SAND2014-15354PE

July 10, 2014

What is worth reproducing in computational science?

Bill Rider

Computational Shock and Multiphysics, Department

Sandia National Laboratories

Sandia National Laboratories is a multi program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

SAND2014-?????C

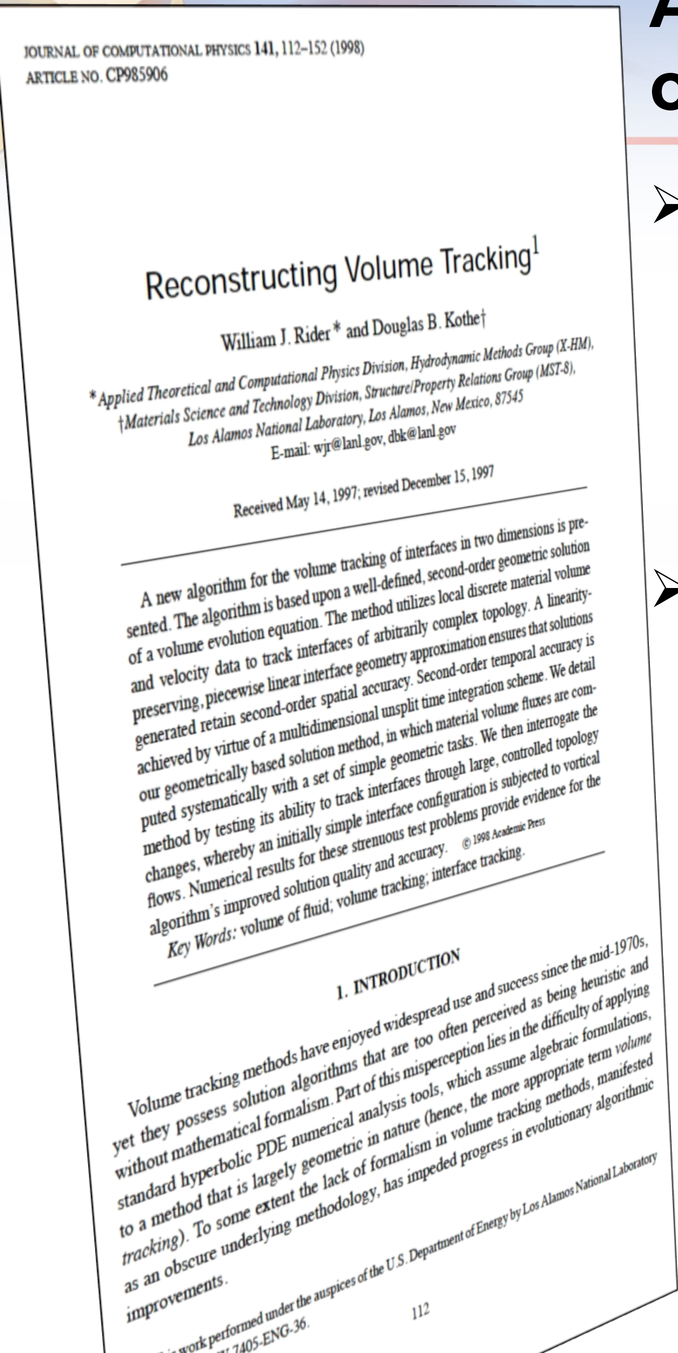
“The road to hell is paved with good intentions.”

Outline

- We are already deluged with publications and now even more data will be available.
  - ✓ Is this necessarily a good thing?
  - ✓ Might a “pull” model work better than a “push”?
- What sort of things will this precipitate?
- What has already happened? Lessons from the history of computational science.
- Has applied math already withdrawn in some ways? Why might I say this?

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 **141**, 112–152 (1998)
ARTICLE NO. CP985906

Reconstructing Volume Tracking[1]

William J. Rider* and Douglas B. Kothe†

*Applied Theoretical and Computational Physics Division, Hydrodynamic Methods Group (X-HM),
†Materials Science and Technology Division, Structure/Property Relations Group (MST-8),
Los Alamos National Laboratory, Los Alamos, New Mexico, 87545
E-mail: wjr@lanl.gov, dbk@lanl.gov

Received May 14, 1997; revised December 15, 1997

A new algorithm for the volume tracking of interfaces in two dimensions is presented. The algorithm is based upon a well-defined, second-order geometric solution of a volume evolution equation. The method utilizes local discrete material volume and velocity data to track interfaces of arbitrarily complex topology. A linearity-preserving, piecewise linear interface geometry approximation ensures that solutions generated retain second-order spatial accuracy. Second-order temporal accuracy is achieved by virtue of a multidimensional unsplit time integration scheme. We detail our geometrically based solution method, in which material volume fluxes are computed systematically with a set of simple geometric tasks. We then interrogate the method by testing its ability to track interfaces through large, controlled topology changes, whereby an initially simple interface configuration is subjected to vortical flows. Numerical results for these strenuous test problems provide evidence for the algorithm's improved solution quality and accuracy. © 1998 Academic Press

Key Words: volume of fluid; volume tracking; interface tracking.

