

Les futurs « grands détournements » ?

ATOMiser ? PHler ? ARMer ? Comment GPU faire cela ?

Accélérateurs MIC & GPU : assez matures pour sortir du HPC ?

Processeurs légers : assez performants pour des serveurs ?

« Détourne-toi des préceptes de ceux qui spéculent sur le monde mais dont les raisons ne sont pas confirmées par l'expérience. »
de Léonard de Vinci

Entre science & technologie

Quelle pertinence aux JRES 2013 ?

- **Dans la presse ou ailleurs**

- **Accélérateurs : Nvidia, AMD, Intel**

- TOP 500 novembre 2013
 - 1er global (Intel Phi), 1er européen (Nvidia)
 - 10 % du total, 40 % des 10 premiers
- Presse : Rapprochement IBM Nvidia

- **Basse consommation :**

- 2013-09 : « Intel lance ses Atom C Series pour serveurs haute densité »
- 2013-10 : « Dell va montrer cette semaine son 1e serveur ARM »
- 2013-11 : « Avec sa puce serveur Berlin, AMD s'apprête à utiliser sa technologie HSA »

- **Plateaux techniques CBP**

- Architectures « exotiques »
- *Compute On My Own Device*

- **Quels objectifs maintenant ?**

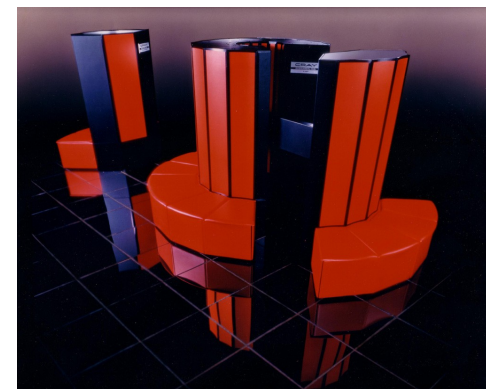
- Fiabiliser/pérenniser les codes
- Calculer/prototyper les codes

- **Quels objectifs futur ?**

- Accélérateurs incontournables ?
- Poste de travail sous COMOD ?

Un léger *flash back* Il y a une génération (humaine)...

- 1984 : ***The Last Starfighter***
 - 27 minutes d'images de synthèse
 - $22.5 \cdot 10^9$ par image
 - Utilisation Cray X-MP (130 kW)
 - 66 jours continus (en fait 1 an)
- 2013 : sur GTX Titan (250 W)
 - 202 secondes
- Comparaison Titan/Cray
 - En performance : 28000
 - En consommation : 10 000 000 !



VS



Un léger *flash back*

Il y a une génération (humaine) et après...

- 1987 : Acorn Archimedes
 - Premiers RISC « de masse »
 - 1 cycle/1 instruction
 - *Pipeline* à 5 étages
 - *Flop* commercial
 - Depuis, **ARM** vend du design
 - Architectures ARM : v7, v8 64 bits
 - Familles Cortex, SoC Exynos
 - Fondateurs 19/4/4 Asie/USA/Europe
 - GPU : Mali, PowerVR, Nvidia



- 2007 : génération *NetBook*
 - Atom : architecture Bonnel
 - Famille Z, E, D, N, S
 - Architecture x86, x86-64
 - Fondateur **Intel** !
- 2009 : fusion avec l'APU
 - APU : **AMD/ATI** CPU/GPU
 - Famille A, E2, Z
 - Fondateur *Global Foundries*
 - GPU : « vrai » Radeon HD !

Ces CPU « *pourquoi si peu gourmands ?* » « *Parce queuuu (on est KISS) !* »

- Consommation : *Thermal Design Power* (enveloppe thermique)
 - $TDP = \frac{1}{2} C V^2 f$: C = Capacitance, V = Tension alimentation, f = fréquence
 - Pour le physicien : un CPU = un radiateur (115 W sur 4 cm²)
 - $Capacitance = Finesse^2 \cdot Nb\ Transistors \cdot Constante\ de\ Mylq$ (~ 0.015)
- KISS : *Keep It Simple, Stupid !*
 - Simplification de l'architecture : *Reduced Instruction Set Computing*
 - Une opération/cycle, Pipeline 5 étages (IF, ID, EX, MEM, WB), registres uniques
 - Peu de transistors pour chaque cœur (facteur 5 entre ARM et i7)
 - Pas de mémoire cache de taille immodérée : i7 80 % des transistors pour L3
 - Pas de *Out-of-Order*

Le banc d'essai : le hard...

