Building an embedded software IDE on top of Eclipse

Gaétan Morice - Anyware Tecnologies
David Pochet - Wavecom
June 25th, 2009
Part 1 : Context
   What we had to do.

Part 2 : M2M Studio
   How we use Eclipse technologies.

Part 3 : Feedback
   What we learned.

Part 4 : Ideas
   Some though on future solutions.
Part 1: Context
What we had to do.

Part 2: M2M Studio
How we use Eclipse technologies.

Part 3: Feedback
What we learned.

Part 4: Ideas
Some though on future solutions.
Wavecom:

“Wavecom is a leading provider of embedded wireless technology for M2M communication.”

Wavecom provide modems

- GSM, Edge, 3G, Satellite, ...

But what exactly is M2M?
• **Machine to Machine**

• **A definition**: “M2M refers to data communication between machines.”

• **Principle**:  

![Diagram showing the principle of M2M communication](image_url)
• Wavecom products may be used as **bit pipes**

![Diagram showing Asset, Device, Connectivity, Network, Modem, AT Commands, SMS, GSM, 3G, Embedded App, Data Processing, Data Storage, Bearing Strategy]

• But can also **embed** business solutions
How **embedded** solution works?
• **ARM Based**
• Different families and **packagings**
  - WMP, Quick, Fastrack, ...
• **Wireless** features
  - GSM, GPRS, EDGE, 3G, ...
• **GPIO** for connectivity
• **Optional** features
  - GPS, USB, Ethernet, Bluetooth, ...
• Several **memory** configuration
• Some provide **JTAG** debug
• **Raw API for**
  - Communication management
  - Memory access
  - Download & update
  - IO access

• **Services** for debug purpose
  - Traces
  - Dump
  - Remote Call
  - Memory monitoring
  - Process monitoring
• Provided by Wavecom
  ► C libraries
  ► High level APIs
  ► Optional hardware management
    – GPS, USB, ...
  ► Utilities
    – TCP/IP stack, encryption, ...

• Custom library
Application

- Provided by Wavecom
  - Special Features
    - CAN Bus, ...
  - Code samples
- Custom application
Tool Chains

- Cross compilation
  - GCC
  - ARM specific compiler
  - Binaries packaging to download

- On host
  - MinGW
• **Complex** environment with:
  ► Specific life cycles
  ► Dependencies rules
  ► Own documentation
  ► Binaries provisioning
  ► External tools set up

• An **IDE** is needed
Current tooling

• What was **provided:**
  ► A bunch of makefile templates
  ► Use of Cygwin
  ► Some basic CDT facilities
  ► Tools for download and target services

• **Problems:**
  ► Install & update (several different pieces)
  ► Maintain makefiles
  ► Versioning of binaries
  ► Newcomers
  ► Synchronisation between tools (error marker, download, ...)

• **A new tool was needed**
Requirements

• **Ease of Use**
  ► User assistance in complexity management

• **Install & update**
  ► Management of the ide, binaries, tools, ...

• **Extendable**
  ► Possibility to add new features

• **Integrated**
  ► Code, compile, download, debug in the same tool
Part 1: Context  
*What we had to do.*

Part 2: M2M Studio  
*How we use Eclipse technologies.*

Part 3: Feedback  
*What we learned.*

Part 4: Ideas  
*Some though on future solutions.*
Need: User Assistance

M2M Studio

User Assistance

- Cheatsheets
- Contextual Help
- Welcome page

Eclipse RCP

- Core / UI
Need: Install / Update

M2M Studio

User Assistance

Cheatsheets  Contextual Help  Welcome page  Update

Eclipse RCP

Core / UI  p2
Need: Domain Specific Model

M2M Studio

User Assistance

Domain Specific Model

Elements
Versions
Compatibilities
Extendable

Eclipse RCP

Core / UI
p2
EMF
Need: Binaries Provisioning

M2M Studio

User Assistance

Packet Manager

Firmwares

Libraries

Tools

Documentation

Import

Domain Specific Model

Repository (File System)

Eclipse RCP

Core / UI  p2  EMF
Need: Project Management

M2M Studio

User Assistance

Packet Manager

Project Manager

Creation

Edition

Dependencies

Domain Specific Model

Eclipse RCP

Core / UI  p2  EMF  CDT
Need: Compilation

M2M Studio

User Assistance

Packet Manager

Project Manager

Creation

Edition

Dependencies

Compilation

Domain Specific Model

Eclipse RCP

Core / UI  p2  EMF  CDT  MBS
Need: Download

M2M Studio

User Assistance

Packet Manager

Project Manager

Target Manager

Launch Config

Download

Domain Specific Model

Eclipse RCP

Core / UI  p2  EMF  CDT  MBS  RSE

Target
Need : Run

M2M Studio

User Assistance

Packet Manager
Project Manager
Target Manager

Launch Config
Download
Terminal
Trace Log

Domain Specific Model

Target

Eclipse RCP

Core / UI  p2  EMF  CDT  MBS  RSE
Need: Debug

M2M Studio

User Assistance

Packet Manager

Project Manager

Creation
Edition
Dependencies
Compilation
Debug

Target Manager

Target
JTAG Probe
Driver
Open OCD
GDB Server

Domain Specific Model

Eclipse RCP

Core / UI  p2  EMF  CDT  MBS  RSE

© Anyware Technologies-All Right Reserved
Part 1: Context
   What we had to do.

Part 2: M2M Studio
   How we use Eclipse technologies.

Part 3: Feedback
   What we learned.

Part 4: Ideas
   Some though on future solutions.
Pros

► EMF domain specific modelling
  - Manage complexity
  - Re-factoring
  - Data centralisation

► CDT complete and extensible solutions
  - Configuration system
  - AST features
  - MBS possibilities
  - Debug
Pros

► RSE target and service management
  - Distinction Service / Connector
  - Model coupling
  - Ease of use

► P2 installer and update
  - Features provided
Cons

► Ergonomics
  - Set up a real ease of use policy

► Conservative user
  - Difficult to set CDT and RSE paradigms

► CDT
  - Complex project management (due to resources problems)
  - Project dependencies
  - Makefile generation
  - GUI customisation (new project wizard)
  - API documentation
Cons

- RSE
  - UI / Core de-coupling
  - TCP/IP orientation

- p2
  - Difficult to set up
  - Documentation
  - maturity
Part 1 : Context
   What we had to do.

Part 2 : M2M Studio
   How we use Eclipse technologies.

Part 3 : Feedback
   What we learned.

Part 4 : Ideas
   Some though on future solutions.
- CDT 6 and future
  - DSF and TCF integration in debug
  - MBS and models improvement

- TCF
  - Embedded agent for services
  - Multiplexing over channel

- RTSC
  - Manage binaries package with lots of meta-data

- D-Pack
  - Packaging solutions for embedded solution development
• OSCi Services
  ▶ More dynamic Architecture

• E4
  ▶ New resource system
  ▶ New declarative GUI