1. INTRODUCTION

Volume tracking methods have enjoyed widespread use and success since the mid-1970s, yet they possess solution algorithms that are too often perceived as being heuristic and without mathematical formalism. Part of this misperception lies in the difficulty of applying standard hyperbolic PDE numerical analysis tools, which assume algebraic formulations, to a method that is largely geometric in nature (hence, the more appropriate term *volume tracking*). To some extent the lack of formalism in volume tracking methods, manifested as an obscure underlying methodology, has impeded progress in evolutionary algorithmic improvements.

[1] This work performed under the auspices of the U.S. Department of Energy by Los Alamos National Laboratory under contract W-7405-ENG-36.

112

As examples, I'll focus several of my own papers.

- The volume tracking paper is highly cited – 998 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"

Rider & Kothe, *J. Comp. Phys.*, **141**, 1998 (RK1998).

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)

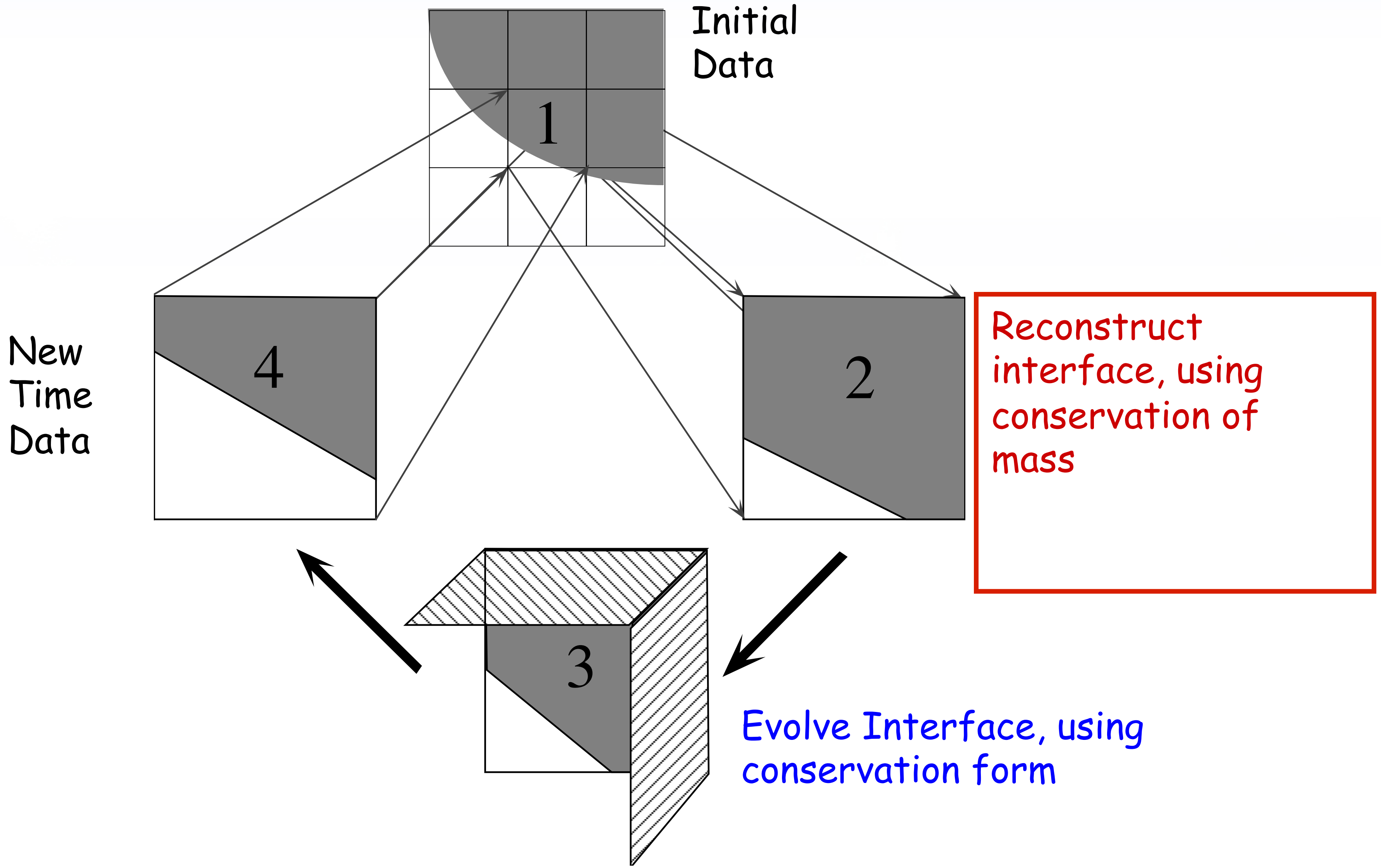
The paper's origin actually had a lot to do with how these methods were programmed.

