

# Programmation sur PDA Techniques & Concepts

# Introduction

# Introduction (1)

---

- De la particularités des PDA
  - Echelle, économie d'énergie, données
- À la particularité du développement sur PDA
  - Système embarqué
  - Interface graphique
  - Communications et stockage de données

# Introduction (2)

---

- Rappel de la situation du marché des PDA
  - Essentiellement divisé en deux plateformes
  - Evolution du matériel
  - Contraintes
- Particularité des PDA par rapport aux autres systèmes embarqués: applications et données

# Introduction (2)

---

- Objectifs théoriques
  - Spécificité de la programmation sur PDA (développement croisé, limites des PDAs)
  - Spécificité des applications sur PDA (modèle événementiel, interfaces graphiques, stockage, communication)

# Introduction (3)

---

- Objectifs pratiques
  - Génie logiciel, stages de développement sur PDA
  - Présentation générale et prise en main d'outils (POSE, Waba)
  - Création d'une application minimale

# Introduction (4)

---

- Plan
  - Langages et outils
    - Enjeux, contraintes des PDA
  - Conception d'une application
    - Particularité des applications sur PDA
    - Exemple de Waba

# Langages et outils

# Langages et outils

---

- Contraintes des PDA
- Langages et API
  - Langages spécifiques
  - Programmation objet sur PDA
  - Java
- Processus de développement sur PDA
  - Cruci-développement et cruci-compilation
  - Le rôle des émulateurs & particularité de Java

# Contraintes des PDA

---

- Ordinateur réduit:
  - Faible capacité de stockage
  - Faible puissance de calcul
- Interface graphique:
  - Taille de l'écran réduite
  - Crayon comme pointeur
- Importance des données
- Economies d'énergie

# Langages spécifiques (1)

---

- Distinction entre langage et APIs
- Insuffisance des bibliothèques du langage par rapport aux PDA
- Logique commerciale des langages particuliers
- Particularités de chaque système d'exploitation

# Langages spécifiques (2)

---

- Langages « naturels » de certains PDA et adaptations
  - NewtonScript
  - MS Visual C++/Basic
  - QT
- Limites des langages génériques: Basic, C, C++, Lisp, SmallTalk (Squeak), Java

# Programmation objet (1)

---

- Les avantages usuels de la programmation par objets
  - Réutilisation du code
  - Conception en classes et en modules
  - Utilisation de diagrammes UML pour communiquer au sein d'une équipe

# Programmation objet (2)

---

- Les avantages spécifiques de la programmation par objets sur PDA
  - Eviter la duplication du code -> limitation de la taille des programmes
  - Systèmes d'exploitation/bibliothèques de classes objet (éléments partagés, éléments en ROM, surcharge)

# Java sur PDA (1)

---

- Satisfait certaines contraintes des PDA
  - Taille du code Java
  - Interface graphique modulable
  - Langage orienté objet et ROM
  - Emulation et intégrité des données
- Intérêt d'un langage non spécifique à une plateforme

# Java sur PDA (2)

---

- Comporte quelques lourdeurs pour les PDA
  - Gourmandise en temps processeur
  - Gourmandise en mémoire vive
- Et quelques fonctions peu utiles sur PDA
  - Processus légers
  - API complexes et lourdes
  - Pas d'API spécifiques

# Java sur PDA (3)

---

- Idée d'une machine virtuelle (et d'API) réduites
  - Waba (Newton, Palm, WinCE, MS-DOS)
  - MIDP [ex. KVM] (PalmOS) de Sun
  - PersonalJava (Zaurus, WinCE) de Sun
  - IBM J9 VM (Palm)
  - Kada VM (Palm)
  - Jeode (Zaurus)
- Plateforme J2ME & JSR 75

# Java sur PDA (4)

---

- Avantage du ramasse-miettes
  - Programmation plus aisée
  - Peu de fuites dans les programmes Java (problème de PalmOS)
  - Un peu plus coûteux en mémoire/temps, ralentissements ponctuels

# Processus de développement (1)

---

- Le développement croisé, typique sur PDAs
  1. Édition
  2. Compilation
  3. Correction des erreurs à la compilation
  4. Retour à l'étape 1
  5. Génération du paquet
  6. Installation
  7. Test
  8. Retour à l'étape 1

# Processus de développement (2)

---

- Avantages des émulateurs:
  - Rapidité d'installation des paquets
  - Développement sur une seule machine
  - Mode déverminage
- Avantages de Java:
  - Exécution sur la machine hôte
- Inconvénients:
  - Transparence de la gestion de la mémoire
  - Gestion de l'interface graphique

# Processus de développement (3)

---

- Le développement croisé avec Waba
  - Edition: n'importe quel éditeur (e.g. vim, emacs, NetBeans, CodeWarrior)
  - Compilation: javac, jikes ou autre
  - Génération du paquet
    - Warp (.exe ou classe Java)
    - Exegen (idem)
  - Installation: dépend du PDA

