Mer: Reconstructing Maemo

An overview of the Mer project

by Carsten Valdemar Munk, lead developer

Outline for this talk

- A call to reconstruct Maemo
- The project and our development method
- Working with hardware vendors
- Accomplishments and Mer 1.0

October 2008: A call to reconstruct Maemo

Tablets are not under-powered embedded systems, they are powerful, power-efficient, economical handheld computers.

Make Maemo a general platform for tablet devices.

Make it more developer-friendly.
More hackable.
Align with standard Linux distributions.

October 2008: A call to reconstruct Maemo (continued)

Separate device and platform code

Open development of the Maemo platform - the device-specific and vendor-specific differentiation development can be closed.

It should be easy to port existing desktop applications - platform peculiarities should be kept to the absolute minimum required for the mobile use-case.

The Mer project & development method

Mer is a maemo.org project, owned by the community, created by the community.

Currently SCRUM-like sprints consisting of 3 weeks of development and 1 week of testing with a release after each sprint

Open development, encouraging team awareness through microblogging

Flat management structure of volunteer mentors. Currently 34 people participating off-and-on.

Mer: What is it?

Mer is essentially a bunch of Debian packages and image generation scripts implementing a Maemo platform based on either Ubuntu or Debian.

The core Mer system is fully open source and cross-platform, open for anyone to adapt to their devices.

Mer is also meant as a research operating system for maemo.org to give the community a playground to develop concepts that could potentially be integrated into Maemo.

Mer is not meant as a competitor to Maemo – there's no Mer without Maemo.

Mer: Comparison with desktop distributions

Mer like Maemo focuses on power saving. A mobile device should be able to function several days always-connected – not 4 hours like a laptop.

A tablet has storage space constraints so there might be a need to shrink packages/dependancies, but not to the level of embedded systems (it's not a 64mb NAND flash system)

Like Maemo, Mer focuses on that a tablet has CPU, memory and bandwidth constraints – but it can be more powerful because of the environment surrounding it.

The Vendor Social Contract

A social contract for the vendors wanting to adapt Mer on their hardware.

Principles for the interaction between the Mer project and hardware vendors.

Modelled on the open attitude by Nokia with regards to the Internet Tablets.

wiki.maemo.org/Mer/Documentation/Vendor_Social_Contract

The Vendor Social Contract: No tivoization

No tivoization (Tivoization is the creation of a system that incorporates software under the terms of a copyleft software license, but uses hardware to prevent users from running modified versions of the software on that hardware.)

No requirement that firmware has to be signed

Publishing information on bootloader and firmware flashers

We avoid scenarios like open system but closed devices (Android on a G1)

VSC: Open source kernel modules

Open source kernel modules: Avoiding the ”stuck to a specific kernel version” problem, making it possible to maintain device support after HW vendor is no longer supporting it.

The advantage for the vendor is that community members can help resolve lingering software issues, beta-test potential improvements, and even continue development for unsupported devices.

Advantages and methods of the vendor social contract

It's hard to get around the binary blobs for hardware support, even on X86

Vendors are encouraged to work with maemo.org to provide repositories containing closed firmware and bits, intended only for specific device types

Based on your ownership of a device you would be able to download images containing these bits through maemo.org or construct your own at home using your token/device ownership

Accomplishments: Where are we

- Easy development for the platform
  - You can develop -and- test your Mer/Maemo platform application straight on your PC
  - Cross-build for your tablet, your Wii or your fridge with osc build <platform> yourpackage.dsc

Accomplishments: Where are we (2)

Easy porting:

Hardware support and Mer is separated

Made Hildon desktop a session in Ubuntu

Several patches made towards Maemo packages (Scratchboxisms, Maemoisms, Libtool, etc.)

The power of Ubuntu (or Debian) together with Maemo (based on Maemo 5.0 sources)

And in addition to that:

- Ubuntu MID will be working with us to get our packages into shape and Ubuntu MID in Karmic will be based upon Mer.

Mer 1.0: We're not quite there yet, but getting there.

Mer 1.0 is meant to be the version that is usable day to day on Nokia N8x0s.

We're lacking in the following areas:

- Control panels (Sounds, Language & Region, etc.)
- Hildon input method plugins (in GTK)
  - (keyboard layouts, thumb keyboard, etc)
- Full english localization
- Hildonized NetworkManager
- Fixing UI glitches (mostly GTK and artwork/icons)
- Distribution rights towards N8x0 tablets of HW support
- Web-browser (Currently Midori, will be webkit-based Tear)

Where has Mer been ported to?

