What does it take to do reproducible computational science? What stands in our way?

Bill Rider
Computational Shock and Multiphysics, Department
Sandia National Laboratories
In conducting and documenting computational science research there are a number of distinct steps that are well defined and generally agreed upon. Among these steps would be defining the basic concept, literature review, derivation (and proofs where appropriate), implementation (i.e., coding in a programming/macro language C, C++, Fortran, Python, Matlab, Mathematica, etc.), debugging/testing, calculations, quality assurance work/V&V, writing and ultimate submission with associated peer review. To conduct “reproducible research” requires additional steps that would necessarily allow scrutiny of previously “private” steps in the research such as the details of a derivation, computer implementation, and the quality assurance process. Some of these steps have crept into publishing practice, as is the case with V&V. This additional scrutiny would then invite additional attention to the packaging and automation of steps that might have been heretofore much less formal. This would likely have the immediate impact of improving the manner in which this work in conducted. Moreover, the availability of these details would likely accelerate follow-on work and provide the basis for faster prototyping of extensions.

All of these impacts are generally positive, but must be countered with increased regulation of information for a variety of reasons including security-related concerns; export control/ITAR laws, intellectual property laws, proprietary information and the editorial policy of publications. Each of these provides a barrier of one sort or another for producing reproducible research that outstrips any of the technical challenges. These barriers are distributed and rather unevenly across organizations engaged in computational science research resulting in the specter of creating a culture of the “have’s” and the “have not’s” in reproducible computational science research.
Who Am I?

- I’m a staff member at Sandia, and I’ve been there SNL for 6 years. Prior to that I was at LANL for 18 years. I’ve worked in computational physics since 1992.
- In addition, I have expertise in hydrodynamics (incompressible to shock), numerical analysis, interface tracking, turbulence modeling, nonlinear coupled physics modeling, nuclear engineering…
- I’ve written two books and lots of papers on these, and other topics.
“Most daily activity in science can only be described as tedious and boring, not to mention expensive and frustrating.”

Steps in creating a computational science publication:

- The basic concept (innovation, question, application)
- Derivation and proof
- Implementation and debugging
- Testing and Results
- Writing
- Submission and peer review
- Does the work have life after the paper is published?

Issues and challenges: policy & culture issues within the scientific community and society at large.
What is the point and purpose of publishing? It is worth examining and being quite intentional

- **What is the point of the literature itself?**
  - Is everyone clear about this? does the educational system actually transmit the essence of the reasoning?
  - We are expected to do it, for status, promotion.
  - To expose ourselves to peer review
  - To communicate! to teach! and to learn!

- **What is the point of attending or presenting at meetings?**
  - Current thinking is troubling to say the least.
  - “To give a talk”
  - or is it to “communicate, speak and listen”
Journal of Computational Physics thoroughly treats the computational aspects of physical problems, presenting techniques for the numerical solution of mathematical equations arising in all areas of physics. The journal seeks to emphasize methods that cross disciplinary boundaries.

Elsevier’s reviewer guidance:

- **Methodology**: Is the methodology appropriate? Does it accurately explain how the data was collected? Is the design suitable for answering the question posed? Is there sufficient information present for you to replicate the research? Does the article identify the procedures followed? Are these ordered in a meaningful way? If the methods are new, are they explained in detail? Was the sampling appropriate? Have the equipment and materials been adequately described? Does the article make it clear what type of data was recorded; has the author been precise in describing measurements?

- **Results**: this is where the author/s should explain in words what he/she discovered in the research, any interpretation should not be included in this section. The results should be clearly laid out and in a logical sequence. You will need to consider if the appropriate analysis been conducted. Are the statistics correct? If you are not comfortable with statistics, advise the editor when you submit your report.
As examples, I’ll focus several of my own papers.

- The volume tracking paper is highly cited – 748 via Google Scholar
  - because of the tests it introduced).
  - The tests (i.e., V&V) are important and in one case became a bit of a tug-of-war with the editor and reviewers.

- Releasing code was achieved in one case, but has become increasingly problematic to virtually unthinkable.
  - The environment at the Lab is becoming less favorable towards (full) openness although it varies with the source of your support.
  - Some sponsors push or require openness, while others ignore it, while others object to it.
  - It may be impossible due to “security”
Why did we write “Reconstructing Volume Tracking”? 

