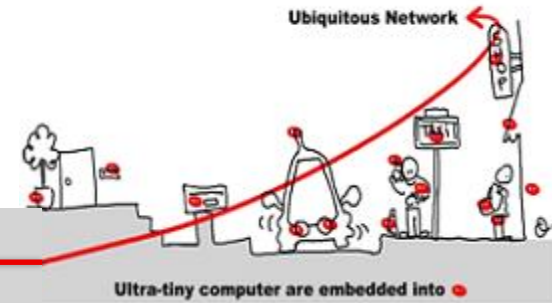


Embedded Microsoft OS and .Net Frameworks



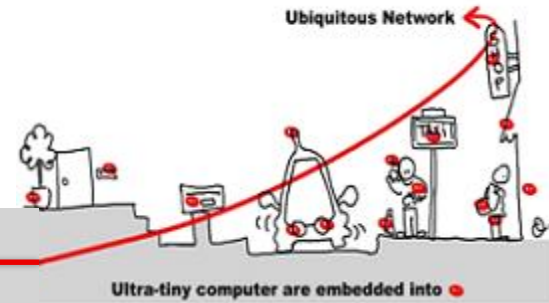
Jean-Yves Tigli

<http://www.tigli.fr>

Polytech of Nice - Sophia Antipolis University

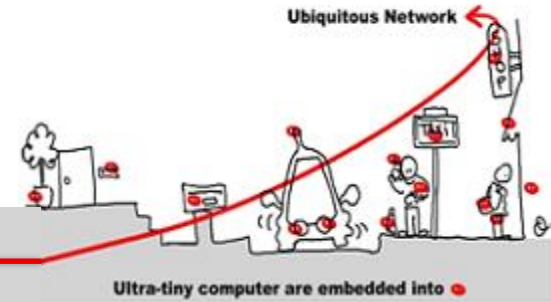
[Email : tigli@polytech.unice.fr](mailto:tigli@polytech.unice.fr)





INTRODUCTION

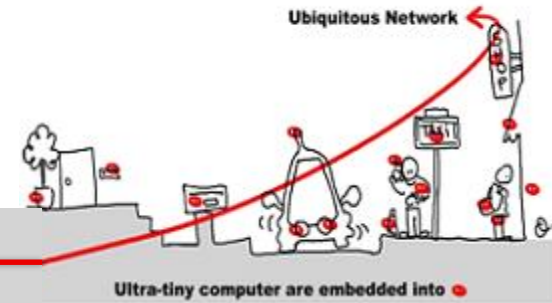
6 familles de solution pour l'embarqué Microsoft



- **Peuvent Différer selon :**
- **Le Kernel**
- **L'OS**
- **Le .Net Framework et le SDK associé**

Référence : Benjamin CHERVY, Responsable Développement et Ressources Technologiques à l'école d'ingénieurs informatiques 3iL, <http://www.lr2i.3il-ingenieur.fr/index.php/systemes-embarques/112-les-systemes-d-exploitation-embarques-microsoft>

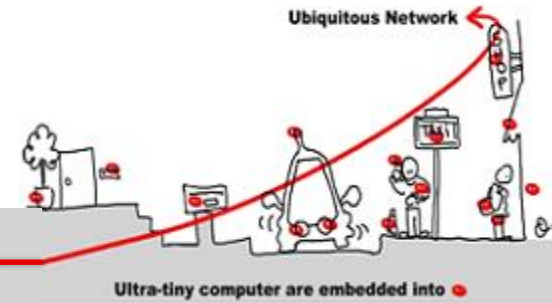
La famille Windows Embedded Standard



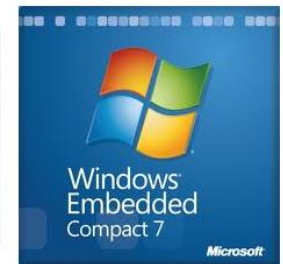
- Cette famille regroupe les systèmes embarqués issus des systèmes classiques de Microsoft pour station de travail que sont Windows XP, Seven et récemment 2008.
- Il s'agit en fait de versions allégées de ces systèmes basées sur le même noyau NT (XP, Seven ou 2008), livrées sous forme de modules binaires qu'il est nécessaire de sélectionner et d'assembler pour générer une image fonctionnelle adaptée à ses besoins.
- Pour exemple, Windows XP Embedded consiste en 12000 composants répartis en 400 fichiers exe, 1600 dll, 300 sys ... avec 176 801 clés et 346 847 valeurs de registre, le Plug & Play, des composants COM, etc.



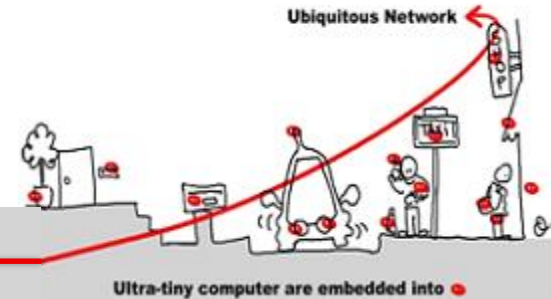
La Famille Windows Embedded CE ou Compact



- Regroupe les différentes évolutions et versions de Windows CE (nommé aussi Windows Embedded CE puis Windows Embedded Compact 7),
- Véritable système d'exploitation embarqué et modulaire, multi-architecture, compact et léger.
- Un système d'exploitation à part entière avec ses propres caractéristiques et son propre code source.

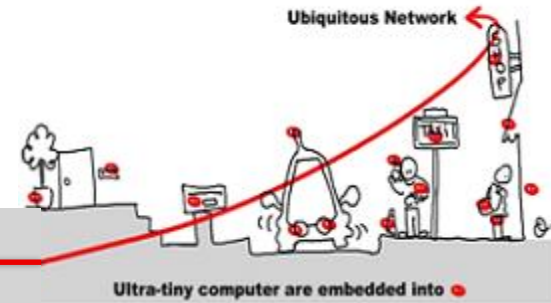


La Famille Windows Embedded CE ou Compact



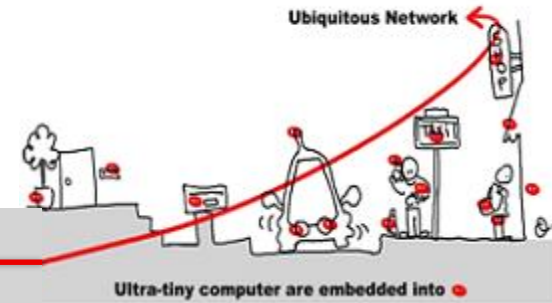
- **Il n'y a aucune relation d'héritage entre un Windows Embedded CE et un Windows classique ou un Windows Embedded Standard.**
- **Pour exemple, les APIs ne sont pas exactement les mêmes, les binaires et drivers ne sont pas compatibles**
- **D'un point de vue caractéristiques, Windows Embedded CE est un système d'exploitation embarqué pouvant être utilisé**
 - **en temps réel,**
 - **avec noyau unifié natif 32 bits,**
 - **supportant de multiples architectures processeurs (x86, ARM dont le dernier ARMv7 pour WEC7, MIPS32 ou SH4),**
 - **multitâche préemptif à priorités fixes, avec ordonnanceur temps réels, et utilisable sur multiprocesseur symétrique (jusqu'à 8 coeurs pour WEC7).**