Nokia 770

Nokia N800

Nokia N810

Fujitsu-Siemens Loox 720

Neo Freerunner

SmartQ 5

Beagleboard

Nintendo Wii

Sharp Zaurus

vmware

VMWare

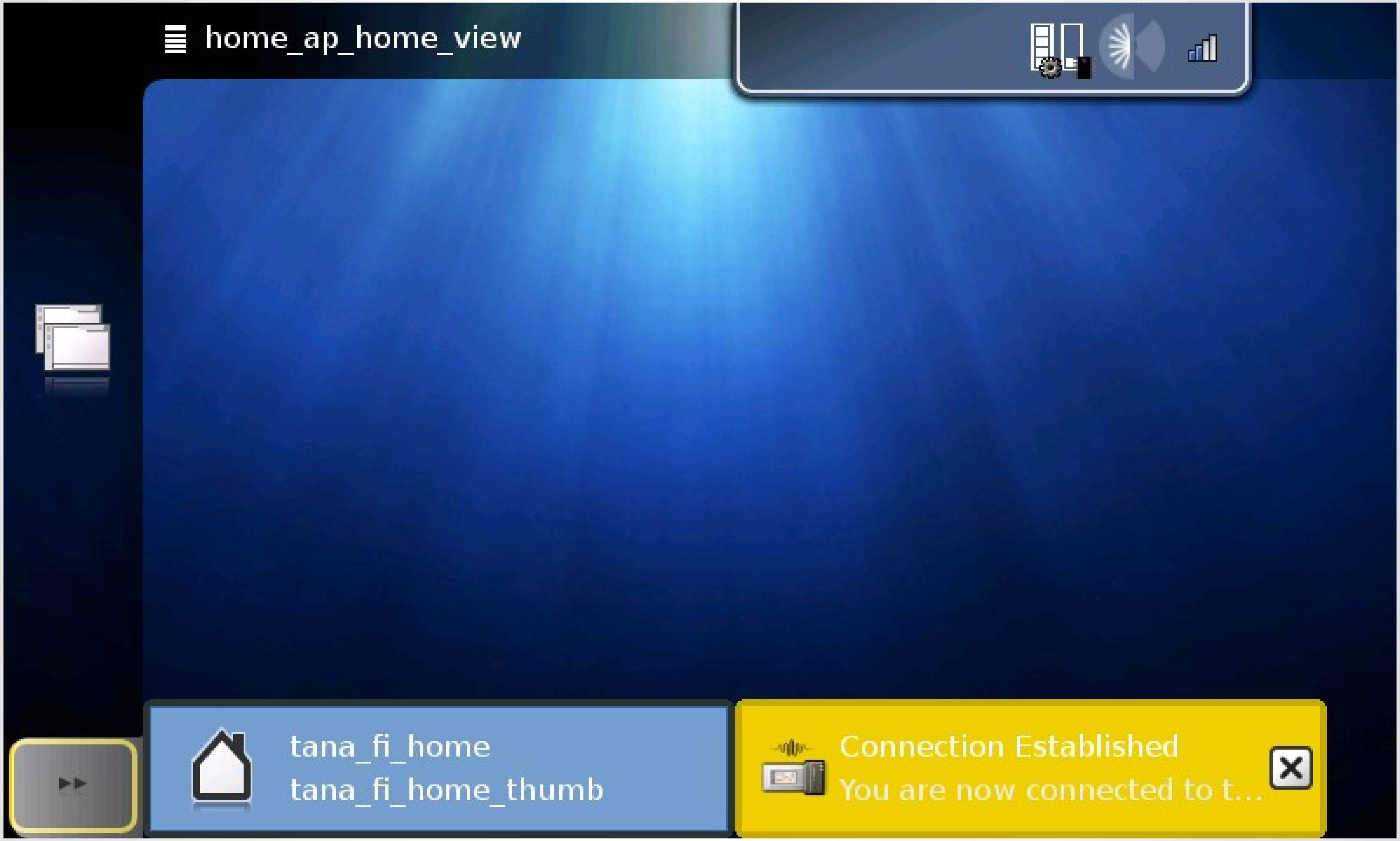
Sun Virtualbox

QEMU
open source processor emulator

QEmu

x86 PC

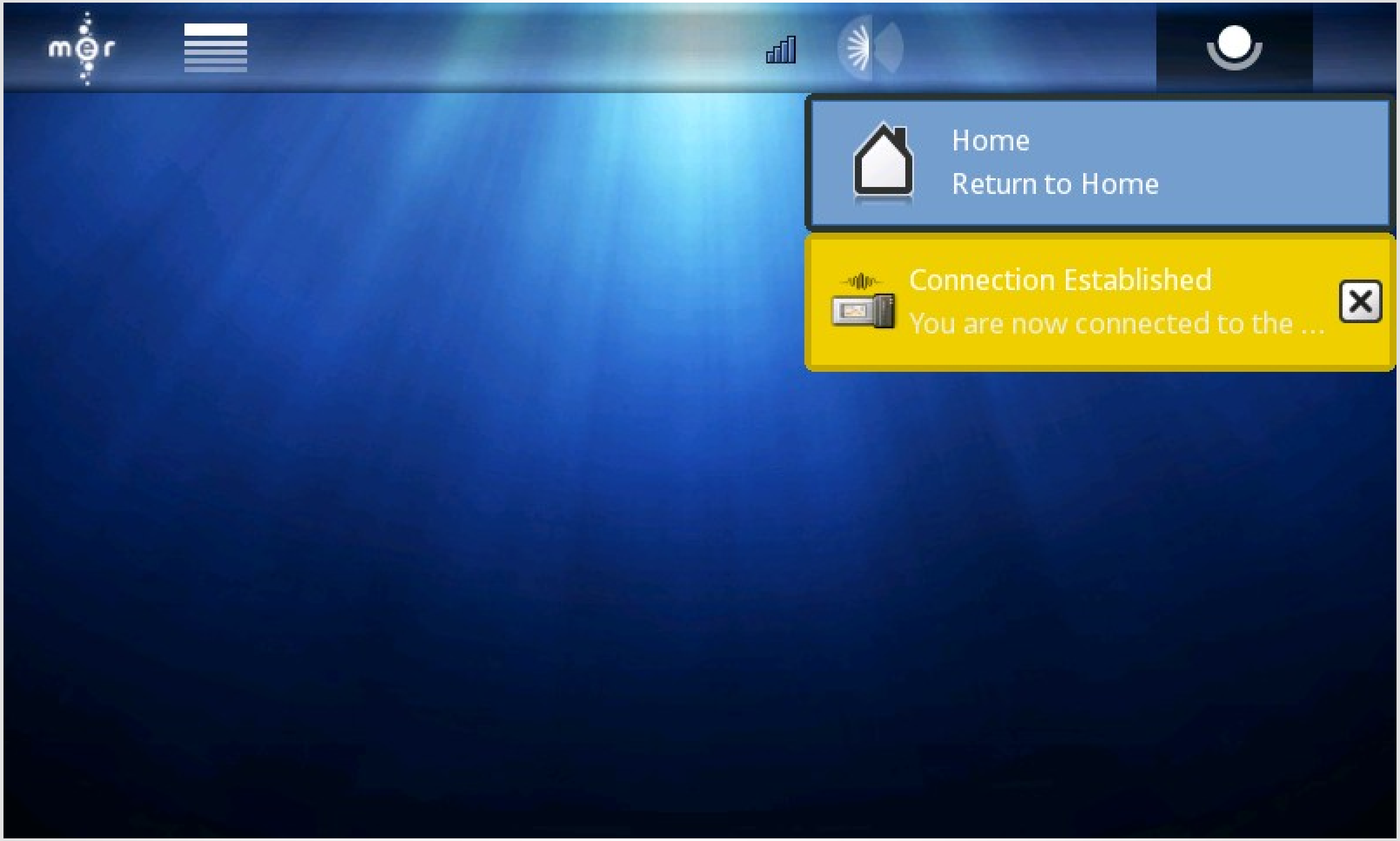
Mer 0.6 -> Mer 0.13



tana_fi_home
tana_fi_home_thumb

Connection Established
You are now connected to t...

Mer 0.13



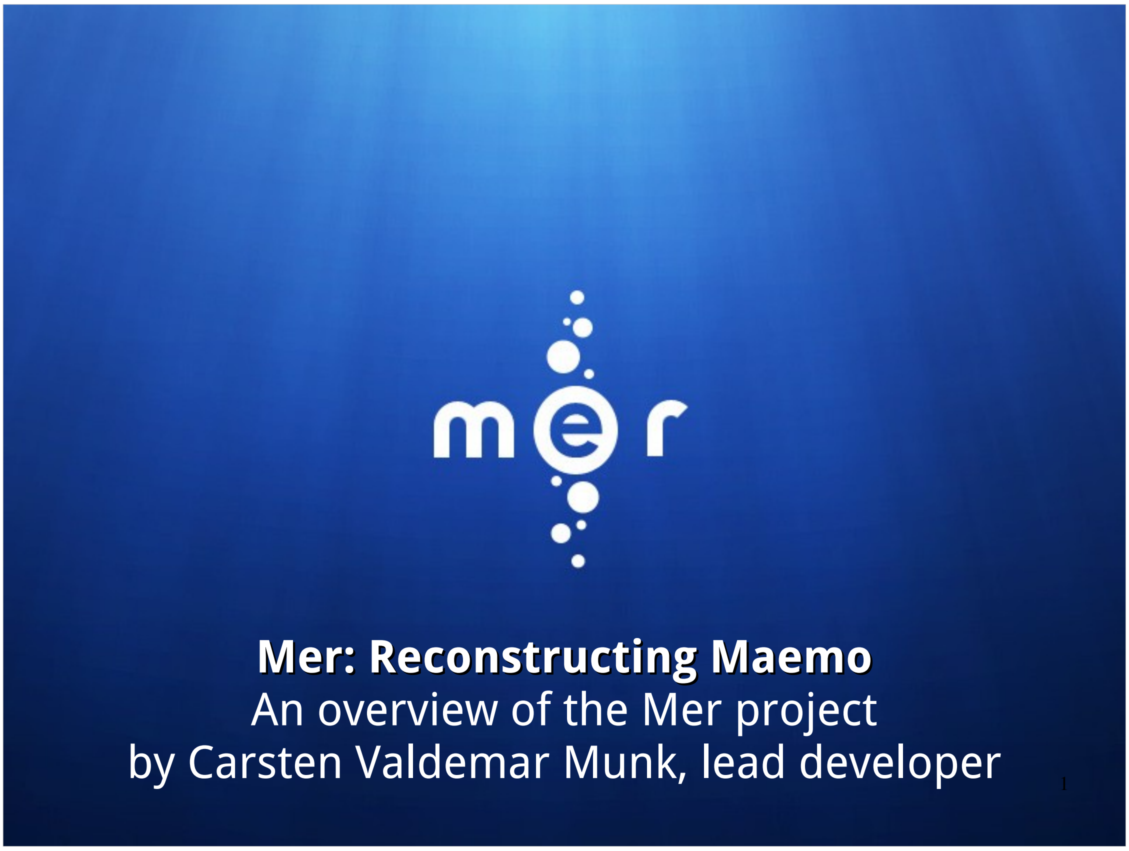
Home
Return to Home

Connection Established
You are now connected to the ...

Questions?

For more information

- http://wiki.maemo.org/Mer
- #mer on irc.freenode.net, IRC
- Or come and talk to us in the hall :)
- David Greaves will be giving a talk on developing for Mer using OpenSUSE Build Service later.
- maemo.org room sessions this weekend:
  - How do we get to Mer 1.0?
  - Mer after 1.0
  - Hands on with OBS & Mer development

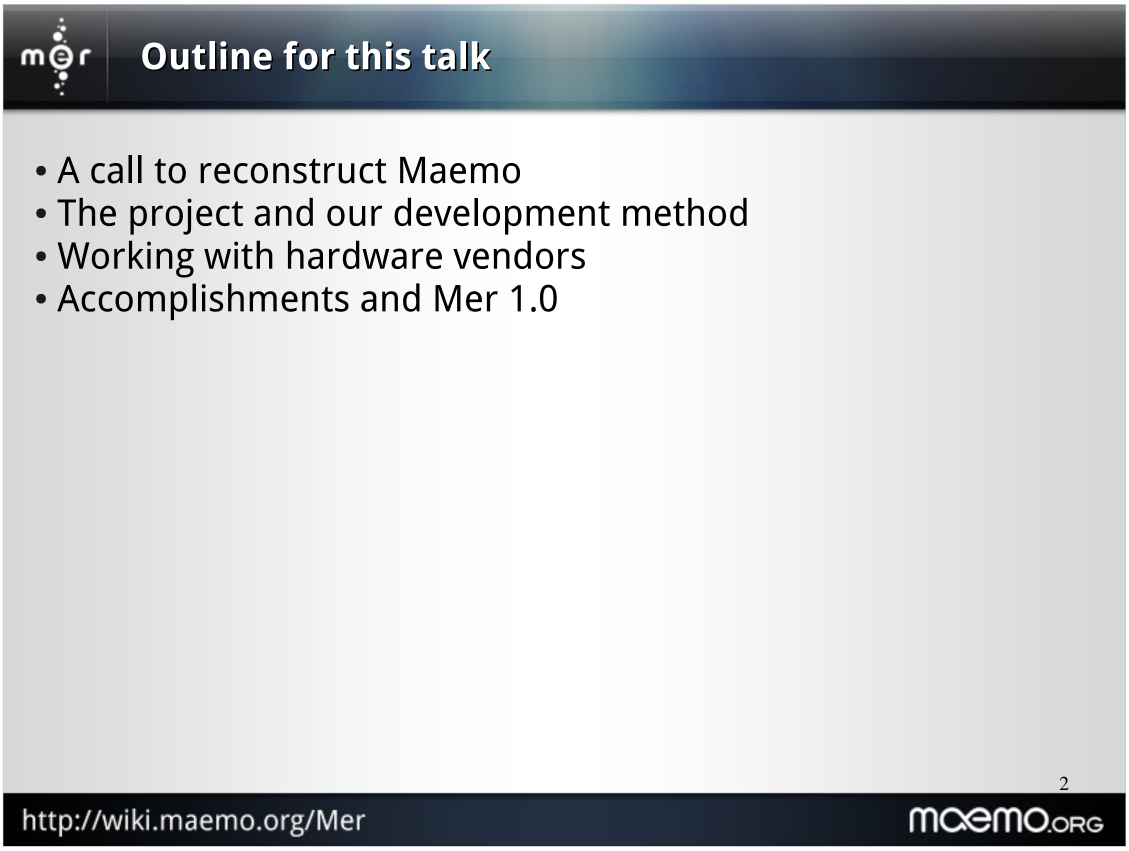


Mer: Reconstructing Maemo

An overview of the Mer project

by Carsten Valdemar Munk, lead developer

So, my name is Carsten Munk, I'm one of the lead developers in the Mer project and I'll be giving a talk on the Mer project called Reconstructing Maemo.



