

Java embarqué

Présentation faite en Java
sur un microcontrôleur AVR32 d'Atmel



Ecole des Mines de Nantes

31 Janvier 2008

fred.rivard@ist-eu.com

www.ist-eu.com

● Milestones

- 2002-2004:
Company incubation
2004: IST is born
- 2004-2006:
Technology development
(more than 30m.y)
- 2006-2007:
MicroJvm on Blackfin, AVR,
AVR32, ARM7, ...

● Shareholders

- ACE Management (Paris)
- Private investors

JEI and R&D lab CIR

● Management

- Fred Rivard, PhD
CEO & CTO
- Régis Latawiec
Sales Director
- François You
Financial Director

● Teams

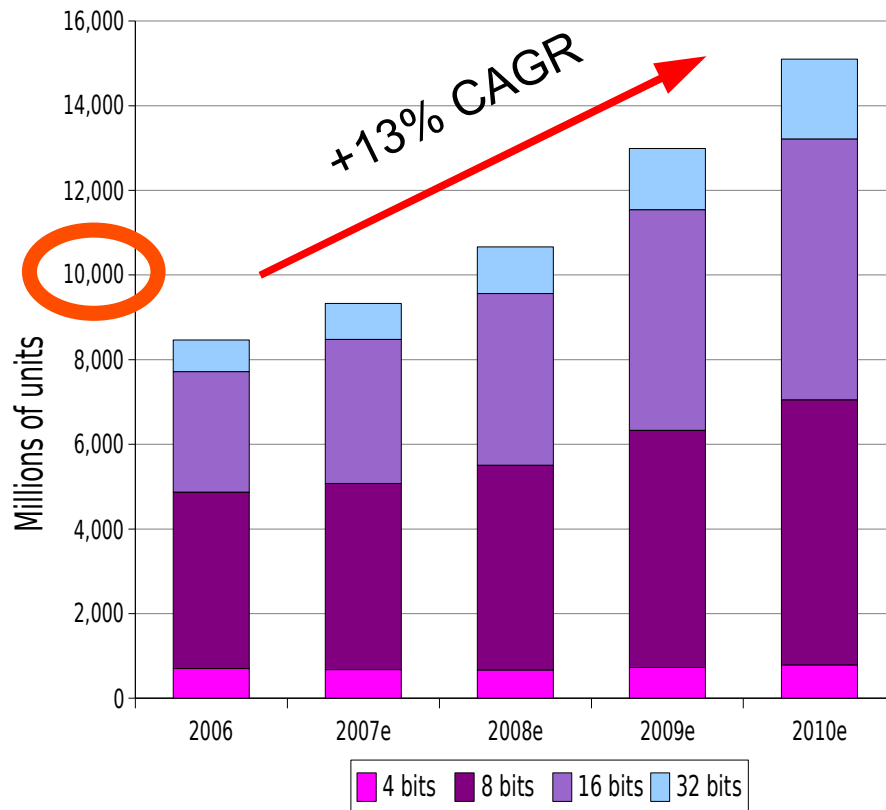
- France:
Headquarters / R&D center
- UK & Germany:
Sales offices & Training center
- Total headcount:
25 people (increasing)

- Medical
- Security
- Telecoms ←
- Handsets
- Multimedia
- Home appliances ←
- Building automation ←
- Industrial control ←
- Automotive ←
- Transportation
- Defense ←
- Avionics ←

