"
            android:background="#ffff00" />
        <TextView
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:text="Tip to leave" />
        <TextView
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            local:MvxBind="Text Tip" />
    </LinearLayout>

```

Dans vos VIEWS il faut maintenant définir le binding. Dans notre exemple il est défini dans le code XML comme suit :

```
local:MvxBind="Text Tip"
```

La propriété Text du TextView est bindée à la propriété Tip du modèle de vue TipViewModel. On va maintenant modifier notre classe de vue pour définir la liaison entre la vue et le modèle de vue. Voici l'implémentation :

```

using Android.App;
using Cirrious.MvvmCross.Droid.Views;
using TipCalc.Core;

namespace TipCalc.UI.Droid.Views
{
    [Activity(Label = "Tip", MainLauncher = true)]
    public class TipView : MvxActivity
    {
        public new TipViewModel ViewModel
        {
            get { return (TipViewModel) base.ViewModel; }
            set { base.ViewModel = value; }
        }

        protected override void OnViewModelSet()
        {
            base.OnViewModelSet();
            SetContentView(Resource.Layout.View_Tip);
        }
    }
}

```

}

Voilà nous disposons maintenant d'une VUE Android rattachée à notre implémentation.

Création d'un projet client pour IOS

Passons maintenant à iOS. Nous allons procéder de la même façon pour créer l'application iOS, on aura des différences dans la création des vues. Comme pour Android nous allons passer à la création du Setup qui va démarrer notre application.

```
using System;
using Cirrious.MvvmCross.Touch.Platform;
using TipCalc.Core;
using Cirrious.MvvmCross.Touch.Views.Presenters;

namespace TipCalc.UI.Touch
{
    public class Setup : MvxTouchSetup
    {
        public Setup (MvxApplicationDelegate appDelegate, IMvxTouchViewPresenter presenter)
            : base(appDelegate, presenter)
        {
        }

        protected override Cirrious.MvvmCross.ViewModels. IMvxApplication CreateApp ()
        {
            return new App();
        }
    }
}
```

Ajout des vues :

Les vues sont définies en utilisant XCode (bientôt ce travail sera fait directement depuis Visual Studio) ou via le designer de Xamarin Studio avec un glisser/déplacer des contrôles. Après il faut modifier la classe de vue pour créer la liaison entre la vue et le modèle de vue

```
using System;
using System.Drawing;
using MonoTouch.Foundation; // API Xamarin iOS.
using MonoTouch.UIKit; // API Xamarin iOS.
using Cirrious.MvvmCross.Touch.Views;
using Cirrious.MvvmCross.Binding.BindingContext;
using TipCalc.Core;

namespace TipCalc.UI.Touch
{
    public partial class TipView : MvxViewController
    {
        public new TipViewModel ViewModel
        {
            get { return (TipViewModel)base.ViewModel; }
            set { base.ViewModel = value; }
        }
    }
}
```

```
public TipView () : base ("TipView", null)
{
}

public override void ViewDidLoad ()
{
    base.ViewDidLoad ();

    this.CreateBinding (this.TipLabel).To( (TipViewModel vm) =>
m.Tip ).Apply();

    this.CreateBinding (this.SubTotalTextField).To( (TipView
del vm) => vm.SubTotal ).Apply();

    this.CreateBinding (this.GenerositySlider.To( (TipView
del vm) => vm.Generosity ).Apply();
}

}
```

Comme dans le précédent exemple nous allons maintenant définir pour notre vue le binding sous IOS. Ce dernier se fait dans la classe de la vue comme suit :

