

L'expérience Sugar Labs préfigure-t-elle une révolution éducative du XXIe siècle ?

Du projet One Laptop per Child (ou OLPC) les grands médias ont surtout retenu qu'il s'agissait de mettre un ordinateur entre les mains des enfants des pays défavorisés. Confondant la fin et les moyens ils sont alors souvent passés totalement à côté de son intérêt principal qui est pédagogique. Negroponte n'a de cesse à juste titre de le répéter : « le projet OLPC n'est pas un projet informatique, c'est un projet éducatif ».



Lorsqu'une écolière Uruguayenne et un écolier Uruguayen allument leur petit ordinateur vert, ils se retrouvent sur une interface qui est fort différente du classique environnement graphique d'un Mac, Windows ou d'une distribution GNU/Linux.

Ici on abandonne la métaphore du bureau. Applications et fichiers sont bien entendu toujours présents mais ce qui est mis en avant c'est l'interaction avec les autres, ce qui apparaîtra de suite à l'écran c'est la présence du camarade, ce sur quoi il travaille, sachant qu'il est alors facile de le rejoindre pour collaborer.

Cette interface innovante et pleine de promesses s'appelle **Sugar** (cf vidéo). Elle est déjà massivement utilisée dans des pays comme l'Uruguay (cf vidéo) et nous voici alors projetés à des années-lumière de ce qu'une école française peut proposer

non seulement comme outil mais aussi et surtout comme conception générale de sa fonction et de ses missions^[1].

En matière d'éducation et de nouvelles technologies, il y a ceux qui pensent qu'il est important de savoir comment mettre en gras dans Word, c'est-à-dire apprendre le mode d'emploi d'un logiciel propriétaire, et il y a ceux qui veulent en profiter pour... changer le monde !

Le créateur de Sugar, Walter Bender, est de ceux-là. Simon Descarpentries l'a rencontré pour nous à Paris à l'occasion de l'Open World Forum 2010 et il a gentiment accepté de nous livrer un texte inédit nous présentant la jeune fondation Sugar Labs, sa philosophie, ses objectifs et ses réalisations.

Il ne s'agit que d'un témoignage mais c'est un témoignage important car il est bien possible que se trouve là l'une des pistes possibles et souhaitables pour l'éducation de demain. Et il n'est guère étonnant de constater la convergence entre une conception dynamique, créative et collective de l'apprentissage et le logiciel libre et sa culture.

Culture communautaire : l'expérience Sugar Labs

Community culture: The experience of Sugar Labs

Walter Bender – décembre 2010 – Licence Creative Commons By-Sa (Traduction Framalang : Siltaar, Goofy, Seb seb, Zitor, Julien et Barbidule)

Dans un article publié il y a 30 ans et intitulé « Critique de l'ordinateur contre pensée technocentrique », Seymour Papert écrivait : « le contexte du développement de l'homme est toujours la culture, jamais une technologie isolée ». Dans un autre passage du même article, Papert offre un aperçu de ce qui est nécessaire pour fonder une culture de l'apprentissage : « Si vous vous demandez que doit savoir un

pratiquant averti du LOGO, la réponse va au-delà de la capacité à utiliser et enseigner le LOGO. L'adepte doit être capable de parler du LOGO, d'en faire la critique, et de discuter des critiques émises par d'autres personnes ».

30 ans après, remplaçons « LOGO » par « Sugar »

Sugar est une plateforme logicielle destinée à l'éducation des enfants. Sugar est développé et maintenu par Sugar Labs, une communauté mondiale de développeurs et d'éducateurs bénévoles. Notre objectif est l'émergence d'une génération de penseurs critiques et de gens capables d'inventer des solutions. À travers Sugar, nous nous efforçons de procurer à chaque enfant une chance d'apprendre et d'apprendre à apprendre, dans un contexte qui va lui permettre à la fois d'entamer un échange dynamique avec d'autres et de développer des moyens indépendants pour atteindre ses objectifs personnels.

