

Atelier scientifique en quatrième : informatique

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 5 Janvier 2010. Dernière mise à jour le 14 Juillet 2010

<http://www.bortzmeyer.org/atelier-scientifique.html>

J'ai participé, comme « tuteur » à un « atelier scientifique » dans une classe de quatrième au Collège Modigliani (<http://clg-modigliani.scola.ac-paris.fr/>). Le thème est l'informatique et le sujet « Faire des images de synthèse avec un langage ».

Le but de cet atelier est de faire réaliser un travail par les élèves de quatrième. Quelques indications sont données mais ils doivent, avec l'aide et la supervision du tuteur, chercher par eux-même des informations, accomplir des « manips », et réaliser un exposé. Je ne sais pas pour eux mais, moi, j'étais content de retourner à l'école.

J'avais proposé deux sujets en informatique, « Faire des images de synthèse avec un langage » (qui a été choisi par les élèves) et « Où se trouve ce site Web dans le monde physique? ». À noter qu'il y avait d'autres sujets sur d'autres sciences, et que l'informatique, que je pensais à la mode, n'a pas enthousiasmé beaucoup d'élèves, peut-être parce que l'équipe avait bien insisté sur le fait que ce serait moins ludique que de jouer à GTA. Pour chaque sujet, il fallait s'assurer qu'il existe suffisamment de documentations en ligne et que les logiciels soient libres (ou, à la rigueur, gratuits), tournant sur Unix et Windows, et installables sur tout ordinateur (tous les élèves n'ont pas une machine de course). Voici les textes d'accroche qu'avaient eu les élèves :

- « Faire des images de synthèse avec un langage » : Certains logiciels permettent d'accomplir des actions en indiquant avec la souris un objet. Cette technique ne permet que des actions prévues à l'avance. Au contraire, un langage permet de décrire des actions entièrement nouvelles. Les langages sont donc plus riches que le simple fait de « montrer du doigt ». Si apprendre un vrai langage de programmation est un objectif trop ambitieux pour cet atelier, en revanche, il est possible d'apprendre un langage de description de scènes comme POV-Ray, qui permet d'indiquer un ensemble d'objets graphiques, dont l'image sera ensuite produite par le logiciel. Cette technique de description de scènes est surtout utile pour faire produire des images automatiquement, par un programme. POV-Ray est un des programmes de production d'images de synthèse les plus populaires, maintenu par une communauté active et enthousiaste. Je n'ai pas d'expérience professionnelle dans ce domaine (contrairement au second sujet).

- « Où se trouve ce site Web dans le monde physique ? » : Sur l'Internet, tout semble virtuel. Sur son écran, on regarde un site Web situé en Chine ou au Brésil comme s'il était à quelques mètres, et avec des temps de réponse quasiment identiques. Mais, en fait, les serveurs Web ne sont pas virtuels du tout. Ce sont de grosses machines, consommant beaucoup d'électricité et situées à un endroit bien précis. La nature ouverte de l'Internet fait qu'on peut facilement savoir où est cet endroit. Il existe des outils logiciels qui permettent d'afficher la liste des sites physiques traversés pour atteindre une machine donnée. Utiliser ces outils permet de mieux comprendre comment fonctionne l'Internet, et qui le fait marcher. (Je sais que les lecteurs de ce blog savent comment faire mais c'est pour des enfants de treize ans.)

Pour « Faire des images de synthèse avec un langage », le but pédagogique est une introduction aux langages informatiques qui soit plus simple que les langages de programmation mais plus gratifiante que le HTML. POV fait aussi faire de la géométrie, puisqu'il faut comprendre axe, coordonnées, etc. Voir un témoignage d'enseignant de troisième (<http://www.framasoft.net/article2253.html#comments16448>) qui utilise ce logiciel.

Naturellement, le tuteur a préparé des ressources, des documentations (en français : c'était une des difficultés du cahier des charges), etc. Mais je ne les ai pas indiquées tout de suite ici, les élèves savent utiliser un moteur de recherche et pourraient trouver cette page :-)

La première scène utilisée en exemple était :

```
#include "colors.inc"
#include "shapes.inc"
#include "textures.inc"

camera {
    location <-6, 1, -3>
    look_at <2, 1, 0.5>
}

sphere {
    <0, 1, 2>, 2
    texture {
        pigment {color Yellow}
        finish {phong 1}
    }
}

light_source {<-3, -4, 3> color White}
```

Et le résultat est : Évidemment, la première image sérieuse, comme toujours en images de synthèse, aurait dû être une théière mais, à ce stade, nous sommes restés modestes. (Merci à Jean-Philippe Pick pour son érudition sur les théières.)

Quelques autres informations sur POV-Ray : il n'est pas complètement libre mais presque. On le trouve sur Ubuntu mais, pour Debian, il faut prendre la section "non-free". Sinon, il est téléchargeable en <http://www.povray.org/download/>.

Le but de l'atelier est de décrire les scènes POV-Ray à la main. Mais, sinon, il existe des modelleurs sur Unix ("*kpovmodeler - a graphical editor for povray scenes*") ou bien Moray, un « "shareware" shareware ». Inkscape peut produire du POV.

