

# STEM Society Meeting, May 12, 2015

James Emery

Last Edit: 5/31/2015

## Contents

<b>1</b>	<b>About the STEM Society and the STEM Society Website</b>	<b>1</b>
<b>2</b>	<b>The May Meeting Announcement</b>	<b>2</b>
<b>3</b>	<b>A Wide Ranging Discussion on Multiple Subjects</b>	<b>3</b>
<b>4</b>	<b>Jim Emery: Printing a Book on the Espresso Book Machine at the Woodneath Library</b>	<b>4</b>
<b>5</b>	<b>Jim Emery: Making Arrows and Vectors With Postscript, Infix, Prefix, Postfix and RPN</b>	<b>4</b>
<b>6</b>	<b>Jim Emery: Vector Analysis Applied to Carpentry</b>	<b>5</b>
<b>7</b>	<b>Creating a Postscript Figure</b>	<b>7</b>

## **1 About the STEM Society and the STEM Society Website**

STEM is an abbreviation for Science, Technology, Engineering and Mathematics. The acronym STEM is commonly associated with K-12 education, but our use of the term is only slightly bound to this meaning. There are over one hundred people on the mailing list, although a much smaller group attends any one meeting. We meet on the second Tuesday of each month at

the Trailside Center at 99th and Holmes in Kansas City, Missouri. The meetings are open to all. The start time is 6PM. We make presentations, have discussions, and have demonstration experiments. These relate to Science, the History of Science, Mathematics, Engineering, Philosophy and Technology at all levels. The topics have ranged from a technical discussion of the mathematics of General Relativity to scientific experiments for young students.

These meeting notes contain links to many other documents, which may be viewed or downloaded by clicking the link. A partial list of documents can be reached by clicking the heading **Documents**. The meeting notes may also be viewed in an archive file (archive.pdf), which is in the list of documents. Many of the documents are PDF files. They may be viewed or downloaded to the computer by clicking, provided Adobe Reader, or another program capable of reading PDF files, is present. There are many more documents available at the site than are listed under **Documents** because the documents.htm file is not at all up to date. The last time I checked, about March 2014, there were about 350 document files on the site. We are in the process of creating better techniques for finding documents and authors. The first meeting of the STEM Society was in November of 2006. For several years we used the content management program called Joomla. It had a fancy looking interface, but was hard to use. It overran the space somehow at our internet provider Bluehost. So we now have a very simple HTML site. It is not so slick looking as Joomla, but is very easy to maintain and modify.

**The web site is:**

<http://www.stem2.org/>

**Direct to the documents list:**

<http://www.stem2.org/je/documents.htm>

**Direct to the archive file:**

<http://www.stem2.org/je/archive.pdf>

## **2 The May Meeting Announcement**

The May meeting of the STEM Society will take place on the second Tuesday of the month, May 12, 2015, at the Trailside Center at 99th and Holmes in Kansas City, Missouri. The starting time is 6PM.

Possible Topics:

- (a) Producing a technical book on the Espresso Book Machine at the Mid-Continent Public Library. The new book will be called **Optics and Electromagnetic Theory**, and will include sections on **Optics, Electromagnetism, Relativity, and Vector Analysis**
- (b) A wood joining project using Vector Analysis.
- (c) Creating postscript figures for scientific documents. Drawing shaded arrows and vectors in figures.
- (d) It might be interesting to talk about the Planck natural units, the Planck length, and Planck time, et cetera.
- (e) A discussion of the war on Science. The congress has passed some kind of law forbidding the EPA to talk to scientists? A very nutty country.
- (f) If time permits other topics in Mathematics, Physics, or Engineering.
- (g) Surprise extemporaneous talks, projects, "show and tell," are encouraged.

### **3 A Wide Ranging Discussion on Multiple Subjects**

I don't remember all the topics, much less the details, but they included the visit to the Galapagos Islands by Tom Grant and Rick Hines, Darwin in general, Solar Panels (Rick is thinking about putting solar panels on the roof of his house in the Dominican Republic, because the cost of electric power there is 50 cents per Kilowatt Watt hour), and a future talk by Tom Grant on Paleontology. Recall that Tom gave a talk on solar panel leasing a few meetings back. Tom and Rick are cave explorers, and Tom travels all over the country and perhaps the world to dig for dinosaur bones. Unfortunately, I have put off writing these notes for almost a month, so parts of the meeting are a little hazy. Steve Cummings may have also presented a show and tell.

## 4 Jim Emery: Printing a Book on the Espresso Book Machine at the Woodneath Library

Here is some documentation for this printing process:

<http://www.stem2.org/je/makebook.pdf>

The latest book printed is titled **Optics and Electromagnetic Theory** by James Emery. Chapters in the book include **Optics**, **Electromagnetic Theory**, **Relativity**, and **Vector Analysis**. This edition is 223 pages long. It definitely needs some more copy editing, but I wanted to put it out as an experiment in making a technical book on the machine. I brought a copy to the meeting for examination.