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).

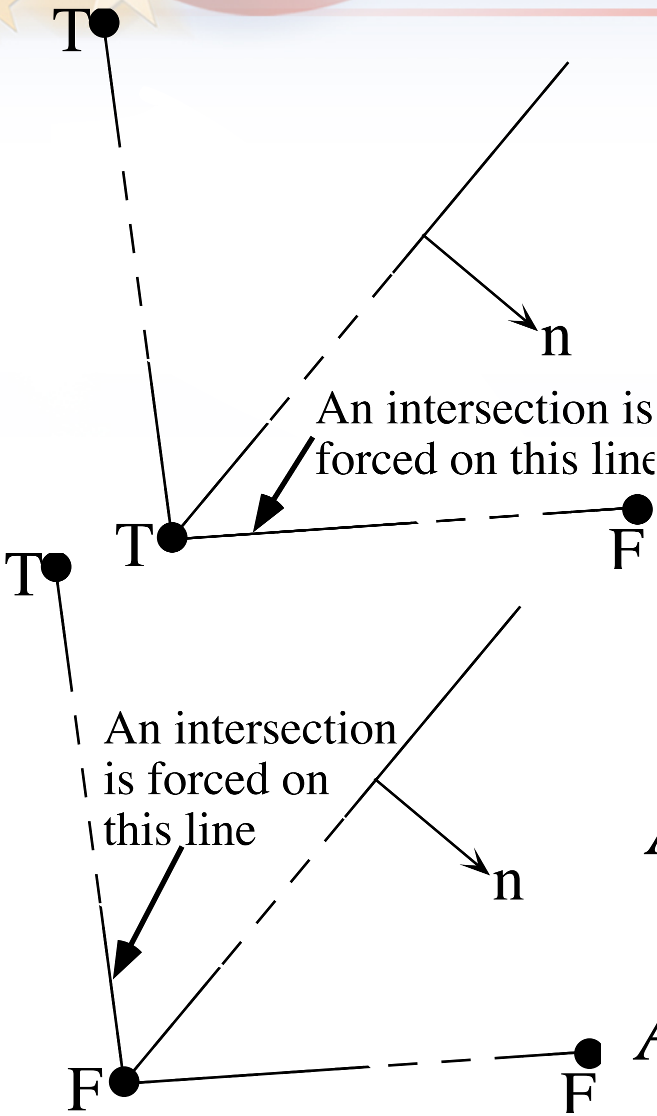
Notes:

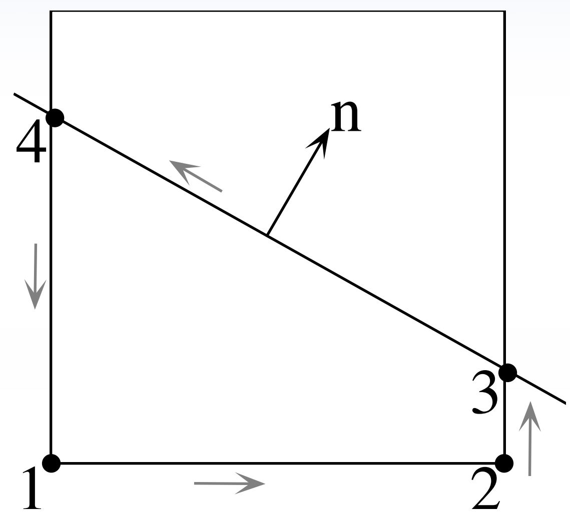
1. The code has high cyclomatic complexity
2. The code is not extensible
3. The code is almost impossible to debug (see #1)

