



EIDORS

# Collaborative open scientific software : *The EIDORS experience*

Andy Adler<sup>1</sup>, William R.B. Lionheart<sup>2</sup>

<sup>1</sup>University of Ottawa, Canada

<sup>2</sup>University of Manchester, U.K.





# What is the problem?

- Science / academia has traditionally been about openness and sharing
- Science is becoming very proprietary
  - Funding
  - Publish or perish
  - Academic as inventor
- Nature of scientific endeavour is changing
  - Most “easy” problems are solved
  - Interdisciplinary research
  - Sophisticated numerical modelling



EIDORS

# “paper” no longer enough

- One key requirement of a scientific paper is to allow replication of results
- But complex code can't be described in a short paper
- Unintentional Errors
  - My grad students code has errors
- Intentional Errors
  - Twiddle factors
  - I find I can't duplicate results



EIDORS

# SW practice problems

- Sloppy SW practices in universities
  - No version control
  - No test cases
  - No build scripts
  - Matlab





# Matlab issues

- It is becoming the lingua franca of engineering / math / physics
- Advantage is that people aren't coding their own matrix inverses
- However
  - Not good for large projects
    - Name Spaces, Automated Testing, OO, ...
  - Syntax easily allows subtle bugs
  - GUI encourages “hacking results”