Que devraient apprendre les enfants et comment devraient-ils apprendre ? Ceux qui apprennent devraient avoir accès aux idées qui nourrissent leur culture locale de même qu'aux idées puissantes qui constituent l'héritage global de l'humanité. Mais ils devraient aussi s'exercer à l'exploration et à la collaboration, et s'approprier des connaissances en menant une démarche authentiquement ouverte de recherche de solutions. Ce qui peut être réalisé au sein d'une communauté éducative construite autour d'une structure de responsabilités, c'est-à-dire avec des apprenants qui s'impliquent dans un processus d'expression, de critique et de réflexion par eux-mêmes. Qu'est-ce que j'apprends ? Comment l'ai-je appris ? Pourquoi est-ce important ? Puis-je l'enseigner à d'autres ? Est-ce que j'en ai une connaissance approfondie en l'enseignant ?

Dans cet essai, je compte exposer la façon dont Sugar nourrit une culture éducative par l'association de deux communautés – les développeurs de Sugar et ceux qui apprennent – participant à créer un « contexte favorable au développement humain » et un changement de culture scolaire.

La culture du logiciel libre

La culture du logiciel libre a influencé le développement de Sugar. Les développeurs du Libre vont au-delà du produit de consommation, ils créent et partagent leurs créations ; ils « débattent » du logiciel libre, ils en font la « critique », et ils « discutent le point de vue critique des autres ». Il ne prennent rien pour argent comptant. Les points communs entre le projet Sugar et le mouvement du logiciel libre sont les suivants : des outils pour s'exprimer, car les enfants créent des contenus autant qu'ils les consomment ; et la collaboration, car les enfants partagent leurs réalisations, s'aident mutuellement, et se lancent dans un processus de réflexion sur eux-mêmes et de critique collective.

Le projet Sugar s'inspire également de la façon dont les acteurs de la communauté du logiciel libre collaborent. Tout comme les développeurs de logiciels, les enfants discutent, se socialisent, jouent ensemble, partagent des médias, s'associent pour créer de nouveaux médias et des programmes, s'observent les uns les autres, dans un cadre à la fois formel et informel. Le projet Sugar facilite le partage, la collaboration et la critique. Les développeurs de logiciels libres et ceux qui apprennent avec Sugar rédigent des documents, échangent des livres et des images, créent de la musique ou écrivent du code ensemble. Les deux communautés s'investissent dans une « pratique de réflexion » : il s'agit de mettre en pratique leur expérience tout en étant guidé et épaulé par des « spécialistes » d'un domaine (ils peuvent être professeurs, parents, membres de la communauté dans un salon de discussion, ou encore de camarades étudiants investis dans un échange critique soutenu).

De la même façon qu'avec le logiciel libre, Sugar encourage chaque enfant à être une force créative au sein de sa communauté. L'apprentissage avec Sugar n'est pas un acte passif où l'enfant reçoit le savoir. Il est actif. On parle de créativité, d'aisance, d'innovation, et de résolution de

problèmes, tout ce qui implique l'expression personnelle et les liens forts à la communauté. Sugar apporte les outils d'expression à portée des enfants pour qu'ils soient libres d'agir à l'intérieur de leur communauté et à travers leurs actions, de changer le monde. Le logiciel libre est une condition nécessaire pour établir cette culture de l'expression et de l'émancipation. Le mot d'ordre de la génération suivante d'élèves sera « montre-moi le code, que je puisse en tirer un apprentissage et l'améliorer. »

Réalisations et défis

Depuis que nous avons établi les Sugar Labs en tant que projet dans le cadre du Software Freedom Conservancy (*NdT : lit. Protection des Libertés Logicielles*) en 2008, nous avons démontré notre engagement à un ensemble de valeurs fondamentales qui comprennent la liberté et l'ouverture ; nous sommes devenus dans une large mesure indépendants de tout matériel et distribution (lorsque nous avons commencé, nous étions liés à une seule plateforme – le netbook XO du projet One Laptop per Child (OLPC)) ; nous avons énormément avancé sur le chemin qui conduit à une version logicielle stable 1.0 ; nous sommes forts d'une vaste communauté qui comprend près de 2 millions d'élèves utilisateurs ainsi que, bien entendu, des développeurs de logiciels et de nombreux professeurs et étudiants qui ont leur franc-parler.