← **Marchés adressés par IST**

Marché des MCU en croissance
16-bit & 32-bit MCUs sont les locomotives

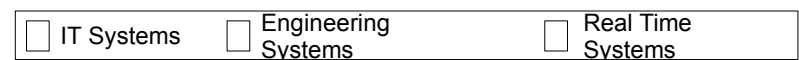
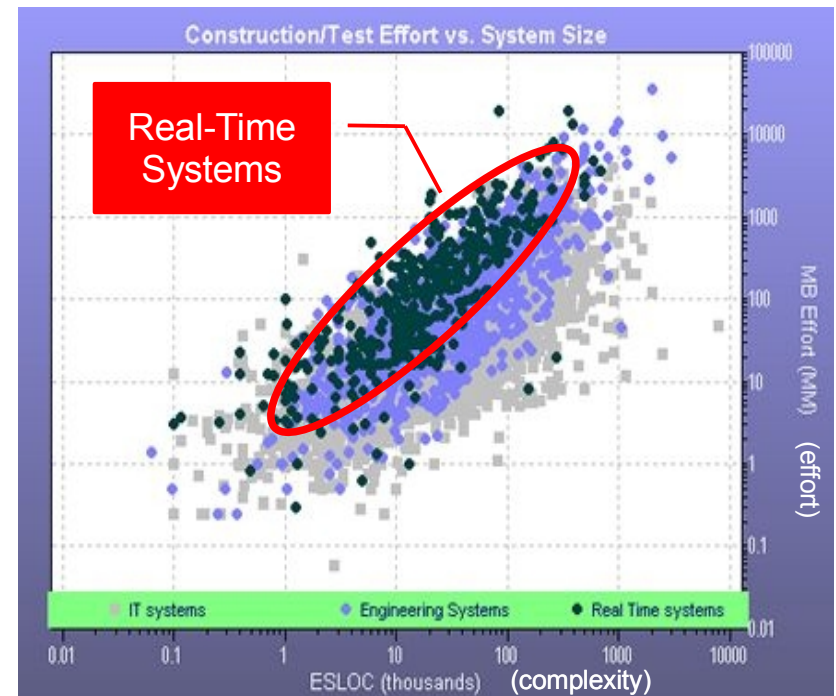
Microcontroller Shipments



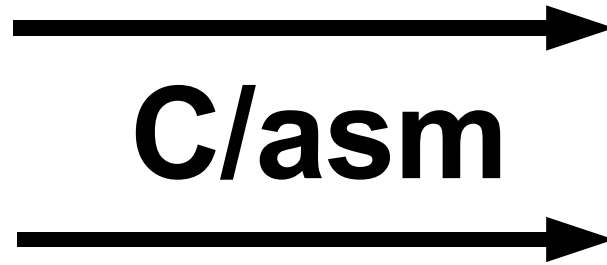
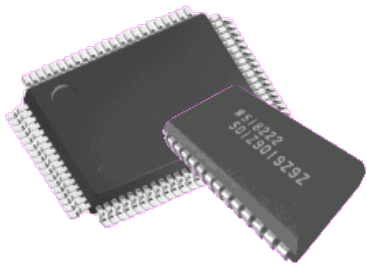
Source: Semico Research Corp.

Les systèmes industriels et temps réels sont très difficiles (chers) à construire/valider

Construction / Test Effort versus System Size & Type

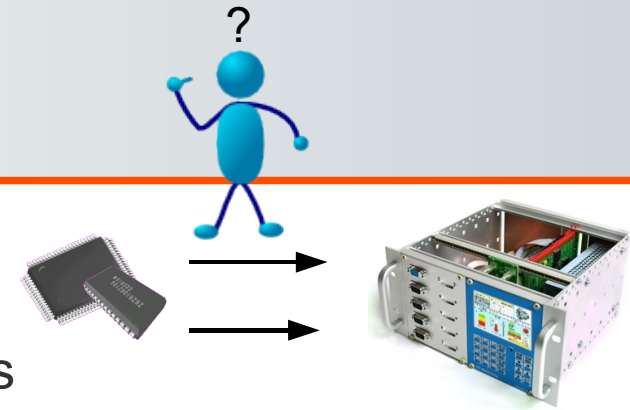


Source: QSM



- Technologies logicielles des années 80, principalement C & asm
- Aucune alternative pour la majorité des produits (85%)
- Matériels et logiciels sont fortement liés

- **Place : la ressource mémoire**
 - RAM : 64 Ko (30.000 fois moins qu'un PC !)
 - Flash / EEPROM / ... : 256 Ko
 - Ecrans B&W (128x128 = 2Ko) versus Couleur VGA (640x480 = 1200Ko)
- **Vitesse : la ressource cpu**
 - 8MHz – 180MHz / Exécution « sur place »
 - Les bus mémoire
- **Consommation : la ressource électrique**
 - Appareils sur batteries
- **Coût unitaire**
 - Time-to-Market (marges)
 - Effet volume



● Briques matérielles

- Des centaines de milliers de références différentes
- Des périphériques internes (mcu), multiplexés, externes (sur cartes), ...