EIDORS

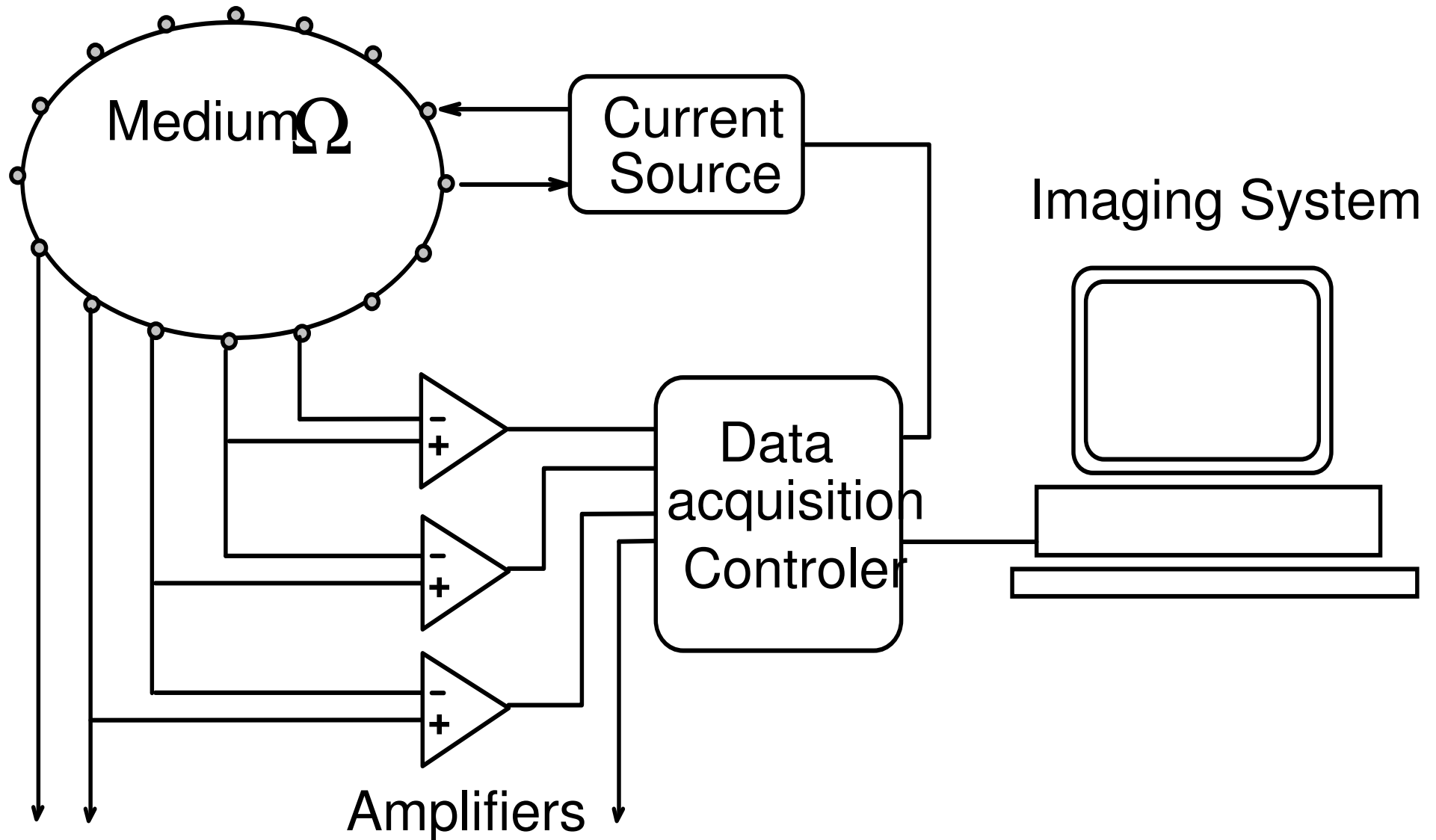
# Open source scientific algorithms

- Documentation
- Testing / Validation
- Comparison between approaches
- Standing on the shoulders of giants
  
- We have attempted to do this in one Medical Imaging field: EIDORS

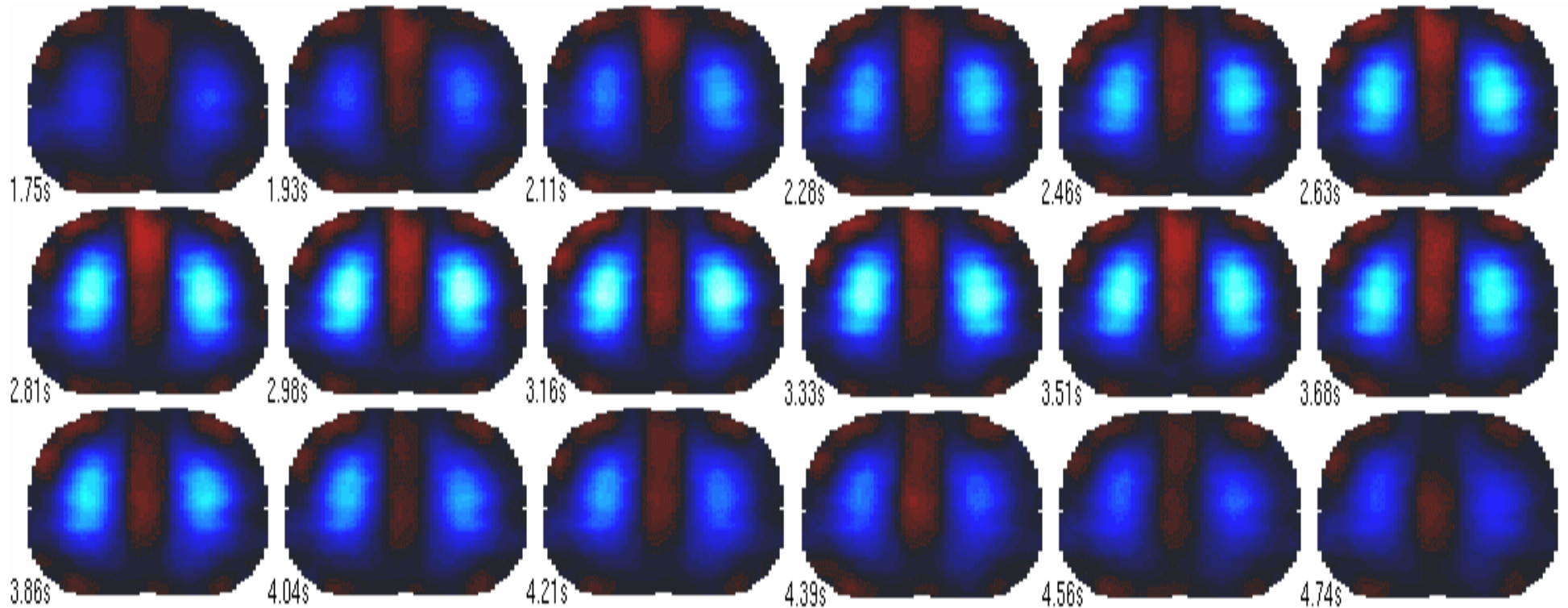


EIDORS

# Electrical Impedance Tomography



# Application: Breathing



Chest cross-section images of author's chest/lungs





EIDORS

# Goal: software community



**Project:** Electrical  
Impedance and  
Diffuse  
Optical  
Tomography  
Reconstruction  
Software





EIDORS

# Why Walruses?

1. EIT images blobby objects in aqueous media; Walrus, a fat, blobby animal that lives in water.
2. Walrus is EIDORS logo
3. Walruses are much funnier than a talk about software



Images credit: [www.biosbcc.net](http://www.biosbcc.net)

© Genny Anderson



EIDORS

# EIDORS Features

## **Open-source:**

- License: GNU General Public License.
- Free to use, modify, and distribute modifications.
- May be used in a commercial product

## **Hosted on Sourceforge.net**

- Software is available for download  
(version 3.1RC1 – Feb 2006)





# Thoughts on Licenses

Software

GPL / BSD etc.

