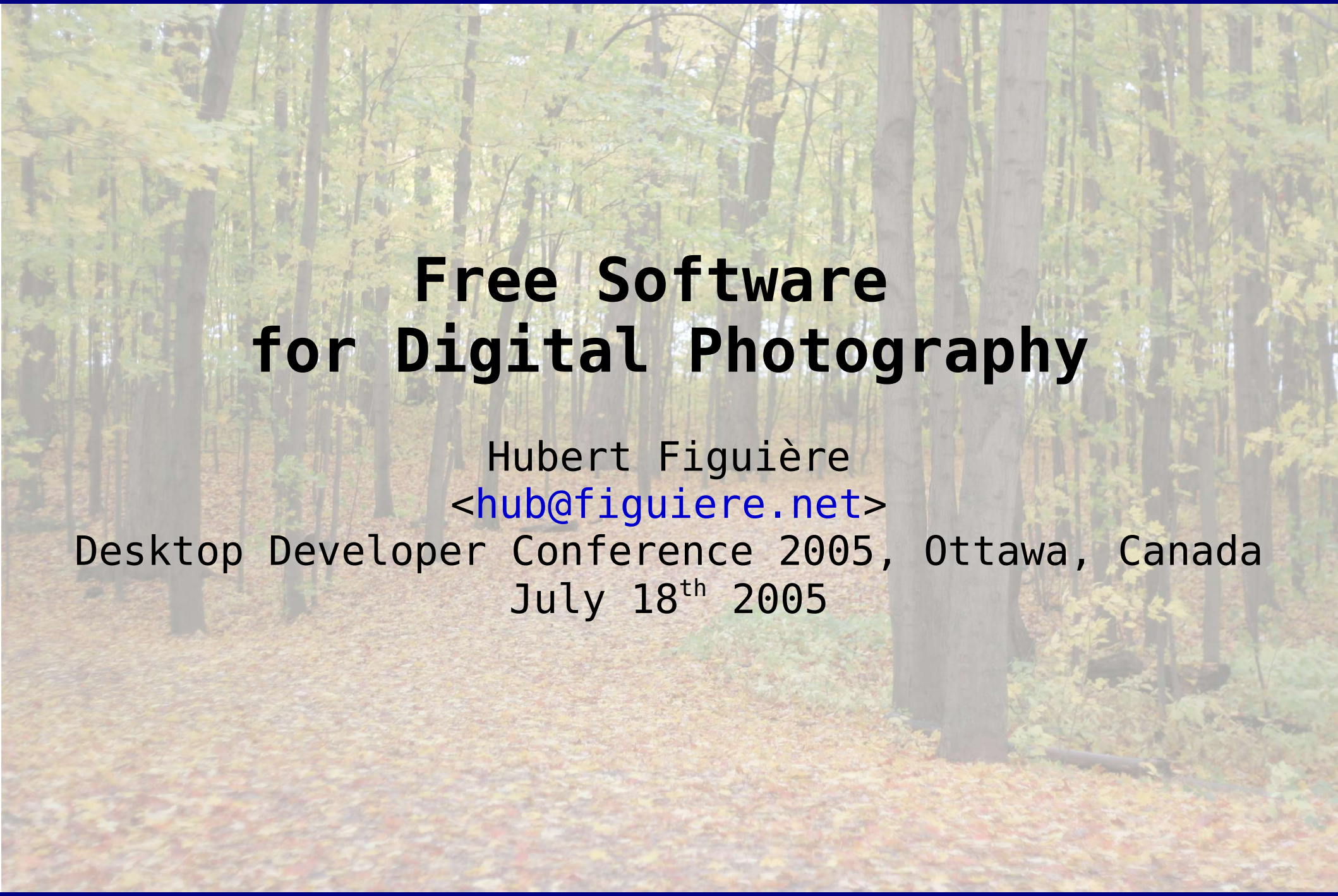


Free Software for Digital Photography

Hubert Figuière
<hub@figuiere.net>

Desktop Developer Conference 2005, Ottawa, Canada
July 18th 2005

Menu

- Why?
- Three Tasks
- What can we do ?

Why ?

