CHEP 2012

21-25 May 2012

New York City, NY, USA

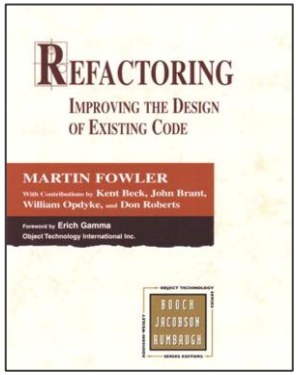
Refactoring, reengineering and evolution: paths to Geant4 uncertainty quantification and performance improvement

Maria Grazia Pia

M. Batič, M. Begalli, M. Han, S. Hauf, G. Hoff, C. H. Kim, M. Kuster, P. Saracco, H. Seo, G. Weidenspointner, A. Zoglauer

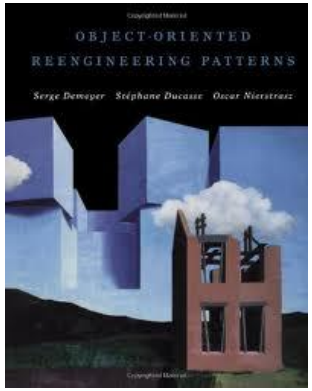
INFN Genova, Italy
State University of Rio de Janeiro (UERJ), Brazil
Hanyang University, Seoul, Korea
Darmstadt Tech. Univ., Germany
PUCRS, Porto Alegre, Brazil
EU XFEL GmbH, Hamburg, Germany
MPI Halbleiterlabor, Munich, Germany
Space Sciences Laboratory, UC Berkeley, USA

“The code slowly sinks from engineering to hacking”.



“**Refactoring** is the process of changing a software system in such a way that it does not alter the external behavior of the code yet improves its internal structure.”

“When you refactor you are improving the design of the code after it has been written.”



Reengineering “seeks to transform a legacy system into the system you would have built if you had the luxury of hindsight and could have known all the new requirements that you know today.”

“A legacy is something *valuable* that you have *inherited*.”
“With rapid development tools *and* rapid turnover in personnel, software systems can turn into legacies more quickly than you might imagine.”

evolution

1. one of a set of prescribed movements
2. a process of change in a certain direction
3. the process of working out or developing
4. the historical development of a biological group
5. the extraction of a mathematical root
6. a process in which the whole universe is a progression of interrelated phenomena

The lantern

State of the art?

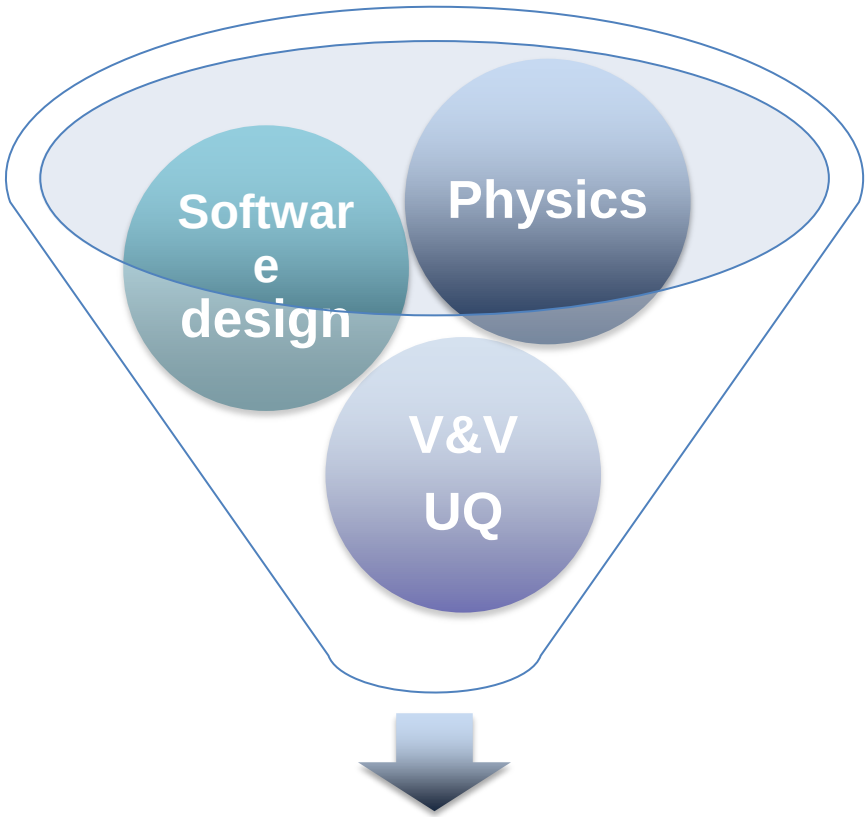
State of the art
in the physics for Monte Carlo particle transport

