

PROGRAMMEZ !

#211 - octobre 2017

le magazine des développeurs



Google vs Apple

ARCore vs ARKit : que la guerre de la réalité augmentée commence

Challenge

**Créer un jeu
en 100 lignes
de code !**



© Yuravector

Hacking

**La menace bien
réelle de
TheShadowBrokers**

**Créer sa boutique
eCommerce** 1ere partie

**.net est mort
vive .net standard**

**Les ressources
indispensables
pour coder
rapidement
avec Unity**

Maker

**Connectez votre
imprimante 3D
en WiFi**



WINDEV®

DÉVELOPPEZ 10 FOIS PLUS VITE



Windows - Linux - Mac - Internet - iOS - Android

**WINDEV 22 EST UN ATELIER DE
DÉVELOPPEMENT COMPLET.
CRÉEZ VOS PROPRES APPLICATIONS.**

Gestion du cycle de vie complet: Idée, Conception, Développement, Génération, Déploiement, Exploitation • Un code multi-plateformes Windows, Linux, Java, Internet, Mobiles • Environnement ALM complet • Toutes les bases de données sont supportées, Big Data • Inclus: HFSQL, base de données locale, Client/Serveur, cluster, embarquée et cloud • Puissant RAD • Intégration continue • Tableau de bord de vos applications • Audit statique & dynamique • Générateur de fenêtres (UI) visuel & intuitif • Utilisation facile de charte graphique • Héritage et surcharge d'interface • Tous les champs (contrôles) sont très puissants et livrés en standard: Champ de saisie, Tableau croisé dynamique (cube), Champ Planning, Champ Diagramme de Gantt, Champ Tableau de bord, Champ Table, Champ Graphe, etc • FAA: chaque application bénéficie automatiquement de Fonctionnalités Automatiques: export vers Excel, vers Word, envoi d'email, etc • Sécurité : Mot de passe de vos applications • Puissant générateur de rapports et codes-barres • Langage de 5ème génération: WLangage • Editeur de code intuitif avec puissant débogueur • Tests unitaires et tests automatiques • Versioning (GDS/SCM) • Webservices SOAP et Rest • Modélisation Merise et UML • .NET, 3-Tier, MVP • Support de tous les standards: XML, USB, Bluetooth, NFC, J2EE, OLE, ActiveX, RPC, SaaS, SMTP, FTP, OPC, DLNA, IoT, Sockets, API, Webservices... • Lien avec Lotus Notes, SAP, Google, Outlook • Multimédia, Domotique • Livré avec des centaines d'exemples et d'assistants • Génère le Dossier technique d'un clic • Télémétrie pour connaître l'utilisation réelle de vos applications • Générateur d'aide • Support de 64 langues étrangères par application • Générateur de procédures d'installation: local, CD, USB, Internet, Réseau, Push... • Robot de surveillance: surveillez vos applications • Gestion des suggestions et incidents utilisateurs • Support Technique Personnalisé Gratuit* • ...

**CONSULTEZ 100 TÉMOIGNAGES
DE SOCIÉTÉS PRESTIGIEUSES SUR LE SITE PCSOFT.FR**

CHATEAU MARGAUX • PHILIPS • VINCI AUTOROUTES • QUICK • LYONNAISE DES EAUX • SYSTEME U • FEDERATION FRANCAISE DE BASKET • BOLORRE • CASIO • TAITTINGER • CCI BORDEAUX • VOLVO • SIEMENS VAI • TRUFFAUT • AIR CALÉDONIE • HONDA EUROPE • COMTESSE DU BARRY • PRONUTIA • PARIS ENGINEERING SCHOOL • ÉCOLE DIRECTE • HOPITAUX DE PARIS • SOCIÉTÉ GÉNÉRALE • PHOTOMATON® • GROUPAMA • CPAM • ...

Elu
«Langage
le plus productif
du marché»

**VERSION
EXPRESS
GRATUITE**
Téléchargez-la !



WWW.PCSOFT.FR

Tél : 04 67 032 032

**WINDEV
VU À LA TÉLÉ**
SUR TF1, M6,
BFM TV, FRANCE 2,
CANAL+

La documentation

Ah la fameuse documentation technique ! Elle est parfois redoutée par le développeur qui doit : soit la lire, soit l'écrire. Parfois elle est aussi vieille que le code, quand elle existe. Format papier ou électronique, peu importe, cette doc fait partie de tout projet de développement.

La documentation technique n'est pas ce que le développeur aime faire. C'est souvent une corvée mais si elle est bien faite et à jour, elle peut vous aider à vous replonger dans un code. Car reprendre un code que l'on écrit 6-8 mois auparavant n'est jamais facile, et on oublie vite ce que l'on a voulu faire. Reprendre le code d'un autre développeur est tout aussi difficile, surtout si celui-ci n'est pas bien écrit. La documentation technique peut rapidement devenir votre meilleur ami.

Le format de cette doc est important aussi. Le traditionnel document Word (ou LibreOffice, ou PDF) est peu souple. Les générateurs de documents, comme l'outil Sphinx, apportent un réel plus comme vous le verrez dans ce numéro.

Bref, RTFM* a de beaux jours

Nous parlerons aussi de eCommerce. Créer une boutique en ligne paraît simple surtout quand on utilise un CMS. Des plugins mèchent le travail. Cependant, elle ne s'improvise pas et elle répond à de multiples contraintes et des règles précises. Comme vous le verrez dans notre dossier, en 2 parties, il s'agit de définir son projet, de savoir ce que l'on va y mettre, comment gérer les stocks, le paiement, le responsive design, définir l'expérience utilisateur, créer le suivi client, etc.

Autre sujet chaud depuis quelques semaines : la réalité augmentée et la rivalité Apple – Google. Apple avait lancé la première pique dès juin dernier avec son ARKit. Google a réagi durant l'été en dévoilant son ARCore. Si le but est de simplifier la réalité augmentée pour le développeur et en faciliter l'accès pour les utilisateurs, il s'agit d'un véritable bras de fer qui s'engage entre les deux géants. Mais les deux terminaux peuvent parfaitement cohabiter, l'un dans iOS, l'autre dans Android. Il est bien trop tôt pour savoir qui va prendre le dessus mais ces technos ouvrent de belles opportunités aux développeurs. Découvrez notre dossier spécial !

Bon code !

* read the fucking manual



Agenda	Tableau de bord
6	4
ABONNEZ-VOUS !	
11	18
Recrutement	
12	Rendre les développeurs heureux Partie 2
Dossier eCommerce partie 1	
23	Outils APM
Projet Rome	
59	Un jeu Android en -100 lignes
	55
GDPR	Anciens numéros
53	47
Pourquoi C++ en 2017 ?	
33	Word to web
ARKit vs ARCore	
39	
C++ Builder	
37	Les outils APM
CommitStrip	
82	
Coder une cracktro sur Amiga	
77	Unity
GoReplay	
67	74
	Maker
64	Imprimante 3D & WiFi
.Net Standard 2.0	
70	61

Dans le prochain numéro !
Programmez! #212, dès le 3 novembre 2017

Infrastructure as Code

Codons notre infrastructure !

Choisir sa base de données

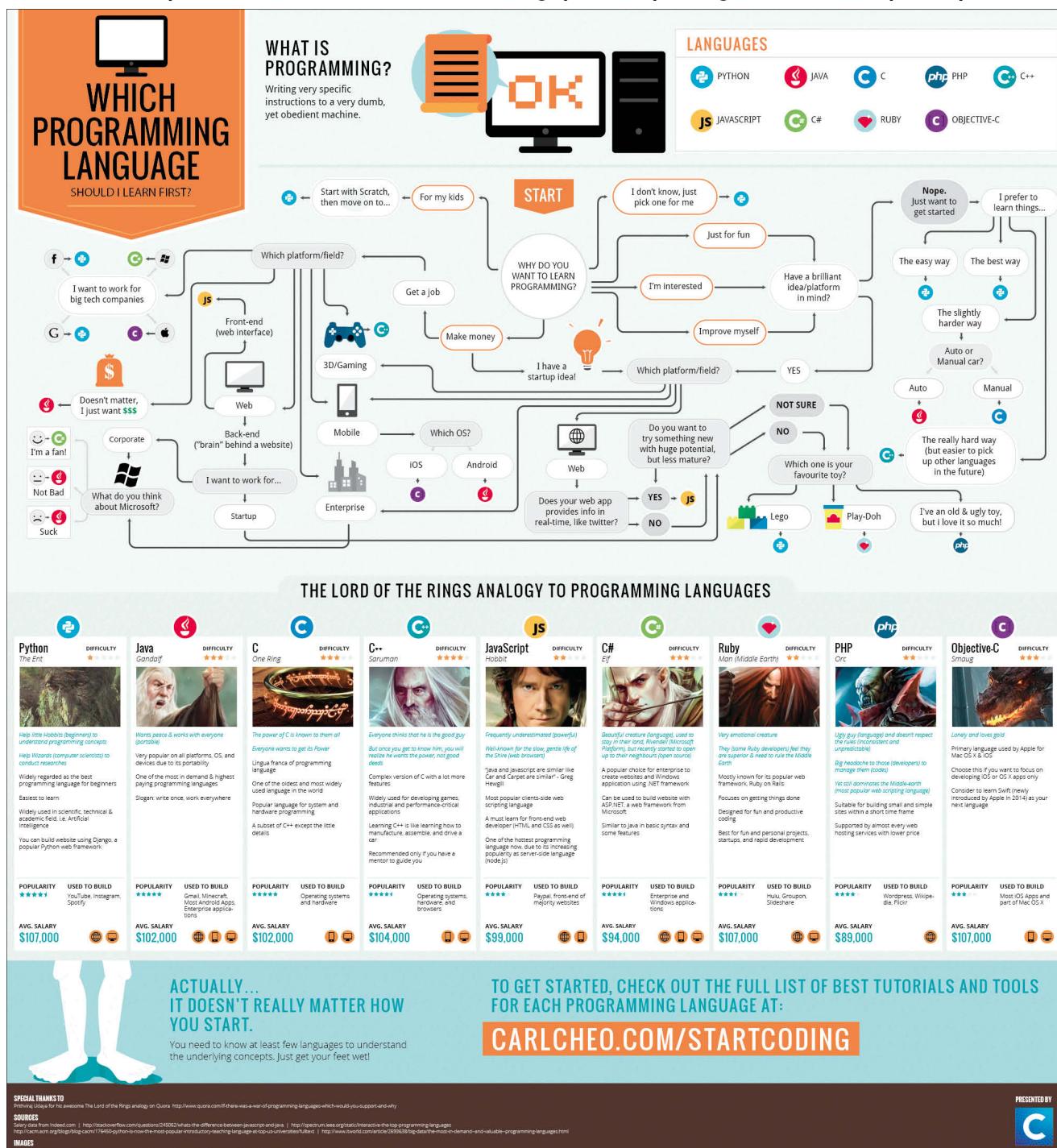
Quelle base de données choisir ? Quels critères techniques retenir ? Version locale ou version cloud ? Les réponses dans notre dossier spécial.

Linux au cœur de Windows 10

Le sous-système Linux de Windows 10 offre des usages inédits pour les développeurs et les sysadmins. Présentation complète.

Quel langage de programmation apprendre en premier ?

Au hasard de nos explorations, nous sommes tombés sur cette infographie. Bien qu'en Anglais, elle mérite une petit coup d'oeil, non ?



INDEX TIOBE DU MOIS

09/2017	09/2016	Evolution	Langage	%	Evolution en %
1	1	=	Java	12.687%	-5.55%
2	2	=	C	7.382%	-3.57%
3	3	=	C++	5.565%	-1.09%
4	4	=	C#	4.779%	-0.71%
5	5	=	Python	2.983%	-1.32%
6	7	▲	PHP	2.210%	-0.64%
7	6	▼	JavaScript	2.017%	-0.91%
8	9	▲	VB.NET	1.982%	-0.36%
9	10	▲	Perl	1.952%	-0.38%
10	12	▲	Ruby	1.933%	-0.03%

Peu de changements ce mois-ci. On constate une belle remontée de Ruby et une légère de PHP.

programmez! - octobre 2017

2,5 milliards de transactions par jour en France passant par des systèmes Z selon IBM. On vous avait bien dit que le mainframe était mort.

Juicero voulait révolutionner le jus de fruit. Bon ok vendre une machine 400 \$ et des recharges propriétaires c'était un peu osé non ? Résultat : le grand fiasco pour les dizaines de millions \$ investis.

PrimeSense a enfin une utilité pour Apple ! A l'origine de Kinect, PrimeSense avait été racheté en 2013. Cette technologie a sans aucun doute servi à concevoir une partie de FaceID d'iPhone X

Les processeurs Sparc sont-ils morts ?

Entre rumeurs de licenciement des équipes et de la fin de Solaris, Oracle a dévoilé mi-septembre une nouvelle génération de Sparc, les M8.

DÉVELOPPEMENT PROFESSIONNEL CROSS-PLATEFORM NATIF WINDEV / WINDEV Mobile

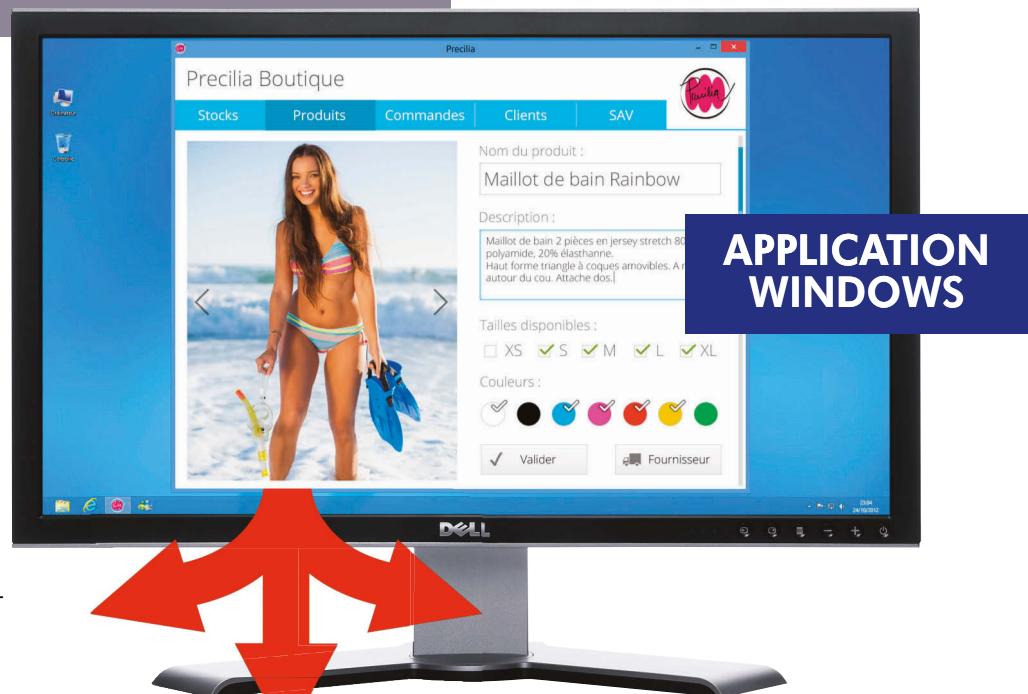


**VOTRE SOCIÉTÉ A UN
«EXISTANT» WINDEV ?**

**PASSEZ-LE SUR MOBILE EN
QUELQUES HEURES !**

Les applications WINDEV passent très rapidement sur mobile: retailliez les fenêtres pour les adapter à la taille des mobiles, supprimez les traitements qui ne sont pas nécessaires sur mobile, adaptez un peu le code, et hop, vous voilà en possession d'une superbe application mobile ! La portabilité des projets entre WINDEV et WINDEV Mobile vous permet de disposer d'applications mobiles très performantes en un délai record.

WINDEV 22 est également compatible avec WEBDEV, pour passer vos applications WINDEV en sites Web gérant des données (applications Web).



**APPLICATION
WINDOWS**



TOUT EST NATIF

Avec WINDEV, vos projets sont cross-plateformes: le langage est le même, le code aussi, mais également les fenêtres, les états, les requêtes, les bases de données, les analyses...

Tél : 04 67 032 032 info@pcsoft.fr
WWW.PCsoft.FR

octobre

DEVOPS REX 2017**2 octobre / Paris**

Une conférence 100 % devops avec des retours d'expérience toute la journée ! Une excellente occasion pour rencontrer la communauté et partager son aventure devops et poser toutes les questions aux experts !

Site : <https://www.devopsrex.fr>**MICROSOFT EXPERIENCES'17****3 & 4 octobre / Paris**

Cette année, les MS Experiences parleront de trois grands thèmes : intelligence artificielle, les nouvelles méthodes et pratiques de travail, et la confiance numérique (blockchain, identité, etc.). L'évènement parlera bien entendu aux responsables IT et aux développeurs. Le format reste grossso modo identique à l'édition 2016 : plénières, ateliers, sessions.

LDAPCON**19 & 20 octobre / Bruxelles**

Cette 6e édition de la conférence européenne de la plateforme d'annuaire LDAP se déroulera à Bruxelles. On y parlera avant tout infrastructure, protocole, architecture LDAP, serveur. LDAPCon est une conférence internationale LDAP qui se déroule une fois tous les deux ans. Il y a deux ans, elle a eu lieu à Édimbourg et il y avait environ 120 participants. Nous attendons approximativement le même nombre de participants cette année à Bruxelles.

Pour en savoir plus : <https://ldapcon.org/2017/fr/>**FORUM PHP 2017****26 & 27 octobre / Paris**

L'incontournable cycle de conférences parisien dédié à PHP attend toute la communauté PHP au Marriott Rive Gauche, les 26 et 27 octobre. Au programme, deux jours de conférences, ateliers pratiques et démonstrations, dans une ambiance festive propre à l'esprit de la communauté. En plus d'un programme pointu et de speakers reconnus, l'audience pourra compter sur la présence des entreprises qui font vivre l'écosystème au quotidien, des moments de convivialités tout au long des deux jours, et, cerise sur le gâteau, la présence du tout nouvel éléphant AFUP, disponible en quantité limitée uniquement pour nos visiteurs ! Ne manquez pas l'événement phare de l'automne organisé par l'AFUP.

Pour en savoir plus : event.afup.org

novembre

NI DAYS PARIS**7 novembre / Paris**

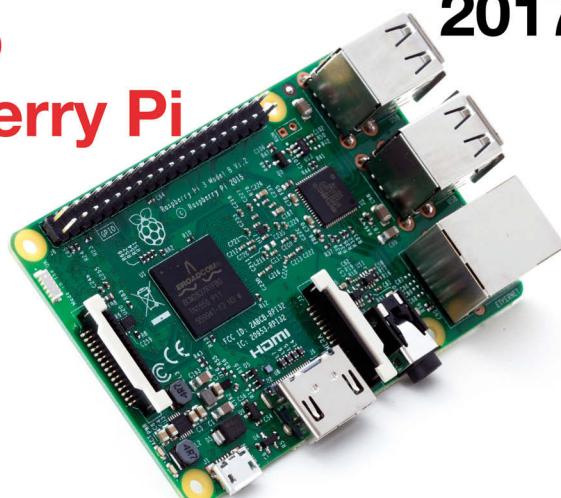
National Instruments est un constructeur / éditeur phare du monde de l'embarqué et de l'industrie. Chaque année, la journée NI Days permet de découvrir les dernières versions des outils d'instrumentation, l'écosystème. De nombreuses sessions sont jouées durant la journée. Pour en savoir plus : <https://www.ni.com/nidays/>

OPENSTACK DAY FRANCE**21 novembre / Paris**

OpenStack Day France sera l'opportunité de rencontrer des représentants de la Fondation OpenStack, les acteurs de l'écosystème du cloud en Open Source, les utilisateurs finaux et les contributeurs de la communauté de développeurs en France. OpenStack Day France est organisé par l'Association OpenStack-FR en collaboration avec Osones et Red Hat. L'évènement compte déjà Osones et Objectif Libre comme sponsors.

Pour en savoir plus : openstackdayfrance.fr**DevCon #4****26/10****100 %
Raspberry Pi
& Co.****2017**

/IoT
/Serveur
/Production
/Docker

**Inscrivez-vous sur www.programmez.com****Accueil dès 13h30 !**

	Amphi	Salle 2
14h – 14h45	plénière #1 Introduction à la journée (François Tonic) Keynote par IBM	-
14h55 – 15h40	session #1 Session (par Etienne Deneuve, Cellenza)	session #2 Showcase maison automatisée avec Raspberry Pi (par Sébastien Warin)
15h50 – 16h15	pause café Quickie #1 i-Guard choisit la Pi (par Thierry Goigoux)	pause café Quickie #2 Une PI pour contrôler l'hydrométrie (par Franck Nguissan)
16h15 – 17h	session #3 Coding4Fun en Raspberry Pi (par Salah Amer)	session #4 Redéfinir l'interface et l'interaction (par François Tonic, Programmez!)
17h10 – 17h50	plénière #2 Pi + Docker pour créer une infrastructure serverless (par Estelle Auberix)	-
17h55 – 18h30	session Q&A	-
18H30 – 20H	pizza party	-

Les intervenants :

Estelle Auberix, Sébastien Warin, Etienne Deneuve, François Tonic, Franck Nguissan, Salah Amer, Thierry Goigoux

Sponsor : IBM

Partenaires techniques :
Cellenza, i-Guard

Où :
Ecole 42
96 Boulevard Bessières
75017 Paris

Réservez dès aujourd'hui votre place sur :

<https://www.programmez.com/content/devcon-4-100-raspberry-pi-co>

ikoula
HÉBERGEUR CLOUD

PRÉSENTE

CLOUD**IKOULA**ONE



 Le succès est votre prochaine destination

MIAMI SINGAPOUR PARIS
AMSTERDAM FRANCFORT ---

CLOUD**IKOULA**ONE est une solution de Cloud public, privé et hybride qui vous permet de déployer **en 1 clic et en moins de 30 secondes** des machines virtuelles à travers le monde sur des infrastructures SSD haute performance.

 www.ikoula.com

 sales@ikoula.com

 01 84 01 02 50

ikoula
HÉBERGEUR CLOUD


Black Hat 2017 : La menace de l'invisible TheShadowBrokers

• Véronique Loquet, Spécialiste open source & Infosec, [@vloquet](#)

Si le mystère reste entier quant à son identité et ses motivations, le hacker ou le groupe de hackers

TheShadowBrokers, est désormais l'un des plus controversés au monde. Il a émergé en ligne en août 2016, déclarant être en possession de secrets classés, d'outils et notes opérationnelles en provenance de l'agence de renseignement américaine, la NSA.



Depuis lors, l'entité a orchestré ses fuites de données hautement confidentielles, en propageant publiquement et à intervalles réguliers, un large éventail d'information sensible, d'outils et de modes opératoires offensifs. Ces dispositifs de cyber-espionnage ont été dérobés à Equation Group, le surnom donné à TAO (Tailored Access Operations), la division offensive de la NSA, dont les cibles s'étendent à une cinquantaine de pays. Ils visent tant les institutions gouvernementales et militaires, que les secteurs stratégiques de l'industrie comme l'énergie, la finance, les télécommunications ou encore les médias. Parmi les données divulguées, figurent également les noms d'agents opérationnels de la NSA. La puissance de l'arsenal volé et l'ancienneté de certains outils renseignent a posteriori sur les nombreuses années d'espionnage massif de l'agence américaine.

Lors de la conférence Black Hat à Las Vegas, Matthieu Suiche, hacker français spécialiste en reverse engineering et fondateur de la société de sécurité Comae Technologies, a présenté les résultats de l'autopsie de ces révélations dans une conférence intitulée « Cyber Fear Game-Changers ».

Une stratégie de diffusion chaotique

Les toutes premières indiscretions disponibles gratuitement contiennent 300Mo de données, dont une livraison d'exploits fonctionnels affectant des équipements de Cisco, Juniper, Fortinet ou Huawei, des outils de reconnaissance et d'extraction de données avec leurs manuels d'utilisation. Une gratification accompagnée d'un message posté sur Pastebin : « Nous avons les cyber-armes des créateurs de Stuxnet, Duqu et Flame. Kaspersky les a nom-

més Equation Group. Nous avons suivi leur trafic, nous possérons leurs sources. Nous les avons piratés. » Le teasing fait l'effet d'une bombe et TheShadowBrokers passe à la vitesse supérieure avec la mise aux enchères des archives suivantes pour 1 million de Bitcoins, soit plus de 2,7 milliards de dollars. Échec du modèle qui ne trouve pas preneur, révision tactique, cette fois les outils volés sont proposés à l'unité sur les plateformes BitTorrent et ZeroNet, pour un équivalent total de près de 6 millions d'euros. Le business ne décolle toujours pas. En juin, revirement stratégique, va pour un service d'abonnement mensuel équivalent à 22000 dollars, « TheShadowBrokers Data Dump of the Month », dont le prix annoncé doublera le mois suivant. Leur porte-monnaie Bitcoin est alors abandonné au profit de la crypto-monnaie Zcash, moins traçable. L'offre comprend des outils pour pirater routeurs, navigateurs web et Windows 10, il serait même possible d'obtenir des données sur les programmes d'armement nucléaire russe, chinois, iraniens et nord-coréen.

Une communication aux accents troubles

L'agenda des fuites du mois d'avril concorde avec le changement de politique des États-Unis en Syrie, qui serait le déclencheur de nouvelles divulgations. TheShadowBrokers légitime ses révélations suite à l'attaque antimissile américaine sur une base syrienne, et reproche au Président Trump d'avoir abandonné ceux qui l'ont élu.

Tandis que ces messages sont rédigés dans un anglais approximatif qui suggère des auteurs non-anglophones, diverses analyses penchent pour un défaut de grammaire intentionnel, une tactique de manipulation supplémentaire afin

de brouiller les pistes. Matthieu Suiche insiste d'ailleurs sur leur excellente connaissance de la division TAO et déclare : « Il serait plus logique d'y voir la main experte d'un insider que celle d'une menace externe russe par exemple ». La NSA serait-elle fragilisée par l'emploi de milliers de contractants, à l'instar de Snowden ou Harold Martin, tous deux issus de la société Booz Allen Hamilton?

Les conséquences de ces fuites sont multiples et sans précédent, notamment parce que moult plateformes à l'échelle mondiale ont été compromises, Windows bien sûr, mais aussi Linux, Solaris...

La fuite majeure a mis en évidence des exploits de Windows (ETERNALBLUE), et des outils dédiés à l'infiltration des banques, tel que JEEPFLA_MARKET, déjà utilisé en 2013 pour une attaque contre EastNets, la plus importante agence de services de gestion des transactions interbancaires (SWIFT) au Moyen-Orient. D'autres documents présents dans le package prouvent l'infiltration réussie de la NSA dans le réseau bancaire de cette région, aux Émirats arabes Unis (Abu Dhabi et Dubaï), Koweït, Yémen, Qatar, Syrie et territoires palestiniens.

Depuis ces révélations, un nombre croissant d'attaques utilisent ces outils de hacking. Le 15 août dernier l'éditeur Kaspersky indiquait avoir bloqué plus de cinq millions d'attaques basées sur des malwares impliquant les exploits en provenance de ShadowBrokers. Outre les infiltrations dont nous n'avons pas connaissance, des offensives aux répercussions inédites ont changé les règles du jeu, comme le 12 mai dernier avec l'attaque WannaCry qui exploitait une faille du protocole SMB de l'OS Windows. La prolifération du ransomware fut alors nettement ralentie par le biais d'un « kill switch »

SERVEURS DÉDIÉS **XEON®**

AVEC

ikoula
HÉBERGEUR CLOUD

Optez pour un serveur dédié dernière génération et bénéficiez d'un support technique expérimenté.

debian ubuntu CentOS Windows Server 2012



POUR LES LECTEURS DE
PROGRAMMEZ*

OFFRE SPÉCIALE -60 %

À PARTIR DE

11,99 €

~~29,99 €~~

CODE PROMO
XEPRO17

Assistance technique
en **24/7**

Interface **Extranet**
pour gérer vos prestations

KVM sur IP
pour garder l'accès

Analyse et surveillance
de vos serveurs

RAID Matériel
en option

Large choix d'**OS**
Linux et Windows

*Offre spéciale -60 % valable sur la première période de souscription avec un engagement de 1 ou 3 mois. Offre valable jusqu'au 31 décembre 2017 23h59 pour une seule personne physique ou morale, et non cumulable avec d'autres remises. Prix TTC 14,39 €. Par défaut les prix TTC affichés incluent la TVA française en vigueur.

CHOISSISEZ VOTRE XEON®

<https://express.ikoula.com/promoxeon-pro>



ikoula
HÉBERGEUR CLOUD

/ikoula

@ikoula

sales@ikoula.com

01 84 01 02 50

découvert par le jeune chercheur en sécurité britannique Marcus Hutchins, a.k.a MalwareTech, arrêté cet été par les agents du FBI à Las Vegas, à l'issue de la conférence Def Con. Il est actuellement détenu par les autorités américaines, et encourt quarante ans de prison. Un second kill switch enregistré par Matthieu Suiche bloquera lui plus d'un million d'attaques.

Selon Europol WannaCry constitue la plus importante offensive par ransomware de tous les temps, avec des centaines de milliers de victimes réparties dans plus de 150 pays. S'ensuit NotPetya en juin, déguisé lui aussi en ransomware, a priori plutôt destiné à la déstabilisation étatique ou économique qu'à recueillir des rançons en Bitcoins. Ce dernier a principalement ciblé l'Ukraine via une backdoor infectant M.E. Doc, le logiciel de comptabilité le plus populaire en Ukraine. La propagation de NotPetya en dehors des frontières de l'Ukraine aura impacté de nombreux industriels dans le monde, dont Saint-Gobain en France ou les laboratoires pharmaceutiques Merck aux États-Unis. Une vulnérabilité inquiétante sur le long terme puisque les exploits SMB permettent l'exécution de code arbitraire à distance sur Windows 7 (SMBv2.1), XP, Vista ainsi que Windows Server 2003 et 2008 (SMBv1).

En conclusion de sa présentation, Matthieu Suiche interpelle sur l'inefficacité des arrangements de Wassenaar, sensés réguler l'usage des armes numériques. Il pose la question de la responsabilité des acteurs impliqués, constructeurs, entreprises, gouvernements, contractants... Saluant cependant la rapidité de Microsoft pour réagir aux attaques et fournir les correctifs, mais quid des plateformes IoT, Solaris... ? Quid du débat sur les backdoors gouvernementales ou autres, puisque ces portes dérobées échappent à leurs auteurs ? On peut aussi déplorer l'absence d'un cadre légal pour protéger les chercheurs en sécurité. Le niveau des attaques en très forte augmentation et leurs degrés de sophistication inédits, représentent certes un risque crucial pour toutes les organisations, et plus critique encore pour les entreprises de taille moyenne qui ne sont pas préparées à ce type de bouleversement, faute de ressources pour y faire face. •

Pour aller plus loin, vous pouvez vous rendre sur le blog de Comae et télécharger la présentation « Black Hat 2017: The Shadow Brokers - Cyber Fear Game Changers » ainsi que le livre blanc : <https://blog.comae.io/the-shadow-brokers-cyber-fear-game-changers-d143706f560f>

BLACK HAT PUBLIE LES RÉSULTATS DU SONDAGE ANNUEL

« 2017 Black Hat Attendee Survey »

- Véronique Loquet, spécialiste open source & Infosec @vloquet

Ce rapport sur l'identification des cybercrises imminentes met en évidence les préoccupations de 580 répondants, professionnels de la cybersécurité aux États-Unis. Concernant le piratage d'infrastructures critiques vitales, comme dans les secteurs de l'énergie ou des télécoms, 60 % estiment qu'une cyberattaque réussie contre un système industriel (SCADA) se produira au cours des deux prochaines années. Seuls 26 % sont convaincus que le gouvernement américain et les dispositifs de défense sont suffisamment équipés pour y faire face de manière appropriée.

69 % pensent que nombre croissant d'attaques contre la souveraineté des Etats-Nations a rendu plus vulnérables les données des entreprises. L'attaque de Sony par la Corée du Nord en 2014 a marqué un tournant et les experts s'attendent à davantage d'offensives de ce type au cours de l'année à venir.

Comme en 2016, la menace la plus importante du classement est le ransomware, considérablement en hausse avec WannaCry et Petya. Le phishing, l'ingénierie sociale et les attaques ciblées arrivent au second rang.

La prise de conscience des dangers liés à l'internet des objets est arrivée tardivement avec Mirai, un malware qui a infecté des centaines de milliers d'objets connectés, pour former un botnet géant et lancer des attaques surpuissantes par déni de service (DDoS). Le rapport cite notamment l'attaque DDoS massive contre le fournisseur de DNS Dyn. Curieusement la menace IoT n'apparaît qu'au dixième rang de l'étude, toutefois à la question de l'évolution de leurs priorités, 34 % pensent qu'elle deviendra leur principale préoccupation dans un avenir proche. Une tendance qui devrait se confirmer d'autant en l'absence de réforme, puisque la mise sur le marché de millions d'appareils s'opère avec l'économie d'audits de sécurité et touche tous les secteurs, y compris les plus sensibles (médical, transport, énergie, domotique, robotique,

armement, agroalimentaire...). Les fake news et la fuite de données confidentielles sont aussi une source d'inquiétude en hausse, 82 % des experts en cybersécurité sont convaincus que les élections américaines ont été influencées par la diffusion sans précédent de fausses informations. Près de trois quarts des répondants estiment que rien ne peut empêcher la création de fausses nouvelles, mais 73 % souhaitent un effort concerté pour éduquer les lecteurs, 46 % évoquent le filtrage des médias sociaux et 29 % opteraient pour la création d'une liste noire des sites douteux afin de prévenir la désinformation.

Sur ce terrain on pourrait noter la non fiabilité des machines de vote qui restent des cibles de choix pour les attaquants, beaucoup d'entre elles ont des vulnérabilités qui peuvent être facilement exploitées et les effets vont au-delà de la perte de confiance dans le processus démocratique. Comme chaque année, les participants à Def Con ont découvert et exploité des failles inédites sur ces systèmes. Cette fois Def Con leur avait dédié un hacking village et les tests ont fonctionné sur une trentaine de machines, cinq marques utilisées dans tout le pays (Sequoia AVC Edge, ES&S iVotronic, Diebold TSX, Winvote, et Diebold Expresspoll 4000), toutes poutrées en 24 heures. Bien en amont de Def Con Jeanette Manfra, sous-secrétaire adjointe par intérim à la cybersécurité du ministère de la Sécurité intérieure, avait déclaré au Comité sénatorial du renseignement que l'agence disposait de preuves que des cyberattaquants avaient pris pour cible les systèmes électoraux de 21 États et que, dans certains cas, des données avaient été volées. Elle ajoutait cependant qu'aucun vote n'avait été modifié. •

Ressource :

<https://www.blackhat.com/docs/us-17/2017-Black-Hat-Attendee-Survey.pdf>

NE RATEZ AUCUN NUMÉRO Abonnez-vous !

PROGRAMMEZ!

le magazine des développeurs

Nos classiques

1 an	49€*
11 numéros	
2 ans	79€*
22 numéros	
Etudiant	39€*
1 an - 11 numéros	

* Tarifs France métropolitaine

Abonnement numérique

PDF	35€
1 an - 11 numéros	

Souscription uniquement sur
www.programmez.com

Option : accès aux archives 10€

Nos offres d'abonnements 2017

1 an	59€
------	-----

11 numéros + 1 vidéo ENI au choix :

- Arduino*
Apprenez à programmer votre microcontrôleur
- jQuery*
Maîtrisez les concepts de base



2 ans	89€
-------	-----

22 numéros + 1 vidéo ENI au choix :

- Arduino*
Apprenez à programmer votre microcontrôleur
- jQuery*
Maîtrisez les concepts de base

Offre limitée à la France métropolitaine

* Valeur de la vidéo : 34,99 €

Toutes nos offres sur www.programmez.com



Oui, je m'abonne

- Abonnement 1 an : 49 €**
- Abonnement 2 ans : 79 €**
- Abonnement 1 an Etudiant : 39 €**
Photocopie de la carte d'étudiant à joindre

- Abonnement 1 an : 59 €**
11 numéros + 1 vidéo ENI au choix :
- Abonnement 2 ans : 89 €**
22 numéros + 1 vidéo ENI au choix :

- Vidéo : Arduino
- Vidéo : jQuery

Mme M. Entreprise : _____ Fonction : _____

Prénom : _____ Nom : _____

Adresse : _____

Code postal : _____ Ville : _____

email indispensable pour l'envoi d'informations relatives à votre abonnement

E-mail : _____ @ _____

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

Je souhaite régler à réception de facture

* Tarifs France métropolitaine

Recrutement de développeurs et en informatique en général : un marché toujours tendu ?



François Tonic

Il ne se passe pas une semaine, ou presque, sans que l'on entende que le marché informatique est toujours sous tension, que l'on manque de développeurs, de compétences informatiques, etc. Nous évoquerons aussi la dernière enquête de stackoverflow.

Une enquête du Crédoc datant d'août 2017 évoque que 61,7 % des projets de recrutements sont difficiles pour les ingénieurs, cadres études, R&D, responsables informatiques. Même s'il faut être prudent avec les statistiques, cela montre une fois de plus les problèmes de recrutements. Bien entendu, ce problème ne concerne pas tous les profils ni toutes les régions.

Retour sur l'étude annuelle de Stackoverflow et notre analyse

Stackoverflow publie chaque année son rapport sur le marché du recrutement et les profils développeurs. L'étude met en avant 3 chiffres :

- 64 % des dévs français sont ouverts à de nouvelles aventures professionnelles. 8 % sont activement en recherche d'un emploi.
- 90 % des dévs déclarent être au moins partiellement autodidactes.
- 87 % des dévs sont employés au moins à temps partiel.

Le terme autodidacte nous gêne un peu. Parler d'autoformation, de veille technologique constante serait plus juste. Car, quand on regarde les % sur les diplômes / formations, on constate que 74,6 % des répondants disent avoir un diplôme (mini bac+2) et sans le dire, une majorité en informatique. Si des profils atypiques et les autodidactes font carrière, ils sont encore peu présents, car en France, l'aspect diplôme est toujours un critère incontournable. Mais cela commence à changer, un peu.

L'Île de France reste la locomotive : 38,9 % des développeurs répondant à l'enquête vivent dans cette région. Seule région qui soit très dynamique : Auvergne – Rhône – Alpes avec 14,8 %. Pour nous, cette situation est un peu tronquée, car les pôles technologiques comme Lyon, Grenoble, Lille, Nantes, Bordeaux, Toulouse, Clermont / Saint Étienne, Montpellier, Aix-en-Provence, Marseille –

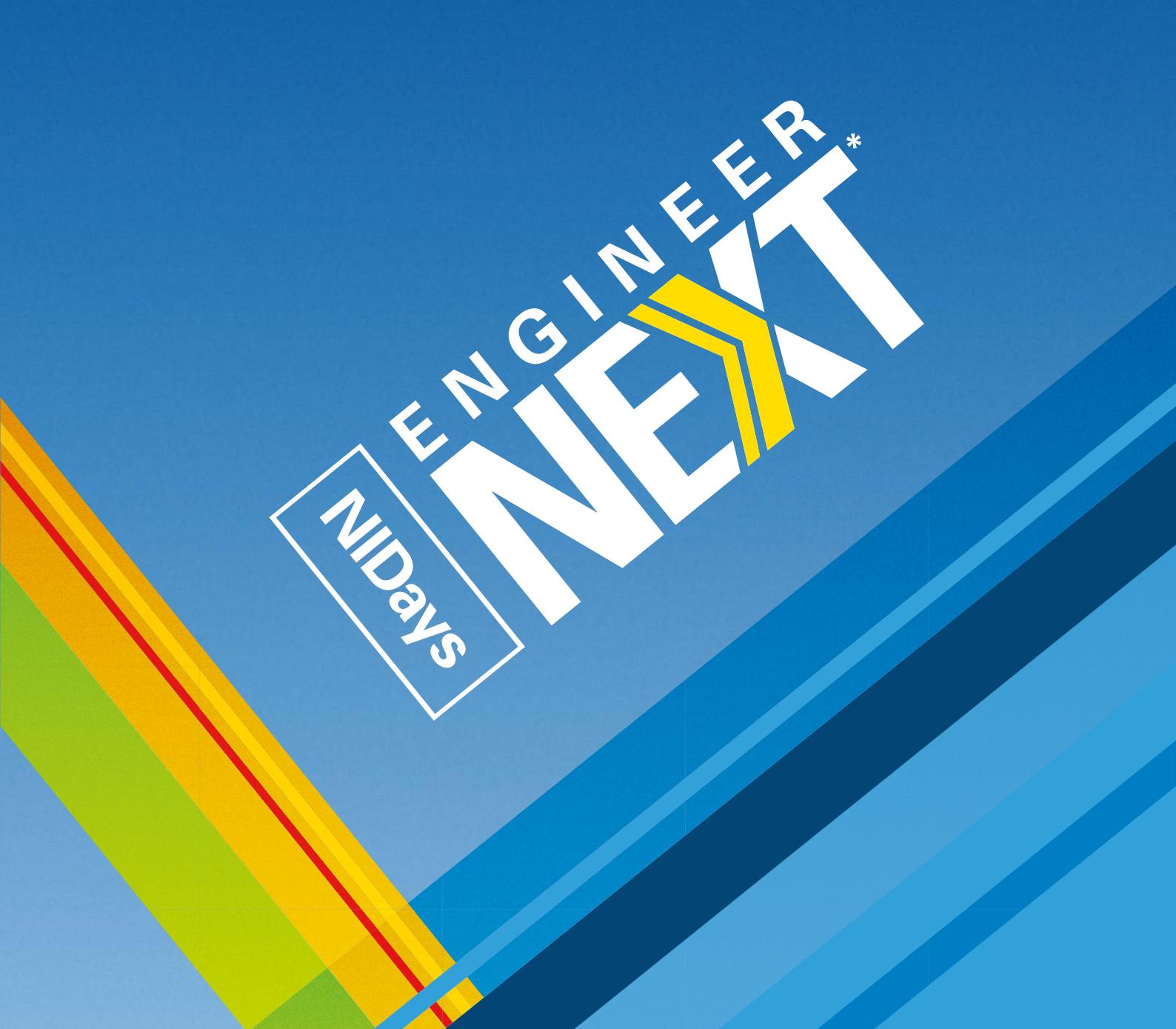
LE CHÔMAGE DES INFORMATICIENS EXISTE

Sujet encore trop tabou en France, le chômage des informaticiens (au sens large) existe et il ne faut surtout pas le négliger. Situation paradoxale vu le niveau de recrutements dans le secteur des nouvelles technologies. Les statistiques de cet été affichaient 48 000 chômeurs dans les métiers informatiques et télécoms, toutes catégories confondues. Pour 2017, les 50 000 recrutements pourraient être atteints (estimation Apec), mais les évolutions technologiques modifient les profils et les compétences recherchées.

Toulon sont très dynamiques en termes de marchés et d'emplois. De nombreuses SSII (indépendants ou agences des gros cabinets de services), et éditeurs sont en province. Ce sont souvent de petites structures 10-20 personnes. Il y a clairement des opportunités à saisir et des possibilités de vivre de nouvelles expériences loin de Paris et de sa région... Dans la topologie des développeurs, nous avons 72 % de développeurs professionnels et 19 % d'étudiants. L'indépendant / free-lance est peu représenté dans l'étude : 9,6 %. Ce qui est peu. En France, le free-lance est assez répandu et les dévs qui montent leur société de développement ne sont pas rares. Les grosses SSII et les intégrateurs drainent toujours la majorité des jeunes développeurs et s'imposent sur le marché. Des pures players très spécialisés peuvent s'imposer dans le marché. Sans surprise pour nous, les développeurs web sont surreprésentés : 68,9 %. Même si un développeur peut avoir plusieurs profils :

mobile, desktop, serveur, web, CMS, etc. Element surprenant, seuls 20,8 % se disent développeur mobile... Et rien sur la partie cloud computing. Il est vrai que le développeur développeur web recouvre aujourd'hui un spectre très technique. Il est intéressant de voir que la majorité des répondants à l'étude de stackoverflow est loin du domaine informatique classique : finance – assurance, secteur public qui draine un % important des postes IT en France, via les intégrateurs le plus souvent. Étonnante aussi la quasi inexistence du jeu vidéo : 1,1 %. Mais il est vrai que les termes logiciels et services web (41 % des réponses) sont très généralistes et peuvent recouvrir les secteurs cités plus haut. La taille des sociétés ne surprendra pas : 41 % des répondants travaillent dans des sociétés ayant au minimum 500 personnes. Les TPE et PME ne sont pas à négliger, car 23,1 % y travaillent. Et c'est un vrai vivier pour le recrutement.

La partie salaire fournit quelques indicateurs intéressants. À noter que les années d'expérience dépassent les 5 ans pour 74 % des répondants ce qui a forcément une influence sur les salaires moyens de l'étude, sans oublier le niveau des diplômes (bac+5 pour 54,7 %). Le développeur mobile, desktop ou web oscille entre 41 et 43 000 bruts annuels. Nous sommes dans les moyennes basses de ce que nous voyons dans d'autres études, mais ce sont des moyennes. Vous aurez aussi des disparités, parfois très fortes entre Paris – Province. Par contre, les salaires moyens pour les profils Data Scientist et DBA sont bas. Soit ce sont des débutants dans ces postes ou alors les répondants ont mal cadré leurs réponses. Autre surprise : le spécialiste DevOps à 49 000 . Pour nous, ce profil ne correspond pas à grand chose, à chercher côté des spécialistes Scrum ? Presque 36 % des répondants se disent ni sous-payés ni surpayés, mais 57,8 % se disent sous-payés (un peu ou largement).



7 NOVEMBRE 2017 | PALAIS DES CONGRÈS DE PARIS

NIDays, c'est l'occasion de rejoindre les acteurs de l'innovation d'un grand nombre d'industries et de découvrir comment les progrès dans les technologies des transports, l'automatisation des tests, la communication 5G et autres encore se conjuguent pour créer un monde plus intelligent et connecté à travers des systèmes définis par logiciel.

INSCRIVEZ-VOUS DÈS MAINTENANT SUR NI.COM/NIDAYS



Effectivement, les rémunérations moyennes bougent finalement assez peu malgré un marché de recrutement tendu. L'arrivée des profils débutants chaque année contribue à cette situation. Cependant attention : les profils expérimentés ou rares connaissent des valorisations de salaires et de conditions de travail. Et les bons profils de développeurs ne sont pas aussi fréquents que cela. Soyez attentif(ve) sur les grilles salariales et les conditions de travail proposées. N'hésitez pas à aller voir les équipes et à dire ce que vous souhaitez. Bien entendu, vos demandes doivent être现实的.

Si vous postulez dans une entreprise :

- regardez le secteur d'activité de la société et si celui-ci vous intéresse ;
- préparez un minimum votre rendez-vous, qu'il soit physique ou en visioconférence : l'improvisation c'est bien mais parfois cela se voit un peu trop ;
- soyez ponctuel (le). Si vous ne pouvez pas venir ou que vous savez que vous aurez du retard, prévenez le recruteur. C'est la moindre des choses. Ce n'est pas parce que vous avez le choix des postes que vous devez être impoli ;
- la maîtrise de l'anglais, au moins en lecture est le b a ba du développeur. Si on vous pose la question : là encore, soyez honnête sur votre niveau réel, surtout si le poste nécessite des interventions ou réunions en Anglais ;

• Vous avez des prétentions et demandes : c'est normal et légitime d'en parler mais ces demandes doivent être现实的.

Sur les réseaux de recrutement, l'étude dégagé les canaux les plus importants :

- un ami, un collègue, une connaissance qui parle de la société où il travaille ou qu'il connaît ;
- l'entreprise qui contacte directement le développeur : soit sur le conseil d'une connaissance commune ou via des réseaux sociaux / un événement technique.

Ces 2 pèsent presque 43 % ! Bref, ne négligez pas vos réseaux.

La partie télétravail est intéressante, car cela pose une véritable question sur le métier de développeur et son évolution. Selon les réponses fournies, 23 % des répondants disent avoir au moins une journée par semaine en télétravail, dont 9 % à plein temps (free-lance ?). 24 % évoquent quelques jours par semaine et 49 % jamais. Le télétravail est une question sensible. Nous connaissons des éditeurs et intégrateurs qui refusent le télétravail. Quand le développeur est en régie (donc chez le client durant x mois), difficile de faire du télétravail sauf accord spécifique. Le temps de transport est un élément important à considérer par le candidat et par l'entreprise. Un accès facile aux transports en commun, le stationnement pour la voiture ou la moto sont des points à ne pas négliger, tout comme bien préciser les frais remboursés ou non.

Quelles motivations pour chercher un autre poste ?

Si vous n'êtes pas satisfait de votre entreprise actuelle, pourquoi ne pas changer ? Et dans ce cas, quelles sont les motivations ? Le rapport de stackoverflow confirme ce que nous voyons sur le terrain :

- les langages, les frameworks, les outils : ce sont des éléments importants pour garder ou attirer. Ce sont des défis intéressants à relever ou tout simplement des technologies sur lesquelles vous avez envie de travailler ;
- opportunité liée à un poste ou une entreprise, un secteur d'activité ;
- les méthodes de développement utilisées : cela peut être un facteur d'intérêt, mais attention à ne pas se focaliser trop dessus ;
- les bureaux, l'ambiance : eh oui ce sont des + ;
- rémunérations et prestations proposées : là encore c'est une tendance lourde depuis plusieurs années. Le salaire n'est pas l'unique critère de choix ;
- Le temps de transport : facteur important à ne jamais négliger ;
- Le matériel est aussi un élément à ne jamais négliger : un développeur a besoin d'un matériel à jour et performant ;
- Le temps libre pour assister à des conférences, collaborer aux communautés ou à des projets open source : c'est très important surtout si le développeur est actif dans le monde open source. L'entreprise y gagnera en notoriété.

À noter un chiffre qui change peu : 91 % des répondants sont des hommes.

A vous de jouer !

Tous les numéros de
PROGRAMMEZ!
 le magazine des développeurs
 sur une clé USB (depuis le n° 100)



34,99 €*

Clé USB. Photo non contractuelle. Testé sur Linux, OS X, Windows. Les magazines sont au format PDF.

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

Clé USB PROGRAMMEZ 34,99 €

Commande à envoyer à : Programmez!
 57, rue de Gisors- 95300 Pontoise

M. Mme

Entreprise : _____

Fonction : _____

Prénom : _____

Nom : _____

Adresse : _____

Code postal : _____ Ville : _____

E-mail : _____

_____ @ _____

Règlement par chèque à l'ordre de Programmez ! / Commandez directement sur www.programmez.com



Rendre les développeurs heureux en entreprise, c'est possible

Partie 2



Cyril Lakech
Cyril Lakech est développeur et leader technique chez AXA France où il a notamment pour mission d'animer la communauté des développeurs.



Romain Linsolas
Romain Linsolas travaille chez Société Générale comme leader technique d'une équipe élaborant les outils de développement d'applications Web pour toute la banque d'investissement.

L'ouverture technologique

Réussir à gérer les technologies utilisées dans l'entreprise d'une manière qui soit adaptée aux développeurs n'est pas chose facile. Ces derniers sont réputés comme étant des collectionneurs de langages, de frameworks et de plateformes. Ils se lassent vite et ont souvent envie de tester les dernières évolutions à la mode.

