

---

# Les objets ont gagné - Nouveaux enjeux

**Christian Queinnec**

*Laboratoire d'Informatique de Paris 6*  
*Université Paris 6 - CNRS*  
*8 rue du Capitaine Scott, 75015 Paris*  
Christian.Queinnec@lip6.fr

---

*RÉSUMÉ.*

*ABSTRACT.*

*MOTS-CLÉS :*

*KEYWORDS:*

---

## 2 L'objet – 10/2003.

La question ne se pose absolument plus : les objets ont gagné... leur place dans les cerveaux des développeurs et dans les cœurs de leurs hiérarchies. Les objets se sont tellement imposés que l'on assiste même au retour de la pensée unique que l'on avait pu croire abandonnée depuis PL/1. « Tout est objet » n'est plus un élément (toujours discutable) du dogme mais une réalité où plus rien ne se conçoit en dehors de C++, J2EE ou .NET.

Le temps n'est plus où les objets ne représentaient qu'un style de programmation parmi tant d'autres que l'on choisissait pour ses qualités et son adaptation au problème ; sont désormais arrivés les jours marquant leur domination sans partage. Les méthodes mêmes de conception/développement ignorent ce qui n'est pas objet.

La seule raison de cet état de fait est l'efficacité industrielle de l'objet. Son nom d'abord ne rebute pas au contraire de « fonction » ; l'image d'autant plus concrète qu'il projette d'un petit animalcule obéissant fidèlement aux messages ; le plaisir qu'il procure au démiurge-programmeur de régner sur ces myriades. Tous ces aspects ont contribué à sa pandémie. Sans oublier les promesses de la réutilisation qui ont permis de financer en une décennie une élévation sensible du niveau moyen des programmeurs mais qui, en retour, favorise l'établissement durable des objets afin de rentabiliser cet investissement.

Cette spectaculaire cristallisation a coïncidé avec l'éclosion d'un grand nombre d'outils graphiques (et l'on sait le goût que les objets avaient pour eux) facilitant de nombreuses tâches informatiques et, concomitamment, avec l'avènement d'ordinateurs peu chers les propageant. Les conditions furent ainsi réunies pour la création et le succès reproductif du « même » objet. Assistons-nous donc à la « fin de l'histoire ? »

La pensée unique correspond à une standardisation qui n'est optimale nulle part mais employable partout, elle met à disposition des objets uniformes qu'une sélection naturelle rendra commodes puis adaptés. Elle poussera à l'abandon d'anciennes pratiques ou catégories : les bases de données gèreront des objets et se dilueront dans les systèmes d'exploitation qui eux-mêmes ont déjà avalé les couches d'interaction graphiques. Cette industrialisation en marche a également contribué à la reconnaissance de nouveaux besoins : la mobilité, la sécurité.

L'apprentissage des langages à objets n'est pas des plus simples. Le classique « SalutMonde » (et son train de mots clés, dans l'ordre : `public class public static void main String[]`) qui ne peut être que d'une complexité rebutante si expliqué mot à mot ou l'occasion d'une kyrielle de mensonges par omission, n'est pas la meilleure introduction. Par contre, la disponibilité d'outils spectaculaires (graphiques ou réticulaires) accroît vite l'intérêt des étudiants. La greffe de fragments de code (et non plus de programmes entiers) par le biais des environnements de développement et de conventions d'architectures facilitant le déploiement, permet l'obtention rapide d'effets remarquables.

Certes la complexité est masquée mais pour les paramétreurs de progiciels (un métier en expansion), ainsi que pour les composeurs (les programmeurs qui assemblent des composants), objet et itération suffisent pour presque toutes les tâches (même ar-

penter des arbres). Ce qui reste difficile est du ressort des experts et un petit nombre d'experts suffit.

Que les hordes barbares envahissent notre univers de lettrés est donc un progrès puisque plus peuvent programmer et ainsi contribuer à enrichir notre univers même si leur culture n'est pas à la hauteur du raffinement que nous a inculqué Jean-François Perrot. Tout comme il est préférable que tous sachent obtenir une racine carrée (avec une calculette) plutôt que de laisser une seule poignée savoir la calculer (avec un crayon et du papier), il est plus satisfaisant de laisser s'empoussiérer nos disputes et de nous tourner vers les deux problèmes majeurs d'enseignement suivants :

- Comment enseigner l'objet *ab initio* aux masses ?
- Comment enseigner l'« informatique » aux futurs experts ?

Et ainsi les hordes pourront-elles fonder une nouvelle civilisation.