Parts on the table
Parts on the table

Apollo 13
A neonatal incubator
A car parts incubator

http://goo.gl/P9ftv
The problem

• You need an OMERO client to extract data
• When working with OMERO-unaware software, the data are duplicated
• The user interface is unfamiliar
• The OMERO client & server versions must match
Solution

More parts on the table!
A possible solution

Expose the data as a filesystem

- All software can access files
- Users are familiar with filesystems
- Data can be generated on-the-fly, avoiding duplication
“Filesystems in User SpacE”

- Generate/access filesystems via a process running as a regular program
- Originally developed for the Linux kernel
- Available also for MacOSX
Proof-of-concept

- Written in Python
- Uses python-fuse
- Uses omero.gateway
- 111 lines of code (no documentation :-)
- quick 'n dirty prototype
Proof-of-concept

Implements:

- `getattr()` (file/directory, size)
- `readdir()`
- `open()`
- `read()`
Running on Ubuntu

- Install FUSE kernel module
- Install python-fuse
- Install ICE 3.3
- Extract lib/python/** from OMERO.server
- Add to PYTHONPATH
- Change host, user and password in script
- ./ome-fuse.py <mountpoint>
• Does not implement reading images yet
• No caching
• No metadata
• Keeps connection alive until unmount
• Uses 4.2.2 API
Wish list

- Opening images
  - in multiple formats
  - possibly split into 2D images
- Importing images
- Reading/writing metadata
- Integration into LDAP/ActiveDirectory/etc
- Adoption by OME :-(
Limitations of OME-FUSE

- Does not work on Windows
- Is not integrated into OMERO, thus cannot benefit from server-side optimizations
- Additional dependencies (when we wanted less)
- Possible performance bottlenecks
Alternatives to FUSE

Local alternatives

• Custom filesystem
  – platform-dependent
  – technically challenging

• Interception through LD_PRELOAD
  – non-portable (only works on Linux/MacOSX)
  – requires special program startup
Alternatives to FUSE

Remote alternatives

• Samba
  – needs on-demand FUSE mounting

• SMB/CIFS
  – JLAN (AndroidSMB)
  – porting AzSMB

• NFS
  – JavaNFS
Thank you!

The OME team  http://www.openmicroscopy.org/
The Bio-Formats team http://www.openmicroscopy.org/
The ImageJ2 team http://imagejdev.org/
The Fiji team http://fiji.sc/