Tools used to make the book included **Latex** for the book-block (inside pages) and the front-matter ( title page, copyright page, and short preface), and **Microsoft Publisher** for the cover. The front matter and the book-block were joined into one file called **oem.pdf**, which is a 8.5 by 11 pdf. The cover file was saved into a pdf **oemcover.pdf**, which is an 11 by 17 pdf. These two files are put into the machine.

The pdf of the cover may be downloaded from

<http://www.stem2.org/je/oemcover.pdf>

The final book was sheared to size 8 by 9 inches in the machine.

## 5 Jim Emery: Making Arrows and Vectors With Postscript, Infix, Prefix, Postfix and RPN

Postscript is a language for creating text and graphics, the language uses postfix notation (RPN, Reverse Polish Notation), like the HP calculators. I wrote some subroutines for creating shaded arrows and vectors for figures. These occur in the book and in the figure for the document discussed in the next section.

The figure picturing the representation of the **E** and the **H** fields, which occurs on the front cover, was produced with a perspective projection based

on principles of **Projective Geometry**. A starting point and ending point in 3d are projected back to the picture plane, and each vector is drawn and shaded in this 2d picture plane. I discussed how this was done.

## 6 Jim Emery: Vector Analysis Applied to Carpentry

See the document

<http://www.stem2.org/je/molding.pdf>

The vectors in the figure of this document were created using my graphics language called EG, which I created many years ago. An eg file can be converted to Postscript using my translation program written in C called **eg2ps.c**

Here is part of the file called **moldingf1.ps**. Toward the end of what is shown here the outline of an arrow is drawn, and it is filled with black with the command

```
0 0 0 setrgbcolor
```

In postscript there is a graphics context, which would set various things such as color, line width and so on.

The purpose of the command `gsave` is to save the context. Then we select a new color, in this case black namely `red=0`, `green=0`, and `blue=0`, Then fill the path of the arrow with the color. The `grestore` restores the old graphics context. So every time a vector is drawn such commands are generated. So look at the figure in the file **molding.pdf**.

Here is the first part of the file:

```
%!PS
%%BoundingBox: 100 50 450 500
%%Creator: eg2ps.c by Jim Emery
%%EndComments
72 300 div 72 300 div scale
100 100 translate
```

```
3 setlinewidth
newpath
1000 1866 moveto
1000 1289 lineto
1500 1000 lineto
1500 1866 lineto
1000 1866 lineto
gsave
.5 1 1 setrgbcolor
fill
grestore
stroke
1500 1866 moveto
1625 1866 lineto
1625 928 lineto
1500 1000 lineto
1500 1866 lineto
gsave
.5 1 1 setrgbcolor
fill
grestore
stroke
1500 1000 moveto
1000 1289 lineto
1000 1866 lineto
1000 1000 moveto
1500 1000 lineto
1000 1000 moveto
1000 1289 lineto
1500 1000 moveto
1500 1866 lineto
1000 1289 moveto
500 1000 lineto
gsave
newpath
1500 1000 moveto
1494 1000 lineto
1494 1077 lineto
```

```

1482 1077 lineto
1500 1125 lineto
1518 1077 lineto
1506 1077 lineto
1506 1000 lineto
1500 1000 lineto
  0 0 0 setrgbcolor
  fill
  stroke
  grestore
  gsave
  newpath
1500 1000 moveto
1497 995 lineto
1430 1033 lineto
1424 1023 lineto
1392 1063 lineto
1442 1054 lineto
1436 1044 lineto
1503 1005 lineto
1500 1000 lineto
  0 0 0 setrgbcolor
  fill
  stroke
  grestore

```

I may take a picture of an actual joint being sawed, and glue it into the file **molding.pdf** with Acrobat.

## 7 Creating a Postscript Figure

Postscript files for figures are merged into Latex files using a Latex command. Here is the program that was used to create the figure in the document with the file name **molding.pdf**:

```

c   moldingf1.ftn figure for crownmolding.tex 4/13/15
implicit real*8(a-h,o-z)
dimension p1(3),p2(3), v1(2),v2(2),v3(2),v4(2)
dimension q1(3),q2(3),q3(3),q4(3),q5(3)
dimension t1(2),t2(2),t3(2),t4(2),t5(2),t6(2)
nfile=1
open(nfile,file='p.eg',status='unknown')
one=1.
pi=4.*atan(one)
vxmn=-1.
vxmx=1.
vymn=-1.
vymx=1.
call xvwpor(nfile,vxmn,vxmx,vymn,vymx)
xmn=-4.
xmx=4.
ymn=-4.
ymx=4.
call xwindo(nfile,xmn,xmx,ymn,ymx)
size = xmx-xmn
if(ymx-ymn .gt. size)size=ymx-ymn
phi=60.*pi/180
theta=45.*pi/180
c   draw corner
a=2.
b=a*tan(theta)
c   b/c=tan(phi)
c   c=b/tan(phi)
p1(1)=0.
p1(2)=c
p1(3)=-a
p2(1)=b
p2(2)=0.
p2(3)=0.

      if(1+1 .eq. 2)then
c fill large face of board in top view
t1(1)=0.