Architectures « légères »



Socles & CPUs



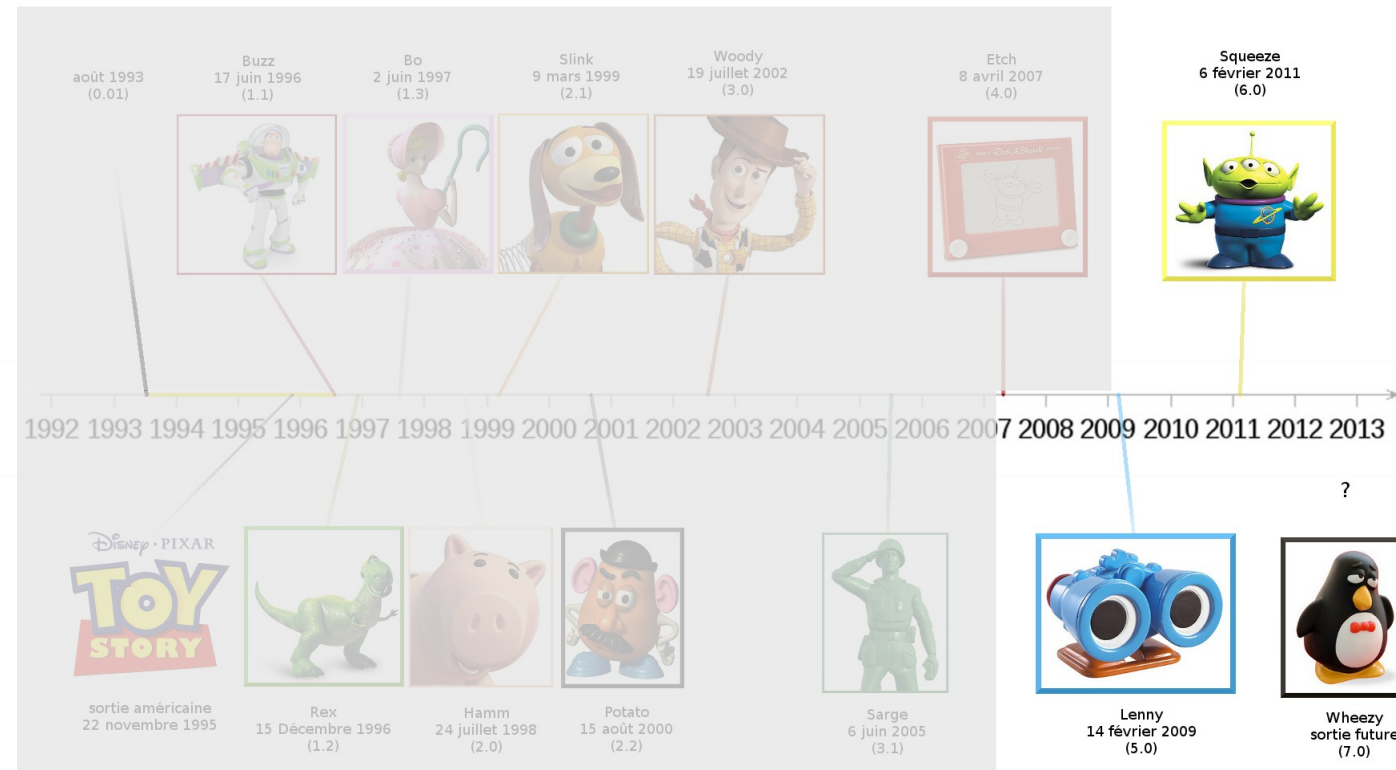
AMD



Nvidia



Le Banc d'Essai : le soft... Debian & CyanogenMod et SIDUS !



Banc d'essai : installation d'OS

Entre simple, bricolé, *hack* et impossible...

- **ARM** : entre simple & hack
 - CyanogenMod 10.1 + chroot Debian sur Galaxy S3 et Asus TF700
 - Debian Wheezy sur SDcard sur Odroid U2 et Odroid XU
- **Atom** : entre simple & impossible
 - *No problemo* sur Clevo M1110 & N550 (Sidus 32 ou 64)
 - *No problemo* sur Dell Mini1010 & Z530 (Sidus 32)
 - Impossible sur Acer Iconia W510 & Z2760
- **Fusion** : bricolé
 - BIOS très pénible : UEFI & Secure Boot
 - Matériel mal supporté : carte réseau inconnue & carte vidéo trop récente

Nbench (héritage de ByteMark)

Lorsque la renormalisation s'impose...

- **Analyse :**

- Avant renormalisation :