● Briques logicielles

- OS/RTOS (des centaines)
- Outils de développements IDE, compilateurs C (linkeurs pas compatibles), choix des options de compilations (des dizaines), bibliothèques “mal” écrites, ...
- Drivers “gratuits” , BSP, framework middleware, etc ...
- Peu de support sémantique dans le langage C (parfois vu comme un macro-assembleur) par d'autres “super outils” ou langages (automates).

● Maintenance et évolution

- Durée de vie des systèmes (en apposition avec le matériel)
- Modification et évolution de systèmes complexes écrits “sans aide”

Eric PILAUD, Membre du directoire de Schneider Electric

« *Assises du Logiciel Embarqué, Paris, mai 2007* »

Schneider Electric est de plus en plus challengé sur son coeur de métier. Il faut 18 mois à un concurrent pour commercialiser un produit équivalent. **Schneider Electric cherche donc à mettre de plus en plus de logiciels dans ses produits car c'est une arme pour de l'innovation intermédiaire.** Capacité à ajouter sur site de nouvelles fonctionnalités dans nos équipements, pour préserver notre avance par rapport à la concurrence.

- **Sûreté de fonctionnement**
 - Vie humaine en jeux (responsabilité)
 - Gros volumes d'appareils (risques financiers)

- **Certification & logistique**
 - DO178B, SIL, CMMi, Preuves formelles, ...

- **Composants logiciels**
 - OSGi, Fractal, ...

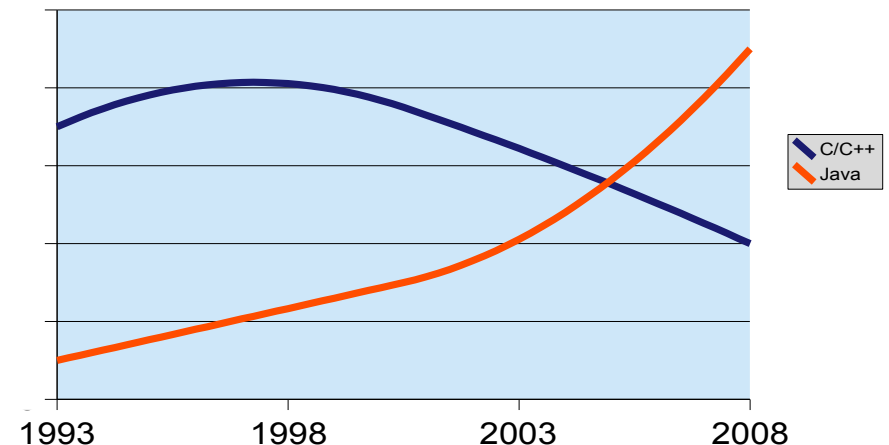
Productivité comparée entre Java et C

Language	Software complexity		Ratio to Java	Cost Reduction	
	Low	High			
Java	14	97	1		
C++	29	178	1.9	C++ → Java	46%
C	9	704	6.4	C → Java	84%