```
this.CreateBinding (this.TipLabel).To( (TipViewModel vm) =>  
    vm.Tip ).Apply();
```

On crée un binding entre le label TipLabel défini dans le designer avec la propriété Tip définie dans le modèle de vue TipViewModel.

A ce stade vous disposez d'un projet CORE, d'un projet UI pour Android et d'un projet UI pour iOS. Vous avez maintenant un aperçu synthétique de ce qui vous attend dans la démarche d'implémentation de vos applications mobiles. Toutefois les surprises ne manqueront pas :-)

Conclusion

Dans cet article, nous avons souhaité présenter un tour d'horizon XAMARIN pour comprendre les principaux concepts du développement mobile. Nous pensons que le premier défi du développeur, dans le cadre du développement mobile, se situe principalement sur le développement UI. La compréhension des patterns UI est vraiment déterminante tant les différences entre les plateformes sont significatives. Par conséquent, en plus de la compréhension des patterns UI, l'usage d'un Framework tel que MVVMCross pour fédérer les UI de vos développements mobiles nous paraissait incontournable.

Au regard des nombreuses références projet mobile, pour le marché du B2B / B2C et même du jeu vidéo, XAMARIN a su convaincre. A travers une expérience de plus de 4 ans dans le domaine du développement mobile en utilisant de nombreux Framework du marché, nous avons pu en effet constater la fiabilité de cette technologie. XAMARIN propose incontestablement une réponse adaptée aux enjeux du développement mobile. Intimement liée aux plateformes mobiles cibles, XAMARIN ne pose donc aucune limite sauf celle liée au subset DotNet qui ne peut être intégralement transposé sur mobile. En utilisant cette technologie, chaque développeur bénéficie d'un écosystème puissant autour de Visual Studio / Team Foundation ainsi que des atouts de la filière de développement Dot-Net qui nous propose une ingénierie des plus performantes.

Pour terminer, nous organisons régulièrement des Meetup autour de XAMARIN. Cette communauté est destinée aux chefs de projets, architectes et développeurs mobiles, créateurs de startup... Venez nous rejoindre au <http://www.meetup.com/Paris-XAMARIN-Cross-platform-Mobile-Meetup/>

Google Maps sous Android : garantir de bonnes performances

La présence d'une carte est souvent nécessaire dans les applications Android, notamment dans des applications liées au transport, ou encore à certains réseaux sociaux indiquant où se trouvent des contacts. L'utilisateur est alors amené à parcourir la carte et à jouer avec le zoom pour situer des repères qui l'intéressent : un ami, un restaurant, un itinéraire... Dès que de nombreux repères sont à placer sur la carte, celle-ci risque d'être ralentie. Mais si la carte n'est pas fluide et se dessine par saccades, l'utilisateur peut vite être découragé et chercher une autre application. Quelles règles élémentaires respecter alors afin de garantir une bonne réactivité de la carte Google ?

Depuis la deuxième version de l'API Google Maps, fonctionnant avec les services Google Play, il est beaucoup plus simple de gérer les repères sur la carte grâce aux événements disponibles sur la classe centrale `com.google.android.gms.maps.GoogleMap`.

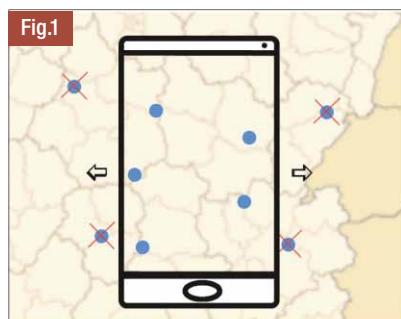


Fig.1

Lorsque de nombreux repères sont à placer sur la carte, par exemple des milliers de repères sur toute la France, une première règle fondamentale consiste à ne placer que ce que l'utilisateur est amené à voir. Placer d'emblée tous les repères sur la carte alourdira celle-ci qui se redessinera lentement et par saccades. Il s'agit donc de placer les marqueurs en mode *lazy* en fonction des déplacements et des modifications de zoom de l'utilisateur Fig.1.

Déterminer les bornes géographiques affichées à l'écran

Afin de permettre déplacements et zooms, l'API Google utilise une caméra qui se focalise sur telle ou telle partie du monde. En sachant précisément sur quelle zone de la carte la caméra est focalisée, le développeur peut déterminer si le marqueur doit être ajouté à la carte, ou, au contraire, si celui-ci n'est pas visible dans l'angle de la caméra et n'a donc pas besoin d'être ajouté. Pour cela, il dispose de la très utile classe `Projection`, disponible au sein de l'API Google Maps V2 et qui agit comme une passerelle entre des coordonnées x et y à l'écran et une latitude et une longitude sur terre. Sinon, sans cette classe, le développeur aurait à écrire de nombreux calculs permettant des conversions d'un système de coordonnées x et y (pixels) vers un système de projection de la surface terrestre. Tandis que via l'API, pour connaître les coordonnées géographiques visibles à l'écran, il suffit d'écrire :