Outline for this talk

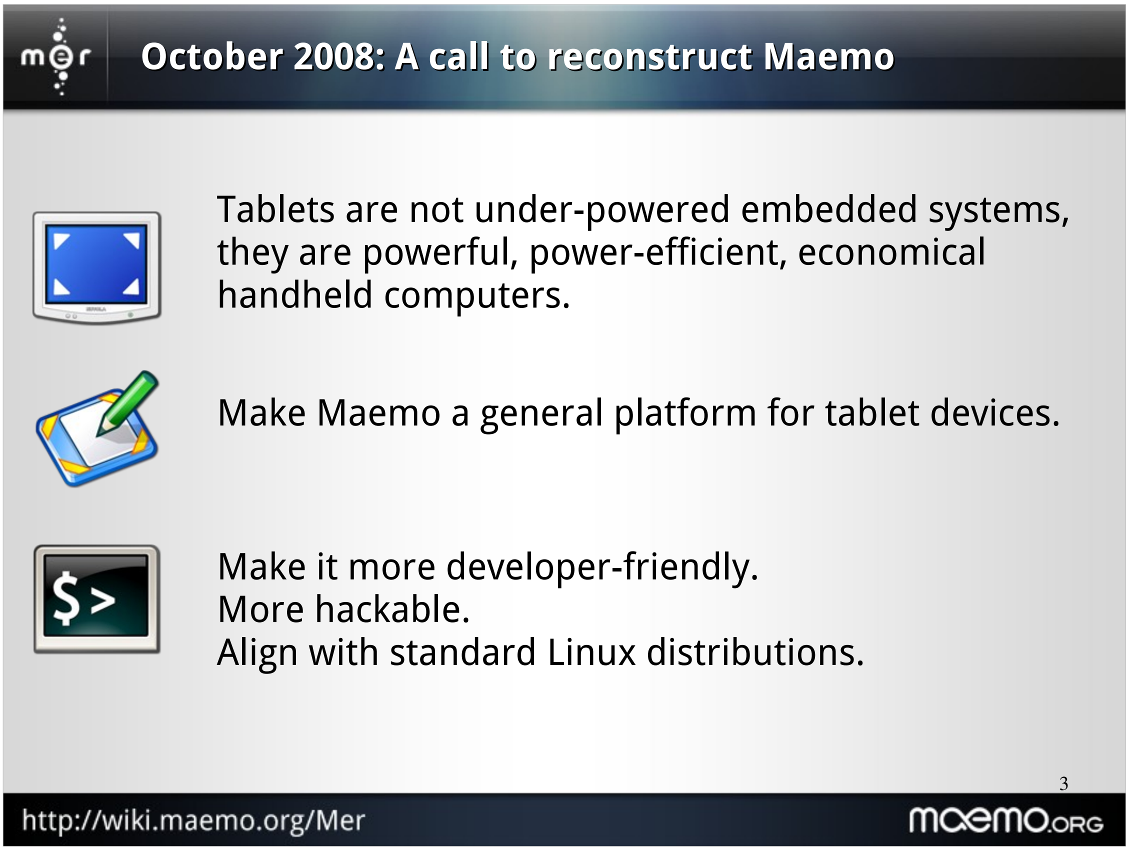
- A call to reconstruct Maemo
- The project and our development method
- Working with hardware vendors
- Accomplishments and Mer 1.0

2

http://wiki.maemo.org/Mer

What you will be hearing in this talk is first “A call to reconstruct Maemo” - some history on how the project came about and what goals we have.

Then i'll be talking more in specifics about how the project is structured, how we collaborate and some technical information about Mer – along with how we try to work with hardware vendors. Finally, I'll talk some about what Mer 1.0 is supposed to be like and accomplishments for the project – and finally some look at our UI progress.



October 2008: A call to reconstruct Maemo

Tablets are not under-powered embedded systems, they are powerful, power-efficient, economical handheld computers.

Make Maemo a general platform for tablet devices.

Make it more developer-friendly.
More hackable.
Align with standard Linux distributions.

3

http://wiki.maemo.org/Mer

maemo.org

In October 2008 a discussion was sparked with the intent of creating a vision for how Maemo could look if we had the chance to reconstruct it completely. Based on input from the maemo org community 6 principles for reconstruction was put together.

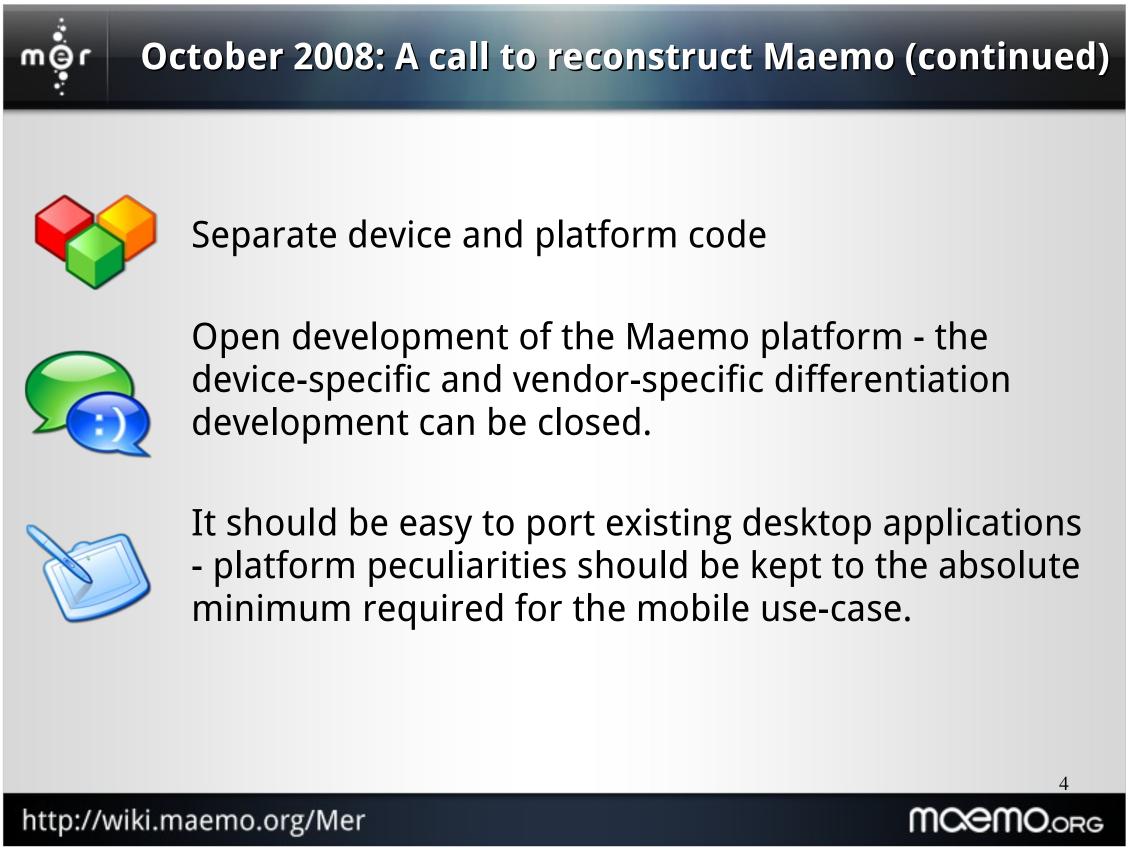
Maemo is harmed by it's history and still suffers from having had to be fit in a sardine can like hardware previously, taking care not to waste any memory, cpu or storage space.

While many of the diffferences to traditional linux distributions still make sense it does not make sense to think of tablets as underpowered embedded systems anymore – they're as powerful as the machines the linux distributions of today began on (Debian, etc).

It is however still necessary to take care of power savings, but hardly as much as on a embedded device.

Maemo is open source – and mature enough to be a strong player in the race to be -the- platform developers target and hw vendors use. Many have tried to adapt Hildon (Debian, Moblin, Ubuntu Mobile) but have hit the wall that the Maemo platform is closely tied to the Nokia Internet Tablets. That has to be changed – multiple platforms, one platform, app for one device, app benefits all devices.

To a traditional open source developer, developing for Maemo can be a headache. You should be able to develop straight on your desktop machine with traditional debian tools, test everything on your desktop machine and then ask a SDK to compile it for a different CPU. Maemo should be more open to hacking – it's a system for your mobile system that's always with you, it should be open to alteration. Maemo has strayed from traditional linux distributions and it's a uphill battle for developers to port simple libraries they know from their desktop surroundings.



October 2008: A call to reconstruct Maemo (continued)

- Separate device and platform code
- Open development of the Maemo platform - the device-specific and vendor-specific differentiation development can be closed.
- It should be easy to port existing desktop applications - platform peculiarities should be kept to the absolute minimum required for the mobile use-case.