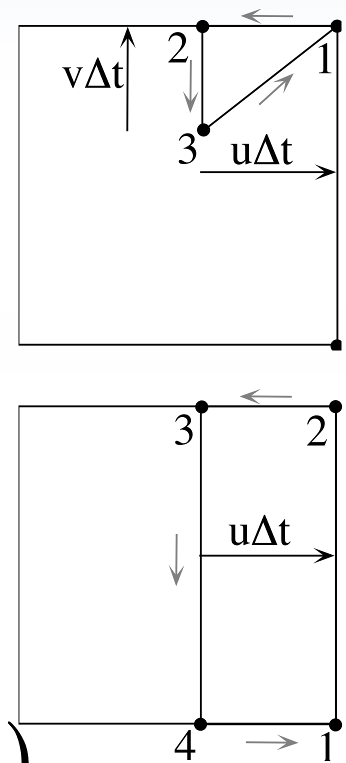
Basic VOF Algorithm



Using Computational Geometry to Construct a VOF or Volume Tracking Method







Fluxes

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

$$A = \frac{\pi}{6}\sum_{v=1}^{n}\left(r_v + r_{v+1}\right)\left(r_v z_{v+1} - r_{v+1} z_v\right)$$

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

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 1997, not the post-9/11/2001 World
-or-
post economic crisis
World either!

I fought making the code this ugly to no avail.

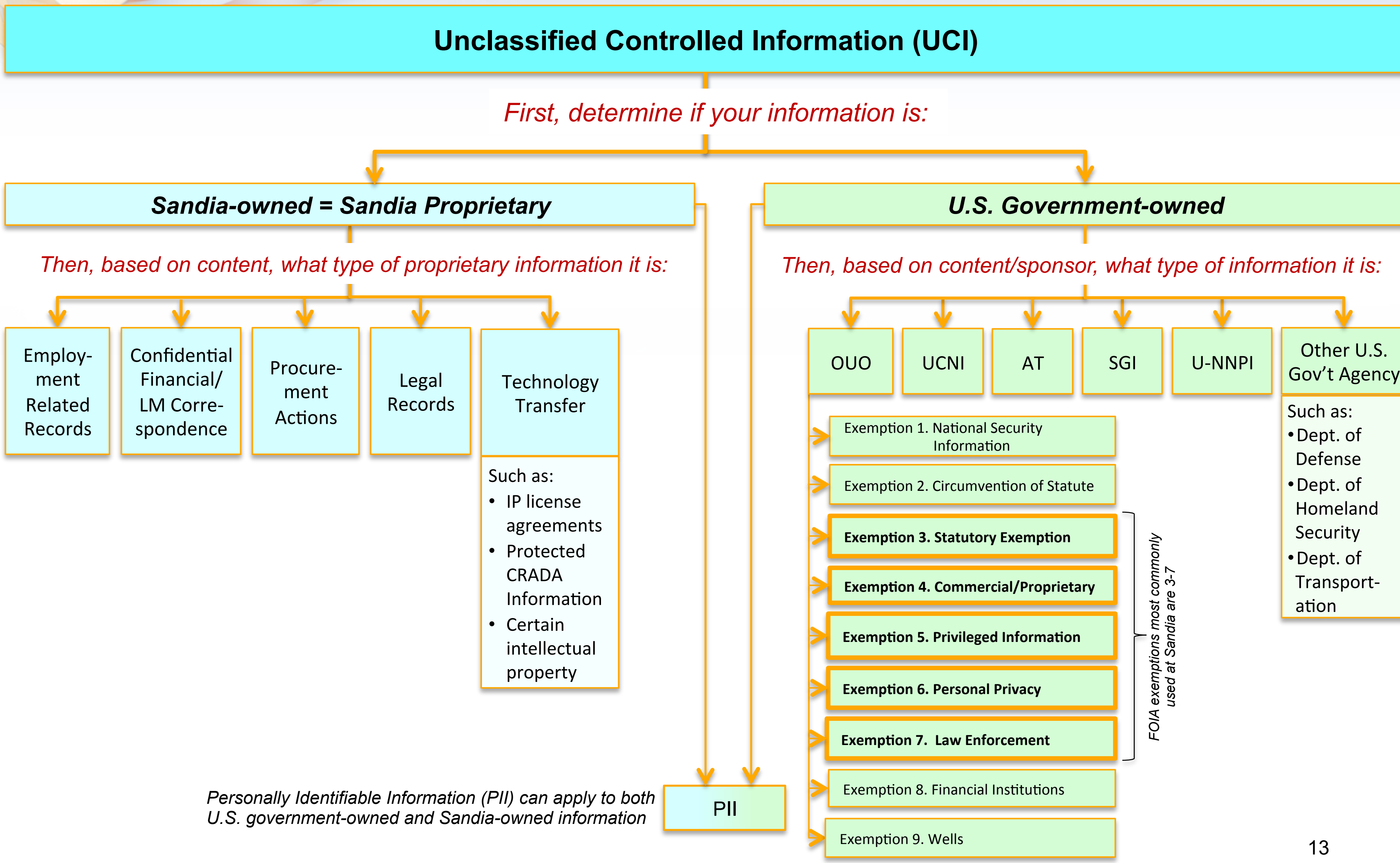
```
Subroutine INTERSECT (a1, rho1, a2, rho2, xi, yi, notparallel)
      Implicit None
      Include "param.h"
      Logical notparallel
      Real a1(1:2)
      Real a2(1:2)
      Real rho1
      Real rho2
      Real xi
      Real yi
      Real smdet          ! small number for parallel line
                     ! 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
c.... 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
```