```
GoogleMap map = getMap(); //obtient la carte via l'activité courante
LatLngBounds bounds = _map.getProjection().getVisibleRegion()
.latLngBounds;
```

Puis il peut tester si un point géographique est contenu dans la zone visible à l'écran par un simple appel :

```
boolean isVisible = bounds.contains(friend.getPosition());
if(isVisible){
    MarkerOptions markerOptions = new MarkerOptions();
```

```
Marker addedMarker = _map.addMarker(markerOptions);
}
```

Retirer les points qui ne sont plus visibles

Bien entendu, il faut également retirer les points auparavant ajoutés qui deviennent invisibles lors de déplacements. La classe `GoogleMap` n'expose pas de collection de marqueurs afin que celle-ci ne soit pas modifiée de l'extérieur, il convient donc de garder une référence vers la liste des marqueurs affichés sur la carte, à synchroniser dès qu'un élément est enlevé ou ajouté à la carte. Le développeur peut donc conserver deux listes en mémoire, une liste de l'ensemble des points à afficher (`_allMarkers`) et une liste des points qui sont sur la carte actuellement (`_markersOnMap`). Lorsque la caméra de la carte change de point de vue, il pourrait exécuter ce code :

```
LatLngBounds mapBounds = _map.getProjection().getVisibleRegion()
.latLngBounds;
for (Marker marker : _allMarkers) {
    if (mapBounds.contains(marker.getPosition()))
        if (!_markersOnMap.contains(marker)) {
            Marker addedMarker = _map.addMarker(new MarkerOptions());
            _markersOnMap.add(addedMarker);
        }
    } else {
        if (_markersOnMap.contains(marker)) {
            marker.remove();
            _markersOnMap.remove(marker);
        }
    }
}
```

D'aucuns diraient qu'il est plus simple d'effectuer à chaque fois un appel à `_map.clear()` pour ensuite rajouter les marqueurs visibles à l'écran. Cette manière de faire entraînerait un effet de clignotement intempestif désagréable au moindre déplacement de l'utilisateur. C'est pourquoi il est préférable de procéder point par point.

Conserver un certain naturel en élargissant les bornes

Cependant, l'utilisateur risque de se déplacer rapidement d'un point à l'autre, ou encore de revenir sur ses pas. Les repères vont alors s'afficher puis disparaître pour s'afficher une nouvelle fois, y compris lors de menus déplacements. Pour éviter cet effet d'apparition, il serait élégant de charger toujours un peu plus que ce qui est strictement visible à l'écran : Fig.2. Des stations qui seraient visibles à quelques pixels de déplacement près pourraient être ajoutées à la carte afin que l'utilisateur ne perçoive pas ces perpétuels ajouts et suppressions. De plus, cela permettrait d'améliorer quelque peu les performances en

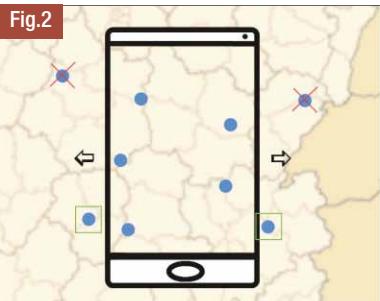


Fig.2 réduisant la fréquence des ajouts et des suppressions. Ainsi, simplement avant de rentrer dans la boucle du code ci-dessus du rendu des marqueurs, le développeur pourrait agrandir les bornes visibles à la caméra qui permettent de déterminer l'ajout ou non d'un marqueur :

```
LatLngBounds mapBounds = _map.getProjection().getVisibleRegion()
    .latLngBounds;
LatLngBounds visibleBounds = GeoHelper.extendLimits(bounds, 3);
for (Marker marker : _allMarkers) { etc...
```

Le code de la fonction `extendLimits` pourrait simplement consister en une extension du rectangle de projection géographique :

