



# Some ideas for the TmL project

A very early brainstorming

31 July 2006

Motorola TmL Team



- Which direction to take TmL
- What are the gaps currently **not** addressed by other Eclipse projects like DSDP, CDT etc.
- Ideas for future prioritization



TmL environment = Eclipse + TmL plugins

➔ TmL perspective

➔ a TmL plugin can connect to one of:

- ✓ a phone or wingboard connected via USB
- ✓ an emulator running on the same computer
- ✓ an emulator running on another computer



- Provide a runtime environment for native phone applications
  - Simulate Phone “on board” devices
    - Multimedia
    - Location
    - Communication
  - Connect simulated devices to Eclipse IDE
  - Allow to adjust phone specific parameters
    - Total memory
    - File system
    - Skinning



## Create and manage *Mobile Linux Devices*

- ➔ similar to a Java or C++ project
- ➔ can be a phone/wingboard or an emulator
- ➔ menu option: *Project* → *New* → *Mobile Linux Device*
- ➔ properties: host, IP address, port
- ➔ menu option: *Connect*
- ➔ menu option: *Disconnect*
- ➔ menu option: *Delete*



## TmL plugins

- ➔ Mobile Linux Device *Navigator* view
  - ✓ displays all Mobile Linux Devices
  - ✓ use colours to distinguish different devices
- ➔ Mobile Linux *Filesystem* view
  - ✓ displays the Linux filesystem on the device
- ➔ *Shell/Console* view
  - ✓ contains a Linux shell/console



## TmL plugins

### ➔ *Explorer* view

- ✓ displays the filesystem content in Windows Explorer format

### ➔ *Flash* view

- ✓ displays information to flash an image file into the device
- ✓ progress bar

### ➔ *Application* view

- ✓ upload, run and debug applications



## TmL plugins

### ➔ *Skimming* view

- ✓ upload and change themes and skins

### ➔ *Virtual Mobile Server* view

- ✓ “virtual mobile exchange”
- ✓ “virtual SMS server”
- ✓ “virtual web server”