Il existe beaucoup de documentations sur POV en français, pour tous les niveaux :

<http://www.bortzmeyer.org/atelier-scientifique.html>

- <http://fr.wikipedia.org/wiki/POV-Ray> Bonne synthèse
- <http://users.skynet.be/bs936509/povfr/> Plein de documentations pour POV-Ray en français
- http://www.oceis.com/reprise_index.html?lien=povray.htm La documentation officielle, traduite
- <http://povray.free.fr/> Didacticiel en français
- http://www.f-lohmueller.de/pov_tut/pov__fra.htm Exemples en français

Pour « Où se trouve ce site Web dans le monde physique ? », qui n'a pas été retenu par les élèves, le but pédagogique était d'avoir une attitude moins « consommation » vis-à-vis d'Internet, comprendre son insertion dans le monde physique, faire de la géographie (codes aéroport, qui servent à nommer les routeurs, pays bien connectés et les autres), comprendre les enjeux économiques et financiers derrière (accords de « "peering" », géographie de la fracture numérique).

Pour revenir à POV-Ray, j'avais envisagé de faire réaliser aux élèves, à la fin, un « chef d'œuvre » sous forme d'une scène POV-ray utilisant des techniques proches de la programmation (comme des paramètres, ou des objets réutilisables). Voici un exemple d'un source POV-Ray (en ligne sur <http://www.bortzmeyer.org/files/table.pov>) pour une telle scène et son rendu (en ligne sur <http://www.bortzmeyer.org/files/table.png>). Beaucoup trop ambitieux, le projet a été abandonné.

Quelques leçons que j'ai tiré de cette expérience :

- Nous avons appris des choses comme le fait que POV a un fond de lumière ambiant, découvert en oubliant accidentellement `light_source`. Vive la sérendipité!
- L'une des premières images réellement créées était une répétition de sphères légèrement décalées : toutes avaient été faites à la main, l'élève ne sachant pas programmer.
- On apprenait aussi des choses en physique (réflexion, transparence) que les élèves ne voient normalement qu'à la fin de l'année.
- Pour l'apprentissage de la programmation (même si ce n'était pas le sujet de l'atelier), POV est très prometteur mais nous n'avons pas eu le temps d'exploiter ces promesses.

Mais il y a eu des problèmes plutôt gênants :

- Le travail en équipe, qui n'est jamais une chose facile. Une tentative (sans suite) a été faite d'utiliser un système de forum en ligne pour la communication. Les élèves utilisaient surtout MSN entre eux et le courrier avec moi. Globalement, je trouve qu'il y a eu peu de communication, par exemple les problèmes ou difficultés étaient toujours signalés trop tard. Il était difficile d'obtenir des informations d'étape. À noter que ce n'est pas spécifique au collège, le monde professionnel offre également de nombreux exemples de telles difficultés.
- La tendance à préférer des documents copiés/collés plutôt qu'à en créer. Cela était vrai aussi bien pour les textes (une recherche « phrase » - en mettant des guillemets - dans Google permettait de retrouver les originaux) mais c'était surtout gênant pour les images. Un exemple typique était la tendance des élèves à préférer, pour les compte-rendus et exposés, l'utilisation d'images récupérées sur le réseau plutôt que d'images faites par eux-mêmes avec POV-ray (alors que ce groupe avait justement l'intérêt de reposer sur de vrais travaux pratiques).
- Comme souvent avec les travaux pratiques d'informatique, les problèmes pratiques sont nombreux. On est loin de la soi-disant informatique conviviale et intuitive! Avec Hotmail, service utilisé par la plupart des élèves, ce fut une catastrophe, notamment pour les images jointes, dont on avait absolument besoin, mais qui sont souvent avalées, ou entraînent le blocage complet du message. Même chose avec Windows (système utilisé par tous les élèves) qui plante quand il ne faut pas (et les élèves ont du mal à donner une description détaillée et factuelle du problème, problème qui existe aussi dans le monde professionnel). Les PC/Windows du collège, qui n'ont pas tous le même comportement, ont également été une grande source de frustration.
- Contrairement à ce qu'on lit souvent, les élèves ne sont pas forcément très à l'aise avec l'informatique, notamment lorsque les choses vont mal (cf. l'excellent article La génération « post-micro » (<http://www.hyperbate.com/dernier/?p=8582>)). Consommer est une chose, créer (ce que font par exemple les auteurs de jeux vidéo, les cinéastes ou les programmeurs) est bien plus complexe. Même chose pour la différence entre bavarder (lorsque le fait de communiquer compte davantage que le contenu de la communication) et échanger dans le cadre d'un travail d'équipe, avec des résultats à atteindre.

Enfin, le bilan est mitigé : le concept n'a pas été reconduit sous cette forme, le bilan officiel note que les élèves de quatrième n'ont pas forcément la maturité nécessaire pour un travail en autonomie (de mon expérience professionnelle, je peux dire qu'un certain nombre d'adultes sont dans le même cas). L'année suivante, les cours prendront une forme plus classique. Dommage, mais ça pourra recommencer autrement, peut-être en troisième.