Source: QSM Function Point Programming Table

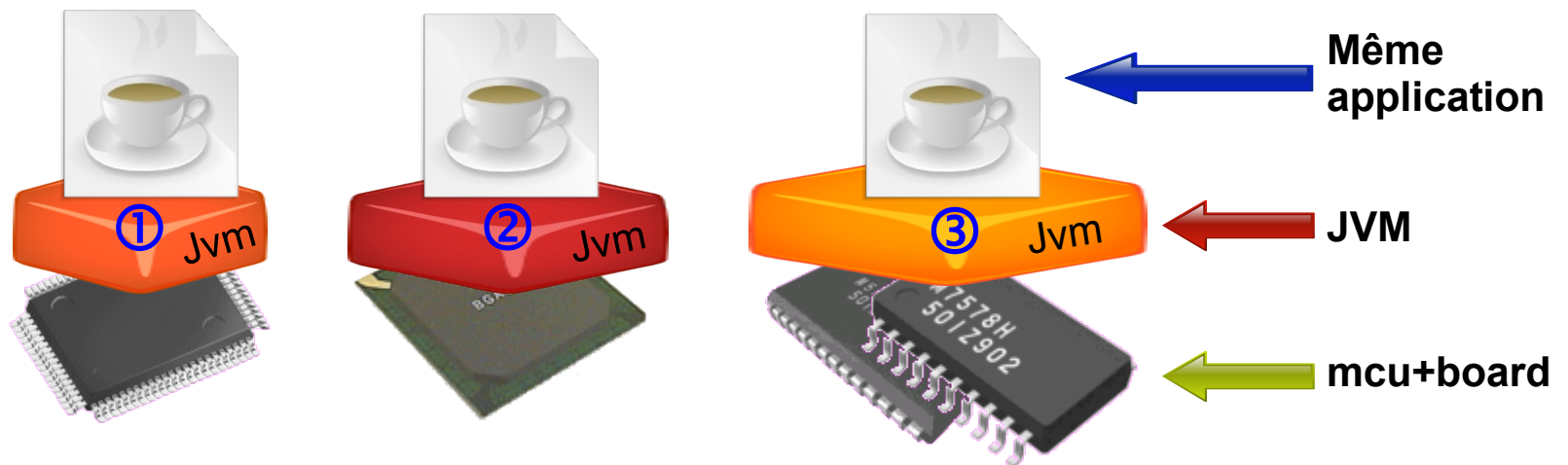
- Productivité C / Java : x6 à x12 dans le monde de l'embarqué
- Programmeurs Java dans le monde > 3.5 millions
- Plus de 4.000.000.000 JVMs

Programming Language Trends (IEEE software 2005)



- **Trois niveaux d'abstraction : une plate forme d'exécution**
 - Matériel ↔ Machine Virtuelle ↔ Application logicielle
- **Une machine virtuelle Java, c'est :**
 - Un processeur 32-bit (~200 instructions)
 - Un gestionnaire automatique de mémoire
 - Un système d'exploitation

**Stable depuis
10 ans**



● J2ME CLDC

- jcp.org
- JSRs

139/271/82/228/239/75/226/229/
180/205/...

● J2ME MIDP

Expert Group

Access Co., Ltd
Anjos, Luiz Carlos Bentes dos
AromaSoft Corporation
Celltick Technologies Ltd
Cisco Systems
Digital Chocolate
Flach, Alcides Luo
Golding, Paul
Hybrid Graphics Ltd.
Infospace, Inc.
Katin, Neil
Luz, Marlon Faria da
Mr. Goodliving LTD
NTT DoCoMo, Inc.
Ostrowalker, Joshua
ProSyst Software GmbH
SavaJe Technologies
Sharp Corporation
Sony Ericsson Mobile Communications AB
Sun Microsystems, Inc.
Telecom Italia
Tira Wireless
Wandarti, Daniel
XCE Co., Ltd.

ACCESS Co., Ltd.
Alcatel Business Systems
Aplix Corporation
ARM Ltd.
Aromasoft Corporation
Beijing ZRRT Communications Technology Co., Ltd.
Danger, Inc. (US)
Ericsson Mobile Platforms AB
esmertec AG
iaSolution
IBM
Ill
LG Electronics
LG Telecom, Ltd.

Matsushita Electric Industrial Co., Ltd.
Motorola
Nokia
Research In Motion Ltd.
Sagem
Samsung Electronics Co., Ltd.
SavaJe Technologies
Sharp Electronics
Siemens AG
Sony Ericsson Mobile Communications AB
Symbian
Tao Group
Toshiba
Veloxsoft, Inc.