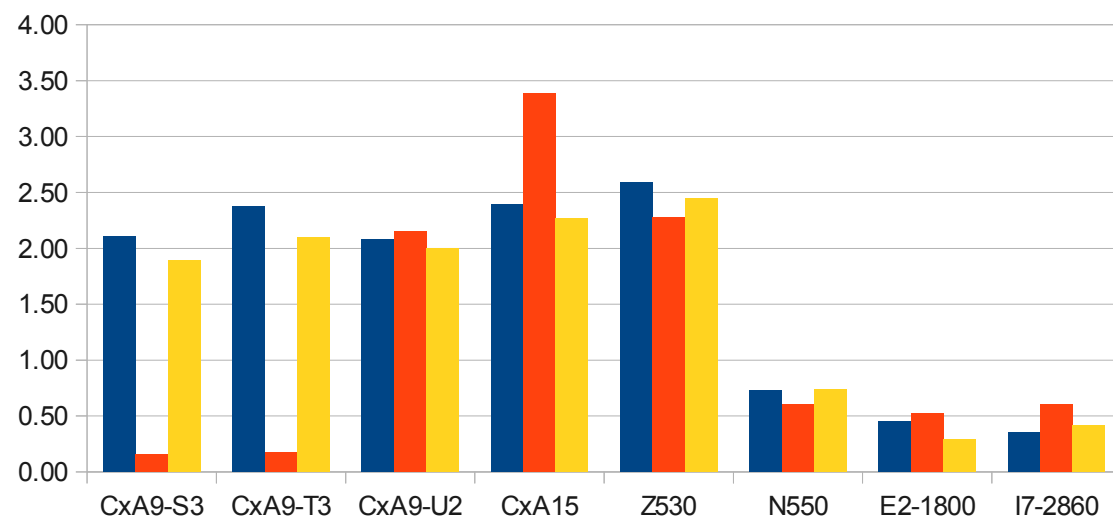
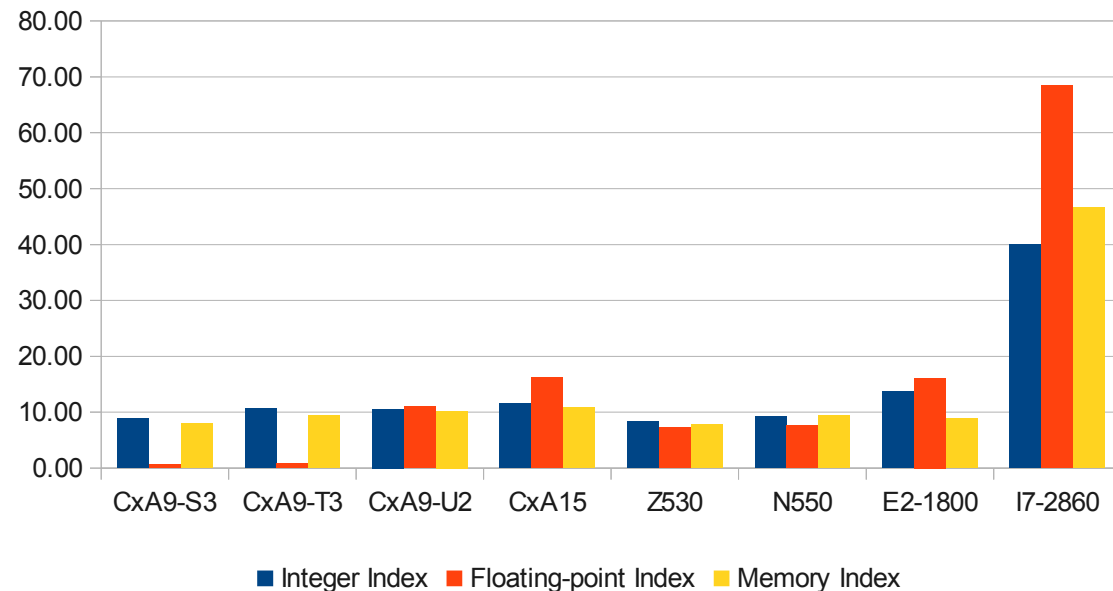
- Entier/Mémoire : « légers » comparables
 - Galaxy S3, TF700 : FP catastrophiques

- Après renormalisation : GHz & TDP

- Cortex A15 & Atom Z530 en tête, i7 bon dernier !

- **Conclusion :**

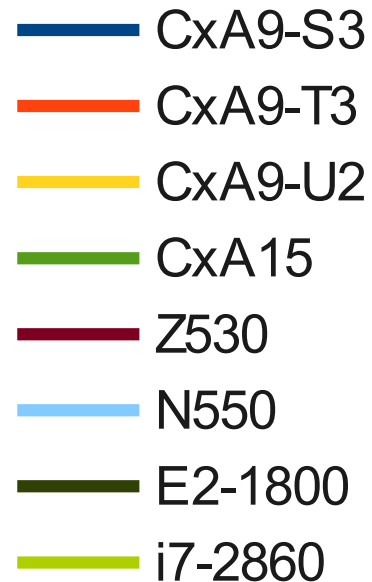
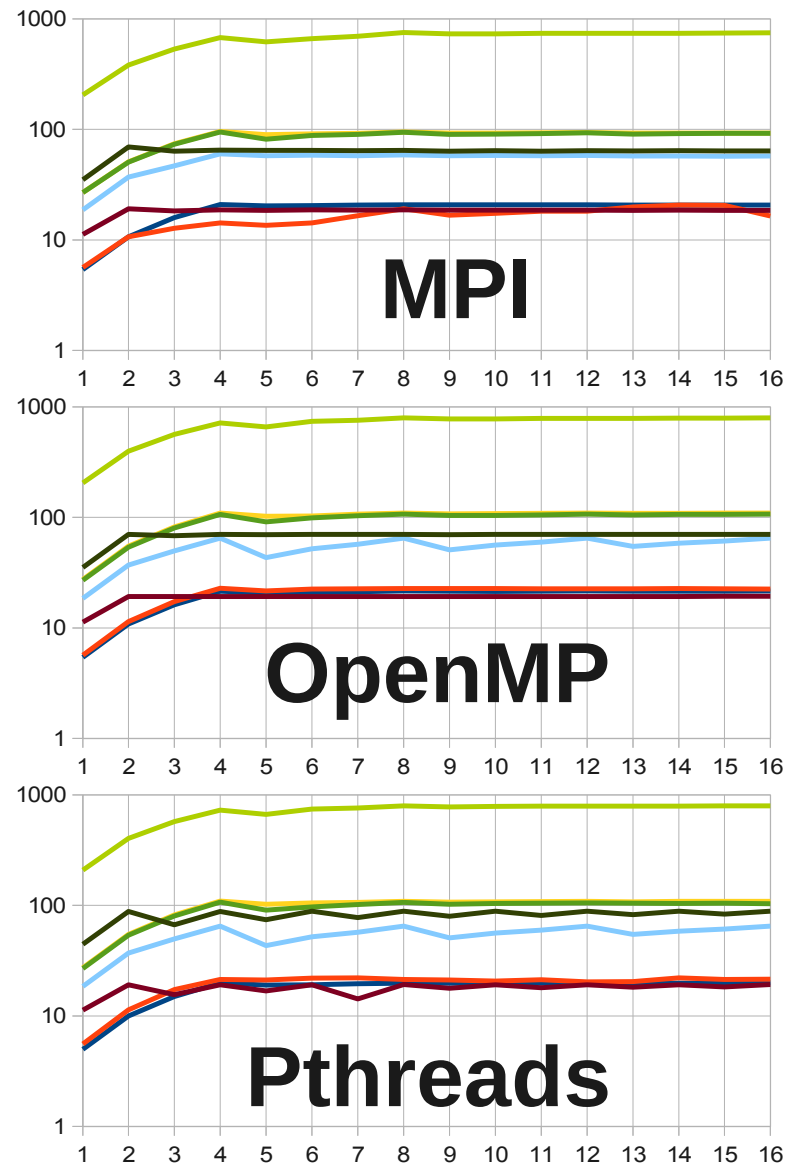
- Facteur 10 entre CPU & CPU léger
- Bizarrerie en virgule flottante sur SoC
- Cortex A15 bonne surprise !
- Efficacité énergétique ARM & Atom !