http://wiki.maemo.org/Mer

maemo.org

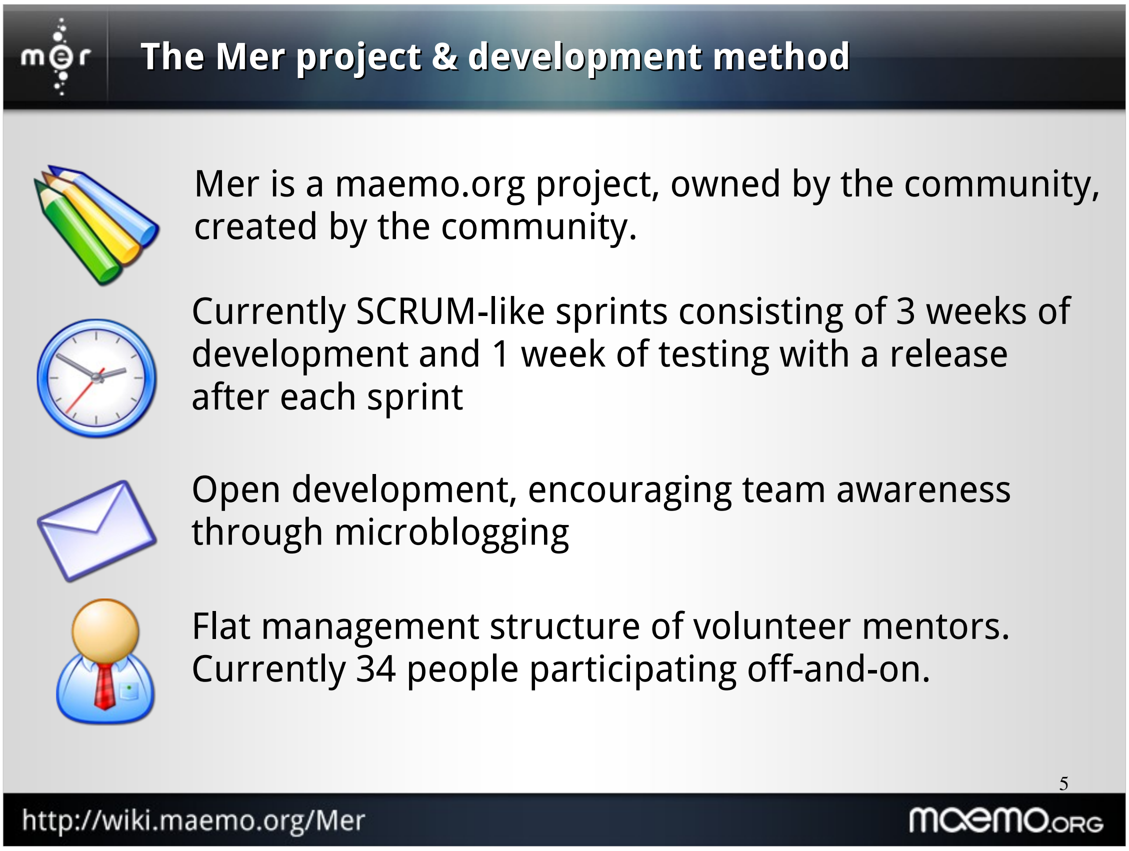
A major part of reconstruction would imply seperating the Nokia Internet Tablet from the Maemo platform – acheiving the possibility to put the Maemo platform on any device that can fit it.

If we are seperating Maemo from the Nokia Internet Tablets – the remaining parts would be device specific or vendor specific differentiation.

Since we are dealing with a bunch of different devices running the Maemo platform, it's only fair that development of the platform is done in the open as to help developers see what is coming and bring input & coders from different device communities.

Hardware support and differentiation is the hardware vendors own business and none of the platforms.

Most existing applications need simple fixes in terms of UI/screen estate and to be well behaved on a mobile platform – and porting to Maemo should be this simple. Instead, it's a uphill battle to be dealing with missing libraries, Scratchbox particularities, Busybox, etc. It should be simpler to port existing desktop applications.



The Mer project & development method

- Mer is a maemo.org project, owned by the community, created by the community.
- Currently SCRUM-like sprints consisting of 3 weeks of development and 1 week of testing with a release after each sprint
- Open development, encouraging team awareness through microblogging
- Flat management structure of volunteer mentors. Currently 34 people participating off-and-on.

5

http://wiki.maemo.org/Mer

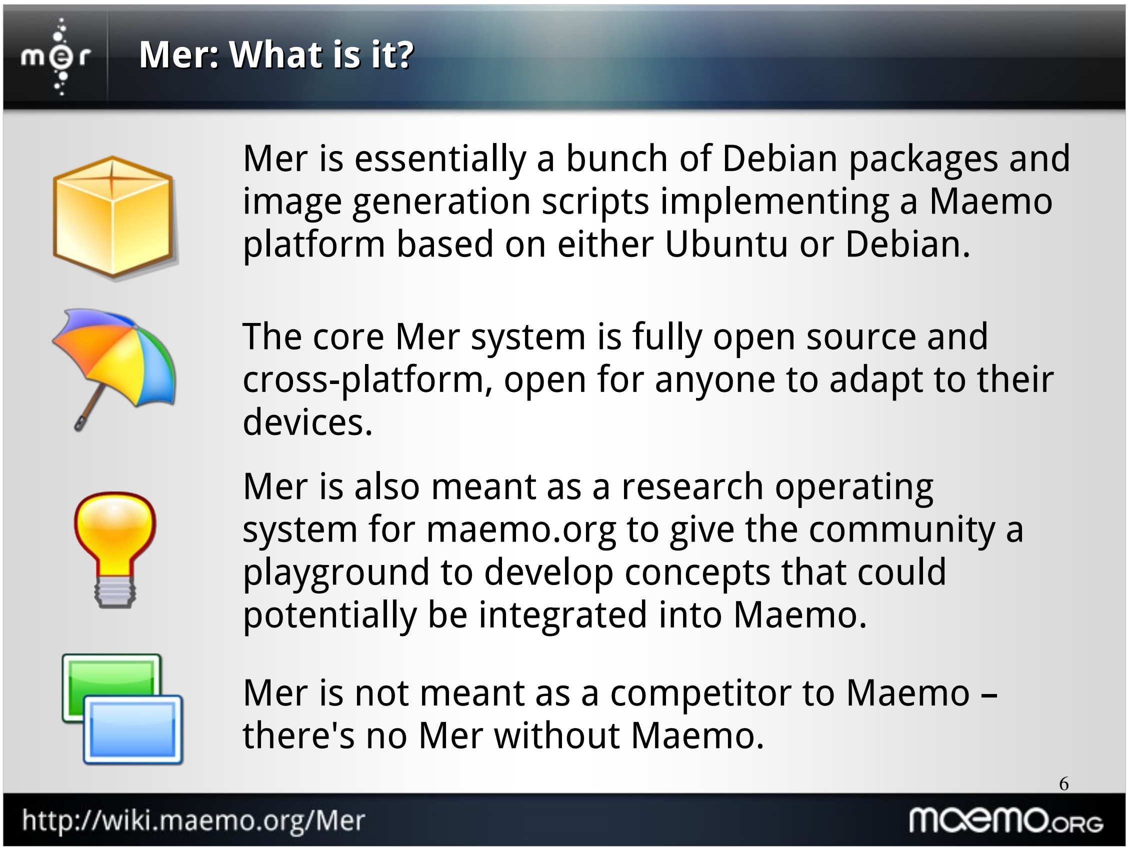
The name “Mer” stems from M-R Proof of Concept which came from Maemo Reconstructed. Hence Mer is the ongoing reconstruction of Maemo.

We consider Mer to be a maemo.org project – we originated within this community. It is owned by the community – there's a lot of small and big contributions.

The way we get things done is through SCRUM like sprints – 3 weeks development and 1 week of testing and a release each month now. People sign up for tasks on a volunteer basis related to their ability.

We run open development, where we use microblogging through short messages to a common mailing list – even accessible through IRC.

We don't have any management as such – people with experience are mentors who help out people to do their tasks within their areas.



Mer: What is it?

Mer is essentially a bunch of Debian packages and image generation scripts implementing a Maemo platform based on either Ubuntu or Debian.

The core Mer system is fully open source and cross-platform, open for anyone to adapt to their devices.

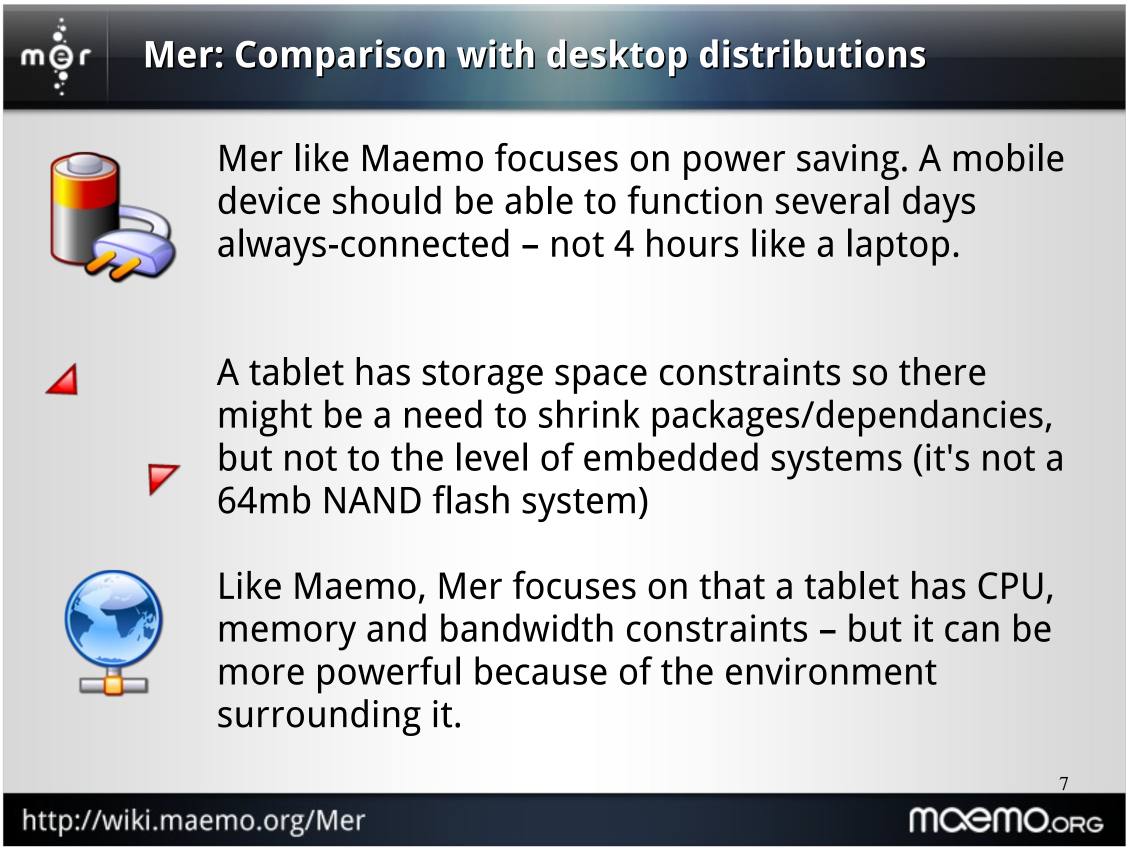
Mer is also meant as a research operating system for maemo.org to give the community a playground to develop concepts that could potentially be integrated into Maemo.