- Volume tracking is an important methodology at LANL for computing multimaterial flows in the Eulerian frame.
- We wrote the paper because the standard way of coding up a volume of fluid method was so hard to debug.
  - We thought we had a better way to put the method together using computational geometry (i.e., a “toolbox”)
- Once the method was coded it needed to be tested:
  - Existing methods for testing these methods were poor
  - We came up with some new tests borrowed from the high-resolution methods community (combining the work of several researchers
    - Dukowicz’s vortex,
    - Smolarkiewicz’s deformation field and
    - Leveque’s time reversal)
The paper’s origin actually had a lot to do with how these methods were programmed.

Horrible computer code in F77 redacted due to legal concerns of my current (and former) employers. Probably because of the impact of the recent America Invents Act (patent law).

Notes:
1. The code has high cyclomatic complexity
2. The code is not extensible
3. The code is almost impossible to debug (see #1)
The logic goes on...

Continued redaction...

by the way there are two columns of 9 point Courier text, so it is a lot of code.
...and on...

More continued redaction of code.
I used to be able to show code in a talk... in a 2003 & 2006 talk when I was at LANL

5th Order WENO Code In icky Fortran

c.... Loop over edges and build stencil
Do i = 0, nc
  v(0:nc,-2:3) = u(0:nc,i-2:i+3) ! Local variables
  Call FLUXES_EULERO (v, g, -3, 3) F(U) -> f
  Call FLUX_SPLIT (v, g, gm, gp, -3, 3) R^{-1}F(U) -> f_+ + f_-
  c.... Do the stencil selection for f_+
    f_+ = \sum_{i=0}^{3} I_i \phi_i(w)
    Call WENO_5_SENSORS (gp, is, 0) (p=0)
    FP(p), F3s(p,0) = w(1)*f3s(p,:1) +w(2)*f3s(p,:2)
    c.... Do the stencil selection for f_-
    F3s(-p,0) = w(1)*f3s(-p,:1) +w(2)*f3s(-p,:2)
    Call WENO_5_SENSORS (gm, is, 1) (p=1)
    F3s(-p,1) = w(1)*f3s(-p,:1) +w(2)*f3s(-p,:2)
    Call FLUX_SPLIT (v, g, gm, gp, -2, 3) F(U) -> f_+ + f_-
    F3s(1,0) = w(1)*f3s(1,:1) +w(2)*f3s(1,:2)
    Call WENO_5_SENSORS (gm, is, 1) (p=2)
    F3s(-p,1) = w(1)*f3s(-p,:1) +w(2)*f3s(-p,:2)
    c.... Compute the raw variables
    Call L_DIFF_1K (w(:,), sm(k), sp(k), ib, ie)
    Call L_DIFF_2K (w(:,), pm(k), po(k), pp(k), ib, ie)
    c.... Start limiting
    Call Q_MP_2K (sm(k), sp(k), pm(k), po(k), pp(k), qm(k), qp(k))
    s(k) = Q_BOUND_2 (s(k), sm(k), sp(k), pm(k), po(k), pp(k))
    c.... Make the slope accurately monotone
    s(k) = X_LIM (s(k), sm(k), sp(k), qm(k), qp(k))
    End If
End Do

Issues with Codes - Smoothness Measures - 13th order

is(0) = 1258825940.D0*f(ic-g)**2 &
  +1667007831.D0*f(ic-g)*f(ic-e) &
  +55298463841.D0*f(ic-e)**2 &
  +46430779053.D0*f(ic-e)*f(ic-d) &
  +30856463663.D0*f(ic-d)**2 &
  +43141879360.D0*f(ic-d)*f(ic-c) &
  +69700140812.D0*f(ic-c)**2 &
  +46420200616.D0*f(ic-c)*f(ic-b) &
  +130158016620.D0*f(ic-b)**2 &
  +98517198380.D0*f(ic-b)*f(ic-a) &
  +59577469818.D0*f(ic-a)**2 &
  +397622832793.D0*f(ic-a)*f(ic-o) &
  +1119254208255.D0*f(ic-o)**2 &
  +794429785406.D0*f(ic-o)*f(ic-b) &
  +7254588877.D0*f(ic-g)**2 &
  +184521097818.D0*f(ic-g)*f(ic-e) &
  +521329653333.D0*f(ic-e)**2 &
  +79743892042.D0*f(ic-e)*f(ic-d) &
  +689449761463.D0*f(ic-d)**2 &
  +166904354747.D0*f(ic-d)*f(ic-c) &
  +5391528799.D0*f(ic-c)**2 &
  -362532756545.D0*f(ic-c)*f(ic-o) &
  +362532756545.D0*f(ic-o)**2