Quantified simulation

Focus on **physics** and **fundamental concepts** of particle transport

Theoretical and technological challenges

Produce concrete deliverables



State-of-the-art, quantified

Geant 4

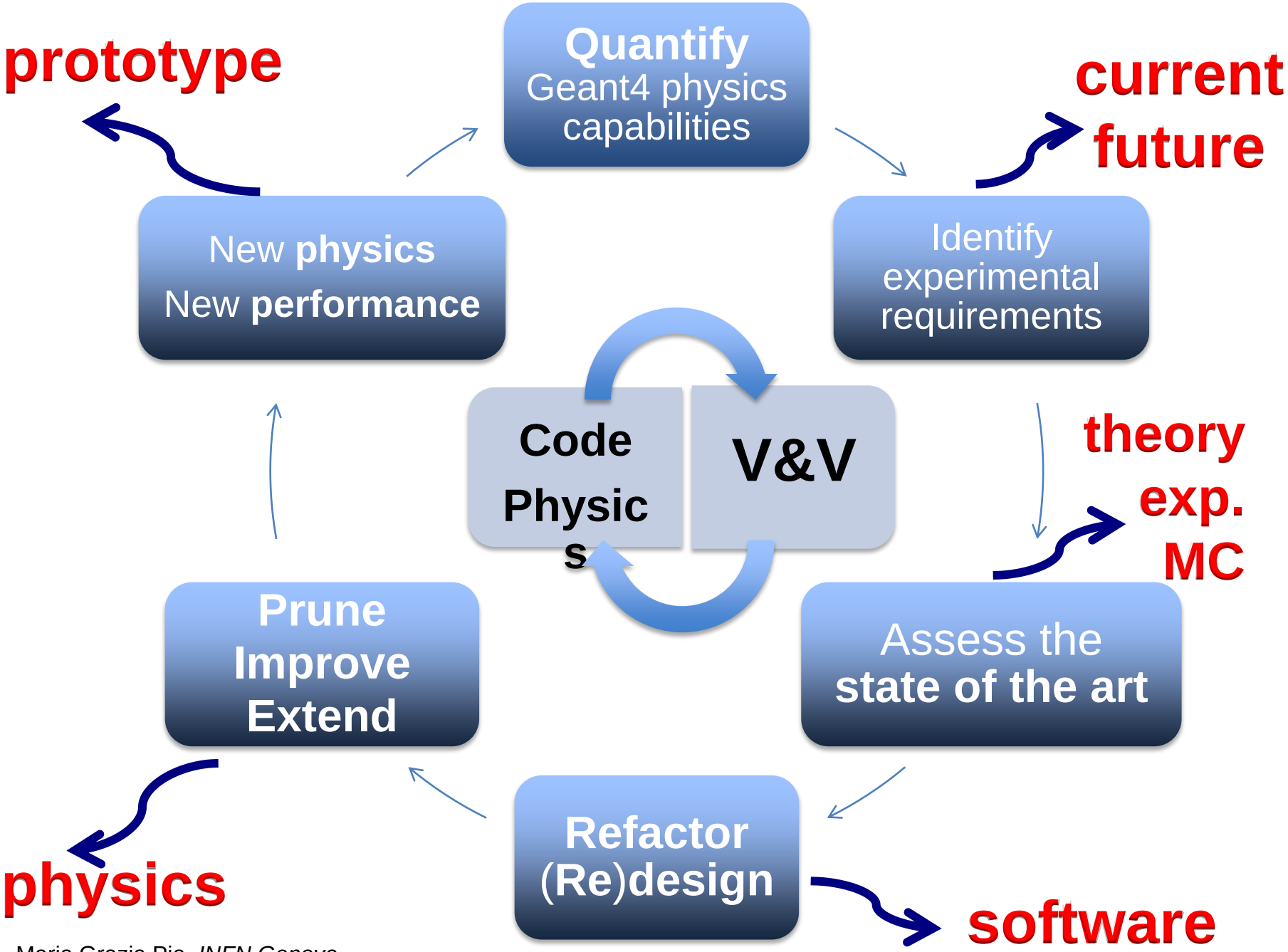