Afin d'éviter un changement trop fréquent de technologies et pour limiter autant que faire se peut les dérives, des murs ont été érigés par nombre d'entreprises qui gravent dans le marbre ce qui est autorisé sur les projets et ce qui ne l'est pas. Les développeurs se trouvent alors en grande difficulté dès que se présente le besoin d'utiliser de nouvelles solutions ; ce problème de communication entraîne généralement des frustrations, voire dans certains cas extrêmes, des démissions. De telles limitations ne sont pas souhaitables tant l'écosystème du développeur évolue à grande vitesse, car le risque de se retrouver rapidement avec des choix technologiques dépassés est important. Mais comment allier flexibilité et maîtrise des technologies utilisées ?

Il existe un concept de radar technologique, rendu populaire par la société Thoughtworks qui vient classer les technologies en quatre catégories que sont les langages de programmation, les outils, les plateformes et enfin les pratiques. Chacune de ces catégories est ensuite divisée en quatre anneaux qui représentent un certain niveau de fiabilité et d'intérêt. Cela commence au niveau de plus haute fiabilité, « Adopt », dénotant les techniques, outils, pratiques et langages qui sont estimés comme pouvant être viables pour un déploiement en

Reconnaissons-le, le développeur jouit d'une situation très enviable : il est courtisé de toutes parts. Le nombre d'offres d'emploi dans le monde du développement peut faire rougir de nombreux autres secteurs d'activité. De l'autre côté du miroir – des entreprises donc – il devient ainsi compliqué de réussir à recruter les bonnes personnes, puis de les conserver au sein de leurs effectifs. Voici quelques pistes pour les entreprises qui souhaitent recruter et garder les meilleurs développeurs.



La partie langages du radar de Thoughtworks (capture du site de ©Thoughtworks)

entreprise. Viennent ensuite les niveaux « Trail », « Assess » et enfin « Hold ». [1] Le concept derrière ce genre de radar peut être appliqué en fonction de la vision d'une entreprise et on peut alors publier la liste des technologies sous cette forme. Il représente également un bon moyen de communiquer clairement sur la vision et l'usage des différentes technologies. Bien entendu, ce radar a pour vocation d'évoluer. Lorsqu'un développeur souhaite utiliser une nouvelle technologie, il prend contact avec un représentant de la communauté dédiée, par exemple la communauté Web si l'on parle du dernier framework front-end à la mode. Ensemble, ils prépareront une présentation de cette technologie à destination d'un collègue de personnes ayant des profils divers et complémentaires au sein de l'IT de l'entreprise : architectes, opérateurs, responsables de la sécurité, développeurs ou experts dans un domaine en lien avec le sujet en question.

Le rôle de ce collectif est d'évaluer ce qui lui est présenté, en se posant les bonnes questions. Quelles sont ses forces et ses faiblesses ? Est-ce que cela présente des risques ou des opportunités ? Quelle est l'expérience de la commu-

nauté avec cette technologie ? Quels sont les coûts induits ? Et ainsi de suite.

Si la technologie passe cette étape, elle peut être validée pour une seconde phase qui va la mettre en conditions réelles, sur un projet pilote par exemple. Une fois ce dernier terminé – ou suffisamment avancé, un retour d'expérience est partagé avec ce même collègue afin de dresser un bilan et de définir la prochaine évolution de cette technologie au sein du radar : soit elle est mise de côté car inadaptée ou présentant trop de risques, soit on la considère comme étant assez intéressante pour qu'elle puisse être utilisée sur d'autres projets.

La encore, cette validation n'est pas figée dans le marbre. Elle sera remise en question régulièrement, soit pour être déployée encore plus largement au sein de l'entreprise, soit pour être petit à petit dépréciée.

Cette méthode du radar technologique permet non seulement de communiquer largement et clairement sur les technologies utilisées dans l'entreprise, mais elle offre également la possibilité aux développeurs d'avoir une emprise sur celui-ci en y contribuant directement.

Il existe d'autres principes pour s'ouvrir aux

évolutions des technologies. Certains projets mettent en place des *feature teams*, mécanisme popularisé par la société Spotify. Elles sont définies par leur petite taille — généralement autour de six à huit personnes — et par le fait qu'elles sont composées de talents complémentaires (développeur Web, développeur *back-end*, testeur, analyste, etc.).

Dans une telle configuration, on pourra laisser libre chaque *feature team* de gérer ses propres langages, plateformes, outils et pratiques. Il faudra néanmoins leur imposer un certain niveau de qualité à atteindre, des APIs d'échanges à respecter, ainsi que des règles de fonctionnement bien précises: une parfaite intégration dans le reste du S.I., une documentation exhaustive et claire, un certain niveau de qualité et de support à respecter, etc.

Ce principe de fonctionnement est déjà en place chez certains géants du Web. Les équipes y gagnent en autonomie et en flexibilité mais il faut s'assurer du bon respect des règles de vie en communauté pour que la cohérence du SI reste gérable dans le temps sans devenir un point noir qui phagocytéra les futures évolutions fonctionnelles.

Créer des communautés

Une fois que l'on a décloisonné l'entreprise, il reste encore un besoin primordial à combler chez le développeur : pouvoir dialoguer avec ses pairs pour trouver de l'aide, aider les autres et contribuer aux besoins communs de sa communauté. Un peu à l'image d'une famille sur laquelle on peut compter, les développeurs ont généralement cette envie de se regrouper avec d'autres développeurs ayant les mêmes compétences ou le même rôle qu'eux.

Il arrive que dans son équipe, un développeur soit le seul ayant un profil de développeur Web par exemple. Vers qui pourrait-il se tourner s'il a besoin d'aide et que ses collègues ne sont pas en mesure de l'aider ? Comment peut-on confronter ses idées à d'autres experts du même domaine ? StackOverflow est-il la seule voie ? Créer des communautés de pratiques dans l'entreprise améliore la solidarité entre équipes et apporte de la cohérence dans le système d'information en augmentant l'intelligence collective dans le domaine de chaque communauté et en incitant à la réutilisation de composants communs. En contrepartie, cela implique une certaine organisation à mettre en place.

Les exemples de communautés ne manquent pas, et il est préférable de laisser les développeurs créer les communautés dont ils ont besoin. Ainsi, on pourra en créer autour des

technologies comme Java, .NET, le développement *front-end*, le développement mobile, NodeJS ou encore autour des pratiques comme *Crafts(wo)manship*, API ou enfin autour des rôles tels que leaders techniques, Scrum Masters, etc. Toute initiative de création de communauté doit être encouragée, la seule chose qui importe étant qu'elle soit utile aux développeurs et qu'elle soit vivante !

La mise en place de ces communautés nécessite un cadre, une vision et des valeurs pour qu'elles s'inscrivent dans le plan d'évolution de l'entreprise afin d'y contribuer pleinement. Il faut en premier lieu identifier des volontaires et des leaders pour démarrer et animer les communautés. Pour faire vivre ces dernières, ils vont avoir besoin de temps et d'espace pour échanger, se retrouver, collaborer.

Dans ce but, les participants pourront disposer de quelques heures par mois pour suivre une communauté, voire davantage afin d'y contribuer activement. Il serait vain de leur demander d'y contribuer sur leur temps libre, uniquement pendant la pause déjeuner, le soir ou les weekends. Des « BOF » (« *Bird of a Feather* »), des points de rencontres réguliers entre membres d'une même communauté pourront être organisés pour partager des présentations, débattre de sujets stratégiques, prendre des décisions, définir des bonnes pratiques ou toute autre tâche nécessaire à la vie du groupe. Dans un but d'efficacité, on pourra leur mettre à disposition des outils collaboratifs tels que Github, Confluence, Slack ou encore JIRA pour aider les communautés à construire, à partager et à s'organiser.

Un exemple concret : actuellement, la communauté NodeJS d'AXA réalise un kit de démarrage qui aidera les équipes en charge de créer de nouveaux projets avec ce type de technologie. Ce kit consiste en une application blanche prête à être intégrée dans le système d'information de l'entreprise, comprenant les bonnes pratiques de développement, pré-configurée pour être déployée dans le Cloud d'AXA et comprenant des composants transverses de *log*, de diagnostic, de surveillance, et ainsi de suite. À terme, cela va faciliter le déploiement de NodeJS et permettre d'augmenter le nombre de projets qui utilisent cette plateforme.

Contribuer à l'open-source

L'open-source est un fondamental dans le développement logiciel. Les développeurs y sont particulièrement attachés, leur travail a toujours été grandement facilité par l'incroyable richesse des solutions open-source, certains d'entre eux

y participant même très activement. Si les entreprises basent nombre de leurs projets sur des solutions issues de cette communauté, le voyage inverse, qui consiste pour ces entreprises à y contribuer en retour, est souvent moins fréquent. Pourquoi ? Les principaux freins à cette contribution sont d'abord liés à des considérations de sécurité : il y a toujours une certaine friabilité à vouloir sortir du code, et il devient parfois nécessaire de mettre en place des mécanismes pour s'assurer qu'il n'y a aucune fuite de données critiques.

Mais ce sont également des contraintes légales qui sont parfois opposées à ces contributions : à qui revient la propriété intellectuelle de la contribution ? Comment gérer le cas d'une équipe composée en partie de prestataires appartenant donc à une autre société ? Malgré ces obstacles, la participation d'une entreprise à l'open-source est très bénéfique. C'est avant tout un formidable vecteur d'apprentissage. En étudiant le code développé par d'autres personnes, on apprend de nouvelles approches, de nouvelles techniques. En montrant le code que l'on produit à d'autres, on apprend également par les retours, les corrections et les commentaires qui sont faits à son endroit.

C'est aussi un bon moyen de faire évoluer ses produits grâce aux contributions externes. Si cet aspect revêt un caractère éminemment positif, cela peut s'avérer chronophage : il faut prendre du temps pour analyser les *Pull Requests* (les contributions faites par des personnes extérieures à l'équipe), mais également pour en assurer le support. Fort heureusement, il existe de nombreux mécanismes automatisés pour faire un premier tri parmi ces contributions, comme ces robots dédiés à scanner le code afin de s'assurer du bon respect des conventions de codage, de l'écriture des tests, et ainsi de suite.

C'est enfin une piste de recrutement à ne pas négliger. Si une entreprise participe activement au développement d'une solution open-source, elle va nécessairement attirer l'attention des personnes ayant recours à cette solution, et potentiellement les intéresser à rejoindre la compagnie.

À l'inverse, une société peut détecter des contributeurs importants sur les solutions open-source stratégiques pour elle et tenter de les recruter. Bien entendu, dans pareilles circonstances, le candidat en question sera très vigilant sur la façon dont l'entreprise contribue à l'open-source. Vouloir recruter un contributeur à l'open-source puis lui refuser le droit de continuer ses activités une fois au sein de l'entreprise n'aurait

en effet aucun sens, et le recrutement évidemment voué à l'échec !

S'il est compliqué dans un premier temps de contribuer à l'open-source, une alternative devrait toutefois être tentée, celle de l'*inner-source*. Il s'agit d'appliquer les pratiques et la notion du partage de l'open-source, mais au sein de l'entreprise, le code restant alors cloisonné au sein du système d'information de celle-ci. Différents outils peuvent être utilisés dans ce but : Github Entreprise, des dépôts privés sur Github.com, GitLab, etc. L'important est de partager le code produit par son équipe au sein de toute l'entreprise. Même si le public touché est bien moindre qu'au travers de l'open-source, les bénéfices d'apprentissage cités précédemment restent valides. C'est également un excellent moyen d'éviter la redondance de développement, puisqu'on évitera ainsi plus facilement de créer une fonctionnalité qui aura pu être développée précédemment par une autre équipe.

Participer à des événements

La notion de partage est extrêmement forte dans le monde du développement informatique. Il suffit de voir le nombre de conférences qui existent pour s'en convaincre : rien que sur la région parisienne, ce sont des dizaines de conférences, de rencontres, de réunions ouvertes qui ont lieu chaque semaine. Les entreprises ont leur rôle à jouer à ce niveau-là, aussi bien pour leur profit que pour celui de leurs collaborateurs.

Le *Brown Bag Lunch* – ou plus simplement *BBL* – représente le moyen le plus accessible d'apporter la connaissance au sein d'une société. Le concept est très simple et peut se résumer ainsi : on invite un(e) expert(e) le temps du déjeuner pour venir réaliser une présentation technique dans son domaine. Le terme de *brown bag* fait à ce propos référence au fameux sac en papier marron que l'on utilise pour transporter son déjeuner sur son lieu de travail. Pour l'hôte – et donc l'entreprise, les avantages sont multiples : cela représente une veille technologique très accessible, très variée tant les sujets proposés lors des BBL sont vastes et cela permet à ses collaborateurs de découvrir de nouveaux langages, outils ou méthodes de développement. Pour l'expert(e), aussi appelé(e) *bagger*, c'est l'opportunité de se faire connaître, de rôder une présentation – utile pour la répéter en amont d'une conférence – ou encore d'exercer ses talents d'orateur, ce qui est plus facile à faire en petit comité d'une dizaine de personnes que

face à un auditoire de trois cents participants. Si vous êtes intéressés par ce type d'événements, il suffit de vous rendre sur le site recensant toutes les personnes disposées à présenter des sujets sur ce format de BBL, <http://www.brownbaglunch.fr>. Ce ne sont pas moins de 250 femmes et hommes sur tout le territoire français qui s'y sont déjà inscrits !

Un autre moyen pour les entreprises d'avoir une part active au sein des communautés est l'hébergement de meetups. Un meetup, qui doit son nom au site Internet éponyme, est le rassemblement récurrent de passionnés autour d'un sujet. Si les meetups eux-mêmes sont gérés par des communautés dédiées, ce sont les entreprises qui les hébergent. Petites startups ou grosses entreprises, c'est un moyen simple de s'offrir de la visibilité auprès d'un ensemble de personnes ayant un attrait commun. Sur la place parisienne, il y a plus de cent meetups dédiés à la technologie qui rassemblent au moins mille inscrits chacun. De quoi largement trouver son bonheur !

Les hackathons sont très en vogue auprès des entreprises. Chaque année des dizaines d'événements de ce type sont mis sur pied à destination des développeurs, des entrepreneurs et des innovateurs. Un hackathon est un concours de développement collaboratif qui met en concurrence une dizaine d'équipes en moyenne pendant un, deux ou trois jours. Les entreprises qui mettent en place ce type d'événements donnent généralement un thème ou certaines contraintes aux candidats, par exemple le fait d'utiliser les *API* publiques de l'entreprise. Au terme du hackathon, les équipes présentent leurs réalisations à un jury, qui finit par élire les projets gagnants.

Si les hackathons ont clairement un intérêt pour les entreprises, il est très important que ces dernières offrent une véritable compensation aux participants. Considérer les hackathons comme de la main d'œuvre pratiquement gratuite risque de créer l'effet inverse de celui escompté, et ainsi de mettre l'entreprise dans une situation délicate. Pour éviter ce fameux effet *bad buzz* il est préférable de prévoir des récompenses de valeur pour les participants, l'idéal serait même d'être en mesure d'incuber une poignée de projets pour leur permettre de concrétiser les projets qu'ils auront ébauchés lors de l'événement.

Un autre type d'événement qui intéresse beaucoup les développeurs est la conférence interne. L'entreprise organise un événement – généralement sur une journée – essentiellement à destination de ses collaborateurs. C'est

l'occasion d'intégrer les membres nouvellement recrutés en leur imprégnant les valeurs de l'entreprise, de faire découvrir de nouvelles technologies, de lancer de nouvelles carrières d'orateurs... Bref, beaucoup de bonnes choses, mais qui représentent cependant une certaine complexité d'organisation et un prix non négligeables. S'il n'est pas toujours facile de monter sa propre conférence interne, une dernière approche bien plus accessible consiste à sponsoriser des événements, des conférences. Le métier du développement informatique jouit en effet d'une chance rare qui est le grand nombre de conférences sur de nombreux thèmes. Devenir le sponsor d'une conférence offre de nombreux atouts aux entreprises. Il s'agit en premier lieu de s'offrir une bonne visibilité voire une amélioration de leur image de marque auprès des développeurs, facilitant ainsi leur recrutement. Cela permet également d'envoyer de nombreux collaborateurs à ces conférences, ce qu'ils apprécient très souvent. [4]

CONCLUSION

Nous avons présenté de nombreuses pistes pour améliorer le quotidien des développeurs, et ainsi les rendre plus heureux, et plus productifs. Si toutes ces pistes ne sont pas applicables chez vous immédiatement, que ce soit par manque de temps, d'argent ou de volonté, il faut en retenir trois des aspects les plus importants. Tout d'abord, se rendre compte que de petites actions peuvent avoir de grandes conséquences. Offrir un environnement stimulant et productif – avec un matériel adapté au développement par exemple – permettra aux développeurs de se sentir dans de bonnes conditions pour réaliser leur travail.

Ensuite, il est indispensable de considérer le développeur en tant que tel. Il est loin le temps où ce métier était un tremplin vers le métier de manager ou chef de projet. D'autres pistes existent, qui permettent aux personnes qui le souhaitent de rester proches de la technique, tout en bénéficiant de la considération de leur hiérarchie. Enfin, il est fortement conseillé d'offrir aux développeurs un milieu proche de celui qu'il peut connaître à l'extérieur de l'entreprise : participation à l'open-source, communautés regroupant des experts dans les mêmes domaines que lui.

La mise en place de ces différents éléments ne résoudra sans doute pas tous les problèmes, mais cela contribuera énormément au bien-être des développeurs, qui se sentiront bien plus attachés à leur entreprise.

Deep learning, un modèle miracle ?

• Zied Abidi
data scientist
IBM France

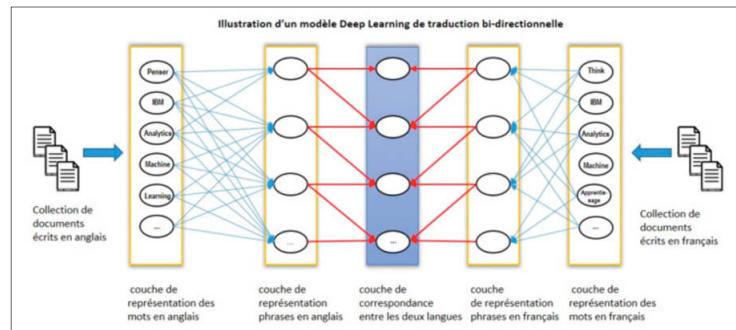
Comprendre et synthétiser le langage naturel, reconnaître des visages... Incontestablement, l'Intelligence Artificielle (IA) est de nouveau tendance. Après des débuts marqués par l'imagination des futurologues et des écrivains de science-fiction, l'IA semble désormais à portée de main et les performances et exploits des systèmes cognitifs IBM Watson ou du programme intelligent AlphaGo de DeepMind sont clairement les prémisses de son succès.

Souvent associé à ces exploits, le **Deep Learning** (apprentissage profond) ne finit pas de susciter l'enthousiasme des data scientists et promet de révolutionner la pratique du **Machine Learning** (apprentissage automatique ou statistique) et de la **Data Science** (science des données). Avec l'augmentation de la puissance des ordinateurs et la profusion des données disponibles, le Machine Learning devient un composant important des projets Big Data et le Deep Learning se développe rapidement. Dans ce propos, nous allons tenter d'expliquer intuitivement le Machine Learning et le Deep Learning.

Du Machine Learning au Deep Learning

Le Machine Learning (ML) est une technique qui permet de rationaliser la prise de décision dans un environnement incertain, un environnement où il est très difficile d'expliquer les règles formelles expliquant les phénomènes d'intérêt. Dans une approche ML, l'acquisition de la connaissance se fait par la découverte des règles implicites dans les données brutes en suivant un processus itératif. Prenons un exemple : Il est impossible de définir des règles qui permettent de déterminer de façon certaine qu'un e-mail est un spam ou pas. Dans ce cas, le Machine Learning est souvent utilisé pour apprendre ces règles à partir des données. À chaque itération d'une modélisation de type Machine Learning, les data scientists procèdent à la préparation et au choix des variables prédictives, c'est-à-dire des données pertinentes pour créer/améliorer un modèle statistique. Cette activité humaine consiste en l'**identification de la meilleure représentation** des variables d'entrée : le data scientist choisit le formatage (ex : entier ou à virgule flottante), la bonne granularité, le niveau de discréétisation des variables continues, etc. Le choix de la meilleure représentation des données est déterminant pour la qualité de la modélisation (« Garbage in, garbage out »), et implique des **décisions subjectives** qui font appel à l'expérience du data scientist, à son flair et parfois sa bonne fortune !

Le Deep Learning (DL) est un sous-ensemble du Machine Learning. Il s'appuie sur les réseaux de neurones artificiels. Le Deep Learning adresse



la problématique d'identification de la meilleure représentation des données en utilisant le Machine Learning non seulement pour trouver le meilleur modèle qui associe les données d'entrée aux données de sortie mais aussi pour **découvrir automatiquement la meilleure représentation** des données d'entrée. Un modèle DL stocke l'intelligence extraite sur plusieurs couches de représentation (neurones artificiels). Chaque couche de représentation est exprimée en fonction d'une autre couche plus primitive (voir illustration ci-dessous).

Deep Learning : un modèle miracle ?

Le mécanisme d'empilement des couches décrit plus haut est le secret de la puissance du Deep Learning mais ce même mécanisme rend les modèles DL difficiles à entraîner. La multiplication des couches, c'est-à-dire des connexions neuronales, et, avec elles, du nombre de paramètres à calibrer, nécessite une puissance de calcul considérable et des techniques très sophistiquées notamment de régularisation (problème d'over-fitting).

A ce jour, l'utilisation du Deep Learning reste limitée à des problématiques manipulant des données à topologie complexe ; ce qui est souvent le cas des données non structurées (structure hiérarchique, dimensionnalité élevée, etc.). Dans la pratique, le Deep Learning excelle dans les applications suivantes :

- La vision assistée par ordinateur (détection d'objets, génération d'images, etc.) ;
- La reconnaissance automatique de la parole ;
- Le traitement automatique de la langue (traduction, questions/réponses, etc.).

Deep Learning : quel outillage ?

Les outils et les plateformes de construction de modèles Deep Learning restent largement orientés autour du développement. Plusieurs librairies **Open Source** fournissent les « briques de base » (architectures types, procédures d'optimisation, moteurs d'exécution sur GPU, etc.) pour la construction d'applications DL.

Voici quelques-unes des technologies Open Source pour créer des modèles Deep Learning :

- TensorFlow pour les programmeurs Python ;
- Caffe pour les développeurs C++ et Python ;
- Theano pour les programmeurs Python ;
- Deeplearning4j pour les développeurs Java/Spark (Scala et Python) ;
- Torch pour les programmeurs Lua.

En conclusion

Le Deep Learning permet de créer des modèles de Machine Learning plus performants, des systèmes capables de s'adapter automatiquement à la complexité structurelle des données manipulées. Toutefois, ces applications restent limitées à cause de la difficulté de mise en œuvre et des environnements de développement peu conviviaux.

Dans les années à venir, l'évolution des technologies et l'augmentation de la puissance de calcul accentueront le phénomène et permettront d'entraîner plus efficacement encore les réseaux de neurones profonds. Mais pour l'heure, les data scientists continuent d'utiliser massivement les modèles de Machine Learning traditionnels, plus aboutis et faciles à entraîner. Le miracle du Deep Learning reste encore à venir !

ENABLING
DIGITAL
EVERYWHERE
opensourcesummit.paris
#OSSPARIS17

1ER ÉVÉNEMENT EUROPÉEN
LIBRE & OPEN SOURCE



PARIS OPEN SOURCE SUMMIT

6 & 7
DÉCEMBRE
2017

DOCK PULLMAN

Email : cjardon@weyou-group.com – Tel : 01 41 18 60 52

SPONSOR DIAMOND



SPONSORS PLATINUM



SPONSORS GOLD



SPONSORS SILVER



ORGANISÉ PAR



EN PARTENARIAT AVEC



APM : un nom bizarre pour des outils utiles

Un outil APM est un outil qui permet de surveiller et de montrer les applications en production. Par définition, l'APM permet de scruter les performances de son app. Tout développeur sait de quoi on parle, non ?

Le développeur surveille les performances de son application, corrige les lenteurs et livre un code ayant des performances optimales. Il teste en conditions réelles son application et les performances globales.

En réalité, ce travail n'est pas fait ou partiellement : faute de temps, délai trop court, pas d'optimisation, etc. Et une application peut se dégrader avec le temps : nombre d'utilisateurs, données volumineuses qui plombent les accès, terminaux non adaptés, fonctionnement sur des systèmes trop récents qui peuvent nuire aux performances des anciennes applications, etc. Les performances d'une application sont conditionnées par énormément de facteurs dépendants du développeur ou non. Certains problèmes de performances se voient uniquement en production et quand l'application fonctionne pleinement.

Toutes ces métriques vont donc vous servir à identifier les problèmes et trouver des solutions si cela vous concerne. Car la performance d'une application ne dépend pas uniquement de vos codes. Vous voilà rassuré !

“L'APM n'est pas un sujet nouveau. Depuis une dizaine d'années, il s'agit de montrer l'état de

santé des applications, de mesurer le niveau de services, de détecter un clic souris dans l'application, c'est savoir s'il a une réponse, mesurer ce qui se passe, le ressenti de l'utilisateur. La 2e chose est quand le problème survient, il faut trouver où cela coince, ce qui se passe dans x couches, où sont les lenteurs” Indique Gilles Portier (Dynatrace). Bref, on montre chaque transaction, chaque action, chaque utilisateur.

Un environnement plus complexe pour l'APM

Jusqu'à présent, l'APM traitait des applications n-tiers, des applications Web, desktops. On déployait des agents pour instrumenter efficacement. Mais avec l'arrivée du Cloud, du mobile, des conteneurs, des microserveurs, l'usage des API, il a fallu repenser l'APM et comment mesurer ces nouvelles applications. Dans ce contexte, difficile d'analyser son application manuellement.

Dans un contexte de VM ou de conteneurs, il n'est sans doute pas utile de montrer toutes les images, mais une image en particulier. Le déploiement des agents peut se faire au niveau de l'image, du système et non pas au niveau de l'application. Mais surtout, on ne sait pas où

l'app sera réellement installée / déployée. Il faut que l'environnement APM soit capable de faire de la découverte, c'est-à-dire de localiser de lui-même l'application.

Comme nous l'a précisé Dynatrace, ce qui est important c'est de pouvoir suivre toutes les couches de bout en bout, quelle que soit l'infrastructure derrière. Mais cela peut provoquer un afflux d'alertes, d'erreurs qu'il faut savoir gérer. Faut-il livrer brutes ces remontées ou les filtrer via des modèles statistiques et de l'intelligence artificielle ? Si un développeur peut s'en sortir avec quelques dizaines d'alertes, la situation devient vite ingérable en manuel.

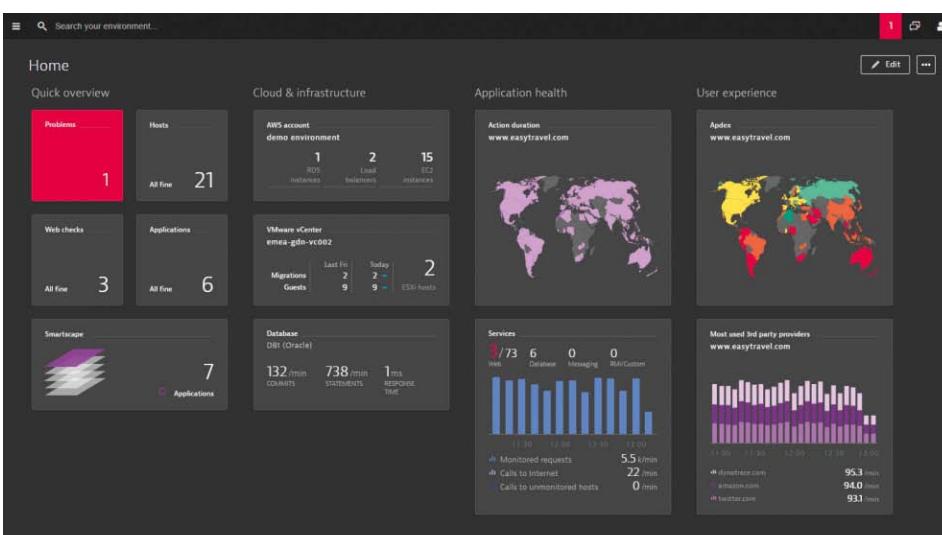
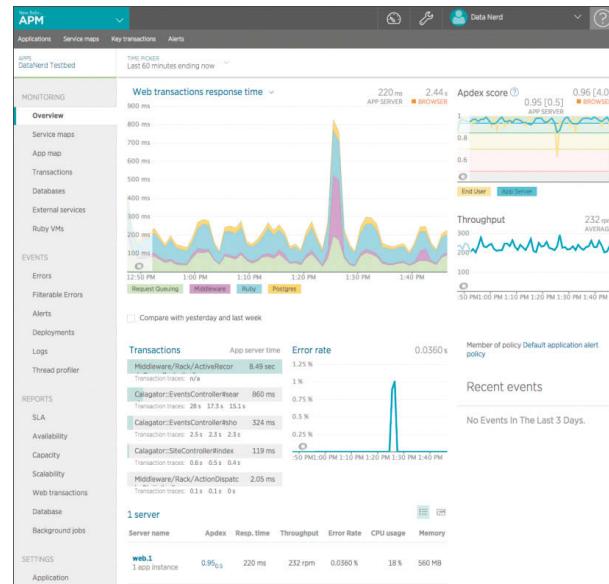
Bien entendu, ce qui va intéresser un développeur c'est d'aller de l'erreur / alerte remontée par le monitoring jusqu'au code en cause. Car l'APM est dans la démarche DevOps : parler aux opérations et aux développeurs.

Le but est toujours le même : détecter les problèmes de performances et savoir où ils se situent. A partir de là, problème des solutions pour améliorer ou supprimer la solution.

Bref, aujourd'hui, l'APM propose deux grands modèles :

- Modèle classique avec des métriques et des statistiques ;
- Modèle “modernisé” avec analyse profonde des données reçues.

Bien entendu, un outil APM n'est pas neutre en coût et temps. Les tarifs varient selon les éditeurs et selon des critères précis (nombre d'applications, d'utilisateurs, le volume des données, etc.). Certains acteurs, tels que IBM et HP, sont très centrés infrastructures limitant de facto leur périmètre. Mais la concurrence est très vive avec des acteurs comme Nudge APM, AppDynamics, Dynatrace, New Relics ou plus récemment Microsoft.



Un outil APM c'est quoi ?



Matthieu Walckenaer
CEO Nudge
www.nudge-apm.com

Tout d'abord définissons "APM" : l'Application Performance Management (APM) permet de superviser les applications en environnements de production et de recettes. Qu'elles soient hébergées en interne ou en externe. Ceci afin de diagnostiquer en temps réel les incidents ou ralentissements jusque dans le code de vos applications.

Beaucoup d'acteurs sur le marché de la supervision proposent de superviser la performance des applications de leurs clients, il y a en effet plusieurs solutions complémentaires :

- **L'analyse des logs**, notamment les **logs générés par le code**; elle nécessite néanmoins de prévoir un travail en amont des équipes de développement pour tracer les bonnes informations.
- **Les Robots**, solution autrement appelée **monitoring synthétique**; elle consiste à simuler des utilisateurs sur une application avec des scénarios prédéfinis. Cela donne une visibilité sur la disponibilité de l'application mais ne permet ni le diagnostic ni la supervision des utilisateurs réels.
- **Certaines sondes réseaux** permettent de remonter des indicateurs de disponibilité en termes de bande passante et remontent parfois des indicateurs transactionnels. Cela donne donc une visibilité de disponibilité de l'application pour les utilisateurs mais ne permet pas d'avoir une vision en profondeur dans l'application.
- **Les « agents », dits passifs**, permettent d'avoir une visibilité sur le comportement des utilisateurs et remontent des indicateurs en temps réel du fonctionnement de l'application jusque dans son code. Sans faire aucune modification ni au code source ni à l'application car les « agents » sont déclarés là où est hébergée l'application.

Le **GARTNER** définit l'APM à travers les 5 dimensions suivantes:

1. **END-USER EXPERIENCE MONITORING ou RUM** : capacité de capter de bout en bout la performance qui impacte les utilisateurs ;
2. **RUNTIME APPLICATION ARCHITECTURE DISCOVERY, MODELING AND DISPLAY** : capacité de faire la cartographie de tous les éléments (ou couches) auxquels fait appel l'application ;
3. **USER-DEFINED TRANSACTION PROFILING** : capacité de tracer les transactions

métiers et les requêtes afin de remonter les incidents et les diagnostiquer;

4. **COMPONENT DEEP-DIVE MONITORING IN APPLICATION CONTEXT** : capacité de descendre dans la ligne de code de l'application pour un diagnostic précis;
5. **ANALYTICS** : capacité d'analyser toutes les données remontées par les agents dans des tableaux de bord et d'aider aux diagnostics voire à l'anticipation des problèmes.

L'APM est donc la suite logique (et complémentaire) des outils de supervision utilisés au sein des organisations informatiques tels que les outils de supervision système et réseau. L'APM permet d'avoir des tableaux de bord complémentaires pour les équipes techniques en charge de superviser l'ensemble de l'architecture d'un système d'information et ainsi leur permettre de mettre le doigt sur l'origine de l'incident plus rapidement.

Une application n'est plus une boîte noire; les outils d'APM permettent de surveiller son comportement comme les équipes techniques surveillent déjà les composants hardware et réseaux utilisés dans l'entreprise.

Que va apporter un outil APM ?

Un outil APM va permettre de dépassionner les débats internes... Fini les discussions interminables entre les métiers, la DSI, la production, la MOE ou la MOA... L'APM permet de connaître précisément la source des incidents et donc de responsabiliser chacun au sein des équipes techniques.

Les valeurs ajoutées qu'apporte un outil APM sont nombreuses, il répond à de nombreux besoins :

- Superviser vos applications et les temps de réponses;
- Augmenter la satisfaction de vos utilisateurs;
- Augmenter vos revenus (chiffre d'affaire);
- Maîtriser les budgets de développements informatiques;
- Maîtriser les budgets de réduction de dettes techniques;
- Permettre aux équipes de concentrer leurs

efforts de travail là où ils ont une véritable valeur ajoutée;

- Savoir comment se comportent les utilisateurs sur les applications;
- Améliorer la fonction support pour apporter à vos clients une meilleure QoS;
- Permettre aux DSI de consacrer plus de budget à des projets favorisant la croissance et le soutien du changement de l'entreprise plutôt qu'à la maintenance du SI. (1)

Les bénéfices qu'un APM apporte aux équipes techniques sont importants (use case) :

- Meilleur dialogue entre les équipes internes et meilleur pilotage avec les prestataires (TMA, hébergement, ...);
- Remontées d'indicateurs clés pour assurer une meilleure satisfaction et un meilleur ressenti des utilisateurs de vos applications (vision de la performance côté utilisateurs et côté serveurs - de bout en bout);
- Des applications plus performantes livrées en production par les équipes Build et les équipes RUN qui peuvent comprendre le comportement des applications (amélioration continue);
- Réduire drastiquement le temps écoulé entre l'apparition de l'incident et sa résolution, même mieux : les anticiper.

Comment fonctionne un outil APM ?

La mise en place d'un outil d'APM est très simple surtout en mode SaaS (Software as a Service). Il suffit, une fois inscrit sur le site de l'éditeur, de télécharger un zip contenant un agent (Java, .Net, ...) ainsi que son fichier properties (selon les éditeurs) qu'il faut rajouter dans un répertoire de votre serveur d'application avec une directive à configurer (de type javaagent pour Java,...) dans le paramètre du

(1) les DSI consacrent plus de la moitié (55%) de leur budget à la maintenance du système d'information. Seulement 22% est alloué à des projets favorisant la croissance de l'entreprise. Et ce sont 23% qui sont affectés au soutien du changement de l'entreprise.

App : {Logiciel de démo} - 22/08 15:27 - 22/08 15:35

Vue globale Profiling Erreurs 0 Détails BETA

Emplacement ligne Nb écha... Runnable Blocked Waiting

Emplacement	ligne	Nb écha...	Runnable	Blocked	Waiting
(100.0 %) /owners	0	142	85.9 %	14 %	0 %
(100.0 %) ApplicationFilterChain.doFilter	208	142	85.9 %	14 %	0 %
(100.0 %) ApplicationFilterChain.internalDoFilter	241	142	85.9 %	14 %	0 %
(100.0 %) WsFilter.doFilter	52	142	85.9 %	14 %	0 %
(100.0 %) ApplicationFilterChain.doFilter	208	142	85.9 %	14 %	0 %
(100.0 %) ApplicationFilterChain.internalDoFilter	241	142	85.9 %	14 %	0 %
(100.0 %) OncePerRequestFilter.doFilter	76	142	85.9 %	14 %	0 %
(100.0 %) HiddenHttpMethodFilter.doFilterInternal	77	142	85.9 %	14 %	0 %
(100.0 %) ApplicationFilterChain.doFilter	208	142	85.9 %	14 %	0 %
(100.0 %) ApplicationFilterChain.internalDoFilter	303	142	85.9 %	14 %	0 %
(100.0 %) HttpServlet.service	227	142	85.9 %	14 %	0 %
(100.0 %) HttpServlet.service	620	142	85.9 %	14 %	0 %
(100.0 %) FrameworkServlet doGet	549	142	85.9 %	14 %	0 %
(100.0 %) FrameworkServlet processRequest	644	142	85.9 %	14 %	0 %
(100.0 %) DispatcherServlet.doService	719	142	85.9 %	14 %	0 %
(95.7 %) DispatcherServlet.doDispatch	790	136	85.2 %	14.7 %	0 %
(4.2 %) DispatcherServlet.doDispatch	817	6	100 %	0 %	0 %

GLOBAL
Temps moyen : 1.228 s
Appels : 14
Surcharge Nudge : 2.11 %
Statuts des threads
Runnable : 85.91 %
Blocked : 14.08 %
Waiting : 0 %

NOEUD
Sélectionnez un noeud pour visualiser ses données

App : {Logiciel de démo} - 22/08 14:32 - 23/08 14:32

Temps : Moyenne - Couches Satisfaction Navigateur

ms

Count

Java JAX-WS SQL JMS

Top 5 Transactions

Nom	Nombre	Trm	Global
Q_owners	3053	1,2 sec	82 %
Q_nurses	1524	0,4 sec	13 %
Q_synchronize/send/owner/2	1524	42,8 ms	1 %
Q_synchronize/merge/pet/{id}	1524	39,2 ms	1 %
Q_synchronize/send/pet/2	1524	33,5 ms	1 %

Top 5 SQL

Nom	Nombre	Trm	Global
Q_select_pets0_owner_id as owner4_0_2..., pets0_id ...	4587097	0,0 ms	50 %
Q_select_visits0_pet_id as pet4_1_1..., visits0_id as id0_ ...	4587106	0,0 ms	44 %
Q_select_owner0_id as id0_..., owner0_address as add0_ ...	3053	1,1 ms	3 %
Q_call ?0	33544	0,1 ms	2 %
Q_Connect	16772	0,0 ms	0 %

Evénements Notables

lancement de l'application. Une fois votre serveur relancé, les données remonteront automatiquement dans les tableaux de bord.

En mode « On Premise », le client doit installer et héberger le portail permettant d'analyser les données et de consulter les tableaux de bord dans son datacenter.

Exemples d'indicateurs et de métriques remontés par un APM :

- Détails de toutes les transactions métiers ;
- Real User Monitoring (RUM) ;
- Identification des points chauds (drill down dans le code) – Profiling ;
- Suivi de la satisfaction des utilisateurs (APDEX) ;
- Détails des transactions en erreurs et frustrantes ;

- Suivi de session des utilisateurs ;
- Vision détaillée de toutes les couches (SQL, NoSQL, Webservices, JMS, ...) ;
- Historique des déploiements ;
- Photographie en temps réel et automatique de l'ensemble des composants, librairies et versions utilisés dans vos applications ;
- Cartographie des flux par couches ;
- Supervision système ;
- Configuration de JMX ;
- Gestion des SLA et Alertes ;
- Reporting ;
- API Rest ;
- Intégration automatique dans des outils en place chez le client (ex : Elastic, Nagios, ...) ;
- ...

En conclusion, l'APM est devenu un outil incontournable, en production, pour superviser le fonctionnement des applications et être capable de diagnostiquer et de comprendre les problèmes de performances en temps réel. Dans les environnements de tests ou de développement, un outil d'APM permet de venir compléter les outils de tests de performance pour diagnostiquer et améliorer les applications avant leurs mises en production.

C'est donc un formidable outil pour fluidifier les communications en interne et avoir une vue d'ensemble du comportement des applications afin d'améliorer l'expérience et le ressenti des utilisateurs.

Du produit à sa **commercialisation** en ligne ... et au-delà : réussir sa boutique en ligne

Partie 1

La conception d'un site e-commerce ne peut pas se résumer à une simple opération technique. Un ensemble de sujets tant métiers que techniques sont à considérer, avant sa mise en ligne. Une fois le site ouvert aux utilisateurs, il convient de l'exploiter, de le faire vivre et le maintenir. Ce dossier développe les aspects théoriques liés à la mise en œuvre d'un site e-commerce, ainsi qu'un exemple de site basé sur le développement d'un instrument de musique MIDI réalisé il y a quelque temps : la tirasse MIDI. [1]

- Jean-Marie HEITZ
 - Xavier FLAMANT
- Netapsys**



Pour les curieux ou celles et ceux que cela intéresse, découvrez plus en détail la réalisation de la tirasse à l'adresse <https://blog.netapsys.fr/tirasse-midi-du-besoin-a-la-realisation/>

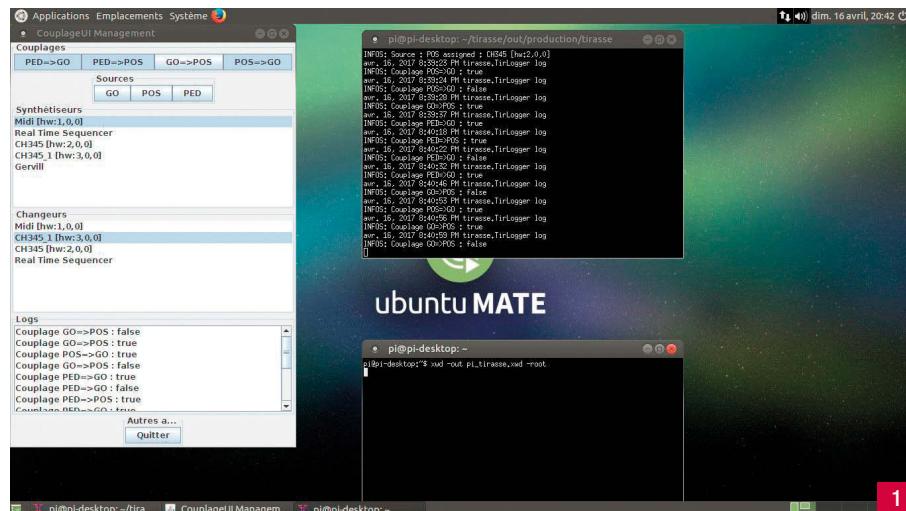
Pour ce dossier, nous allons utiliser la plateforme Magento. Les grands principes et les fonctionnalités détaillées sont valables dans les autres plateformes comme Prestashop, UberCart, etc. [1]

Etapes préalables à la conception du site : identifier son marché et sa cible de clients

Préparons notre plan de conquête du ... marché

Il faut conserver à l'esprit qu'un site e-commerce est un "outil" et non une fin en soi. Son objectif premier est de permettre à une marque d'offrir à ses clients une expérience d'achat unique et singulière qui parfois s'inscrit en complément d'espaces physiques existants (magasins). Mais ce n'est pas tout, ce nouveau canal de vente doit aussi répondre à des attentes économiques notamment en termes d'activité de vente et de profits. Ainsi, un site e-commerce est un canal de vente à part entière mais non exclusif. Des places de marché, comme Amazon Marketplace, ou Ebay sont également d'autres supports envisageables.

Trop d'entreprises se contentent d'un site "vitrine" de présentation de leurs produits et déportent la vente sur un autre canal (échange de mails puis un paiement par virement par exemple). Cette présence limitée, si elle permet une présence en ligne en offrant un premier



Le logiciel de tirasse en action sur un Raspberry Pi

niveau de services, n'est toutefois pas totalement satisfaisante.

// Et pour la tirasse ...

En quelques mots, la tirasse MIDI est à la base un dispositif logiciel qui reçoit en entrée un signal MIDI (note de musique, durée, volume, ...) en provenance d'un instrument et qui transmet en sortie un signal MIDI à destination d'autres instruments également. La tirasse est donc une solution logicielle, qui lie à la fois musique et informatique. Nous allons dans notre exemple commercialiser cette solution sur un site e-commerce.

La création d'un site e-commerce naît d'une réflexion souvent portée par une direction métier (commerce, marketing) qui doit, avant de se lancer, apporter des réponses à des éléments structurants et essentiels : nom du service (marque), typologie des clients, nature de la concurrence directe et indirecte, objectifs de ventes, moyens de promotion et de communication, politique de prix, stratégie d'animation commerciale, analyse des flux comptables et

logistiques, stratégie de sourcing... Ces éléments sont souvent formalisés dans un business plan qui pourra s'appuyer sur des analyses stratégiques de type PESTEL ou Porter par exemple.

// Et pour la tirasse ...

Dans le cadre du futur site e-commerce destiné à commercialiser la tirasse MIDI, nous avons identifié un marché de niche : les organistes (disposant d'une installation modulaire). Ce cœur de cible peut être élargi aux revendeurs d'instruments, les écoles de musique et éventuellement les chorales. Le site devra donc répondre aux besoins de ces différentes typologies de clients. Le produit sera une solution sous licence libre, mais des frais de distribution pourront être demandés, sous la forme d'une promesse de don à faire à une œuvre caritative laissée au choix du client. Dans notre exemple, les flux sont plutôt limités :

- Pas de flux logistique (logiciel délivré en ligne à la fin de la transaction) ;

- Pas de flux comptable (pas d'interfaçage avec un ERP ou un back office de gestion) ;
- En revanche, existence de flux commerciaux via une base de données Client ou une solution tierce de type CRM (disponibilité de nouvelle version, services proposés, produits connexes (cross sell)).

La charte graphique est celle par défaut de Magento.

Il est important avant l'ouverture d'un site e-commerce de ne pas oublier certains aspects légaux par exemple :

- Les conditions générales de vente ;
- Les conditions générales d'utilisation du site,
- Les propriétés spécifiques du site (identité de l'éditeur, adresse physique, hébergeur ...) ;
- Les informations relatives au traitement automatisé des données nominatives (obligations relatives à la CNIL, cookies, publicité, ...) ;
- Les mentions légales obligatoires comme le RCS ;

Les prérequis techniques



Pas de bons résultats sans de bons outils et une préparation minutieuse. Alors préparons-nous précautionneusement !

Les prérequis techniques sont en général ceux exigés par les sites Internet, complétés par les prérequis des systèmes utilisant des bases de données de manière assez intensive. À cela s'ajoutent souvent dans les sites e-commerces des composants quasi obligatoires tels que des serveurs de cache. Mais passons en revue les principaux prérequis :

Un nom de domaine : bien qu'un site pourrait, du point de vue technique, tout à fait utiliser un nom de machine fourni par l'hébergeur, les clients doivent pouvoir identifier simplement l'entreprise : un nom de domaine ainsi qu'une résolution de nom correspondante est donc indispensable. Une précaution pour l'avenir sera néanmoins de vérifier si le registrar sera en capacité de supporter les extensions liées à IPv6 et celles liées à DNSSEC.

// Et pour la tirasse ...

Pour les besoins de l'article le domaine sera tirasseorguelectroniqueliturgique.fr. Une recherche rapide effectuée auprès d'un registrar indique que ce nom est disponible. Pour le développement local, tirasse.local sera employé.

- Une messagerie électronique : à minima, il convient de disposer de redirections vers des adresses électroniques tierces, mais pour capitaliser sur la marque, il est plus pertinent

d'avoir un service de messagerie pour le domaine fournissant des services à accès sécurisés (IMAP pour la réception, SMTP STARTTLS ou SMTPS pour l'envoi et bien entendu HTTPS pour le webmail)

// Et pour la tirasse ...

Au niveau de la messagerie électronique, nous choisissons d'avoir au moins deux adresses : une adresse de contact et une adresse pour la boutique.

- Une plateforme e-commerce contrainte par ses propres prérequis techniques. Ce choix doit être fait avec minutie, il est en général délicat a posteriori de passer d'un système à un autre.

// Et pour la tirasse ...

Dans le cadre de cet article, nous utiliserons la solution open source (leader) Magento.

- Un serveur HTTP (avec sa bande passante) : il servira à fournir la plupart des éléments relatifs au site e-commerce. En fonction de l'importance du site, il peut être utile de le doubler (un serveur de front-office et un autre de back-office). Plusieurs noms pourront éventuellement être servis de sorte à accroître le nombre de connexions en parallèle d'un navigateur pour récupérer le site plus rapidement.

// Et pour la tirasse ...

Le site étant modeste, nous choisissons un seul serveur web dédié situé dans un réseau interne hébergé, sur lequel plusieurs serveurs physiques sont possibles. En termes de serveur HTTP, le choix se fait entre Nginx et Apache HTTPD; nous retiendrons pour notre part ce dernier.

- Un ou plusieurs certificats HTTPS : jusqu'il y a peu, on considérait que seules certaines sections critiques se devaient d'être accédées via HTTPS, comme les informations clients mais aujourd'hui, la fourniture d'un service d'accès totalement en HTTPS devient la norme. Trois choix s'offrent à nous :

- Un seul certificat avec un nom unique ;
- Un seul certificat avec un nom générique (un certificat wildcard) ;
- Un certificat par nom desservi.

En cas de fort trafic, il arrive que des proxys SSL frontaux soient mis en place pour chiffrer les communications avec les utilisateurs finaux, de sorte à délester les serveurs métiers.

// Et pour la tirasse ...

Le site étant initialement modeste, nous utiliserons un certificat wildcard de manière à pouvoir bénéficier des effets de multiplexages dus à l'utilisation de plusieurs noms. Compte

tenu du nombre de serveurs HTTP, c'est une décision fonctionnelle. Néanmoins, si le site se développe, il faudra s'orienter vers des certificats non Wildcard (chaque instance disposera d'une clé privée propre).

- Le serveur de base de données : il est la clé de voûte de toute plateforme e-commerce. C'est lui qui stocke une bonne partie des informations sur les produits, clients, devis et commandes. Il est donc critique dans l'architecture de la solution. Hautement sollicité, c'est un système qui doit être performant sous l'angle matériel (disques durs, contrôleurs de disques durs, bus, mémoire vive) mais aussi vis-à-vis de l'administration logicielle (dimensionnement de l'instance du serveur de base de données pour ne citer qu'elle). Une particularité de la base de données dans un site e-commerce est que cette dernière n'est pas figée, elle "vit", dans la mesure où à tout moment des actions utilisateurs peuvent engendrer des créations/modifications de données.

// Et pour la tirasse ...

Le serveur de base de données choisi est MySQL, SGBD traditionnellement utilisé avec Magento. Pour l'hébergement de la base de données, nous souhaitons que celle-ci soit hébergée sur des SSD redondés (via du RAID 10). Nous souhaitons une sauvegarde complète chaque nuit, associée à une sauvegarde des logs de la base de données. Nous acceptons de perdre jusqu'à une journée de commandes, sachant que nous allons configurer Magento pour nous envoyer une copie de mail pour chaque création de compte ou commande passée. Ainsi, nous pourrons quand même disposer des informations nécessaires pour recréer au besoin les clients et les commandes en se basant sur les mails.