Issues with Codes - Smoothness Measures - 9th order

is(0) = 22658.D0*f(ic-d)**2 &
  +482663.D0*f(ic-d)*f(ic-b) &
  +1514936.D0*f(ic-b)**2 &
  +285007.D0*f(ic-b)*f(ic-a) &
  +1358240.D0*f(ic-a)**2 &
  +2462076.D0*f(ic-a)*f(ic-o) &
  +1020563.D0*f(ic-o)**2 &
  +411487.D0*f(ic-o)*f(ic-b) &
  +758623.D0*f(ic-b)**2 &
  +107918.D0*f(ic-b)*f(ic-a) &
  +60871.D0*f(ic-a)**2 &
  +337018.D0*f(ic-a)*f(ic-o) &
  +611976.D0*f(ic-o)**2 &
  +70237.D0*f(ic-o)*f(ic-b) &
  +337018.D0*f(ic-b)**2 &
  +18079.D0*f(ic-b)*f(ic-a) &
  +165513.D0*f(ic-a)**2 &
  +22658.D0*f(ic-a)*f(ic-o) &
Basic VOF Algorithm

Initial Data

1. Initial Data

2. Evolve Interface, using conservation form

3. Reconstruct interface, using conservation of mass

New Time Data

4.
Split versus Unsplit Time Integration

OLD Standard method

NEW research method

2 steps versus 1 step for a time step
“Logically all things are created by a combination of simpler, less capable components”

- Scott Adams (as Dogbert in the Dilbert Comic Strip), and noted by Culbert Laney in Computational Gasdynamics
Using Computational Geometry to Construct a VOF or Volume Tracking Method

An intersection is forced on this line

\[ n \]

\[ A = \frac{1}{2} \sum_{v=1}^{n} (x_v y_{v+1} - x_{v+1} y_v) \]

\[ A = \frac{\pi}{6} \sum_{v=1}^{n} (r_v + r_{v+1})(r_v z_{v+1} - r_{v+1} z_v) \]
We presented a serious rethink of the programming approach to these methods

“Beautiful” F77 computer code redacted due to legal concerns of my current and former employers.

Notes:
1. The code has low cyclomatic complexity
2. The code is extensible
3. The code is simple to debug (see #1)
We even included the code... with serious restrictions imposed by LANL

Subroutine INTERSECT (a1, rho1, a2, rho2, xi, yi, notparallel)
Implicit None
Include "param.h"
Logical notparallel
Real a1(1:2)
Real a2(1:2)
Real rho1
Real rho2
Real xi
Real yi
Real smdet ! small number for parallel line detection
Real det ! determinant of the linear system
smdet = Max (eps, smallvof * Abs(a1(1) * a2(2)),
& smallvof * Abs(a2(1) * a1(2)))
c.... first compute the determinant of the linear system
det = a1(1) * a2(2) - a2(1) * a1(2)
c.... if the determinant is approximately zero, the linear system is not solvable and we have parallel (approximately) lines.
If (Abs(det) .gt. smdet) Then
c..... nominal (nonparallel) case
xi = (rho1 * a2(2) - rho2 * a1(2)) / det
yi = (rho2 * a1(1) - rho1 * a2(1)) / det
notparallel = .true.
Else
  c....... set the flag to show that parallel lines have been found
  notparallel = .false.
End If
Return
End

As a condition of making the code available, I had to strip out most of the comments and formatting. this is just computational geometry!

This is just 1996, not the post-9/11/2001 World either!