What does that original part of the algorithm look like after the research?

[redacted]

[redacted]

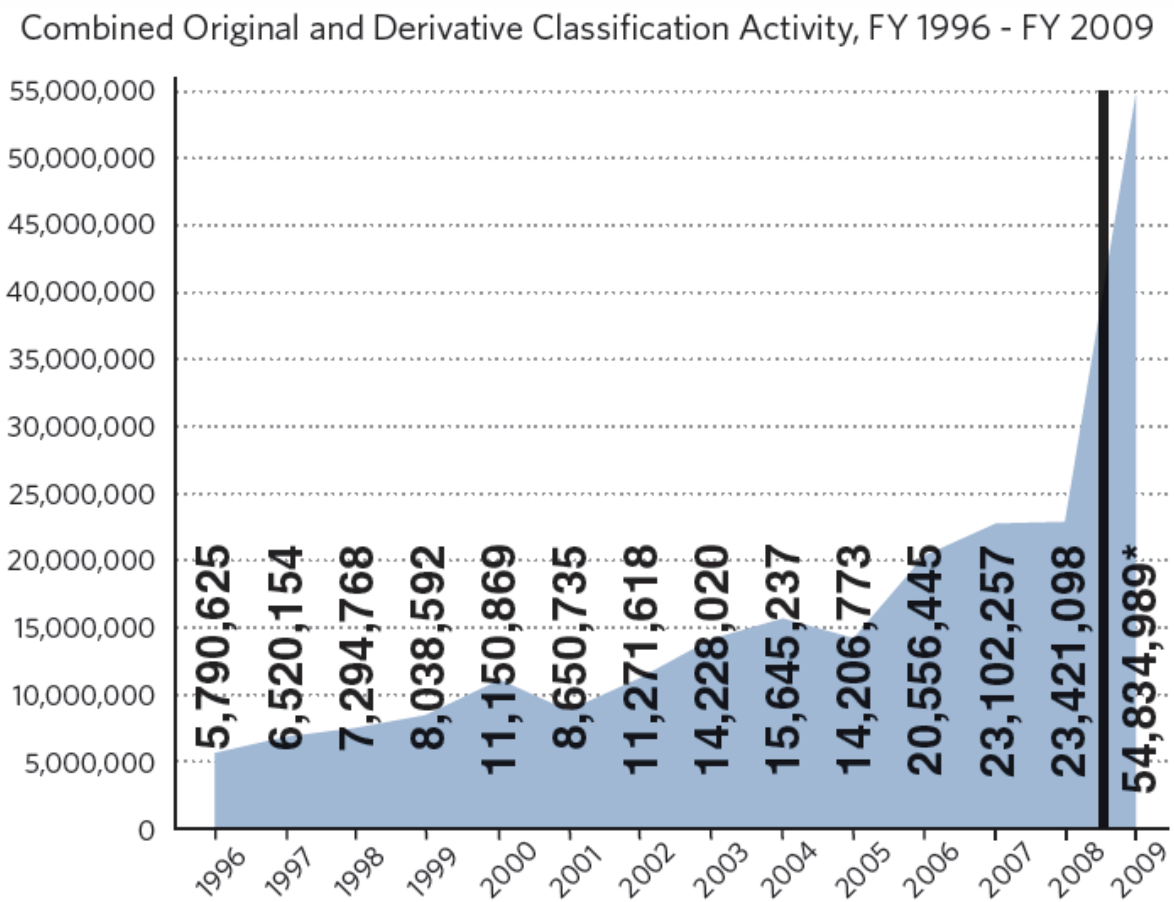
Even if its not classified it falls into this Labyrinth.