Alticast Corp.
Aplix Corporation
Beijing ZRRT Communications Technology Co. Ltd.
Chee, Wei-Meng
Communology GmbH
Ericsson AB
Freire de Albuquerque Junior, Herval
Google Inc.
IBM
Intel Corp.
Lahoda, Pavel
Monotype Imaging Inc
Musiwave
Orange France SA
OZ Communication Inc
Research In Motion, LTD (RIM)
SBC
Siemens AG
Sorrent Inc.
Symbian Ltd
TELUS Mobility
Veloxsoft, Inc.
Wideray
Zuschlag, Jesper

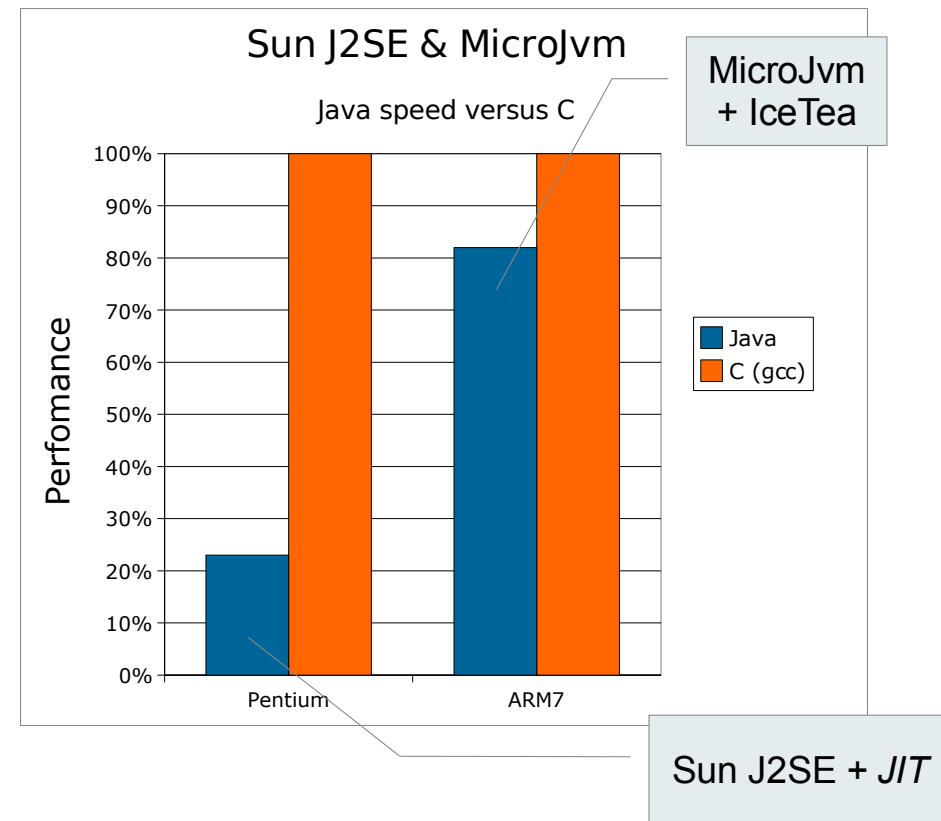
Andreani, Stefano
ARM Limited
Cellon France SAS
China Mobile Communications Co. Ltd
Danger, Inc.
Esmertec AG
Gemalto
Hutchison 3G UK Limited
In-Fusio SA
JAMDAT Mobile Inc
LG Electronics Inc.
Motorola
Nokia Corporation
Ortiz, C. Enrique
Philips Electronics UK Ltd
Samsung Electronics Corporation
Seven Networks, Inc
SK Telecom Co., Ltd.
Sprint
T-Mobile Austria GmbH
Tessarotto, Marco
Vodafone Group Services Limited
Wisegram, Inc

- **De 50ko à plusieurs Moctects**
 - La place mémoire n'est plus un facteur limitant. Java est disponible sur 8-bit, 16-bit, 32-bits et 64-bits multi-coeurs.
 - Tailles des binaires Java plus petites qu'en C (ratio 25% avec code Thumb de ARM7)
- **Java est aussi rapide que du C bien écrit**
 - Sémantique beaucoup plus riche (tableaux bornés, receveur à null, initialisation des objets / tableaux, variables globales initialisée (classes init), polymorphisme, etc ...)
 - Technique d'accélération multiples (JIT, AOT, type Jazelle, IceTea, FPGA, FNI/JNI, ...)
- **Certaines JVMs jouent le rôle de l'OS/RTOS**
 - Les temps de démarrage se chiffrent en 50ms à 67Mhz...
- **Une application Java peut être HRT**
 - Plusieurs sociétés visent à terme la certification avionique DO178B-levelA
 - Groupe de travail au niveau mondial

