PHILIPS

Presenting iPronto

Wireless Home Control Panel
Market trend: convergence of domains

Entertainment Consumption

PC & Internet

Control & Navigation
Market trend: convergence of domains

Entertainment Consumption

PC & Internet

Connected home

Control & Navigation
iPronto – Dashboard for the digital home

**Product Form**
- Portability vs Readability optimum

**Product Fit**
- High frequency of use
- Short interval of use
- Mobile through the home

**Product Functions**
- Control
- Navigation (Content)
- Infotainment

Device Control … Media Control … Home Control
User interfaces and hardware

- Touchscreen LCD
- Touchstrip
- Stylus
- Microphone
- Scrollwheel
- Speakers
- Hard buttons

- Intel Xscale 400 MHz processor
- Linux / Java / OSGi application framework
- WiFi connectivity
- 4 hours Li-ion Battery
- MMC/SD card slot
- USB connector

Optimized for Portability & Readability
iPronto – Applications

Remote Control
*Full customisation and personalisation*

Electronic Program Guide
*with channel switching*

Instant information
*through a browser*
Dashboard - 8 Layouts
iPronto Partner Platform Strategy
Philips Go To Market Model

Complete Solution

ARM Processor, e.g., Intel® PXA250

Philips Brand

OEM Brands
PRONTO ++ Platform Solution Components

Complete Solution

- Hardware
- Software
- Services

Community
PRONTO ++ Platform Components

- **Modular HW Architecture**
- **Modular & Portable SW Framework**
- **Web Service Infrastructure**

**Community (Users, Developers & Partners)**
PRONTO ++ Modular HW Architecture

• **ARM Based Processor**
  - Currently Intel X-Scale Architecture
  - Other processors possible

• **HW Options (Highlights but not complete)**
  - 64 Mb RAM and Flash memory
  - LCD displays
    - 2 – 10”
  - Connectivity solutions
    - 802.11b/a, Bluetooth, GPRS, DVB-T, etc.
  - Standard connector solutions
    - PCMCIA, Compact Flash, MMC/SD
    - USB, 1394, etc.
  - Hard Disk interface for Gigabytes of Storage
  - Dedicated HW for Video Processing
  - etc.

• **Expandable by End User**
  - Accessories
PRONTO ++ SW Framework

• **Complete software stack**
  - Hardware (processor) independent
  - Operating system agnostic

• **Core Applications provided by Philips**
  - Navigation Dashboard UI
  - Remote Control
  - Electronic Program Guide
  - Browser
  - Audio & Video Players
  - Picture Viewer
  - Java Games
  - Maps & Navigation
  - Other Applications

• **Software development kit**
  - to develop customer applications
PRONTO ++ Web Service Infrastructure

• **Service Infrastructure**
  - Web portal interface
  - Machine – Machine interface

• **Basic Services**
  - User and Device Registration
  - Software Upgrade Management
  - Service Subscription Management
  - New Application Downloads
  - Service Fee Billing

• **Application Service Examples**
  - Electronic Program Guide
  - Music & Video databases, etc.
iPronto OEM/Partner model

“Co-Marketing”

Philips/OEM channel
OEM Applications

Philips Branded iPronto

OEM channel
OEM Applications

OEM Branded product

“True OEM”

iPronto platform: Hardware, Mechanics, Application Framework and Applications
iPronto – SW architecture overview
iPronto - implementation

• Linux/pJava-based embedded system
• Application management: OSGi v2
  • Technology provider

• Why OSGi?
  • Technology:
    – Component-based model & lifetime management
    – Dynamic execution environment
    – Network protocol neutral
    – Secure
    – Remote Management
    – SW deployment (very useful also for debugging)
    – Interoperability

• Strategic
  – Pure Java
  – Promising industry standard
  – Multiple vendors
  – Multiple service provisioning solutions
iPronto – SW architecture overview

Application 1
- <<bundle>> Remote Control
- <<bundle>> IR
- Java Security

Application 2
- <<bundle>> Browser
- <<bundle>> Application Framework

Application 3
- <<bundle>> EPG
- <<bundle>> EPG / Smartsurf

Application N
- <<bundle>> UPnP A/V Contol Point
- <<bundle>> Preferences

OSGi Framework
- Util
- System Interface
- Boot
- SBM

Portability boundary

Java codec or wrapper (JNI)
intent JTE
Java driver wrapper (JNI)

Codec
Linux
Driver

TurboBoot
Intel Cotulla
Device
Current demo implementation

• OSGi used for application framework:
  – Starting/stopping apps
  – Upgrading in development environment

• Prototype:
  – Includes management agent on device
  – Connects to provisioning system (backend)
  – Current use cases:
    • Apps (IM, weather, games) pull by customer
    • Multiple users
    • Subscribe/activate

• Demo!
Q&A

For further info, please contact:
Frederik Leemans
frederik.leemans@philips.com
+32 16 394557