```
public static LatLngBounds extendLimits(LatLngBounds bounds,
    int ratio) {
    double extendedLongitude = Math.abs(bounds.northeast.longitude
        - bounds.southwest.longitude) / ratio;
    double extendedLatitude = Math.abs(bounds.northeast.latitude
        - bounds.southwest.latitude) / ratio;
    LatLng topRight = null;
    LatLng bottomLeft = null;
    // Longitudes
    double topRightLongitude = bounds.northeast.longitude
        + extendedLongitude;
    double bottomLeftLongitude = bounds.southwest.longitude
        - extendedLongitude;
    // latitudes
    double topRightLatitude = bounds.northeast.latitude + extendedLatitude;
    double bottomLeftLatitude = bounds.southwest.latitude - extendedLatitude;
    topRight = new LatLng(topRightLatitude, topRightLongitude);
    bottomLeft = new LatLng(bottomLeftLatitude, bottomLeftLongitude);
    return new LatLngBounds(bottomLeft, topRight);
}
```



Fig.3



Fig.4

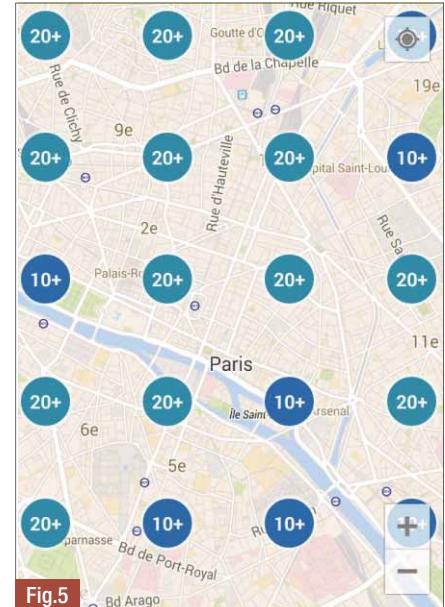


Fig.5

Regrouper les points lors de dézooms

Charger les points en mode *lazy* est une première étape. Mais si des milliers de points relativement proches sont à placer sur la carte, même si le rendu est circonscrit à la zone visible, une multitude de points se chevauchant risquent d'apparaître et de nuire à la lisibilité de la carte. C'est pour de tels scénarios que la technique du *clustering* est intéressante : au lieu d'afficher chaque point sur la carte, le développeur les regroupe en paquets plus ou moins importants selon le niveau de zoom. Ces paquets peuvent par exemple se représenter par des cercles, avec des chiffres, indiquant le nombre de repères contenus dans un paquet : Fig.3. Or, depuis la version 2 de l'API Google Maps, des librairies open source ont émergé proposant des solutions de clustering toutes faites, et cependant extensibles. Deux librairies retiennent notamment l'attention, Google Maps Android API Utility et Android Maps Extension.

Des librairies de clustering extensibles

La première de ces deux librairies, Google Maps Android API Utility, embarque deux algorithmes différents de clustering, détaillés plus avant sur <https://developers.google.com/maps/articles/toomanymarkers>. Accessoirement, elle possède aussi des classes de rendu capables de lancer de jolies animations sur les clusters lorsque ceux-ci se multiplient ou se rejoignent.

Le premier algorithme, exprimé par la classe `GridBasedAlgorithm`, est le plus simple : les points sont placés dans une grille, dont la taille des cases varie en fonction du zoom. En fonction de la case dans laquelle les points se trouvent, ils sont regroupés entre eux et ne forment plus qu'un point au centre de la case. S'il est performant, cet algorithme peut s'avérer plutôt approximatif quant à la position des clusters. En effet, si deux points sont très rapprochés mais ne se situent pas dans la même case, ils seront chacun dispersés dans un cluster différent. De plus, lorsque les clusters sont nombreux, l'utilisateur aperçoit clairement une grille de points, parfaitement alignés, ce qui n'est pas toujours du meilleur effet. Ainsi, cet algorithme peut être mis en œuvre lorsque les performances sont cruciales et lorsque les points ne sont pas trop nombreux aux mêmes endroits Fig.4..

Le second, exprimé par la classe `NonHierarchicalDistanceBasedAlgorithm` est plus complexe mais aussi plus intelligent. Chaque point est considéré comme un centre de gravité et un cluster potentiel. Ensuite, chaque point est évalué et ajouté à un cluster si sa distance est inférieure à une certaine valeur. Et sinon, il reste lui-même un cluster. Un ajustement a ensuite lieu si un point a été rattaché à un cluster alors qu'une plus petite distance encore le sépare d'un autre cluster. Cet algorithme est intéressant lorsque de nombreux points coexistent de manière rapprochée sur une carte Fig.5..

Ainsi, on préférera utiliser le second algorithme, sans doute moins rapide à mettre en place, mais beaucoup plus précis et plus agréable à observer. Mais mettre en place un mécanisme de clusters ne dispense pas le développeur, qui possède beaucoup de points à placer, de charger ceux-ci en mode *lazy* comme indiqué précédemment. Malheureusement, Google Maps API Utility ne permet pas nativement de chargement des points en mode *lazy*. Cependant, la librairie est hautement extensible, elle fonctionne notamment par contrats. Il suffit alors d'étendre une interface chargée du rendu des points pour ajouter un fonctionnement *lazy*. L'API contient par défaut une classe `DefaultClusterRenderer`, il serait alors possible de créer un `LazyClusterRenderer`, qui se chargerait d'ajouter les points, comme vu précédemment, en fonction de leur visibilité à l'écran : [Fig.6](#).

C'est la méthode `onClustersChanged` qui est surtout intéressante à surcharger car elle est automatiquement appelée par le gestionnaire de clusters de l'API, la classe `ClusterManager`, lorsque la caméra change de niveau de zoom. La librairie n'étant pas prévue pour un chargement *lazy* des points, il peut être nécessaire d'étendre `ClusterManager`, notamment à cause de l'implémentation de cette méthode :