Alors que nous nous débattons quotidiennement avec des défis techniques, notre défi principal est l'un des engagements avec notre communauté : comment pouvons-nous nous assurer qu'il y a un dialogue fructueux entre le développeur et les communautés éducatives liées à Sugar ? En d'autres termes, comment pouvons-nous transmettre à la communauté éducative la culture de la collaboration et de l'esprit critique qui est essentielle au développement de la plateforme Sugar, et à mieux nous permettre d'apprendre de nos utilisateurs finaux ? L'un des rôles que joue la communauté Sugar est de sensibiliser l'ensemble de l'écosystème du logiciel libre aux

besoins des enseignants. Un autre rôle est de sensibiliser l'ensemble de l'écosystème éducatif au pouvoir de l'expression, de la critique et de l'auto-critique. Dans nos interactions avec les deux communautés, nous prenons grand soin de nous demander nous-mêmes : « Quel effet cela a-t-il sur l'apprentissage ? ».

Afin d'élargir nos efforts, un équilibre entre la fréquence des déploiements Sugar et la fréquence des nouveautés apportées par les Sugar Labs doit être maintenu. Nous avons un bon bilan dans notre réactivité aux besoins identifiés par les déploiements ; dans le même temps, nous sommes pro-actifs en sollicitant une plus grande participation de la communauté.

Les Sugar Labs sont aussi axés sur les besoins des enseignants. Nous avons des discussions régulières sur la façon de solliciter leurs retours. Certaines initiatives, tel qu'une liste de discussions fréquentée par des enseignants et des conversations hebdomadaires sur la pédagogie sont très productives. Un exemple de notre succès est que des enseignants commencent à apporter des modifications à Sugar et à ses activités. Un autre exemple est que des professeurs d'université enseignent l'informatique avec des logiciels libres dont Sugar.

Sugar Labs se décline au pluriel

Sugar Labs est une communauté globale qui se charge de définir des objectifs clairs et de maintenir l'infrastructure dont a besoin le projet dans son ensemble. Mais la communauté Sugar encourage et facilite également la création de « labs locaux » qui apportent leurs spécificités et une autonomie pour les déploiements régionaux, y compris en partenariat avec des entreprises locales à but lucratif, ce que le Sugar Labs « central » ne peut pas faire.

Ces labs locaux :

- adaptent la technologie et la pédagogie à la culture et

- aux ressources locales (ex : développement d'activités et de contenus spécifiques à une région) ;
- aident à traduire Sugar en langues régionales ;
 - gèrent les déploiements Sugar dans les écoles de la région ;
 - créent des communautés locales adhérentes aux principes des Sugar Labs, rendant Sugar plus ouvert et autonome ;
 - permettent la communication entre ces communautés locales et la communauté mondiale Sugar Labs ;
 - hébergent, co-hébergent ou s'associent dans l'organisation de conférences, ateliers, discussions et rencontres relatifs à l'utilisation et au développement de Sugar.

Avec le temps, la charge technique se répartit sur les labs locaux (la sortie récente de « Dextrose », pour les OLPC XO construits au Paraguay, est un exemple de comment les labs locaux – menés par une communauté de volontaires – peuvent travailler ensemble pour résoudre des défis techniques et pédagogiques).

En « amont » et en « aval »

Marco Presenti Gritti, développeur Sugar et co-fondateur des Sugar Labs, me rappelait que lorsque nous avons créé les Sugar Labs, nous avons pris une décision réfléchie sur l'étendue du développement. « En suivant le modèle de l'environnement graphique GNOME, nous n'allions pas tout créer et gérer nous-même, mais nous allions nous intégrer et nous appuyer sur les distributions GNU/Linux et le projet OLPC pour le faire ».

Classiquement, un projet en amont^[2] développe du code et un processus de publication. En aval, les distributions créent des paquets avec des personnalisations et distribuent un produit pour l'utilisateur final (cela implique habituellement un processus QA bien défini et un mécanisme de support).