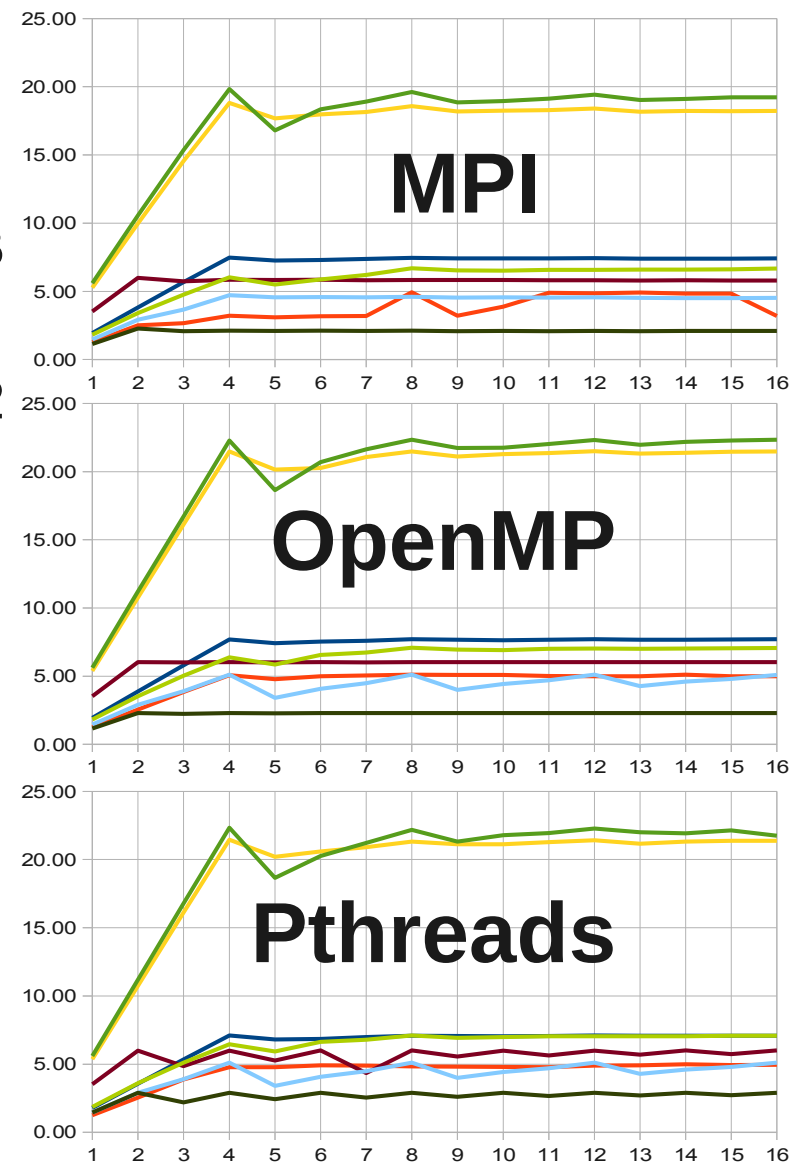
Pi Monte Carlo en parallélisme

De 1 à 16 processus sur 8 machines

En direct, en log !



Renormalisé !