- **Digital Photography is a hot desktop “application”**
  - **We need a solid foundation**
  - **We have good building bricks**
  - **We don't want to reinvent the wheel: stop the NIH syndrome**
  - **There is still a lot of work to do**

Three tasks

- Getting images out of the camera
- Displaying meta-data
- Decoding proprietary RAW files

Task 1

- Getting the picture from the camera

Three ways

- By using a memory card reader
- By mounting the camera
- By using gphoto2

In fact it is two

- Insert the card in the reader
- *or* connect the camera
- Let the desktop take care of that and mount it
- Enjoy your file manager

Example


Computer
File
Edit
View
Places
Help
Floppy
CD-ROM
131M Removable Media
Filesystem
Computer
4 items

And one less easy

- gphoto2 required for lot of camera
  - that do not connect using USB
  - that do not support Mass Storage
- Works usually well
- ...but can be really painfull on the user side

Driving the camera

- libgphoto2 is the library that drives cameras
  - used by developers to import pictures
    - gphoto2
    - gtkam
    - f-spot
    - digikam
    - gthumb
    - kio_slave
  - extensible to support new cameras

libgphoto2: Facts

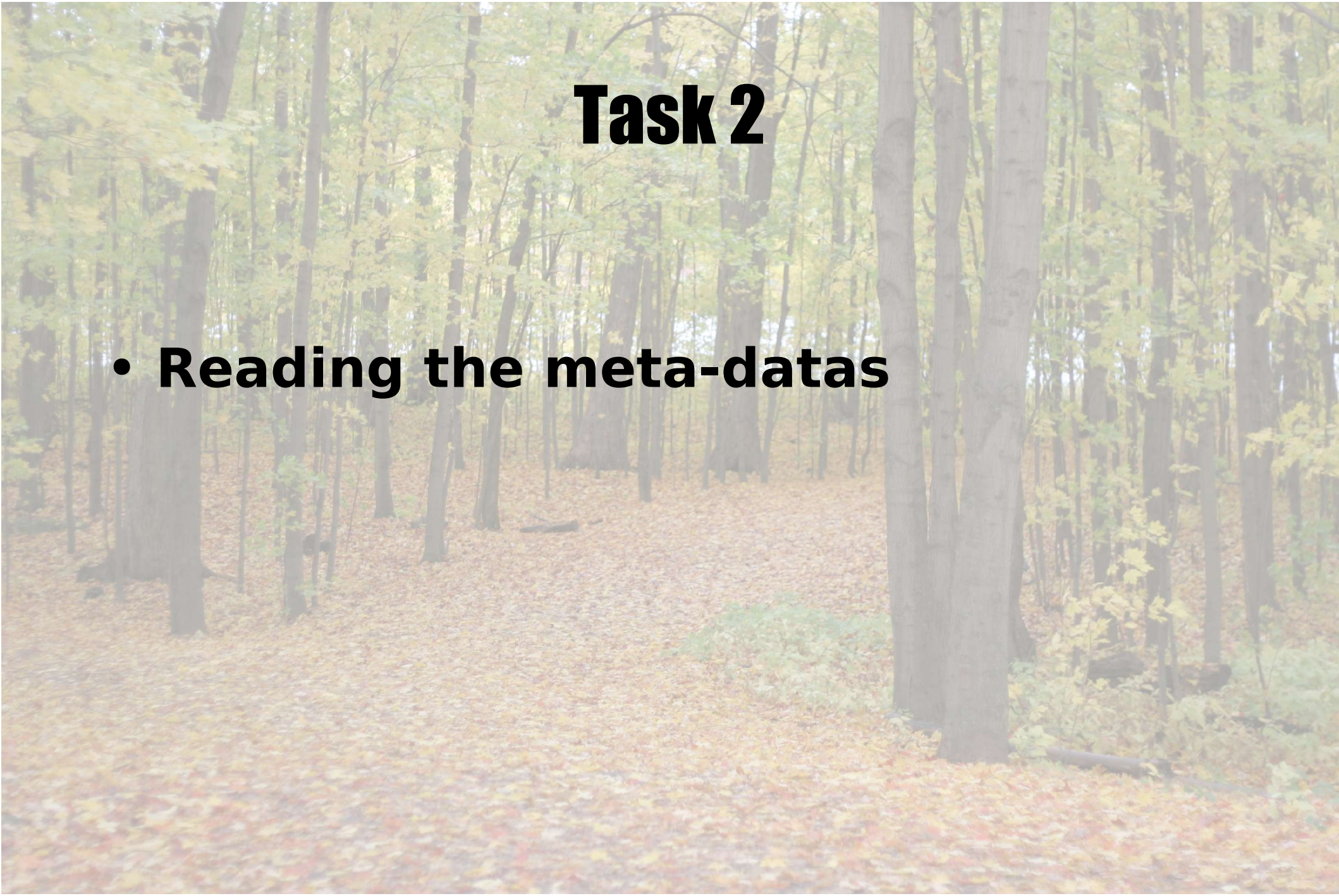
- License: LGPL
- Language: C
- Bindings: C#, Java
- Support over 500 cameras
  - more than any other OS built-in support
  - more than the manufacturer
- Over 5 years of development