La Famille Windows Embedded CE ou Compact



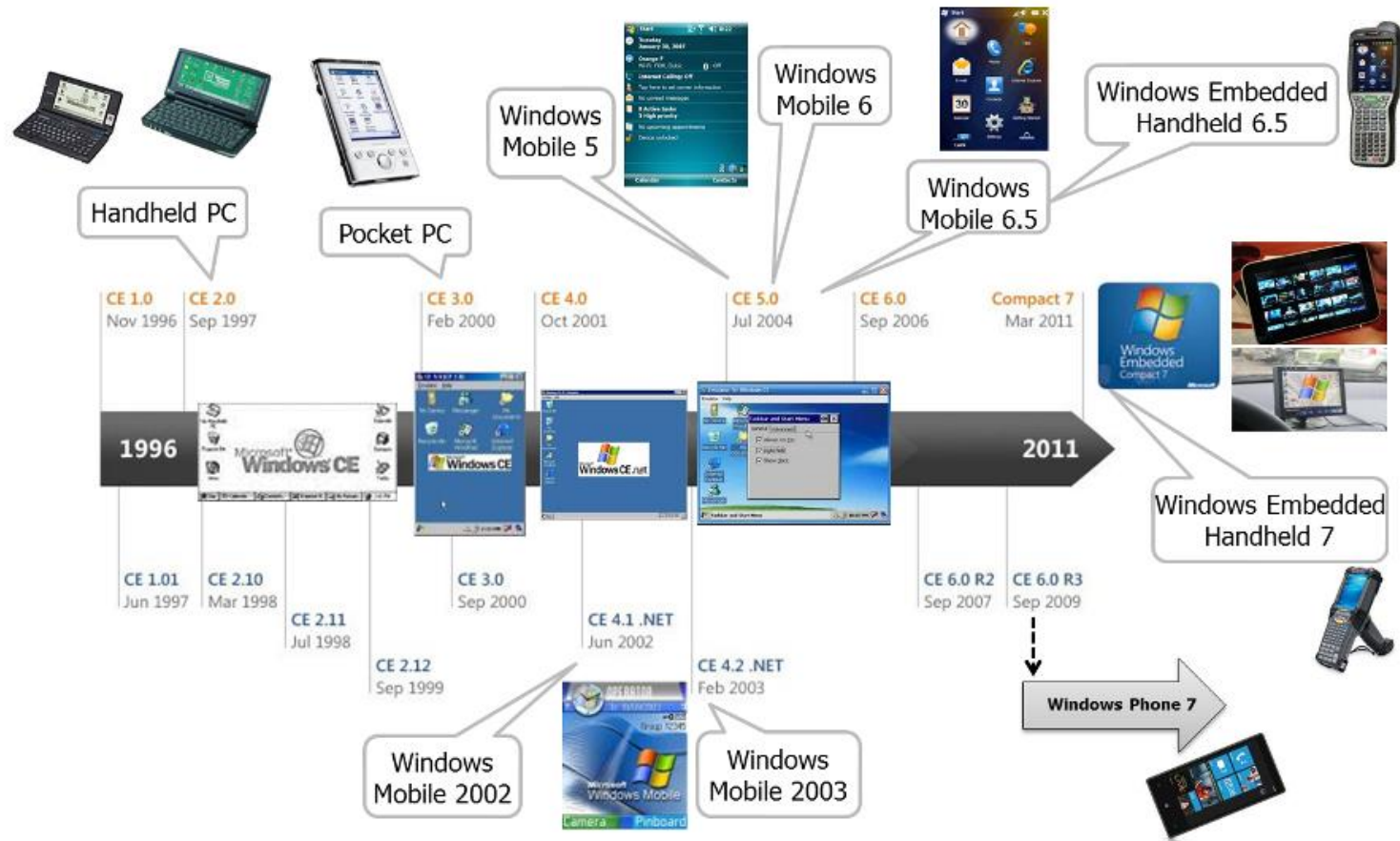
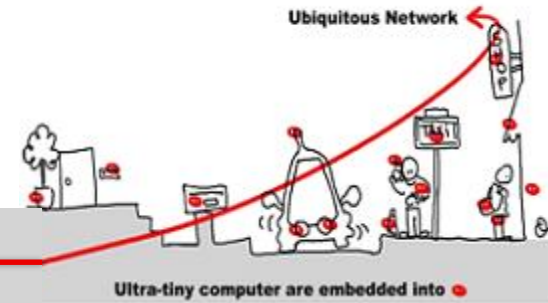
- C'est un système léger qui peut tenir sous 500 Ko de ROM (sans interface graphique) et jusqu'à quelques dizaines de Mo, et qui peut s'installer sans disque dur (résidant en ROM ou RAM).
- Notons que ce système implémente de nombreuses options système, de communication et de réseaux (WiFi, Bluetooth, réseaux téléphoniques GSM/GPRS, IPv6, SD, USB, ...).
- S'il est possible d'implémenter autant d'architecture et de fonctionnalités, c'est tout simplement que ce système nous est livré sous forme de module (plus de 700 composants) à assembler et à compiler en fonction des cibles désirées (processeurs et périphériques).

La Famille Windows Embedded CE ou Compact

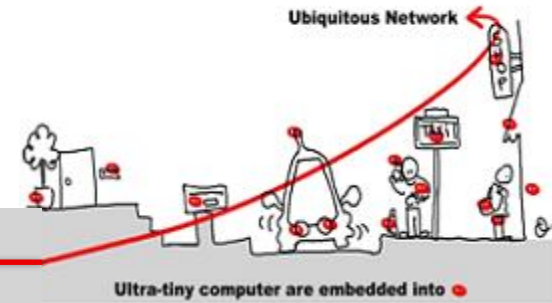


- **Il n'existe pas de Windows Embedded CE générique prêt à installer.**
- **Le système entièrement modulable se construit par compilation à partir d'une sélection de modules divisés en composants sélectionnables individuellement.**
- **Windows Embedded CE est un package logiciel livré avec des codes source à partir duquel il faut créer et compiler l'image du système.**
- **Pour générer son système, les outils nécessaires sont Visual Studio et le logiciel Microsoft Platform Builder qui s'installe comme plugin du premier.**