- Le mécanisme de stockage du cache : dans un site e-commerce, il est important que les temps de réponse soient rapides même lors de la consultation par un grand nombre de clients simultanément. Or, générer une page complète peut prendre du temps. C'est là que les caches entrent en jeu : ils permettent de ré-exploiter des entités qui ont déjà été calculées et supposées être encore valides pour l'usage qu'on souhaite en faire (techniquement, on calcule une clé de stockage dans le cache, le résultat peut être réutilisé si cette clé est présente et l'entrée n'est pas trop vieille). On trouve des caches pour un certain nombre d'éléments, tels que la gestion de la

configuration ou la génération de blocs HTML dans la page. L'inconvénient est qu'il est nécessaire de disposer de clés de cache suffisamment précises pour tenir compte des différents éléments conditionnant l'affichage ; il faut parfois changer le type de programmation (usuellement remplacer de la programmation PHP par de la programmation Ajax au niveau du navigateur) pour optimiser l'utilisation du cache. Il faut également stocker le résultat, dans un stockage rapide.

// Et pour la tirasse ...

Magento propose plusieurs types de stockage, dont le stockage du cache dans un serveur Redis. Le serveur Redis a comme gros avantages sa rapidité et sa sauvegarde de données dans un fichier, ce qui permet de ne pas "forcément" partir d'une situation de cache froid lors d'un redémarrage (en contexte de déploiement, l'habitude est de vider les caches).

- Le mécanisme de stockage des sessions : le maintien des sessions est, comme vu précédemment, essentiel au bon fonctionnement d'un site e-commerce. Il s'agit non seulement de stocker la liaison avec le cookie de session transmis au navigateur pour retrouver les éléments liés dans la base de données, mais également stocker les données d'état volatiles qui doivent être accessibles rapidement, par exemple la protection de formulaires avec des clefs.

À nouveau, Magento propose plusieurs types de stockage, dont du stockage Redis. À ce niveau, nous pouvons comprendre que le stockage sur disque dur présente un intérêt, de sorte qu'un client notamment ne perde pas ses données de panier lorsqu'il n'est pas logué et que le serveur de cache de sessions est redémarré. Néanmoins étant donné la volatilité des données et le fait que l'on peut souhaiter en cas de problème repartir sur des bases saines et une situation "stable", nous prenons le parti de dire que perdre les données de session est un moindre mal et qu'il n'est pas nécessaire de les stocker. Un serveur Memcached peut également faire l'affaire dans notre exemple, car les produits proposés sur le site ne seront pas très variés. Il est donc rapide et simple pour un utilisateur de refaire son panier si nécessaire. À noter qu'en multi-instance (plusieurs serveurs Magento faisant tourner le même FO par exemple), il devient primordial d'avoir un mécanisme de sessions centralisé, pour que le traitement sur différents frontends successifs conduise à une expérience homogène pour l'utilisateur.

- Le serveur de cache full page cache en frontend : ce genre de serveur est un serveur qui peut remplacer un full page cache intégré à Magento : se positionnant devant les serveurs frontend, il récupère d'abord les requêtes et ne les transmet au serveur frontend que s'il ne peut pas y répondre lui-même. Ce type de serveur se trouve surtout dans les installations où il n'y a pas de full page cache, ou en frontal d'instances extrêmement sollicitées. La programmation des pages est impactée par ce type de serveur. De plus sa configuration n'est pas anodine. Un exemple connu de serveur de cache est Varnish.



Pour les curieux : mettre en place du Varnish n'est pas toujours évident et peut demander des efforts de débogage : <https://blog.netapsys.fr/retour-dexperience-sur-un-debuggage-varnish/>

Comme le site de la tirasse ne sera vraisemblablement pas grandement sollicité, la mise en place d'un full page cache en frontend serait disproportionnée. Pour qu'un cache se justifie, il faut que ce cache soit réellement utilisé pour servir des pages : avec un trafic moindre, il est probable que des délais de validité extrêmement longs soient nécessaires à son utilisation.

- Le service de recherche : généralement, sur les grands sites de vente, la quantité d'articles disponibles est telle qu'il est plus efficace d'utiliser les moteurs de recherche intégrés pour trouver le produit désiré. Cependant, la réalisation d'un système de recherche présente quelques difficultés : il faut savoir dans quel contenu chercher (pages de produits, pages de catégories, pages CMS, ...) et tenir compte des documents pour les indexer au mieux. En ce qui concerne la tirasse, l'intérêt de pouvoir faire des recherches ne réside pas tant sur le fait de rendre disponible un produit que de pouvoir chercher à fournir à l'utilisateur la réponse à son besoin sous la forme d'un aspect produit auquel il sera sensible (sous diverses formes : textes, images, voire du son et des vidéos). Ces derniers types de documents sont certes difficilement indexables en eux-mêmes, mais il est toujours possible de chercher à renseigner leurs métadonnées pour les rendre présentables par un moteur de recherche. De plus, il est important de disposer d'outils puissants d'analyse des requêtes utilisateurs. Dans le cadre d'une évolution future, nous partirions donc plutôt pour un serveur SOLR dédié, avec une configuration et une interface d'utilisation adaptées au besoin, donc personnalisées.

- Les outils d'analyse de trafic : ils sont importants, car ils permettent de suivre la fréquentation du site et d'en savoir plus sur les réactions des utilisateurs afin de pouvoir l'optimiser. Ces outils peuvent se situer aussi bien au niveau du serveur (logs Apache HTTPD par exemple) qu'au niveau client (l'un des outils les plus connus étant Universal Analytics). Cependant, comme ces outils doivent pouvoir être, dans une certaine mesure, déployables par des non techniciens, de plus en plus de gestionnaires de balises sont utilisés pour les intégrer, ces derniers permettant d'avoir une plus grande souplesse par rapport à des procédures de déploiement souvent longues à mettre en place.

Même dans le cadre d'un site modeste il peut être intéressant de disposer d'outils de ce genre : une utilisation standard de GTM et de UA permettront d'obtenir d'emblée des informations avec peu d'efforts. L'effort de personnalisation (qui consiste souvent d'abord en l'enrichissement du DataLayer, puis en l'enrichissement des tags) pourra être réalisé lors d'une évolution du site.

- Les outils d'analyse de performance : ils permettent de tester la rapidité de réponse des pages et proposent généralement des axes d'amélioration. Ils peuvent également avoir un impact sur l'appréciation des moteurs de recherche, en vérifiant, par exemple, que les moteurs de recherche réussiront à analyser les documents produits. Dans cette rubrique, on pourra donc utiliser des éléments comme les Google Web Tools, PageSpeed, ou Gtmetrix.

Il est important de tenir compte de ce genre d'outils qui cherchent à améliorer le confort des utilisateurs, d'autant que ce confort est également une qualité qui pourrait éventuellement être prise en compte dans le classement de résultats sur une recherche.

- Le Content Delivery Network : si on souhaite proposer des contenus "gros consommateurs" en bande passante, il devient intéressant d'utiliser un réseau de distribution de contenu qui est spécialisé dans ce genre de besoins, afin de préserver les ressources pour le véritable site boutique. Certains CDN font maintenant même partie de la "vie courante", comme Youtube ou Cloudfront.

Etant donné que le site de la tirasse devra fournir des vidéos pour la montrer en configuration et en action, le CDN sera requis dans le cadre de ce projet.

- Les transporteurs (flux de livraison) : généralement plusieurs situations d'enlèvement et de livraison de colis spécifiques sont considérées :
 - pas de livraison du tout, notamment lorsque le produit est numérique, ou que le client se déplace sur le site physique ;
 - dépôt à un centre d'envoi ou récupération dans un point colis correspondant ;
 - enlèvement chez le fournisseur ou réception à domicile.

En outre, il faut surtout veiller qu'il existe un, voire deux moyens de livraisons pour tout produit vendu sur le site et toute adresse de livraison acceptée. En fonction des conditions contractuelles avec les transporteurs, il arrive que la mise en place de la gestion des frais de livraison soit plus ou moins simple à intégrer, allant de frais fixes jusqu'à l'appel de web services tiers.

Dans le cadre de la tirasse, le produit étant logiciel, nous compterons sur les mécanismes natifs de Magento pour assurer la livraison sous forme de téléchargement.

- Les moyens de paiement : il y a de nombreux prestataires et méthodes de paiement disponibles ; cependant, on peut segmenter les moyens de paiement globalement en quatre catégories :
 - Les paiements hors ligne : ces paiements se font de manière non électronique : typiquement, il s'agit de paiements en espèce ou en chèque, ou de virements qui sont effectués hors site. Après réception du paiement, le paiement doit être validé manuellement dans le logiciel e-commerce depuis le back-office ;
 - Les paiements internes au site : c'est le cas des cartes cadeaux, des avoirs, voire des coupons de réduction, qui sont des flux enregistrés dans le site, utilisés pour réduire le montant à payer, éventuellement jusqu'à obtenir une commande gratuite ;
 - Les paiements par carte (et les paiements de type Paypal) : le plus généralement, le système prépare les paramètres nécessaires pour rediriger le client sur le site de paiement avec lequel le commerçant a un contrat. Dans ces paramètres, sont mentionnés le client, le commerçant, le montant, ainsi qu'une signature des paramètres qui est réalisée grâce à un secret partagé avec le système de paiement. Le client passe alors de manière directe ou indirecte (iframe) sur le site de paiement et saisit ses identifiants de carte bancaire. De là, il y a une demande d'autorisation et éventuellement une demande de capture du montant qui est demandée. Si la demande de capture

n'est pas produite en même temps, elle peut être émise ultérieurement, par exemple par un traitement récurrent automatisé. Le résultat du traitement est passé du site de paiement au site e-commerce via un appel direct appelé généralement IPN (Instant Payment Notification) et le client est ensuite redirigé sur le site du commerçant, sur une URL dépendant généralement du résultat (URL de succès, URL d'échec, URL d'annulation, si le client n'a pas cherché à payer). Il est à noter qu'avec la différence entre autorisation et capture, on peut parfois demander une autorisation pour un certain montant et faire ensuite des captures successives jusqu'à concurrence du montant autorisé : cela peut par exemple être utilisé dans le cadre d'une commande envoyée en plusieurs fois, où le commerçant débite les articles uniquement lors de l'envoi ;

- Les paiements par prélèvement SEPA : depuis son avènement, le système SEPA simplifie les paiements électroniques. Concrètement : le vendeur établit un Mandat SEPA, avec un certain formalisme (présence de mentions obligatoires). Le client le signe et le vendeur le conserve. Le vendeur pourra ensuite demander via son prestataire de paiement des ordres de prélèvement et ce, en accord avec le mandat qui a été signé et uniquement après information préalable du client. En ce qui concerne la signature du mandat SEPA, elle peut se faire soit manuellement (signature papier) soit électroniquement (et plutôt numériquement, via des certificats). La démarche de signature électronique est d'ailleurs facilitée par une pratique assez récente appelée le "cloud signing" : un tiers s'occupe d'identifier la partie qui doit signer un document, génère un certificat pour le signataire et le signe lui-même pour le compte du signataire (avec son accord) après s'être assuré de son identité et de sa volonté de signer. De la sorte, l'usage de la signature électronique peut simplifier les démarches. Avec la signature du mandat SEPA qui est obligatoire, on peut également chercher à ajouter le contrat de vente aux documents à signer. Néanmoins, il faut faire attention à ce que le tiers s'occupant de la signature électronique soit bien reconnu, comme prestataire de services de confiance qualifié, par les autorités, de sorte que la signature électronique dispose de sa valeur juridique. Il arrive que des prestataires de paiement fournissent aussi des possibilités de signature électronique. Le SEPA a un intérêt surtout dans le cas des paiements récurrents (par exemple, des paiements d'un abonnement).

Dans le cadre de la tirasse, un nouveau mode de paiement sera implémenté : la promesse de don. L'idée est de donner une somme indicative au niveau des produits, somme reversée aux associations. Dans ce cas, le produit "commercialisé" sur le site n'est pas la tirasse en elle-même, mais la fourniture de la tirasse.

- La supervision et la maintenance sont deux domaines qu'il ne faut en aucun cas négliger. En termes de supervision, il y a tout d'abord la surveillance des différents serveurs qui ont été cités jusqu'à présent : il s'agit donc de surveillances système et applicatives pour lesquelles il existe de nombreux outils, tels que Nagios et ses différents plugins. Cependant, il est envisageable de mettre en place une surveillance spécifique à Magento, comme vérifier si les tâches de maintenance dans le cron Magento se planifient et s'exécutent bien. Ou encore il est envisageable, après un certain temps de fonctionnement du site, de définir des paramètres minimums de fonctionnement (par exemple sur l'accroissement du nombre de paniers ou de commandes), ou chercher à détecter des problèmes dans les flux de commandes (par exemple, monitorer le flux des commandes pour contrôler la bonne arrivée des IPN). D'autre part, en ce qui concerne la maintenance, il y a lieu de prévoir dans la mise en place du site la possibilité de mettre le site en mode "maintenance", c'est-à-dire un mode qui permettra un accès restreint pour des déploiements, notamment, qui permettra de ne laisser arriver sur la partie "e-commerce" du site que certaines IP autorisées. Il convient également de se familiariser avec des outils en ligne de commande de Magento, tels que les shells Magento et ne pas hésiter à se développer des shells répondant aux besoins métiers spécifiques. Enfin, il faut souligner que certains outils, comme n98-magerun (<https://github.com/netz98/n98-magerun>) sont particulièrement utiles, qu'il s'agisse de vérifier les planifications, de lancer une tâche cron manuellement, ou de lancer des morceaux de code particuliers. [2]



Pour les curieux : une autre possibilité explorée : interfaçer Magento avec du SNMP, ce qui est une manière d'intégrer la supervision d'un applicatif avec des outils systèmes (<https://blog.netapsys.fr/integration-magento-et-snmp/>).

```
root@arttir:/var/www/magento# ./n98-magerun.phar sys:info
It's not recommended to run n98-magerun as root user

[Magento System Information]

+-----+-----+
| Name | Value |
+-----+-----+
| Version | 1.9.3.3 |
| Edition | Community |
| Cache Backend | Zend_Cache_Backend_File |
| Cache Directory | /var/www/magento/var/cache |
| Session files | |
| Crypt Key | 2d9703c0d59b627d5017cb8efaa6dd8 |
| Install Date | Fri, 07 Jul 2017 19:15:55 +0000 |
| Vendors (core) | Mage, Zend |
| Vendors (community) | Ome, Phoenix |
| Vendors (local) | Jitn |
| Attribute Count | 132 |
| Customer Count | 2 |
| Category Count | 3 |
| Product Count | 2 |
+-----+-----+
root@arttir:/var/www/magento#
```

Exemple d'utilisation de magerun

- La sauvegarde : il est connu que cette dernière est un élément clef de tout système d'information. À travers la liste des composants énumérés ci-dessus, nous avons déjà une bonne idée de ce qu'il faut absolument sauvegarder et des pertes permises (données volatiles ou reconstructibles). Il est évident qu'il faut sauvegarder la base de données puisque c'est le cœur du système. De plus, il ne faut pas oublier qu'en cours d'utilisation normale, des fichiers peuvent être importés sur l'instance, notamment des images. Ainsi, il faudra aussi penser à sauvegarder le répertoire /media, sachant que ce répertoire peut devenir assez volumineux. En ce qui concerne les fichiers liés à Magento même, il peut toujours être arrangeant de disposer des fichiers sous forme de sauvegarde, mais il faut également pouvoir être en mesure de réinstaller les fichiers en se basant par exemple sur les sources stockées dans un dépôt Git, notamment en cas de suspicion de compromission. Dans le même esprit, on gardera en particulier un exemplaire de fichiers importants de configuration, comme app/etc/local.xml, ou app/etc/applied.patches.list. Pour l'indexation et le cache, le mieux est de pouvoir les régénérer si nécessaire.



Il faut bien garder à l'esprit qu'une sauvegarde n'est utile que si elle peut être restaurée : il convient donc de vérifier l'utilisabilité des sauvegardes et la mise à jour des procédures qui en découlent en fonction de la vie du système. Un bon exercice de restauration est la mise à jour des environnements de pré-production et de recette, par exemple.

- Les différents environnements requis : il est bien entendu nécessaire de disposer d'un environnement de production, mais ce n'est pas le seul. Il faut au moins disposer aussi d'un environnement de pré-production, ressemblant le

plus possible à la production. De même il faut disposer d'un environnement d'intégration pour lancer les tests unitaires et d'un environnement de développement (ce dernier étant le plus souvent présent sur les postes des développeurs). Le fait d'avoir ces différents environnements permet de pouvoir s'assurer le plus possible que les modifications apportées ne porteront pas préjudice à la production.

Dans le cadre de la tirasse, comme le projet restera petit, il est envisageable de mettre la production et la pré-production sur les mêmes machines. Néanmoins, pour des projets plus vastes, il convient de bien séparer et isoler les environnements, car il ne faut pas que des problèmes comme une surconsommation de ressources en pré-production n'affecte la production.

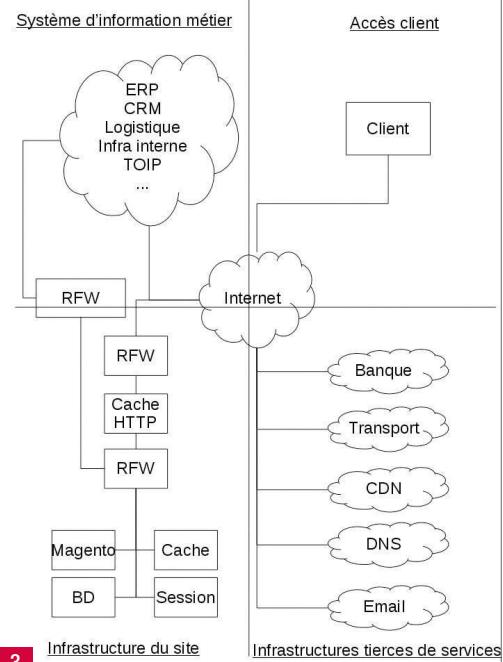
On a maintenant une longue liste des prérequis techniques. Voyons brièvement comment nous pouvons les situer dans un exemple de schéma conceptuel simplifié d'infrastructure : [3]

Le schéma divise l'infrastructure en quatre parties :

- L'infrastructure client, nécessaire au client pour accéder au site ;
- L'infrastructure tierce de services : fournisseurs de services apportant des fonctionnalités au site par divers moyens techniques ;
- Le système d'information métier : cœur du système d'information de l'entreprise ;
- L'infrastructure du site : infrastructure dédiée au site.

Comme les serveurs et sites web sont des cibles communes d'attaques, il convient d'isoler le site web pour limiter les éventuels dégâts : d'où des routeurs firewall (pour simplifier, car ils pourraient aussi assurer des fonctionnalités plus avancées comme de l'inspection de trafic ou du proxy applicatif, voire de l'offloading SSL) à plusieurs niveaux. Etant donné qu'il y a beaucoup de trafic en mode anonyme (client non logué) avec des pages récurrentes, on peut penser mettre un cache HTTP en front et ne laisser arriver au serveur Magento que du trafic qui est passé par lui (contrôles supplémentaires possibles). Au niveau du cœur de l'infrastructure du site Magento, nous avons uniquement représenté les éléments les plus fréquents à savoir le serveur web, le serveur de BD, le stockage du cache et le stockage des sessions - ce qui n'exclut pas les autres outils présentés auparavant. En revanche, nous pouvons nous poser la question s'il convient de laisser les machines dans la même zone de sécurité, ou s'il ne faudrait pas plutôt chercher à contrôler le trafic entre les machines ... Mais aucune des deux

Exemple de schéma conceptuel d'infrastructure



Exemple de schéma conceptuel d'infrastructure

approches n'est parfaite. Il est à remarquer que deux liaisons distinctes sont représentées pour la connexion à Internet entre l'infrastructure métier et l'infrastructure site web : les caractéristiques de trafic ne sont pas les mêmes et l'hébergement du site peut par exemple être déporté sur un cloud. Néanmoins, il pourrait être intéressant d'avoir une liaison privilégiée entre les deux infrastructures, en particulier pour les opérations d'exploitation (commerciaux utilisant le back office de Magento) ou pour les opérations d'administration/ maintenance.

MISE EN PLACE DU SITE E-COMMERCE

Dans un premier temps nous préparons l'installation pour le site sur un environnement de développement : un serveur Apache HTTPD et un serveur MySQL peuvent suffire pour commencer et installer une instance Magento (la résolution de nom peut être assurée par des entrées dans un fichier host ou par l'utilisation du domaine .local lié au multicast DNS). Le serveur Apache HTTPD doit pouvoir effectuer des réécritures et satisfaire des prérequis techniques de Magento. (http://docs.magento.com/m1/ce/user_guide/magento/system-requirements.html). Au niveau du serveur MySQL, une base avec un compte réseau ayant les droits, est nécessaire. Pour installer Magento, il suffit de récupérer sur le site de Magento l'archive, de la décompresser dans

la racine de l'hôte configuré dans Apache HTTPD et appliquer les permissions selon les recommandations de la documentation (http://devdocs.magento.com/guides/m1x/install/installer-privileges_before.html) puis http://devdocs.magento.com/guides/m1x/install/installer-privileges_after.html). Un installateur web permettra d'installer Magento. Les informations demandées sont assez triviales (acceptation de la licence, informations de locale, timezone et devise, informations de base de données, d'URL front et back, stockage de session, informations pour créer un login admin en BO). Lors de l'installation, est produite ou indiquée une clé qui permet le chiffrement symétrique de données dans la base et est créé le compte d'administration du back-office. La configuration principale est stockée dans le fichier « app/etc/local.xml ». C'est surtout ce dernier qu'il faudra décliner pour chaque environnement. [4]

Une fois réalisée l'installation de Magento, tant que tout le code est natif, il est nécessaire de rechercher et d'appliquer les patchs disponibles.



Pour les curieux : il est également possible de mettre l'installation de Magento en échec ... quand on cherche à utiliser IPv6 : <https://blog.netapsys.fr/magento-et-ipv6/>

Configuration de mon "magasin" : faite presque une fois pour toute

Ensuite, nous préparons le site dans le back-office. Il y a de nombreux éléments à prévoir. Le premier est la mise en place de la structure hiérarchique des « stores ». Magento dispose de

trois niveaux de configuration de boutique (en dehors du niveau de configuration global) :

- Le website : niveau le plus élevé de configuration. Par exemple nous trouverons des listes de clients séparées (ou bien communes) entre les différents websites à ce niveau ;
- Le groupe de stores : c'est à ce niveau qu'est définie une catégorie racine et où est proposée une navigation personnalisée ;
- Store : gestion de la valeur des attributs EAV. [5]

Lors de son installation, Magento crée par défaut deux websites, deux store groups et deux stores, en raison de la différence entre le front office et le back office.

Dans le cadre de la tirasse, nous resterons sur une boutique domiciliée en France qui prévoit de vendre uniquement à des clients résidant en France. Il n'y a pas de raison de faire des vues séparées pour les différentes typologies de clients car les arguments qui intéressent une typologie pourraient aussi être pris en considération par une autre typologie. De ce fait, la structure par défaut de Magento convient.

La configuration de Magento consiste en grande partie en des réglages qui sont situés dans le back office, dans le menu Système, Configuration. Ce système de configuration constitue un pilier de Magento, utilisant un système générique de stockage de configuration, avec possibilité de changer la configuration en fonction du contexte. Les éléments de configuration eux-mêmes sont hiérarchisés : on trouve d'abord la notion de section, qui est la page de configuration sur laquelle on se trouve, puis la notion de groupe (les zones centrales qui peuvent

s'étendre ou se réduire), puis la notion de champ de configuration.

Il va tout d'abord falloir faire la configuration de base de la boutique et parcourir les différents éléments de la configuration pour régler les informations sur les pays desservis, les informations de contact liées à la boutique, le header et le footer des pages, les adresses mails à employer... : c'est dans le groupement « General » qu'il y a le plus de configuration initiale à faire. Ensuite, dans les groupements Catalog et Customer, il s'agit plutôt de vérifier les options de Magento et éventuellement les adapter à son mode de fonctionnement métier, notamment pour désactiver les fonctionnalités non voulues (la gestion des stocks par exemple, dans le cadre de la tirasse). Il faut penser à bien sauvegarder chaque section avant de passer à la suivante. Mention spéciale pour les templates d'adresses (Customers, Customer Configuration) qu'il convient de configurer pour avoir une présentation des adresses conforme aux standards postaux (surtout pour la position du code postal).

Les murs porteurs : les réglages liés aux ventes : savoir établir un devis

Le groupement de section Ventes est assez critique, car il permet notamment de configurer au moins partiellement les moyens de livraison et de paiement. En outre, certains réglages d'affichage doivent également être configurés avec soin, en particulier l'affichage des prix en HT ou en TTC et ce, dans les différents endroits où ils sont présents (fiche produit, panier, commande, facture, ...).

Un magento brut, après installation (et sans les données d'exemple)

```
juillet@juillet-OptiPlex-5090:~$ mysql -h 127.0.0.1 -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 5.7.18-0ubuntu0.17.04.1 (Ubuntu)

Copyright (c) 2000, 2017, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> select * from magento.core_website;
+-----+-----+-----+-----+-----+-----+
| website_id | code | name | sort_order | default_group_id | is_default |
+-----+-----+-----+-----+-----+-----+
| 0 | admin | Admin | 0 | 0 | 0 |
| 1 | base | Main Website | 0 | 1 | 1 |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from magento.core_store_group;
+-----+-----+-----+-----+
| group_id | website_id | name | root_category_id | default_store_id |
+-----+-----+-----+-----+
| 0 | 0 | Default | 0 | 0 |
| 1 | 1 | Main Website Store | 2 | 1 |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from magento.core_store;
+-----+-----+-----+-----+-----+-----+
| store_id | code | website_id | group_id | name | sort_order | is_active |
+-----+-----+-----+-----+-----+-----+
| 0 | admin | 0 | 0 | Admin | 0 | 1 |
| 1 | default | 1 | 1 | Default Store View | 0 | 1 |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> 
```

La hiérarchie des magasins dans la BD de Magento

Pour pouvoir configurer Magento correctement, il faut avoir une bonne vision de son fonctionnement, notamment en ce qui concerne le tunnel de vente et les collecteurs de totaux. En pratique, les différents utilisateurs sont reconnus par leur cookie de session. La session de l'utilisateur se divise en plusieurs namespaces et l'un des namespaces (checkout) contient l'ID du devis (quote) en cours pour l'utilisateur, qu'il soit identifié ou non. En cas de modification du panier (ajout, suppression, modification d'article), mais également en cas d'affichage du panier (checkout/cart), les totaux sont recalculés. A cette fin, les articles sont associés à une adresse de livraison (il n'y en a généralement qu'une), ou de facturation. Le système charge en mémoire les éléments constitutifs du devis, remet les totaux à 0, puis lance des collecteurs de totaux par adresse. Les collecteurs sont lancés successivement, dans un ordre "plus ou moins" précis (l'ordre des collecteurs peut être régi soit par des contraintes, qui indiquent que tel collecteur doit passer avant tel autre, mais il est également possible de fixer leur ordre numériquement et dans ce cas, l'ordre est alors figé). Parmi les collecteurs, on distingue les collecteurs travaillant sur les articles "nominaux" (c'est-à-dire, les articles récurrents), des articles non nominaux, qui sont les articles que l'on rencontre usuellement. Il y a un grand nombre de collecteurs qui sont lancés, plus ou moins spécialisés, mais on peut résumer globalement le fonctionnement à un calcul du prix HT de l'article et de la ligne de commande, puis le calcul des frais de livraisons, l'application des promotions, le calcul des taxes et enfin le calcul du grand total. Connaissant ce fonctionnement interne, on en déduit qu'il est préférable d'insérer les prix en HT et d'appliquer les promotions sur les prix HT, de sorte à éviter des complexités comme le recalcul de l'effet d'une promotion en HT en se basant sur les réglages généraux du magasin pour identifier les taxes qui devraient être appliquées.

La justesse du devis dépend donc également de l'étape dans le tunnel de vente. Celui-ci est usuellement initié par l'affichage du panier, puis le client se logue, si ce n'est déjà fait, les adresses de livraison et de facturation sont renseignées, le moyen de paiement est choisi et les conditions générales de ventes acceptées avant de payer. Après le paiement, une page de confirmation de commande sert de validation terminale.

- Le tunnel de vente peut être personnalisé. Il y a tout d'abord deux types de tunnels :
- Le tunnel onepage, prévu pour les cas clas-

siques, où on n'a qu'une seule adresse de facturation et une seule facture de livraison. Ce tunnel fonctionne à grands renforts de JavaScript et d'AJAX, de sorte qu'effectivement le client ne voit pas de changement de page dans son navigateur ;

- Le tunnel multishipping, qui, lui, passe par plusieurs pages distinctes, avec une page par étape. Ce mode est néanmoins rarement utilisé.

A côté de cela, certaines étapes peuvent être supprimées : par exemple, si la passation des commandes en mode invité est autorisée, alors la nécessité du login n'est plus requise. Cependant, ceci peut avoir des conséquences même au niveau métier, car des commandes passées en mode invité ne sont pas liées à un compte, donc l'utilisateur ne dispose pas d'historique de vente et ne dispose donc pas des facilités liées (par exemple la consultation de ses commandes, voire factures, depuis son espace client, ou la fusion d'une préparation de panier en mode authentifié avec le panier non authentifié au moment du login). Si les produits sont téléchargeables, Magento prévoit même la possibilité de ne pas autoriser de passer en commande invité pour les commandes avec de tels produits. Techniquement, le passage de commandes en mode invité est également assez impactant dans le processus de commandes, car les informations d'objet « customer » ne seront pas utilisables et il faut donc tout stocker dans la session et tout récupérer depuis cette dernière.

Enfin, il est également possible de ne pas imposer l'adhésion explicite aux conditions de ventes. Les étapes nécessaires à la réalisation de l'achat peuvent facilement être allégées et cela peut accroître la conversion de panier en commande pour l'utilisateur. Mais il faut bien comprendre, accepter et prendre en charge les conséquences liées à ce regain de spontanéité.

Dans le cadre de la tirasse, le tunnel de onepage (qui est implémenté sur une seule page avec des appels AJAX) est suffisant, puisqu'il n'y a pas de livraison physique. De plus, aucun mode de livraison n'est requis (produit virtuel). Etant donné que divers aléas peuvent survenir lors d'un achat - toujours au plus mauvais moment - il est plus judicieux de rendre le passage "Création compte client" obligatoire de sorte à pouvoir éventuellement leur proposer le téléchargement de leurs produits virtuels. En ce qui concerne les affichages, nous montrerons aux acheteurs à la fois les montants HT et TTC, pour leur fournir une information complète par rapport

à la valeur fixée (ou tout du moins estimée) du produit. Étant donné que les dons peuvent, en fonction de leur destinataire, donner droit à des réductions fiscales relativement importantes, ce sera déjà une manière de sensibiliser les clients aux problématiques fiscales et éventuellement d'ajuster le don réel qu'ils feront aux œuvres caritatives.

Concernant le moyen de paiement, son implémentation est généralement réalisée par l'ajout d'un module spécifique à Magento et la désactivation des moyens de paiement natifs non souhaités. Généralement, les prestataires de paiement fournissent 3 types d'environnements :

- Un environnement bac à sable, qui est particulièrement utile pour les développeurs pour effectuer la majorité du travail d'adaptation de la solution de paiement à l'existant ;
- Un environnement de préproduction, qui sera utile pour faire des tests de bout en bout et permettra de valider dans le temps le bon fonctionnement du tunnel de paiement complet ;
- Un environnement de production.



A ce propos, le mode de paiement Promesse de don est un mode assez inédit, qui nécessitera un développement spécifique.

Configuration des taxes

La configuration des taxes sous Magento est relativement complexe à mettre en oeuvre, car elle cherche à pouvoir s'adapter à divers besoins métiers. Elle s'inscrit dans une démarche mêlant la configuration des produits, des clients, des zones géographiques et des taux.

Configuration des clients : les clients peuvent être répartis dans des groupes et ces derniers permettent de faire des différences au niveau des prix (par exemple, des règles de promotion peuvent ne s'appliquer qu'à un certain groupe). Deux groupes existent par défaut : le groupe général des clients et le groupe des clients non authentifiés.

Configuration des classes de taxe : les classes de taxe permettent de grouper ensemble des produits qui ont les mêmes caractéristiques fiscales. Il existe un groupe par défaut pour les produits.

A nouveau, une seule classe de taxe suffit pour le projet.

Configuration des calculs de taxe : le premier élément à configurer se trouve dans la section Taxes, groupe Classes de Taxes, champ taxe de classe pour la livraison : positionnement à la

même classe que pour les produits.

Groupe des paramètres de calcul : les prix de catalogue et les prix de livraison seront indiqués en hors taxe, les taxes sur le prix unitaire seront calculées par rapport à l'adresse de livraison et les réductions seront appliquées sur les prix avant le calcul de taxe. Le calcul de taxe se fera sur le prix après réduction. Pour les réductions, elles sont en hors taxe.

Enfin, la destination par défaut est configurée pour le calcul des taxes à la France, dans le prochain groupe.

De la sorte, les taxes sont appliquées le plus tard possible et cela réduit les problématiques d'arrondis de centimes et simplifie les calculs (notamment, il n'y a pas besoin de calculer une estimation du HT à partir du TTC en se basant sur les réglages de Magento).

Configuration des zones de taxes : les produits vendus sont potentiellement livrés à un client. Ce client a une adresse de livraison, ou au minimum de facturation. Ce lieu est utilisé comme référence pour les taxes à appliquer : ainsi, pour chaque taux de taxe applicable est utilisé une zone de taxe, qui inclut le taux de la taxe et la délimitation géographique du lieu.

Il est toujours intéressant de disposer de la configuration TVA au cas où : nous pouvons configurer les différents taux existants en spécifiant le pays France, pour la France métropolitaine.

Configuration des règles de taxes : il existe des classes de taxe produit, des groupes de clients et des zones de taxe : les règles de taxes constituent l'intersection entre ces trois éléments, en autorisant en plus la définition d'un ordre d'application des taxes, ce qui permet si nécessaire de mettre en œuvre des taxes composées (prise en compte de taxes appliquées à un prix pour appliquer d'autres taxes).

Nous supposons avoir besoin d'une règle unique, le produit n'étant destiné qu'au marché français.

Configuration des mails

La configuration des mails est importante, car il s'agit souvent de communications automatiques envoyées à des clients. Au niveau de Magento, deux mécanismes sont mis en jeu dans l'envoi des mails dits "transactionnels" :

- L'identité de l'expéditeur, réglée depuis le BO et stockée dans core_config_data. Celle-ci est également sélectionnable pour les mails transactionnels et cette sélection est également stockée dans core_config_data ;

- Création du modèle de mail qui sera envoyé : un tel modèle de mail contient des champs dynamiques (le nom du client, le numéro de commande, ...), qui seront injectés avant l'envoi du mail. Les mails peuvent être personnalisés via le code (les modèles se trouvent généralement dans « app/locale/<langue>/template/email ») et référencés dans les configurations (fichiers config.xml). Ils peuvent aussi éventuellement être personnalisés par la suite depuis le BO et stockés par le SGBD. La sélection du modèle de mail sera également conservée dans core_config_data. La difficulté de la configuration des mails est surtout liée à la création de bons modèles pour les mails, dans un objectif de maintenance facilitée. Il est intéressant de chercher à isoler les en-têtes et les pieds des mails pour les réutiliser dans les différents modèles. Et à côté de cette rigueur de conception, il y a également la difficulté inhérente au contenu métier du mail.

toConnect un thème gratuit et libre de droits pour adapter uniquement le (S)CSS..

LE CATALOGUE

La notion d'EAV

Entity Attribute Value, ou les structures génériques dans Magento

Le catalogue réunit en pratique deux éléments : la gestion des produits et la gestion des catégories. Les produits, tout comme les catégories, ont sous Magento une architecture un peu spéciale, appelée EAV (Entity Attribute Value). L'idée est de pouvoir dynamiquement ajouter des attributs à un objet sans toutefois modifier le schéma de BD. L'ensemble des types EAV, de même que les attributs disponibles pour chaque type est défini dans des tables spécifiques. Ces informations permettent de stocker les valeurs des attributs dans des tables de stockage. Ainsi, pour un produit, on aura par exemple la table catalog_product_entity, qui contiendra certaines informations de base nécessaires à identifier l'objet, puis des tables comme « catalog_product_entity_varchar », « catalog_product_entity_int », etc. qui vont stocker l'ID de l'attribut, le scope de l'attribut, l'ID du produit et finalement la valeur de l'attribut. Les attributs sont classables en groupes et en sets d'attributs, ce qui va influencer l'affichage des attributs, surtout dans les produits. En fonction des types d'entité, des informations supplémentaires concernant les attributs peuvent être requises, par exemple pour savoir si l'attribut peut être affiché en FO, ou si l'attribut peut être utilisé dans les comparaisons de produits : à cet effet, il existe des tables supplémentaires pour stocker ces informations.

Les catégories sont également implémentées en EAV. Néanmoins, il faut bien voir que pour récupérer des propriétés d'un objet de cette manière, il est souvent nécessaire de faire des requêtes contenant un grand nombre de jointures, ce qui nuit aux performances. C'est pour cela que Magento dispose également d'une possibilité d'index plat sur certains objets EAV : il s'agira de créer une table qui va contenir tout ou partie des valeurs des propriétés de l'objet et ce par store.

Tous les produits ne se ressemblent pas : choisir ou créer son type

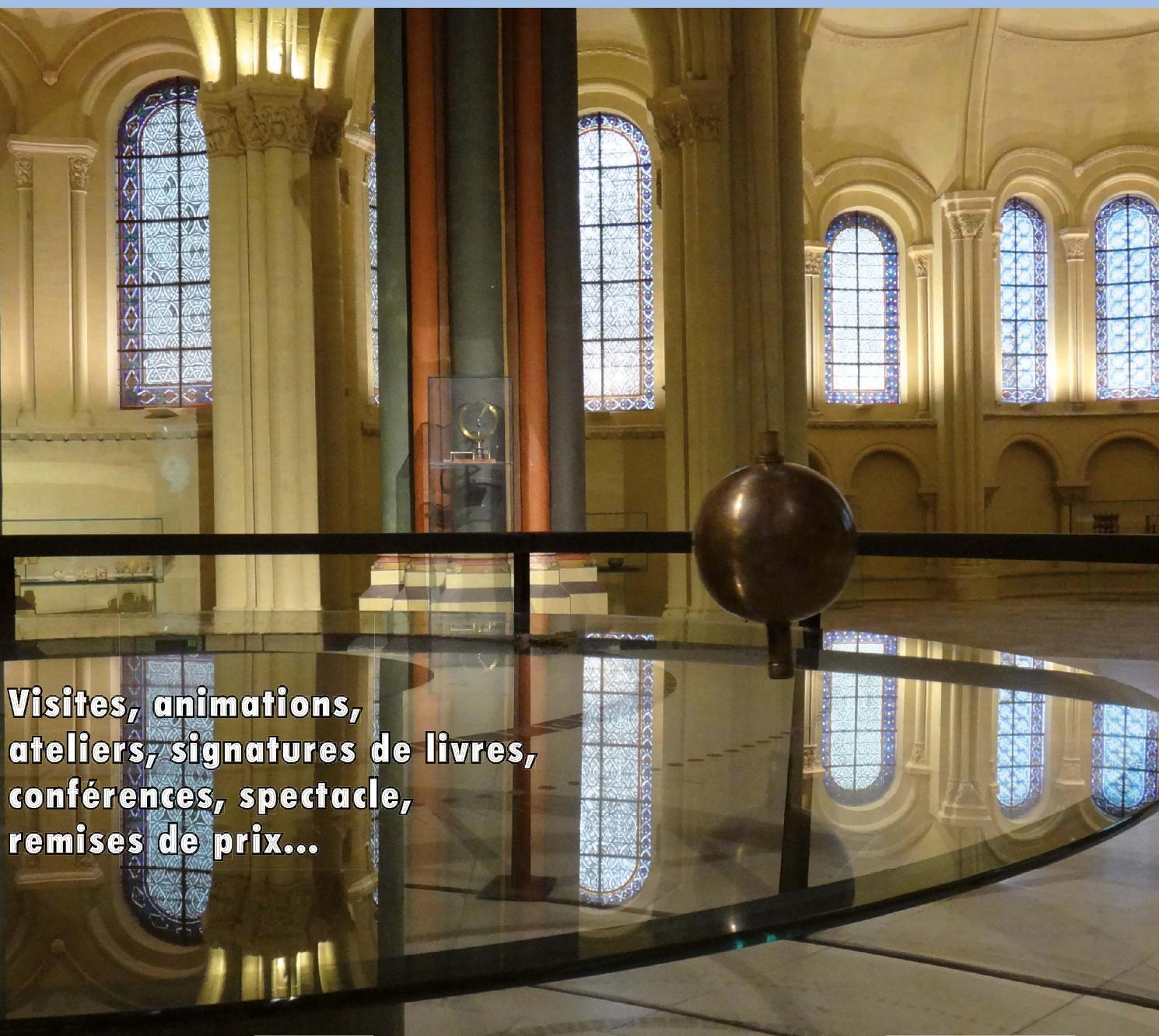
En plus de la notion d'attributs, les produits disposent d'une notion de type. Le type du produit influe sur certaines opérations réalisables sur le produit, opérations qui sont implémentées notamment dans la classe représentant le produit. En pratique, sous Magento, on dispose d'un certain nombre de types de produits :

Pour la tirasse, nous chercherons sur Magen-

LES 30 ANS DE

tangente

Dimanche 3 décembre au Musée des arts et métiers



Visites, animations,
ateliers, signatures de livres,
conférences, spectacle,
remises de prix...

Le Musée sera ouvert gratuitement de 10h à 18h.
60 rue Réaumur - Paris 3^e - www.arts-et-metiers.net

Club
tangente

musée
des arts et métiers
le cnam !

- Simple : c'est un produit quelconque du type le plus souvent utilisé ;
- Configurable : il permet de présenter une série de produits représentant des déclinaisons d'un même produit. L'exemple souvent vu est au niveau vestimentaire un T-Shirt qui est proposé en plusieurs coloris et tailles. Chaque combinaison coloris/taille pourra être intégrée sous forme d'un produit simple et on laissera choisir la combinaison souhaitée à l'utilisateur au niveau du configurable ;
- Groupé : on décide de vendre certains produits ensemble, dans une configuration "fixe" ;
- Bundle : on décide de vendre certains produits ensemble, mais l'utilisateur a une certaine latitude de décision dans la composition du bundle ;
- Virtuel : des produits qui n'ont pas de tenant physique ;
- Téléchargeables : produits numériques.

En fonction du type de produit, il est possible d'ajouter au produit des options qui pourront être configurées par le client. Il est fondamental de choisir un type de produit approprié pour représenter les entités qu'on veut vendre, et, si nécessaire, créer son propre type de produit, son ou ses propres jeux d'attributs, ainsi que les attributs qui décrivent le produit. Au niveau de la création de ces éléments, tout comme la création de produits, trois approches théoriques sont envisageables :

- Soit créer les éléments manuellement dans une base de données de production, puis restaurer un dump de production en pré-production et fournir le dump aux développeurs pour qu'ils puissent avoir une base à jour : c'est faisable si les adaptations nécessaires sont réalisables directement en Back Office ;
- Soit créer les éléments via des installateurs depuis le code et c'est la mise à jour du code qui déclenchera la création des éléments dans les différents environnements, allant de l'environnement de développement jusqu'à l'environnement de production ;

Une autre possibilité est la création d'un système d'importation de produits.

Bien entendu, comme la tirasse est téléchargeable, nous utiliserons le produit téléchargeable comme type de produit. Et comme il s'agira d'un nouveau site, nous mettrons directement en production une base de données pré-configurée.

La création de produit nécessite l'indication d'un certain nombre d'informations, comme le SKU du produit, son prix, mais aussi la description courte et la description normale. En ce qui

concerne les descriptions, il faut garder à l'esprit qu'elles sont essentielles, car un effort de rédaction pourra faciliter le référencement naturel de la page, d'une part et convaincre les acheteurs d'autre part. Il ne faut donc pas sous-estimer le temps nécessaire pour bien renseigner une fiche produit. Il est d'ailleurs même possible de différencier les pages produits les unes des autres par des présentations spécifiques, en agissant sur le layout d'une page produit.



Par ailleurs pour un produit tel que la tirasse, des vidéos peuvent également être très utiles, pour démontrer l'utilisation du produit ; une inclusion de vidéo peut être effectuée via de la saisie de code HTML (par exemple, un iframe vers youtube).

La fiche produit comporte également une gestion d'inventaire. Il faut prendre garde à bien configurer ce point, car une mauvaise configuration pourrait rendre simplement l'article non vendable. De même, il faut également faire attention au statut (activé, désactivé) et à la visibilité du produit, le produit pouvant être visible ou non dans une recherche ou par le catalogue. Enfin, la fiche produit permet également de définir des relations avec d'autres produits, comme des produits connexes ou de substitution. Pour les curieux : il y a d'autres façons de présenter des produits ... On peut aussi chercher à faire des modèles 3D, par exemple via la photogrammétrie : <https://blog.netapsys.fr/photogrammetrie-utilisation-possible-avec-magento/>

Les catégories et la navigation

Les catégories permettent de regrouper les produits et offrent de la sorte un classement des produits qui sera utilisé par la navigation. Ainsi, les catégories forment une arborescence depuis une catégorie racine (non affichée en Backoffice) et nous pouvons définir pour chaque groupe de stores la catégorie racine qui va permettre la navigation dans le site et qui sera également utilisée pour générer le sitemap. Chaque produit peut être dans aucune, une ou plusieurs catégories.

En ce qui concerne la navigation, les catégories (tout comme les produits et les pages CMS d'ailleurs) peuvent se voir attribuer une URL spécialisée, qui est plus parlante pour l'utilisateur (et pour le moteur de recherche) : il s'agit du mécanisme d'URL rewriting, qui est un mécanisme pour identifier les pages spécifiques, de sorte à ne pas être générée par la forme "canonique" des URL Magento

(routeur/controller/action/, puis nom et valeur de paramètres). De même, il est possible d'activer sur les catégories le filtrage des produits selon des critères définis par l'utilisateur, ce qui peut permettre dans certains cas à un utilisateur d'accéder plus rapidement au produit recherché.



Comme indiqué au début de l'article, les catégories pourront servir pour présenter la tirasse selon plusieurs angles de vue.

La recherche

Magento intègre une possibilité de recherche se basant sur les attributs qui sont marqués "utilisable" dans une recherche. L'utilisateur peut faire des recherches précises en se basant sur ces attributs et les produits qui en résultent sont alors affichés. Une recherche rapide est également disponible, où l'utilisateur indique uniquement le texte qu'il recherche. Ces recherches servent à présenter des résultats qui sont des produits et le type de recherche est soit un like, soit une recherche fulltext, ce qui signifie que ces recherches doivent rester relativement simples et n'offrent pas les possibilités d'un moteur Lucene par exemple. C'est pour cela que la version Enterprise de Magento permet également d'utiliser une instance SOLR et permettrait par exemple des recherches avec des termes pondérés. On notera que d'autres approches sont éventuellement envisageables, puisqu'une implémentation d'un moteur Lucene est déjà présente dans les bibliothèques Zend. D'autre part, des possibilités d'utiliser des moteurs de recherche externes existent (Google Custom Search par exemple).

Dans le cadre de la tirasse - produit unique s'il en est - les recherches peuvent porter sur des éléments précis dans les pages HTML mais aussi surtout dans des documents annexes, par exemple dans des fichiers PDF de documentation. De ce fait, il semble opportun de considérer des éventuelles évolutions du site e-commerce pour interroger un moteur externe qui indexera non seulement l'ensemble des pages produits, mais également les pages CMS et les documentations. La mise en place d'un tel moteur de recherche dépassera donc le cadre strict d'une boutique e-commerce.

Dans la partie 2 de ce dossier nous verrons la vie quotidienne de notre site e-Commerce, comment l'intégrer à notre infrastructure existante et comment assurer sa maintenance dans la durée. •

Rendez-vous dans Programmez ! 212.

Pourquoi C++ en 2017 ?



Pourquoi utiliser C++ en 2017 ? La question est posée... C++ est un descendant de C, regardez : fig. 1.

Christophe PICHAUD
Consultant sur les technologies Microsoft
christophepichaud@hotmail.com
www.windowscpp.net

Pour rappel, tous les logiciels que j'utilise sur mon PC portable sont écrits en C/C++ :

- Windows et ses 3400 dlls ;
- Explorer, l'explorateur de fichiers ;
- Le gestionnaire de tâches ;
- MS Paint ;
- Mon lecteur multimédia ;
- Les navigateurs Web (Chrome, Firefox, Internet Explorer, Edge) ;
- La suite Office (Word, Excel, PowerPoint, Outlook, etc.) ;
- Notepad++ ;
- Acrobat Reader ;
- Mon vieux MSDN Library 2008 SP1.

La seule exception que j'ai sur mon PC est Visual Studio qui est une application hybride C/C++ .NET COM.

Le PC portable que j'utilise pour écrire cet article est un portable low-cost acheté en 2015 :

- Processeur Intel Celeron Quad Core ;
- Intel HD Graphics ;
- 4 GB RAM ;
- 500 GB Disk.

Et pourtant avec un PC low-cost peu puissant, je peux toujours utiliser en même temps Windows, un browser, Word, MS Paint, Visual Studio et écouter de la musique en arrière-plan.

Et je n'ai pas de soucis de performance ; la preuve voici le gestionnaire de tâche : [2]

Bien sûr, ce PC ne fera pas de montage vidéo

ou de la compression diverse et variée... Mais par contre, pour une utilisation standard, il me permet de travailler.

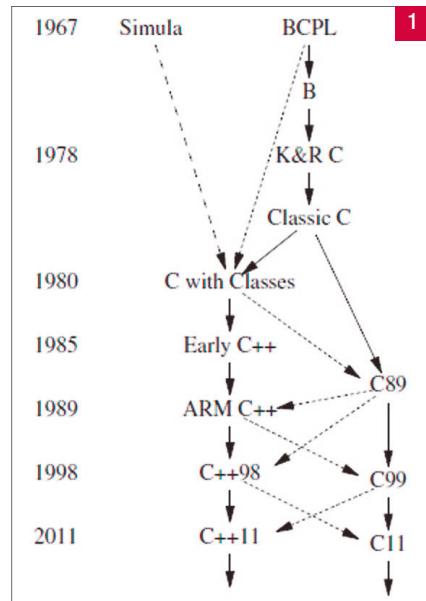
Anatomie des applications

Les applications que je lance consomment peu de mémoire et sont réactives. Pourquoi ? Parce qu'elles utilisent directement les APIs du système d'exploitation qu'est Windows. Les APIs Windows sont utilisables depuis un simple compilateur C/C++. Il suffit de faire `#include <windows.h>` et vous avez accès à des milliers de fonctions. Chaque édition de Visual Studio distribue le SDK Windows. Les applications consomment peu de mémoire et ça c'est la clé de la réussite.