Mer is not meant as a competitor to Maemo – there's no Mer without Maemo.

Like Maemo, Mer is a bunch of Debian packages and image generation scripts. These compile on either Ubuntu or Debian – but the patches are usable in any distribution – and anyone can adapt the code in their distribution or devices.

Mer is also the playground for the maemo.org community. Ideas and concepts that are well tested in Mer could serve as finalized patches to upstream Maemo.

To say it once and for all: Mer isn't a competitor to Maemo. There's no Mer without Maemo and all the hard work done by Maemo SW. Mer is a project to reconstruct Maemo and to create a Maemo-compatible platform.



Mer: Comparison with desktop distributions

Mer like Maemo focuses on power saving. A mobile device should be able to function several days always-connected – not 4 hours like a laptop.

A tablet has storage space constraints so there might be a need to shrink packages/dependancies, but not to the level of embedded systems (it's not a 64mb NAND flash system)

Like Maemo, Mer focuses on that a tablet has CPU, memory and bandwidth constraints – but it can be more powerful because of the environment surrounding it.

7

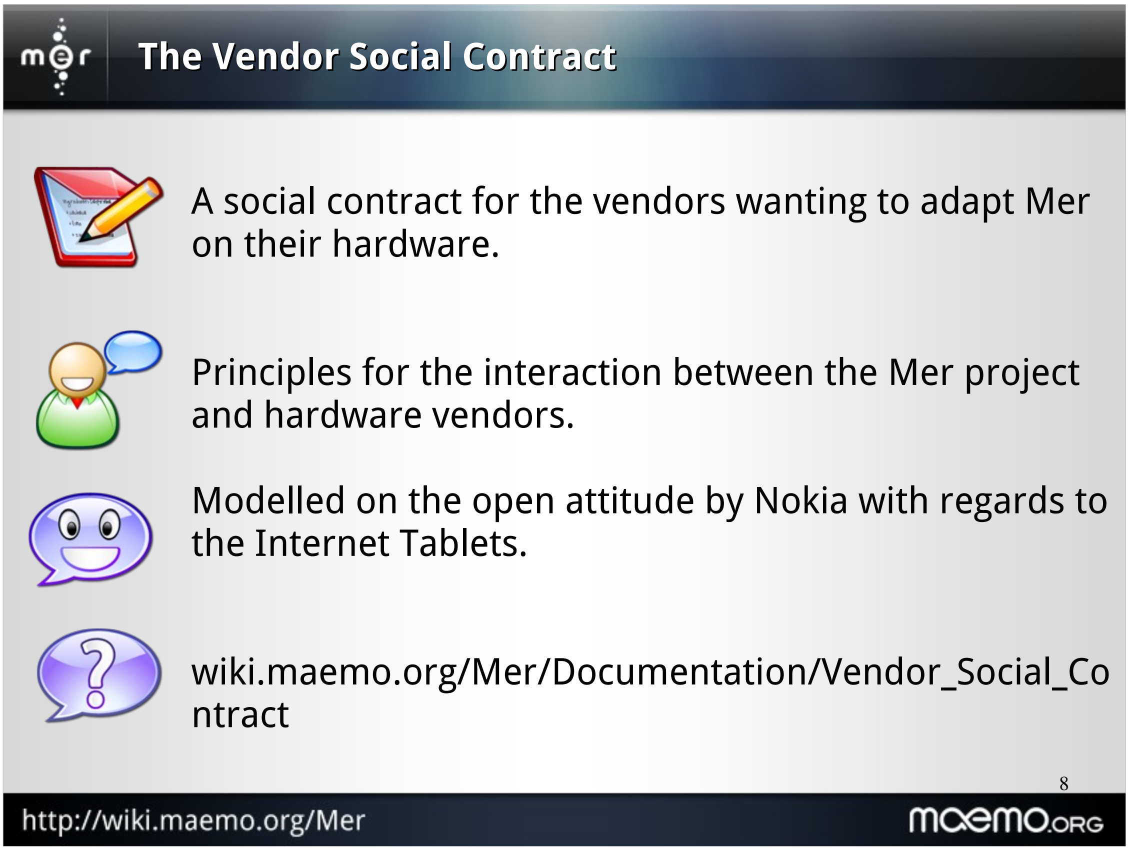
http://wiki.maemo.org/Mer

Is Mer just yet another linux distribution? Many people would disagree since we break quite a lot of things (Maemo GTK is a big one!), and I'll tell you some of the ways we differ.

Like Maemo we focus on power saving – some things that make sense on desktop/laptop distributions simply doesn't make sense on mobile devices that's supposed to last on the same charge several days being always-connected.

A tablet still comes with limited storage space – so there might be a need to shrink packages or dependancies (why do we need a weather library for NetworkManager?) - but not hard as on an embedded system – there's miles difference between 256m flash and 64mb flash.

Yes, you can't run Crysis on a tablet – it has CPU, memory and bandwidth constraints and apps should be suited towards that. But, what it can do is to be more powerful because of the environment and services surrounding it.



The Vendor Social Contract

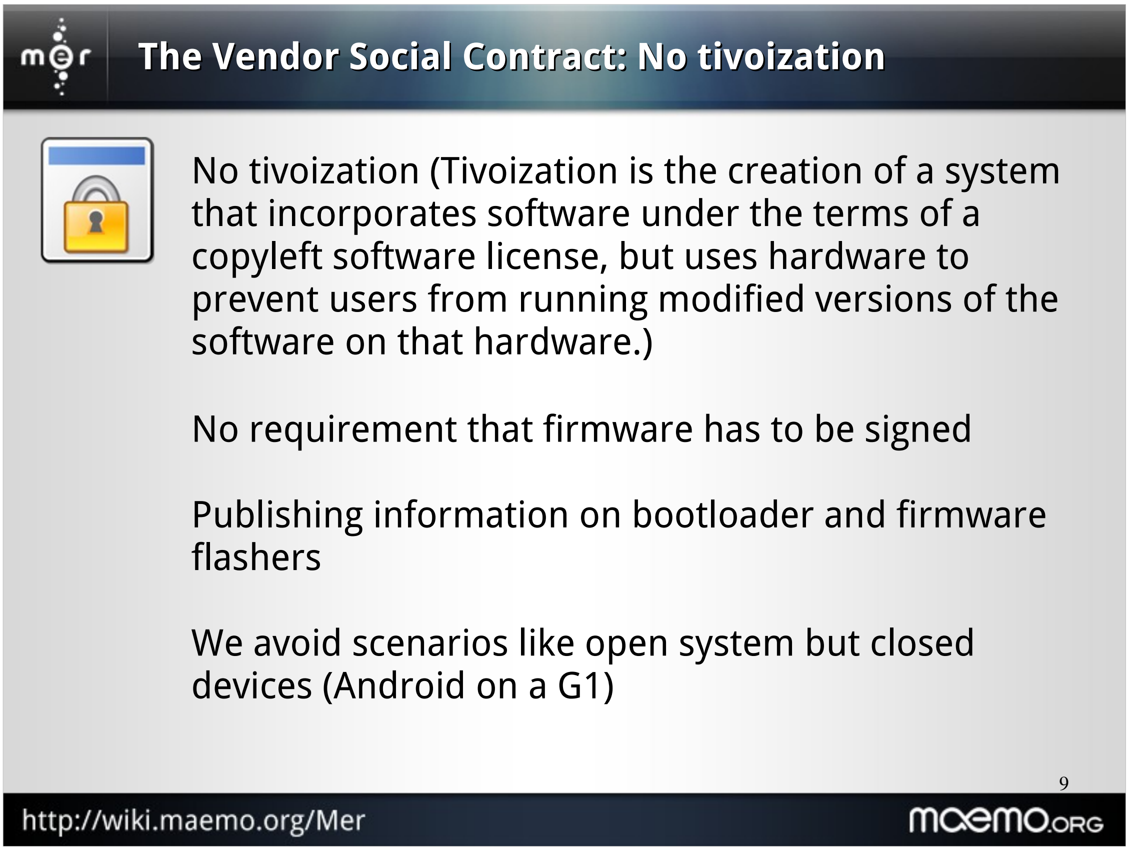
A social contract for the vendors wanting to adapt Mer on their hardware.

Principles for the interaction between the Mer project and hardware vendors.

Modelled on the open attitude by Nokia with regards to the Internet Tablets.

wiki.maemo.org/Mer/Documentation/Vendor_Social_Contract

A mobile distribution will not be worth anything without hardware to run on. We have worked on something we call the Vendor social contract, which is for the vendors wanting to adapt Mer on their hardware – and how we interact with them in the project and how this cooperation can be proper in the longer term. It was modelled on the open attitude Nokia has shown with regards to the internet tablets. You can read more about it at the URL listed. It consists of three simple principles.



The Vendor Social Contract: No tivoization

No tivoization (Tivoization is the creation of a system that incorporates software under the terms of a copyleft software license, but uses hardware to prevent users from running modified versions of the software on that hardware.)