# Conception d'applications sur PDA

## L'exemple de Waba

# Conception d'applications

---

- Adéquation de Waba
- Interface graphique
- Le modèle événementiel
- Le stockage des données
- Interfaces entrées/sorties

# Adéquation de Waba (1)

---

- Machine virtuelle d'abord conçue pour le Palm (Interface graphique, utilisation du stylet)
- API spécifiques pour les applications sur PDA (catalogues, son)
- API simplifiées (pas de gestionnaires, menus, layouts)
- Machine virtuelle réduite (pas de processus légers)

# Adéquation de Waba (2)

---

- Gestion efficace de la mémoire de Waba en différentes zones
  - Object heap: mémoire pour les objets Java, ramasse-miettes
  - Class heap: table pour les classes, pas de ramasse-miettes
  - Pile Java (1 Ko)
  - Pile Machine virtuelle (300 octets)
- Projet Open Source

# L'interface graphique (1)

---

- Une structure inspirée de AWT
  - Une fenêtre principale (et unique) pour l'application: `MainWindow`
  - La structure de Waba: des composants (classe `Control` et sous-classes) et des conteneurs (classe `Container`)
  - Un élément de base: les étiquettes (`Label`)

# L'interface graphique (2)

---

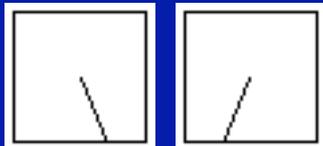
- L'ajout d'éléments se fait en général en précisant les coordonnées
- Pas de Layout, mais:
  - RelativeContainer (précise la position relative d'un objet par rapport au précédent)
  - GridContainer

# L'interface graphique (3)

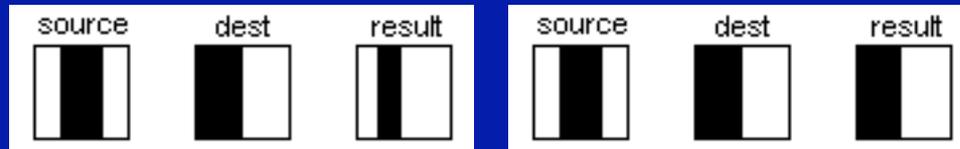
---

- Dessin
  - Méthode onPaint appelée par la machine virtuelle
  - Texte, lignes, couleur, images, etc.
  - Attention aux bogues de PalmOS

Clip



AND



# Le modèle événementiel (1)

---

- Boucle des événements gérée par l'OS
  - Élément essentiel des applications sur PDA
  - Permet des économies d'énergie
  - Incite à la coopération entre les applications

# Le modèle événementiel (2)

---

- Avec Waba
  - Méthode onEvent à surcharger
  - Plusieurs types d'événements prédéfinis:
    - PenEvent & KeyEvent (viennent de la machine virtuelle)
    - ControlEvent: entre les objets Waba
  - Possibilité d'ajouter des événements

# Le stockage des données (1)

---

- Importance des données sur PDA
  - Rôle des PDA
  - Type des données: enregistrement dans des tables (carnet d'adresses, agenda, notes)
- Les catalogues
  - Spécificité des PDAs: un catalogue par type de données/application
  - Synchronisation avec les ordinateurs de bureau
  - Accès depuis n'importe quelle application: intégration des données entre les programmes

# Le stockage des données (2)

---

- Les catalogues dans Waba
  - Ouverture/création/suppression de catalogue
  - Ajout/suppression d'enregistrements
  - Accès octet par octet
  - Fermeture avec le ramasse-miettes

# L'échange des données

---

- Classes Socket et SerialPort
  - Lecture synchrone: grande simplicité
  - Difficultés pour le port série
    - Ajout de caractères sur Palm
    - Problèmes de l'accès synchrone sur Newton
  - Limitations pour les sockets TCP/IP
    - Synchrones: 1 seul échange de données à la fois
    - Pas de serveur possible

# Conclusion

# Conclusion (1)

---

- Enjeux
  - Particularité du développement d'applications sur PDA
    - Modèle événementiel
    - Stockage des données sur catalogues
  - Intérêt de Java sur PDA
    - Nécessité de machines virtuelles adaptées et d'API spécifiques

# Conclusion (2)

---

- Limites du modèle d'application
  - Programmation systèmes sur PDA (spécifique à une plateforme, impossibilité d'utiliser des langages interprétés, utilisation réduite des émulateurs)
  - Applications non orientées données (jeux)
  - Spécificité des systèmes
  - Intégration avec les autres programmes
  - Autres développements: conduits (intégration avec les ordinateurs de bureau)

# Conclusion (3)

---

- Perspectives
  - Marché des PDA divisé entre PocketPC et Palm: supériorité des solutions multiplateformes
  - Evolution vers des PDA de plus en plus puissants: changement de références (J2ME & JSR 75)