Les Windows Mobile et Windows Embedded Handheld



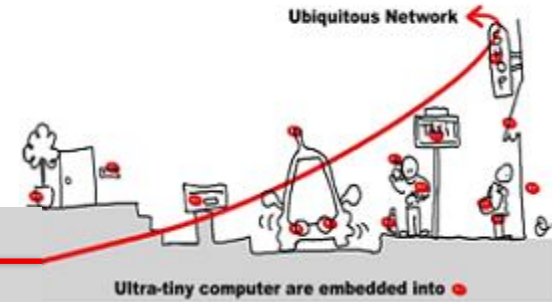
Les Windows Mobile et Windows Embedded Handheld



- **Windows Mobile est la famille des plateformes pour PDA et Smartphone distribuée par Microsoft avant l'arrivée de Windows Phone 7.**
- **Ce n'est pas un système à part entière car il s'agit de plateformes issues de compilations basées sur Windows Embedded CE que nous venons de voir ci-dessus.**



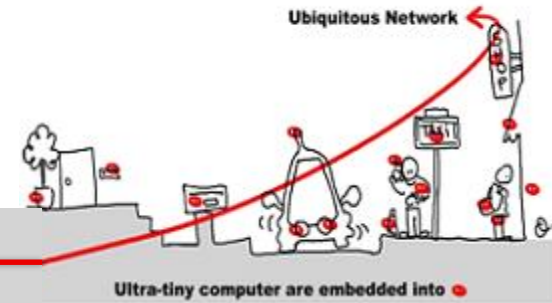
La famille Windows Embedded Server



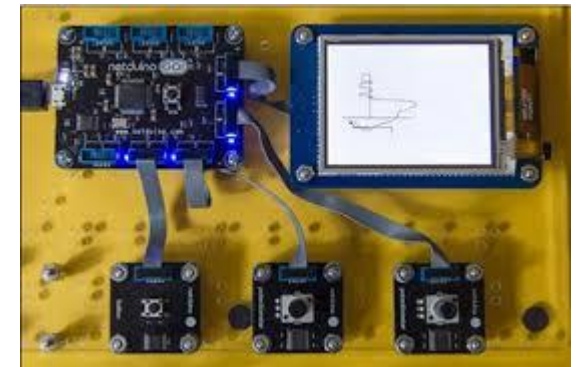
- **Il s'agit en fait de versions de Server 2008 ou 2003, x64, mis à disposition des fabricants ou intégrateurs d'équipements serveurs embarqués dédiés**
 - **aux télécommunications (serveur PABX, serveur de centre d'appels, visioconférence et serveur vocal interactif),**
 - **aux serveurs de médias, à l'imagerie médicale, ou aux automates industriels.**



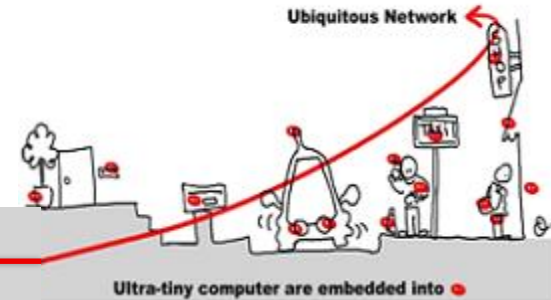
.Net Micro Framework



- le .Net Micro Framework est un sous ensemble du .NET Framework destiné aux systèmes embarqués ultra légers dont les processeurs ne disposent pas de MMU et qui ne requiert pas d'OS pour fonctionner.
- On peut le voir comme un environnement d'exécution bootable dont le but est de faire fonctionner un unique programme (non temps réel).

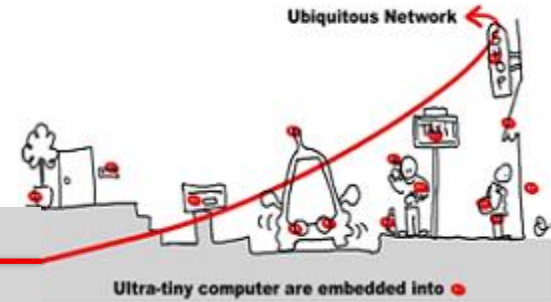


.Net Micro Framework



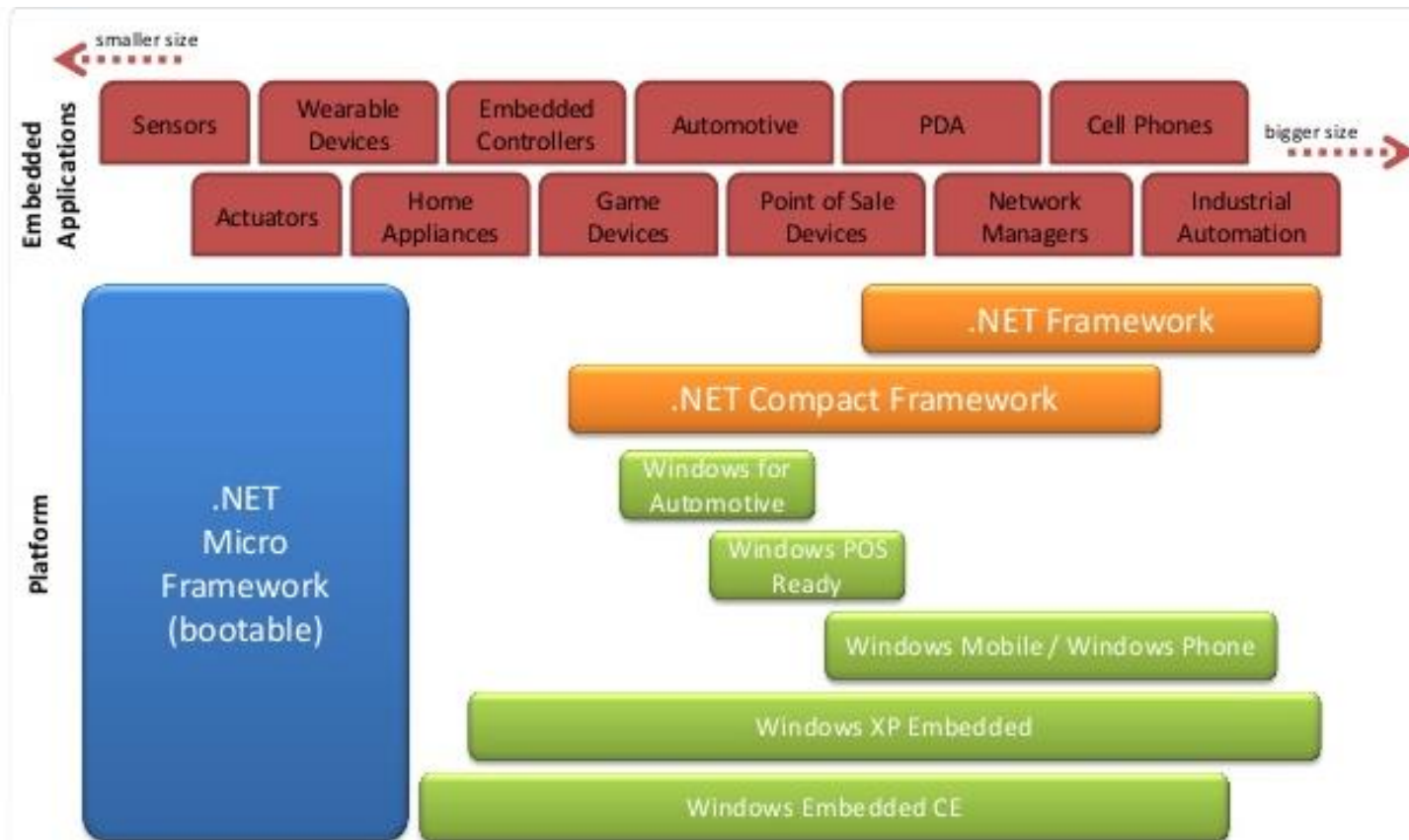
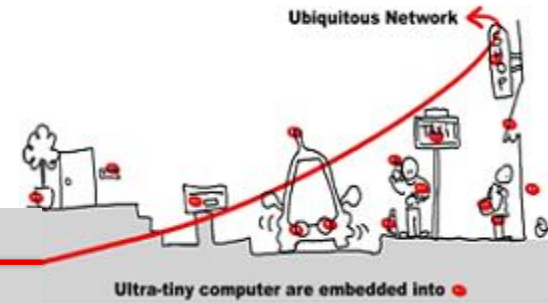
- Il démarre et s'exécute en mémoire flash sur des coeurs Arm7, 11 et CortexM3, nécessite 256 Ko de RAM et 512 ko de Flash.
- Il n'y a plus de 1,5 million d'équipements divers utilisant le micro-Framework (télécommandes, capteurs et tensiomètres, robots, périphériques audio / vidéo, ...).
- Il est maintenant disponible en licence Open Source Apache 2 depuis 2009
- Le développement d'application s'effectue en C# avec Visual Studio et un SDK dédié
- A noter que le développement reste très particulier car le .Net Micro Framework ne fournit que quelques classes principales du Framework .Net (70 au lieu de 1400).