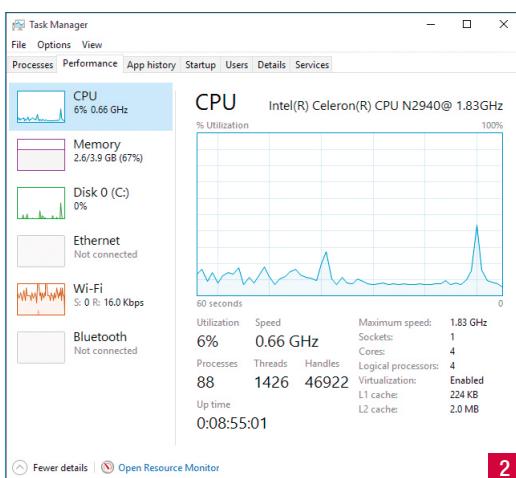
Mythes et réalités

Vous allez me dire, ouais mais faire une application en C++ *from scratch*, c'est difficile et ce n'est pas productif comme langage le C/C++ hein ? Je réponds de faire attention au discours du marketing et aux préjugés... [3]

En effet, pour développer une application, Microsoft préconise de faire du C#, VB.NET ou JavaScript. Oui mais bon... la question est la suivante : pourquoi Microsoft ne fait pas ses applications en .NET ? On nous avait dit que Microsoft pratiquait le *dog-fooding* : c'est-à-dire que lorsque MS sort une technologie, MS la réutilise dans ses produits. Ah oui, mais c'est un



peu plus complexe que ça. Pour Microsoft, le C++ c'est l'électricité, c'est naturel. Pour le marketing, .NET c'est limpide. Donc, qui croire ? Si vous voulez faire une application qui ne fonctionne que sous Windows, oui .NET peut-être une alternative mais attention, il faut redistribuer le framework .NET. Vous allez me dire oui mais avec .NET Standard 2.0, on peut compiler sous Windows, Mac et Linux. Je vais vous la faire courte... Il y a 10 ans, il existait un package nommé Mono qui permettait de faire du C# sous Linux. Ça a été un flop. Maintenant que Microsoft a racheté la société Xamarin et que .NET Standard 2.0 est sorti, le marketing nous (re)envoie ses sirènes pour faire des applications mobiles et desktop via .NET et C# sur plusieurs plateformes... Méfiez-vous des effets d'annonces car ce n'est pas aussi simple que cela est énoncé. Faites-vous votre propre expérience. En C/C++ sous Windows, il suffit de redistribuer MSVCRest.exe qui fait 10 MB. Ce package contient la runtime du C, la STL, les DLLs des



C++, mythes et réalités

principes et techniques

- pointers**
 - Un langage orienté objet compliqué !
 - Des macros et de goto !
 - Bas niveau !
 - Non sécurisée, dangereux !
 - C'est du C !
 - new / delete
 - Un bon code
- gestion des erreurs**
 - Code crado mais je continue sur le même modèle...
 - Orienté objet
 - Peu productif !
 - Interfaces riches
 - Template meta-programming
 - Code dur à maintenir
 - le style C++ 11

MFC. De plus, Windows contient déjà plusieurs versions de ces modules.

Pour faire une application de bon standing, il faut être capable de fournir une interface graphique élégante, des librairies qui consomment peu de mémoire et des opérations qui ne pous- sent pas le processeur dans ses derniers retran- chements. Et là, c'est le principe des couches. Si vous développez avec un runtime et un frame- work, il y aura plusieurs couches à traverser avant que votre code ne rejoigne les APIs du système d'exploitation ou les instructions de votre processeur. La mode est de préserver la batterie, de soulager les disques, et de consom- mer peu de mémoire. Seul le C/C++ vous per- met cela. Au niveau du langage C, on trouve les fondamentaux suivants :

- Gestion de la mémoire avec les fonctions malloc/free ;
- Gestion des I/O : stdio.h et les fonctions open, close, read, write, fopen, fprintf, fread, fwrite ;
- Gestion des pointeurs int * ptr ;
- Etc.

Le C++ fournit un niveau plus abstrait via la STL (Standard Template Library) qui gère :

- La gestion automatique de la mémoire : on évite new/delete via unique_ptr<T> et shared_ptr<T> ;
- Gestion des strings et des buffers avec string, wstring et les streams ;
- Gestion des containers (array, vector, list, map, set, etc.) ;
- Des algorithmes ;
- Etc.

Au lieu de présenter mon PC portable low-cost, je pourrais vous parler des distributions Linux qui tournent sur de vieilles machines. Pourquoi ? Tout est fait en C/C++ à 99%. Bref, on positionne le curseur sur les éléments suivants :

- Gérer la batterie ;
- Gérer la mémoire ;
- Gérer le processeur.

Avec les langages dit productifs, vous n'avez aucun moyen d'optimiser ces 3 facteurs car vous êtes dans les hautes sphères d'un runtime (CLR ou JRE) qui fait plein de choses tout

seul... Sur un poste de bureau, on peut cacher cela, mais sur un mobile, la sanction est immé- diate. Si l'application sucre la batterie ou si une application passe son temps à chauffer la bat- terie, ce n'est pas bon et l'application n'aura pas un grand succès.

Le C/C++ est portable et natif

Sur chaque système (Windows, Linux, Mac), il existe un compilateur C/C++. Il est ainsi pos- sible de faire du code portable. Bien sûr, le code qui utilise les API Windows ne compilera pas sur Linux mais le code métier, les classes Poco et les libs *third-party* compilent. L'avantage du C/C++ est que le système d'exploitation est fait avec du C. Les autres APIs des librairies *third- party* sont aussi faites en C/C++. Donc quel est cet avantage ? Sous Windows, vous voulez envoyer par exemple des messages cryptés ; rien ne vous interdit de récupérer une librairie de chiffrement sur Linux, de la recompiler, et de l'utiliser sous Windows. Les combinaisons sont infinies.

De plus, les compilateurs C/C++ sont gratuits.

La documentation

Si vous voulez débuter, il existe une ressource terrible : MSDN Library 2008 SP1. Cet ensemble d'articles contient toutes les res- sources pour développer sous Windows. On y trouve aussi des articles sur le C/C++ et les MFC.

Téléchargez ce fichier ISO de 2.8 GB depuis le Microsoft Download Center et installez-le. [4]

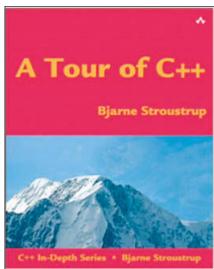
Il existe aussi une version plus récente de la documentation Microsoft. Avec Visual Studio, il existe une application nommée *Help Viewer* et dans laquelle il est possible de télécharger dif- férentes sections de l'aide (.NET, Visual C++, SQL Server, etc). [5]

L'IDE

Là, le choix se porte sur Visual Studio. VS exis- te en version gratuite dite *Community Edition*. Cette version est le produit d'appel vers la version payante de VS comme VS Pro. Le coût du produit est environ 500 .

Visual Studio est très confortable mais il faut bien comprendre que ce produit contient moult fonctionnalités. Le problème de Visual Studio est qu'il est de plus en plus lourd et donc il faut savoir patienter si votre machine de dévello- pement n'est pas très rapide... Bref, vous êtes prévenu ! [6]

Il est possible aussi d'utiliser VS Code mais ce n'est pas pareil...



Par où commencer ?

Le premier pas est d'apprendre le langage dans sa mouture C++ 11 ou C++ Moderne. Un bon ouvrage, écrit par le créateur du langage, est disponible depuis 2014 : *A Tour of C++* par Bjarne Stroustrup.

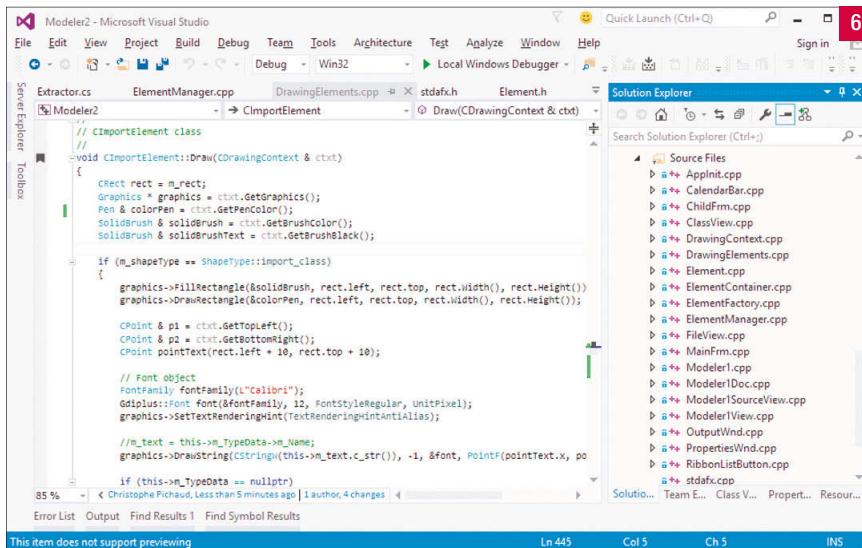
La STL (Standard Template Library)

La STL est la bibliothèque du C++. Elle contient de nombreux fichiers d'en-têtes. L'évolution du compilateur est indissociable de l'évolution de la STL. La STL fournit des classes templates pour satisfaire tous les besoins principaux : cela va des algorithmes aux containers, des strings aux threads, etc. Voici les fichiers d'en-têtes : [7]

Les classes

Il est possible de créer des types concrets ou abstraits. Pour créer une classe abstraite, il suffit de déclarer une méthode virtuelle pure. L'héritage peut être simple ou multiple.

<algorithm>	copy(), find(), sort()	7
<array>	array	
<chrono>	duration, time_point	
<cmath>	sqrt(), pow()	
<complex>	complex, sqrt(), pow()	
<forward_list>	forward_list	
<iostream>	fstream, ifstream, ofstream	
<future>	future, promise	
<ios>	hex, dec, scientific, fixed, defaultfloat	
<iostream>	istream, ostream, cin, cout	
<map>	map, multimap	
<memory>	unique_ptr, shared_ptr, allocator	
<random>	default_random_engine, normal_distribution	
<regex>	regex, smatch	
<string>	string, basic_string	
<set>	set, multiset	
<sstream>	istringstream, ostringstream	
<stdexcept>	length_error, out_of_range, runtime_error	
<thread>	thread	
<unordered_map>	unordered_map, unordered_multimap	
<utility>	move(), swap(), pair	
<vector>	vector	



Les templates

Un template est une classe qui prend en paramètre un ou plusieurs types. Le template est évalué à la compilation. Il existe aussi des fonctions template.

Le C++ Moderne

Voici les principaux éléments du C++ moderne :

- Utilisation de la stack plutôt que le heap ;
- Utilisation de auto pour masquer le type réel ;
- Utilisation des smart pointers à la place des pointeurs normaux ;
- Utilisation des types std::string ou std::wstring à la place de char[] ;
- Utilisation des containers de la STL vector, list et map à la place des structures customs ;
- Utilisation des algorithmes de la STL à la place des routines customs ;
- Utilisation des exceptions pour remonter des erreurs ;
- Utilisation du type STL std::atomic< > à la place des mécanismes IPC custom ;
- Utilisation des lambdas *inline* au lieu des petites fonctions déclarées séparément ;
- Utilisation du range for pour le parcours des tableaux et autres containers ;

L'utilisation du mot-clé auto permet de masquer des types complexes à utiliser. Ainsi pour déclarer un itérateur sur un vector de shared_ptr< T >, il ne faut plus écrire :

```
vector<shared_ptr< MyClass >>::const_iterator it = v.begin();
```

A la place on écrit ça :

```
auto it = v.begin();
```

Avec le C++ moderne, il n'y pas besoin d'utiliser les mécanismes de new/delete pour allouer ou libérer de la mémoire. On utilise, à la place, des smart pointers comme unique_ptr< T > ou

shared_ptr< T > qui savent libérer la mémoire automatiquement.

Le développement multi-plateformes

Revenons sur deux définitions qui sont galvaudées : le natif et le multi-plateformes. Quand on parle de développement natif, on parle de développement qui utilise le même langage que celui qui a permis de faire le système d'exploitation. Concrètement c'est du C/C++. Sous Windows, Linux et Mac, c'est comme ça. Ça veut dire que l'on utilise directement les APIs du système d'exploitation qui sont exploitées dans les entêtes .h fournies par le SDK de l'OS. Pour un programme Windows, cela veut dire qu'il charge kernel32.dll, user32.dll et gdi32.dll. A partir du moment où vous utilisez un langage comme C# qui contient un garbage collector et utilise un runtime, vous n'êtes pas dans le natif.

Le développement hybride C++ /CLR

Sous Windows, il est possible de faire du code C++ et du code qui appelle des classes .NET via le C++/CLI. Pour faire cela, il faut que le projet supporte l'option *Common Language Runtime Support /clr*. Il est ainsi possible de faire des ajouts de référence à des assemblies .NET dans un projet C++. Si vous avez du code .NET, vous pouvez le réutiliser et cela se fait très facilement. L'avantage c'est que si vous avez des besoins XML ou ADO.NET, vous aurez de suite le support du .NET Framework à portée de main.

Framework graphique

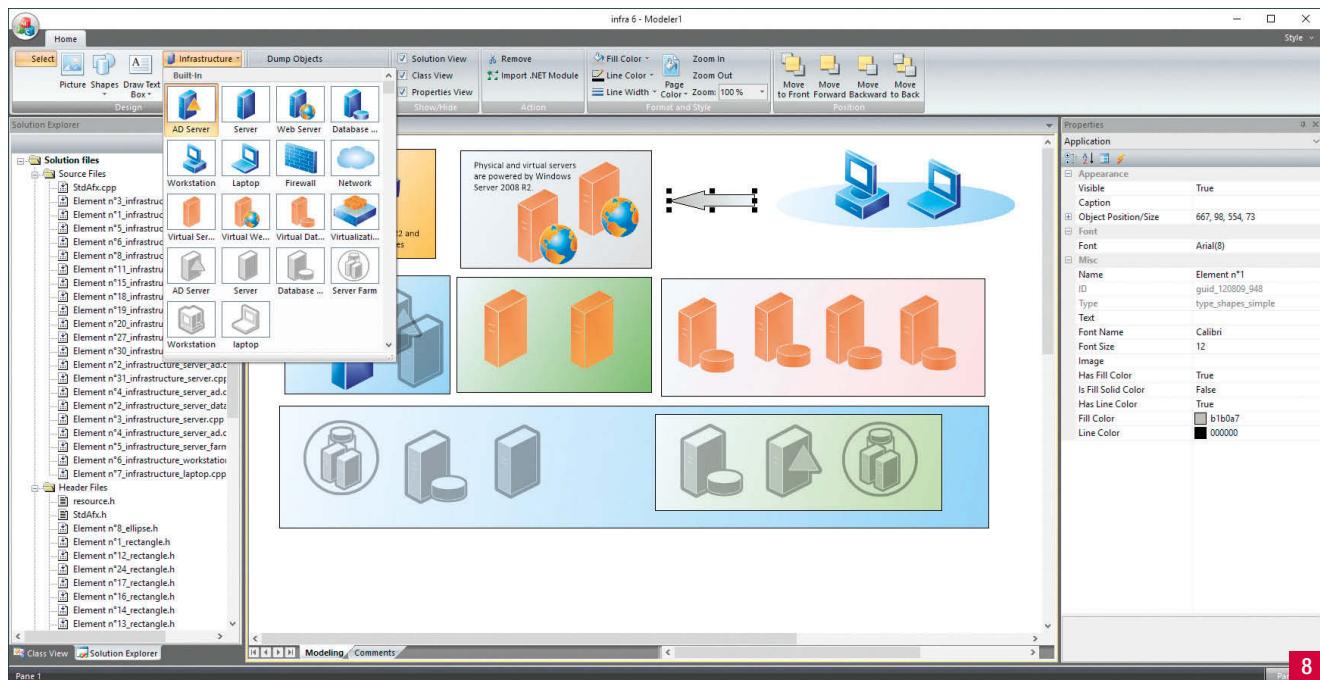
Pour développer une application graphique, il faut avoir des APIs pour afficher des contrôles, des fenêtres, des menus, des boîtes de dialogue, etc. Chaque système (Windows, Linux, Mac) fournit des APIs pour cela. Il existe aussi différents Frameworks pour arriver à faire des applications fenêtrées :

- GTK
- Qt
- MFC
- WTL
- AppKit, Cocoa
- WxWidgets

Sous Windows, il y a 3 options :

- GDI32 : API C historique de gestion des fenêtres ;
- MFC : Microsoft Foundation Classes ;
- WTL : Windows Template Library.

Personnellement, j'ai commencé avec GDI32 pour apprendre les bases. De nos jours, je suis un adepte des MFC car je suis très productif



avec. Le framework fournit les bases des applications modernes à la sortie du Wizard. Avec GDI32, il faut tout coder et cela prend du temps. De plus, MFC fournit des contrôles sophistiqués comme les docking pane, les properties grid, le Ribbon, les MDI avec un modèle document-vue ; c'est un peu différent du MVC mais les principes sont les mêmes.

Voici le type d'applications que l'on peut faire facilement avec les MFC : [8]

Cette application possède tous les contrôles graphiques qui sont sexy et fonctionnels.

Le Back-end en C++

Il est possible de construire la partie serveur d'une application via plusieurs technologies. Il est possible de construire :

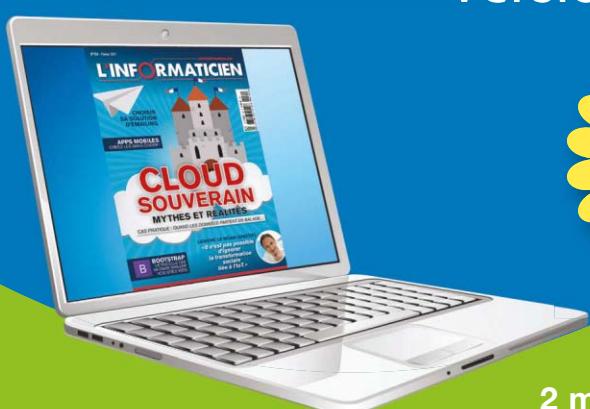
- Un service Windows avec des IPC ;
- Un serveur XML-RPC custom ;
- Un serveur Web API qui retourne du JSON via le REST SDK.

L'avantage du REST SDK, c'est qu'il peut retourner des data au format JSON pour un front qui n'est pas forcément en C++. Le serveur C++ fait avec le REST SDK fera l'objet d'un article séparé.

CONCLUSION

Sous Windows, le compilateur Visual C++ fête ses 25 ans ! Il n'a jamais été autant surpuissant. Le C++ moderne ouvre de grandes possibilités car le code est plus simple à écrire, plus lisible et plus performant. La STL a de nombreuses fonctionnalités à utiliser comme les containers, les smart pointeurs. Bref, le langage est très abordable même si vous avez des bases de Java ou C# car ceux-ci ne sont que des descendants du C++... •

L'INFORMATICIEN + PROGRAMMEZ ! versions numériques



2 magazines mensuels
22 parutions / an+ accès aux archives PDF

PRIX NORMAL POUR UN AN : 69 €
POUR VOUS : 49 € SEULEMENT*

* Prix TTC incluant 1,01€ de TVA (à 2,10%).

Souscription sur www.programmez.com

C++ Builder : multiplateforme par nature

- Marion Candau, **MVP Embarcadero**
- Bernard Roussely

Un développeur a aujourd’hui le choix entre de nombreux langages et environnements de programmation pour effectuer ses tâches. Au-delà des effets de mode, le C et le C++ restent des langages très importants depuis 45 ans pour le premier, et 25 ans pour le second.



Builder permet de compiler tout programme C ou C++, parfois au prix que quelques adaptations. Il offre en plus la possibilité de générer des exécutables pour les plateformes Windows, macOS, iOS, Android et, dans certains cas, Linux. Nous montrons ici avec un exemple simple comment développer et générer une application qui fonctionne sur les quatre premières plateformes citées.

Pour parvenir au résultat, nous utilisons l’environnement graphique « FireMonkey » ou FMX ainsi que les composants « mobiles » de C++ Builder(1). C++ Builder propose deux « frameworks », la Visual Component Library (VCL pour bibliothèque de composants visuels), en pratique une surcouche de l’API Windows qui ne fonctionne que sur Windows, et FMX, conçue pour le multiplateforme. Le même code FMX compilera sur les différentes plateformes, bien que le rendu visuel diffère compte tenu des spécificités de chaque cible.

Prérequis

Il faut d’abord s’assurer de la présence des SDK Android, mac OS et iOS dans « Outils->Options->Gestionnaire de SDK » et les installer selon les procédures du Wiki RAD Studio si nécessaire(2) et (3). Il faut ensuite vérifier les connexions à une machine Android et à un Mac, auquel une machine sous iOS doit être connectée le cas échéant. Android doit être mis en mode « développeur » (« Paramètres->Plus->Options de développement ») et le Mac doit être équipé de XCode (IDE d’Apple). Le Mac doit en plus être équipé de PAserver, livré avec C++ Builder(4) et autoriser la connexion réseau depuis la machine équipée de C++ Builder.

Le projet

Nous allons construire une application qui affiche des pavés numériques fixes, d’une part, et aléatoires, d’autre part. Ce projet comprend trois fiches graphiques ou « forms » dans le langage Embarcadero.

Les étapes du projet sont les suivantes :

- Créez un nouveau projet C++ multiplateformes dans « Fichier->Nouveau->Application multi-périphérique » (choisir « Application vide ») ;
- Ajoutez les objets graphiques : 2 TButton, 1 TEdit, 1 TImage (facultatif) ;
- Sauvegardez le projet (SecurityFormsProject dans le ZIP) ;
- Ajoutez deux autres fiches multi-périphériques et sauvegardez-les sous les noms de FixedPad et RandomPad ;

- Dans le panneau en haut à droite, cliquez sur l’onglet « Plateformes cibles » et ajoutez les toutes une à une (le type de votre machine Android apparaît à côté de l’icône « téléphone » si elle est connectée et que la cible Android est choisie) ;

• Compilez pour valider que tout va bien avant de continuer. Ajoutez ensuite 12 TImage de 100 x 100 dans un TPanel sur FixedPad ainsi que 8 TRadioButton dans un deuxième TPanel. De même, ajoutez 25 TImage de 100 x 100 dans un TPanel et les mêmes TRadioButton dans un TPanel sur RandomPad. Enfin, un TButton vient compléter les composants graphiques de RandomPad.

Les composants placés sur la vue principale peuvent ensuite être repositionnés sur des vues Android, mac OS et iOS dérivées, en fonction de la taille de l’écran de la cible.

Le code

Nous allons charger des images dans chaque TImage à partir de fichiers stockés dans un répertoire appelé « Graphix ». La seule difficulté est de placer ce répertoire au bon endroit sur chaque plateforme. Il est possible que vous ayez à faire des adaptations en fonction de la plateforme Android que vous utilisez (5).

Pour s’assurer du bon fonctionnement sur chaque cible, nous utilisons des macros fournies par Embarcadero (6) : __Windows, __APPLE__ et __ANDROID__

```
void __fastcall TFixedPadForm::FormCreate(TObject *Sender)
{
#ifdef __Windows
    MyIconPath = ".\Graphix\DarkBlueSquare\";
#elif __APPLE__ et __ANDROID__
    MyIconPath = lutils::TPath::GetDocumentsPath0 + "/Graphix/DarkBlueSquare/";
#endif
    AssignImages0;
}
```

Ce code est repris et dupliqué pour la fonction qui gère les clics pour chaque bouton. Il est possible de faire une seule fonction qui traite les clics de tous les boutons pour réduire (un peu) la taille globale du code. Le code de SecurityForms est trivial et ne consiste qu’à afficher les fiches

(1) Il faut avoir une version C++ Builder Entreprise avec le pack « mobile » au moins pour pouvoir reproduire l’exemple.

(2) <http://docwiki.embarcadero.com/RADStudio/Berlin/en/Android>

(3) http://docwiki.embarcadero.com/RADStudio/Berlin/en/OS_X_Application_Development

(4) C:\Program Files (x86)\Embarcadero\Studio\19.0\PAserver

(5) http://docwiki.embarcadero.com/RADStudio/Seattle/en/Standard RTL Path Functions_across_the_Supported_Target_Platforms

(6) http://docwiki.embarcadero.com/RADStudio/Tokyo/en/Predefined_Macros

FixedPad et RandomPad.

Le code de FixedPad consiste à faire changer les images qui s'affichent pour chaque clic (ou appui dans le cas des plateformes mobiles) sur un TRadioButton.

Le code de RandomPad est plus compliqué car l'accès à un générateur de nombres aléatoires varie d'une plateforme à l'autre et on utilise respectivement l'API cryptographique de Windows et /dev/urandom pour OSX, iOS et Android, avec des variantes mineures.

```
void __fastcall TRandomPadForm::AssignImages()
{
int MyRandomNumber, i;
TImage *MyImage;
UnicodeString ImageId;
bool MyBooleanList[25] = { false};

// generate a new random sequence
for(i = 0; i < 25; i++) {
    do {
        MyRandomNumber = RandomInt();
        MyRandomNumber = MyRandomNumber
            - (MyRandomNumber / 25) * 25;
    } while(MyBooleanList[MyRandomNumber] == true);
    ImageId = L"Image" + IntToStr(i + 1);
    MyImage = dynamic_cast<TImage *>
        (RandomPadForm->FindComponent(ImageId));
    if(MyRandomNumber < 10) {
        MyImage->Bitmap->LoadFromFile(MyIconPath +
            IntToStr(MyRandomNumber) + ".png");
    }
    else {

```

```
    MyImage->Bitmap = (TBitmap *)NULL;
    }
    MyBooleanList[MyRandomNumber] = true;
}
}
```

Les composants graphiques sont les mêmes pour toutes les plateformes, mais les traitements doivent être ajustés pour tenir compte de la structure de fichiers, des emplacements autorisés en écriture, et des spécificités des systèmes d'exploitation et des périphériques supportés. Cependant, le recours à C++ Builder permet de réduire les écarts entre plateformes grâce à la disponibilité de fonctions (hors fonctions graphiques) portables, au-delà de la bibliothèque C standard.

Le résultat

FixedPad permet d'afficher les vues suivantes (Windows et Android) :

[1] et [2]

RandomPad permet d'afficher les vues suivantes (Windows et Android) :

[3] et [4]

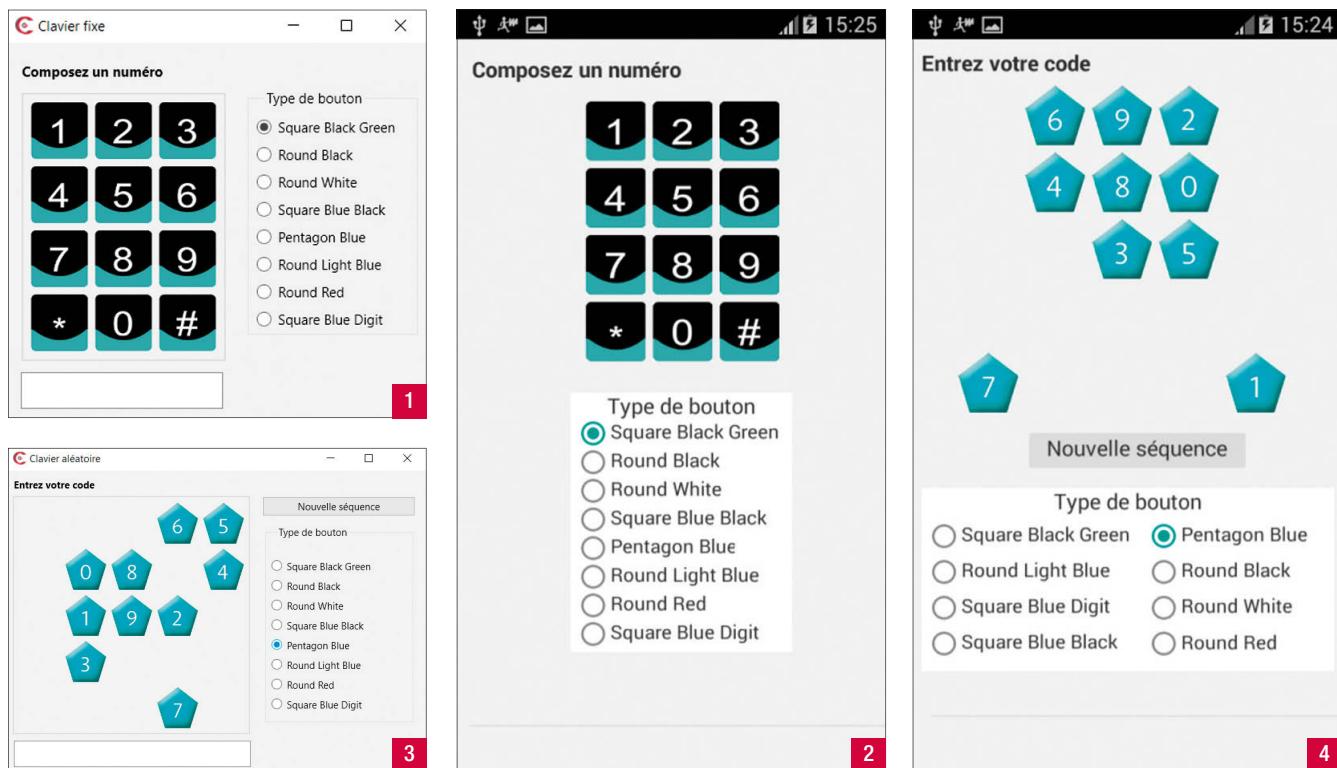
On peut donc se servir de FixedPad et de RandomPad avec quelques évolutions pour saisir des codes fournis par un utilisateur, par exemple.

Conclusion

Nous avons montré sur un exemple simple comment créer une application multi-plateformes avec C++ Builder. Cette application utilise des macros prédéfinies pour ajuster les traitements à la cible ainsi que différentes vues FMX dérivées d'une vue principale qui contient l'ensemble des composants graphiques.

Remerciements

Clementine Roussely pour la conception des images pour les boutons •



Apple vs Google / ARKit vs ARCore : la guerre de la réalité augmentée est ouverte



Google pouvait-il raisonnablement laisser Apple et son ARKit sans réponse concrète ? Il faut dire que depuis la présentation de cette couche de réalité augmentée en juin dernier, les développeurs se sont jetés dessus et surtout, aucune réaction des concurrents directs n'est venue sauf Samsung avec des pubs sur le sujet. Bon ok, c'était en bêta, avec iOS 11, et aucun matériel optimisé n'était présenté côté iPhone ou iPad. Mais avec la disponibilité d'iOS 11 et des nouveaux iPhone, la situation change.

• François Tonic

ARCore est clairement la réponse directe et frontale de Google à Apple. Objectif simple : proposer une couche à Android pour créer des apps de réalité augmentée pour son téléphone, sans passer par un support de type CardBoard. On peut voir ARCore comme une évolution de Tango. La plateforme matérielle Tango n'a jamais su convaincre ni s'imposer sur les smartphones. ARCore fonctionne sans capteurs de profondeur ni caméra supplémentaire. Un plus mais aussi un moins.

Lors de l'annonce de cette couche, Google a précisé que seuls les modèles Galaxy S8 et Pixel seront compatibles, en attendant un support étendu. Car le succès d'ARCore passera par le nombre de modèles supportés et le parc de smartphones dans la nature. ARKit fonctionne quant à lui avec les appareils fonctionnant avec les processeurs A9, A10 et maintenant A11. Cela signifie que les iPhone SE, 6s / 6s Plus, 7 / 7 Plus, 8 / 8 Plus, X et les iPad 2017 sont compatibles. Bien entendu, l'ajout des capteurs spécifiques comme la caméra 3D et le traqueur de mouvement est important. La puissance du processeur A11 sera sans aucun doute un argu-

ment important même si les "anciens" modèles étaient déjà performants. Avec au minimum 100 millions de devices compatibles, ARKit a un argument non négligeable même si c'est avec les nouveaux terminaux que le potentiel pourra se révéler. Bien entendu, on verra tout et n'importe quoi dans les apps et les jeux proposés mais le marché ne demande qu'à décoller. Les capteurs dédiés permettront de transformer le device en outil de mesure par exemple, miroir intelligent pour le maquillage, la coiffure, etc. Merci au face tracking de l'iPhone X... Sur ce sujet précis, cette courte vidéo montre tout le potentiel et comment le développeur peut l'utiliser :

<https://developer.apple.com/videos/play/fall2017/601/>

Le modèle de développement d'ARKit ne surprendra pas beaucoup les développeurs iOS (Objective-C, Swift). Il faut iOS 11, XCode 9 et un équipement compatible.

Pour Google, un des fondamentaux sera l'accès à la réalité augmentée / mixte via le navigateur, ce que ARKit ne fait pas (pas encore ?) même si Apple suit les travaux actuels. Mais dans le principe, les deux SDK sont différents dans leur

approche. ARKit est une technologie par odométrie visuelle-inertielle permettant d'estimer la position d'un objet en mouvement. ARCore utilise l'IMU (inertial measurement unit ou unité de mesures de l'inertie). Cela oblige à connaître le matériel (caméra et le capteur IMU) implémenté dans l'appareil pour le supporter ; ce qui a limité le support initial. Apple fait sauter un des verrous en maîtrisant mieux le matériel. Tous les tests de comparaison pourront être refaits avec le matériel optimisé d'Apple et là, on pourrait réellement se faire une idée des performances réelles d'ARKit... face à l'ARCore. Au-delà de la pure technique, le contenu sera primordial. L'avantage d'Apple est la disponibilité du matériel et du logiciel dès maintenant, et l'iPhone X en novembre. Et les apps utilisant ARKit arriveront très vite sur l'App Store. Il faudra voir comment les concurrents réagissent même si nous ne parlons pas de casques comme Oculus, Vive ou Hololens mais ils auraient tort de ne pas observer l'affrontement entre les deux géants.

Alors serez-vous ARCore ou ARKit ?

La rédaction.

ARCore : la réponse de Google à l'ARKit d'Apple

• Sylvain SAUREL
Développeur Java / Android
sylvain.saurel@gmail.com
<https://www.ssaurel.com>

En Juin 2017, lors de sa grande messe annuelle qu'est le WWDC, Apple annonçait le lancement d'ARKit, un framework iOS donnant la possibilité aux développeurs d'intégrer de la réalité augmentée au sein de leurs applications. Google n'aura pas tardé à répondre puisqu'il vient d'annoncer la mise à disposition d'un framework pour Android répondant au nom d'ARCore. Dans cet article, nous vous proposons un tour d'horizon des possibilités d'ARCore ainsi qu'une comparaison de celles-ci par rapport à ARKit.

Sujet autrefois réservé aux technophiles, la réalité augmentée est en train d'atteindre le grand public, et les grandes compagnies cherchent donc à tirer profit au mieux de ce nouveau territoire à conquérir. Ainsi, lors de sa conférence WWDC 2017, Apple a lancé ARKit un framework permettant aux développeurs iOS d'intégrer de la réalité augmentée dans leurs applications. Les futurs appareils de la marque à la pomme permettront aux développeurs de proposer aux utilisateurs des expériences, on l'espère, créatives dans le domaine.

Travaillant déjà sur la réalité augmentée avec son projet Tango, Google se devait de répondre avec une solution plus simple. En effet, Tango se révèle contraignant en termes de prérequis hardware. Ainsi, le premier smartphone compatible, le Lenovo Phab 2 Pro, nécessite la présence de 3 caméras. Difficile dès lors d'atteindre le grand public. En lançant ARCore, Google propose un framework du même acabit qu'ARKit puisque ne nécessitant qu'une seule caméra pour fonctionner. Il pourra donc fonctionner sur tous les smartphones du marché.

Qu'est-ce que la réalité augmentée ?

Avant tout, il convient de définir ce qu'est la réalité augmentée. Il s'agit de combiner images réelles et virtuelles en temps réel. Cette technologie permet de filmer une table sur laquelle on pourra superposer des éléments de décors (figure 1). Un cas d'utilisation plus concret concerne un vendeur de meubles proposant une application permettant de voir les produits qu'il vend directement dans votre maison avant l'achat.

Pour bénéficier de la réalité augmentée, on passe bien souvent par un appareil mobile. Des lunettes à réalité augmentée sont bien en développement, telles les Google Glass, mais elles n'ont pas encore eu un grand succès. L'arrivée de la réalité augmentée auprès du grand public se fera donc par le biais des smartphones et des tablettes.

Que sont ARCore et ARKit ?

La réalité augmentée est une technologie plus qu'excitante pour les technophiles d'autant plus que tout le monde possède déjà un smartphone ou une tablette. Néanmoins, cela ne suffira pas à faire de la technologie un succès. Google et Apple l'ont bien compris et c'est pour cette raison qu'ils proposent respectivement ARCore et ARKit. Le but étant de fournir aux développeurs Android et iOS des frameworks leur permettant de laisser libre cours à leur créativité pour proposer des applications de réalité augmentée originales aux utilisateurs.

Vous l'aurez ainsi compris, c'est avant tout les développeurs en proposant des applications à base de réalité augmentée qui pourront permettre



1

Réalité Augmentée sur Mobile

à la technologie de décoller auprès du grand public. ARCore et ARKit ne sont donc pas des applications mais bien des outils au service des développeurs leur permettant de créer des applications de réalité augmentée.

ARCore vs ARKit : le duel

Comme de coutume dans le monde mobile, nous aurons affaire à un duel entre Google et Apple dans le domaine de la réalité augmentée pour mobiles. Voyons donc les points clés de ces frameworks. En ce qui concerne ARCore :

- **Suivi des mouvements :** via l'image de l'appareil hôte, il est possible de suivre des points clés pour établir et maintenir des positions d'objets en temps réel ;
- **Compréhension de l'environnement :** le framework permet de savoir si quelque chose est sur le sol ou une table ;
- **Estimation de la lumière :** cela aide les développeurs à placer des objets tout en appliquant des sources d'éclairage particulières ;
- **Disponibilité :** ARCore est en mode preview et sera donc limité à 100 millions d'appareils dans un premier temps avant une mise à disposition générale sans ajouts de hardware ;
- **Support :** ARCore sera supporté sous Android Studio mais également sous Unity et Unreal.

Pour ARKit, nous pouvons retenir les points clés suivants :

- **Suivi des mouvements** : fonctionnement identique à celui d'ARCore ;
 - **Compréhension de l'environnement** : l'environnement est reconnu via un renvoi de limites de manière basique ;
 - **Estimation de la lumière** : un certain nombre de patterns de reconnaissance seront à disposition ;
 - **Disponibilité** : ARKit fonctionnera uniquement sur les appareils Apple mis à jour vers iOS 11 ;
 - **Support** : ARKit sera supporté sous SceneKit mais également sur Unity et Unreal.

A la lecture de ces points, ARCore semble légèrement plus avancé qu'ARKit mais la différence réelle se fera au niveau des développeurs puisque c'est entre leurs mains que résidera le niveau de qualité des applications proposées.

Prise en main d'ARCore

Lors de l'annonce d'ARCore, Google a précisé que le Google Pixel et le Samsung Galaxy S8/S8+ étaient les premiers appareils compatibles. Pour pouvoir tester ARCore, il vous faudra donc posséder un de ces appareils puisque Google ne propose pas de support pour l'émulateur Android. Le téléchargement du SDK ARCore en preview pour Android Studio se fait à l'adresse suivante :

<https://github.com/google-ar/arcore-android-sdk/releases/download/sdk-preview/arcore-android-sdk-preview.zip>

La suite de la préparation de l'environnement de développement consiste à télécharger la bibliothèque ARCore Service sur GitHub à l'emplacement suivant :

<https://github.com/google-ar/arcore-android-sdk/releases/download/sdk-preview/arcore-preview.apk>

de via ADB :

```
$ adb install -r -d arcore-preview.apk
```

L'environnement de travail prêt, nous allons tester l'application de démonstration fournie avec le SDK accessible à l'emplacement suivant :

```
<ARCore SDK Folder>/samples/java arcore hello ar
```

Une fois le projet importé au sein d'Android Studio, vous devriez avoir à l'écran quelque chose de similaire à ce qui est présenté à la [figure 2](#). Le code Java de l'activité principale de l'application de démo montre clai-

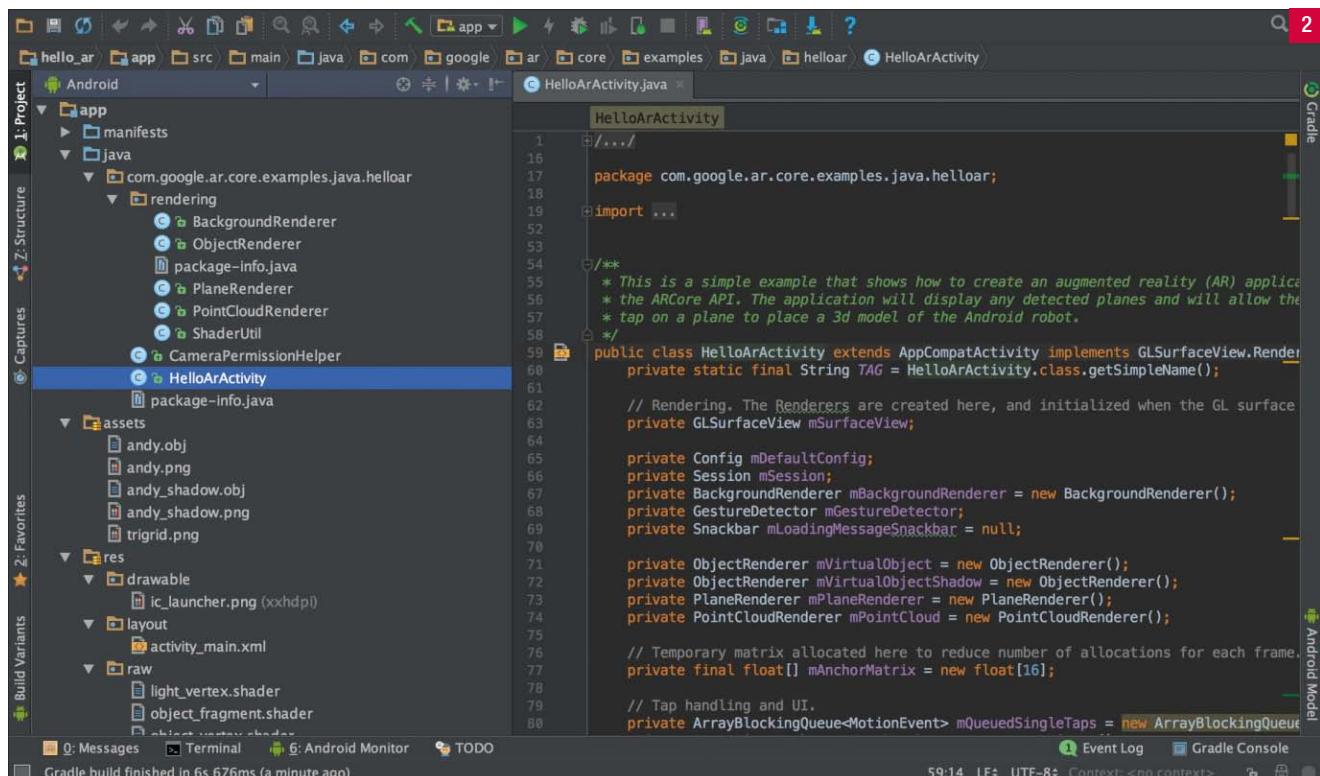
Le code Java de l’activité principale de l’application de demo montre clairement l’imbrication forte entre ARCore et OpenGL. En effet, l’activité principale implémente l’interface `GLSurfaceView.Renderer` afin d’interagir lors des phases de rendu de l’objet `GLSurfaceView` associé au layout de l’activité. A la création de l’activité, une session ARCore est créée via l’objet `Session`. Dans la méthode `onResume`, on appelle la méthode `resume` de l’objet `Session` et la méthode `onResume` de `GLSurfaceView`.

La partie intéressante se situe ensuite au sein des méthodes `onSurfaceCreated`, `onSurfaceChanged` et `onDrawFrame`. Au sein de la première, la texture devant être rendue est préparée puis passée à la session ARCore courante. Ici, la texture devant être superposée à l'image réelle étant naturellement un petit robot vert.

Dans la méthode `onSurfaceChanged`, on notifie la session ARCore que la taille de la vue a été modifiée et qu'il faut donc recalculer la matrice de perspective via un appel à la méthode `setDisplayGeometry` de l'objet `Session` avec en entrée ses nouvelles dimensions. Enfin, la méthode `onDrawFrame` permet de rendre à l'écran les différents petits robots virtuels devant être superposés à l'image réelle.

Tout ceci nous donne alors le code suivant pour ces 3 méthodes de l'activité principale :

```
    @Override
    public void onSurfaceCreated(GL10 gl, EGLConfig config) {
        GLES20.glClearColor(0.1f, 0.1f, 0.1f, 1.0f);
```



```

// Création de la texture à rendre et passage à la session ARCore
mBackgroundRenderer.createOnGLThread(this);
mSession.setCameraTextureName(mBackgroundRenderer.getTextureId());

// Rendu du robot
try {
    mVirtualObject.createOnGLThread(this, "andy.obj", "andy.png");
    mVirtualObject.setMaterialProperties(0.0 f, 3.5 f, 1.0 f, 6.0 f);
    mVirtualObjectShadow.createOnGLThread(this, "andy_shadow.obj", "andy_shadow.png");
    mVirtualObjectShadow.setBlendMode(BlendMode.Shadow);
    mVirtualObjectShadow.setMaterialProperties(1.0 f, 0.0 f, 0.0 f, 1.0 f);
} catch (IOException e) {}

// Rendu du plan
try {
    mPlaneRenderer.createOnGLThread(this, "trigrid.png");
} catch (IOException e) {}

mPointCloud.createOnGLThread(this);
}

@Override
public void onSurfaceChanged(GL10 gl, int width, int height) {
    GLES20.glViewport(0, 0, width, height);
    // Notification des changements de dimension de la vue à la session ARCore
    // pour recalcul de la matrice de perspective
    mSession.setDisplayGeometry(width, height);
}

@Override
public void onDrawFrame(GL10 gl) {
    GLES20.glClear(GLES20.GL_COLOR_BUFFER_BIT | GLES20.GL_DEPTH_BUFFER_BIT);

    try {
        // On récupère la frame courante via appel à la session ARCore
        Frame frame = mSession.update();

        // Gestion des taps à l'écran
        MotionEvent tap = mQueuedSingleTaps.poll();

        if (tap != null && frame.getTrackingState() == TrackingState.TRACKING) {
            for (HitResult hit: frame.hitTest(tap)) {
                // On vérifie si un plan a été touché
                if (hit instanceof PlaneHitResult && ((PlaneHitResult) hit).isHitInPolygon()) {
                    // On limite le nombre d'objets virtuels à 16
                    if (mTouches.size() >= 16) {
                        mSession.removeAnchors(Arrays.asList(mTouches.get(0).getAnchor()));
                        mTouches.remove(0);
                    }
                }
            }
        }
        // L'ajout d'une ancre indique à ARCore que les mouvements de cet objet
        // doivent être suivis
        mTouches.add(new PlaneAttachment(((PlaneHitResult) hit).getPlane(),
            mSession.addAnchor(hit.getHitPose())));
    }
}

```

```

        break;
    }
}

// On dessine l'arrière-plan
mBackgroundRenderer.draw(frame);

// Si le suivi n'est pas activé, on sort
if (frame.getTrackingState() == TrackingState.NOT_TRACKING) {
    return;
}

// On charge la matrice de projection
float[] projmtx = new float[16];
mSession.getProjectionMatrix(projmtx, 0, 0.1f, 100.0f);

// On charge la matrice de la caméra
float[] viewmtx = new float[16];
frame.getViewMatrix(viewmtx, 0);

// Calcul de la lumière l'éclairage
final float lightIntensity = frame.getLightEstimate().getPixelIntensity();

// On visualise les points suivis
mPointCloud.update(frame.getPointCloud());
mPointCloud.draw(frame.getPointCloudPose(), viewmtx, projmtx);

if (mLoadingMessageSnackbar != null) {
    for (Plane plane: mSession.getAllPlanes()) {
        if (plane.getType() == com.google.ar.core.Plane.Type.HORIZONTAL_UPWARD_FACING &&
            plane.getTrackingState() == Plane.TrackingState.TRACKING) {
            hideLoadingMessage();
            break;
        }
    }
}

// on visualise les plans
mPlaneRenderer.drawPlanes(mSession.getAllPlanes(), frame.getPose(), projmtx);

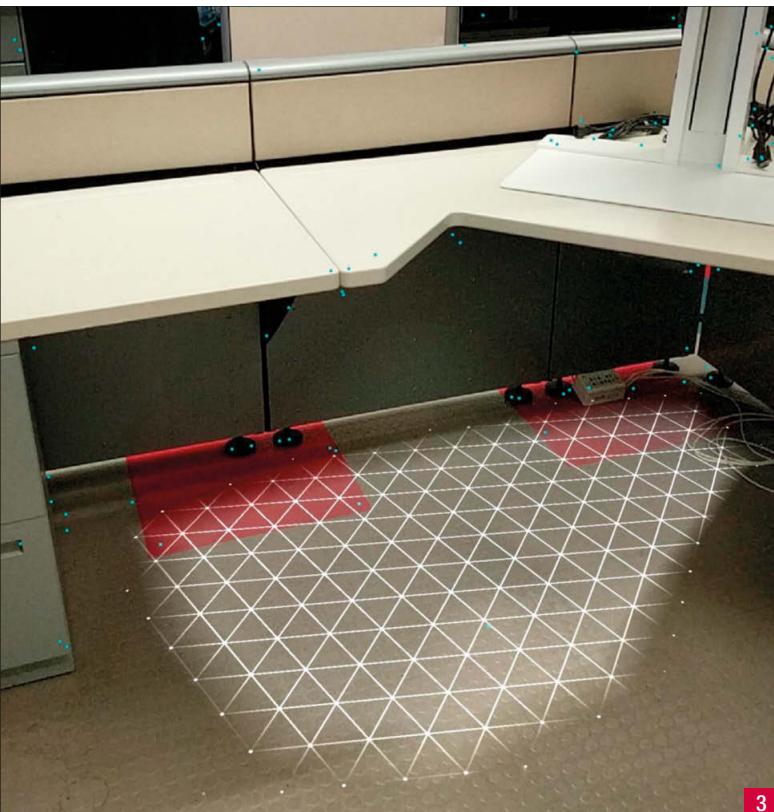
// Enfin, on visualise les ancrés créées
float scaleFactor = 1.0f;

for (PlaneAttachment planeAttachment: mTouches) {
    if (!planeAttachment.isTracking()) {
        continue;
    }

    planeAttachment.getPose().toMatrix(mAnchorMatrix, 0);

    // Dessin du modèle
    mVirtualObject.updateModelMatrix(mAnchorMatrix, scaleFactor);
    mVirtualObjectShadow.updateModelMatrix(mAnchorMatrix, scaleFactor);
    mVirtualObject.draw(viewmtx, projmtx, lightIntensity);
    mVirtualObjectShadow.draw(viewmtx, projmtx, lightIntensity);
}

```



Surface plane horizontale mappée

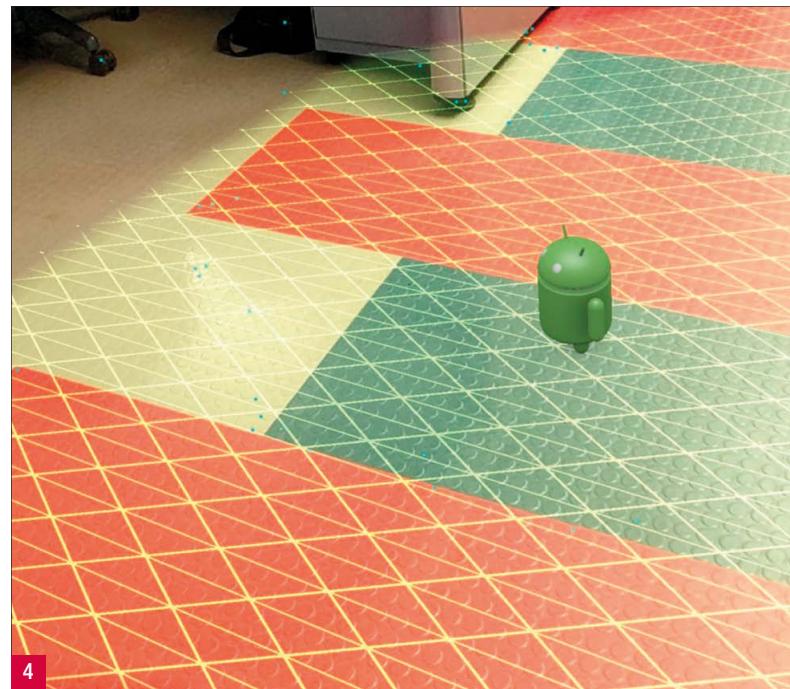
```
    }  
  } catch (Throwable t) {  
  }  
}
```

Test d'une application ARCore

La partie centrale du code de l’application de démonstration présentée, nous allons pouvoir la lancer sur un appareil de test. Une fois l’application lancée, un message à l’écran indique que l’application est en recherche de surfaces planes horizontales telles qu’une table, un bureau ou le sol tout simplement. Notez bien ici l’intérêt d’ARCore qui ne s’appuie que sur une seule caméra pour cette détection, là où le Projet Tango en demandait 3 ! Une fois le sol détecté par l’application, une grille apparaît alors pour montrer à l’écran la région qui est mappée (**figure 3**).