- **Adaptation des JVMs au matériel sous jacents**
 - Faire appel au vaste éco-système : multiples acteurs (gros, moyens, petits)
 - BootLoader intégré, DMA, MPU, HSB, SPI/USB, DAC, ...
 - Tailles des objets (header, monitor, length, hashcode, ...)
- **Linkage sur desktop**
 - Incohérence du télé-chargement dynamique avec business model.
 - Transformations et optimisations du code binaire
- **Type d'objets**
 - Immortels, en flash, etc ...
- **Connexion avec le C**
 - FNI / JNI
 - C (drivers, bas niveaux), Java (niveau applicatif, haut niveau)

- **Relative performance on real applications compared to C (STR7 ARM7 series)**

- Consider C as the maximum speed on a given platform (Pentium / ARM7)
- Sun J2SE + JIT* compilation is 75% slower than C
- MicroJvm + IceTea is just 20% slower than C



- **Time critical code optimization**

- Use acceleration techniques to speed up identified code sections
- Reach C speed while keeping Java benefits

* *JIT = Just In Time compilation, i.e. acceleration*

MicroJvm™ Benchmarks : Footprint

32-bit MCU Example (AVR32)

- VM + Libraries (Atmel AVR32 AP7 series)

VM & Natives	Size (max)
MicroJvm + CLDC	65KB
Drivers (LCD, ...)	10KB
MIDP 2.0	65KB
TCP/IP	23KB

Java Libraries	Size (max)
CLDC	47KB
MIDP 2.0	107KB

- MicroJvm

- ~ 160KB for a full featured MIDP VM
 - graphic Q-VGA LCD
 - TCP/IP over Ethernet

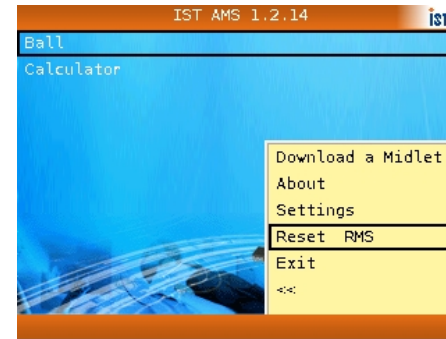


- MIDP Graphics & Internet access

- ~ 320KB to run a complex cellphone game that would required the whole CLDC+MIDP libraries
- Average ~250KB is enough

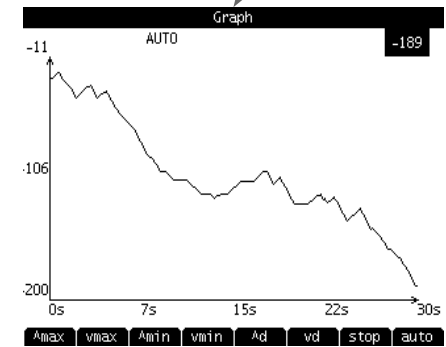
- **Keep high selling prices**
 - Add nice features to products using IST ready-to-use Java solutions
 - Use productive design processes and be first on the market

- **Cut Bill Of Material cost**
 - Access to Java benefits but do not increase the bill of material



High-End Example

Same solution inspired from cellphone technologies



Low-End Example

Minimum System Requirement (32-bit MCU)

.NET Micro Framework

256KB RAM / 512KB Flash

CPU Price ~ \$8.20

JVM

32KB RAM / 128KB Flash

CPU Price: ~ \$3.50

32-bit CPU Price difference is close to 5\$

- **Increase productivity**

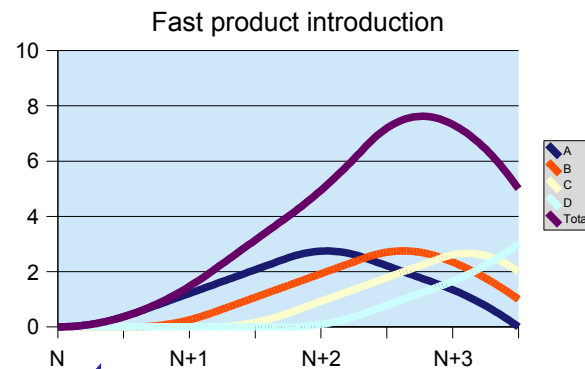
- Speed-up development
- Shorten prototyping, qualification and maintenance processes
- Design more products with less engineers
- Prefer productivity rather inefficient (offshore) contracting