Windows Home Server

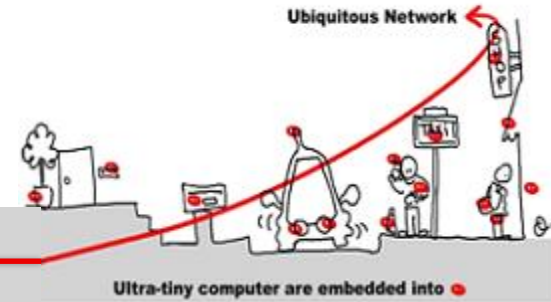


- **Pour serveur numérique permettant la centralisation, la sauvegarde, et la diffusion des contenus multimédias au domicile des particuliers**
- **Version Windows Server 2008 allégée qui implémente un réseau domestique, l'accès à distance et le partage de fichiers multimédia**

Récapitulatif



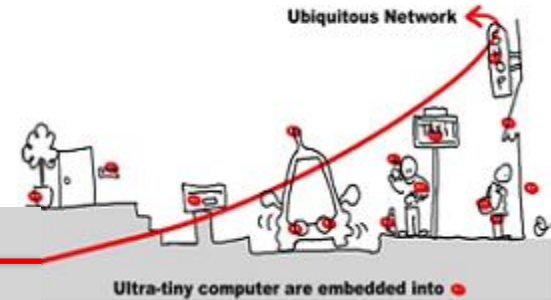
Trois catégorie d'OS



- **Catégorie « Modulaire »**
- **Catégorie « Verticalisé »**
- **Catégorie « Grand Public »**

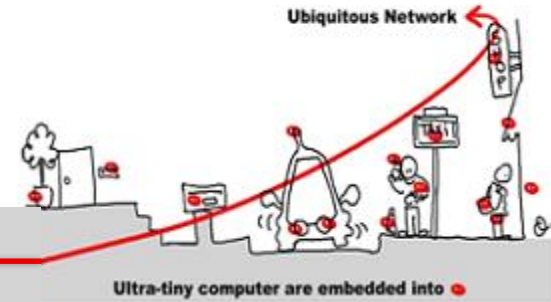
- **Kernels CE ou NT**
- **Windows Embedded Compact 2013 passé sous NT**

catégorie « modulaire »



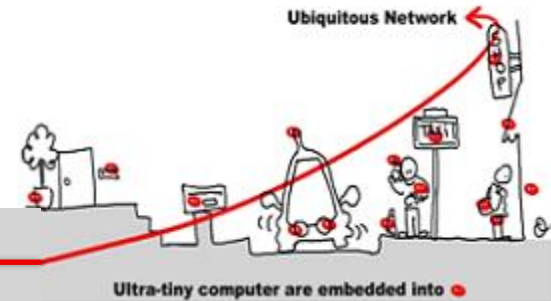
- **Une première catégorie « modulaire » qui est la plus souple.**
 - Elle offre au développeur la liberté de customiser son O/S grâce à des outils qui permettent de sélectionner les composants de l'O/S
 - Windows Embedded Standard 8 qui s'appuie sur un Windows 8 « grand public »
 - Windows Embedded Compact 2013 (nouvelle génération après Windows CE) OS entièrement conçu pour l'embarqué et qui épouse un panel plus large de processeurs (ARM, MIPS et X86) et qui intègre des fonctions de temps réel

catégorie « verticalisée »



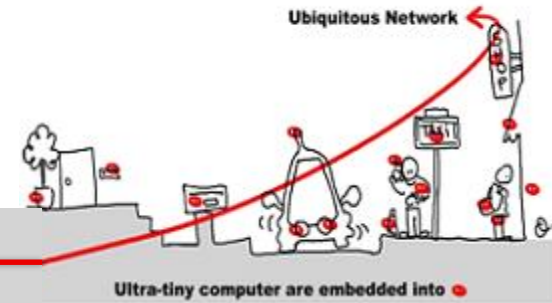
- **Une deuxième catégorie « verticalisée » qui offre des O/S optimisés pour des environnements/terminaux spécifiques**
 - **Windows Embedded PosReady : pour les terminaux Point de Service (caisses enregistreuses ou bornes interactives)**
 - **Windows Embedded Handheld : pour les terminaux Mobiles d'entreprise**
 - **Windows Embedded Automotive : pour l'info-divertissement à bord des véhicules**

catégorie « O/S grand public »











- **Une troisième catégorie « O/S grand public » regroupé dans la famille Windows Embedded Entreprise**
 - Ce sont des versions complètes de Windows 7, Vista ou XP.
 - Le développeur installe un Windows entier mais un contrat de licence spécifique à l'embarqué restreint l'usage à une fonction ou application dédiée.