More problems: Classified vs Export Control

- **Penalty for releasing classified information**
  - ✓Up to 10 years, and unspecified fines
  - ✓The classification guidance is technical and voluminous
  - ✓Well-defined, well understood, well administered
- **Penalty for violating export control**
  - ✓Up to 20 years imprisonment and $1,000,000
  - ✓The guidance is non-technical and virtually non-existent
  - ✓Ill to poorly defined, but very large threats to the people administering the system

The number of classified documents is growing

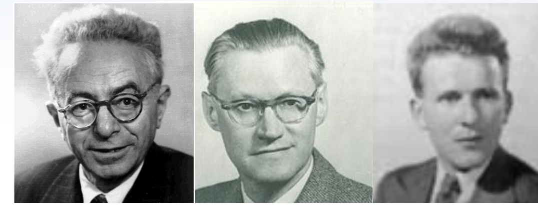


The dramatic increase between FY 2008 and FY 2009 derivative classification totals reflects ISOO's issuance of revised guidance concerning the counting of classification actions.

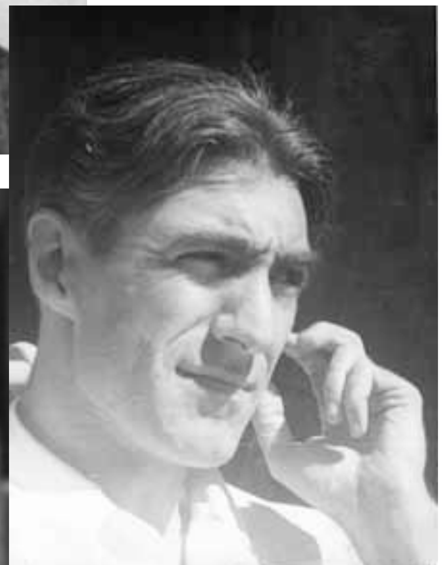
CFD was developed by many great minds

John Von Neumann

Lord Rayleigh & G. I. Taylor

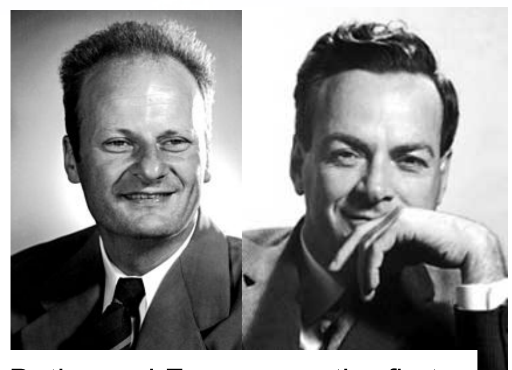


Courant, Friedrichs, Lewy – 1928 paper



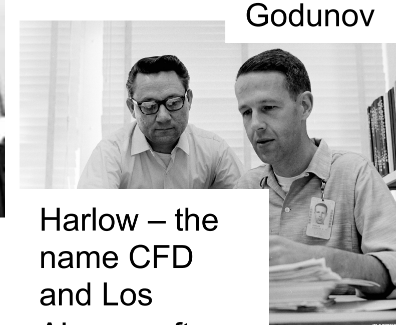
Robert Richtmyer

Peter Lax

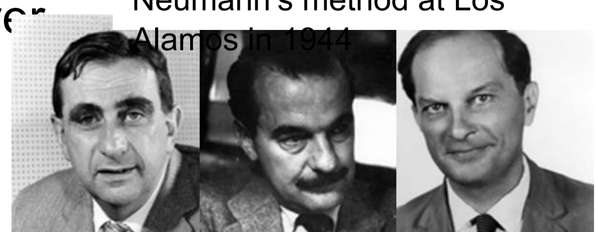


Bethe and Feynman – the first calculations using Von Neumann's method at Los Alamos in 1944

Godunov



Harlow – the name CFD and Los Alamos often



Teller, Metropolis, Ulam – Monte Carlo Methods and the H-Bomb

Landshoff & Rosenbluth

The origin of hydrodynamic calculations

- The first hydro calculation was reported in a Los Alamos report on June 20, 1944 – lead author Hans Bethe
  - ✓Feynmann was the calculational lead
  - ✓They used two methods to compute shocks, but only one of them worked (the shock fitting). The other finite difference method failed catastrophically!

- The first codes were 1-D and Lagrangian, shocks were tracked (no viscosity, finite differences failed completely till 1948).
- Von Neumann developed a simple finite difference method at Aberdeen and published a report on March 20, 1944.