En touchant à l'écran la grille représentant la surface mappée, un petit robot tout vert est alors rajouté (figure 4). C'est à ce moment-là d'une application de réalité augmentée que la créativité des développeurs entrera en action et pourra faire la différence.

En continuant de toucher l'écran sur la zone mappée, d'autres petits robots pourront alors être ajoutés dans une limite de 16 comme vu dans le code précédemment présenté. Et ce, pour des raisons de performance de rendu. Il est bon également de préciser qu'ARCore s'appuie sur la caméra, l'accéléromètre ainsi que le gyroscope de l'appareil sous-jacent pour se situer dans l'espace. En vous déplaçant dans la pièce autour du robot vert, vous pourrez alors constater qu'il reste à la même position sur le sol.



Réalité augmentée en action



Conclusion

Fraîchement annoncé par Google, le framework ARCore est plus que prometteur comme vous avez pu le voir tout au long de cet article. Il jouera un rôle essentiel pour Google dans la bataille qu'il livrera à Apple et son ARKit dans le domaine de la réalité augmentée dans les années à venir. Au final, et une fois encore, ce sont les développeurs et in fine les utilisateurs qui devraient en sortir gagnant.

Néanmoins, ne vous attendez pas à voir débarquer des applications de réalité augmentée au sein du Google Play Store avant le premier semestre 2018.

Ce lancement d'ARCore pour la rentrée de Septembre étant avant tout un moyen pour Google de rappeler à tous sa présence dans le domaine de la réalité augmentée mais également de titiller l'intérêt des développeurs Android. Pari réussi en somme !

ARCore : une technologie très prometteuse



Wajdi Ben Rabah
Ingénieur en informatique spécialisé en technologies mobile et multimédia chez SFEIR. Il est très présent dans la communauté des développeurs. Sa passion pour les nouvelles technologies de réalité parallèle (VR/AR) est sans limite.

Email: benrabah.wajdi@gmail.com
Web: www.wajdiibr.com
Twitter: [@WajdiBenRabah](https://twitter.com/WajdiBenRabah)

La RA est très utilisée dans de nombreux domaines et ne s'arrête pas aux technophiles. Nous pouvons citer le domaine de l'éducation, de tourisme (en affichant des indications sur le contenu réel tel que les guides etc.), la santé (en affichant des informations supplémentaires du patient), les jeux vidéo (cette industrie est très rentable et se sert des dernières technologies pour pouvoir innover et vendre du rêve). La RA est utilisée également dans le domaine militaire et ceci est expliqué par la cause préventive. En ayant des informations qui s'ajoutent aux soldats, ces derniers peuvent prendre une meilleure décision dans certains cas et éviter sûrement des pertes humaines. Cette technologie est également utilisée en publicité et divertissement, des domaines très populaires bien rentables pour les entreprises. Dans le monde de la réalité augmentée, on distingue les applications basées sur des marqueurs d'un côté et les applications basées sur la localisation d'un autre côté. Les marqueurs fonctionnent en procédant par la reconnaissance d'un certain modèle (pattern) comme un code à barres ou un symbole lorsque la caméra le pointe. Le résultat serait de superposer un contenu digital par dessus et de l'afficher sur l'écran de l'appareil utilisé. Si l'image ou le modèle est en 3D ou animé, l'effet prendra vie sur la surface ciblée. [1]

La localisation, elle, détecte la position du utilisateur et utilise les informations collectées afin de fournir un contenu digital adapté à cette position. Elle est très utilisée dans le milieu touristique. [2]

Google ARCore : allons-y !

Selon la documentation officielle, ARCore est une plateforme qui permet de créer des applications de réalité augmentée sur Android. Elle utilise différentes technologies afin d'intégrer le contenu virtuel dans le monde réel en employant les caméras intégrées dans les téléphones.

La réalité augmentée (RA) est un concept qui a séduit beaucoup de gens. Contrairement à la réalité virtuelle, la RA présente une technologie qui permet de fusionner un contenu virtuel (en 3D ou 2D) avec notre monde réel en utilisant la superposition du contenu digital en temps réel.

• **Suivi de mouvement** : permet au téléphone de comprendre et suivre sa localisation par rapport au monde réel ;

• **Compréhension de l'environnement** : permet au téléphone de détecter la localisation et la taille d'une surface horizontale comme une table par exemple ;

• **Compréhension de la lumière** : permet au téléphone de comprendre l'état des lumières dans l'environnement actuel.

Pour déployer et tester votre projet ARCore il vous faut un téléphone compatible capable de faire tourner ces applications en RA. A l'heure de l'écriture de l'article, il vous faudra avoir soit un téléphone Google Pixel ou un Pixel XL ou encore un Samsung S8.

Globalement, ARCore assure deux fonctionnalités principales : le suivi de la position du téléphone, et la compréhension et augmentation du monde réel.

Actuellement ARCore est en version preview. La prochaine nouvelle version aura certainement des changements radicaux.

Préparation de l'environnement technique

Pour commencer à développer pour Google ARCore, il vous faudra techniquement :

• Installer Android SDK 7.0 ou supérieure (API niveau 24). Pour installer le SDK, veuillez installer Android Studio ici :

<https://developer.android.com/studio/index.html>

• Pour mettre à jour votre SDK existant, rendez-vous à l'outil SDK Manager dans Android Studio.

• Installez Unity 2017.2 Beta 9 avec le composant Android Build Support.

• Téléchargez ARCore SDK. La manière la plus simple serait de cloner le repository depuis Github ou de télécharger manuellement le projet ici <https://github.com/google-ar/arcore-unity-sdk.git>.

• Vous devez disposer d'un téléphone Android Galaxy S8, Google Pixel ou Google Pixel XL.



1

Exemple d'application de RA basée sur un marqueur.



2

Exemple d'application de RA basée sur la localisation.

Les appareils virtuels ne supportent pas les applications ARCore.

• Activer les options développeur.

• Activer le débogage USB.

• Installer ARCore service sur votre téléphone : vous pouvez déployer l'apk directement via adb ou encore l'installer depuis le lien ou en vous envoyant un mail avec l'apk en pièce jointe. Vous pouvez également opter pour la manière la plus "technique" : Téléchargez ARCore service ici <https://github.com/google-ar/arcore-android-sdk/releases/download/sdk-preview/arcore-preview.apk>. Connectez votre téléphone en mode USB à votre machine. Installez le service en exécutant cette commande : adb install -r -d arcore-preview.apk

Projet Unity

1 Lancez Unity 2017.2 Beta 9 ;

2 Créez un nouveau projet. File > New project ;

3 Donnez un nom (ARCoreTest pour cet article), laissez en 3D et validez ;

Note : vous pouvez vous connecter à votre compte Unity afin de synchroniser vos assets et vos projets.

4 Importez le SDK ARCore : Assets > Import package et sélectionnez : arcore-unity-sdk-preview.unitypackage.

Lecture du projet fourni par Google

Vous allez avoir le dossier nommé GoogleARCore. Je vous invite à lancer la scène donnée à

titre d'exemple sous le dossier GoogleARCore>HelloARExample>Scenes>HelloAR. Nous allons nous baser dessus vu que l'équipe de Google ont employé chaque fait usage de chaque fonctionnalité de ARCore dans ce projet. Si vous avez n'avez pas touché aux préférences d'affichage de Unity, vous allez avoir une rubrique nommée Hierarchy à gauche. Le premier composant ARCore Device est très important. C'est un prefab (existant dans le dossier prefabs de Google ARCore et donc réutilisable) qui contient une caméra en tant que fils. Sélectionnez ARCore Device et naviguez sur le menu inspector à droite. Vous allez trouver deux scripts C#. Le premier est Tracked Pause Driver. Il se charge de définir le device à utiliser (par défaut Generic XR Device), la source du tracking, le type (suivre la position et la rotation ou juste une seule d'entre les deux par exemple), le mode de mise à jour du suivi (avant le rendu par exemple) et finalement l'utilisation de l'option use relative transform. Tout comme ARKit d'Apple, ARCore est relatif à la position à laquelle vous lancez votre téléphone et se base dessus pour le mouvement plus tard. Le deuxième script est Session Component. Ce composant est géré par DefaultSessionConfig qui est associé à AR Session Config. Cliquez deux fois sur le sessionconfig et vous allez atterrir sur un fichier sous Assets>GoogleARCore>Configurations>DefaultSessionConfig. [3]

Vous pouvez créer votre propre fichier (sous assets clic droit>Create>GoogleARCore>SessionConfig), mais l'utilisation de ce dernier par défaut devra faire l'affaire. Vous pouvez constater que les fonctionnalités qu'on manipule ici sont des bases pour un projet ARCore. Personnellement je ne vois aucun cas dans lequel vous seriez amené à désactiver le fond en réalité augmentée, la détection des surfaces ou la désactivation des Pointcloud.

Parmi les composants restants, nous trouvons

un Canvas, qui se charge d'afficher un texte GUI afin de guider l'utilisateur ou donner plus d'informations. Nous avons également une lumière dans la scène appelée Environmental Light qui est un composant prefab. Ce dernier englobe un script en langage C# qui contrôle la lumière dans la scène. Nous trouvons également un composant appelé ExampleController qui contient un script appelé Hello AR Controller. Ce dernier n'est qu'un Game Object qui contrôle le déroulement de la scène en général: il garde une référence sur la caméra, et donc l'utilisateur en mode first person. Il contrôle la surface à détecter dans le monde réel. Il garde une référence sur le GUI qui affiche les informations sur l'état de reconnaissance du terrain et finalement l'objet à afficher (pour notre cas c'est un prefab nommé Andy : le bonhomme d'Android en 3D. Juste en bas aussi vous allez trouver un Game Object nommé Point Cloud. Ce dernier est un composant géré par un shader et contrôle les vertices du cube. C'est possible de changer la taille du point ainsi que la couleur depuis le menu Inspector à droite sur votre écran. Les deux composants restants sous le menu hierarchy sont : EventSystem et Directional light. Le premier est utilisé pour la gestion des événements levés sur la scène, et le deuxième est juste une lumière avec une direction pré définie. Veuillez noter qu'ils n'ont rien à voir avec ARCore mais plutôt avec les fonctionnalités Unity. Maintenant que nous avons expliqué tout ce qui est présent sur la scène Hello AR, nous allons pouvoir créer notre propre projet ARCore en utilisant ces mêmes composants.

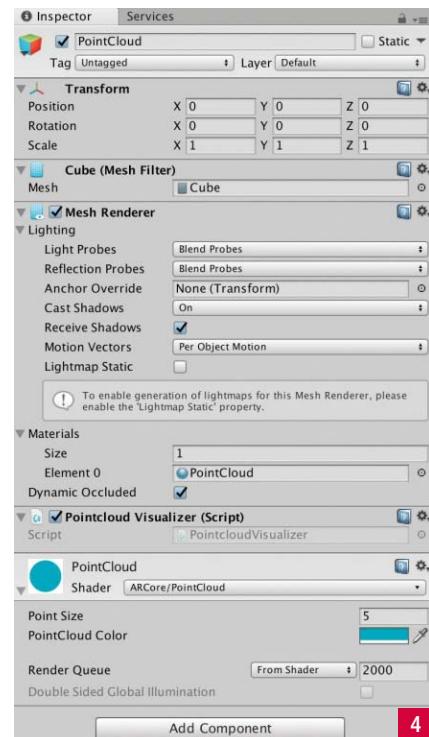
Notre premier projet ARCore de zéro

Sur Unity, tout en bas, vous allez trouver un menu nommé Project. Rendez-vous sur le chemin principal (donc sous Assets). Maintenant

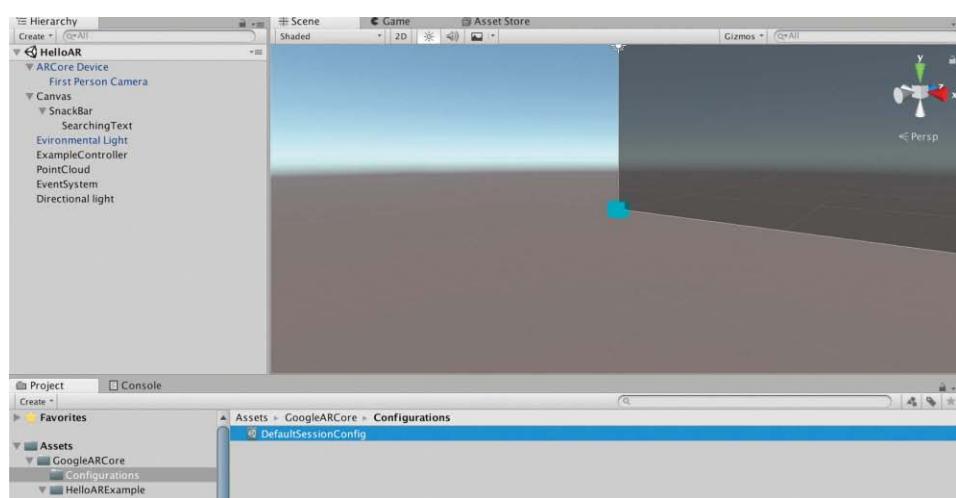
créez votre scène en suivant ces étapes: Menu File>New Scene.

PointCloud

Créer un Cube dans votre scène afin d'utiliser les points de reconnaissance (pointcloud). Menu GameObject>3D Object>Cube. À droite sous le menu Inspector, changez le nom Cube en PointCloud. Supprimez le box collider en cliquant sur l'engrenage face au composant et sélectionnez remove component. Juste en bas et en restant sur le cube dans le menu hierarchy, vous allez trouver un menu Materials. Étendez votre menu et vous allez trouver un sous menu avec les options Size et Element 0 qui est affecté à Default-Material. Cliquez sur le petit point juste devant et chercher PointCloud, puis valider afin de remplacer votre material par



4



3

ce dernier. Assurez-vous que l'option Dynamic Occluded est sélectionnée. Maintenant, appuyez sur Add Component et cherchez le script Pointcloud Visualizer. Le rendu final du composant doit ressembler à ceci. [4]

Astuce: Vous pouvez créer un prefab à partir de la scène principale donnée par Google sans passer par tout ce processus. Pour ce faire, il suffit juste de sélectionner PointCloud depuis le menu Hierarchy, et le glisser dans un dossier sous votre projet (par exemple, créer un dossier nommé prefabs sous Assets). Pour l'utiliser dans un autre projet, il vous suffit maintenant de le glisser et le jeter dans la scène. Veuillez ajuster la position du composant comme dans notre imprimé écran.

ARCore Device

Maintenant, importez ARCore Device depuis le dossier GoogleARCore>Prefabs. Pour donner un peu de lumière à notre rendu, veuillez importer également le prefab Environmental Light qui réside sous le même dossier que ARCore Device.

Animation du monde augmenté

Nous allons maintenant créer notre monde virtuel qui se superposera avec notre réalité. Pour ce faire, j'ai pensé à mettre un objet 3D tel qu'un cube, mais nous allons changer pour ce projet. Nous allons ajouter des particules sous la forme d'étoiles filantes, qui s'afficheront dans notre rendu. Il faut donc importer les particules, et pour ce faire rendez-vous au menu d'en haut Assets>import package>Effects. Sélectionnez tout et validez. Maintenant vous pouvez créer des particules par défaut en suivant ce processus : Game Object>Effects>Particule System. Ensuite, ce qui suit est une personnalisation Unity classique du composant des particules. Pour faire simple, nous allons ajouter un peu de gravité, changer la couleur des étoiles, et leur position. Nous allons également augmenter la durée et le

nombre de particules maximum à afficher. [5]

Dans l'onglet Emission nous allons ajuster la fréquence d'émission puis dans Shape, nous allons faire en sorte que les particules sortent en cercle d'un rayon de 30. [6]

Pour finir, nous allons changer le material par celui des étoiles. C'est très simple il suffit juste de cliquer sur le petit point devant le material par défaut sous render et chercher Star. [7]

Le rendu ressemblera à peu près à ce qui est ci-dessous. [8]

Maintenant, il ne vous reste qu'à sauvegarder la scène (File>Save scene et donner un nom). Pour finir, branchez votre smartphone et procédez au déploiement comme indiqué dans la prochaine section.

Déploiement

Rendez-vous dans le menu build settings (File>Build settings ou le raccourci CTRL+SHIFT+B) Modifiez la plateforme cible à Android en cliquant sur Switch Platform. Dans le nouveau menu sous Player Settings, ouvrez Android Player Settings et changez ces paramètres:

- Other Settings > Multithreaded Rendering: Off (afin de forcer le rendu graphique et pas sur votre CPU) ;
- Other Settings > Package Name : un identifiant unique afin de différencier votre application en cas de commercialisation sur les stores, comme com.programmez.arcoretest ;
- Other Settings > Minimum API Level: Android 7.0 or higher ;
- Other Settings > Target API Level: Android 7.0 or 7.1 ;

XR Settings > Tango Supported: On.

Les scènes dans le menu global de build indiquent les scènes que vous allez inclure dans votre produit. Ajoutez la scène en cours, vérifiez que votre téléphone est bien branché en USB et qu'il est reconnu par votre machine et lancez le build en cliquant sur Build & Run. Unity construit votre projet puis lance le résultat sur le téléphone cible. Il se peut que vous ayez des problèmes de déploiement. Je vous invite à vérifier la compatibilité de votre téléphone ou la bonne installation des services ARCore. Dans le cas échéant, veuillez vous rendre sur la page Github du projet ici <https://github.com/google-ar/arcore-android-sdk/issues> ou sur les forums d'entraide.

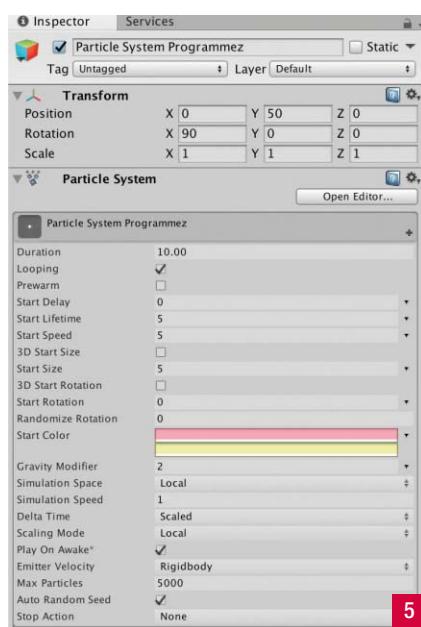
Conclusion

ARCore est plus que prometteur comme vous avez pu le voir tout au long de cet article. Néanmoins, ne vous attendez pas à voir débarquer des applications de réalité augmentée au sein du Google Play Store avant le premier semestre 2018. L'annonce d'ARCore à la rentrée de Septembre étant avant tout un moyen pour Google de rappeler sa présence dans le domaine de la réalité augmentée. Un framework qui a permis au géant de gagner son pari.

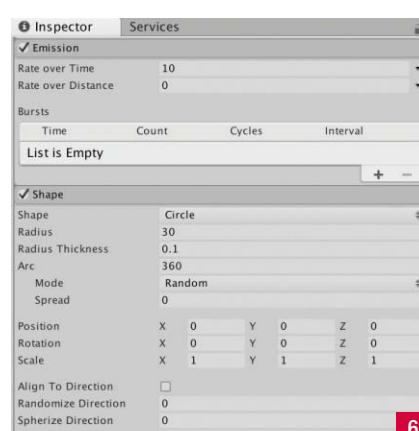
Il existe un projet Github pour faire fonctionner ARCore sur les téléphones non compatibles. Je vous invite à y jeter un coup d'œil si vous voulez. <https://github.com/tomthecarrot/arcore-for-all>.

Sources et compléments :

https://github.com/wajidibr/arcore_programmez



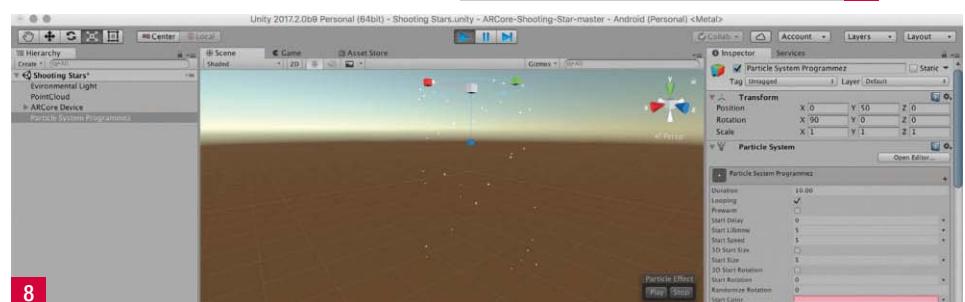
5



6



7



8



Construisez votre 1ère application de réalité augmentée avec ARKit

• Kevin Ha
OCTO Technology

Durant la WWDC 2017, Apple a présenté son framework de réalité augmentée ARKit. La réalité virtuelle n'est pas nouvelle mais Apple a créé un véritable engouement de la part des développeurs notamment par la rapidité et la facilité de créer une expérience de réalité virtuelle sur iOS 11. La sortie du dernier OS ainsi que l'arrivée de iPhone X et 8 devraient faire émerger de plus en plus d'applications de RA sur l'AppStore.

La principale force d'ARKit réside dans le fait qu'il n'y ait besoin d'aucun capteur supplémentaire pour être utilisé. Cependant, vous devez vous doter d'un appareil embarquant une puce A9 ou supérieure et donc au minimum un iPhone SE ou 6S ou un iPad mini 2. ARKit dispose des fonctionnalités suivantes :

Tracking

Il permet de suivre la position de l'appareil dans l'espace et en temps réel (principe de l'odométrie) avec un haut degré de précision grâce à la caméra et aux capteurs de mouvements de l'iPhone. Ainsi au travers des informations récupérées, la position est recalculée pour chaque image vidéo, au moins 30 fois par seconde.

Scene Understanding

Apple a appelé cette technique "Plane detection" qui fournit la possibilité de détecter les plans (seulement horizontaux pour le moment) comme les tables ou les sols. Elle est actuellement encore en bêta car d'autres fonctionnalités comme la détection de murs devrait venir alimenter le framework.

Lighting Estimation

ARKit est également capable d'utiliser le capteur de luminosité pour déterminer la quantité de lumière dans la pièce environnante pour appliquer le degré de luminosité sur l'objet virtuel.

TrueDepth Camera

Avec l'introduction de l'iPhone X, ARKit a la possibilité de détecter la topologie, la position et l'expression du visage de l'utilisateur avec un haut degré de précision en temps réel. Cela rend plus facile d'appliquer des expressions de visage sur des personnages en 3D ou des effets pour les selfies.

Rendering

ARKit intègre des librairies graphiques 2D/3D internes telles que SpriteKit (2D), SceneKit (3D) ou encore Metal, mais fonctionne également avec des librairies externes comme Unity ou Unreal Engine.

Concepts de base

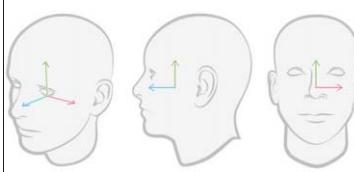
Session

Une session représente le cœur d'ARKit, il vous faut démarrer une **ARSession** unique à chaque fois que vous lancez votre application. Elle

va gérer la collecte de données des capteurs de mouvements et de la caméra. Un **ARConfiguration** doit être fourni en fonction des capacités de l'iPhone ou de l'iPad pour lancer la session.

Configuration

ARKit fournit 3 types de configurations :

	AROrientationTrackingConfiguration <i>pour seulement suivre l'orientation de l'appareil sur les 3 axes de rotation (profondeur, inclinaison, direction) communément appelés "3 degrees of freedom (3DOF)". Cette configuration est adaptée pour les iPhone 6S par exemple.</i>
	ARWorldTrackingConfiguration <i>est quant à lui utilisé pour suivre la position relative, l'orientation et les informations de l'environnement sur 3 axes de translation (avant/arrière, haut/bas et droite/gauche) en plus des 3 axes de rotation cités plus tôt appelés "6 degrees of freedom (6DOF)".</i>
	ARFaceTrackingConfiguration <i>fournit la possibilité de suivre le mouvement et l'expression du visage grâce à la caméra frontale de l'iPhone X.</i>

Frames

Lors d'une session, ARKit va capturer en continu des images stockées dans un **ARFrame** qui va contenir les informations de suivi et de l'environnement telles que les points suivis et le degré de luminosité.

Feature Points

Les "features points" représentent une liste de points **ARPointCloud** collectés durant le temps fournissant des informations détaillées de l'espace. C'est ainsi qu'ARKit est capable d'estimer ou de détecter des plans horizontaux.

Anchors

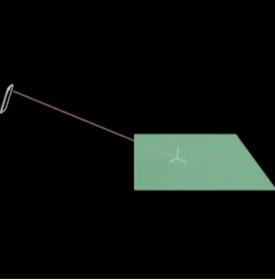
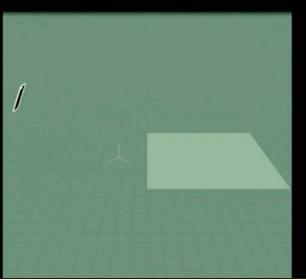
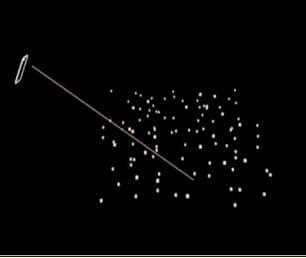
Un **ARAnchor** illustre une position et une orientation dans l'espace, il sert de point de référence sur lequel ARKit va fixer un objet virtuel et

donner l'illusion que celui-ci ne bouge pas lorsque l'iPhone se déplace autour. Ils peuvent être ajoutés manuellement ou automatiquement en utilisant l'option `.planeDetection` de la configuration.

Hit-Testing

Cette technique recherche les objets réels ou des **ARAnchors** dans les images capturées au cours d'une **ARSession** à un point spécifique dans l'espace. Cette méthode renvoie une liste de features points ordonnée du plus près au plus loin de la caméra.

4 types de détection sont disponibles

	
<code>.existingPlaneUsingExtent</code>	<code>.existingPlane</code>
En détectant le point d'ancrage présent dans la scène (grâce à l'option <code>.planeDetection</code>) respectant la taille du plan.	En détectant le point d'ancrage présent dans la scène (grâce à l'option <code>.planeDetection</code>) sans considérer la taille du plan.
	
<code>.estimatedHorizontalPlane</code>	<code>.featurePoint</code>
En estimant un plan dont l'orientation est perpendiculaire à la gravité.	En utilisant les features points

Codons !

D'abord, vous devez vous procurer la dernière version Xcode 9 afin de créer votre première application ARKit. [1] et [2]

Configuration

Ici nous souhaitons utiliser la caméra arrière de l'iPhone pour placer des objets dans l'espace mais nous aurions aussi prendre en compte le **ARFaceTrackingConfiguration** vu précédemment.

```
if ARWorldTrackingConfiguration.isSupported {
    configuration = ARWorldTrackingConfiguration()
} else {
    configuration = AROrientationTrackingConfiguration()
}
```

Créer et démarrer une session

Pour simplifier le code, ici nous initialisons une **ARSession** mais si vous intégrez SceneKit ou SpriteKit vous pouvez directement accéder aux sessions à travers les classes **ARSCNView** et **ARSKView**.

```
/** Crée une session */
let session = ARSession()

/** Nomme self comme étant le delegate de la session pour avoir
 accès aux méthodes de ARSessionDelegate() */
session.delegate = self

/** Lance la session avec la configuration */
session.run(configuration)
```

Gérer une session

Une fois la session démarrée, vous avez la possibilité de la gérer comme ceci :

```
/** Suspend la session */
session.pause()

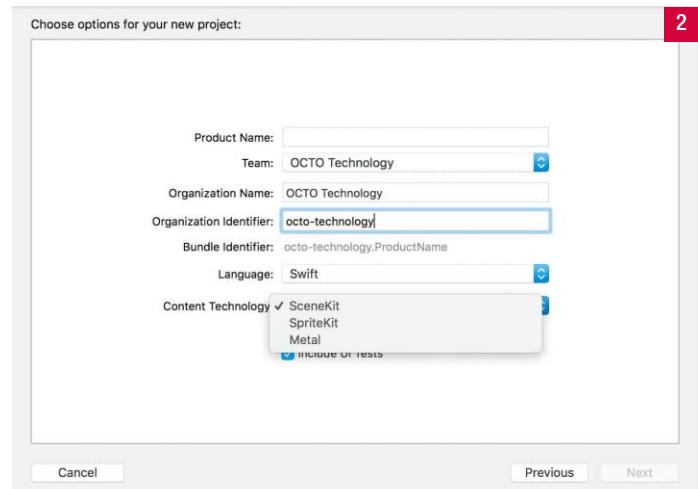
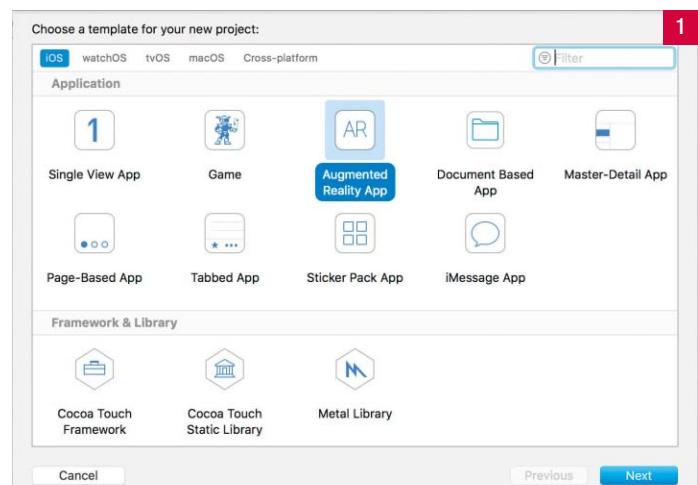
/** Resume la dernière session lancée */
session.run(session.configuration!)

/** Change la configuration de la session */
session.run(newConfiguration)

/** Redémarre la session */
session.run(configuration, options: .resetTracking)
```

Gérer les mises à jours lors d'une session

Après avoir nommé le delegate, vous aurez accès aux méthodes qui gèrent les événements de la session :



```
/** Fournit une mise à jour de la dernière frame qui contient
l'image capturée, les points de suivis, le degré de luminosité et
les informations de l'environnement */
func session(_: ARSession, didUpdate: ARFrame) { }

/** Est appelé lorsqu'un ARAnchor est ajouté à la session */
func session(_: ARSession, didAdd: [ARAnchor]) { }

/** Est appelé lorsqu'un ARAnchor a été mis à jour */
func session(_: ARSession, didUpdate: [ARAnchor]) { }

/** Est appelé lorsqu'un ARAnchor a été supprimé */
func session(_ session: ARSession, didRemove anchors: [ARAnchor]) { }

/** Gère les erreurs lors de la session */
func session(_: ARSession, didFailWithError: Error) { }

/** Gère les différents états de la caméra */
func session(_ session: ARSession, cameraDidChangeTrackingState camera:
```

Gérer les interactions avec le Hit-testing

A partir de là, nous allons placer un objet virtuel dans l'espace avec le hit-testing vu précédemment qui va rechercher les feature points les plus proches de la caméra.

```
/** Déclenche le hit-test sur le centre de la vue en choisissant l'option .featurePoint */
let planeHitTestResults = sceneView.hitTest(view.center, types: .featurePoint)

/** Recupère le point le plus proche de la caméra */
if let closestResult = planeHitTestResults.first {

    /** Place un point d'ancrage sur ce point là */
    let anchor = ARAnchor(transform: closestResult.worldTransform)

    /** Ajoute le point d'ancrage à notre scène */
    sceneView.session.add(anchor: anchor)
}
```

La création d'un **ARAnchor** fera le lien avec un **SCNNode** : une classe spécifique à SceneKit, à travers la méthode `render` du **ARSCNViewDelegate**.

```
/** Crée un noeud associé au point d'ancrage ajouté à la session */
func render(_ renderer: SCNSceneRenderer, nodeFor anchor: ARAnchor) -> SCNNode? {

    /** Crée un cube en 3D */
    let box = SCNBox(width: 0.05, height: 0.05, length: 0.05, chamferRadius: 0.01)
    box.firstMaterial?.diffuse.contents = UIColor.blue

    /** Crée un noeud en lui attribuant le cube */
    let node = SCNNode(geometry: box)

    /** Initialise sa position avec un vecteur (x,y,z) */
    node.position = SCNVector3Zero

    /** Ajoute le noeud à la scène */
    sceneView.scene.rootNode.addChildNode(node)

    return node
}
```

Nous obtenons le résultat suivant : [3]

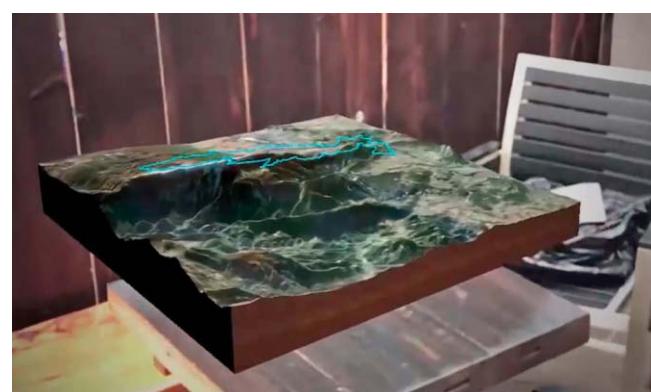
Pour aller plus loin...

Nous avons eu une première approche d'ARKit avec sa prise en main en intégrant SceneKit pour mettre en place des objets virtuels dans l'espace. Bien évidemment, ARKit est encore amené à évoluer et à proposer plus de fonctionnalités pour rendre des contenus plus réalistes notamment avec l'introduction de l'iPhone X qui embarque de nouvelles caméras comme la caméra "TrueDepth" proposant d'appliquer des effets virtuels sur nos visages.

Apple semble avoir donné un nouvel élan à la réalité augmentée, il suffit de consulter (@madewitharkit) pour se faire une idée de la puissance du framework et de ce que pourrait être le futur de celle-ci.

Ceci a également poussé Google à sortir rapidement son framework ARCore pour venir concurrencer celui de la pomme.

A vous de trouver les meilleurs cas d'usages dès à présent. De grands acteurs semblent d'ores et déjà s'emparer de cette technologie comme en peut témoigner l'application d'IKEA qui vous aidera à meubler



app FitnessAR

virtuellement votre logement avant même de vous rendre en magasin. Allons-nous enfin assister à l'avènement de la réalité augmentée chez le grand public ?

Référence :

<https://developer.apple.com/documentation/arkit>
<https://developer.apple.com/documentation/scenekit>
<https://www.whitesmith.co/blog/arkit-introduction/>

Documentation : Word to Web



Maxime Pean,
Expert Technique,
SQLI Nantes



Dans de nombreuses entreprises, toute la documentation est rédigée dans de bons vieux documents Word. C'est un standard connu de tous qui permet de faire de la mise en page sur du papier numérique. Le patrimoine de documents Word est souvent ancien, volumineux, mal versionné, mal structuré, mal présenté, pas à jour donc pas fiable.

Pas évident d'y retrouver ce que l'on recherche... Pourtant on l'a injecté dans une GED, ça devrait nous rendre la vie plus facile, non ?

Bref, ni les utilisateurs ni vous ne vous y référez plus. Résultat, on ne sait plus ce que l'application est censée faire et tout le monde perd son temps. On ouvre des tickets dans Jira, on se demande si c'est un bug ou une évolution et on va finalement demander son avis à un expert. Vous connaissez déjà sa réponse : " Il faut regarder dans le code ". Sauf que souvent, le code, il est aussi ancien et obscur que sa doc.

Finalement la documentation ne joue plus son rôle : nous aider.

Il faut reconnaître que cette représentation papier n'est pas des plus pratiques dans le monde du développement informatique car elle est complètement décorrélée de nos outils de développement. Et puis à quoi ça sert de passer du temps à faire de la mise en page alors que la plupart du temps le but n'est pas d'imprimer sa documentation sur du papier mais de la consulter sur un écran.

Alors arrêtons de passer du temps sur la forme alors que ce qui compte c'est le fond ! Pour trouver une meilleure façon de présenter une documentation il n'y a pas besoin de chercher bien loin, par exemple le `readme.md` de GitHub est simple et présente bien. On trouve également beaucoup de documentations de frameworks ou de logiciels qui sont superbes avec une présentation uniforme, de la coloration syntaxique du code, un index automatique, une recherche etc.

Vous n'allez pas me dire que vous n'avez jamais rêvé comme moi d'avoir une documentation comme celle de Python ou de MongoDB par exemple ? [1]

Nous sommes d'accord, alors allons-y, passons notre documentation au format web !

Qu'est-ce qu'on peut faire ?

Pour migrer notre documentation au format web il faut, comme pour tous les projets non prioritaires, d'abord éliminer les freins qui sont souvent le budget et le temps. J'ai donc cherché des outils permettant de migrer facilement et rapidement notre vieille documentation au format web.

Sphinx

J'ai découvert que de nombreux éditeurs comme MongoDB, Google ou Python utilisent un outil nommé Sphinx qui permet de générer des sites web (entre autres) à partir de fichiers textes rédigés en un markup langage nommé ReStructuredText (.rst). [2]

Le site web est généré à partir d'un template paramétrable grâce à un fichier de configuration, ce qui permet d'obtenir rapidement un look'n'feel à votre goût. Il existe une collection de thèmes Sphinx et vous pouvez aussi créer le vôtre. [3]

Odt2sphinx

Dans l'écosystème de Sphinx, on trouve odt2sphinx, un outil de conversion de fichiers Open Document en une arborescence directement

The screenshot shows the Python 3.5.1 documentation homepage. It features a sidebar with links for 'Download', 'Docs for other versions', 'Other resources', and a 'Quick search' bar. The main content area is titled 'Python 3.5.1 documentation' and includes sections for 'What's new in Python 3.5?' (with a link to 'all "What's new" documents since 2.0'), 'Tutorial' (with a 'start here' link), 'Library Reference' (with a 'keep this under your pillow' link), and 'Language Reference' (with a 'describes syntax and language elements' link). A small red box in the bottom right corner contains the number '1'.

Exemple de site Web Sphinx : la documentation python

```

1 ... maDocphinx documentation master file, created by
2 sphinx-quickstart on Tue May 17 18:06:00 2016.
3 You can adapt this file completely to your liking, but it should at least
4 contain the root 'toctree' directive.
5
6 Welcome to maDocphinx's documentation!
7 =====
8
9 Contents:
10
11 ... toctree::
12   :maxdepth: 2
13
14
15 Indices and tables
16 =====
17
18
19 * :ref:`genindex`
20 * :ref:`modindex`
21 * :ref:`search`

```

Exemple de code RestructuredText

The screenshot shows the Sphinx documentation generator interface. It includes sections for 'Builtin themes', 'Theme overview', and 'HTML theming support'. The 'HTML theming support' section shows examples of how to use themes like 'classic' and 'sphinx_rtd_theme'. A small red box in the bottom right corner contains the number '2'.

The screenshot shows the Sphinx theme configuration interface. It displays the 'HTML theming support' configuration for the 'classic' theme, including options for 'HTML theme', 'HTML theme options', and 'HTML theme settings'. A small red box in the bottom right corner contains the number '3'.

Les thèmes Sphinx

Une fois votre configuration à jour, vous pouvez lancer de nouveau une commande make html et observer leurs impacts sur votre documentation web regénérée.

Trucs et astuces

Les espaces typographiés

Si vous avez des espaces en fin de ligne typographiés en gras, en italique ou soulignés, cela génère du code invalide, par exemple : "***mon texte ***". Sphinx relèvera ces erreurs au make et elles peuvent être assez nombreuses dans certains documents. N'hésitez pas à sélectionner entièrement le contenu de votre document et à supprimer toutes ces typographies, ça fait gagner du temps.

Les grandes images

Certaines images intégrées à des documents Word peuvent être très grandes et leur taille affichée est modifiée directement dans Word. Il faudra redimensionner ces images pour qu'elles ne soient pas immenses dans votre documentation Web.

Qu'est-ce que ça change ?

L'édition

Vous pouvez maintenant éditer votre documentation dans votre outil de développement préféré (Atom, Sublime, Brackets, Webstorm, Eclipse, ...) qui vous proposera souvent une preview.

Les gitations des versions

Vous pouvez gérer vos fichiers dans votre outil de gestion de versions

préféré, git par exemple. Terminé la scission entre votre code et votre doc, tout est intégré à votre environnement de développement, rien que ça : c'est énorme !

Le build

Il ne vous reste plus qu'à "builder" votre doc dans votre outil de build continu préféré comme Jenkins par exemple et en profiter pour générer automatiquement une page de release notes par exemple. Vous pouvez publier une version de votre documentation par version de votre projet.

La publication

Votre documentation accessible dans un simple navigateur web. Vous pouvez y faire référence avec des liens hypertextes ou l'intégrer dans un site de votre intranet par exemple.

Le rendu

Le rendu de votre documentation est homogène et paramétrable. Le code est coloré syntaxiquement.

L'indexation

L'index de votre documentation est généré automatiquement et un champ permet de faire des recherches par mots clés.

Qu'est-ce qu'on attend ?

Comme vous pouvez le voir, migrer sa documentation au format web est assez abordable et les gains sont immédiats alors foncez !

Abonnez-vous ! à

Nos classiques

1 an 49€

11 numéros

2 ans 79€

22 numéros

Etudiant 39€

1 an - 11 numéros

PROGRAMMEZ !

le magazine des développeurs

Abonnement numérique

PDF

1 an - 11 numéros

35€

Souscription uniquement sur
www.programmez.com



Le GDPR : contrainte ou opportunité ?



Olivier Chotin,
Consultant Senior SQLI

Applicable dès le 25 mai 2018⁽¹⁾, soit dans moins de 8 mois, le GDPR (General Data Protection Regulation), Règlement Général européen de Protection des Données personnelles, s'inscrit dans le prolongement de la loi Informatique et Liberté française. Il élargit les droits des Citoyens sur la gestion de leurs données personnelles.

(1) A partir de cette date, NDLR

GDPR – Quels droits pour le Citoyen ?

En premier lieu voici listés les droits du citoyen tels qu'ils sont définis dans les articles du GDPR

- Consentement : aucune donnée ne peut être collectée sans accord explicite et positif (art. 7) ;
- Transparence : droit de savoir à quoi servent ses données (art. 13 et 14) ;
- Droit d'accès et de rectification : droit de consultation et de modification (art. 15 et 16) ;
- Droit à l'oubli : suppression et limitation de conservation des données (art. 17) ;
- Portabilité : droit de récupérer ses données pour les transférer ailleurs (art. 20) ;
- Droit d'opposition : la personne concernée a le droit de s'opposer à tout moment au traitement de ses données (art 21) ;
- Profilage : droit de ne pas faire l'objet d'une décision fondée exclusivement sur un traitement automatisé (art. 22) ;
- Minimisation : gestion des données uniquement nécessaires à la finalité réelle (art. 5) ;
- Sécurité : droit de voir ses données systématiquement protégées (art. 32) ;
- Notification : droit à l'information en cas de fuite de données (art. 33).

GDPR – Quels devoirs pour l'Entreprise ?

Le GDPR oblige l'Entreprise à satisfaire les nouveaux droits des citoyens décrits précédemment, mais également à mettre en œuvre des actions qui garantissent la bonne application des principes de gestion efficiente des données personnelles, y compris les moyens de contrôle et surveillance.

Il faut assurer des actions d'organisation, notamment la nomination d'un Data Protection Officer (art. 37), chargé de mettre en œuvre la conformité au GDPR.

Il est nécessaire de mettre en place un principe de responsabilisation du responsable des traitements et de ses sous-traitants (art. 24) et de réaliser des actions pour démontrer que les traitements respectent le GDPR. Il faut aussi faire appliquer des directives et contrôles, notam-

ment la tenue d'un Registre des traitements (art. 30) qui recense de manière précise les activités et actions ou événements concernant les données personnelles.

Une approche méthodologique permettant la mise en place dès la conception du respect du GDPR (Privacy by Design, art. 25) est indispensable avec la mise en place des seuls traitements nécessaires (Privacy by default).

La mise en œuvre d'analyses de risques, au travers notamment d'analyse d'impact (DPIA – Data Protection Impact Analysis, art. 34) doit être conduite, afin d'évaluer les risques de toute activité ayant un impact potentiel important sur la protection des données personnelles.

La mise en œuvre de mesures de sécurité (art. 32) doit être déployée, afin de garantir la sécurité du traitement (cryptage, anonymisation, confidentialité, intégrité, restriction de partage et d'utilisation des données), ainsi que le test, l'analyse et l'évaluation régulière de l'efficacité des mesures de sécurité.

Il faudra déclarer la notification d'incidents de sécurité « Data Breach Notification » (art. 33 et 34) auprès de l'autorité de contrôle compétente et des personnes concernées.

QUELS IMPACTS POUR L'ENTREPRISE ?

Le GDPR a un impact majeur sur tout le cycle de vie des données à caractère personnel. [1] Ceci nécessite donc de renforcer en particulier la gouvernance des données et de parfaitement maîtriser les processus Métiers et les données associées.

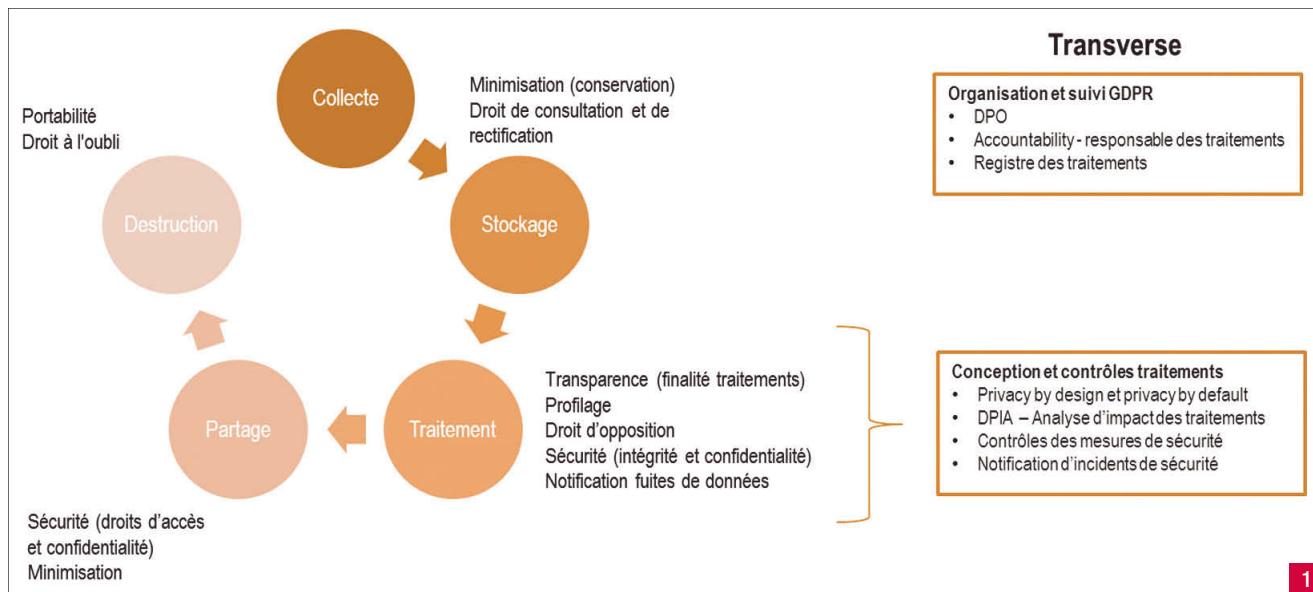
De manière plus générale, l'impact de la GDPR est majeur, et nécessite de mettre en œuvre une approche globale de transformation sur les 8 axes que sont :

La stratégie Data : Il s'agit de définir une stratégie Data et de création de valeur respectant le GDPR, par la collecte et la valorisation des données dans le cadre d'une finalité claire et transparente (approche Data Centric).

La gouvernance des données, directives et culture : une gouvernance, portée par le top management est indispensable, afin de donner la bonne direction à toute l'entreprise, et s'assurer une surveillance optimale de la mise en œuvre. Cette démarche sera initiée par la définition de principes et directives liés au GDPR, un code de conduite sur la gestion des données personnelles, une sensibilisation de tous les acteurs et leur implication dès le début dans la conduite du changement de culture.

L'organisation, les ressources et compétences : cette démarche globale de transformation doit être assurée et coordonnée par le DPO. L'implication du ou des Responsables des traitements et autres acteurs clés doit être garantie (juridique, RSSI, responsables processus Métiers, ...). Elle impose une définition claire de leurs rôles et missions, de leur positionnement et de leurs responsabilités, au travers d'un véritable management collaboratif, afin de garantir une cohérence et une complémentarité des acteurs. Leur montée en compétence doit être assurée au travers de formations et d'accompagnements.

Les Processus Métiers : la conformité au GDPR commence par la maîtrise des processus Métiers, idéalement pilotés par des responsables de processus. La mise en œuvre d'un registre des traitements et la réalisation de DPIA sur les traitements sensibles assureront une maîtrise des risques ainsi qu'une transparence vis-à-vis de la CNIL. L'intégration dès la conception des processus et traitements, des contrôles d'accès, des règles de sécurité, et des règles GDPR (Privacy by Design, Security by Design) sera réalisée en conformité avec les directives et codes de conduite relatifs au traitement des données personnelles. La réflexion couvrira également l'intégration des données personnelles, gérées par le nouveau processus ou traitement, dans les activités et applications de gestion des droits des citoyens.



Les activités GDPR : l'ensemble des activités relatives au GDPR est à définir et déployer, que ce soit la gestion des droits par le citoyen (consultation, effacement, transfert) ou le pilotage opérationnel de la démarche au sein de l'Entreprise (suivi des activités GDPR et gestion des risques, revues, audits et contrôles, suivi de sous-traitants, reporting, communication, gestion des alertes, ...).

La gestion des données et de l'information : une cartographie générale des données personnelles, leur structuration et modalités de stockage garantira une mise en œuvre de processus Métiers et traitements cohérents entre eux. L'information interne couvre la communication, ainsi que les indicateurs de suivi et de pilotage du projet GDPR dans l'Entreprise. L'information externe s'adresse aux citoyens sur leurs droits et données personnelles gérées par l'Entreprise et les incidents, et s'adresse également aux autorités compétentes (CNIL).

L'implémentation technique : l'architecture du SI de l'Entreprise doit permettre l'intégration des exigences du GDPR. Une architecture Data Centric facilitera cette adaptation. D'autre part le développement d'applications est indispensable pour gérer de manière optimale la gestion de la relation avec les citoyens et de leurs droits (information, modification, transfert, effacement, ...).

Le risque majeur est un afflux de demandes, en particulier une concentration au début, et une impossibilité pour l'Entreprise de répondre dans les délais.