Récapitulatif

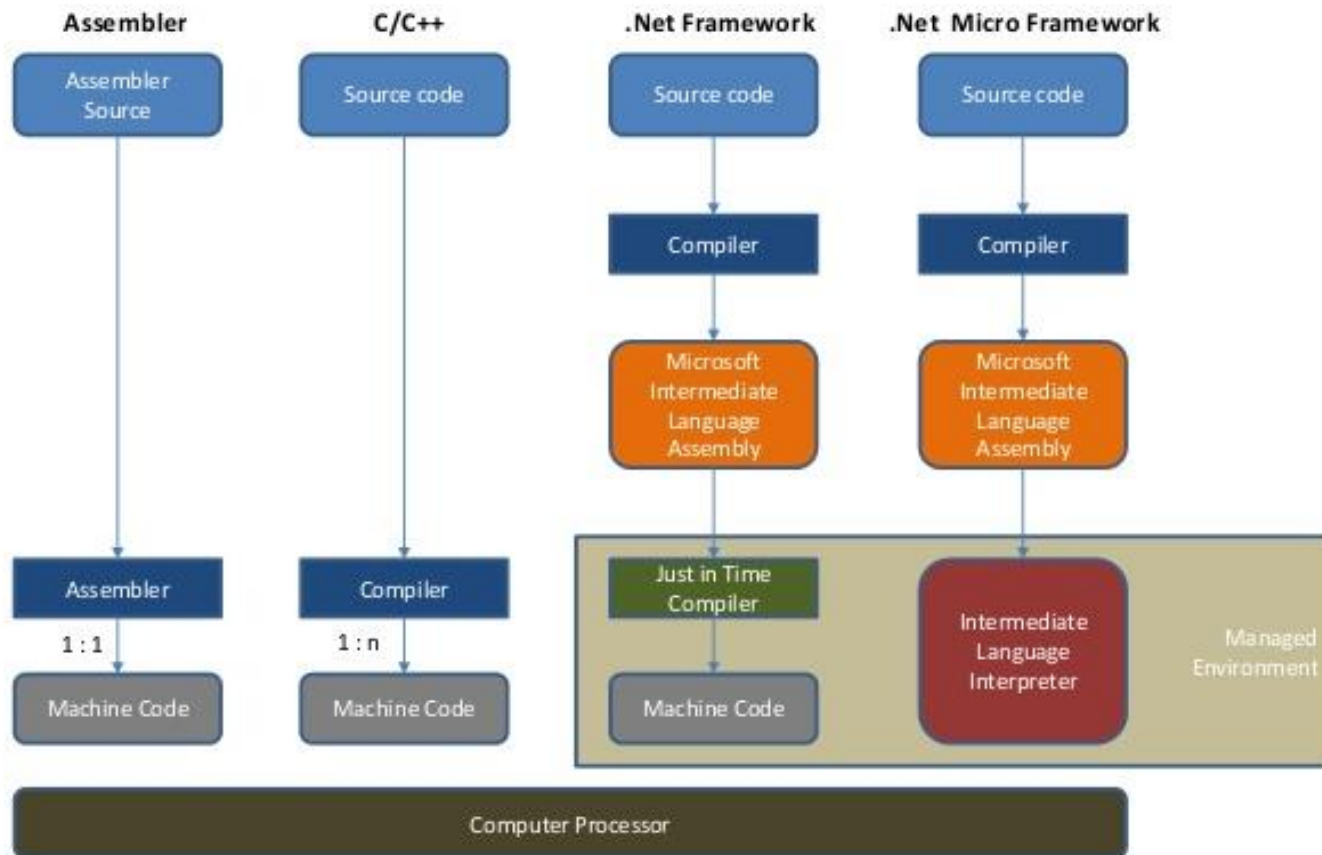
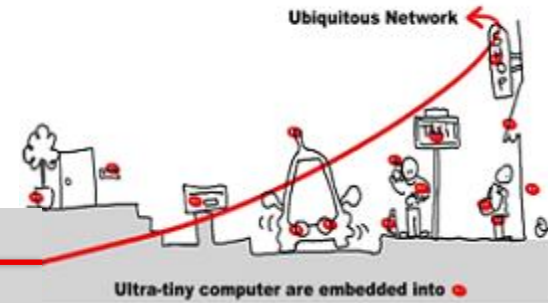


- Les solutions verticales Automotive s'appuient sur Windows CE, POS Ready sur Windows 7 ou XP Pro

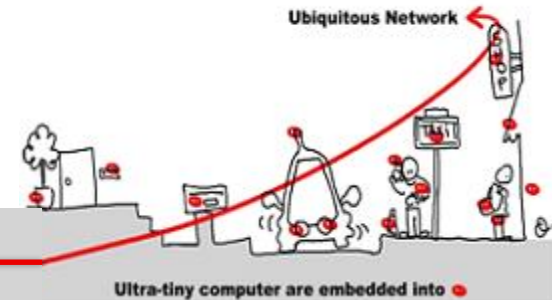
Appareils :

							
Signalisation numérique	Média connectés	Appareils portables	Client léger	Automatisation	Médical	Vente au détail	Automobile
• Standard	• Compact • Standard • Server	• Handheld • Compact	• Compact • Standard • Server	• Compact • Standard • Enterprise • Server	• Compact • Standard • Enterprise • Server	• POSReady • Compact • Standard • Enterprise • Server	• Automotive

Développement et Frameworks

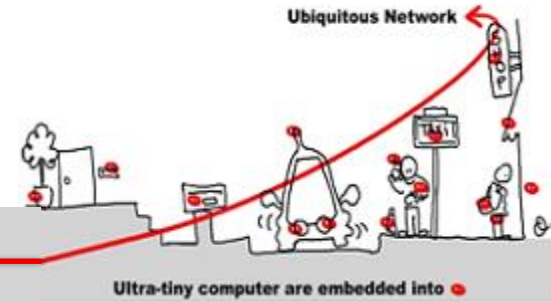


Stratégie Microsoft pour l'IoT



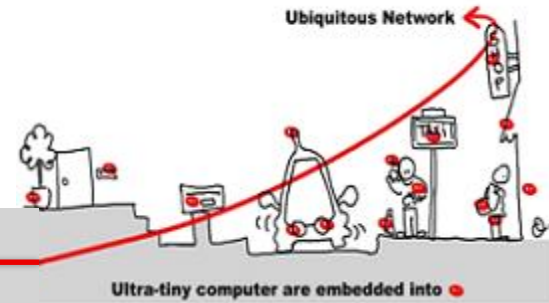
- **Windows Embedded Compact 2013**
 - Système allégé, dont la phase de démarrage et la consommation de mémoire ont été optimisées
 - Kernel NT et plus CE
- **Windows Embedded Compact 2013 a pour vocation de s'appuyer sur Microsoft Azure (Cloud)**
 - Windows Embedded Compact est là pour gérer l'écran tactile de l'objet et les interactions avec l'utilisateur, le traitement des données en découlant serait alors dévolu à Azure