I fought making the code this ugly to no avail.
What does that original part of the algorithm look like after the research?

```
c-----------------------------------------------------------------------
c.... Loop over x/r edges - and test flow direction

Do j = 1, nyz
   Do i = 1, nxr+1
      If (active(i,j) .or. active(i-1,j)) Then
         smallvel = softzero * (xrc(i) - xrc(i-1)) / dt

         If (velxr(i,j) .gt. smallvel) Then
            velmod = velxr(i,j) * dt / (one + aremap * gradvelxr(i-1,j))
            Call FLUX_VOL_XR(i, j, velmod, xr, yz)
         Else If (velxr(i,j) .lt. -smallvel) Then
            velmod = velxr(i,j) * dt / (one + aremap * gradvelxr(i,j))
            Call FLUX_VOL_XR(i, j, velmod, xr, yz)
         End If

         If (mixed(i-1,j)) Then
            fluxvofxr(i,j) = VOL_FLUX(xr, yz, 4, avof(1,i-1,j),
                                     avof(2,i-1,j), rhovof(i-1,j),
                                     axi)
         Else
            fluxvofxr(i,j) = vofin(i-1,j) * POLY_VOL(xr, yz, 4, axi)
         End If

         fluxvofxr(i,j) = fluxvofxr(i,j) * (one + adiv * divvel(i-1,j))
      Else If (velxr(i,j) .lt. -smallvel) Then
         fluxvofxr(i,j) = zero
      End If
   End Do
End Do
c-----------------------------------------------------------------------
Return
End
```
"...what can be asserted without evidence can also be dismissed without evidence.”
Christopher Hitchens

"Apart from the question of whether the simulation is telling us about the true solution or not, we must consider how much of its behavior we are prepared to see. What we see in a simulation may be biased strongly by what we expect to see.” Thomas P. Weissert, in *The Genesis of Simulation in Dynamics*. 
Tim Trucano’s (one of the father’s of V&V theory) observations on V&V...

- Key V&V themes have not changed “for decades”:
  - “Codes are not solutions, people are solutions.”
  - “Credibility of computational simulations for defined applications is evolutionary...”
  - “… at worst, credibility is non-existent in specific applications.”
  - “Single calculations will never be ‘the right answer’ for hard problems.”
  - “Real V&V and real UQ are a lot of work.”

- Trucano’s four insights on V&V:
  1. “V&V — pay me now or pay me later.”
  2. “Journal editorial policies and practices must change.”
  3. “Ask ‘What’s good enough?’”
  4. “Saying you don’t need verification is like saying you don’t need oxygen.”
“The purpose of computing is **insight**, not pictures” – Richard Hamming
This is the way validation is usually presented in the literature.

This is what you’ll see in most Journals. It is not quality control fit for moving forward.

This is how Homer does it.
Here is a notion of how a “converged” solution might be described.

With a third resolution convergence can be assessed, this is converged (~1\textsuperscript{st} order).
This sequence of meshes can be used to extrapolate the solution.

With three grids plus a convergence rate a converged solution can be estimated.
The experimental “error” has two components (observation & variability).
A great deal can be done to quantitatively assess the credibility of the calculations.
Why did this paper get cited so much?

Test Problems

Translation

Zalesak’s disc

Solid Body Rotation

Too Easy!

For Debugging

Vortex

Translation

Vortex

Deformation Field

Solid Body Rotation

\[ u = -\frac{\partial \Psi}{\partial y}, v = \frac{\partial \Psi}{\partial x} \]

\[ \Psi = \frac{1}{\pi} \sin^2(\pi x) \cos^2(\pi y) \]

\[ \Psi = \frac{1}{4\pi} \sin(4\pi(x + \frac{1}{2})) \times \cos(4\pi(y + \frac{1}{2})) \times \cos(\pi t/T) \]

J. Dukowicz produced the earliest example I found.

From R. Leveque

From P. Smolarkiewicz
## Convergence Rates for VOF with Time Reversal

<table>
<thead>
<tr>
<th>Grid</th>
<th>$T=0.5$</th>
<th>$T=2.0$</th>
<th>$T=8.0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$32^2-64^2$</td>
<td>2.36</td>
<td>2.01</td>
<td>2.78</td>
</tr>
<tr>
<td>$64^2-128^2$</td>
<td>1.86</td>
<td>2.16</td>
<td>2.27</td>
</tr>
<tr>
<td>$32^2-64^2$</td>
<td>1.62</td>
<td>0.81</td>
<td>1.52</td>
</tr>
<tr>
<td>$64^2-128^2$</td>
<td>1.95</td>
<td>0.91</td>
<td>0.84</td>
</tr>
</tbody>
</table>
Single Vortex: Front Tracking Solutions

32x32 grid

128x128 grid

solutions by Damir Juric
Deformation Field: PPIC Solutions

128x128 grid

solutions by Glenn Price
In fact 3-D Versions of these problems now exist thanks to Fedkiw, Enright, Ferzinger and Mitchell.