Le spécificité éducative de notre projet a nécessité d'élargir

le modèle et les communautés impliquées. Le développement et les déploiements de Sugar sont évidemment engagés dans la construction d'images, de QA, des tests, dans la recherche d'erreurs à corriger, dans la documentation, le support... qui relèvent de programmeurs experts. Mais, comme mentionné précédemment, nous travaillons également avec des étudiants et lycéens et à l'occasion un professeur qui connaît suffisamment bien le Python peut contribuer aux correctifs.

Afin de créer un produit viable et gérable, nous devons établir un équilibre entre notre travail comme projet logiciel « en amont » et les efforts « en aval » des distributeurs GNU/Linux. C'est ainsi que nous travaillons activement avec la communauté Fedora (laquelle a pris à son compte une grosse partie de la charge associée au support du matériel OLPC), la communauté Debian, openSUSE, Trisquel, Mandriva, Ubuntu (ex : le Sugar Ubuntu remixé), etc.. À l'occasion nous devons assumer un rôle de leader, comme quand nous avons pris à bras-le-corps les initiatives naissantes pour créer un Live USB – « Sugar on a Stick ».

Optimisé pour la communauté

À la conférence LIBREPLANET en 2010, Eben Moglen a accordé un entretien sur tout ce qui avait été accompli par la communauté du logiciel libre. Le logiciel libre n'est plus une possibilité ; il est « indispensable », a-t-il affirmé. Ce logiciel « fiable et qui a un coût de production quasi nul » présente de nouvelles et nombreuses opportunités, en particulier dans le secteur de l'éducation, qui est toujours grevé par un budget serré. Seul le logiciel libre est « écrit une fois mais exécuté partout ».

Nous voulons aussi écrire du code fiable qui permette à Sugar d'être exécuté « partout », et nous avons réalisé de grands progrès en suivant les pas de la grande communauté GNU/Linux. Mais la communauté Sugar a un objectif supplémentaire : nous souhaitons que nos utilisateurs finaux participent également à

l'amélioration du code, parce que cela participe de l'apprentissage. Si tout le monde est capable d'écrire du code et si ce code est écrit avec les modifications des utilisateurs finaux en tête, nous aurons un monde dans lequel chacun est engagé dans le « débogage », ce que Cynthia Solomon a décrit une fois comme « l'une des grandes opportunités éducatives du XXIe siècle ».

Oui la licence GPL (General Public License) utilisée par les Sugar Labs garantit que le logiciel peut être modifié par l'utilisateur final. Mais, pour la plupart des utilisateurs, ceci n'est qu'une liberté théorique si la complexité du logiciel représente une barrière insurmontable. Par conséquent, les critères habituels (fiabilité, efficacité, maintenance, etc.) sont nécessaires mais non suffisants pour l'éducation.

Aux Sugar Labs, nous faisons un pas supplémentaire en nous assurant que notre code est à la fois libre et ouvert, mais également « ouvert à la manipulation des utilisateurs finaux ».

Voici quelques actions entreprises par Sugar Labs pour encourager et faciliter les modifications des utilisateurs finaux :

- Susciter des attentes et des envies en établissant une culture dans laquelle c'est la norme d'utiliser les libertés permises par le logiciel libre et articuler la liberté pour modifier les aspects du logiciel libre (1ère liberté).
- Offrir des outils qui facilitent l'accès aux sources (ex : un menu « voir les sources » toujours disponible, rendant la source de chaque application à portée d'un « clic de souris »).
- Utiliser des langages de script (Python, Javascript, et SmallTalk dans le cas de Sugar) pour que ces changements

puissent être immédiats et faits directement.

- Mettre en place des paliers pour permettre à l'utilisateur final de commencer en faisant des petits pas (alors que le langage de programmation C peut avoir une « couche haute », il n'a pas de très « basse couche »).
- Réduire le risque associé aux erreurs en proposant des « zones tampons » ; si en touchant au code vous introduisez des bugs collatéraux ou irréversibles alors les gens seront vite conditionnés à ne pas se livrer à des comportements à « risque » en modifiant le code.
- Fournir de « vrais » outils : s'assurez-vous que la vraie version puisse être modifiée et non une version répliquée indépendante mais peu motivante.
- Être une communauté de soutien ; on peut dire à juste titre de la communauté Sugar qu'elle est accueillante et tolérante avec les « nouveaux venus », poser une question c'est déjà devenir membre de la communauté, nous sommes pointilleux pour ce qui concerne l'octroi de privilèges sur le « projet principal » mais nous donnons les droits pour encourager la création de branches expérimentales.