L'utilisation d'outils, dont de nombreux restent à développer, doit faciliter la réalisation d'audits

techniques automatisés (audit de code, test d'applications Web ou smartphone, ...).

Le juridique : une forte implication du directeur juridique et de son équipe est nécessaire pour coordonner, contrôler et valider au niveau juridique les actions liées au GDPR et les contrats avec les partenaires et sous-traitants. Une veille juridique est indispensable, renforcée par l'accompagnement d'un juriste spécialisé externe, notamment au démarrage.

GDPR – UNE OPPORTUNITE DE CREATION DE VALEUR DURABLE POUR L'ENTREPRISE

Le GDPR est une démarche de maturité autour de la Data. Le législateur européen aurait pu se contenter de définir les droits des citoyens en matière de données personnelles et les sanctions pour les Entreprises qui n'appliquent pas ces obligations. Il a bien compris que le succès de l'application de ce règlement passait par la mise en œuvre de sanctions importantes, mais aussi par la forte incitation à mettre en œuvre des bonnes pratiques de gouvernance et de management des données personnelles, qui s'appliquent tout naturellement également aux autres données gérées.

Dans ce cadre, il a formalisé en articles la mise en œuvre obligatoire de ces bonnes pratiques afin de garantir une maturité minimale des Entreprises. Ces bonnes pratiques visent à :

- Consolider ou développer une cartographie des traitements et des données personnelles et mieux évaluer les risques (register des traitements, DPIA, ...);

- Développer et organiser une démarche d'Entreprise (implication top management, DPO, responsabilisation des acteurs, gouvernance, plan de transformation, ...);
- Améliorer la culture « Data » de l'Entreprise (sensibilisations de tous, code de conduite, ...);
- Améliorer la conception des traitements et applications associées (Privacy by Design, ...);
- Améliorer la sécurité (Security by Design, accès, confidentialité, ...) et les contrôles associés
- Améliorer la maîtrise des sous-traitants;
- Améliorer la surveillance, les revues et contrôles internes;

Le GDPR est contributeur à la transformation digitale et à la création de valeur. Ainsi, dans le cadre d'une généralisation des approches « Data Centric », ces bonnes pratiques deviennent des piliers de la transformation digitale des Entreprises :

- Elles contribueront à l'amélioration de la Gouvernance et à la Culture de la donnée;
- Elles faciliteront la maîtrise des risques, la qualité des données, ainsi que la sécurité et le contrôle des données;
- Elles seront un vecteur d'optimisation des ressources, techniques mais aussi humaines, en développant leurs compétences sur le sujet.

Au final, cette approche contribuera activement à la création de valeur durable pour l'Entreprise :

- Par une meilleure performance des traitements;
- Par une meilleure connaissance de ses données, et le développement de stratégies nouvelles de valorisation;
- Par une communication client transparente et responsable, vecteur d'une relation de confiance et durable.

Faire un jeu Android avec HTML5 et Pixi.js en moins de 100 de lignes de code !



• Fabien Pigère
freelance html5 en remote, spécialisé dans l'optimisation et le multimédia.

Faire un jeu est le rêve de pas mal de programmeurs débutants. Avant il était nécessaire d'apprendre le C++, ce qui était long et périlleux. Avec la montée en puissance des machines et des API HTML5, cela devient abordable à tous les programmeurs Front. Nous allons donc voir comment réaliser un jeu 2D pour Android (qui marchera aussi sous vos navigateurs, bien sûr !).

Pourquoi Pixi.js ?

Pixi.js simplifie la programmation du canvas, et apporte quelques outils (fonte bitmap, gestion des clicks sur les objets...). Le canvas étant une vieille balise, très bien documentée et très puissante d'elle-même, pourquoi prendre cette librairie ? En fait Pixi.js permet d'utiliser, si présent, l'API WEBGL d'HTML5 (la version web du célèbre OpenGL) permettant un gain de vitesse énorme, tout en utilisant le canvas classique si WebGL n'est pas présent, ce qui peut arriver sur les vieux portables. Pixi.js en est à sa version 4, très mature, et se détache de plus en plus d'AS3 (le langage de flash) dont il était inspiré auparavant. Pour apprendre à l'utiliser, nous ferons un jeu de tir au canard en 2D, qui marchera sur navigateurs mais aussi sur portable (Exemple pour Android dans cet article).

Etape 1 : Comment installer Pixi.js ?

Plusieurs méthodes existent :

Par NPM :

```
npm install pixi.js
```

Par GitHub : <https://github.com/pixijs/pixi.js>

Par CDN :

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/pixi.js/4.5.1/pixi.min.js"></script>
```

Ou directement sur son site : <http://www.pixijs.com/>

Comme notre jeu permettra de jouer offline, nous prendrons la dernière méthode !

A ce stade, nous faisons une page quasiment vide ! Il n'y aura qu'une base HTML dans notre jeu, le fameux canvas, et nous ferons tout nous-mêmes (adieu problème de css) :

```
<!DOCTYPE html>
<html>
<head>
<title>M00000TS</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<script src="pixi.min.js"></script>
<style>
body, html {
height: 100%;
display: grid;
```

```
border: 0px;
margin: 0px;
}
canvas {
margin: auto;
}
</style>
</head>
<body>
</body>
</html>
```

Ce CSS permet au jeu d'être toujours centré (il y a beaucoup d'autres façons de faire bien sûr).

Etape 2 : Commencer à afficher

Pour cette étape, cherchez donc un canard.png (PNG à cause de la transparence), par exemple sur Google image (attention au droit d'auteur !). Nous allons commencer le JS par la création d'une application très simple :

```
document.addEventListener("DOMContentLoaded", ready); // on attend que tout soit prêt
var app ; // global !
function ready()
{
    app = new PIXI.Application({width: 1000, height: 1000});
    var txt = PIXI.Texture.fromImage('canard.png');
    var canard = new PIXI.Sprite(txt);
    canard.interactive = true;
    app.stage.addChild(canard);
}
```

Ceci permettra d'afficher le canard bêtement en 0,0, et nous permet de vérifier que tout marche bien.

Nous tirerons sur le canard grâce à nos doigts sur le portable il faut donc préciser à Pixi.js que l'objet est interactif.

Nous allons réunir ce code pour afficher des centaines (c'est la saison de la chasse ?) de canards sur notre écran :

```
// fonction qui renvoie un nombre entre [min,max]
function rand(min, max)
{
```

```

    return Math.floor(Math.random() * (max - min + 1) + min);
}

function createSprite(texture, x, y, sprite)
{
    sprite = new PIXI.Sprite(txt);
    sprite.x = x;
    sprite.y = y;
    sprite.interactive=true ;
    app.stage.addChild(sprite);
    return sprite;
}

var txt = PIXI.Texture.fromImage('canard.png');
var TabCanard=[];
for(var i=0;i<200;i++)
    TabCanard.push(createSprite(txt,rand(0,1000),rand(0,1000));

```

Voilà 200 petits canards affichés à l'écran !

Etape 3 : Animer le tout !

Pixi.js fournit une méthode d'animation reposant sur le `requestAnimationFrame` si présent, ou bien un `setTimeOut`. Nous allons utiliser cette méthode pour animer nos 200 petits canards, cependant nous utiliserons une façon intelligente de le faire. En effet, 200 est un nombre arbitraire qui pourrait être lent sur un portable. Nous retirerons donc des canards si l'affichage est lent, un peu d'IA en somme !

```

var debutb = +new Date(); // TIPS ! + permet de transformer directement en nombre
app.ticker.add(function ()
{
    var d = +new Date() - debutb;
    if (d > 50 && TabCanard.length > 0)
    {
        var aux = TabCanard.pop();
        app.stage.removeChild(aux); // +de 50 ms ! Quelle horreur ! Retirons donc un canard
    }
    for (var i = 0; i < TabCanard.length; i++)
    {
        var c = TabCanard[i];
        c.x = c.x + rand(0, 4); // le canard va de gauche à droite
        c.y = c.y - rand(-2, 2); // et flottille un peu, pour éviter le chasseur
        if (c.x > 1000)
        { // le canard a disparu, on l'affiche à l'écran !
            c.x = rand(-10, 1000);
            c.y = rand(-10, 1000);
        }
    }
    debutb = +new Date();
});

```

Si le temps entre deux appels est supérieur à 50ms, on retire un canard car nous voulons rester fluide.

Etape 4 : Le tir !

Maintenant que nos canards sont libres de voler comme ils le souhaitent, tirs-les-leur dessus ! Pour cela, nous allons utiliser un évènement pointeur.

Ce type d'évènement concerne le Touch sur un tactile, et la souris sur un browser, d'où un seul code pour les deux plateformes. La syntaxe est :

```

sprite.on('pointerdown', function () {
    // mettre le code ici
});

```

Notre fonction de création de canard est donc maintenant :

```

var score=0 ;
var txt = PIXI.Texture.fromImage('canard.png');
var TabCanard=[];
for(var i=0;i<200;i++)
{
    var c=createSprite(txt,rand(0,1000),rand(0,1000));
    c.on('pointerdown', function () {
        score+=10 ;
        this.x=rand(-10,1000) ; // on replace le canard
        this.y=rand(-10,1000) ;
    });
    TabCanard.push(c);
}

```

Etape 5 : Afficher le score

Nous avons aussi besoin d'un score, pour cela Pixi.js permet d'utiliser une fonte présente sur votre page, ou une fonte bitmap :

```

var txtScore;
function majScore()
{
    if (txtScore !== undefined)
    {
        app.stage.removeChild(txtScore);
    }
    txtScore = new PIXI.Text("score : " + score);
    txtScore.x = 50;
    txtScore.y = 50;
    app.stage.addChild(txtScore);
}

var score = 0;

```

Nous avons donc de quoi afficher le score, ainsi que nos sprites !

Pourquoi la fonction `majScore` retire le score s'il existe ? Pour ne pas avoir deux textes au même endroit !

Le jeu marche, voici le code complet, qui donne l'image suivante (fig 1) :

```

<!DOCTYPE html>
<html>
<head>
    <title>Tir au canard !</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <script src="pixi.min.js"></script>
    <script>
        function rand(min, max)
        {

```



```
    return Math.floor(Math.random() * (max - min + 1) + min);
}
```

```
function createSprite(txt, x, y, sprite)
{
    sprite = new PIXI.Sprite(txt);
    sprite.x = x;
    sprite.y = y;
    sprite.interactive = true ;
    app.stage.addChild(sprite);
    return sprite;
}
```

```
var txtScore;
function majScore()
{
    if (txtScore !== undefined)
    {
        app.stage.removeChild(txtScore);
    }
    txtScore = new PIXI.Text("score : " + score);
    txtScore.x = 50;
    txtScore.y = 50;
    app.stage.addChild(txtScore);
}
```

```
var score = 0;
function initGFX()
{
    majScore();
    var TabCanard = [];
    var txt = PIXI.Texture.fromImage('canard.png');
    for (var i = 0; i < 200; i++)
    {
        var c = createSprite(txt, rand(0, 1000), rand(0, 1000));
        c.on('pointerdown', function () {
            score += 10 ;
        });
    }
}
```

```
    majScore();
    this.x = rand(-100, 10) ;
    this.y = rand(0, 1000) ;
});
TabCanard.push(c) ;
}
var debutb = +new Date(); // TIPS ! + permet de transformer directement en nombre
app.ticker.add(function ()
{
    var d = +new Date() - debutb;
    if (d > 50 && TabCanard.length > 0)
    {
        var aux = TabCanard.pop();
        app.stage.removeChild(aux);
    }
    for (var i = 0; i < TabCanard.length; i++)
    {
        var c = TabCanard[i];
        c.x = c.x + rand(0, 4); // le canard va de gauche à droite
        c.y = c.y - rand(-2, 2); // et flottille un peu, pour éviter le chasseur
        if (c.x > 1000)
        { // le canard a disparu, on l'affiche à l'écran !
            c.x = rand(-100, 10);
            c.y = rand(0, 1000);
        }
    }
    debutb = +new Date();
});
}
var w = 1000;
var h = 1000;
document.addEventListener("DOMContentLoaded", ready); // on attend que tout soit prêt

var app ; // global !
function ready()
{
    app = new PIXI.Application({width: w, height: h});
    app.renderer.backgroundColor = 0xaaaaaa ;
    document.body.appendChild(app.view);
    initGFX();
}
</script>
<style>
body, html {
    height: 100%;
    display: grid;
    border: 0px;
    margin: 0px;
}
canvas {
    margin: auto;
}
</style>
</head>
<body>
```

```
</body>
</html>
```

Etape 6 : Transformation en application Android

Nous avons besoin d'une grande étape de ligne de commande etc, bon courage d'avance.

Avant de charger Cordova, il faut installer la JDK :

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

Ainsi que le dernier SDK d'Android :

<https://developer.android.com/studio/index.html#downloads>

Et ne prenez que l'outil de ligne de commande, lancez le SDK manager et prenez la dernière version du SDK.

Il faut mettre dans le PATH les deux répertoires, soit, dans mon cas :

```
ANDROID_HOME=C:\android-sdk ; JAVA_HOME=c:\java\jdk1.8.0_51
```

Maintenant il faut NPM, qu'on obtient en chargeant Node : <https://nodejs.org>
Une fois tout ceci fait, l'installation devient rapide, le plus dur est fait :

```
NPM cordova -g
```

Maintenant positionnez-vous dans le répertoire de votre projet et créez le projet cordova :

```
cordova create duck com.pigere.duck duck
cd duck
cordova platform add android
```

Allez ensuite dans le répertoire www pour remplacer l'index.html par notre jeu (renommé au cas où bien sûr le fichier en index.html), puis :

```
cordova run
```

Et si votre téléphone est branché et en mode développement (bidouille dépendant de votre version d'Android, cela dépasse le cadre de cet article) le jeu s'affiche, à vous de jouer !

Etape 7 : Amélioration possible

Par défaut le jeu est en 1000,1000 ce qui est très agréable sur un ordinateur, mais trop énorme pour la plupart des portables, il faut donc redimensionner l'écran pour l'adapter :

```
var w = window.innerWidth;
var h = window.innerHeight;
app = new PIXI.Application({width: w, height: h});
w = Math.min(w, 1000);
h = Math.min(h, 1000);
app.stage.scale.x = w / 1000;
app.stage.scale.y = h / 1000;
var el = document.querySelector('canvas');
```

```
el.setAttribute('width', w + "px");
el.setAttribute('height', h + "px");
```

Ne pas oublier de faire de même dans évènement window.resize, notre utilisateur pouvant tourner son téléphone en mode paysage.

De même nous devons charger cordova pour pouvoir utiliser les plugins associés. Ceci se fait en mettant un `<script src="cordova.js"></script>` à l'intérieur de l'html, il faut ensuite attendre l'évènement deviceready en plus du documentready. Cependant, cela entraîne une duplication des sources, en une version mobile et navigateur. Pour détecter si on est dans une app portable, plusieurs méthodes possibles, j'utilise la suivante :

```
document.addEventListener("DOMContentLoaded", phone);
var isPhone=false;
function phone()
{
  if (document.location.protocol === 'file:')
  {
    isPhone = true;
    document.addEventListener("deviceready", ready, false);
  } else
  {
    ready();
  }
}
function ready()
{
// ici votre jeu
}
```

CONCLUSION

Nous avons créé un jeu "à l'ancienne" en moins de 100 lignes ! Puis nous l'avons transformé en application Android, un simple "cordova run" permet de le tester. Nous avons étudié en surface Pixi.js, que je vous recommande, ou bien utilisez son grand frère, Phaser.js, qui contient d'autres outils pour créer des jeux (<https://phaser.io/>). Il faut également une IHM, j'utilise personnellement celle de EZGUI (<https://github.com/Ezelia/EZGUI>). D'autres améliorations possibles ? Gérer le fullscreen pour le Desktop, mais aussi Android (pour faire disparaître la barre d'état), des notifications (pour relancer votre joueur), faire vibrer le téléphone sur un bon shoot, etc.), les plugins Cordova sont disponibles : <https://cordova.apache.org/plugins/>

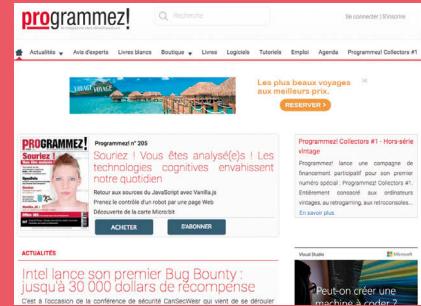
Il en existe plus de 2500 ! De quoi varier les plaisirs ! Cependant avant d'installer un plugin pour faire quoi que ce soit, vérifié d'abord qu'il n'existe déjà en HTML5 pur, évitant de faire grossir l'application : <https://whatwebcando.today/>

Enjoy !

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



Projet Rome

• Charles DE VANDIERE
Développeur chez
Infinite Square
blogs.infinitesquare.com



Le Projet Rome a pour but d'offrir une expérience cross-device. A partir de la version Anniversary de Windows 10, il est possible de commencer une activité sur un appareil et de la poursuivre sur un autre. Le contexte de l'application est partagé entre les appareils connectés. Par exemple, comme l'explique la documentation de Microsoft, vous pouvez écouter la radio dans votre voiture grâce à votre téléphone mobile, puis une fois rentré chez vous, transférer la lecture de la radio sur votre Xbox pour mieux en profiter.

Principe de fonctionnement

Architecture

La communication entre deux appareils est architecturée de manière très simple. Les appareils s'inscrivent dans le Cloud (étape 1 sur le schéma ci-contre). La liaison se fait par le compte Microsoft de l'utilisateur. Puis ils peuvent communiquer librement (étape 2 sur le schéma ci-contre).

Découverte

Pour que l'appareil client puisse communiquer avec l'appareil hôte, il faut d'abord qu'il le découvre. Cette découverte peut se faire de quatre manières différentes :

- Cloud : les appareils sont inscrits dans le Cloud Microsoft et peuvent se connecter par ce biais ;
- USB ;
- Bluetooth ;
- Réseau local.

Vous pouvez filtrer la recherche des appareils grâce au `IRemoteSystemFilter`. Voici les trois types de filtres disponibles :

- Type de connexion (Cloud, réseau local, Bluetooth, USB) ;
- Type d'appareil (Desktop, Phone, Xbox, ...) ;
- Statut de l'appareil (disponible, non disponible).

Pour découvrir un autre appareil dans votre application, vous devez déclarer la capacité « Remote System » dans le manifest de l'application.

Ensuite, il vous faut demander l'autorisation de l'utilisateur pour accéder à un autre appareil :

```
await RemoteSystem.RequestAccessAsync();
```

Enfin, vous pouvez lister les appareils disponibles de la manière suivante :

```
var watcher = RemoteSystem.CreateWatcher();
watcher.RemoteSystemAdded += OnRemoteSystemAdded;
watcher.RemoteSystemRemoved += OnRemoteSystemRemoved;
```

Communication

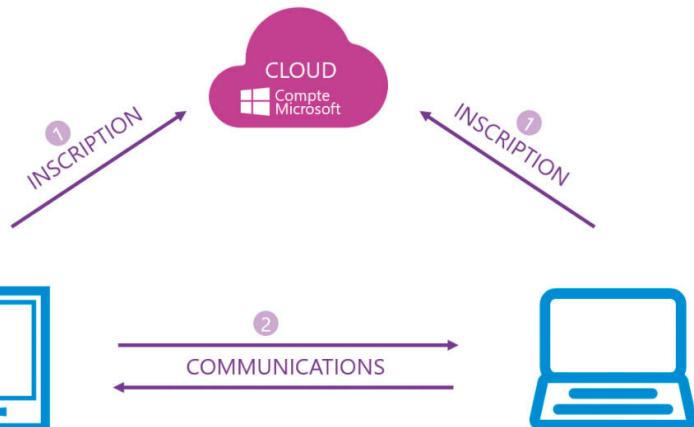
Les appareils peuvent communiquer de deux manières différentes :

- Via un protocole : le développeur définit un protocole à utiliser et la communication se fait via des URI ;
- Via un App Service : l'application cliente consomme l'App Service de l'application hôte pour faire des actions.

Un App Service est un service qui s'exécute dans le contexte de l'application hôte et qui peut être consommé par l'application cliente. Le service s'exécute en arrière-plan.

Ouverture d'une URI (utilisation d'un protocole)

Tout d'abord, il faut choisir la cible sur laquelle on souhaite ouvrir l'URI :



```
var targetRs = RemoteSystems
    .FirstOrDefault(r => r.Status == RemoteSystemStatus.Available);
var request = new RemoteSystemConnectionRequest(targetRs);
```

Puis, lancer la requête :

```
RemoteLauncher.LaunchUriAsync(request, uri);
```

Communication via un App Service

Côté hôte :

Éditez le manifest à la main pour déclarer l'App Service :

```
<uap:Extension Category="windows.appService">
<uap3:AppService Name="com.infinitesquare.CustomRain"
    SupportsRemoteSystems="true" />
</uap:Extension>
```

Puis, créez l'App service de manière classique.

Côté client :

Commencez par créer une requête de connexion comme précédemment puis ouvrez une connexion à l'App Service :

```
var connection = new AppServiceConnection
{
    AppServiceName = "com.infinitesquare.CustomRain",
    PackageFamilyName = Package.Current.Id.FamilyName
};
await connection.OpenRemoteAsync(connectionRequest);
```

Enfin, utilisez l'App Service de manière classique.

Expérience cross-plateformes

Lors de l'annonce du projet Rome, il n'était possible de l'utiliser que sur Windows avec l'Anniversary Update. Depuis, le projet a été porté sur

Android, puis très récemment sur iOS comme annoncé lors de la BUILD 2017 de Microsoft. L'assistant personnel de Microsoft Cortana, qui est présent sur les trois systèmes d'exploitation, utilise également le Projet Rome pour interagir entre tous les appareils de l'utilisateur.

Toutefois, les appareils fonctionnant avec d'autres systèmes que Windows, c'est-à-dire les téléphones et tablettes Android ou iOS, ne peuvent agir qu'en tant que clients. En effet, le mécanisme de fonctionnement d'un appareil hôte est très fortement lié à Windows. L'appareil doit être enregistré dans le Cloud, lié au compte Microsoft de l'utilisateur et exposer un App Service ou un protocole qui sont propres à Windows. De ce fait, il est pour le moment impossible aux appareils fonctionnant sous Android ou iOS de devenir des hôtes. Néanmoins, cela n'exclut pas certains scénarios d'utilisation cross-plateformes très utiles. Par exemple, dans le cas d'une application multimédia, on peut imaginer la déportation de l'utilisation depuis un iPhone vers une console Xbox pour profiter d'un écran beaucoup plus grand.

Microsoft Graph

Au-delà des SDK disponibles pour les différents systèmes d'exploitation, le projet Rome fonctionne aussi avec les API Graph de Microsoft.

Continue on PC

La première application qui utilise le Projet Rome

Le 17 octobre prochain, Windows 10 va recevoir sa nouvelle mise à jour nommée Fall Creator Update. Dans cette mise à jour, Microsoft a prévu de nouvelles interactions entre les téléphones portables et les PC Windows. Il suffit de télécharger l'application "Continue on PC", disponible sur iOS et Android et de s'authentifier avec son compte Microsoft.

Grâce à cette application, vous allez pouvoir naviguer sur internet avec votre téléphone, et continuer sur votre ordinateur très simplement.

- [1] Dans le menu Partager, choisissez « Continue on PC »
- [2] Choisissez « Continuer maintenant »
- [3] Sélectionnez l'un de vos PC dans la liste proposée
- [4] Microsoft Edge s'ouvre automatiquement sur votre ordinateur avec la même URL

Il s'agit d'une première application plutôt pratique utilisant le Projet Rome. Espérons que d'autres arriveront bientôt pour que nous puissions profiter de toutes ces nouvelles possibilités.

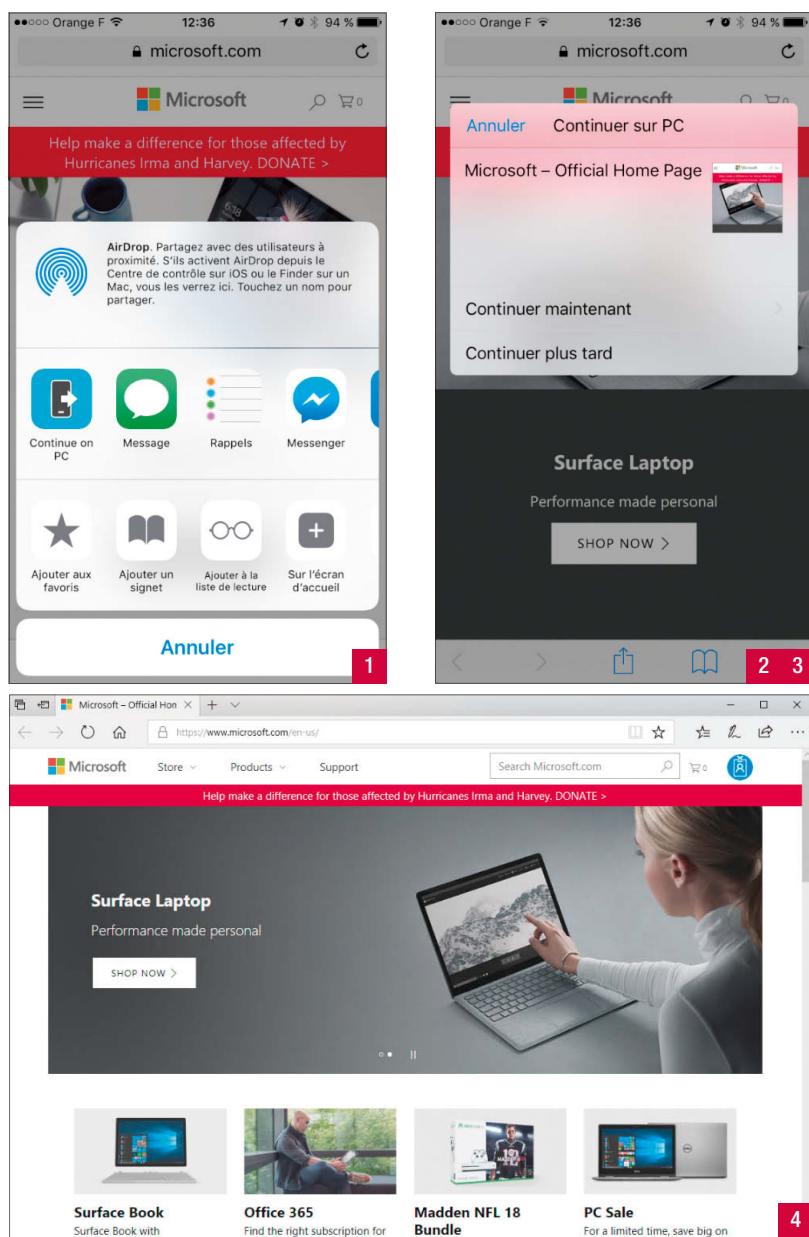
Pour rappel, les API Graph permettent d'accéder à tous les services en ligne de Microsoft (OneDrive, Outlook, Azure, Office, ...) par une simple requête REST.

Vous pouvez donc également envoyer un message à une application sur l'un de vos appareils par une requête REST.

Pour aller plus loin

Si vous souhaitez en savoir plus sur l'implémentation du Projet Rome, voici quelques liens utiles :

- Documentation officielle : <https://docs.microsoft.com/en-us/windows/uwp/launch-resume/connected-apps-and-devices>
- Article de blog Windows sur Le Project Rome : <https://blogs.windows.com/buildingapps/2017/05/16/project-rome-driving-user-engagement-across-devices-apps-platforms/#qeallIEQT194gylc.97>
- Article de blog Windows sur l'implémentation sur Android : <https://blogs.windows.com/buildingapps/2017/03/23/project-rome-android-update-now-app-services-support/#2bw32tCmV1uraFBs.97>



.Net Standard 2.0 : le futur de .Net



Christophe Gigax
MVP Visual Studio
and Development
Technologies

La sortie officielle de .Net Standard 2.0 est enfin survenue mi-Août, après plusieurs mois de développement par les équipes de Microsoft afin d'enrichir les APIs. Pour rappel, le but de ce standard est d'offrir un jeu d'APIs universel entre toutes les plateformes, qu'elles soient web, mobile ou desktop. Grâce à cela, le développeur sera capable de partager du code entre les différentes plateformes qu'offre l'écosystème .NET, Xamarin, .NET Core, .NET Framework... Cette nouveauté peut poser plusieurs questions : vers quel futur allons-nous ? Comment intégrer .Net Standard dans nos projets ? Est-ce la fin des PCL ?

Qu'est-ce que .Net Standard

.Net Standard est un jeu d'APIs unique que les plateformes .NET doivent implémenter pour être utilisées. Elle permet de résoudre le problème du partage de code dans l'écosystème Microsoft de plus en plus grandissant. La version 2.0 sera utilisable par les plateformes suivantes :

- .NET Framework
- .NET Core
- Xamarin

Concernant .NET Core, .Net Standard va permettre d'apporter de nombreuses APIs qui manquaient depuis le début du lancement du framework.

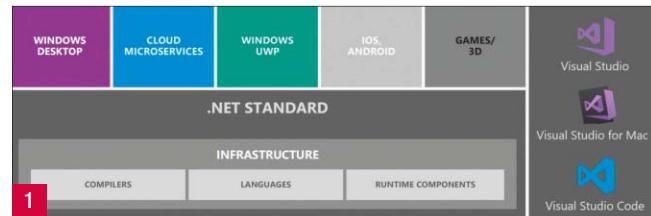
Les avantages

Il est clair que ce nouveau standard va permettre d'unifier la plateforme .NET autour d'APIs communes, mais ce n'est pas tout. Le développeur va pouvoir maximiser son partage de code au sein de ses projets, améliorant ainsi la productivité de l'équipe de développement. Net Standard permet également d'éviter les différentes fragmentations qui pourraient survenir entre les différentes plateformes .NET. [1]

La version 1.6 de .Net Standard implémentait à peu près 13000 APIs. Aujourd'hui, la version 2.0 implémente environ 32000 APIs, couvrant ainsi la quasi-totalité des demandes les plus importantes en termes de fonctionnalités. Cela facilite notamment le portage vers une librairie Net Standard. Avec ce nombre d'APIs implémentées, cela rend approximativement 70% des librairies NuGet compatibles Net Standard, ce qui est loin d'être négligeable.

Au niveau des outils, le développeur a le choix. Sur Windows, la dernière mise à jour de Visual Studio 2017 (la 15.3) supporte parfaitement .Net Standard et permet de créer des librairies facilement. Sur Mac, Visual Studio for Mac permet également de créer des librairies .Net Standard. Il en va de même pour l'éditeur de JetBrains : Rider. Indépendamment de l'IDE que vous désirez utiliser, vous devez toujours installer le SDK .NET Core séparément, qui embarque .Net Standard 2.0. L'utilitaire **dotnet**, intégré au SDK, permet de créer et référencer une librairie .Net Standard 2.0. Les IDE ne font qu'encapsuler les commandes de **dotnet**. Cela veut dire que vous pouvez même tout gérer vous-même à partir de Visual Studio Code (pour la partie développement), avec, à côté, une console pour les commandes **dotnet** et ainsi la gestion de projet. Cette liberté d'utilisation est un vrai point fort de .Net Standard.

Pour finir, la nouvelle version du standard embarque une nouveauté très intéressante : la compatibilité avec le .NET Framework. Concrètement, cela veut dire qu'une librairie en .Net Standard 2.0 sera capable de référencer une librairie .NET Framework, c'est-à-dire une grande majorité des librairies .NET sur NuGet. Dans certains cas, cela peut ne pas fonctionner (en fonction des APIs utilisées dans la librairie en .NET Framework), mais pour la majorité des cas cela peut convenir.



L'unification des plateformes avec .Net Standard

.NET Standard	1.0	1.1	1.2	1.3	1.4	1.5	1.6	2.0
.NET Core	1.0	1.0	1.0	1.0	1.0	1.0	1.0	2.0
.NET Framework	4.5	4.5	4.5.1	4.6	4.6.1	4.6.1	4.6.2	4.6.1 vNext
Mono	4.6	4.6	4.6	4.6	4.6	4.6	4.6	5.4
Xamarin.iOS	10.0	10.0	10.0	10.0	10.0	10.0	10.0	10.0
Xamarin.Mac	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.8
Xamarin.Android	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.5
Universal Windows Platform	10.0	10.0	10.0	10.0	10.0	vNext	vNext	vNext
Windows	8.0	8.0	8.1					
Windows Phone	8.1	8.1	8.1					
Windows Phone Silverlight	8.0							

Tableau des versions de .NET Standard

Les inconvénients

L'inconvénient majeur de l'arrivée de .Net Standard 2.0 est la migration des applications ou des librairies existantes. Afin de cibler toujours un maximum de développeurs et de rester à la pointe technologique dans l'écosystème Microsoft, les développeurs vont devoir faire des efforts supplémentaires pour porter leurs codes sur .Net Standard 2.0. D'autant plus que rien ne garantit à 100% la compatibilité avec leur code actuel, puisque pour illustration, il reste toujours 30% des paquets NuGet non compatibles .Net Standard 2.0, ce qui n'est pas négligeable. Ensuite, Microsoft a clairement fait un choix vers le futur pour cette version et délaisse complètement Silverlight et Windows Phone 8.0 / 8.1. D'ailleurs, .Net Standard n'était compatible qu'avec la version 1.2. Les systèmes tournant sur ces plateformes vont ainsi devoir également porter leurs codes vers les nouvelles plateformes (UWP, Xamarin, .NET Core ...).

Quelle version choisir

Afin de savoir quelle version le développeur peut cibler, il suffit de se référer au tableau ci-dessous. [2]

Chaque colonne représente une version de .Net Standard, et chaque ligne représente les différentes plateformes qui implémentent le standard. La version indiquée dans la cellule indique ainsi la version *minimum* que la plateforme doit avoir pour utiliser la version de .Net Standard spécifiée par la colonne. On peut voir que .Net Standard supporte les der-

nières versions d'Android et d'iOS via Xamarin.iOS et Xamarin.Android. Il en va de même pour .NET Core : uniquement la version 2 est compatible avec la version .Net Standard 2.0. Avec tout ceci, comment déterminer la version de ma librairie ? La réflexion est simple : pour cibler un maximum de plateformes, il faut cibler les versions basses de .Net Standard. Sinon, pour cibler plus d'API au sein de la librairie, il faut choisir les versions hautes de .Net Standard. Le tout est de savoir le *scope* de votre projet : doit-il cibler beaucoup de plateformes, où ai-je besoin de beaucoup de fonctionnalités ?

Au niveau du versioning, il existe 2 règles importantes :

- **Additive** : les versions hautes intègrent toutes les APIs des versions basses, et aucun *breaking change* n'est introduit entre 2 versions ;
- **Immutable** : une fois sortie, une version de .Net Standard ne bougera plus. Ceci permet de garantir la pérennité des librairies que vous produisez et que vous consommez. De manière générale, Microsoft préconise de cibler la plus petite version de .Net Standard possible, c'est-à-dire celle qui fournit les APIs suffisantes à votre application. Ensuite, version par version, il suffit d'augmenter la version de .Net Standard jusqu'à ce que le projet ne compile plus.

D'un point de vue des packages, la distribution de .Net Standard se fait par paquet NuGet. Un paquet NuGet peut cibler un ou plusieurs frameworks par nature. Concernant, Net Standard, Microsoft a créé un framework à part entière s'intitulant '.NET Standard'. Dans le **.csproj**, il suffit de cibler le framework .NET Standard utilisant la notation *netstandard* suivie de la version voulue, par exemple *netstandard1.6*.

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <TargetFramework>netstandard1.6</TargetFramework>
  </PropertyGroup>
  <ItemGroup>
    <PackageReference Include="NETStandard.Library" Version="2.0" />
  </ItemGroup>
</Project>
```

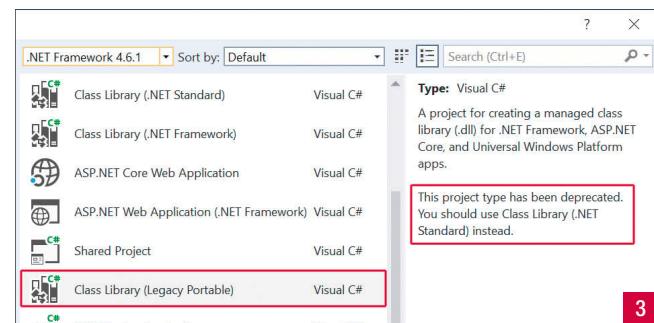
Un point important ici est l'utilisation du meta-package *NETStandard.Library*. Ce dernier n'est pas un paquet NuGet qui contient directement les DLLs, mais simplement un package qui référence d'autres packages. Ce dernier permet ainsi de référencer des packages différents selon la version. Par exemple, le package *System.Security.Cryptography.X509Certificates* est disponible uniquement en version 1.3 ou au-dessus, cela veut dire qu'il n'apparaît pas dans le métapackage si la librairie cible .Net Standard 1.0, 1.1 ou 1.2.

Le métapackage n'est ainsi qu'un ensemble de définition d'APIs, et c'est chaque plateforme qui va implémenter ces APIs selon ses spécifications.

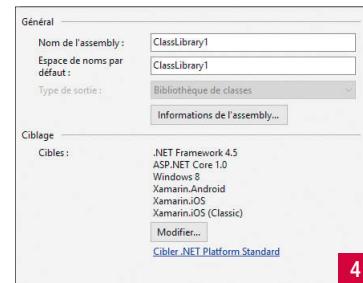
La fin des PCL ?

La question de la pérennité des PCLs est remise en cause avec .Net Standard. Microsoft a été clair à ce sujet lors de la sortie de la version 2 : les PCL sont officiellement dépréciés, et ne doivent plus être utilisés pour créer, dans le futur, des librairies .NET. Nous pouvons qualifier aujourd'hui ces 2 types de librairies de la manière suivante :

- Les PCLs existantes sont des '*profile-based PCLs*' c'est-à-dire des PCLs qui se restreignent d'elles-mêmes selon les plateformes ciblées via un jeu de profils ;
- Les librairies .NET Standard sont des '.NET Standard-based PCLs'. Elles restent des PCLs, mais basées sur le framework .NET Standard, ouvrant ainsi plus de possibilités par plateforme.



Visual Studio indique que les PCLs sont maintenant dépréciés



4

Lien de migration vers .NET Standard

Les différences sont notables :

- Les PCLs sont des intersections d'APIs implémentées sur des plateformes, alors que .NET Standard est un jeu d'APIs uniformisé entre toutes les plateformes ;
- .NET Standard propose un versioning linéaire, ce qui n'est pas le cas des profils des PCLs.
- Les PCLs sont liées à la plateformes Microsoft, avec le .NET Standard reste agnostique à la plateforme (non lié, ça reste une définition).

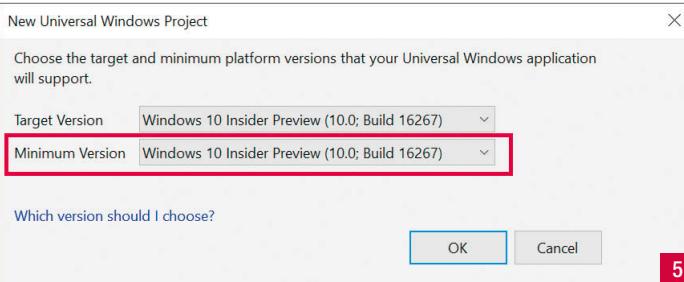
Comment migrer ma librairie

Avec la sortie de .Net Standard 2.0, la question de la migration se pose naturellement. Une chose est certaine : la migration du code de la librairie peut être plus ou moins douloureuse, suivant les APIs qui ont changé avec .Net Standard. Cette problématique est commune à toutes les librairies PCL d'aujourd'hui. Cependant, la migration du projet PCL en projet .Net Standard est très simple via l'IDE Visual Studio. Si vous avez une version inférieure à la 2017, le plus simple est de mettre à jour votre IDE car il intègre déjà un tas de processus de migration qui n'étaient pas intégrés par défaut dans Visual Studio 2015. Il vous aidera également à migrer vos **project.json** en **.csproj** facilement (si c'est le cas). Ouvrez votre librairie avec l'IDE, puis allez dans les paramètres du projet. Sous l'onglet **Bibliothèque** (ou **Library** selon votre langue), Visual Studio vous propose alors de cibler .NET Standard. [4]. Visual Studio va vous avertir que ce changement peut potentiellement rendre votre code incompatible avec certaines APIs que vous utilisez. Ce comportement est tout à fait normal puisque, pour rappel, .Net Standard est une spécification d'API unique implémentée sur chaque plateforme, alors que les PCLs sont une intersection des APIs définies sur chaque plateforme. Un message d'erreur peut apparaître vous demandant de désinstaller tous vos packages avant la migration. Cela veut dire que vous n'utilisez pas NuGet 3.0. Dans ce cas, après la migration, il suffit de réouvrir le projet et de réinstaller les paquets NuGet que vous aviez, compatibles avec la version de .Net Standard que vous souhaitez cibler.

Remarque : pour cibler .Net Standard 2.0, il est toujours nécessaire d'installer .NET Core 2.0 via un exécutable. Cela permet à Visual Studio de détecter que .Net Standard 2.0 est disponible pour votre projet.

Une fois l'opération terminée, on peut analyser les packages qui sont installés dans la nouvelle librairie :

- **NETStandard.Library** : c'est le métapackage de .Net Standard ;
- **Microsoft.NETCore.Portable.Compatibility** : ce package n'est pré-



Ces commandes ci-dessus permettent de :

- Créer un dossier `MonProjet` qui va contenir la solution entière ;
- Créer le `.sln` de la solution ;
- Créer le projet `.NET Core` dans `/src/MonProjet.Core`.

Nous devons ensuite référencer notre projet dans la solution via la commande suivante (au même niveau que le `.sln`) :

```
dotnet sln add src/MonProjet.Core/MonProjet.Core.csproj
```

Faisons de même pour les projets `Net Standard` et `xUnit`. Pour le projet `.Net Standard` :

```
mkdir src/MonProjet.Standard
cd ./src/MonProjet.Standard
dotnet new classlib
```

```
cd ../..
dotnet sln add src/MonProjet.Standard/MonProjet.Standard.csproj
```

```
cd ../MonProjet.Core
dotnet add reference ../MonProjet.Standard/MonProjet.Standard.csproj
```

Les 2 dernières lignes permettent de rajouter une référence sur notre projet `.NET Core` à notre projet `NET Standard`. Faisons ainsi de même avec notre projet `xUnit`, qui référencera notre projet `.NET Standard` pour effectuer les tests sur la librairie :

```
mkdir src/MonProjet.Test
cd ./src/MonProjet.Test
dotnet new xunit
dotnet add reference ../MonProjet.Standard/MonProjet.Standard.csproj

cd ../..
dotnet sln add src/MonProjet.Test/MonProjet.Test.csproj
```

Il suffit ensuite de lancer les commandes suivantes pour que la solution soit prête à être lancée :

```
dotnet restore
dotnet build
```

Et c'est tout ! La solution est maintenant prête à être utilisée. Elle met en œuvre les technologies suivantes :

- `.NET Core` pour la partie Web ;
- `.NET Standard` pour la partie librairie ;
- `xUnit` pour la partie testing.

Nous aurions pu imaginer d'autres projets tels que `Xamarin` pour compléter la solution. Pour lancer les tests, il est également très simple de le faire avec la commande suivante :

```
dotnet test
```

Les futurs travaux

Malgré la sortie de la version, il reste encore beaucoup de travail avec `.Net Standard`. Tout d'abord, le standard n'est pas implémenté sur les applications `UWP` de Microsoft. Une version preview est actuellement diffusée depuis le 25 Août permettant de travailler avec les 2 technologies. Le prérequis est que le projet `UWP` doit cibler la version `Fall Creator Update` pour fonctionner. Nul besoin de mettre à jour votre PC, il suffit d'installer le `SDK` correspondant. [5]. Selon le repo Github de `CoreFX` (<https://github.com/dotnet/corefx>), `UWP` implémentera officiellement `.Net Standard 2.0` le 17 Octobre, date de sortie de la mise à jour `Fall Creator Update` pour `Windows 10`. La version de `UWP` sera alors la version 6. •

PCL Profile	.NET Standard	PCL Platforms
Profile7	1.1	.NET Framework 4.5, Windows 8
Profile31	1.0	Windows 8.1, Windows Phone Silverlight 8.1
Profile32	1.2	Windows 8.1, Windows Phone 8.1
Profile44	1.2	.NET Framework 4.5.1, Windows 8.1
Profile49	1.0	.NET Framework 4.5, Windows Phone Silverlight 8
Profile78	1.0	.NET Framework 4.5, Windows 8, Windows Phone Silverlight 8
Profile84	1.0	Windows Phone 8.1, Windows Phone Silverlight 8.1
Profile111	1.1	.NET Framework 4.5, Windows 8, Windows Phone 8.1
Profile151	1.2	.NET Framework 4.5.1, Windows 8.1, Windows Phone 8.1
Profile157	1.0	Windows 8.1, Windows Phone 8.1, Windows Phone Silverlight 8.1
Profile259	1.0	.NET Framework 4.5, Windows 8, Windows Phone 8.1, Windows Phone Silverlight 8

Ce tableau récapitule les compatibilités qui existent entre `.Net Standard` et les `PCLs`. Par exemple, une librairie en `.Net Standard 1.2` va pouvoir cibler les profiles `PCL 32, 44 et 151`. Dans le projet, le résultat sera l'ajout de cette ligne dans le `.csproj` :

```
<PackageTargetFallback>portable-net45+win8+wpa81+wp8</PackageTargetFallback>
```

Ainsi, la librairie `.Net Standard` va pouvoir cible des `PCLs` avec le profile 259.

Utiliser `.Net Standard` dans mon projet

La création d'un projet `.Net Standard` est extrêmement simple avec les IDE modernes d'aujourd'hui. Via une interface graphique, ils ne font qu'utiliser les commandes disponibles via l'outil en ligne de commande `dotnet`. L'objectif de cette partie est de créer une solution de zéro mettant en jeu un projet `.NET Core`, un projet `.NET Standard` et un projet de test `xUnit` afin de créer des tests unitaires. Commençons par les premières commandes nous permettant de créer un projet `.NET Core` vide.

```
mkdir MonProjet
cd ./MonProjet

dotnet new sln

mkdir src/MonProjet.Core -p
cd ./src/MonProjet.Core
dotnet new web
```

Arduino : un coffre lumineux pour l'été !

• Renaud Pinon

Le retour des beaux jours implique beaucoup de choses : la nature se réveille, le soleil se couche plus tard et le plus important... les apéros reviennent ! Mais comment être sûr d'impressionner vos amis lorsque, le crépuscule venu, vous leur demanderez de quel breuvage ils veulent se délecter ? En fabriquant un coffre à bouteilles lumineux bien-sûr ! [1]

Je vous propose aujourd'hui un petit montage simple qui, même si son utilité peut toujours être discutée (autour d'un bon verre évidemment !), tirera partie à merveille des *Neopixels*, ces petits ensembles de *LEDs RGB* développés par *Adafruit* qui viennent sous diverses formes : en anneaux, en lignes, en tableau et même à l'unité à coudre dans un vêtement. Chaque *LED* d'un *Neopixel* peut-être pilotée pour avoir sa propre couleur de façon très simple grâce à la bibliothèque *Adafruit* pour *Arduino IDE*. Il est même possible de chaîner plusieurs *Neopixels* !

J'utiliserai pour ma part un seul anneau de 12 *LEDs RGB* faisant 5 cm de diamètre. Je me suis fourni chez mon revendeur chinois habituel, dont le produit porte le nom *WS2812* (ce qui est également le nom du contrôleur utilisé par *Adafruit*). Nous piloterons tout cela bien évidemment avec un *Arduino* : un *Pro Mini* pour l'occasion. Ceci étant dit, en avant pour une petite séance bricolage !

Un peu de logistique

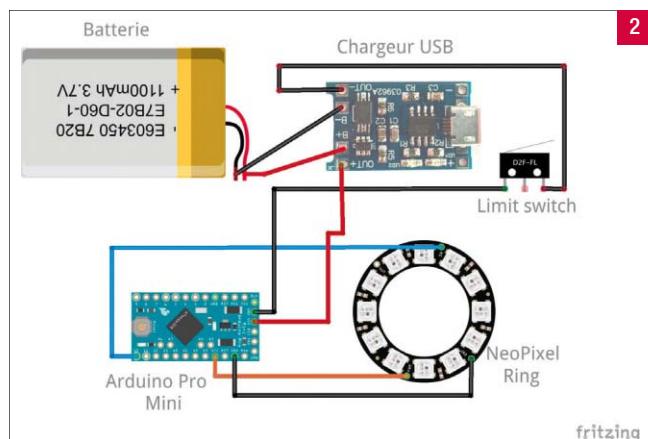
Ingrediénts :

- 1 coffre ;
- Des fils ;
- 6 vis 3mm * 30 mm (pour fixer le boîtier au coffre) + écrous ;
- 4 vis (à pointe) 2mm * 20mm (pour fixer le *NeoPixel* au boîtier) ;
- 1 *NeoPixel Ring* (12 leds, 5 cm de diamètre) ;
- 1 *Arduino Pro Mini* ;
- 1 batterie Li-ion Polymère 3.7V d'une capacité correcte (dans mon cas 1 000 mAh) ;
- 1 chargeur de batterie avec prise *USB* et convertissant 3.7V en 5V.

Recette : [2]

Principe de fonctionnement

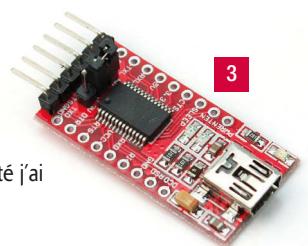
A l'ouverture du coffre, nous allumons toutes les *LEDs* de la même couleur suivant une couleur cible choisie aléatoirement. Nous créons pour cela un effet de pulsation lumineuse en faisant passer progressivement les *LEDs* de éteintes à allumées jusqu'à la couleur cible choisie (phase ascendante), puis progressivement vers éteintes (phase descendante), avant de choisir une nouvelle couleur cible et de répéter la même procédure jusqu'à la fermeture du coffre. Un interrupteur *limit switch* (similaire aux imprimantes 3D) permet d'alimenter le circuit à l'ouverture du coffre et de le couper lorsqu'on referme le couvercle. A noter que nous allons souder les fils de cet interrupteur pour qu'il fonctionne en mode « normalement fermé » (« normally close » ou « NC »), ce qui permettra d'alimenter le circuit – et donc de décharger la batterie – uniquement lorsque le coffre est ouvert. Dans mon cas, sur les 3 broches du *switch*, il s'agit de celles à chaque extrémité (celle du milieu reste donc libre). Enfin, le chargeur de batterie s'occupe non seulement de nous fournir une prise *USB* pour recharger notre batterie, mais aussi de booster le voltage de cette dernière de 3.7V vers 5V afin d'alimenter notre *Arduino Pro Mini*. Première chose à remarquer : l'*Arduino Pro Mini* ne possède pas d'entrée



USB, il faudra donc lui adjoindre un module *USB* vers *FTDI* pour téléverser notre programme. Cet achat est à mon sens un investissement car sincèrement, je trouve la forme et la taille du *Pro Mini* bien plus pratiques que celles du *Nano* par exemple. De plus, pour les projets nomades avec batterie, le port *USB* d'un *Arduino* devient inutile. Par ailleurs, si vous voulez un jour programmer un *ESP8266* ou *ESP32*, ce module *FTDI* sera indispensable ! [3]

Deuxièmement, nous laissons une des 4 broches du *Neopixel Ring* déconnectée, à savoir celle permettant de chaîner un autre *Neopixel*. Pour la version *Adafruit*, c'est la broche « *Data out* ». Pour ma version, c'est « *DO* » (pour *Data Out* aussi, ce n'est pas un zéro). Nous allons donc connecter les 3 autres broches qui sont *5V* (ou *PWR*) et *GND* reliées respectivement aux broches *5V* et *GND* de l'*Arduino* (sur le côté, pas en bas), tandis que la broche *Data IN* (ou *DI*) sera reliée à la broche *D10* de l'*Arduino*.