```
@Override
public void onCameraChange(CameraPosition cameraPosition) {
    if (mRenderer instanceof GoogleMap.OnCameraChangeListener) {
        ((GoogleMap.OnCameraChangeListener)mRenderer).onCameraChange(cameraPosition);
    }
    // Don't re-compute clusters if the map has just been
    // panned/tilted/rotated.
    CameraPosition position = mMap.getCameraPosition();
    if (mPreviousCameraPosition != null && mPreviousCamera
    Position.zoom == position.zoom) {
        return;
    }
}
```

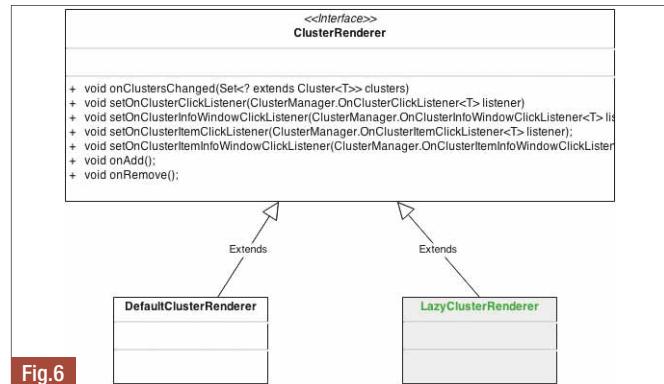


Fig.6