The artificial viscosity paper by Von Neumann and Richtmyer, J. Appl. Phys. 1950

A Method for the Numerical Calculation of Hydrodynamic Shocks

J. VonNeumann and R. D. Richtmyer
Institute for Advanced Study, Princeton, New Jersey
(Received September 26, 1949)

The equations of hydrodynamics are modified by the inclusion of additional terms which greatly simplify the procedures needed for stepwise numerical solution of the equations in problems involving shocks. The quantitative influence of these terms can be made as small as one wishes by choice of a sufficiently fine mesh for the numerical integrations. A set of difference equations suitable for the numerical work is given, and the condition that must be satisfied to insure their stabilty is derived.

I. INTRODUCTION

In the investigation of phenomena arising in the flow of a compressible fluid, it is frequently desirable to solve the equations of fluid motion by stepwise numerical procedures, but the work is usually severely complicated by the presence of shocks. The shocks manifest themselves mathematically as surfaces on which density, fluid velocity, temperature, entropy and the like have discontinuities; and clearly the partial differential equations governing the motion require boundary conditions connecting the values of these quantities on the two sides of each such surface. The necessary boundary

(but preferably somewhat larger than) the spacing of the points of the network. Then the differential equations (more accurately, the corresponding difference equations) may be used for the entire calculation, just as though there were no shocks at all. In the numerical results obtained, the shocks are immediately evident as near-discontinuities that move through the fluid with very nearly the correct speed and across which pressure, temperature, etc. have very nearly the correct jumps.

It will be seen that for the assumed form of dissipation (and, indeed, for many others as well), the Rankine-Hugoniot equations are satisfied, provided the thick-

LA-671 a precursor to the Von Neumann-Richtmyer paper. By Richtmyer (only!)

Classified till 8/26/93. In the period right after WWII almost all Lab reports were classified.

LA-671

Series A

[redacted]

9 March 1948

PROPOSED NUMERICAL METHOD FOR CALCULATION OF SHOCKS

Work Done By:

R. D. Richtmyer

Report Written By:

R. D. Richtmyer

The details and conception of artificial viscosity is different, and it is called a fictitious viscosity instead. It is less a "pressure" and closer to an additional viscous term. The form more closely follows the true entropy generation term from thermodynamics.


[redacted]