Bien-sûr, quand je dis « broches », je m'entends : vous ne pourrez pas juste les brancher, il faudra évidemment jouer du fer à souder pour raccorder tout cela. De mon côté j'ai



prévu des fils pas trop longs (à part pour le *limit switch*) pour pouvoir empaqueter tout cela dans un boîtier imprimé en 3D dont vous aurez le lien de téléchargement à la fin de cet article. Si vous n'avez pas d'imprimante 3D, pas de panique : il est possible de réaliser ce genre de boîtier avec de la récup' (boîte de chocolat transparente, carton ou agglo, ... souvenez-vous de vos séances bricolage en maternelle !), l'essentiel étant que le montage soit maintenu à la face avant de votre coffre (à l'intérieur), qu'il soit protégé des chocs tout en vous laissant la place pour connecter le câble *USB* servant à recharger la batterie, ainsi que laisser sortir les fils menant au *limit switch*.

Le code

(Code complet sur www.programmez.com)

```
#include <Adafruit_NeoPixel.h>
// Numéro de broche du neopixel :
#define kPinNeopixel 10
// Nombre de leds sur le NeoPixel :
#define kNumPixels 12

#define kSteps 0.01
#define kDelay 20

// Variables globales:
// Phase ascendante ou descendante :
bool _makeBrighter = true;

// Objet NeoPixel :
Adafruit_NeoPixel _pixels = Adafruit_NeoPixel(kNumPixels, kPinNeopixel, NEO_GRB + NEO_KHZ800);

// pourcentage de la valeur cible (0.0 à 1.0):
float _factor = 0.0;

// Couleurs RVB cibles:
int _rTarget = 0;
int _gTarget = 0;
int _bTarget = 0;

// Prototypes:
void SetTargetColor();
void PixelsLoop();

// Fonctions:
// Initialisation
void setup()
{
    _pixels.begin(); // Initialise les NeoPixels.

    // initialise le générateur de nombre aléatoire :
    randomSeed(analogRead(0));
    // Définit la première couleur RGB:
    SetTargetColor();
}
```

Dans la fonction *loop()*, j'ai choisi une approche qui « laisse la main » à l'*Arduino*. J'aurais pu en effet allumer et éteindre progressivement les *LEDs* dans une seule boucle *while* bien sentie, mais le problème est que notre *Arduino* ne peut rien faire d'autre pendant ce temps-là (environ 4 secondes quand même !). Or, même si ce n'est pas notre cas, que se passe-t-il si l'on doit vérifier régulièrement l'état *LOW* ou *HIGH* d'une broche ? Ou si l'on doit rafraîchir l'affichage d'un écran *LCD* ? N'oublions pas que l'*Arduino* est mono-fil d'exécution !

Dans la fonction *Setup()* nous initialisons le générateur de nombres aléatoires. Un développeur sur ordinateur a tendance à trouver cette étape particulièrement basique alors que pour les microcontrôleurs ce n'est pas si anodin. En effet, cette initialisation, quelle que soit la machine, se fait à partir du nombre de millisecondes depuis lequel elle est démarrée : sur un ordinateur, il est rare que l'utilisateur démarre un programme à la même millisecondes près ! En revanche sur un microcontrôleur, le programme démarre toujours au même moment, ce qui induit une suite de nombres aléatoires toujours strictement identique. Nous allons donc induire un peu d'aléatoire « électronique » grâce à la broche analogique *A0* : si cette dernière n'est raccordée nulle part (ni à un module ni même à *GND*), alors les courants parasites feront varier la valeur que l'on peut y lire (0 à 1023) de façon totalement aléatoire : idéal pour initialiser notre générateur de nombre ! La fonction *SetTargetColor()* quant à elle permet de trouver la prochaine couleur cible aléatoire pour chacune des composantes *RGB*, que l'on va stocker dans 3 variables globales.

Enfin, la fonction *PixelLoop()* définit le pourcentage d'illumination des *LEDs* par rapport à la couleur cible (de 0% à 100%, soit 0.0 à 1.0) et avec un incrément égal à *kSteps* (ici 1 %, soit 0.01). Dans la phase ascendante on additionne *kSteps*, tandis qu'en phase descendante on le soustrait. Nous allumons ensuite toutes les *LEDs* de l'anneau à la couleur graduelle trouvée. Enfin, si nous sommes dans la phase descendante et que notre pourcentage appliqué à la couleur cible est inférieur à l'incrément *kSteps* (soit ici un pourcentage intérieur à 1%), nous définissons une nouvelle couleur cible et repassons en phase ascendante.

Le résultat est là : une belle pulsation où la couleur des *LEDs* change à chaque début de phase ascendante. Vous pouvez vous amuser à modifier les constantes *kSteps* et *kDelay* : vous verrez, vous obtiendrez des effets stroboscopiques assez intéressants !) [4]

Et maintenant on assemble !

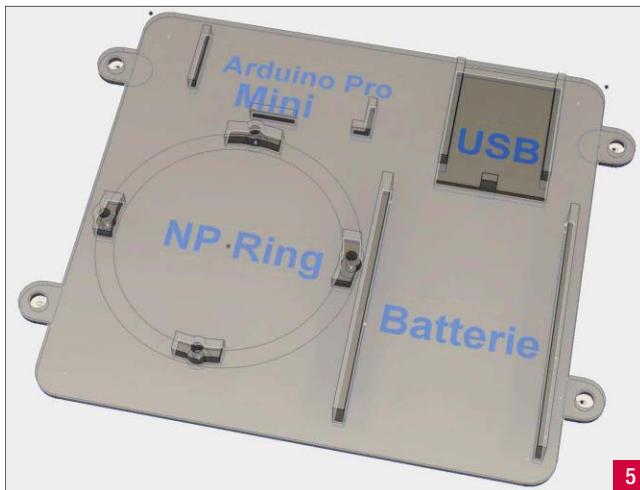
Pour ceux qui possèdent une imprimante 3D, les modèles STL du boîtier peuvent être téléchargés à l'adresse : https://github.com/renaudpinon/Coffre_LED. Le fichier *Fusion 360* (.f3d) est également présent si vous souhaitez apporter des modifications : modifiez juste les dessins pour changer la forme et tout devrait se mettre en place sans que vous n'ayez à rajouter d'objet ou modifier de faces. Regardez le fichier *Readme* pour plus d'informations sur la modification et l'impression des modèles. [5]

Notez que je conseille *Fusion 360* à tous ceux qui font de la conception assistée par ordinateur : les possibilités sont absolument immenses et les systèmes d'historiques, contraintes et paramètres utilisateur permettent une utilisation d'une puissance que je n'ai encore jamais vue dans un autre logiciel. Et en plus il est gratuit pour les particuliers, il suffit de créer un compte sur le site d'*AutoCAD* ! La soudure des composants est plutôt simple donc je ne m'attarde pas dessus : suivez simplement le schéma *Fritzing* en haut de l'article. Vérifiez que vous connectez la batterie correctement aux bornes *bat+* et *bat-* du chargeur *USB*. Durant cette opération vous allez sans doute couper le connecteur *JST* de la batterie pour souder les extrémités des fils aux bornes du chargeur : faites attention à ce que les fils de la batterie ne se touchent pas ! Si cela arrive vous devriez vous en rendre compte car il y aura des étincelles : si ce n'est qu'un court instant ce n'est pas dramatique, mais c'est tout de même dangereux pour la batterie (et potentiellement : 3.7V dans votre corps c'est supportable, mais pas agréable !). Reliez ensuite les bornes *VCC*

(5V) et GND de l'Arduino Pro Mini respectivement aux bornes Out+ et Out- du chargeur (via le *limit switch*, que vous maintiendrez judicieusement coupé en l'appuyant avec du scotch). Une fois que tout est soudé (Arduino, NeoPixel Ring, chargeur, batterie et *limit switch*) et que vous avez vérifié que tout fonctionne, je vous conseille très fortement de tout isoler en utilisant un pistolet à colle chaude : barbouillez toutes les connexions de colle pour les renforcer et être sûr qu'aucun fil ne se détache et vienne court-circuiter votre montage ! C'est particulièrement vrai pour les 4 connexions du chargeur de batterie ! Notez également que l'Arduino Pro Mini n'a pas de trous pour être vissé, de même que le chargeur USB : je les colle donc abondamment au boîtier grâce à la colle chaude. En revanche ne faites pas de même pour la batterie : si vous avez besoin de fixer la vôtre, utilisez plutôt du scotch d'électricien ou du scotch double face. Pensez enfin que votre *limit switch* devra être en contact avec une partie du couvercle pour être en position appuyée (OFF pour nous) : vous aurez donc à fixer une « cale » au couvercle qui viendra précisément appuyer sur le *switch* lorsque le coffre sera fermé.

Point sécurité [6]

Soyez très prudent sur la soudure des fils du chargeur de batterie : aucun fil ne doit se toucher sous peine d'éclatements/court-circuit/chauffe. Ne mettez jamais de bouteilles ouvertes/mal fermées dans le coffre et vérifiez également que chaque bouteille est propre (pas de liquide alcoolisé qui aurait coulé le long de la bouteille et qui irait s'évaporer dans le coffre ensuite). N'ouvrez pas non plus de bouteille au-dessus du boîtier pour éviter que du liquide n'entre en contact avec le circuit ! Si vous suivez ces directives simples, rien de fâcheux ne doit arriver (d'autant plus que le circuit n'est alimenté qu'une fois le couvercle du coffre ouvert). N'oubliez pas que l'électronique c'est super, mais c'est encore mieux quand on ne devient pas une statistique des accidents domestiques ;)



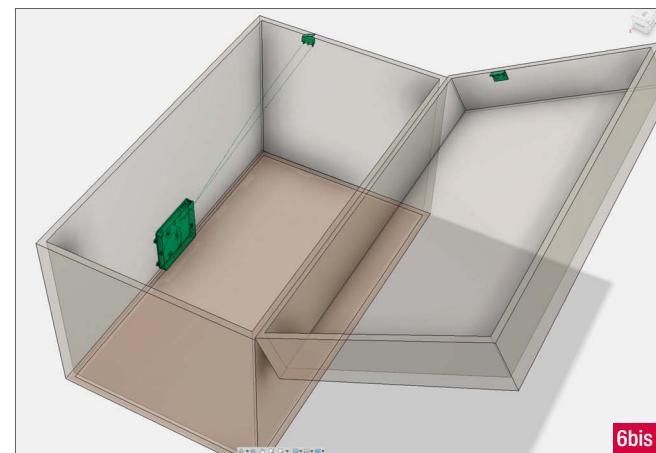
Bilan du projet

Vous avez donc maintenant de quoi faire un coffre lumineux. Les bouteilles à l'intérieur modifieront agréablement la teinte de la lumière pour donner de nombreuses réflexions colorées qui émerveilleront vos invités ! Mais au fait, quel a été le coût de ce projet ? Vous trouverez sur mon GitHub un tableau au format PDF indiquant les références des pièces, leur coût et où les acheter. Vous constaterez un tarif global de 14,17 euros. Bien-sûr, je n'ai pas compté le prix du coffre, ni les vis, le module FTDI, la colle chaude ou le prix des impressions 3D (1,09 euros me dit Simplify3D pour la base + le couvercle + la cale... Mais ça dépend du prix auquel vous achetez votre filament).

J'espère dans tous les cas que ce projet simple vous aura plu et que les différentes notions abordées vous auront appris des choses utiles (je pense notamment aux chargeurs de batterie USB qui peuvent être utilisés dans tous vos projets nomades, mais aussi aux NeoPixel Rings qui égayeront vos créations).

Et pour finir, comment aller plus loin et améliorer ce projet ? Voici quelques pistes que je vous laisse le loisir d'explorer :

- Remplacer la pulsation lumineuse par un autre schéma lumineux (par exemple allumer successivement et avec un délai chaque LED de l'anneau avec une couleur aléatoire) et pourquoi pas intégrer des boutons pour choisir le type de schéma lumineux.
- Chaîner plusieurs NeoPixel Rings pour, par exemple, en mettre un à droite et l'autre à gauche du coffre (pour une meilleure illumination !)
- Connecter un buzzer à l'Arduino et lui faire jouer un air le temps de l'ouverture du coffre (vous ne pourrez plus boire en cachette :))



Simplifiez-vous la vie avec **Unity**

• Loïc Kempf
Magma Mobile

La mission que s'est donnée Unity auprès des développeurs s'articule autour de 3 valeurs : démocratiser le développement, résoudre des problèmes compliqués et favoriser la réussite.

Pour ce faire, Unity développe depuis la version 5.2 toute une série de services et d'outils ayant pour objectif de simplifier la vie des développeurs. Cela va des solutions de suivi et d'analyse de trafic aux systèmes de gestion d'achats intégrés. Ces derniers remplaçant parfois en quelques lignes de code ce qui pouvait représenter des temps de développement conséquents. Tous ces services ont l'avantage d'être intégrés directement dans le moteur, ce qui évite d'avoir à embarquer des librairies externes dans son jeu, qui peuvent être sources d'erreurs en plus d'alourdir la taille des builds.

L'offre Unity a beaucoup évolué au fil des ans et des nouveautés arrivent régulièrement avec chaque mise à jour. C'est pourquoi nous vous proposons un tour d'horizon des ressources et services qui pourront vous faire gagner du temps ou vous simplifier la vie dans vos projets avec Unity afin de vous concentrer sur l'essentiel : faire un jeu de qualité.

Faites le plein de ressources

Une des forces d'Unity est sa communauté de développeurs très active, que ce soit sur les forums pour répondre aux éventuels problèmes techniques, ou pour la création d'outils. Faites un tour à la conférence Unity Unite, et vous vous rendrez compte à quel point ses membres, d'horizons divers, sont enthousiastes.

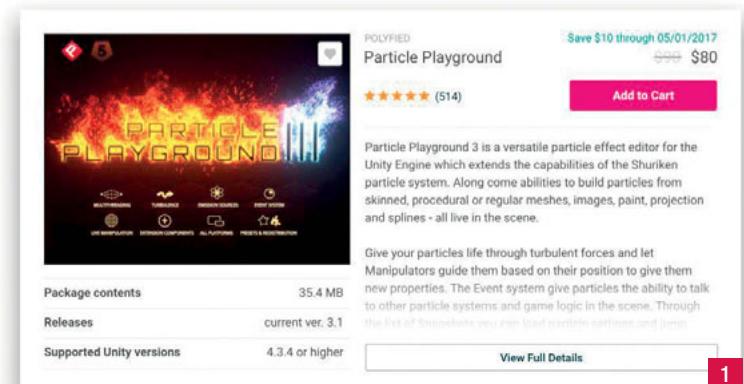
Cela se traduit par un marché de ressources, l'Unity Asset Store, qui comporte à l'heure actuelle plus de 40 000 packages qui peuvent être importés dans un projet Unity en quelques clics. [1]

Les packages que l'on trouve sont extrêmement variés. Ils peuvent être soit des éléments graphiques comme des personnages ou des décors en 3D, des ensembles de scripts pour implémenter des fonctionnalités dans votre projet sans avoir à les coder, ou alors carrément des jeux entiers ; ceux-ci peuvent servir de base pour votre propre projet ou peuvent être utilisés à titre éducatif.

A noter que bon nombre des assets présents sur l'Unity Asset Store, dont certains sont très puissants comme le WRLD Unity SDK qui permet de créer des villes en 3D, sont gratuits et il y a très régulièrement des promotions. [2]

L'Unity Asset Store est une ressource très intéressante qui pourra faire gagner beaucoup de temps et d'argent pour peu que l'on fasse l'effort de faire des recherches et de faire le tri entre les packages intéressants ou incompatibles avec les dernières versions d'Unity. Cela se révèle assez laborieux avec la version actuelle, chose qui sera corrigée avec le déploiement imminent d'une refonte intégrale de la place de marché, qui va mettre l'accent sur la découverte d'assets et les recommandations personnalisées.

Si vous ne trouvez pas votre bonheur sur l'Unity Asset Store et que vous n'avez pas les ressources nécessaires à disposition, sachez que Unity a aussi pensé à vous avec sa plateforme Unity Connect. Comme son nom l'indique, elle permet de prendre contact avec d'autres membres de la communauté Unity - que ce soient des développeurs ou des artistes - et de les embaucher pour qu'ils vous aident dans la réalisation de votre projet. A l'heure actuelle il y a plus de 50 000 personnes inscrites et environ



WRLD Unity SDK sur l'Asset Store :
<https://www.assetstore.unity3d.com/en/#!/content/86284>

400 travaux en attente de développeurs. A noter qu'il y a également des offres d'emplois sur Unity Connect.

Travaillez ensemble

Les projets Unity peuvent faire intervenir de nombreuses personnes d'horizons divers comme des programmeurs évidemment, mais aussi des artistes qui, comme nous avons pu le voir lors de la conférence Unity Unite, avec notamment des outils comme Timeline ou Cinemachine, ont vocation à avoir de plus en plus la main sur le moteur.

Il convient donc de réussir à faire travailler tous ces profils variés ensemble. On pourra pour cela utiliser des solutions classiques de contrôle de sources telles que Perforce ou Plastic SCM qui sont directement intégrées dans Unity, ou encore Git, qui dispose d'un plugin officiel pour le moteur. Ces solutions présentent toutefois des limites résultant de la nature d'Unity, qui rendent complexes certaines tâches comme le travail simultané sur une même scène.

Unity propose en réponse sa solution maison appelée Unity Collaborate, qui vient tout juste de sortir de beta. Elle permet de partager très facilement un projet à des collaborateurs et donne l'avantage de pouvoir travailler à plusieurs sur la même scène.

Unity Collaborate fait partie du package Unity Teams qui propose par

ailleurs Unity Cloud Build qui offre la possibilité de faire des builds de jeu et les déployer facilement sur des appareils à travers le Cloud.

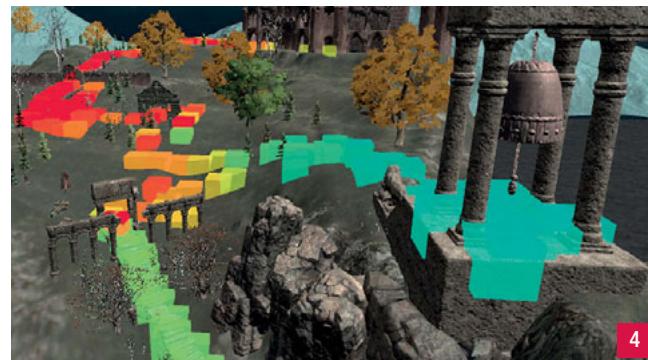
Prenez les bonnes décisions

Unity embarque en son sein une solution de suivi et d'analyse des joueurs appelée tout simplement Analytics pour que vous puissiez suivre les actions des joueurs dans vos jeux, mesurer leur engagement et détecter les éventuels problèmes. [3]

Il suffira alors d'appeler les différents événements correspondant aux actions des joueurs pour les voir apparaître en direct dans une console en ligne dédiée.

On pourra ensuite mettre en place des entonnoirs de conversion selon l'enchaînement des actions qu'ils effectuent dans le jeu, mais aussi classer les joueurs en segments en fonction de certains attributs. A noter que plusieurs segments comme la rétention des joueurs ou leur provenance géographique sont présents par défaut, ce qui vous donnera un rapide aperçu de l'activité de votre jeu. L'intégration native de Unity Analytics permet des applications puissantes comme par exemple le package Heatmaps qui affiche le cheminement des joueurs et les éventuels points de blocage directement dans l'éditeur. [4]

Unity Remote Settings est quant à elle une toute nouvelle fonctionnalité avec laquelle on peut définir des variables dans la console Unity Analytics qui seront mise à jour directement chez le joueur la prochaine fois qu'il lancera le jeu. Cela permet par exemple d'ajuster dynamiquement le niveau de difficulté ou alors de déclencher des événements saisonniers comme un weekend où l'expérience gagnée dans le jeu sera doublée. Evidemment, on pourrait faire la même chose en allant chercher la valeur sur un serveur mais on appréciera tout de même le gain de temps.



4

Unity Analytics Heatmaps : <https://www.assetstore.unity3d.com/en/#!/content/65801>

using UnityEngine;

```
public class LiveEvents : MonoBehaviour {
```

```
    public bool DefaultDoubleXP = false;
```

```
    public static bool DoubleXP{ get; private set; }
```

```
    void Start () {
```

```
        DoubleXP = DefaultDoubleXP;
```

```
        RemoteSettings.Updated +=
```

```
            new RemoteSettings.UpdatedEventHandler(HandleRemoteUpdate);
```

```
    }
```

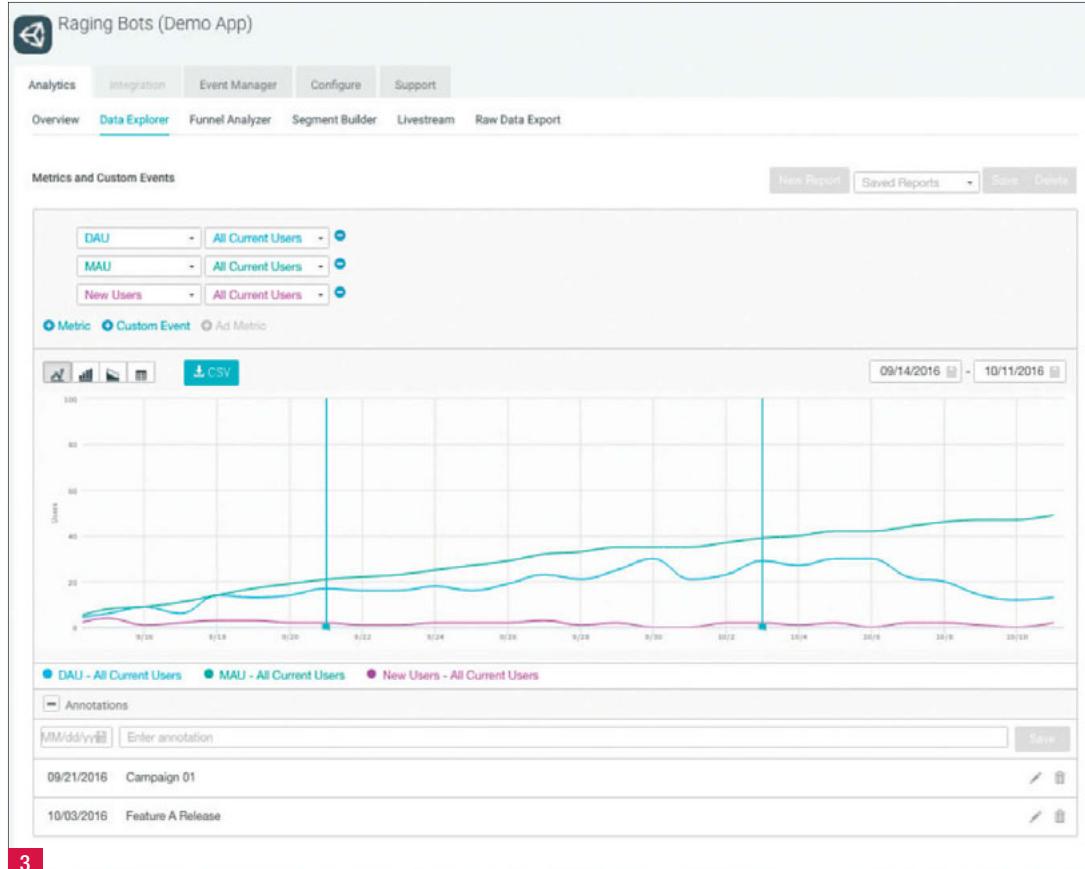
```
    private void HandleRemoteUpdate(){
```

```
        DoubleXP = RemoteSettings.GetBool ("DoubleXP", DefaultDoubleXP);
```

```
    }
```

```
}
```

Récupérer une valeur sur Unity Analytics Remote Settings, c'est aussi simple que cela.



3

Toujours dans la même veine, Unity Performance Reporting affiche en temps réels les erreurs et plantages dans les applications. L'avantage par rapport à d'autres solutions est que l'on va voir la stack trace complète, permettant ainsi un débogage plus efficace. [5]

Boostez vos jeux

Le multijoueur est une fonctionnalité puissante qui, bien utilisée, peut démultiplier le potentiel d'un jeu. Ceux qui s'y sont frottés savent que développer une couche multijoueur robuste pour un jeu est loin d'être chose aisée. Certaines entreprises se sont donc lancées sur ce créneau. C'est le cas par exemple de Photon qui propose une solution complète avec matchmaking, salons de chat et discussions vocales.

Unity offre également sa solution baptisée Unity Multiplayer, qui se cantonne à l'essentiel mais qui a l'avantage d'être intégrée directement dans le moteur ; elle inclut surtout un certain nombre d'utilisateurs concurrents en fonction de l'abonnement Unity que vous possédez, que vous pourrez échelonner en fonction de vos besoins.

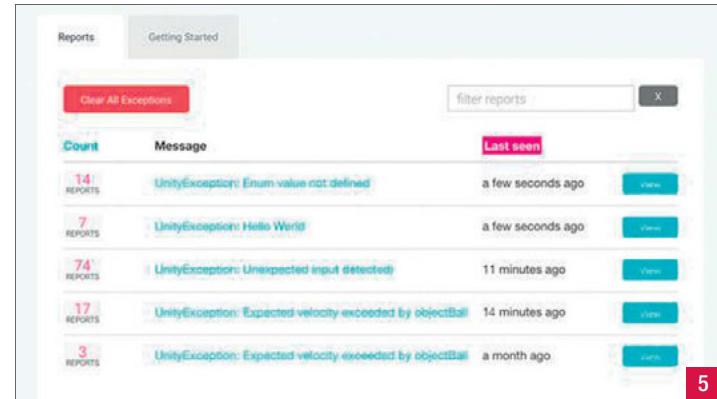
Gagnez de l'argent

Unity est bien conscient que le marché du jeu vidéo, notamment sur mobile, est très concurrentiel et comme ailleurs le nerf de la guerre est l'argent. C'est pourquoi Unity propose depuis maintenant quelques années sa solution de publicités vidéo Unity Ads. Très simple d'intégration, Unity Ads permet aux développeurs de commencer à gagner de l'argent en quelques clics. Le principe d'Unity Ads est d'offrir des récompenses, par exemple le fait de pouvoir continuer une partie lors d'un Game Over ou obtenir des objets dans le jeu, en échange du visionnage d'une vidéo, s'éloignant ainsi des publicités classiques souvent jugées gênantes par les joueurs. Le tout se fait encore une fois en quelques lignes de code.

Si le business model de votre jeu s'oriente plutôt autour des achats In App, Unity s'occupera aussi de tout puisqu'il est possible d'intégrer vos achats d'objets virtuels facilement pour Android et iOS. Mieux encore, Unity propose le support natif des marchés d'applications alternatifs comme l'Amazon Appstore ou Samsung Galaxy Apps.

Vers l'infini et au-delà

Tout ceci n'est évidemment qu'une partie de ce qu'offre le moteur en matière d'outils et de services. Cette volonté de démocratisation se tra-



duit aussi par la mise en place de programmes d'éducation. Il y a également le partenariat qu'Unity a fait avec l'entreprise Xiaomi pour permettre aux développeurs de publier leurs jeux sur les marchés chinois. On l'aura compris, Unity met tout en œuvre pour simplifier la vie de ses développeurs car un développeur heureux qui réussit est un développeur qui va continuer à utiliser Unity, et ce n'est pas prêt de s'arrêter si l'on en croit la roadmap pour les mois à venir.

Liens utiles :

- <https://blogs.unity3d.com/2017/06/27/unite-europe-2017-keynote-recap-connecting-creating-and-the-future-of-unity/> - Un récapitulatif de toutes les annonces de la keynote Unity Europe 2017
- <https://blogs.unity3d.com/> - Le blog officiel d'Unity avec les annonces relatives au moteurs
- <https://unity3d.com/fr/services> - Le portail des services Unity
- <https://connect.unity.com/> - La plateforme de mise en relation Unity Connect
- <https://unity3d.com/fr/unity/roadmap> - La feuille de route d'Unity
- <http://blog.theknightsofunity.com/> - Un blog avec plein de tutoriaux et bonnes pratiques pour améliorer vos projets Unity
- <https://unity3d.com/community/evangelists> - La liste des évangélistes Unity, qui partagent régulièrement des ressources sur leurs comptes Twitter

Dans le prochain numéro ! Programmez! #212, dès le 3 novembre 2017

Infrastructure as Code

Codons notre infrastructure !

Choisir sa base de données

Quelle base de données choisir ? Quels critères techniques retenir ? Version locale ou version cloud ? Les réponses dans notre dossier spécial.

Linux au cœur de Windows 10

Le sous-système Linux de Windows 10 offre des usages inédits pour les développeurs et les sysadmins. Présentation complète.

Connecter son imprimante 3D en WiFi avec un Raspberry Pi et Octoprint



Mike FAHRASMANE
Titulaire d'un master en génie électrique et acteur du mouvement maker et DIY (Do-It-Yourself) passionné d'innovation. Il est le fondateur de MAKERSGENERATION (www.makersgeneration.net) société proposant des afterschools et camps autour des technologies émergentes pour les enfants, adolescents et adultes ainsi que la recherche et le développement en électronique.

www.makersgeneration.net |
contact@makersgeneration.net

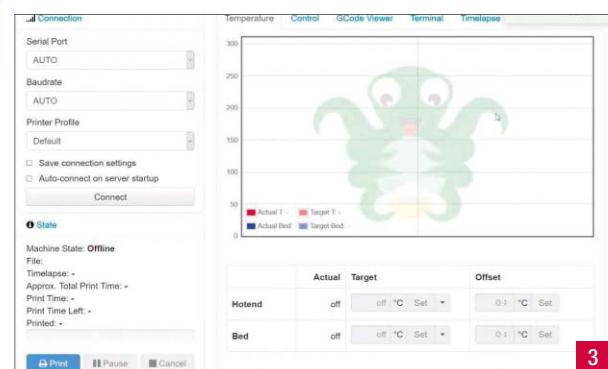
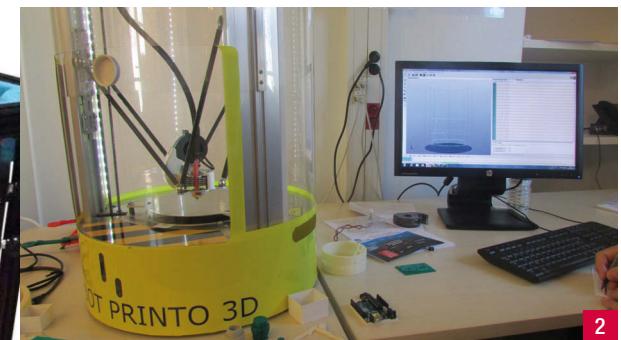
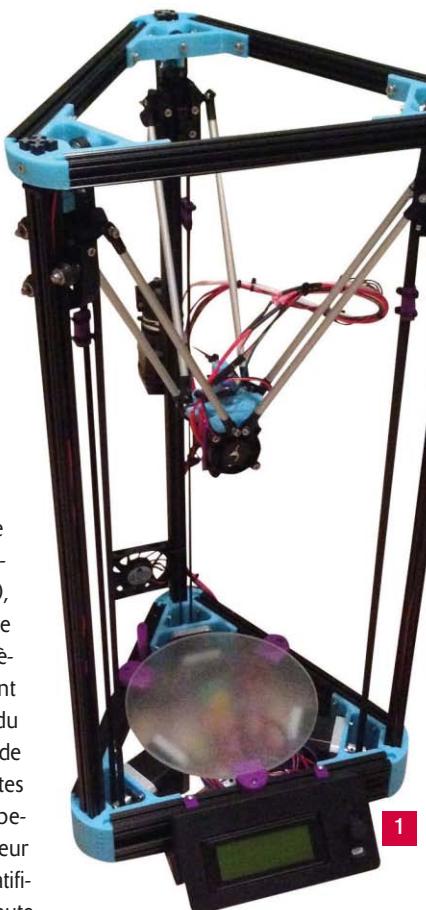
De nos jours, les imprimantes 3D prolifèrent. On en trouve partout et à tous les prix, à partir de 200 € et pouvant parfois atteindre des millions d'euros. Ce qui bien sûr est hors de notre portée en termes de prix et pour le genre de travail que nous voulons effectuer.

Les différents types de technologies d'imprimante 3D

On retrouve des imprimantes de type SLA (imprimantes avec de la résine photopolymère comme matière première), des imprimantes SLS ou frittage de poudre (avec de la poudre photopolymère) et pour finir les plus connues sont celles fonctionnant avec du fil fondu (FDM) fonctionnant avec des bobines de PLA ou d'ABS Fig. 1. Ces imprimantes pour fonctionner ont le plus souvent besoin d'être connectées à un ordinateur d'où l'on générera grâce à un logiciel stratificateur (Cura, Pronterface, Repetier...) toute la configuration qui sera ensuite envoyée à la machine (Fig. 2). On peut aussi charger le programme sur une carte SD et ensuite insérer celle-ci dans l'imprimante et ainsi lui permettre de fonctionner sans besoin de la connecter à l'ordinateur.

Connexion sans fil grâce à Octoprint

Une autre façon de contrôler son imprimante est via le réseau local ou Internet. Dans notre cas, nous allons connecter notre imprimante à notre réseau local grâce au WiFi. Vous pouvez donc, sans avoir besoin de câble USB connecté



à votre ordinateur, accéder à votre imprimante 3D. Tout cela est rendu possible grâce à Octoprint. Octoprint est une application web permettant de contrôler une imprimante 3D à distance, voire le plateau et le contrôle (Fig 3). Nous essaierons ici de contrôler à distance une imprimante 3D avec comme petit plus que celle-ci soit connectée au WiFi. En plus clair, que notre imprimante ne soit connectée à aucun ordinateur à la maison, mais seulement au réseau WiFi.

Matériels nécessaires

Pour vous lancer dans cette aventure, il vous faudra acquérir les éléments suivants :

- Une imprimante 3D Reprap : pour le côté open source ;
- Une Raspberry Pi 3 (module WiFi intégré à la carte) ;
- Un câble d'alimentation pour le Raspberry Pi ;
- Une carte micro SD de 8Go minimum (16 ou 32 Go pour le confort) ;
- Un câble USB type B.

Logiciels nécessaires :

- La dernière image d'Octoprint « Octopi » pour Raspberry Pi téléchargeable sur le site : octoprint.org/download/ ;
- Win32 Disk Imager pour le formatage de la carte SD (si vous êtes sur Windows) ;

- Putty : le logiciel pour la gestion à distance de votre ordinateur via l'internet utilisant le « protocole SSH ».

Tout cela en main, vous pouvez télécharger l'image d'Octoprint « Octopi » pour Raspberry Pi en relation avec votre système d'exploitation (Windows, Linux ou macOS), Windows dans notre cas sur « octoprint.org » (Fig. 4)

Installation de Raspbian OS et Octopi

Après avoir téléchargé le dossier, décompressez le fichier image sur votre ordinateur pour trouver un fichier avec l'extension suivante "img".

Télécharger Win32 Disk Imager que vous

pourrez retrouver sur sourceforge.net et installez-le sur votre ordinateur. Win 32 Disk Imager permettant d'installer le fichier image sur notre carte SD pour le Pi.

Ouvrez l'interface de Win32 manager et sélectionnez le dossier où vous avez enregistré votre fichier image « img » sur votre machine et ainsi que la carte SD et cliquez sur write pour copier l'image sur la carte SD (Fig 5).

Ceci fait, nous devons configurer le WiFi sur notre Raspberry Pi. Pour ce faire, rendez-vous sur votre carte SD et ouvrez le fichier « octopi-network » avec un éditeur de texte tel Notepad++ (Fig. 6). Ceci fait, changez le « WPA-SSID » qui est le nom de votre réseau WiFi et le « WPA-PSK » qui est le mot de passe de votre

WiFi. Rechercher dans le fichier « Put SSID here » et « Put password here ». Une fois trouvé, remplacez-les par vos identifiants et mot de passe.

Retirer le # placé à la ligne juste avant pour valider celle-ci. La description de la ligne est la suivante « iface wlan0 inet manual ».

Après avoir terminé cette étape, retirez la carte SD de votre ordinateur et insérez-

la dans votre Raspberry.

Profitez-en pour mettre votre Raspberry Pi en marche en le branchant à une alimentation grâce au câble USB mini.

Vous pouvez tester si tout fonctionne correctement avec votre Raspberry Pi et que nous sommes connectés au réseau local grâce au terminal Windows en tapant la commande suivante : « ping octopi.local »

Vous devriez recevoir un message de succès avec l'adresse IP de votre Pi ainsi que zéro paquet perdu « 0% loss ».

Adresse IP du Raspberry Pi

Pour trouver l'adresse IP de votre routeur sur votre réseau local, vous devez vous connecter à l'interface utilisateur de votre box (Free, Orange, Numéricâble...). En général, votre routeur affecte automatiquement à votre Pi une adresse IP grâce au protocole DHCP. Sachant que les interfaces utilisateurs varient plus ou moins en fonction du fournisseur internet.

Rendez-vous dans DHCP grâce à l'interface utilisateur de votre box et vous devriez trouver tous les périphériques connectés à votre box. Un exemple avec le box Free (Fig. 7).

Il nous faut maintenant nous connecter à notre carte en mode sécurisé avec le protocole SSH. Pour cela, téléchargez le logiciel Putty sur putty.org/ et installer-le sur votre ordinateur.

The screenshot shows the OctoPi download page. It features a large green button labeled "Download OctoPi 0.14.0". Below the button, it says "MD5Sum of the .zip: 52c855c9bcc88fd3bf81bbe8e58a2e4c". It also mentions compatibility with Raspberry Pi A, B, A+, B+, B2, 3, Zero and Zero W, and a note about buying one of the available versions.

The screenshot shows the Win32 Disk Imager interface. It displays the "Image File" as "V/Documentation/logiciels telecharges/octopi-jessie-lite-0.14.0.img" and the "Device" as "E:1". The progress bar is shown, and the "Write" button is highlighted. A red box labeled "5" is placed over the "Write" button.

```

5
6  ## WIFI CONFIGURATION #####
7  # The three segments below should cover you in most cases if you run
8  # a wifi network that uses either WPA/WPA2 or WEP encryption.
9  #
10 # Just uncomment the lines prefixed with a single # of the configuration
11 # that matches your wifi setup and fill in SSID and passphrase.
12 #
13 ## WPA/WPA2 secured
14 #iface wlan0 inet manual
15 #    wpa-ssid "put SSID here"
16 #    wpa-psk "put password here"
17 #
18 ## WEP secured
19 #iface wlan0 inet manual
20 #    wireless-ssid "put SSID here"
21 #    wireless-key "put password here"
22 #
23 ## Open/unsecured
24 #iface wlan0 inet manual
25 #    wireless-ssid "put SSID here"
26 #    wireless-mode managed
27 #
28 ## WIRED CONFIGURATION #####
29 # The following segment allows you to configure your wired connection
30 # with a static IP.
31 #
32 # Just uncomment the lines prefixed with a single #. Then connect
33 # a cable to the Pi and another system, e.g. a Laptop, and set that
34 # other system's network configuration to:
35 #
36 #    address: 192.168.250.10
37 #    netmask: 255.255.255.0

```

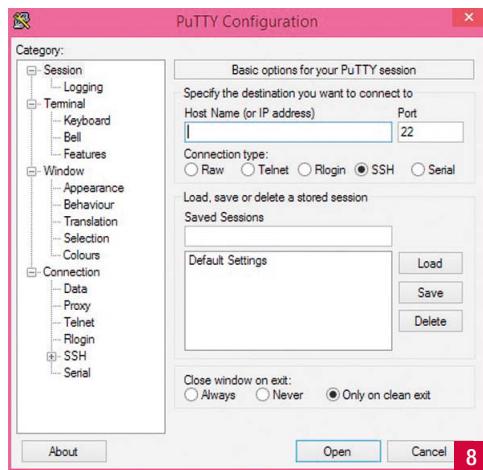
6

The screenshot shows the "Paramètres de la Freebox" interface. It has tabs for "Mode Simplifié" and "Mode avancé". Under "Connexion Internet", the "DHCP" section is highlighted with a red box labeled "7". Other sections include "Configuration", "Configuration IPv6", "DNS Dynamique", "Serveur VPN", "Client VPN", and "Gestion des ports". Other tabs like "Téléphonie" and "Réseau local" are also visible.

7

Connexion à notre réseau local et notre imprimante

Ouvrez Putty en prenant garde à bien choisir SSH pour le type de connexion. Dans la partie où il est écrit « Host name », tapez l'adresse IP de votre Pi ou « octopi.local » (Fig.8). Ensuite, cliquez sur Yes quand : "Warning potential security breach" apparaîtra à l'écran. Une fenêtre



```
pi@octopi ~ $ passwd
Changing password for pi.
(current) UNIX password:
Enter new UNIX password: 8
```

Configure Access Control

Please read the following, it is very important for your printer's health!

OctoPrint by default now ships with Access Control enabled, meaning you won't be able to do anything with the printer unless you log in first as a configured user. This is to prevent strangers - possibly with malicious intent - to gain access to your printer via the internet or another untrustworthy network and using it in such a way that it is damaged or worse (i.e. causes a fire).

It looks like you haven't configured access control yet. Please [set up an username and password](#) for the initial administrator account who will have full access to both the printer and OctoPrint's settings, then click on "Keep Access Control Enabled".

Username:

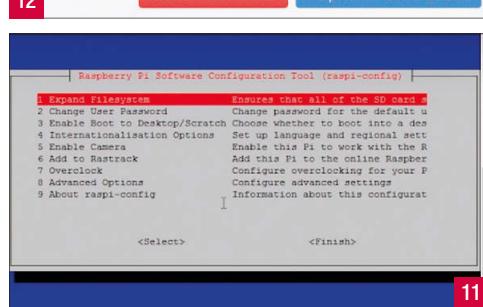
Password:

Confirm Password:

Note: In case that your OctoPrint installation is only accessible from within a trustworthy network and you don't need Access Control for other reasons, you may alternatively disable Access Control. You should only do this if you are absolutely certain that only people you know and trust will be able to connect to it.

Do NOT underestimate the risk of an unsecured access from the internet to your printer!

[Disable Access Control](#) [Keep Access Control Enabled](#)



avec Terminal a maintenant fait son apparition vous demandant de taper votre login qui est « pi » et votre password qui sera « Raspberry » par défaut Fig 9.

Il est fortement recommandé de changer votre mot de passe et ainsi limiter les intrusions extérieures à votre Raspberry Pi. Pour ceci, tapez « passwd » dans votre terminal. Il vous demandera de rentrer votre mot de passe actuelle ainsi que le nouveau (Fig. 10).

Afin d'optimiser l'utilisation d'Octoprint, vous pouvez étendre le système d'exploitation sur toute la carte SD en tapant la commande suivante dans votre terminal : sudo raspi-config. La fenêtre suivante apparaîtra et vous choisirez « Expand File system ». Cela étendra votre fichier système (Fig. 11).

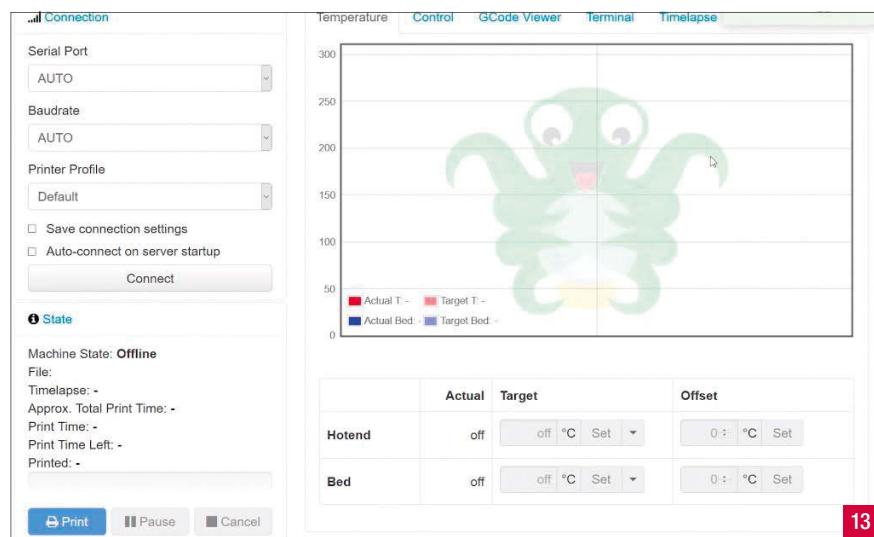
Il vous demandera ensuite de redémarrer votre Pi afin de prendre en compte les changements effectués.

Connexion à notre imprimante grâce à Octoprint

Vous pouvez maintenant ouvrir votre navigateur web (Safari, Chrome, Firefox...) et taper l'adresse IP de votre Raspberry ou « octopi.local » dans l'URL. Ensuite apparaîtra la fenêtre suivante (seulement à la première connexion) où vous devrez rentrer un nouvel identifiant ainsi qu'un mot de passe. Cliquez ensuite sur « Keep Control Access Enabled » pour avoir un accès à votre imprimante 3D à distance de manière sécurisée (Fig. 12).

Rafraîchissez votre navigateur web. Une interface permettant de contrôler notre machine à distance apparaîtra et vous choisirez « Expand File system ». Cela étendra votre fichier système (Fig. 11).

Pensez à cliquer sur « yes » pour une mise à jour s'il vous le demande afin de corriger d'éventuels bugs. Rendez-vous dans l'onglet « System » (en haut à droite) et cliquez sur « Reboot » pour redémarrer le Pi et prendre en compte la mise à jour (Fig. 14).



```
octopi.local - PuTTY
login as: pi
pi@octopi.local's password: 9
```

Set up your printer profile

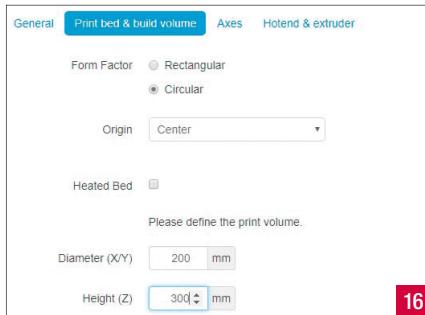
Please make sure the settings below match your printer. If you plan on connecting more than one printer to OctoPrint, you can also configure additional printer profiles under Settings > Printer Profiles

[General](#) [Print bed & build volume](#) [Axes](#) [Hotend & extruder](#)

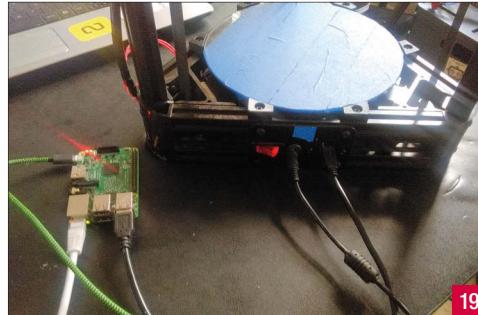
Name	Kossei
Identifier	_default
Model	Generic RepRap Printer

Configurer Octoprint pour notre imprimante 3D

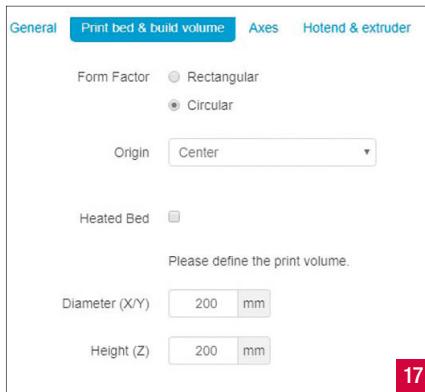
Il est temps maintenant de configurer votre imprimante pour son utilisation avec Octoprint. Rendez-vous dans l'onglet « Settings » dans la partie supérieure « icône avec un engrenage » de votre interface pour ensuite choisir : Printer profiles add profile (Fig. 15). Dans « add profile », entrer un nom spécifique à votre machine et les propriétés de celle-ci (Vitesse de déplacement de X, Y, Z, de l'extrudeur, les dimensions de l'imprimante...) (Fig.16 et Fig. 17). Cliquez ensuite sur « Confirm » pour finir le profil. Cherchez le profil de l'imprimante que vous venez de créer et cliquez sur l'étoile juste à côté de celle-ci pour la rendre effective par défaut. Sur le menu de gauche, choisissez votre imprimante avec le « port » et le « baud rate » et finissez la manœuvre en cliquant sur « Connect » pour se rattacher à notre imprimante Fig. 18.



16



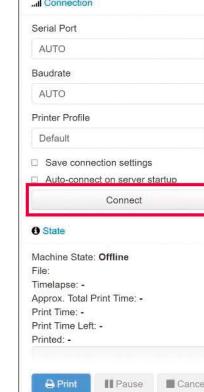
19



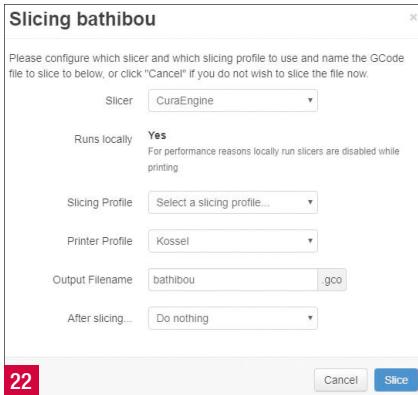
17



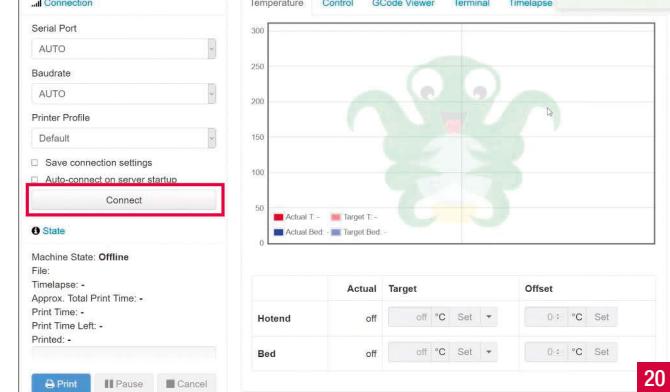
21



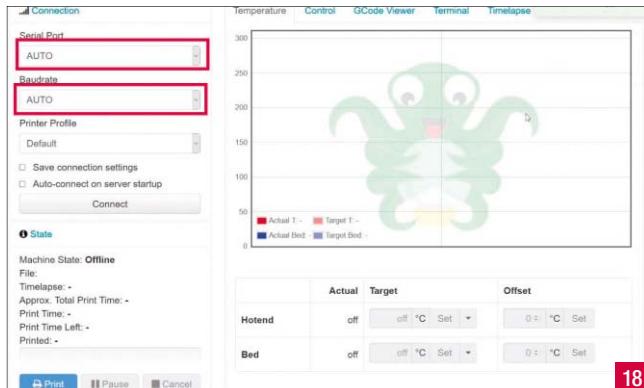
22



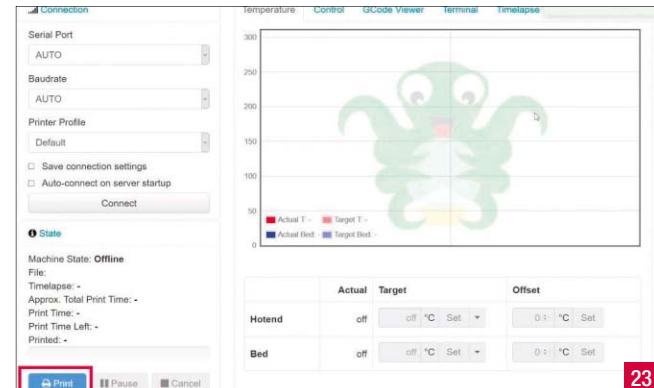
Cancel Slice



20



18



23

Connecter l'imprimante au Pi

Maintenant que nous avons créé notre profil d'imprimante 3D (Kosel, Prusa i3...), nous pouvons connecter notre machine au Raspberry avec un câble USB (Port USB de l'imprimante au port USB du Raspberry Pi), (voir Fig. 19). Rendez-vous dans la partie gauche de l'interface et cliquez sur « Connect » (Fig. 20).