“Java is 3x to 9x more productive compare to traditional design means”

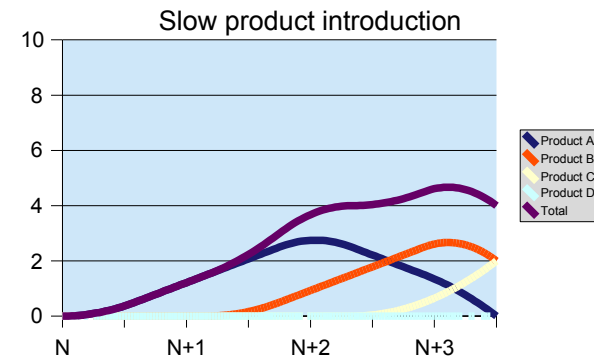
“Java is 6x faster to find bugs in programs”

- **Productivity impact**

- Speed-up revenues
- Improve ROI

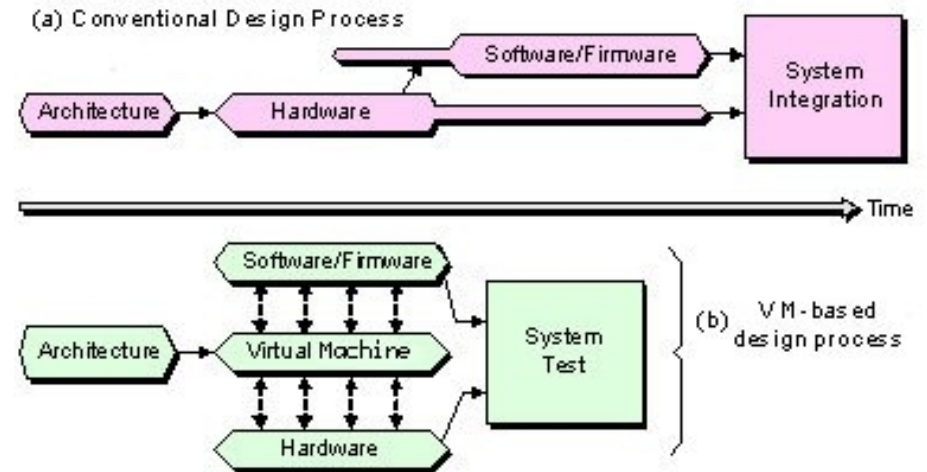


← Shorten product introduction



- **Split hardware and software processes**

- Remove software teams from project critical path
- Design one software program
- Use several hardware platforms



- **Cost reduction impacts**

- Reduce development time
- Capitalize on software investment and improves ROI
- Reduce impact on software due to hardware changes (for cost reduction, etc.)

- **No risk to migrate to Java**

- No hardware changes required to use embedded Java
- Reuse your legacy code when required

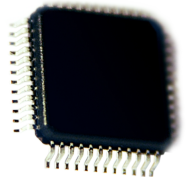
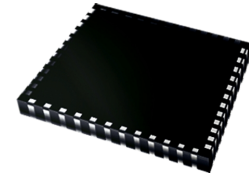


- **Java trends**

- Java become the preferred language
- Best engineers' choice

● Multi-sourcing

- Most devices have electronic parts multi-sourced
- Most the time microprocessors are *not* multi-sourced
- Java programs are binary portable across operating systems and microprocessors
- Java programs are independent from microprocessors



● Semiconductor supplier management

- Avoid procurement shortage
- Maintain pressure on suppliers



Java embarqué

Présentation faite en Java
sur un microcontrôleur AVR32 d'Atmel



Ecole des Mines de Nantes

31 Janvier 2008

fred.rivard@ist-eu.com

www.ist-eu.com