CC w/ Attrib

Data

CC noDeriv w/Attrib

Results

CC noDeriv  
w/Attrib??

Procedures /

Designs /

Questionnaires

?? haven't thought



# History

- Current version 3.1 (Feb 2006)
- Previous versions (1+2) were monolithic code “dumps” from PhD theses (Vauhkonen, Polydorides)
- Problem was that people would copy code and make a few interspersed changes
- Contributors would then “dump” entire code back to us



EIDORS

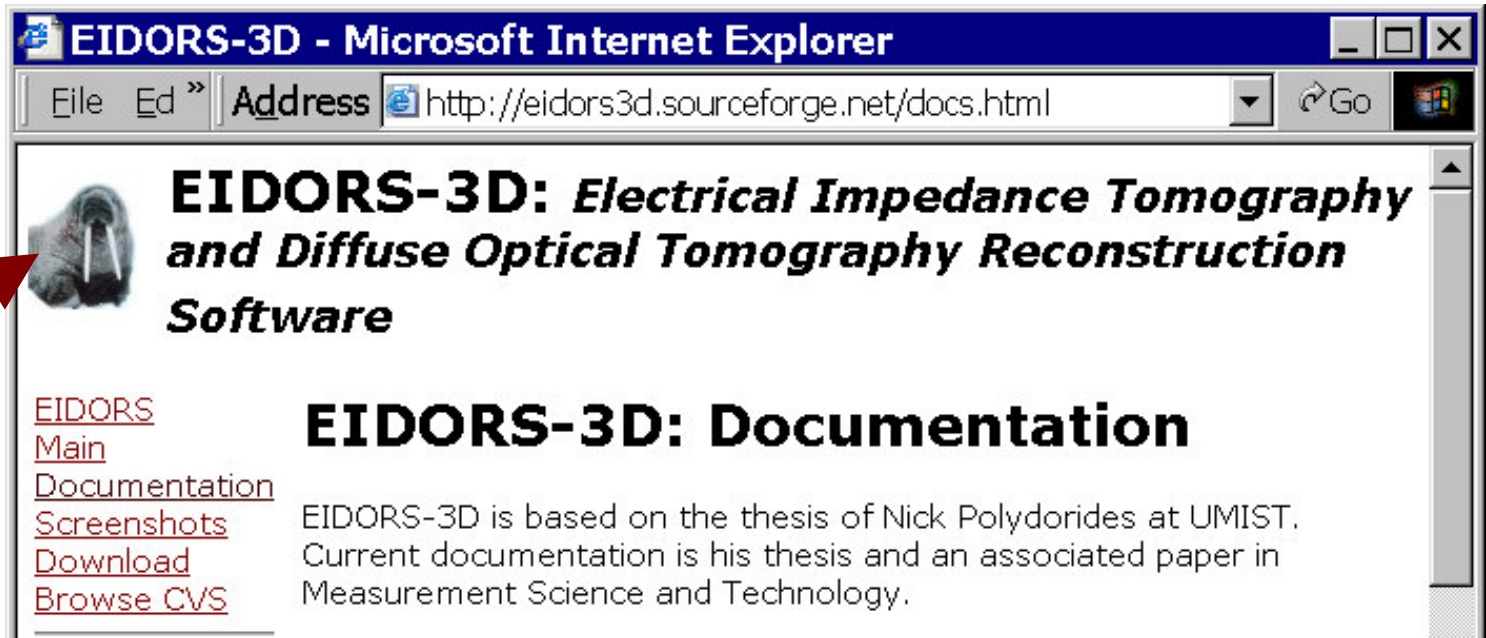
# EIDORS Goals

- Community Software
- Pluggable
- Language Independent
- Testable
- Automatic caching



# Community Software

- GPL Licence
- Sourceforge hosting



**EIDORS-3D: *Electrical Impedance Tomography and Diffuse Optical Tomography Reconstruction Software***

**EIDORS-3D: Documentation**

EIDORS-3D is based on the thesis of Nick Polydorides at UMIST. Current documentation is his thesis and an associated paper in Measurement Science and Technology.