Stratégie Microsoft pour l'IoT



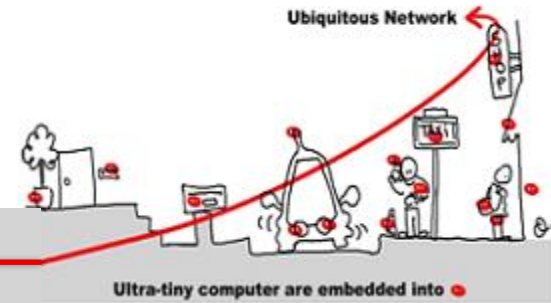
- **Microsoft Azure Mobile Services**
 - optimisés pour servir les terminaux Windows 8.1, Windows Phone 8, mais aussi iOS et Android
 - socle pour la couche serveur d'objets connectés outillés par Windows Embedded Compact 2013





.Net Micro Framework et .Net Gadgeteer

.Net Micro Framework et .Net Gadgeteer

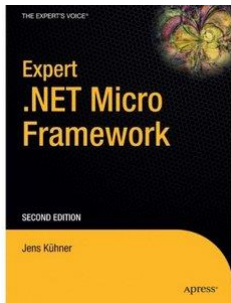


Jean-Yves Tigli

<http://www.tigli.fr>

Polytech of Nice - Sophia Antipolis University

[Email : tigli@polytech.unice.fr](mailto:tigli@polytech.unice.fr)

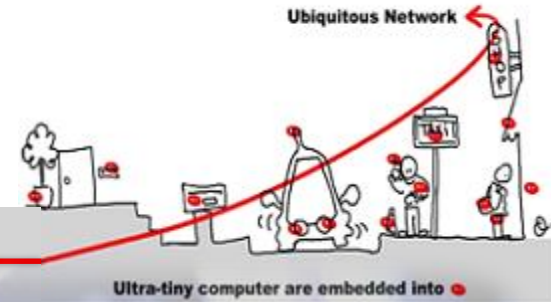


http://www.windowsfordevices.com/files/misc/Kuhner.NETMicroFramework_Ch4_sample.pdf

<http://www.thierry-lequeu.fr/data/Beginners-Guide-to-NETMF-French.pdf>

<http://www.tinyclr.com/support/>

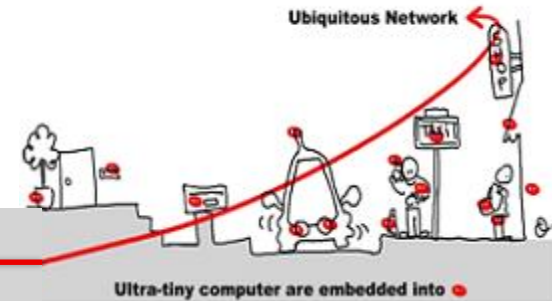
.NET MicroFramework



- **Historique & état des lieux**
- **Architecture**
- **A l'intérieur du MicroFramework**



Historique



- **2001 : Début de SPOT chez Microsoft Research (Smart Personal Object Technology)**
- **Produits déjà lancés:**
 - Smart Watches (2004)
 - Microsoft TV (2005) (maintenant sous CE)
- **En cours de lancement :**
 - Microsoft Windows Vista SideShow



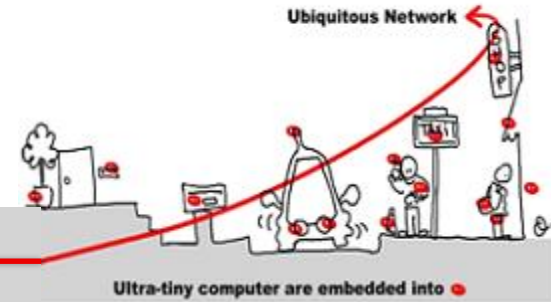
Hardware

- PortalPlayer PP5024
Dual-core ARM7
- USB connection to host PC
- Self-contained battery
- 320x240 QVGA TFT display
- WMA/MP3 Playback



Motivation

Ce que propose Microsoft ...



- **General**

- Microsoft Windows XP Embedded
- Microsoft Windows CE

- **Dérivés**

- Microsoft Windows Automotive 5.0
- Microsoft Windows Embedded for Point of Service

- **Taille de Windows CE**

- Le noyau de CE est $> 600\text{Ko}$ selon les options
- Windows CE + .NET Compact Framework (CF) = 12Mo
- Windows CE a besoin d'un MMU