Analyse :

- Avant renormalisation
- Comportement comparable
- CPU/CPU légers : x10
- *HyperThreading* efficace
- Après renormalisation
- N+1 fall
- Cortex A9 & A15 au dessus

Conclusion :

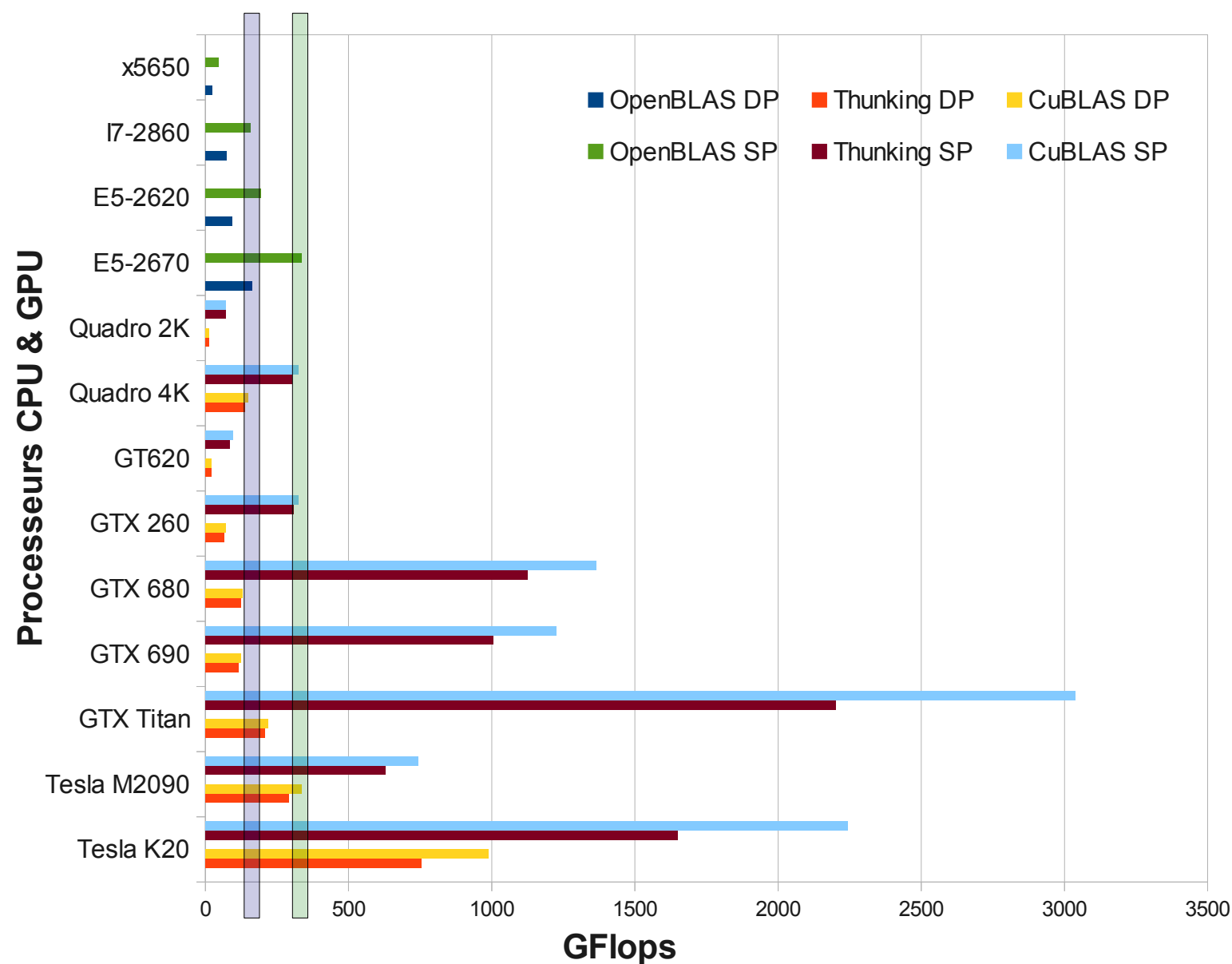
- InOrder+HT : pertinent !
- Scalabilité & Résistance
- Cortex A9 & A15 efficaces

La migration sur les « légers »

- Pour l'Atom & Fusion : compatibilité totale x86 et x86-64
 - 64 bits : Fusion toujours, Atom compliqué (Z530 non, N550 oui, Z2760 non) !
 - Mémoire : pas de cache L3, RAM < 32 GB
 - Performance ~ 10x moindre
- Pour l'ARM : un support maintenant mature
 - Distribution Android, Windows 8 et Debian (& Ubuntu)
 - Volumes des paquets (Debian) : 90 % du total x86 en ARM
- **Attention !**
 - Comportement « trop intelligent » : extinction des cœurs, fréquence variable, ...
 - Noyau généralement très spécifique (grosse colère de L. Torvalds)
 - Tout dans le SoC : performance des Entrées/Sorties

GPU : Le cas « qui marche » xGEMM

En SP, ça torche. En DP, il faut pousser...