The new tests came from the level set community, who were originally quite resistant to these problems. In the process they have made their method a great deal better.

Moving to later work, Greenough & Rider formed the motivation for the later papers.

- **WENO5** is much more efficient for linear problems
- **PLMDE** is more efficient than **WENO5** on all nonlinear problems (with discontinuities)
- The advantage is unambiguous for Sod’s shock tube and the Interacting Blast Waves
- The advantage is less clear-cut for the “peak” problem
- At a given mesh spacing **WENO5** gives better answers for the Shu-Osher problem, but worse than **PLMDE** at fixed computational expense
Greenough & Rider (2005) provided quantitative errors for these problems.

<table>
<thead>
<tr>
<th>N</th>
<th>$E_{L_1}$</th>
<th>$L_1$ rate*</th>
<th>$E_{L_{\infty}}$</th>
<th>$L_{\infty}$ rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>8.22e-03</td>
<td>–</td>
<td>0.22e-00</td>
<td>–</td>
</tr>
<tr>
<td>200</td>
<td>4.48e-03</td>
<td>0.88</td>
<td>0.25e-00</td>
<td>-0.20</td>
</tr>
<tr>
<td>400</td>
<td>2.62e-03</td>
<td>0.77</td>
<td>0.33e-00</td>
<td>-0.37</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>N</th>
<th>$E_{L_1}$</th>
<th>$L_1$ rate**</th>
<th>$E_{L_{\infty}}$</th>
<th>$L_{\infty}$ rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>1.58e-02</td>
<td>–</td>
<td>0.37e-00</td>
<td>–</td>
</tr>
<tr>
<td>200</td>
<td>8.24e-03</td>
<td>0.93</td>
<td>0.40e-00</td>
<td>-0.01</td>
</tr>
<tr>
<td>400</td>
<td>4.47e-03</td>
<td>0.88</td>
<td>0.46e-00</td>
<td>-0.18</td>
</tr>
</tbody>
</table>

We plotted the errors as a function of position too. WENO is worse than PLMDE almost everywhere, but for a much greater computational expense.

In RGK2007 we combined the methods to correct these issues.

*We expect 0.75 in the limit of $\Delta x \rightarrow 0$ due to ABR2008

**We expect 0.86 in the limit of $\Delta x \rightarrow 0$ due to ABR2008
In some respects the code here was even more important than the previous example.

- This work arose out of a question of how efficient WENO methods were compared to older high-order Godunov methods for shock problems important and LANL & LLNL.
  - LLNL was using a code from Brown by Shu et al.
  - I had developed a code with both classes of method using the same philosophy.

- Including the code would have been more central to the validity of the scholarship, and less thinkable given previous experience...

- Time to solution was essential and of course this depends on the implementation.
Conclusions: Efficiency is also important! “High-Order must pay its way”

- Both methods are implemented in a similar manner (by me)
- For these 1-D problems WENO5 was about 6 times the cost of PLMDE for the same spatial resolution.
  - Two nonlinear differences per edge through the flux-splitting (~1.5X)
  - Multistage Runge-Kutta instead of forward-in-time (~3X)
  - Smaller CFL number (~1.5 X)
- Error/CPU - provides a real measure of efficiency
  - The code would seem to be quite important to the scholarship.
With regard to my latest published work...

- The work is still along the same general vein as previous efforts, numerical method development.
- The fact that the implementation of the research is directly into an export-controlled code makes publishing the code impossible under current law.
- Everything else is better too, but it can’t be shared!
Even if it's not classified, it falls into this Labyrinth.

**Unclassified Controlled Information (UCI)**

First, determine if your information is:

Sandia-owned = Sandia Proprietary

Then, based on content, what type of proprietary information it is:

- Employment Related Records
- Confidential Financial/LM Correspondence
- Procurement Actions
- Legal Records
- Technology Transfer
  - Such as:
    - IP license agreements
    - Protected CRADA Information
    - Certain intellectual property

U.S. Government-owned

Then, based on content/sponsor, what type of information it is:

- OUO
- UCNI
- AT
- SGI
- U-NNPI
- Other U.S. Gov’t Agency
  - Such as:
    - Dept. of Defense
    - Dept. of Homeland Security
    - Dept. of Transportation

PII

Personally Identifiable Information (PII) can apply to both U.S. government-owned and Sandia-owned information.

FOIA exemptions most commonly used at Sandia are 3-7.