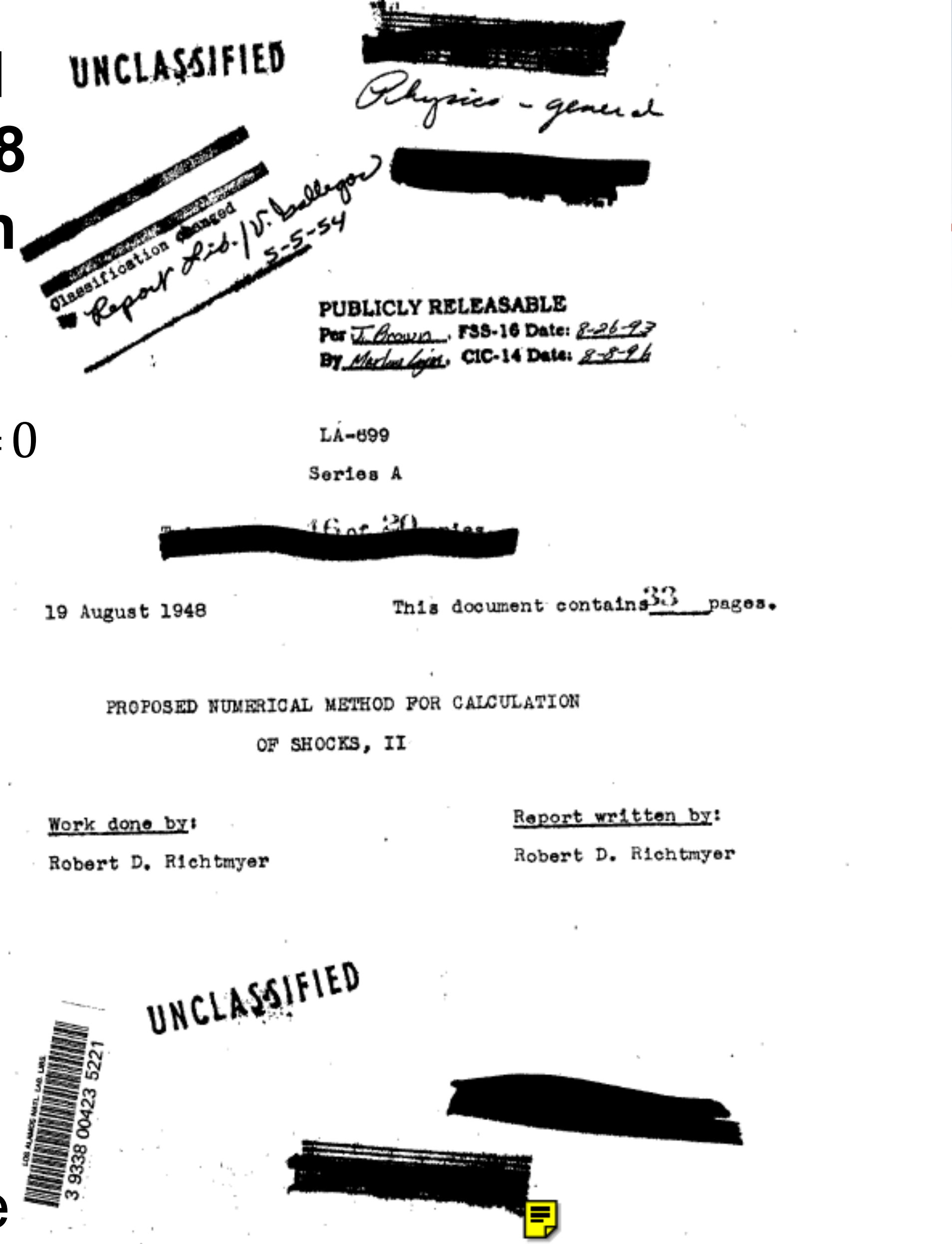
UNCLASSIFIED

Richtmyer published a second report five months later in 1948 (March to August) reporting on numerical experiments.

$$\frac{\partial u}{\partial t}+\frac{\partial}{\partial m}\left(p+q\right)=0\rightarrow\frac{\partial u}{\partial t}+\frac{\partial}{\partial m}\left(p+\mu\frac{\partial u}{\partial x}\right)=0$$

$$T\Delta S=-\tfrac{1}{6}G\frac{1}{c^2}\left(\frac{\Delta V}{V}\right)^3\rightarrow$$

$$T\Delta S=\mu\left(\frac{\partial u}{\partial x}\right)^2\rightarrow\mu\propto\left(\Delta x\right)^2\left|\frac{\partial u}{\partial x}\right|$$



UNCLASSIFIED

[redacted]

Physics - general

[redacted]

[redacted] Classification changed [redacted] Report Lib. / V. Gallegos 5-5-54

PUBLICLY RELEASABLE
Per J. Brown, FSS-16 Date: 8-26-93
By Marlene Lujan, CIC-14 Date: 8-8-96

LA-699

Series A

[redacted]

19 August 1948

This document contains 33 pages.

PROPOSED NUMERICAL METHOD FOR CALCULATION OF SHOCKS, II

Work done by:
Robert D. Richtmyer

Report written by:
Robert D. Richtmyer

UNCLASSIFIED

[redacted]

[redacted]

He uses both the term “fictitious” and “mock” to describe the term, But not “artificial”. All of these are unfortunate in their connotation.

“V&V takes the fun out of computational simulation”

– Tim Trucano

Definitions: Verification and Validation

- **ASC(I): Advanced Simulation and Computing Program**
  - ✓**Verification** → Verification is the process of determining that a computational software implementation correctly represents a model of a physical process.
  - ✓**Validation** → Validation is the process of determining the degree to which a computer model is an accurate representation of the real world from the perspective of the intended model applications.
- **Close to the DMSO, ASME and AIAA definitions.**

- **Alternative for computational science and engineering:**
  - ✓**Verification** = Accumulating evidence that the equations are solved correctly.
  - ✓**Validation** = Accumulating evidence that the equations are correct *for the intended application*

Definitions Continued:

- **Calibration** = **"The process of adjusting numerical or physical modeling parameters in the computational model for the purpose of improving agreement with experimental data." (AIAA Guide)**
- **Code** = **everything that goes into producing the final numbers, unless I'm speaking about "Code Verification," in which case I mean the particular software.**
- **Comments:**
  - ✓ **Calibration** *is not* **validation**, especially for predictive applications.
  - ✓ **Validation** is defined to be be dependent on the intended application.
  - ✓ In the sense of "solution accuracy," **verification** is dependent on the intended application.

Definitions Continued:

- **<u>Calibration</u> = "The process of adjusting numerical or physical modeling parameters in the computational model for the purpose of improving agreement with experimental data." (AIAA Guide)**
- **Comments:**
  - ✓ **Calibration** *is not* **validation**, especially for predictive applications.
  - ✓ **Validation** is defined to be be dependent on the intended application.

“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

So, What is the path forward?

- **Publishing should serve it's 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.

Who is winning today?

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

Who Am I ?

- I’m a staff member at Sandia, and I’ve been there SNL for 7 1/2 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.