Challenges

- **Supporting a camera without the camera**
- **Supporting a camera without the documentation**
- **Supporting non supporting vendors**

Problems

- The user shouldn't have to care
  - Developer should
  - They should provide the same features
- 2 APIs
  - one for Mass Storage
  - one for others



Task 2

- **Reading the meta-datas**

What kind ?

- **Exif, standard**
  - **Specify a lot of shooting information**
- **MakerNotes: proprietary tags inside Exif**
  - **Specify camera specific information**
  - **Undocumented**

Why ?

- **Contain information that can help enhance the picture**
  - **Color balance**
  - **Aperture**
- **Contain information about the lens**
  - **Geometry correction**
  - **Stitching**

How to decode Exif ?

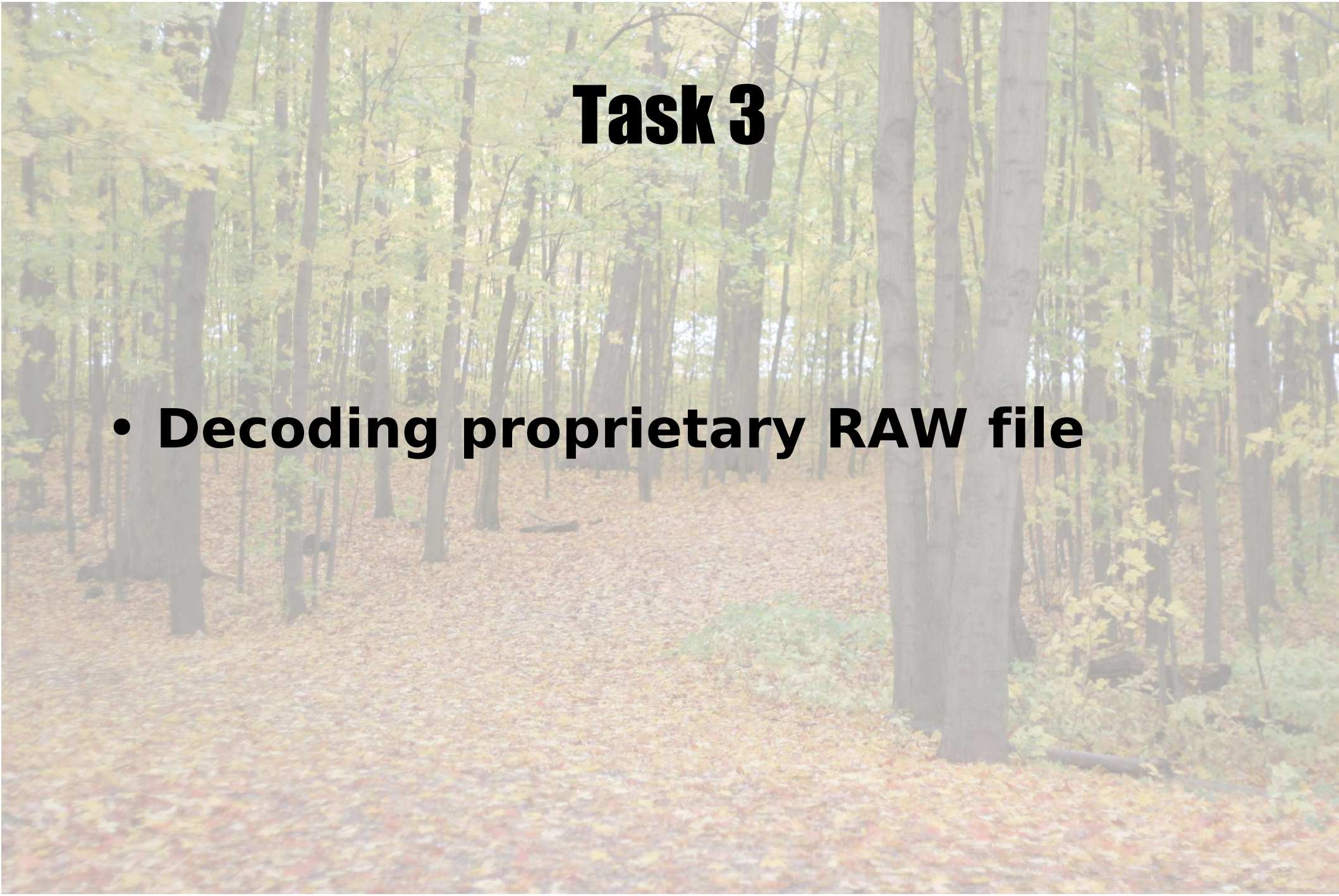
- libexif
  - by far the most widely used
- Image::ExifTool from CPAN
  - The Perl5 module
  - Really complete
- Exiv2