No requirement that firmware has to be signed

Publishing information on bootloader and firmware flashers

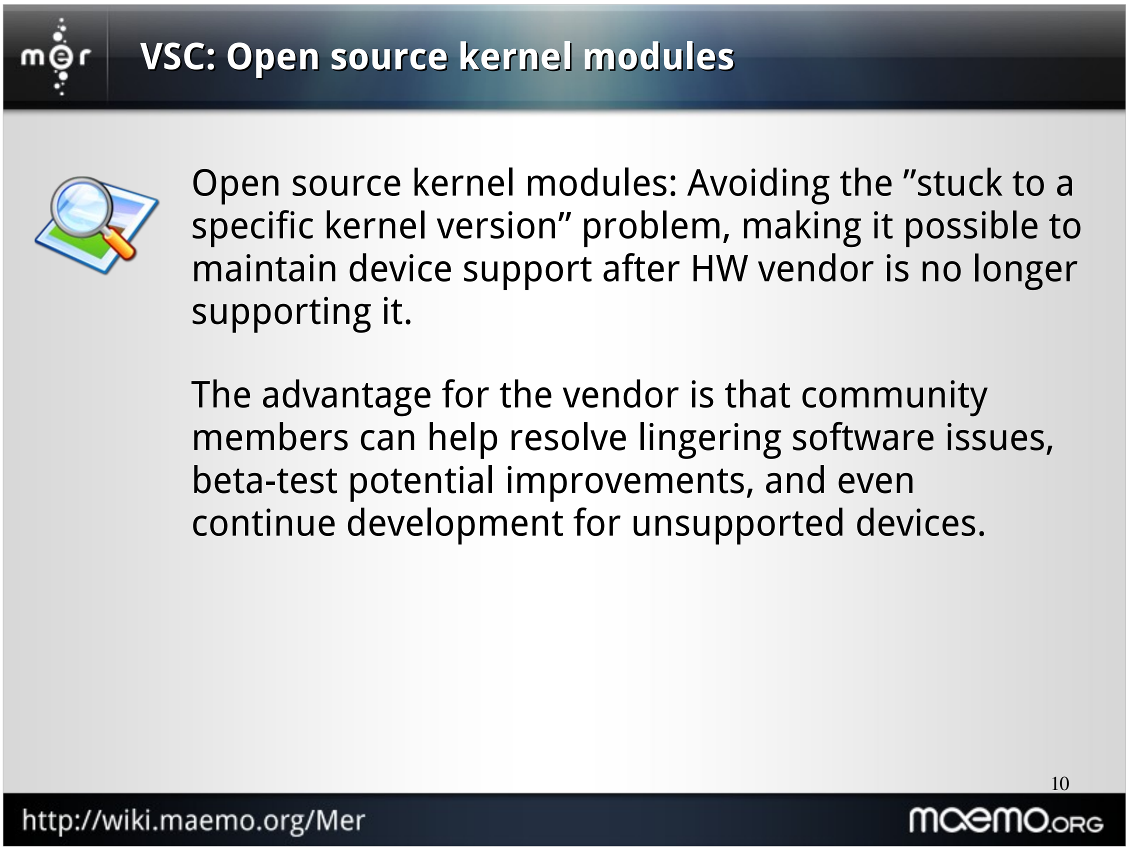
We avoid scenarios like open system but closed devices (Android on a G1)

9

http://wiki.maemo.org/Mer maemo.org

For a community project to have any kind of long-lasting relationship targetting a device, it's obvious that tivoization is unacceptable. It should be possible for a developer and to anyone to flash any firmware image of their own making to the device, boot their own kernels, without having to resort to signing firmware images, or guessing on bootloader details.

The intention is also to avoid scenarios like it is seen on T-Mobile G1 – Android is open source but you can't alter the OS on your Android phone to your own needs.



VSC: Open source kernel modules

Open source kernel modules: Avoiding the ”stuck to a specific kernel version” problem, making it possible to maintain device support after HW vendor is no longer supporting it.

The advantage for the vendor is that community members can help resolve lingering software issues, beta-test potential improvements, and even continue development for unsupported devices.

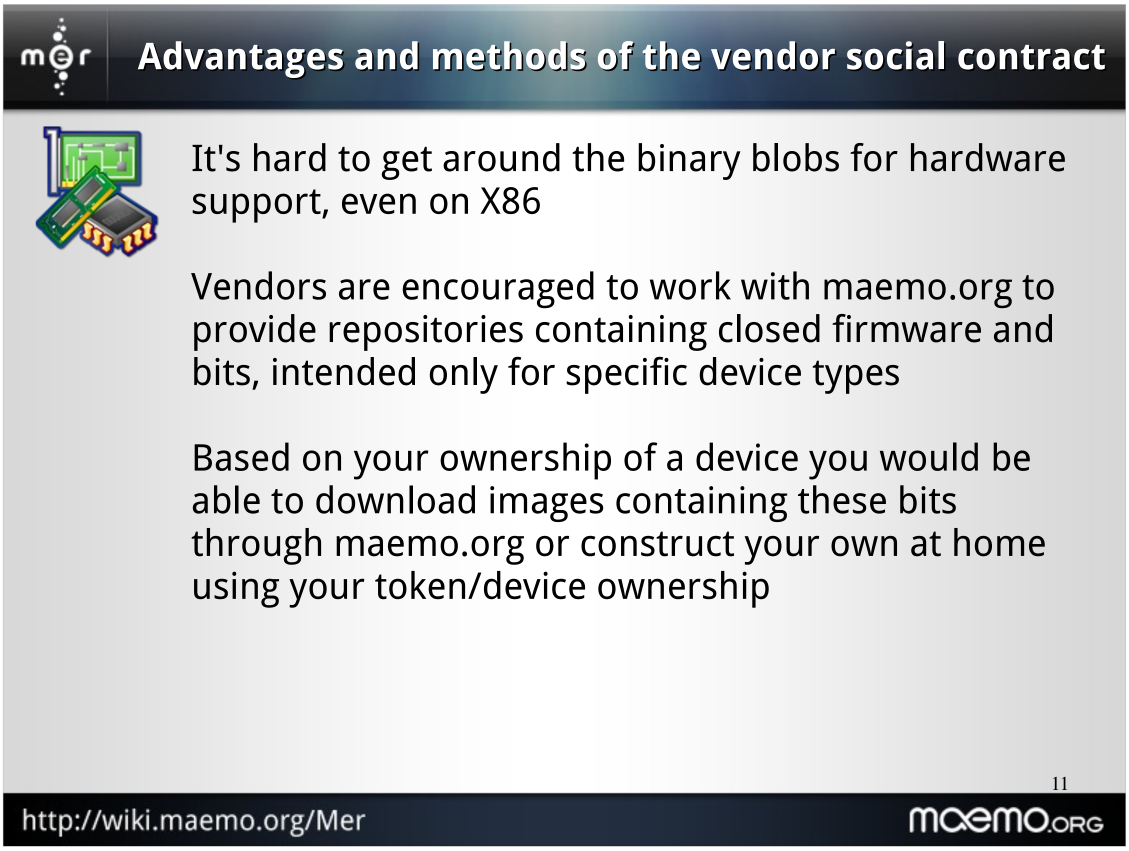
10

http://wiki.maemo.org/Mer

maemo.org

One problem that is seen quite often is the use of closed source kernel modules, which in all device communities cause the Linux kernel to be stuck at a certain version when the HW vendor no longer supports it.

Opening kernel modules allows the community to help resolve lingering software issues, beta-test potential improvements and even continue development for unsupported devices.



Advantages and methods of the vendor social contract

It's hard to get around the binary blobs for hardware support, even on X86

Vendors are encouraged to work with maemo.org to provide repositories containing closed firmware and bits, intended only for specific device types

Based on your ownership of a device you would be able to download images containing these bits through maemo.org or construct your own at home using your token/device ownership

11

http://wiki.maemo.org/Mer

maemo.org

There's binary blobs everywhere – even on X86. WiFi, Bluetooth firmware, battery daemons, licensed libraries for 3d support. And in the current hardware climate insisting they should be open source won't go down well.

What we try to do in Mer is encourage HW vendors to provide repositories containing these closed firmware and bits. The vendors often have agreements that these bits are distributable on basis of being put on certain devices (paying per device or other methods). Ideally these would be hosted at maemo.org.

The idea is that a user can based him proving his ownership of a certain device (and accept EULA) download firmware images mixed Mer and closed bits, or closed bits as debian binary packages through maemo.org.

Accomplishments: Where are we

- Easy development for the platform
  - You can develop -and- test your Mer/Maemo platform application straight on your PC
  - Cross-build for your tablet, your Wii or your fridge with osc build <platform> yourpackage.dsc

12

http://wiki.maemo.org/Mer

So what have we acheived in this reconstruction process so far?

We have established ways to make it easy to develop for Mer – you can develop -and- test your Mer/Maemo platform application straight on your PC (Vmware images exist).

After you have it throughly tested and you want to experience it on your tablet, your Wii, or even you fridge, you hand it to OpenSUSE Build Service and it'll build either locally or on a build farm for your intended CPU target – giving you freshly baked binary packages.

Mer is extremely portable – we have seperated hardware support and Mer platform. Took me 23 hours from getting it to port Mer to a SmartQ5 tablet.

We have made Hildon a session like any other in Ubuntu – and developed several patches fixing problems with the Maemo packaging – resulting in the power of Ubuntu (or Debian) together with Maemo 5.0 APIs.