```
mPreviousCameraPosition = mMap.getCameraPosition();
cluster();
}
```

Comme le montre ce code et comme l'indique le commentaire, le `ClusterManager` ne provoquera pas de clustering si la caméra est déplacée. Or, si l'on veut charger les points en mode *lazy*, il faut appeler la méthode `cluster` quoi qu'il arrive. Cela ne signifie pas que l'algorithme sera exécuté à chaque fois, car, en coulisse, la librairie Google Maps API Utility, suivant le pattern décorateur, conserve un cache des clusters rangés selon un niveau de zoom sous forme de dictionnaire clé-valeur, et ce, grâce à une classe décorateur, `PreCachingAlgorithmDecorator`. Cela signifie donc simplement qu'il faut, à chaque déplacement ou changement de zoom, passer les bons clusters au `ClusterRenderer` pour que les points se mettent à jour à l'écran [Fig.7](#). La seconde librairie, *Android Maps Extension*, ne propose que le premier algorithme qui est basé sur le système de grilles.

Elle a cependant l'avantage de proposer nativement un chargement *lazy*. Une application de test, *Android Maps Extensions Demo*, est à télécharger sur le store de Google. Si l'utilisateur se déplace vite, il peut alors voir les points se redessiner sur la carte, ce qui témoigne d'un chargement dynamique. L'effet de grille que l'on pourrait chercher à éviter est atténué dans cette démo, du fait de la relative dispersion des points sur le globe.

Cette librairie est également extensible et le développeur peut très bien proposer sa propre implémentation de l'interface `ClusteringStrategy`, et implémenter l'algorithme basé sur les distances. Ainsi, à l'inverse de la librairie précédente, il n'aurait pas à implémenter de `renderer` car le chargement est déjà *lazy*, par contre, il aurait à fournir un autre algorithme pour éviter un effet de grille en cas de nombreux points proches.

Conclusion

Ainsi, on pourra retenir plusieurs bonnes pratiques lorsque de trop nombreux éléments sont à ajouter sur une carte Google Maps, laquelle risque d'être souvent manipulée par l'utilisateur. Le développeur favorisera le chargement des points en mode *lazy* pour éviter une carte surchargée, lente et peu réactive, surtout sur des téléphones d'entrée de gamme. Il convient aussi de garder en tête qu'il s'agit d'être le plus économique possible : de ne supprimer de la carte que ce qui n'y apparaîtra plus et d'éviter de