Quand on m'a demandé combien de correctifs ont été fournis par les utilisateurs de Sugar, j'ai répondu que des membres de la communauté ont contribué aux correctifs mais que je n'avais pas connaissance de correctifs apportés par des enfants. Encore faut-il faire la distinction entre correctifs envoyés et acceptés, car l'apprentissage commence en créant le correctif, en le soumettant, et en le partageant avec d'autres même lorsqu'il ne se retrouve pas accepté. Sugar a inculqué aux enfants et à leurs professeurs le sentiment qu'ils peuvent être créatifs et utiles avec l'informatique.

Cependant, après deux années d'expérience concrète de Sugar, nous commençons à voir des contributeurs émerger de sa communauté d'utilisateurs. Par exemple, en Uruguay, qui a été

le premier pays à fournir des outils éducatifs libres à chaque enfant, quelques préadolescents sont en train de coder activement (un enfant de 12 ans d'une petite ville à des heures de Montevideo fréquente notre canal IRC, y pose des questions et poste du code, à la mi-décembre 2010, il a déjà envoyé huit activités sur notre portail). Quand le président uruguayen José Mujica a entendu parler de ces réalisations, il a souri et a dit avec une voix remplie de fierté : « Nous avons des hackers ». Il y a peut-être 12 enfants qui développent du logiciel libre aujourd'hui en Uruguay. L'an prochain ils seront 100. Dans 2 ans, ils seront 1000. L'Uruguay est en train d'expérimenter un changement de culture lié à un changement dans les attentes que le pays a pour ses enfants, un changement accéléré par la culture du logiciel libre.

Maximiser nos efforts

Qu'est-ce qui motive nos contributeurs et qu'est-ce qui motive les professeurs (que nous aimerions voir adopter Sugar) ?

Pour tenter d'y répondre je me suis appuyé sur l'article *L'économie comportementale : les sept principes des décideurs* publié par le *New Economics Foundation* :

- **Le comportement des autres personnes compte.** Nous devons sensibiliser les professeurs aux meilleures pratiques de Sugar pour qu'ils puissent faire des émules. Pouvons-nous identifier les « génies », « contacts », « commerciaux » dans nos communautés cibles ? Quelles ressources pouvons-nous mettre en place pour les inciter à adopter Sugar ? Ainsi je travaille avec une petite école de quartier dans la ville de Boston dont l'exemple est suivi par d'autres quartiers bien plus importants. Si nous pouvons avoir une influence sur un professeur « génie » du quartier, nous pourrions avoir un gros avantage. Cela signifie également que nous devons être vigilants quant à la qualité pédagogiques de nos

activités proposées.

- **Les habitudes sont importantes.** Ces habitudes qui participent au status quo ne doivent pas être négligées. Qu'est-ce qui motive et encourage le changement ? Quelles actions pouvons-nous mener pour soutenir et engager les changements dans les pratiques et les comportements ?
- **Les gens sont motivés pour « faire ce qu'il faut ».** Mettons alors cette notion de « faire ce qu'il faut » (*NdT : do the right thing*) en débat avec les enseignants, essayons de voir avec eux si leurs conceptions peuvent évoluer. En géométrie, il n'y a pas de chemin réservé aux rois, disait Euclide.
- **Les attentes des gens influencent leur comportement : ils veulent que leurs actions soient en phase avec leurs valeurs et leurs engagements.** C'est un travail de longue haleine pour nous car nous ne sommes pas toujours en phase au départ avec ces attentes. Cependant, tant que nous respectons et sommes fidèles à nos valeurs, nous pouvons convaincre et avoir de l'influence.
- **Les gens sont réticents au changement de peur de perdre ce qu'ils possèdent.** Utiliser Sugar à partir d'un clé USB (« Sugar on a Stick », qui emprunte seulement un ordinateur sans rien modifier dedans) n'implique aucune changement irréversible tout en permettant de faire une nouvelle expérience pédagogique.
- **Les gens hésitent souvent lorsqu'il s'agit de prendre de grandes décisions.** Ils sont souvent intimidés par les perspectives d'apprentissage de nouvelles choses (jusqu'à vraiment les faire). De plus les pertes immédiates peuvent décourager et faire perdre de vue les récompenses à long terme. Nous devons accorder une grande importance à ce moment crucial du démarrage en accompagnant ceux qui acceptent de prendre un tel