R&D Project

Series of
pilot projects
going on since 2008

Highlights

No time to go into details

Archival literature



Maria Grazia Pia, *INFN Genova*

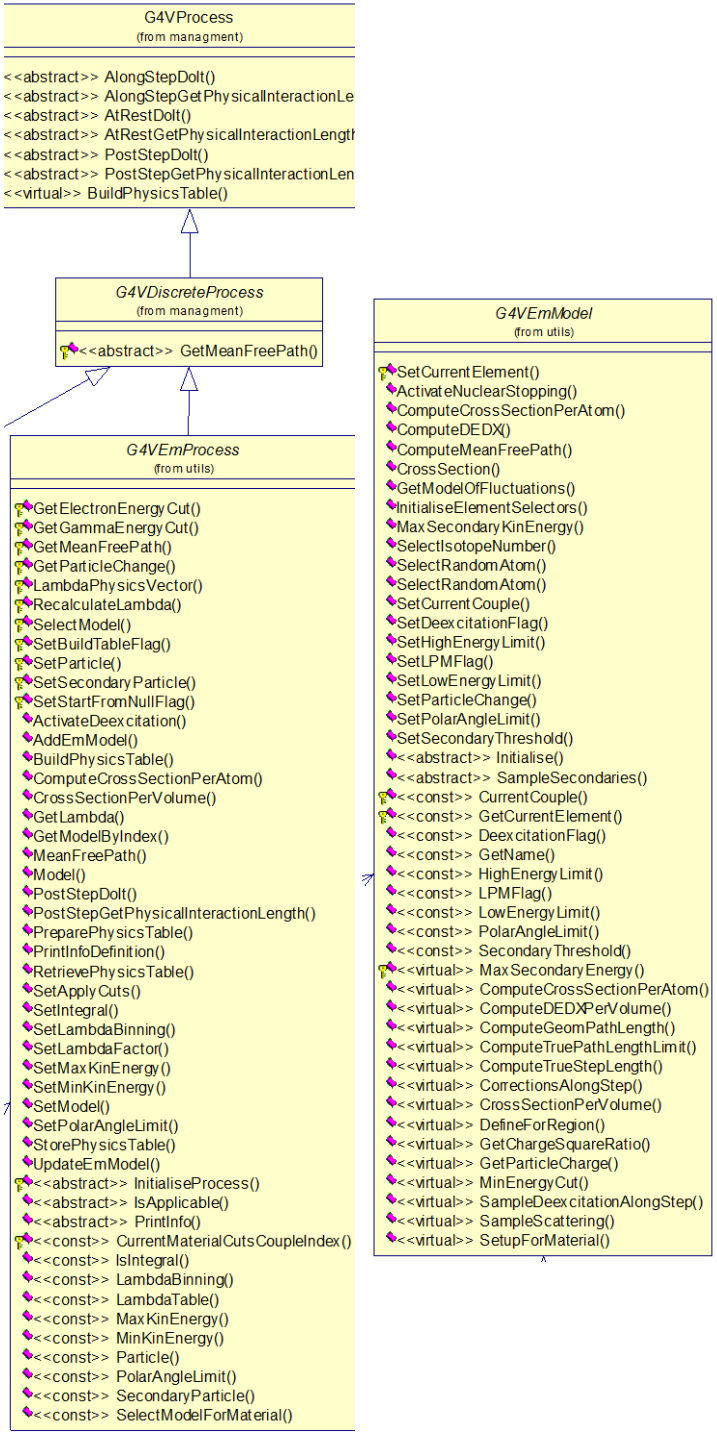
Smells

If it stinks, change it.