tout supprimer pour tout ajouter à chaque fois. Ce chargement dynamique ne dispense pas de mettre en place un mécanisme de clustering, notamment lorsque les points sont nombreux et rapprochés. Pour cela, depuis la version 2 de Google Maps pour Android, des librairies de clustering ont vu le jour et proposent différents mécanismes de clustering extensibles et personnalisables.

Anna Yafi | consultant indépendant Android et .NET
annayafi@gmail.com

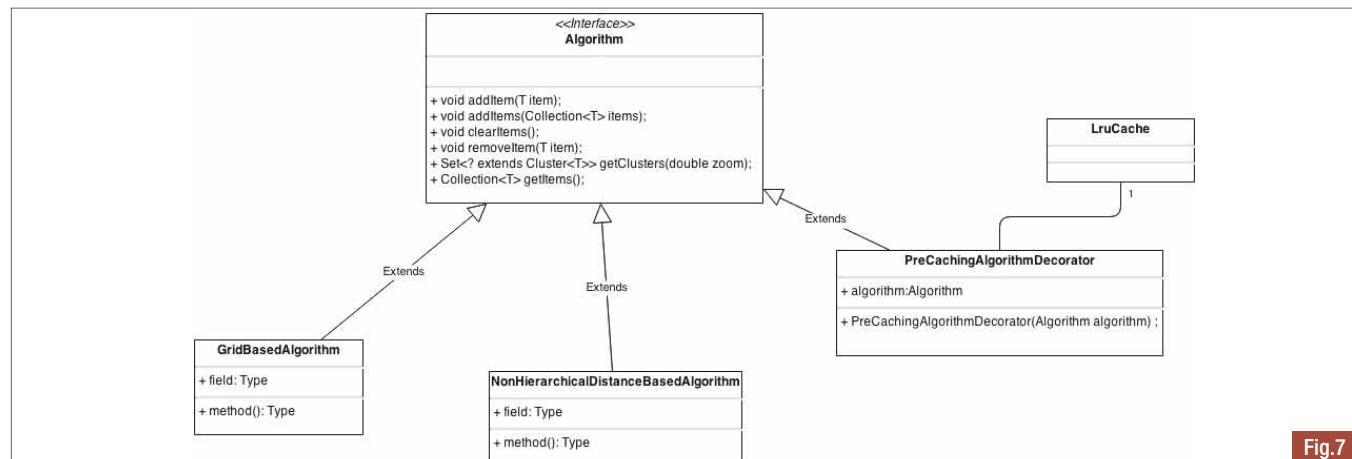
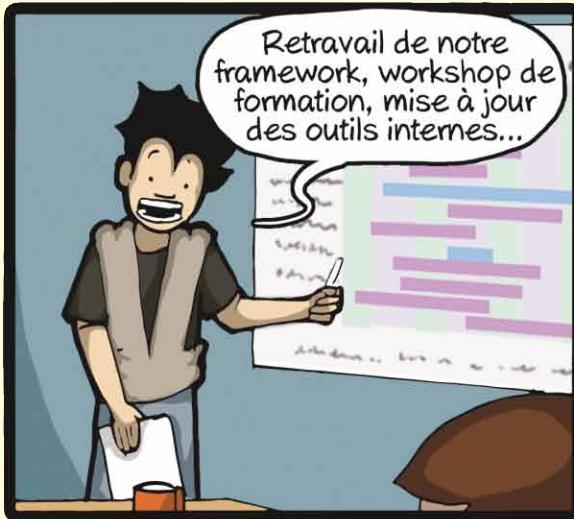


Fig.7



Abonnement : Programmez, 17, Route des Boulangers, 78926 Yvelines Cedex 9 - Tél. : 01 55 56 70 55 - abonnements.programmez@groupe-gli.com - Fax : 01 55 56 70 91 - du lundi au jeudi de 9h30 à 12h30 et de 13h30 à 17h00, le vendredi de 9h00 à 12h00 et de 14h00 à 16h30. **Tarifs** abonnement [magazine seul] : 1 an : 11 numéros France métropolitaine : 49 € - Etudiant : 39 € - CEE et Suisse : 55,82 € - Algérie, Maroc, Tunisie : 59,89 € - Canada : 68,36 € - Tom : 83,65 € - Dom : 66,82 € - Autres pays : nous consulter. **PDF** : 30 € (Monde Entier) souscription exclusivement sur www.programmez.com



Directeur de la publication & rédacteur en chef : François Tonic

Ont collaboré à ce numéro : Sylvain Saurel, François Moog
Secrétaire de rédaction : Olivier Pavie
Experts : F. Brindille, Jean Vidame, Frédéric Mazué, G. Renard, Z. Chalal, A. Yafi, A. Zamouchi, M. El Farouk, M. Frey, C. Liban, J-B Raffin, P-E Faidherbe, J. Laborde, P-A Louyat, T. Royer, N. Freier, C. villeneuve, M. Hubert, D. Marchand, V. Kovalsky David, M. Gaul, M. Dunand, R. Jarraud, J. Allébe, C. Zhou, O. Pekelman, L. Cros, J. Canler, D. Wurtseisen, A. Jolly, A. Barralon, M. Simonet, A. Guidet, C. Lakech, Ch'ti JUG Leader, F. Hebrard, L. Harivel

Une publication Nefer-IT
7 avenue Roger Chambonnet
91220 Brétigny sur Orge
redaction@programmez.com
Tél. : 01 60 85 39 96

Crédits couverture : © Commitstrip
Maquette : Pierre Sandré

Publicité : PC Presse,
Tél. : 01 74 70 16 30, Fax : 01 41 38 29 75
pub@programmez.com

Imprimeur : S.A. Corelio Nevada Printing, 30 allée de la recherche, 1070 Bruxelles, Belgique.
Marketing et promotion des ventes :
Agence BOCONSEIL - Analyse Media Etude
Directeur : Otto BORSCHA oborscha@boconseilame.fr
Responsable titre : Terry MATTARD
Téléphone : 0987320934
Ce numéro comporte un encart jeté PC Soft pour les abonnés - France métropolitaine uniquement.