risque.

- **Les gens ont besoin de se sentir écoutés et impliqués pour s'engager dans le changement.** Nous avons une communauté qui tente d'accorder le plus grand soin à l'accueil des participants et à l'examen de leurs contributions. Ceci est une de nos grandes forces.

Est-ce que cela fonctionne ?

L'évaluation de projets éducatifs a toujours été difficile, en partie parce qu'il est difficile d'arriver à un consensus sur les mesures d'évaluation.

Il semble plus facile de prendre le problème par la négative où le consensus sur ce qu'il ne faut pas faire est plus facile à trouver. Ainsi Michael Trucano, qui blogue sur le portail éducation de la Banque mondiale, a publié un « top 10 » des pires pratiques de l'utilisation des nouvelles technologies dans l'éducation. Liste que je prends ici comme référence négative pour le projet Sugar avec comme exemples probants et prometteurs les deux déploiements d'envergure que sont le Paraguay Educa et le Plan Ceibal en Uruguay.

1. Parachuter du matériel dans les écoles et espérer qu'un miracle se produise.

C'est une critique souvent entendue pour le projet One Laptop per Child (un ordinateur portable par enfant), mais dans les faits, il y avait d'importants mécanismes d'aide et de mise en place en Uruguay et au Paraguay avant même que le matériel ne soit livré. En Uruguay, en plus du vaste support proposé directement par le gouvernement (incluant un programme de formation des professeurs, un centre d'appel, une vidéothèque des bonnes pratiques, etc.), deux initiatives communautaires au niveau national ont vu le jour : Ceibal Jam, qui fournit des logiciels et du contenu local aux enfants d'Uruguay, et Red de Apoyo al Plan Ceibal (RAP-Ceibal), qui assure un réseau d'aide pour les professeurs. Paraguay Educa a une équipe de

conseillers qui travaille à temps plein dans les écoles, en aidant les professeurs. Et les éducateurs des deux pays participent régulièrement à des forums mondiaux.

2. Concevoir via l'OCDE des environnements d'apprentissage à implémenter partout.

Les « pays développés » proposent du contenu et quelques règles de bonnes pratiques, mais ce sont avant tout les équipes pédagogiques locales en Uruguay et au Paraguay qui échangent et conçoivent leurs propres matériels et programmes pour répondre à leurs besoins locaux (par exemple, un professeur de la campagne péruvienne a écrit un livre sur l'utilisation de Sugar en salle de classe qui est internationalement lu et reconnu par les autres professeurs).

3. Penser les contenus éducatifs après la mise en place du matériel.

En Uruguay et au Paraguay, c'est la pédagogie qui a guidé la vitesse de déploiement d'un projet vu avant tout comme une plateforme d'apprentissage (incluant les ordinateurs portables, la connectivité, les serveurs, la formation, la documentation, le support, l'assistance de la communauté, etc.).

4. Supposer que vous pouvez uniquement importer du contenu venu d'ailleurs.

Le mot clé ici est « uniquement ». L'Uruguay et le Paraguay profitent bien entendu des contenus créés ailleurs (comme par exemple ceux de la communauté Etoys) mais ils n'oublient de favoriser la production de ressources locales, qu'il s'agisse de nouveaux contenus ou de contenus modifiés à partir de ceux récupérés ailleurs.