Accomplishments: Where are we (2)

Easy porting:

Hardware support and Mer is separated

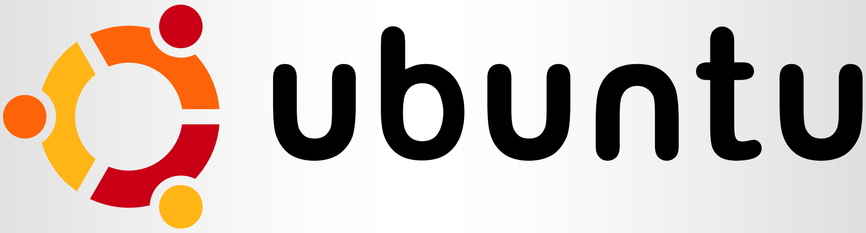
Made Hildon desktop a session in Ubuntu

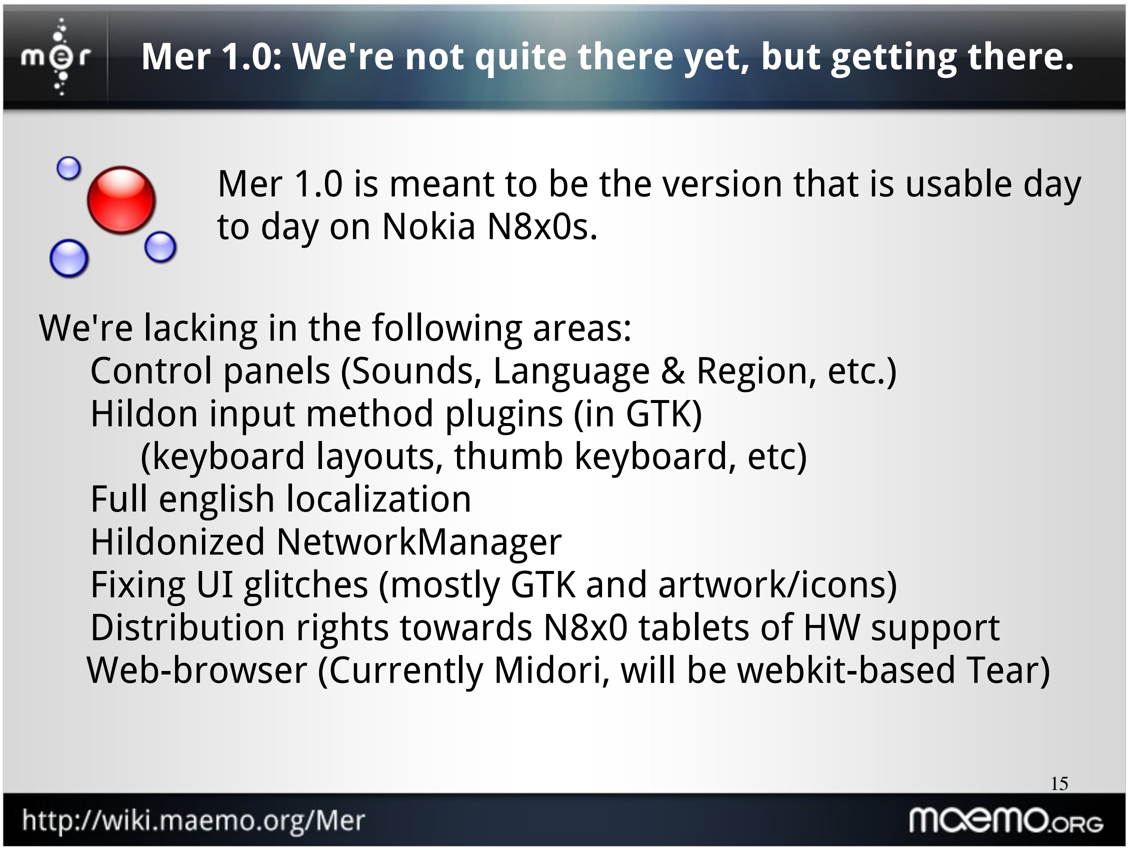
Several patches made towards Maemo packages (Scratchboxisms, Maemoisms, Libtool, etc.)

The power of Ubuntu (or Debian) together with Maemo (based on Maemo 5.0 sources)

And in addition to that:

- Ubuntu MID will be working with us to get our packages into shape and Ubuntu MID in Karmic will be based upon Mer.





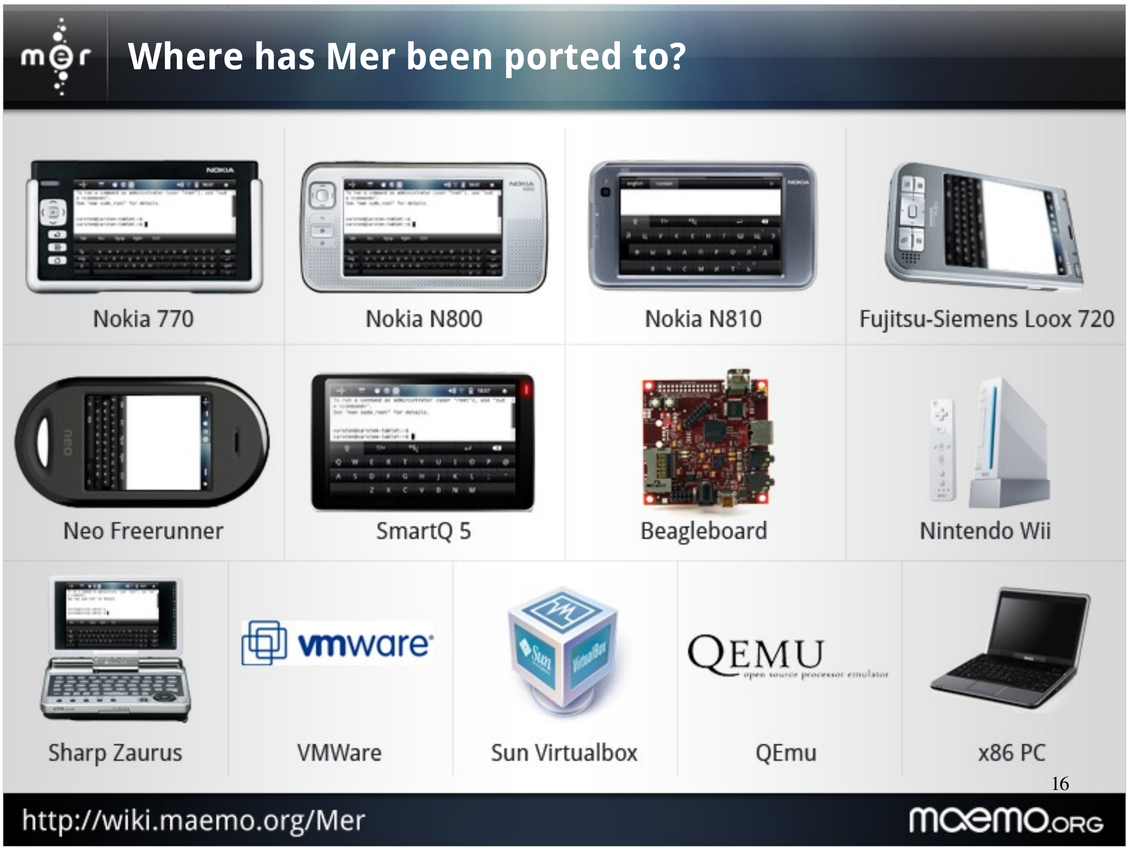
Mer 1.0: We're not quite there yet, but getting there.

Mer 1.0 is meant to be the version that is usable day to day on Nokia N8x0s.

We're lacking in the following areas:

- Control panels (Sounds, Language & Region, etc.)
- Hildon input method plugins (in GTK)
  - (keyboard layouts, thumb keyboard, etc)
- Full english localization
- Hildonized NetworkManager
- Fixing UI glitches (mostly GTK and artwork/icons)
- Distribution rights towards N8x0 tablets of HW support
- Web-browser (Currently Midori, will be webkit-based Tear)

While we've been active for a while we still have a bit to go and that's one of the reasons we're participating here. We lack in several areas but some people have already started using it on a day to day basis. If you know anyone with experience within the areas below who is interested in mobile platforms, let them know about us.



Where has Mer been ported to?

Nokia 770

Nokia N800

Nokia N810

Fujitsu-Siemens Loox 720

Neo Freerunner

SmartQ 5

Beagleboard

Nintendo Wii

Sharp Zaurus

VMWare

Sun Virtualbox

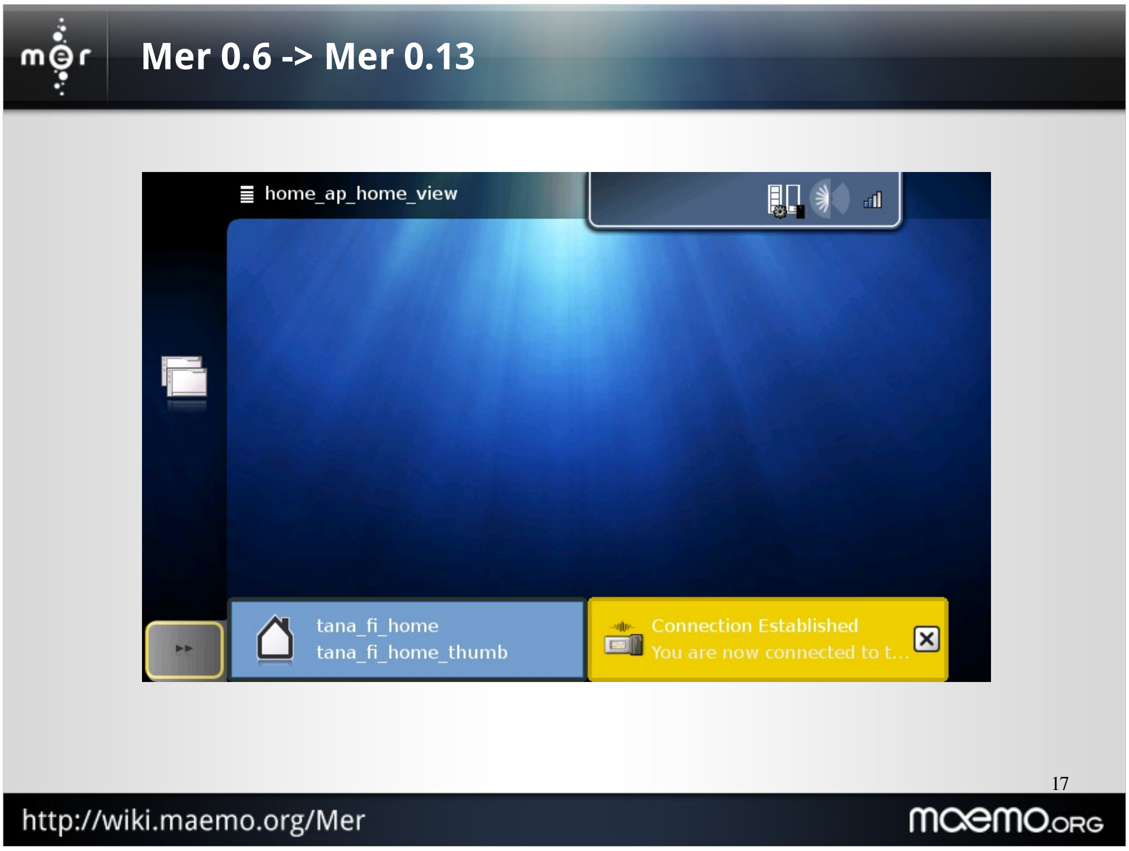
QEmu

x86 PC

http://wiki.maemo.org/Mer

maemo.ORG

As I said before, Mer is extremely portable – all the pictures here list what devices we have had Mer running on. I wasn't kidding about the Wii port either. The hope is that we can help focus the efforts in many different device communities towards the Maemo platform. Do note that we also target Vmware and VirtualBox – you can run a full X86 Mer in your Vmware.