Libexif

- License: LGPL
- Language: C
- Bindings: C#
- Used by lot of software, including GNOME, EOG, f-spot, gphoto2
- Decode MakerNotes

Image::ExifTool

- License: Same as Perl
- Language: Perl5
- Really comprehensive



Task 3

- **Decoding proprietary RAW file**

The RAW problem

- Undocumented & proprietary
- Unusable as such
- Different for almost every camera
- Camera maker are liars

Facts

- **Most of them are based on standards**
- **All of them have been decoded in some way**
- **RAW file is a requirement for optimum quality**

True Lies

- There is no universal solution
  - They could extend TIFF as they currently do
  - DNG is a technically good example as it provide enough flexibility to be extended as a standard

More True Lies

- ***"Canon clearly recognizes the desires of its customers to preserve their images indefinitely. But we have never dropped support for RAW image processing with any of our EOS digital cameras, and there is no reason to suppose that we ever will. Frankly, I find any insinuation to the contrary to be baseless and reprehensible."*** **-- Chuck Westfall, Canon USA**
  - **Digital Photo Pro 1.6.1 from Canon do not support Canon D30 (released in October 2000 for USD$3000)**
  - **DCS-3, DCS-1, D2000 and D6000 are not supported by current software at all**

Even More Lies

- ***"We are committed to and will continue to work with third party software vendors who wish to support the Epson raw file format (.erf)."*** **-- Epson USA**
  - **My attempts to contact Epson through official channels to obtain the documentation has been left unanswered.**

Decoding RAW file

- **Only one software**
  - **dcraw**

dcraw

- License: “MIT like” + GPL
- Language: C
- Does not support meta-datas
- Really hard to integrate in an application
- Decode virtually everything
- Written by reverse engineering

Color Management

- Required for RAW decoding
- And even for viewing JPEG
- Needs infrastructure
  - xicc spec for screen color profile
  - toolkit support for easy color proofing
- Lot of libraries including
  - lcms
  - (see OpenIcc on Freedekstop.org)

Future

- What can be done ?
  - Unify download
    - by using libgphoto2 for Mass Storage device
  - Enhance plug and play of cameras
    - by providing nice applications to download pictures
  - Provide a library for RAW file processing
    - libopenraw

Libgphoto2 and Mass Storage

- New "`disk:`" device
  - use libhal to locate the disks
  - specify like this "`disk:/mount/point`"
  - use the "Directory" driver that just copy from the file system
- Planned for 2.2 (almost complete in CVS)
- Compatible with current APIs

Enhance Plug and Play

- We have
  - libhal
  - gnome-volume-manager on GNOME
  - libgphoto2
  - a nice GUI app (gthumb, f-spot)
- What else do we need?
  - solve the Mass Storage “issue”

Write or contribute to Applications

- Image processing
- Image editing
  - Enhance gimp
- Image cataloging


Questions
?