Exemption 1. National Security Information
Exemption 2. Circumvention of Statute
Exemption 3. Statutory Exemption
Exemption 4. Commercial/Proprietary
Exemption 5. Privileged Information
Exemption 6. Personal Privacy
Exemption 7. Law Enforcement
Exemption 8. Financial Institutions
Exemption 9. Wells
“An expert is someone who knows some of the worst mistakes that can be made in his subject, and how to avoid them.”

- Werner Heisenberg
A Rogue’s Gallery of V&V Practice

Bill Rider

Computational Shock and Multiphysics Department
Sandia National Laboratories, Albuquerque, NM
wjrider@sandia.gov

MeV Summer School Lunch Talk
The status of V&V in the late 90’s.

- V&V was making inroads in professional societies.
  - AIAA, ASME, (J. Fluids Engr.)
  - JFE’s editorial statement was a critical moment (more later)
- The NRC had made validation and uncertainty integral to Reactor regulation
- Early work was being codified through Pat Roache’s publications bringing V&V to the “mainstream”
  - Important contributions by Oberkampf and Blottner (Sandia)
- Another key moment in the evolution toward V&V was the “CFD-vs-Wind Tunnel” debacle.
  - This is was enormously damaging to the community
  - Remarkably, DOE’s ASCI program made the same mistake 20 years later!
- V&V needs to be a collaborative endeavor that embraces both experimental and theoretical science as essential partners.
“Journal of Fluids Engineering disseminates technical information in fluid mechanics of interest to researchers and designers in mechanical engineering. The majority of papers present original analytical, numerical or experimental results and physical interpretation of lasting scientific value. Other papers are devoted to the review of recent contributions to a topic, or the description of the methodology and/or the physical significance of an area that has recently matured.”
“Although no standard method for evaluating numerical uncertainty is currently accepted by the CFD community, there are numerous methods and techniques available to the user to accomplish this task. The following is a list of guidelines, enumerating the criteria to be considered for archival publication of computational results in the *Journal of Fluids Engineering.*”

Then 10 different means of achieving this end are discussed, and a seven page article on the topic.
Excerpt from the editorial policy of JFE (digging even deeper, more fine print!)

“An uncertainty analysis of experimental measurements is necessary for the results to be used to their fullest value. Authors submitting papers for publication to this Journal are expected to describe the uncertainties in their experimental measurements and in the results calculated from those measurements and unsteadiness.”

- The numerical treatment of uncertainty follows directly from the need to assess the experimental uncertainty.

- I found that the treatment of uncertainty in either case to be uncommon.
“The Journal of Fluids Engineering will not consider any paper reporting the numerical solution of a fluids engineering problem that fails to address the task of systematic truncation error testing and accuracy estimation. Authors should address the following criteria for assessing numerical uncertainty.”

It's difficult to find language this strong for other publications, it's also clear that this policy is not uniformly implemented.
So, What is the path forward?

- Publishing should serve its proper role in the conduct of science – communication
- Complete documentation of computational should include code used to demonstrate algorithms or compute results
- Numerous challenges exist with respect to policy largely dependent on the source of support and your employer (or customer/funding agency)
  - Intellectual property law and security concerns provide distinct barriers.
- Any policy should be thought through with regard to unintended consequences.
  - Could this become a wedge issue between communities of scientists that have worked well in the past?
Putting the current milieu into perspective

There must be no barriers to freedom of inquiry ... There is no place for dogma in science. The scientist is free, and must be free to ask any question, to doubt any assertion, to seek for any evidence, to correct any errors – J. Robert Oppenheimer

During the Manhattan Project in WWII Oppenheimer and Gen. Leslie Groves fought about scientific openness at Los Alamos where Groves wanted compartmentalization of information. Oppenheimer ultimately prevailed.

Where I work Groves appears to be winning the argument.
A final (and happier, but cautionary) note!

“… what were the causes of the flowering of applied mathematics in America after World War II? Perhaps the most important factor was the war itself, which demonstrated to all the crucial importance of science and technology for such projects as radar, the proximity fuse, code breaking, submarine hunting, and the atomic bomb. Mathematicians, working along with physicists, chemists, and engineers, made substantial and in some cases decisive contributions; without these developments, the United States might have lost the war. ”

From THE FLOWERING OF APPLIED MATHEMATICS IN AMERICA, by Peter Lax, SIAM Review, December 1989