[EIDORS](#)  
[Main](#)  
[Documentation](#)  
[Screenshots](#)  
[Download](#)  
[Browse CVS](#)

**Walrus**

- Conduct workshops



# Software Structure

EIDORS /

*calc\_jacobian.m*

algorithms /

np\_2003 /

*np\_calc\_jacobian.m*

## Usage:

```
mdl.nodes= ...
```

```
mdl.elems= ...
```

```
mdl.jacobian = 'np_calc_jacobian'
```

```
OR = @np_calc_jacobian
```

```
mdl.np_calc_jacobian = parameters
```

```
J=calc_jacobian( mdl, ... )
```





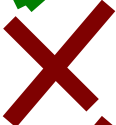



# “Pluggability”

Need to

- let multiple users contribute
- allow mix-and-matching of algorithms
- show as much (or as little) internals as necessary

OO Features needed

- abstraction 
- encapsulation 
- polymorphism 
- inheritance 



# Namespaces

- Matlab syntax namespace support sucks
- Matlab OO syntax is hard to understand – and may change in next few versions
- Octave doesn't support Matlab OO
- Scientists / engineers don't get OO





EIDORS

# Language Independence

Support for

- Matlab  $\geq$  6.0
- Octave  $\geq$  2.9.5

Desire to support Scilab / NumPy, but no-one has volunteered to do the work





# Language Issues

## Matlab

- Big changes across versions (6,6.5,7)
- graphics / syntax
- toolboxes (we want to not require any)

## Octave

- Sparse matrix support not complete (eigs, svd, luinc, cgls)
- Graphics (not big issue – we use Mayavi)
- Mex files



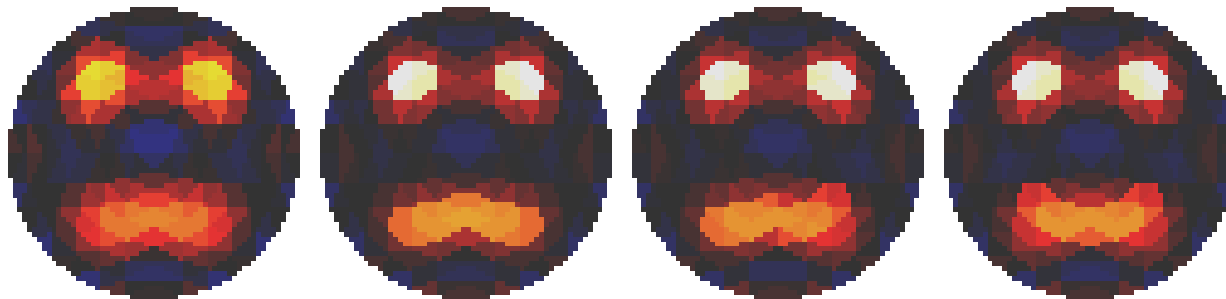
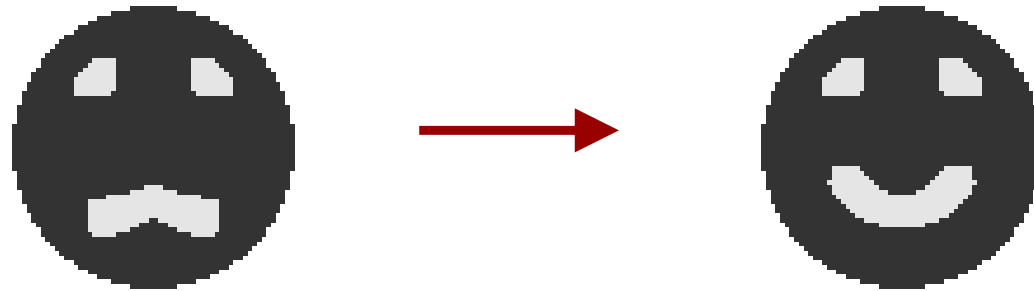
# Testability

- Software is intrinsically difficult to test.
- Numerical software is probably more difficult
- Many scripting languages manage testing well (ie. Perl Test::modules)
- Attempt to implement regression testing.
  - Need to choose threshold for equality



# Numerical SW bugs

Example: *happy transform*



Reconstructed images illustrating the effect modifying the weighting of edge preserving image priors. *Left to Right*: Edge prior with no weighting, with weighting for positions in sad face with weighting for positions in happy face, with weighting for sad face (left) and happy face (right),



# Matrix Caching

- User code often says

```
for i=1:N
    if i==1
        J= calc_jacobian( ... )
    end
    % use J and i
end
```

- This is prone to errors
- We want to automatically cache the result of `calc_jacobian(p1,p2,p3)` if we have ever seen it before



EIDORS

# Automatic Matrix Caching

- Problem:
  - Var may have same value but be created twice
- Solution:
  - Iterate over each var's storage space and do a SHA1 hash
  - store in global variable:  
`eidors_obj.id_0102...`





EIDORS

# Thanks