```

```

t1(2)=3.*c
t2(1)=0.
t2(2)=c
t3(1)=b
t3(2)=0.
t4(1)=b
t4(2)=3.*c
t5(1)=t4(1)+ .5
t5(2)=t4(2)
t6(1)=t3(1)+.5
t6(2)=t3(2)-.5*tan(pi/6.)
call xmove(nfile,t1(1),t1(2))
    call xdraw(nfile,t2(1),t2(2))
    call xdraw(nfile,t3(1),t3(2))
    call xdraw(nfile,t4(1),t4(2))
    call xdraw(nfile,t1(1),t1(2))
write(nfile,'(a)')'p gsave'
    write(nfile,'(a)')'p .5 1 1 setrgbcolor'
    write(nfile,'(a)')'p fill'
    write(nfile,'(a)')'p grestore'
    write(nfile,'(a)')'p stroke'

call xmove(nfile,t4(1),t4(2))
call xdraw(nfile,t5(1),t5(2))
call xdraw(nfile,t6(1),t6(2))
call xdraw(nfile,t3(1),t3(2))
call xdraw(nfile,t4(1),t4(2))

write(nfile,'(a)')'p gsave'
    write(nfile,'(a)')'p .5 1 1 setrgbcolor'
    write(nfile,'(a)')'p fill'
    write(nfile,'(a)')'p grestore'
    write(nfile,'(a)')'p stroke'

endif

```

```

c      draw vector from p2 to p1 in top view
      call xmove(nfile,p2(1),p2(2))
call xdraw(nfile, p1(1),p1(2))

      v1(1)=0.
      v1(2)=1.
      v1(3)=0.

c      draw top view

c      draw left board edge
      x=0.
y=c
call xmove(nfile,x,y)
x=0.
y=3*c
      call xdraw(nfile,x,y)
c draw small coordinate axis
      x=0.
y=0.
call xmove(nfile,x,y)
      x=b
      call xdraw(nfile,x,y)
      x=0.
y=0.
call xmove(nfile,x,y)
      y=c
      call xdraw(nfile,x,y)

c      draw right board edge
      x=b
y=0.
call xmove(nfile,x,y)
x=b

```

```

y=3.*c
  call xdraw(nfile,x,y)

c      draw left pointing edge of corner
      y=c
x=0
call xmove(nfile,x,y)
x=-2.*c*cos(30.*pi/180.)
y= c -2.*c*sin(30.*pi/180.)
  call xdraw(nfile,x,y)
c

h=.003*size

c      draw vector v1
x1=b
  y1=0.
alen=.5
call drawvector2d(nfile,x1,y1,pi/2.,alen,h)

c      draw vector v2
x=p2(1)
  y=p2(2)
c
  do i=1,2
    v2(i)=p1(i)-p2(i)
  end do
write(*,*)' v2= ',v2(1),v2(2)
v2n=sqrt(v2(1)**2+ v2(2)**2)
write(*,*)' length=',v2n
  do i=1,2
    v2(i)=v2(i)/v2n
  enddo
write(*,*)' v2= ',v2(1),v2(2)
angv2=atan2(v2(2),v2(1))
c      angv2=atan2(1d0,-1d0)
write(*,*)' ang=',angv2*180./pi

```

```

    alen=.5
    call drawvector2d(nfile,x,y,angv2,alen,h)
c
c    draw vector v4, normal to cut face
    x=(p1(1)+p2(1))/2
    y=(p1(2)+p2(2))/2
    ang= -120.*pi/180
    alen=.5
    call drawvector2d(nfile,x,y,ang,alen,h)

c    draw front view
c    origin point
c    zoffset of view
    zoff=1.
    q1(1)=0.
    q1(3)=0.-zoff
    write(*,*)' q1=',q1(1),q1(3)
    call xmove(nfile,q1(1),q1(3))
    q2(1)=b
    q2(3)=0.-zoff
    write(*,*)' q2=',q2(1),q2(3)
    call xdraw(nfile,q2(1),q2(3))
    q3(1)=0.
    q3(3)=-a -zoff
    write(*,*)' q3=',q3(1),q3(3)
    call xdraw(nfile,q3(1),q3(3))
    call xdraw(nfile,q1(1),q1(3))
c
    call xmove(nfile,q2(1),q2(3))
    q4(1)=q2(1)+ .5
    q4(3)=q2(3)
    write(*,*)' q4=',q4(1),q4(3)
    call xdraw(nfile,q4(1),q4(3))
    q5(1)=q3(1)
    q5(3)=q3(3)-.5
    write(*,*)' q5=',q5(1),q5(3)
    call xdraw(nfile,q5(1),q5(3))
    call xdraw(nfile,q3(1),q3(3))