Dans la partie centrale de notre interface, nous avons différents onglets pour la gestion et les retours de l'imprimante :

- Température : température de notre extrudeur et du plateau si vous en avez un ;
- Control : régler la vitesse de déplacement de notre extrudeur et autre ;
- Gcode viewer : visualisation de notre objet ;
- Timelapse : gestion de la visualisation à distance de l'imprimante 3D avec une webcam ;
- Terminal : visualisation des commandes exécutées par l'imprimante.

Pour imprimer un objet, vous cliquez sur

« Upload » qui est situé dans la partie gauche de votre interface et choisissez votre fichier STL

Fig. 21. Vous pouvez si le souhaitez slicer votre fichier 3D avec le logiciel stratificateur de votre choix pour le transformer en Gcode Fig. 22.

Cliquez ensuite sur « Print » pour lancer l'impression (Fig. 23).

À vous maintenant de vous lancer dans l'aventure et de bannir ces câbles USB. J'espère que cet article vous à donner envie de passer à Octoprint. Pensez à partager vos réalisations. C'est un peu ça l'esprit open source. •

Sources :

www.reprap.org

www.octoprint.org

www.raspberrypi.org

Migration d'un legacy avec **GoReplay**

- Yohan Legat
Zenika

Icon made by Pixel perfect (<http://www.flaticon.com/authors/pixel-perfect>), Made by oliver (<http://www.flaticon.com/authors/madebyoliver>) and Freepik (<http://www.flaticon.com/authors/freepik>) from www.flaticon.com

Une fois la nécessité d'un big bang acceptée, du moins sur le principe, il va falloir mettre en place une procédure permettant de garantir, autant que possible, que la mise en production de votre nouvelle application se passera sereinement. C'est là l'objet de cet article. La procédure que nous allons voir a été utilisée avec succès pour migrer un composant majeur du système d'information de Libon. Il s'agissait de l'application backend responsable de la gestion des crédits de communication des utilisateurs ainsi que des autorisations d'appels.

Note: à partir de maintenant, nous appellerons legacy le composant legacy et experimental son remplaçant.

Un impératif : ne vous inspirez pas de l'existant, reproduisez-le !

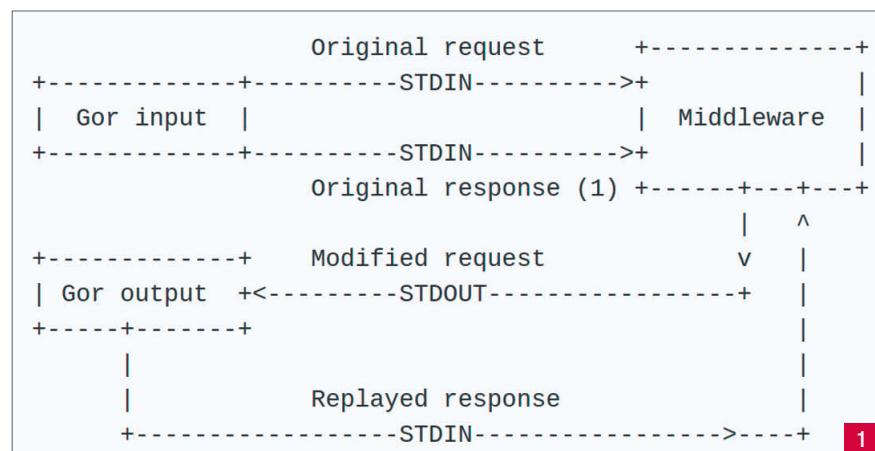
Pourquoi reproduire l'existant à l'identique ?

- Pour pouvoir mettre *legacy* ou *experimental* en production de manière transparente pour les autres composants de votre SI ;
 - Pour faciliter la mise en place d'une phase de double-run (nous en parlerons un peu plus tard).
 - Dans un contexte où la connaissance du domaine s'est perdue, vous ne voulez pas jouer à la roulette russe avec les règles métier.

Vous n'êtes pas satisfait de l'API d'origine ? Ravalez votre frustration et reprenez-la telle quelle ! Les règles métier sont inutilement complexes ? Faites avec ! On ne parle pas de dette technique pour rien, et vous allez devoir la trimballer pendant encore un moment...

Les seuls éléments que vous pouvez modifier sont ceux qui pourront également l'être sur *legacy*. Mais ne comptez pas trop là dessus : si vous pouviez facilement corriger votre *legacy*, vous ne seriez sans doute pas en train de tout réécrire. En revanche, vous êtes libre de modéliser votre nouvelle base de données comme bon vous semble. Une fois que *experimental* sera en production vous aurez tout le loisir de modifier règles métier et APIs.

Remplacer un composant critique d'un SI sans adopter une démarche incrémentale est une stratégie risquée. Toutefois, il arrive que la mise en œuvre d'un tel projet soit également inéluctable : lorsque le fonctionnel est complexe, mal maîtrisé, et que le code existant est mal testé et qu'il freine depuis trop longtemps l'implémentation de nouvelles fonctionnalités métier.



Migration des données de production.

Dans le cas où vous décidez de changer votre modèle de données, vous allez devoir mettre en place un batch permettant de migrer les données de l'ancien modèle vers le nouveau ainsi que de valider cette migration. Avant tout, il est probable que *legacy* ait connu des mises au point difficiles et que des traces de ces accidents soient présentes dans sa base de données. C'est le moment de partir à la recherche de ces incohérences et de les corriger en production. Toujours dans l'optique de faciliter l'étape de migration, n'hésitez pas à supprimer toutes données qui ne seraient plus utilisées. Pour prévenir toute erreur de manipulation, pensez à sauvegarder dans des tables temporaires tout ce que vous serez amené à supprimer ou à modifier.

N'oubliez pas que le jour J vous devrez arrêter tous les services concernés le temps de la migration. Demandez donc à votre ops préféré de mettre à votre disposition une machine de compétition qui vous permettra d'exécuter vos batchs de migration et de validation aussi vite que possible. Plus vous aurez de CPU et de RAM, plus vous pourrez paralléliser et limiter vos I/O.

Idéalement, le code de validation des données ne devrait pas être écrit par le groupe responsable du code de migration. Ces deux groupes doivent travailler isolément pour éviter que des

erreurs de compréhension du modèle de données de *legacy* ne se répandent dans les deux batchs par simple contagion.

Validation fonctionnelle.

Vous avez lancé la première release de *experimental* et vous vous êtes assuré de son bon fonctionnement à l'aide de nombreux tests. Ce n'est toutefois pas suffisant : certaines configurations de données, certains comportements à la limite, vous auront sans doute échappé. Comment s'assurer qu'aucun bug majeur ne sera découvert après la mise en production ? En amenant la production à votre application grâce à GoReplay.

GoReplay est une application open-source écrite en Go et dont le rôle est d'écouter afin de dupliquer un trafic réseau. Pour ce faire, *GoReplay* utilise *libpcap* : il s'agit de la librairie qui est utilisée par *tcpdump* et *Wireshark* pour capturer des paquets de données. Cette opération se déroule donc de manière transparente : vous n'avez pas besoin de configurer quoi que ce soit sur votre serveur de production.

En plus de rejouer des requêtes, *GoReplay* vous permet également de les modifier avant envoi ainsi que de comparer les réponses des serveurs ciblés. Les possibilités sont en réalité plus nombreuses grâce au système d'extension proposé (*middleware*, dans la terminologie *GoReplay*). Il n'existe en revanche pas d'extension prête à l'usage, vous devrez donc dédier

du temps à implémenter le comportement désiré. [1]

Nous allons utiliser les différentes possibilités de *GoReplay* pour mettre en place notre processus de validation métier. Pour cela nous avons besoin d'une instance de *legacy* et de *experimental* dans un environnement de validation dédié. [2]

Et voici dans le détail le processus que vous pouvez mettre en place [3] :

- Exécutez un dump de la base de données *legacy* de production.
- Dans le même temps, lancez *GoReplay* afin d'enregistrer dans un fichier de log les requêtes de production.
- Chargez le dump dans la base de données *legacy* de validation.
- Migrez ces données vers la base de données *experimental*. Assurez-vous que ces données ont été migrées correctement à l'aide du batch de validation dont nous parlions plus haut.
- Lancez *GoReplay* pour rejouer les requêtes de production sur *legacy* et *experimental*. Utilisez la fonctionnalité middleware de *GoReplay* pour comparer les réponses de vos deux serveurs. Elles doivent être identiques. Si votre environnement de validation ne tient pas la charge, *GoReplay* vous permet de réduire la fréquence avec laquelle il jouera vos requêtes.

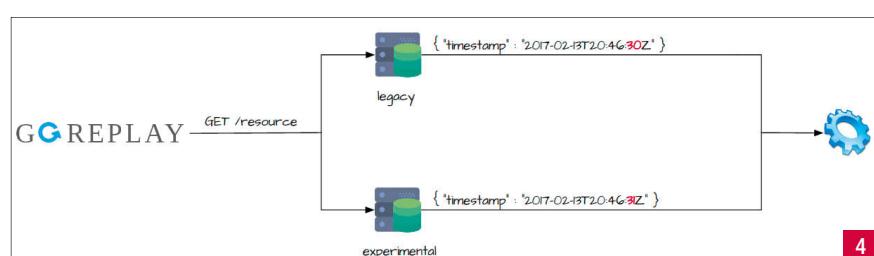
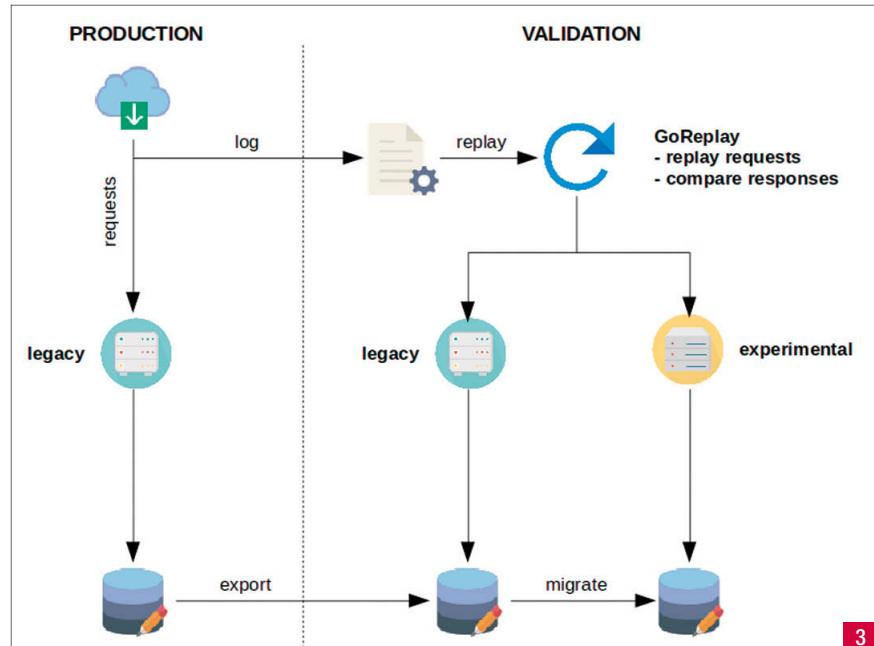
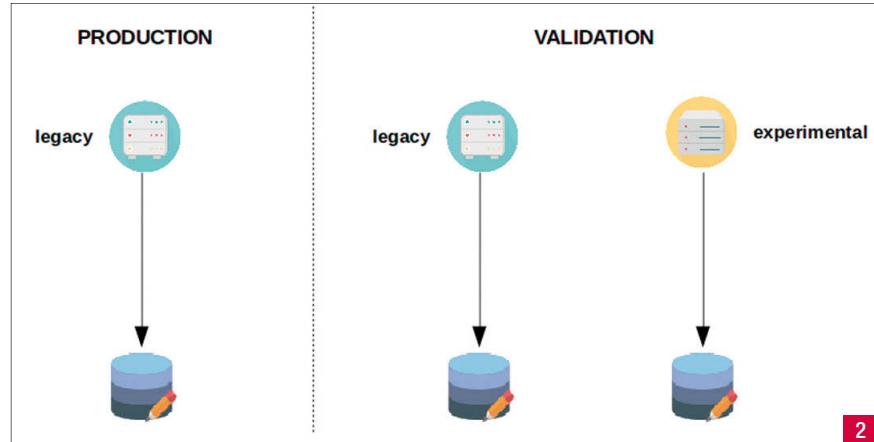
Dans la mesure où les étapes 1 et 2 ne peuvent pas être tout à fait synchronisées, il est possible que le résultat des premières requêtes que vous enregistrez (et donc que vous allez rejouer) soit déjà présentes dans le dump que vous allez extraire de votre base de données de production. Cela ne devrait pas poser de problème : la création d'un utilisateur pourtant déjà présent en base devrait échouer de la même manière sur *legacy* et *experimental*. C'est la seule chose qui importe.

Gestion des conflits entre legacy et experimental.

Même si vous vous êtes efforcé de diverger le moins possible du comportement de *legacy*, il est très probable que vous deviez gérer de nombreux edge-case dans votre middleware. Ainsi, d'éventuels timestamps générés par *legacy* et *experimental* ne seront pas identiques à la milliseconde près. Dans le cas où ils apparaîtraient dans vos payloads, votre middleware devra tolérer de tels écarts. [4]

Un autre problème concerne les identifiants techniques :

- Dans le cas où les identifiants des ressources



créées sont générées aléatoirement vous n'aurez aucune correspondance entre *legacy* et *experimental*. Vous n'aurez pas non plus de correspondance avec les identifiants générés en production. Cela signifie que certaines des requêtes que vous allez rejouer cibleront des identifiants inconnus par *legacy* et *experimental*. Ce n'est pas un problème en soi, du moment que *legacy* et *experimental* réagissent de la même manière (code 404 par exemple).

- Les choses sont plus compliquées dans le cas des identifiants auto-incrémentés : même si

vos séquences sont répliquées sur *legacy* et *experimental*, vous risquez de voir un décalage se produire avec le temps pour peu qu'une création de ressource échoue sur l'une ou l'autre des applications. En conséquence, les identifiants générés en production et présents dans les requêtes de rejoue ne cibleront pas les mêmes ressources sur *legacy* et *experimental*. Ici, une des solutions peut être d'initialiser vos séquences de manière à être certain que l'intersection entre les valeurs générées par la production, par *legacy* et par *experimental* soit vide (on se retrouve alors

dans une situation similaire à celle où les identifiants sont générés aléatoirement).

- Dans tous les cas, une différence entre les identifiants générés en production et en validation risque de diminuer considérablement l'intérêt d'un double-run sur le long terme puisque toutes les requêtes ciblant ces ressources ne pourront pas aboutir. [5]

Une solution idéale consisterait à réaliser un mapping au sein du *middleware GoReplay* entre les identifiants de *production*, de *legacy* et de *experimental*. Il faudrait alors pouvoir, lors du rejeu, modifier les requêtes de manière à insérer le bon identifiant pour chaque plateforme. A ce jour, une telle fonctionnalité n'a pas encore été implémentée dans *GoReplay*.

Enfin, même des temps de réponses différents peuvent suffire à provoquer des divergences.

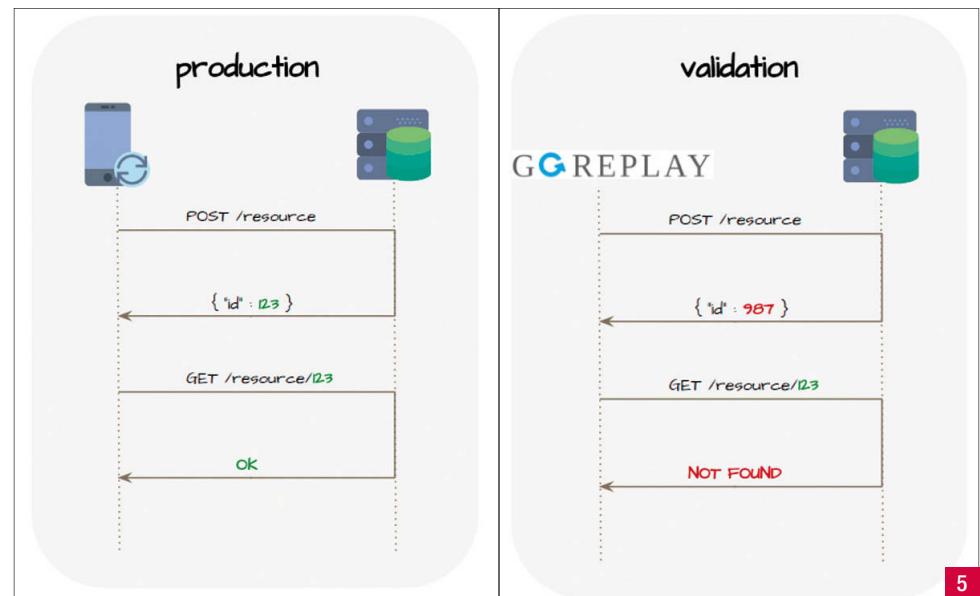
Ainsi, si la durée de création d'une ressource prend plus de temps sur *experimental* que sur *legacy*, et que cette requête de création est immédiatement suivie pour un appel à *GET* sur la même ressource, vous risquez de générer une erreur sur *experimental*. [6]

Ici, une solution simple consiste à configurer *GoReplay* de manière à ce qu'il augmente artificiellement la latence entre deux requêtes successives.

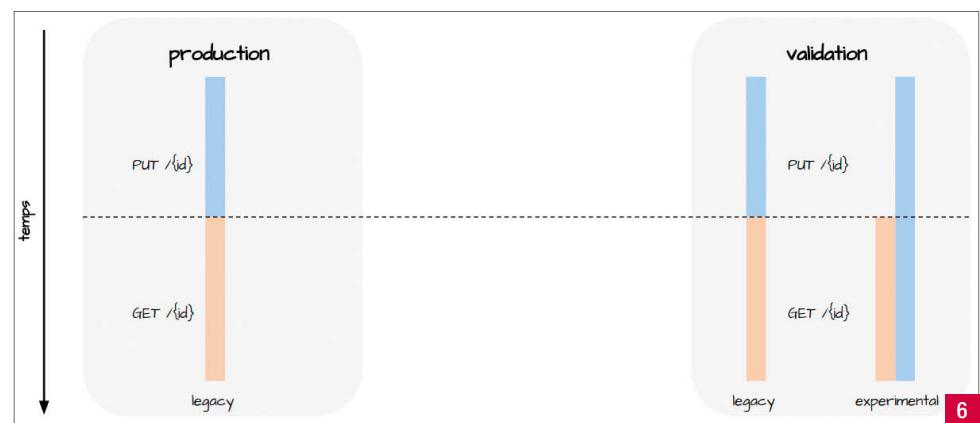
De prime abord, on pourrait croire que l'ensemble de ces différences entre les réponses de *legacy* et de *experimental* sont mineures et qu'il est donc inutile de passer du temps à les aplatisir en complexifiant le code de validation. Le problème est que, aussi mineures soient elles, elles risquent de polluer vos logs et de masquer de véritables bugs. Pour s'en convaincre, il suffit de faire le calcul suivant : considérez 100.000 requêtes enregistrées au cours d'une seule journée et un taux d'échec de 0.5% lors du rejeu dû à des raisons purement techniques (identifiants autogénérés différents, écart d'une seconde entre deux timestamps etc), c'est 500 triplets (requêtes, réponse de *legacy*, réponse de *experimental*) qui apparaîtront dans vos logs d'erreurs et que vous devrez analyser un par un pour déterminer s'ils sont ou non liés à des bugs dans *experimental*. Ce n'est pas une situation acceptable.

Validation technique.

Vous êtes désormais confiant dans le code de *experimental*. Il vous reste à prouver que l'application tiendra la charge une fois en production. On pourrait croire que cette preuve a déjà été apportée lors de la phase de validation fonctionnelle, mais souvenez-vous que nous testons alors une instance de *experimental* au sein



5



6

d'un environnement isolé. Ici l'idée est de procéder à une validation technique dans des conditions identiques à celles de la production (même hardware, même nombre d'instances, même base de données).

Pour cela vous pouvez encore tirer profit des fonctionnalités de *GoReplay* : déployez *experimental* en production et configuez *GoReplay* pour dupliquer toutes les requêtes en temps réel. L'application *legacy* fait toujours foi pour vos utilisateurs et vous n'avez pas besoin de comparer les réponses cette fois-ci. Vous devez juste vous assurer qu'après plusieurs jours sous le régime de la production *experimental* ne souffre pas de fuites mémoire ou de ralentissements liés à un garbage collector mal optimisé (gceasy est votre ami). En conséquence vous n'avez pas besoin d'arrêter vos services de production le temps de migrer vos données : certaines requêtes ne pourront pas être traitées par *experimental* mais cela importe peu. Chez *Libon*, cette phase de validation technique n'a pas été superflue : elle a mis en évidence

des problèmes de garbage collector que nous avons résolus en retravaillant la modélisation de nos *event* (nous utilisons un architecture event-sourced) ainsi que les algorithmes d'invalidation de nos caches applicatifs.

Epilogue

Les nombreuses phases de validation fonctionnelle et technique qui ont précédé la mise en production ont permis de rassurer les équipes sur la stabilité du code ainsi que de les familiariser avec la procédure de migration des données.

Mises à part les 30 minutes de downtime nécessaires à la migration des données, nous avons pu remplacer un composant legacy majeur de notre SI en toute transparence pour nos clients. Un seul bug mineur a été détecté deux semaines après la mise en production de *experimental*. Il s'agissait d'une opération d'administration qui n'avait jamais été effectuée au cours des phases de double-run... ●

Coder une cracktro sur Amiga

Partie 2

• Denis Duplan
Sociologue et développeur
à ses heures.

Blog : <http://www.stashofcode.fr>

Cet article est le second, et le dernier, d'une série consacrée au développement d'une cracktro sur Amiga (Figure 1). Dans le premier article, nous avons vu comment mettre en place un environnement de développement en assembleur 68000 dans le contexte d'un émulateur Amiga, et nous sommes rentrés dans une présentation d'un des deux coprocesseurs graphiques, le Blitter, exemple à l'appui. Ici, nous présenterons pareillement le Copper et nous conclurons sur l'intérêt de revisiter ce passé.

Pour rappel, vous pouvez voir et entendre ce que la cracktro donne à l'URL suivante :

<https://www.youtube.com/watch?v=UQ6R32HAp4Y>

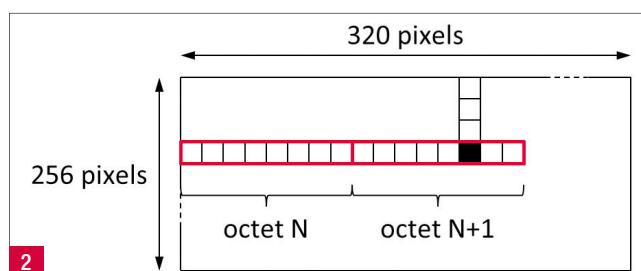
Et vous pouvez en télécharger le code (à peine un gros millier d'instructions en assembleur 68000) et les données à l'URL suivante :

<http://www.stashofcode.fr/code/coder-une-cracktro-sur-amiga-1/cracktro.zip>

Les bases de l'affichage

Comme chacun sait, une image affichée à l'écran est composée de pixels, chaque pixel étant codé en mémoire par une valeur qui permet de déterminer la couleur dans laquelle il doit être affiché. De nos jours, cette valeur est généralement un quad, les composantes rouge, verte et bleue de la couleur du pixel étant codées chacune sur un octet (le quatrième octet étant réservé au degré de transparence, ou alpha). Sur Amiga, la solution est très différente.

Exception faite du mode Hold-And-Modify (HAM) exceptionnellement utilisé, l'affichage sur Amiga fonctionne sur le principe suivant. Pour afficher en N couleurs, N étant une puissance de 2, il faut N plans de bits, ou bitplanes, dont les dimensions correspondent à celles de la surface affichée – par exemple, 320x256 pixels. Le premier bitplane fournit les bits 0 des pixels, le deuxième fournit les bits 1 des pixels, etc. (Figure 2). L'ensemble des bits d'un pixel issus des bitplanes aux coordonnées du



Un bitplane.

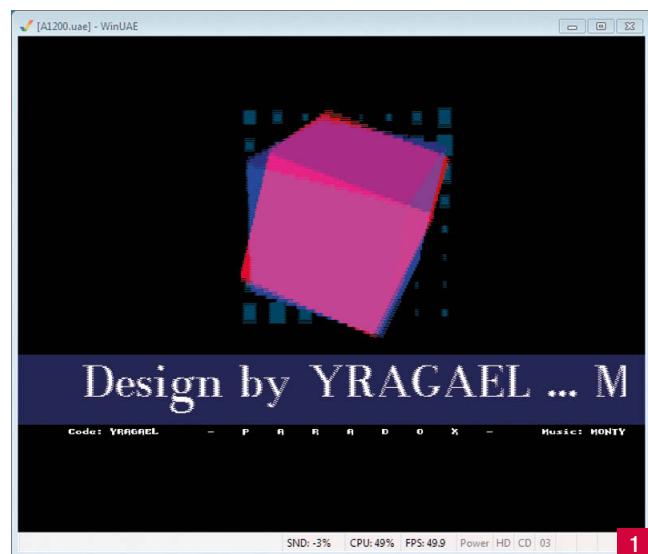
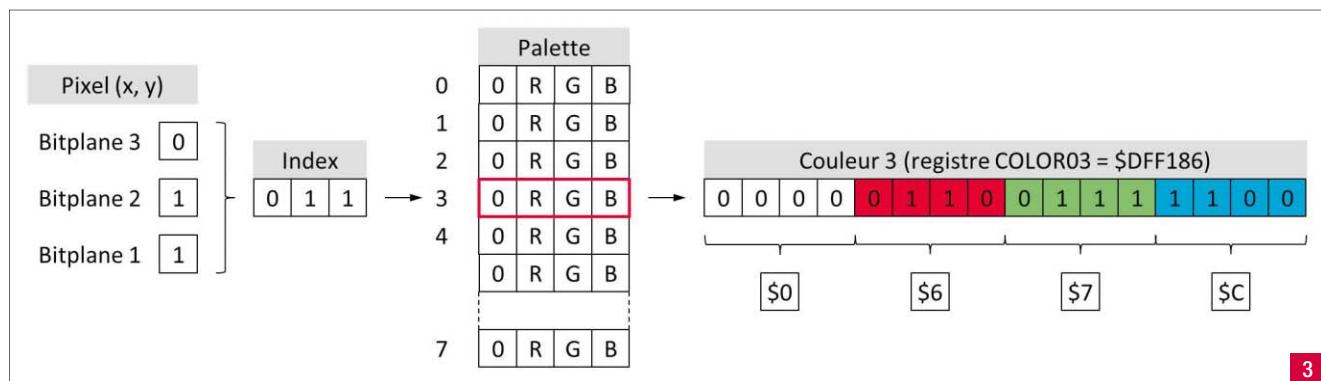


Figure 1 : La cracktro en cours d'exécution dans WinUAE.

pixel en question constitue une valeur. Cette valeur correspond à l'index de la couleur dans laquelle le pixel doit être affiché. Cette couleur est extraite d'une palette de couleurs correspondant à la suite des registres COLOR00 à COLOR31. Enfin, cette couleur est codée sur un mot où chacune des composantes rouge, verte et bleue est codée sur quatre bits, ce qui donne 4 096 possibilités (Figure 3). Au passage, il convient de noter que les N bitplanes sont par convention étrangement numérotés de 1 à N . L'Amiga peut afficher en basse ou haute résolution, ce qui joue sur le nombre de couleurs :

- En basse résolution (ie : 320x256 en PAL), l'Amiga peut afficher au plus 5 bitplanes, donc 32 couleurs. Il existe un mode particulier, l'Extra Half-Brite (EHB) où l'Amiga peut afficher 6 bitplanes, mais les 32 cou-



Adressage de la couleur d'un pixel (affichage en 8 couleurs).

leurs supplémentaires sont alors des variantes des 32 premières dont la luminosité est automatiquement divisée par 2.

- En haute résolution (ie : 640x256 en PAL), l'Amiga peut afficher au plus 4 bitplanes, donc en 16 couleurs.

Dans les deux cas, le nombre de lignes peut être doublé en activant un mode entrelacé. C'est au prix d'un scintillement, car l'entrelacé consiste à afficher alternativement les lignes paires et impaires en misant sur un effet de rémanence visuelle.

Ces possibilités ont été étendues au fil des évolutions du hardware, mais on se limitera ici à évoquer l'Amiga 500, donc l'Original Chip Set (OCS) et non les suivants – Enhanced Chip Set (ECS) pour l'Amiga 500+ et Advanced Graphics Architecture (AGA) pour l'Amiga 1200.

Parler de basse ou haute résolution en évoquant un nombre de pixels affichés horizontalement est un abus de langage. En fait, il faut entendre que le faisceau d'électrons met un certain temps à balayer une ligne de l'écran, et qu'il est possible de lui demander de tracer plus ou moins longtemps, donc sur une certaine longueur, un pixel : 140 nanosecondes en basse résolution, 70 en haute résolution.

C'est pourquoi indépendamment de la résolution, il faut distinguer la surface d'affichage et les données affichées. La surface correspond à la partie de la région balayée par le faisceau d'électrons effectivement utilisée. Elle doit être spécifiée via les registres DIWSTART et DIWSTOP. D'autres registres, DDFSTART et DDFSTOP, permettent de spécifier les positions horizontales du faisceau d'électrons entre lesquelles les données doivent être lues et affichées au rythme imposé par la résolution adoptée. Les données doivent être lues un peu avant leur affichage pour laisser le temps au hardware de jouer sur le faisceau d'électrons.

Jouer la « playlist » vidéo

Le Copper est un des coprocesseurs de l'Amiga, essentiellement utilisé pour contrôler l'affichage en écrivant dans des registres tels que ceux déjà cités pour spécifier des paramètres de ce dernier.

Avant d'aller plus loin, il faut clarifier deux points :

- Tout d'abord, le recours au Copper pour contrôler l'affichage est facultatif. Il serait parfaitement possible d'écrire dans les registres avec le CPU pour parvenir au même résultat. Toutefois, il faudrait synchroniser ces écritures avec la position du faisceau d'électrons, et ne pas manquer de rafraîchir les registres dont le contenu est modifié lors d'une trame – par exemple BPL1PTH et BPL1PTL, registres qui contiennent l'adresse du bitplane 1, sont incrémentés tandis que ce bitplane est affiché. Dans ces conditions, pourquoi ne pas utiliser le Copper ?
- Ensuite, il a été dit, le Copper est *essentiellement* utilisé pour contrôler l'affichage. C'est que le Copper peut écrire dans d'autres registres qui permettent de contrôler d'autres fonctionnalités du hardware – pas tous les registres disponibles, mais bon nombre d'entre eux. Par exemple, le Copper peut fort bien écrire dans les registres qui contrôlent le Blitter pour paramétrier et déclencher la copie d'un bloc en mémoire ou un tracé de ligne.

Dans le précédent article, nous avons vu que le Blitter se programme en attaquant des registres. Pour sa part, le Copper se programme aussi en attaquant des registres, mais surtout en lui fournissant une liste composée d'instructions : la Copper list, qui est exécutée automatiquement à chaque trame. Dans la Copper list, une instruction est fournie sous la forme de deux mots. Autrement dit, le Copper se programme directement en opcodes, généralement écrits en hexadécimal. Le Copper détecte de quelle instruction il s'agit en se basant sur la combinaison des deux bits 0 des mots d'une instruction.

Le Copper comprend simplement trois instructions :

- WAIT indique au Copper qu'il doit attendre que le faisceau d'électrons atteigne ou dépasse certaines coordonnées à l'écran. Si l'ordonnée est simplement le numéro de la ligne indépendante de la résolution, l'abscisse est le numéro d'une série de pixels dont la longueur dépend de cette dernière. Ainsi, en 320 x 256, l'abscisse peut être précisée tous les quatre pixels. Ordonnée et abscisse peuvent être masquées, mais cette fonctionnalité est exceptionnellement utilisée.
- Généralement, une instruction WAIT prend donc la forme d'un premier mot `((y & $FF) << 8) | ((x & $FE) << 1) | $0001` et d'un second mot `$FFFE`, le masquage étant ainsi inhibé.
- MOVE indique au Copper qu'il doit écrire un mot dans un des registres dont l'offset (nécessairement pair) est fourni en présumant qu'il va être ajouté à l'adresse `$DFF000` pour localiser ce registre en mémoire. Par exemple, `$01820FF0` permet d'écrire `$0FF0` dans `$DFF182` (registre COLOR01), ce qui revient à passer la couleur 1 de la palette à jaune (dans ce registre, la couleur est codée sur les 12 premiers bits, 4 bits par composante). Le Copper ne peut pas écrire dans tous les registres `$DFFxxx`, et dans certains il ne peut écrire que sous condition qu'un bit particulier soit positionné – ce bit permet donc de sécuriser l'exécution de la Copper list en prévenant des écritures par débordement.
- SKIP indique au Copper qu'il doit sauter l'instruction qui suit si jamais le faisceau d'électrons atteint ou dépasse certaines coordonnées à l'écran. Ces coordonnées sont spécifiées comme pour l'instruction WAIT.

Un WAIT `$FFFFFFFE` indique au Copper qu'il a atteint la fin de la Copper list. Autant l'intérêt de MOVE et de WAIT est évident, autant celui de SKIP ne l'est pas. SKIP est utile dans le contexte d'usages (très) avancés, tout particulièrement :

- modifier les registres COPxLCH, COPxLCL et COPJMPx pour sauter de Copper list en Copper list et ainsi réaliser des boucles ;
- déclencher et attendre le Blitter – un bit d'un WAIT dont l'existence n'a pas été mentionnée plus tôt le permet.

La Copper list doit être écrite en mémoire à une adresse qu'il faut fournir au Copper via ses registres COP1LCH et COP1LCL. Une fois l'adresse écrite, une écriture dans le registre COPJMP1 permet de demander au Copper de désormais traiter en boucle la Copper list désignée – tout comme BLTSIZE, COPJMP1 est un strobe. Il ne reste plus alors qu'à activer l'accès à la mémoire du Copper pour visualiser le résultat à l'écran, ce qui s'effectue via certains bits du registre DMACON.

Dans la cracktro, la Copper list permet d'organiser l'affichage de la manière suivante, la transition entre la première et la seconde partie s'effectuant à l'aide d'un WAIT :

- Une première partie de l'écran en basse résolution de 320x160 pixels sur 5 bitplanes (0 et 1 pour un des cubes, 2 et 3 pour l'autre, et 4 pour le damier) ;
- Une seconde partie de l'écran en haute résolution et entrelacée de 640x92 pixels sur 1 bitplane.

Par ailleurs, la Copper list est modifiée à chaque trame pour modifier les valeurs stockées dans les registres des couleurs des cubes en fonction des faces visibles. En fait, il y a deux Copper list identiques, l'une activant l'autre quand elle se termine en écrivant dans COP1LCH et COP1LCL l'adresse de son homologue – c'est une condition du fonctionnement de l'affichage en entrelacé.

Il serait trop long de restituer ici le code de la Copper list de la cracktro. Réécrite à l'occasion de la rédaction de cet article en s'en inspirant, voici une Copper list basique. Elle commande l'affichage d'un écran en

320x256 en deux couleurs (un unique bitplane) en changeant la couleur de fond à mi-hauteur (Figure 4). Les variables `copperlist` et `bitplane` contiennent les adresses de zones de mémoire réservées par allocation système pour stocker les données de la Copper list et du bitplane.

```

DISPLAY_DX=320
DISPLAY_DY=256
DISPLAY_X=$81
DISPLAY_Y=$2C
DISPLAY_DEPTH=1

movea.l copperlist,a0

movea.l copperlist,a0
move.w #$008E,(a0)+ ;DIWSTRT
move.w #((DISPLAY_Y<<8)DISPLAY_X,(a0)+)
move.w #$0090,(a0)+ ;DIWSTOP
move.w #((DISPLAY_Y+DISPLAY_DY-256)<<8)|(DISPLAY_X+DISPLAY_DX-256),(a0)+move.w #$0100,(a0)+ ;BPLCON0
move.w #((DISPLAY_DEPTH<<12)|$0200,(a0)+move.w #$0102,(a0)+ ;BPLCON1
move.w #$0000,(a0)+move.w #$0104,(a0)+ ;BPLCON2
move.w #$0000,(a0)+move.w #$0092,(a0)+ ;DDFSTRT
move.w #((DISPLAY_X-17)>>1)&$00FC,(a0)+move.w #$0094,(a0)+ ;DDFSTOP
move.w #((DISPLAY_X-17+((DISPLAY_DX>>4)-1)<<4)>>1)&$00FC,(a0)+move.w #$0108,(a0)+ ;BPL1MOD uniquement car un seul bitplane
move.w #0,(a0)+

;Adresse du bitplane

move.w #$00E0,(a0)+ ;BPL1PTH
move.l bitplane,d0
swap d0
move.w d0,(a0)+move.w #$00E2,(a0)+ ;BPL1PTL

```



Figure 4 : Exécution de la Copper list de base.

```

swap d0
move.w d0,(a0)+

;Couleur de fond à rouge

move.w #$0180,(a0)+ ;COLOR00
move.w #$0F00,(a0)+

;Attendre le milieu de l'écran (128 lignes sous $32)

move.w #((DISPLAY_Y+(DISPLAY_DY>>1)&$00FF)<<8)|$0001,(a0)+move.w #$FF00,(a0)+

;Couleur de fond à bleu

move.w #$0180,(a0)+ ;COLOR00
move.w #$000F,(a0)+

;Comptabilité ECS avec l'AGA

move.l #$01FC0000,(a0)+

;Fin de la Copperlist

move.l #$FFFFFFFE,(a0)

;Activer la Copper list

move.w #$7FFF,$DFF096 ;DMACON
move.l copperlist,$DFF080 ;COP1LCH et COP1LCL
clr.w $DFF088 ;COPJMP1
move.w #$8380,$DFF096 ;DMACON

```

Au passage, un effet très couru consiste à enchaîner les MOVE pour changer la couleur de fond sur le nombre de pixels que le faisceau d'électrons parcourt le temps que cette instruction soit exécutée ; soit le temps d'afficher 8 pixels en basse résolution. En répétant une séquence de couleurs formant un dégradé cyclique sur une ligne, et en répétant cette opération sur plusieurs lignes en commençant à chaque ligne dans la séquence à un indice qui varie de ligne en ligne selon une fonction



Figure 5 : Superbe effet plasma dans la Megademo de Humanoids.

telle qu'un sinus, on produit un effet « plasma » qui peut être animé ([Figure 5](#)).

Dans la cracktro, la Copper list est notamment modifiée à chaque trame pour étendre artificiellement la palette des couleurs utilisées pour afficher les faces d'un cube. En effet, comme expliqué dans l'article précédent, chaque cube est affiché sur deux bitplanes, ce qui permet d'afficher en quatre couleurs.

Or les faces opposées d'un cube sont mutuellement exclusives. Par exemple, quand la face avant est visible, la face arrière est nécessairement cachée – pour déterminer si une face est visible, l'orientation de sa normale est testée. Dès lors, l'astuce consiste à afficher des faces opposées de sorte que leurs pixels référencent la même couleur dans la palette, couleur qui leur est réservée. Quand une face est visible, sa couleur est modifiée en écrivant dans la Copper list pour écraser la valeur que cette dernière écrit dans le registre correspondant. Par exemple, sans que cela corresponde nécessairement à ce qui se retrouve dans la cracktro :

Face	Bitplane 2	Bitplane 1	Registre	Valeur
Haut	Non	Oui	%01 = COLOR01	\$0F00
Bas	Non	Oui	%01 = COLOR01	\$00F0
Gauche	Oui	Non	%10 = COLOR02	\$000F
Droite	Oui	Non	%10 = COLOR02	\$0OFF
Avant	Oui	Oui	%11 = COLOR03	\$OFF0
Arrière	Oui	Oui	%11 = COLOR03	\$0FOF

Chaque face étant affichée dans une couleur qui lui est propre, le cube semble affiché en sept couleurs alors qu'il n'est affiché que sur deux bitplanes.

Sprites, dual-playfield et scrolling... hardware !

Le hardware de l'Amiga ne permet pas que d'afficher des bitplanes. Il permet aussi d'afficher des sprites, de découpler l'affichage des bitplanes sur deux plans – le dual-playfield –, de faire défiler un plan des bitplanes : autant de fonctionnalités qu'il est encore une fois possible de contrôler au CPU ou au Copper en écrivant dans certains registres.

La cracktro n'utilise aucune de ces fonctionnalités, mais il serait malheureux de ne pas profiter de l'occasion pour les évoquer car elles permettent de réaliser des effets graphiques qu'on retrouve dans nombre de démos et autres cracktros.

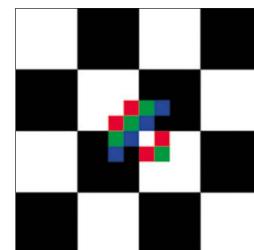
Les sprites. Chacun sait ce qu'est un sprite : une image qui doit être affichée par-dessus un décor dans lequel elle se déplace en évitant de recouvrir les pixels qui, dans le sprite, sont censés être transparents – une couleur ou un masque permet de les identifier. Normalement, l'affichage des sprites est effectué au niveau du processeur, ce qui implique d'écrire le code requis pour non seulement afficher mais aussi effacer le sprite en restaurant la partie du décor qu'il recouvre dans sa position antérieure. Sur Amiga, ces sprites software sont souvent affichés au Blitter car ce dernier permet de combiner logiquement lors d'une seule copie les pixels du sprite, de son masque et du décor – raison pour laquelle ces sprites sont désignés comme des « bobs », pour « Blitter objects ». Aussi le terme de sprite est-il réservé pour désigner les sprites hardware du Copper.

Le hardware peut afficher huit sprites de 16 pixels de large en 4 couleurs dont une couleur transparente, sur une hauteur illimitée. Les sprites sont couplés (le 0 avec le 1, le 2 avec le 3, etc.). Dans un couple, les sprites partagent la même palette (0 et 1 utilisent les couleurs 16 à 19, 2 et 3 utilisent les couleurs 20 à 23, etc.), et ils peuvent de plus être combinés. Deux sprites combinés forment un sprite toujours de 16 pixels de large

et de hauteur illimitée, mais en 16 couleurs dont une couleur transparente. Enfin, un système de priorités permet de gérer finement quel sprite est affiché devant ou derrière quel playfield (cf. ci-après).

Un sprite se contrôle très simplement. En effet, ses données se présentent sous la forme d'une liste de mots. Les deux premiers permettent de spécifier l'abscisse, l'ordonnée, et la hauteur du sprite. Suivent deux mots par ligne du sprite : le premier donne les bits de poids faible et le second les bits de poids fort permettant de déterminer l'indice de la couleur des pixels correspondants du sprite. Un couple de mots nuls marque la fin des données du sprite. Par exemple, un sprite formant un petit damier cases de 4x4 pixels dans ses 4 différentes couleurs affiché en (23, 23) dans un repère classique commençant en (\$81, \$2C) ([Figure 6](#)) :

```
sprite:    dc.w $444C, $5401    ;Coordonnées (codage un peu pénible)
          REPT 4
          dc.w $0F0F, $00FF    ;Première ligne de cases
          ENDR
          REPT 4
          dc.w $FOFO, $0FF0    ;Seconde ligne de cases
          ENDR
          REPT 4
          dc.w $0FOF, $FF00    ;Troisième ligne de cases
          ENDR
          REPT 4
          dc.w $FOFO, $FO0F    ;Quatrième ligne de case
          ENDR
          dc.w $0000, $0000    ;Fin
```



6

Un sprite de 16 pixels en 4 couleurs affiché sur un bitplane.

D'ingénieux codeurs maîtrisant parfaitement le hardware ont élaboré des techniques très élaborées pour enrichir l'affichage en jouant avec le Copper, notamment avec les sprites. Le site Codetapper (<http://codetapper.com/amiga/sprite-tricks/>) permet de constater tout ce que l'analyse de la Copper list de différents jeux à succès révèle... Passionnant !

Le dual playfield. En positionnant simplement un bit dans un registre, les 6 bitplanes qu'il est possible d'utiliser peuvent être regroupés : les pairs, d'une part, les impairs, d'autre part. Chaque ensemble constitue alors un playfield, c'est-à-dire un décor à part entière doté d'une palette de 8 couleurs, dont une couleur transparente.

Le dual playfield permet d'afficher les deux décors avec effet de transparence, l'un laissant entrevoir l'autre dans ses zones de pixels de couleur transparente, sans avoir à faire intervenir le CPU. Comme par ailleurs, le scrolling de chaque playfield peut être géré indépendamment de l'autre tant verticalement qu'horizontalement (cf. ci-après), le dual playfield permet de réaliser un effet de parallaxe où un décor se déplace derrière l'autre à une vitesse différente, comme s'il se trouvait à une certaine distance de l'autre le long d'un axe qui, partant de l'observateur, s'éloigne vers le fond de l'écran.

Le scrolling. Pour afficher un bitplane, il faut en fournir l'adresse au hardware dans une paire de registres : BPL1PTH et BPL1PTL pour le bit-

plane 1, BPL2PTH et BPL2PTL pour le bitplane 2, etc. Or, que ce soit pour le groupe des bitplanes pairs, d'une part, ou celui des bitplanes impairs, d'autre part, il est possible de contrôler deux paramètres du hardware lorsqu'il lit les données d'un groupe de bitplanes pour afficher une ligne de pixels :

- Retarder de 0 à 15 pixels l'affichage d'une ligne ;
- Sauter un certain nombre d'octets (le modulo) à la fin d'une ligne.

En combinant ces deux paramètres, il est possible de produire un scrolling non seulement horizontal, mais aussi vertical, un effet ici encore totalement pris en charge par le hardware. C'est ce mécanisme qui peut être utilisé pour produire l'effet de parallaxe dont il a été question plus tôt. Pour être complet dans cet inventaire des fonctionnalités vidéo du hardware, il faut mentionner la détection de collisions à la précision du pixel. Toutefois, c'est une fonctionnalité plus utile aux jeux qu'aux démos et autres cracktros. Quoi que... ?

Que retenir de tout cela pour aujourd'hui ?

Comme expliqué dans l'introduction du premier article de cette série, le propos de cette présentation illustrée de la programmation des deux coprocesseurs graphiques de l'Amiga en assembleur n'était certainement pas d'inciter à programmer sur cette machine de nos jours, même si cela peut être des plus instructifs.

Le propos était plutôt d'enrichir la culture générale de ceux qui n'ont jamais eu l'occasion de s'adonner à une telle programmation en leur montrant brièvement en quoi elle pouvait consister. Au passage, il s'agissait aussi de montrer à tous comment l'affichage d'une image à l'écran peut, à bas niveau, fonctionner – un mécanisme qu'il est d'autant plus essentiel de connaître que son principe n'a, somme toute, guère évolué. Rappelons qu'il subsiste un petit monde de demomakers qui produisent

des exercices de style fascinants mêlant graphismes, musiques et code, désormais bien plus sophistiqués que la simple cracktro présentée – qui, même si elle n'était pas mal, était loin d'être un chef-d'œuvre de l'époque dans son genre. On ne saurait trop conseiller aux jeunes de rejoindre cette « scène ». Même s'ils ne feront pas de la programmation de routines graphiques leur métier, ils acquerront une connaissance du hard et du soft qui leur servira dans bien d'autres circonstances par la suite.

Ah ! Peut-être un conseil pour tous pour terminer... Comme déjà mentionné, c'est l'*Amiga Hardware Reference Manual* rédigé par les ingénieurs de Commodore qui a servi de référence pour élaborer cette série d'articles. Autrement dit, l'information à la source. Or c'est toujours à la source qu'il faut s'alimenter pour découvrir une technologie : même si elle paraît indigeste parce que très technique, privilégiez la documentation de référence sur la documentation dérivée. Par exemple, si vous souhaitez faire du JavaScript, commencez par lire la spécification ECMAScript. A défaut, vous risquez de penser que JavaScript est un langage orienté objet à base de classes et, commettant cette erreur, vous passerez à côté d'aspects fondamentaux du langage. •

Liens utiles

From Bedrooms to Billions: The Amiga Years :

<http://www.frombedroomstobillions.com/amiga/>

WinUAE : <http://www.winuae.net/>

Amiga Forever : <https://www.amigaforever.com/>

ASM-One : <http://www.theflamearrows.info/documents/ftp.html>

ReqTools : <http://aminet.net/package/util/libs/ReqToolsUsr.lha>

Amiga Hardware Reference Manual :

https://archive.org/details/Amiga_Hardware_Reference_Manual_1985_Commodore

DevCon #4
100 %
Raspberry Pi
& Co.

/IoT
/Serveur
/Production
/Docker

26/10
2017



Inscrivez-vous sur www.programmez.com

C'est le progrès



CommitStrip.com



Une publication Nefer-IT, 57 rue de Gisors, 95300 Pontoise - redaction@programmez.com
Tél. : 09 86 73 61 08 - Directeur de la publication & Rédacteur en chef : François Tonic
Secrétaire de rédaction : Olivier Pavie
Ont collaboré à ce numéro : S. Saurel
Nos experts techniques : V. Loquet, Z. Abidi, C. Lakech, R. Linsolas, Jean-Marie HEITZ, Xavier FLAMANT, C. De Vandiere, F. Pigière, O. Chopin, M. Pean, W. Ben Rabah, C. Pichaud, M. Canada, B. Roussely, M. Walckenaer, D. Duplan, Y. Legat, L. Kempf, R. Pinon, M. Faharasmine, C. Gigax, Ha
Couverture : © Yuravector - Maquette : Pierre Sandré.
Publicité : PC Presse, Tél. : 01 74 10 16 30, Fax : 01 40 90 70 81 - 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 : 09 67 32 09 34
Contacts : Rédacteur en chef : ftonic@programmez.com - Rédaction : redaction@programmez.com - Webmaster : webmaster@programmez.com - Publicité : benoit.gagnaire@programmez.com - Evenements / agenda : redaction@programmez.com
Dépôt légal : à parution - Commission paritaire : 1220K78366 - ISSN : 1627-0908 - © NEFER-IT / Programmez, septembre 2017
Toute reproduction intégrale ou partielle est interdite sans accord des auteurs et du directeur de la publication.

Abonnement : Service Abonnements PROGRAMMEZ, 4 Rue de Mouchy, 60438 Noailles Cedex. - 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 : 35 € (monde entier) souscription sur www.programmez.com



Sur abonnement ou en kiosque

Le magazine des pros de l'IT



Mais aussi sur le web

DEVOPS D-DAY

MARSEILLE JEUDI 9 NOVEMBRE 2017

Orange Vélodrome

LA JOURNÉE DE CONFÉRENCES IT LA PLUS ATTENDUE EN FRANCE



ORACLE®

 Microsoft

TREEPTiK.
A LINKBYNET COMPANY

WWW.DEVOPS-DDAY.COM