Analyse :

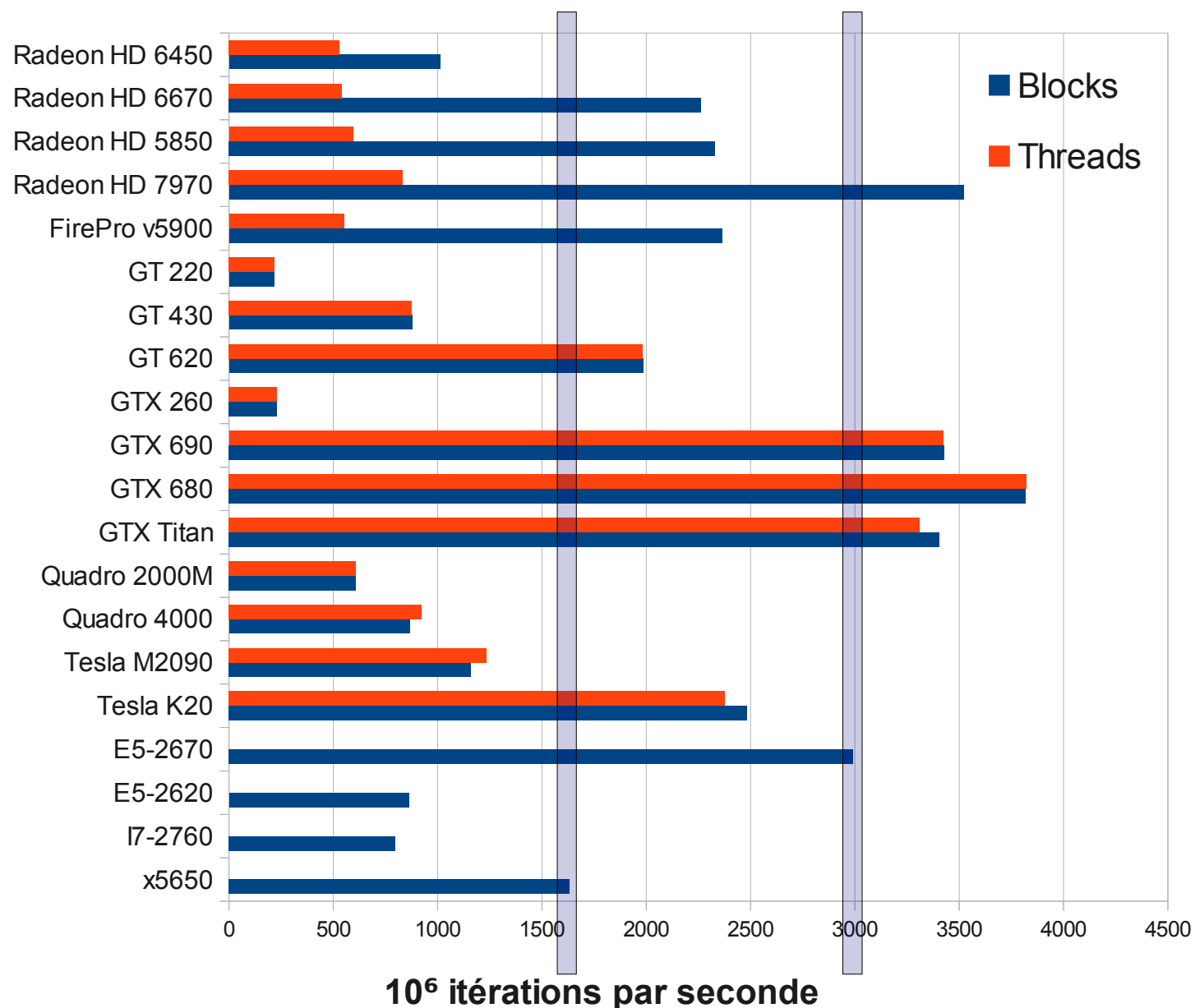
- CPU/Tesla : $\frac{1}{2}$ entre DP/SP
- GTX : $\frac{1}{10}$ entre DP/SP
- Tesla/GTX : $>$ en DP, $<$ en SP
- 30 % de perte avec Thunking
- Efficace pour taille > 1000

Conclusion :

- GPU efficace oui **MAIS !**
- Différence entre GTX & Tesla
- *Tuning* nécessaire

Pi Monte Carlo en OpenCL sur GPU/CPU

Résultats en demi-teinte ?



Analyse :

- GTX récents / HD7970 > CPU 16c
- GT/HD récents > CPU 12c
- Limitation *Threads* (256 ou 1024)
- GTX 680 la meilleure !