```

```

c      write(nfile,'(a)')'p stroke'
c      fill board in front view
write(nfile,'(a)')'p newpath'
      call xmove(nfile,q2(1),q2(3) )
      call xdraw(nfile,q4(1),q4(3) )
      call xdraw(nfile,q5(1),q5(3) )
      call xdraw(nfile,q3(1),q3(3) )
      call xdraw(nfile,q2(1),q2(3) )
write(nfile,'(a)')'p gsave'
      write(nfile,'(a)')'p .5 1 1 setrgbcolor'
      write(nfile,'(a)')'p fill'
write(nfile,'(a)')'p stroke'
      write(nfile,'(a)')'p grestore'
      x=(q2(1)+q3(1))/2.
      y=(q2(3)+q3(3))/2.
      alen=.5
      ang=135.*pi/180.
call drawvector2d(nfile,x,y,ang,alen,h)
h=.002*size
x=0.
y=0.
ang=pi/2.
alen=1/3.
call drawvector2d(nfile,x,y,ang,alen,h)
ang=0.
alen=1/3.
call drawvector2d(nfile,x,y,ang,alen,h)
y=0.-zoff
ang=pi/2.
call drawvector2d(nfile,x,y,ang,alen,h)
ang=0.
call drawvector2d(nfile,x,y,ang,alen,h)
      end
c+ drawvector2d
      subroutine drawvector2d(nfile,x1,y1,ang,an,h)
      implicit real*8(a-h,o-z)
      dimension u(2),v(2)

```

```

        x2=x1+an*cos(ang)
        y2=y1+an*sin(ang)
        call drawarrow(nfile,x1,y1,x2,y2,h)
        return
    end

c
c+ drawarrow create arrow outline and postscript fill
    subroutine drawarrow(nfile,x1,y1,x2,y2,h)
        implicit real*8(a-h,o-z)
        dimension u(2),v(2)
        dimension p1(2),p2(2),p3(2),p4(2)
        dimension p5(2),p6(2),p7(2),p8(2)
        write(nfile,'(a)')'p gsave'
        write(nfile,'(a)')'p newpath'
c        x1,y1  coordinates of arrow tail
c        x2,y2  coordinates of arrow point
c        h      1/2 arrow shaft width
c        ahw    arrow head width
        ahw=4*h
c        ahl    arrow head length
        ahl=8*h
        theta = atan2(y2-y1,x2-x1)
        u(1)= cos(theta)
        u(2)= sin(theta)
        v(1) = -sin(theta)
        v(2)= cos(theta)
c        alen arrow length
        alen=sqrt((x2-x1)**2 + (y2-y1)**2 )
        p1(1)=x1
        p1(2)=y1
        p2(1)=p1(1)+ h*v(1)
        p2(2)=p1(2)+ h*v(2)
        p3(1)=p2(1)+(alen-ahl)*u(1)
        p3(2)=p2(2)+(alen-ahl)*u(2)
        p4(1)=p3(1)+ 2*h*v(1)
        p4(2)=p3(2)+ 2*h*v(2)
        p5(1)=x2
        p5(2)=y2

```

```

p8(1)=p1(1)- h*v(1)
p8(2)=p1(2)- h*v(2)
p7(1)=p8(1)+ (alen-ahl)*u(1)
p7(2)=p8(2)+ (alen-ahl)*u(2)
p6(1)=p7(1) - 2*h*v(1)
p6(2)=p7(2) - 2*h*v(2)
    call xmove(nfile,p1(1),p1(2))
    call xdraw(nfile,p2(1),p2(2))
    call xdraw(nfile,p3(1),p3(2))
    call xdraw(nfile,p4(1),p4(2))
    call xdraw(nfile,p5(1),p5(2))
    call xdraw(nfile,p6(1),p6(2))
    call xdraw(nfile,p7(1),p7(2))
    call xdraw(nfile,p8(1),p8(2))
    call xdraw(nfile,p1(1),p1(2))

    write(nfile,'(a)')'p 0 0 0 setrgbcolor'
    write(nfile,'(a)')'p fill'
    write(nfile,'(a)')'p stroke'
    write(nfile,'(a)')'p grestore'
end

```