S'ouvrir sur d'autres périphériques

.NET Micro Framework



Wearable Devices



Auxiliary Displays



Health Monitoring



Remote Controls

Windows CE



Windows Mobile Smartphone



Windows Mobile Pocket PC Phone



Portable Media Center



Windows Automotive



Mobile handhelds



VoIP phones



Set-top boxes



Gateways

Windows XP Embedded



Retail Point-of-Sale



Windows-based terminals



Medical devices

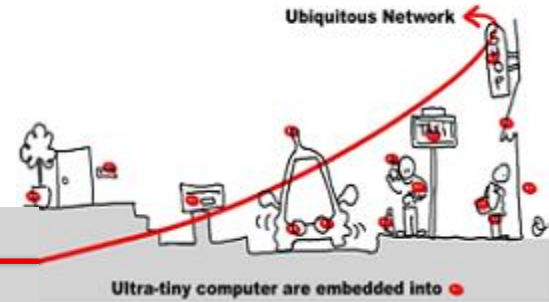


Entertainment devices

Fonctionnalités croissantes

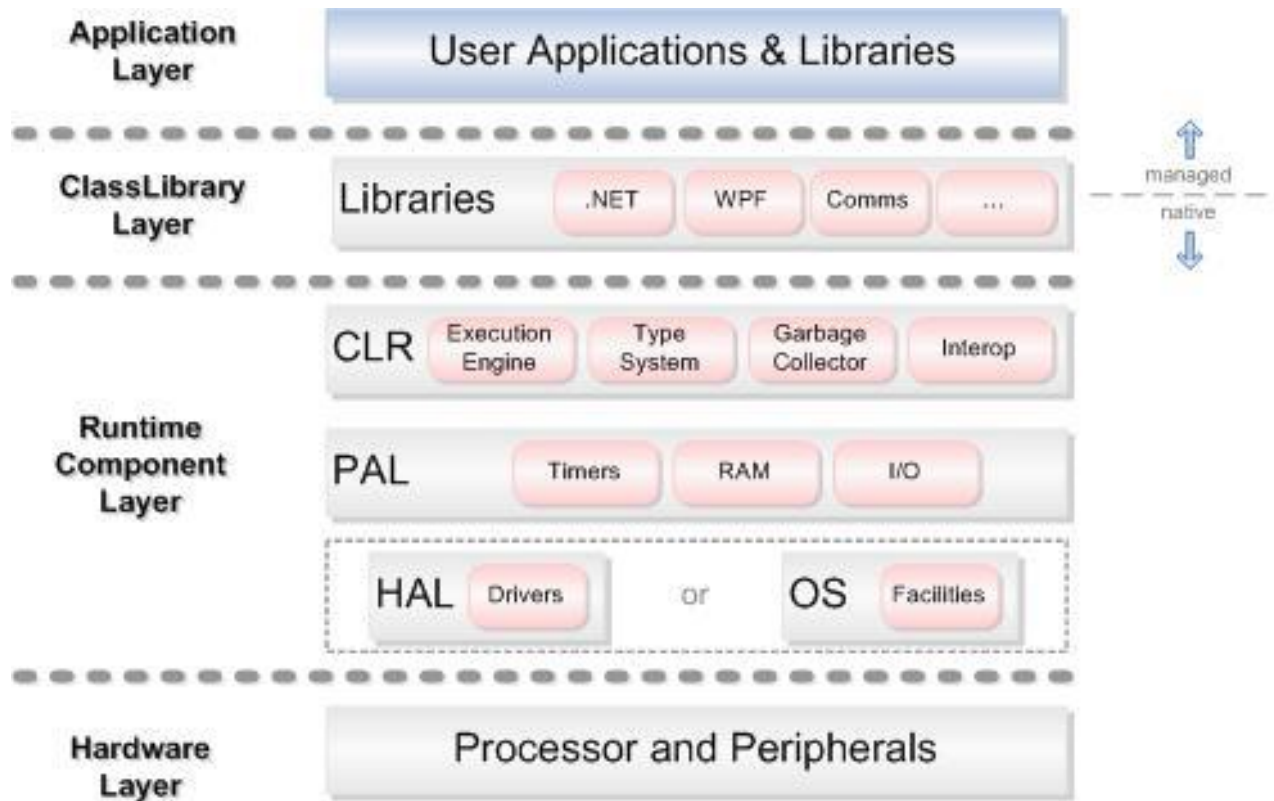
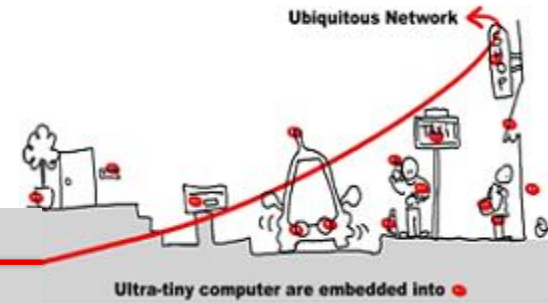
Architecture

Philosophie et buts

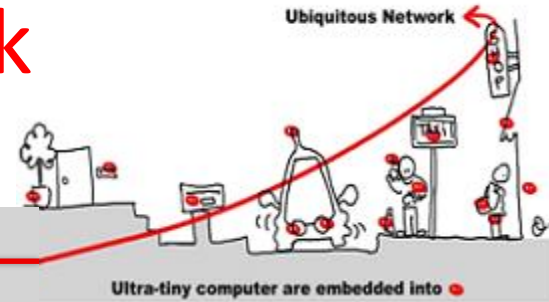


- **Bootable .NET**
 - Bénéfices du code managé, proche du hard
- **Créer la plus petite plateforme compatible .Net pour les périphériques ayant peu de ressources (CPU, mémoire, alimentation)**
- **Compromis**
 - Empreinte mémoire réduite, options de boot avancées et BSP (Board Support Package)
 - CLR runtime
 - Librairies essentielles (mscorlib, graphics, HW, Net)
 - Librairies sous forme de composants pour étendre des périphériques particuliers

Architecture



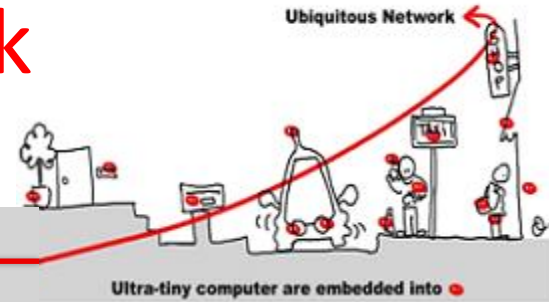
A l'intérieur du MicroFramework



- **Chipsets supportés**
 - ARM
 - ARM7TDMI, 25Mhz, 384Kb RAM, 1/2Mb FLASH, 8Kb D/I cache
 - ARM920T, 96Mhz, 4Mb RAM, 2Mb FLASH, 16Kb D/I cache
 - ARM Cortex M3 (en cours de développement)
- **Plateformes supportées (en tant qu'hôte pour le CLR)**
 - M68k
 - Win Embedded XP
- **Porter vers un nouveau hardware : ~80-100 fonctions**
- **Porter vers une nouvelle plateforme : ~20-30 fonctions**
- **Un kit pour porter le micro framework vers un nouveau hardware est disponible**

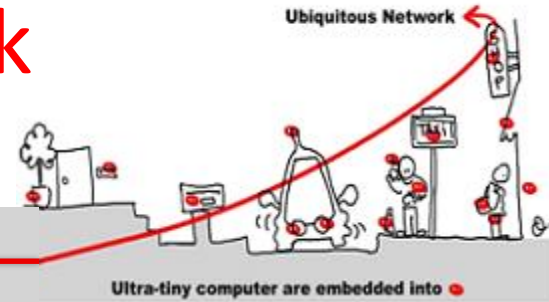
A l'intérieur du MicroFramework

HAL + PAL



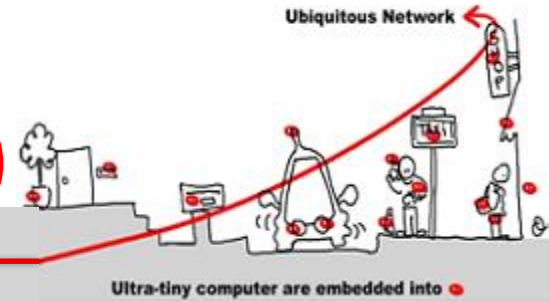
- **Conçus pour .NET Micro Framework**
 - Bootstrap
 - Abstraction du matériel
 - Mécanisme de report d'appel de procédure (Deferred Procedure Call (DPC))
 - Priorités des appels de fonction
 - Gestion des appels asynchrones (entrées/sorties)
 - Forme de multitâche coopératif
- **Simple**
 - Ni scheduler ou gestion de mémoire
 - Blocage minimum (seulement les ISRs)
 - Seul le CLR est exécuté