5. Ne pas surveiller, ne pas évaluer.

À Plan Ceibal, ils ont un fonctionnement étendu pour surveiller l'état du réseau, des serveurs, et des ordinateurs

portables lors du déploiement. Il y a beaucoup d'évaluations en cours du programme, aussi bien internes qu'externes. Paraguay Educa a été l'objet d'une évaluation externe par la Banque Interaméricaine de Développement (IDB Inter-American Development Bank).

6. Faire un gros pari sur une technologie qui n'a pas fait ses preuves.

C'est en particulier le cas lorsque l'on se base sur un unique distributeur et sur des standards fermés et/ou propriétaires. C'est alors une épée de Damoclès qui pèse sur l'avenir du projet. Les deux programmes mentionnés ci-dessous ont fait l'objet d'appels d'offre public et ont plusieurs distributeurs. Les deux utilisent abondamment des logiciels libres.

7. Ne pas être transparent sur le coût global de l'opération.

L'Uruguay a été assidue en publiant les chiffres de leur coût total de possession, maintenance et services du projet (chiffres, basés sur les coûts mesurés sur le terrain, qui se sont avérés plus bas que ce que certains avis pessimistes avaient prévu).

8. Négliger les problèmes d'équité.

En Uruguay ce sont avant tout les familles modestes qui sont ainsi équipées en informatique avec un accès Internet gratuit.

9. Ne pas former vos professeurs (ni votre directeur d'école).

Le plus gros investissement dans le programme au Paraguay a été la formation des professeurs. C'est sûrement la principale clé de la réussite du projet et nous veillons à ce que cette formation soit toujours plus efficace et adaptée aux réalités du terrain.

Trucano laisse le point numéro 10 comme exercice ouvert pour

le lecteur. J'ajouterais :

10. Ne pas impliquer la communauté.

Dans les deux communautés uruguayenne et paraguayenne l'implication fait partie du projet par nature. Pour ce qui concerne Sugar, c'est un effort d'une communauté globale qui implique des centaines d'ingénieurs et des milliers de professeurs. Un résultat remarquable est le degré d'implication des parents dans les programmes.

Regarder vers le futur

Comme il est de mise avec chaque projet piloté par une communauté, il y a un débat permanent sur la vision de Sugar. Il peut y avoir des divergences d'opinion sur l'étendue de la mission des Sugar Labs (allant d'un point d'attention particulier sur les outils de collaboration à une vision plus large sur tout ce qui est nécessaire pour des déploiements réussis de l'OLPC). Mais tout le monde s'accord à dire qu'il y a une communauté Sugar de développeurs et d'apprenants pleine de vie et d'énergie et que les plateformes d'apprentissage basées sur des logiciels libres encouragent l'appropriation du savoir quel que soit le domaine que l'apprenant explore : musique, navigation sur internet, lecture, écriture, programmation, dessins, etc.

Carla Gomez Monroy, une pédagogue qui a participé à de nos nombreux déploiements, décrit Sugar comme « un environnement émergent et collaboratif, où la communauté identifie, code, utilise, innove, conçoit et re-conçoit ses propres outils » Les membres de la communauté d'apprentissage de Sugar s'engagent dans le *débogage* de leur créativité et des outils mis en place pour exprimer cette créativité. Ils investissent Sugar en tant que technologie mais aussi et surtout comme une culture de l'apprentissage passant par l'expression et la critique collective.

L'expérience Sugar Labs est « une participation collaborative

pour *apprendre à apprendre* avec des outils qui nous correspondent ».

Walter Bender est le fondateur et le directeur exécutif de Sugar Labs, une fondation à but non lucratif. En 2006, Bender a co-fondé « One Laptop per Child », une organisation à but non lucratif avec Nicholas Negroponte et Seymour Papert.

Notes

[1] Crédit photo : Danishkanavin (Creative Commons By-Sa)

[2] Dans le développement logiciel, la métaphore de la rivière est utilisée pour décrire où les différentes activités et responsabilités se situent dans l'écosystème. L'« Amont » fait référence aux auteurs et mainteneurs du logiciel. L'« Aval » fait référence aux distributeurs et aux utilisateurs du logiciel.