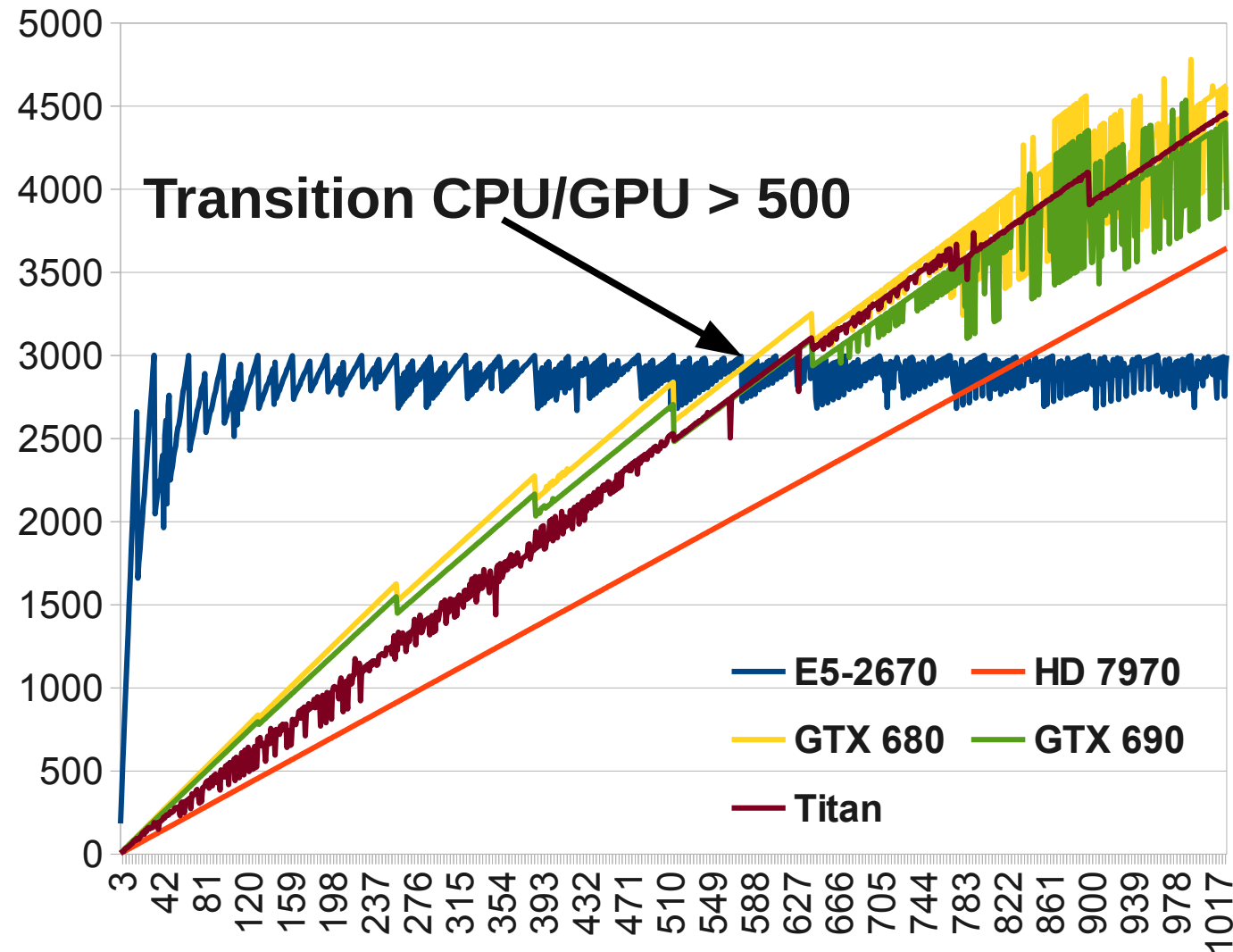
Conclusion :

- GPU/CPU : 30x plus lent !
- Threads/Blocks comparables
- Aller plus loin (en processus)...