A l'intérieur du MicroFramework HAL + PAL



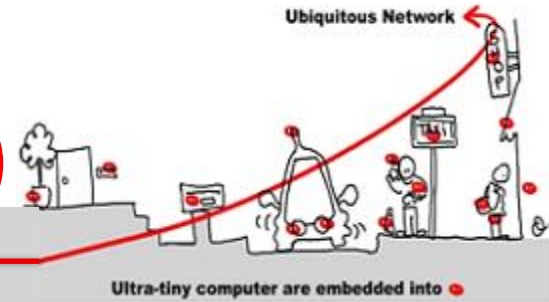
- **Petit**
 - 40Ko (avec les drivers principaux)
- **Execution du CLR**
 - Directement sur le HW (via HAL, Hardware Abstract Layer)
ou
 - En tant que tâche d'un RTOS (via PAL, Physics Abstraction Layer)
- **PAL est une surcouche de HAL**

A l'intérieur du MicroFramework Common Language Runtime (CLR)



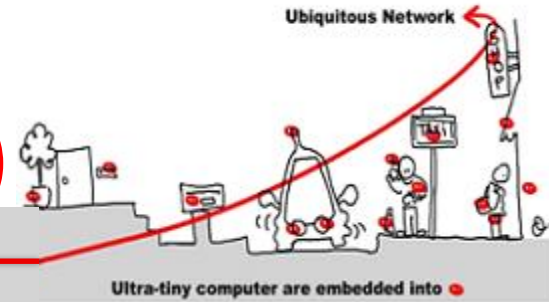
- **Dérivé du standard ECMA (CLI)**
 - CLS/CTS/VES (common language specification, common type system, Virtual Execution System)
 - Aucun appel unsafe
 - Quelques extensions spécifiques à l'environnement embarqué
- **Taille très réduite**
 - runtime + framework library: ~200 KB
 - <500KB toutes options (UI, XML, ...)

A l'intérieur du MicroFramework Common Language Runtime (CLR)



- **Gestion mémoire**
 - Garbage collector
 - Support pour de la mémoire non-volatile
 - Optimisé pour le ratio size/performance
- **Moteur d'exécution**
 - Charge et exécute le code managé (MSIL)
 - MSIL interprété
 - Execution avec une faible granularité
 - Code plus petit
 - Multi-threads
 - Gestion des exceptions

A l'intérieur du MicroFramework Common Language Runtime (CLR)



- **Moteur de sérialisation**
 - Optimisé pour la mémoire non-volatile
- **RPC**
 - Communication bi-directionnelle
 - Au travers de différents types de liens (Bluetooth, USB,...)
- **Réseau**
 - Drivers 802.15.4 (Chipcon cc2420) et BT (Promi-SD)
 - Bientôt une stack TCP/IP (System.Net.Sockets)
- **Drivers managés**
 - GPIO, PWM, VTU32, I2C, SPI, USART

.NET Framework

Ubiquitous Network

System.Web

- Services
 - Description
 - Discovery
 - Protocols
- UI
 - HTML controls
 - Web controls

Cache

Security

Configuration

Session state

System.Windows.Forms

Design

Component model

System.Drawing

Drawing 2D

Printing

Imaging

Text

System.Data

ADO.NET

SQL Client

Design

SQL ServerCE

System.XML

XML Document

Serialization

Xslt/XPath

Reader/writers

System

Collections

IO

Security

Net

Text

Reflection

Globalization

Resources

Configuration

Service process

Diagnostics

Threading

- Runtime
 - Interop services
 - Remoting
 - Serialization

.NET Compact Framework

Ubiquitous Network

System.Web

• Services

- Description
- Discovery
- Protocols

• UI

- HTML controls
- Web controls

Cache

Security

Configuration

Session state

System.Data

ADO.NET

SQL Client

Design

SQL ServerCE

System

Collections

IO

Security

Net

Text

Reflection

Globalization

Resources

System.Windows.Forms

Design

Component model

System.Drawing

Drawing 2D

Printing

Imaging

Text

System.XML

XML Document

Serialization

Xslt/XPath

Reader/writers

Configuration

Service process

Diagnostics

Threading

• Runtime

- Interop services
- Remoting
- Serialization



.NET MicroFramework

Ubiquitous Network

System.Web

- Services
 - Description
 - Discovery
 - Protocols
- UI
 - HTML controls
 - Web controls

Cache

Security

Configuration

Session state

System.Windows.Forms

Design

Component model

System.Drawing

Drawing 2D

Printing

Imaging

Text

System.Data

ADO.NET

SQL Client

Design

SQL ServerCE

System.XML

XML Document

Serialization

Xslt/XPath

Reader/writers

System

Collections

IO

Security

Net

Text

Reflection

Globalization

Resources

Configuration

Service process

Diagnostics

Threading

• Runtime

• Interop services

• Remoting

• Serialization

.NET MicroFramework

Ubiquitous Network

System.Web

- Services
 - Description
 - Discovery
 - Protocols

Cache

Configuration

- UI
 - HTML controls
 - ...

System.Windows.Forms

Design

Component model

System.Drawing

Drawing 2D

Printing

Text

System.Data

ADO.NET

SQL Client

Serialization

Design

SQL ServerCE

Reader/writers

System

Collections

IO

Configuration

• Runtime

Security

Net

Service process

• Interop services

Text

Reflection

Diagnostics

• Remoting

Globalization

Resources

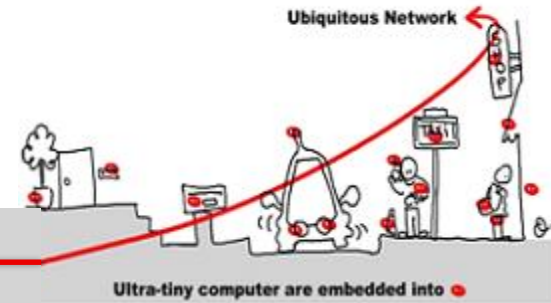
Threading

• Serialization

OPEN SOURCE
Cf. Codeplex



Environnement de développement



- Visual C# express 2012 ou 2013
- <http://www.microsoft.com/express/vcsharp/>
- .NET Micro Framework V4.3 SDK-R2-Beta
- <http://netmf.codeplex.com/releases/view/133285>
- .NET MF 4.3 RTM (QFE1)
- <https://netmf.codeplex.com/releases/view/118283>
- All we need for Gadgeteer
- <https://www.ghielectronics.com/support/netmf>
- SDK GHI NETMF. <http://www.tinyclr.com/dl/>