Contacts

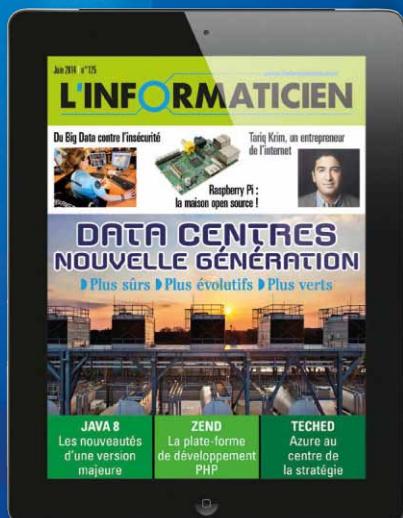
Rédacteur en chef : ftonic@programmez.com
Rédaction : redaction@programmez.com
Webmaster : webmaster@programmez.com
Publicité : pub@programmez.com
Événements / agenda : redaction@programmez.com
Dépot légal : à parution - Commission paritaire : 1215 K 78366 - ISSN : 1627-0908
© NEFER-IT / Programmez, juin 2014
Toute reproduction intégrale ou partielle est interdite sans accord des auteurs et du directeur de la publication.



Sur abonnement ou en kiosque

Le magazine des pros de l'IT

Mais aussi sur le web



Ou encore sur votre tablette

COMMANDÉZ WINDEV MOBILE 19 OU WEBDEV 19 OU WINDEV 19 ET RECEVEZ 2 GALAXY S5

Aucun abonnement à souscrire pour bénéficier de cette offre.

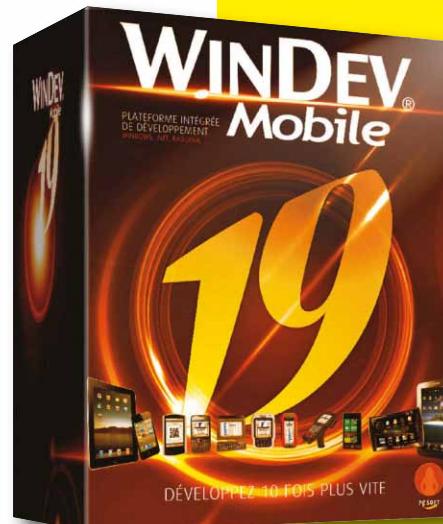


Le tout dernier smartphone de SAMSUNG: GALAXY S5 • Ecran 5,1" 1920 x 1080 • Android 4.2.2 • APN 16M • Caméra frontale 2M • Capture vidéo full HD (1920 x 1080) • 145 grammes • Quadri Bande • 4G • GPS • Etanche 30mn sous 1 mètre d'eau • Empreinte digitale • Emplacement carte Micro SD • Wifi • NFC • USB 3.0 • Etc...

Ou choisissez 2 Tablettes Galaxy Tab 4 (nouveau modèle) ou 2 PC portables Samsung ou encore 1 Télé Samsung 140cm.

Pour bénéficier de cette offre exceptionnelle, il suffit de commander WINDEV Mobile 19 (ou WINDEV 19, ou WEBDEV 19) chez PC SOFT au tarif catalogue avant le 4 juillet 2014. Offre réservée aux sociétés, administrations, mairies, GI&E et professions libérales..., en France métropolitaine. Aucun abonnement n'est à souscrire pour bénéficier de cette offre. Le développement pour Android et iOS s'effectue avec WINDEV Mobile ou WEBDEV. Le développement pour Windows et Linux s'effectue avec WINDEV ou WEBDEV. Voir tous les détails et des vidéos sur : www.pcsoft.fr

Le Logiciel et le matériel peuvent être acquis séparément; merci de vous connecter au site www.pcsoft.fr pour consulter la liste des prix et les dates de disponibilité. Tarifs modifiables sans préavis.



BONISSEZ SUR
L'OPÉRATION
2 POUR 1 EURO DE +

JUSQU'AU
1 JUILLET

WINDEV Mobile 19 permet de créer facilement et rapidement des applications pour iOS, Android, Windows Phone et Windows Mobile. Liaison facile à votre SI et à toute base de données.

Fournisseur Officiel de la
Préparation Olympique

www.pcsoft.fr

Des centaines de références sur le site



WINDEV AGL N°1 en FRANCE