Comparaison CPU/GPU

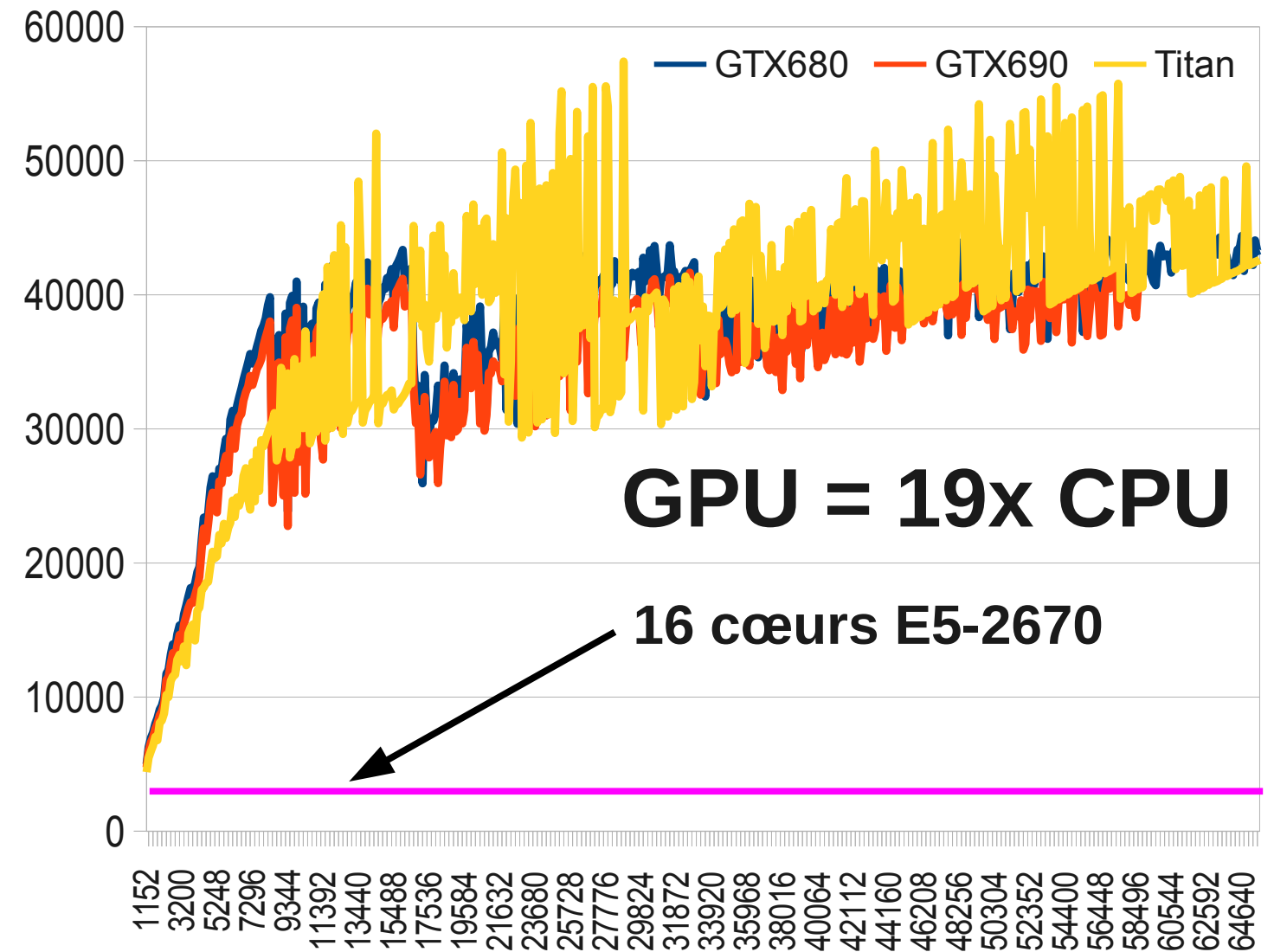
De 1 à 1024 processus

GPU > CPU ? GTX 680 le meilleur ?



De 1024 à 65536 processus

96 % pour 690, 86 % pour Titan



La migration sur GPU

- **La « bonne » : approche intégrateur**
 - Exploiter dès l'origine des API génériques : BLAS, Lapack, FFTw, ...
 - Convertir les appels vers des appels GPU (CuBLAS, Magma, CuFFT)
- **La « brute » : approche développeur**
 - Utiliser un langage « haut niveau » : Python Numpy/Scipy
 - Identifier les zones « qui coûtent » : profileur
 - Convertir le code coûteux de Python à C99 : split/join pour le //isme
 - Appeler les périphériques correctement : *cast...*
- **La « truande » : approche br(ico|an)leur**
 - Confier la conversion au compilateur (PGI, HMPP, ...)
 - Utiliser des outils « libres » style par4all

Prélude à l'épilogue...

GPU (accélérateur)

- Incontournable en HPC
- Indispensable en traitement
- Encore « vert » ailleurs
- Parfait pour BigData ?

- **Convergence ?**

- Puissance de traitement FPU dans GPU sur SoC : Xbox One, PS 4
- AMD/ATI : remplacer la FPU (déjà partagée) par le GPU
- Nvidia : utiliser un ARM comme CPU & GPU comme accélérateur (Cell ?)

Processeurs Légers

- Performance / 10 & TDP / 40
- Serveurs : marché *Cloud*
 - Alternative virtualisation
- COMOD : pour prototypage
 - Mieux exploiter le GPU

Conclusion

- **« Sans la maîtrise, la puissance n'est rien ! »**
 - Accélérateurs GPU & MIC : un « modèle en couche » dépassé
 - Apprentissage langage indispensable : « penser » *Manycore* !
 - Comportement « complexe » à appréhender : ouvrir le capot en permanence !
 - Processeurs légers : une destination « pailleasse numérique »
 - Réduction de la consommation
 - Intrication importante entre hardware/OS/software :
- **Rien sans l'humain :**
 - Explorer les nouvelles technologies & partager les résultats
 - Anticiper les besoins des cœurs de métiers

Iconographie

- <http://www.nasa.gov/centers/dryden/news/FactSheets/FS-008-DFRC.html>
- <http://cg2010studio.wordpress.com/2011/10/04/>
- <http://www.cray.com/crayXMP.htm>
- http://s1.ibtimes.com/sites/www.ibtimes.com/files/styles/v2_article_large/public/2013/03/05/cyanogenmod-10-1.jpg
- http://ascendances.files.wordpress.com/2011/10/timeline_debian_1993_20111.png
- http://www.break-in-studios.com/archive/TheLastStarfighter_CG.htm
- http://www.pointsdactu.org/article.php3?id_article=1611
- ...

« Catalyser » l'informatique scientifique : CBP : Maison de la Simulation & Centre d'essais



Nasa X29

- Cellule de F5
- Moteur de F18
- Servos de F16
- Études
 - Flèche inversée
 - Incidence $>50^\circ$
 - « *Fly-By-Wire* »

Le CBP (via son pilote d'essais) réutilise et explore de nouveaux domaines...

Une Plate-Forme Expérimentale : Des plateaux techniques

- Multi-nœuds : 76 permanents de 4, 24 et 48 nœuds
- Multi-cœurs : 100 stations/serveurs/nœuds de 2 à 20 cœurs
- **Multi-shaders : 28 stations/nœuds avec 20 (GP)GPU différents**
- Intégration logicielle (versions de distribution)
- Paillasse numériques : expérience/démonstrateur/prototype
- **Intégration matérielle (Sparc, PowerPC, ... et ARM)**
- Visualisation scientifique : plateau 3D
- **Plateau COMOD : « *Compute On My Own Device* »**

Pour étudiants, enseignants, chercheurs, ingénieurs