Mer 0.6 -> Mer 0.13

home_ap_home_view

tana_fi_home
tana_fi_home_thumb

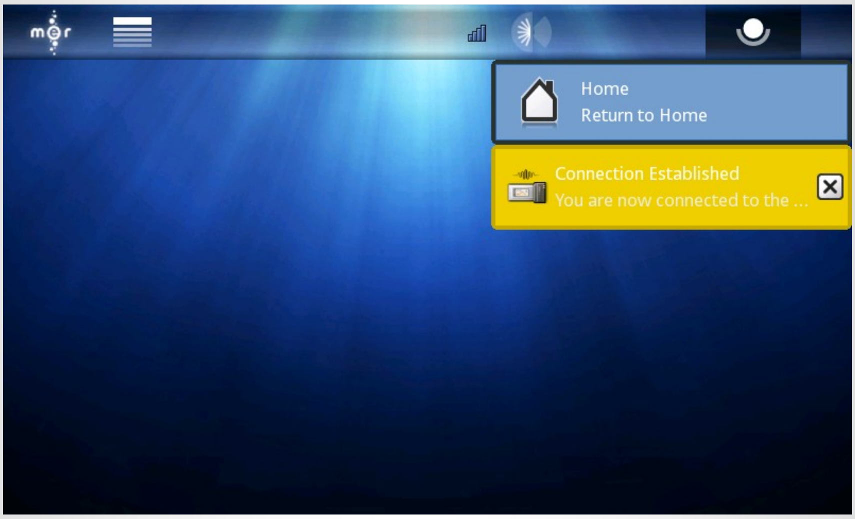
Connection Established
You are now connected to t...

http://wiki.maemo.org/Mer

maemo.org

Now, to show you how Mer looks have improved over time – we began with a clone of Maemo 4.0 – the themes and icons are all open source – theme was laying on a contributor's harddisk and we gladly accepted it. Notice the lack of localized strings.

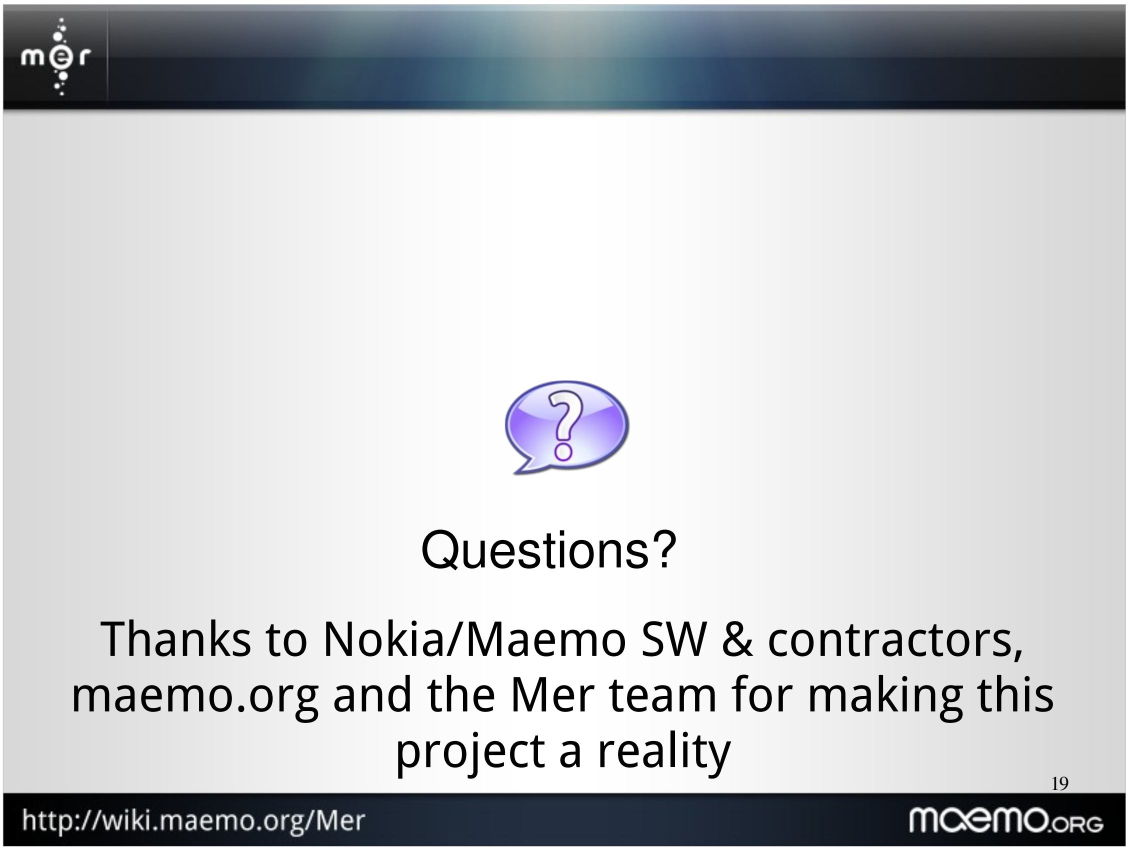
Mer 0.13



Home
Return to Home

Connection Established
You are now connected to the ...

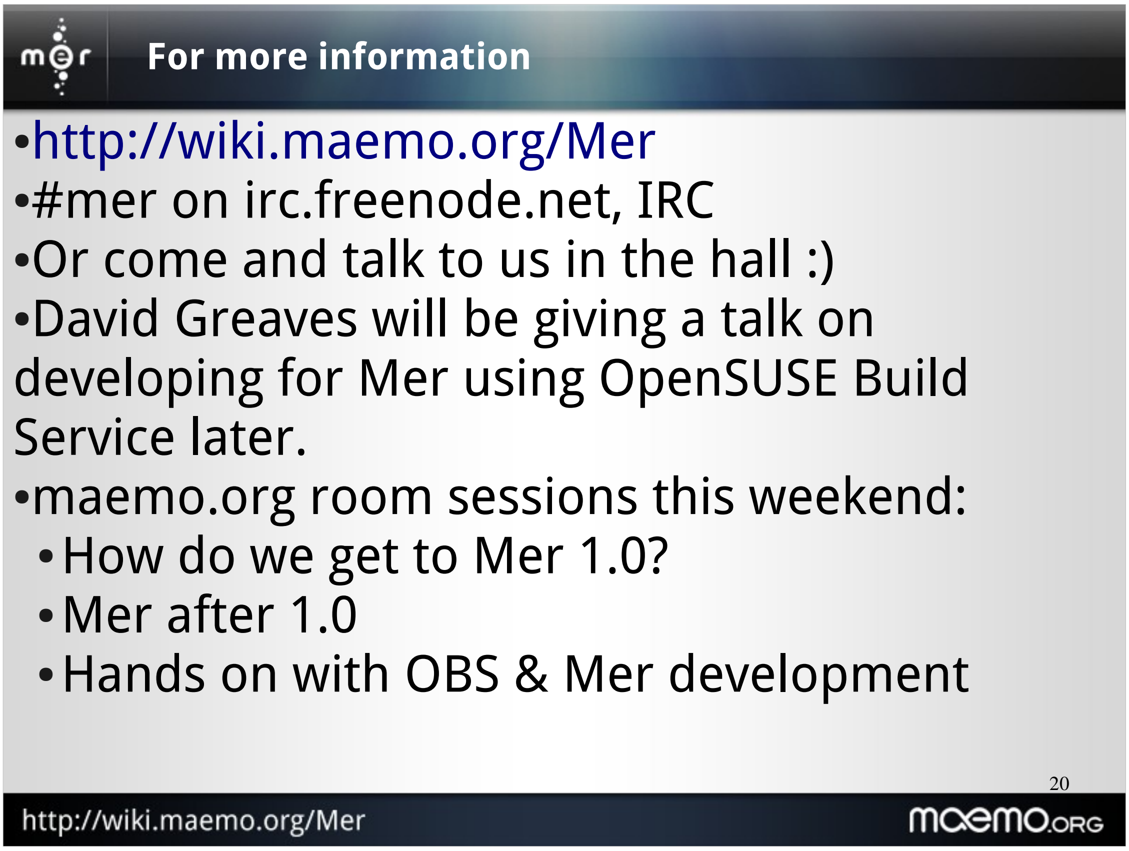
We gained the interest of a amazing russian designer who came up with mockups fof how Mer could look – and this is part of the end result – some flaws still obviously.



Questions?

Thanks to Nokia/Maemo SW & contractors, maemo.org and the Mer team for making this project a reality

So, any questions?



For more information

- http://wiki.maemo.org/Mer
- #mer on irc.freenode.net, IRC
- Or come and talk to us in the hall :)
- David Greaves will be giving a talk on developing for Mer using OpenSUSE Build Service later.
- maemo.org room sessions this weekend:
  - How do we get to Mer 1.0?
  - Mer after 1.0
  - Hands on with OBS & Mer development

If you'd like to know more --

Thank you for listening!