Grandma Beck, discussing child-rearing philosophy

M. Fowler, K. Beck et al.,
Refactoring: Improving the Design of Existing Code

- Duplicated Code
- Long Method
- Large Class
- Long Parameter List
- Divergent Change
- Shotgun Surgery
- Feature Envy
- Data Clumps
- Primitive Obsession
- Switch Statements
- Parallel Inheritance Hierarchies
- Lazy Class
- Speculative Generality
- Temporary Field
- Message Chains
- Middle Man
- Inappropriate Intimacy
- Alternative Classes with Different Interfaces
- Incomplete Library Class
- Data Class
- Refused Bequest



Maria Grazia Pia, *INFN Genova*

Electromagnetic smells

Coupling

σ_{tot} and final state modeling have been decoupled in hadronic physics design since RD44

"model"

Total cross section

Whether a process occurs

Final state generation

How a process occurs

Dependencies

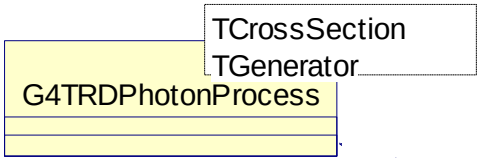
on other parts of the software

One needs a geometry
(and a full scale application)
to test (verify) a cross section

Difficult to test ➔ no testing
often

Problem domain analysis
Improve domain decomposition

Policy-based class design



Benefits

Transparency

Ease of **maintenance**

Simplicity of **testing** for V&V

> *Numera ciò che è numerabile, misura ciò che è misurabile,*
> *e ciò che non è misurabile rendilo misurabile.*
>
> Galileo Galilei (1564-1642)

Basic physics V&V can be performed by means of **lightweight unit tests**

Exploring new models (calculations) is made easier

Quantification of accuracy is facilitated

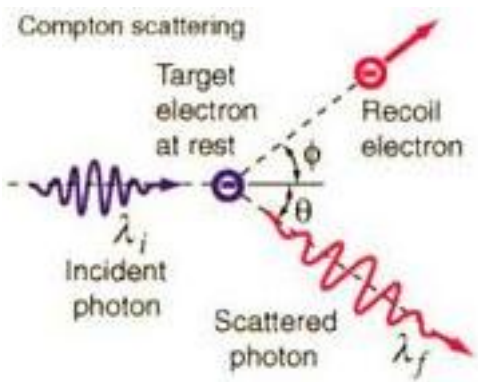
QED ≠ QCD

Distinguish genuine modeling alternatives from evolving degree of complexity

Electromagnetic processes in particle transport:

- Final state to be generated is well identified
- **Theory**, rather than **model**
- Various degrees of refinement in theoretical calculations
  - e.g. electron at rest, scattering functions, Compton profiles
- Experimental data for validation are (in general) available

Bare-bone physical functionality



Shift from shopping list of alternatives

Decorations on top

Motion of atomic electrons
Doppler broadening

Binding effects

scattering functions

Doppler profiles

Vacancy creation
Atomic relaxation

An example:
Photon elastic scattering

State of the art

Form factor approximation:
non relativistic, relativistic, modified + anomalous scattering factors

2nd order S-matrix calculations
recent calculations, not yet used in Monte Carlo codes

Quantification

Statistical analysis, GoF + categorical

Differential cross sections

	Penelope 2001	Penelope 2008	EPDL	Relativ. FF	Non-Rel. FF	Modified FF	MFF ASF	RFF ASF	SM NT
ε	0.27	0.38	**0.38**	0.25	0.35	0.49	0.52	0.48	**0.77**
error	±0.05	±0.06	±0.06	±0.05	±0.06	±0.06	±0.06	±0.06	±0.05

ε = fraction of test cases compatible with experiment, 0.01 significance

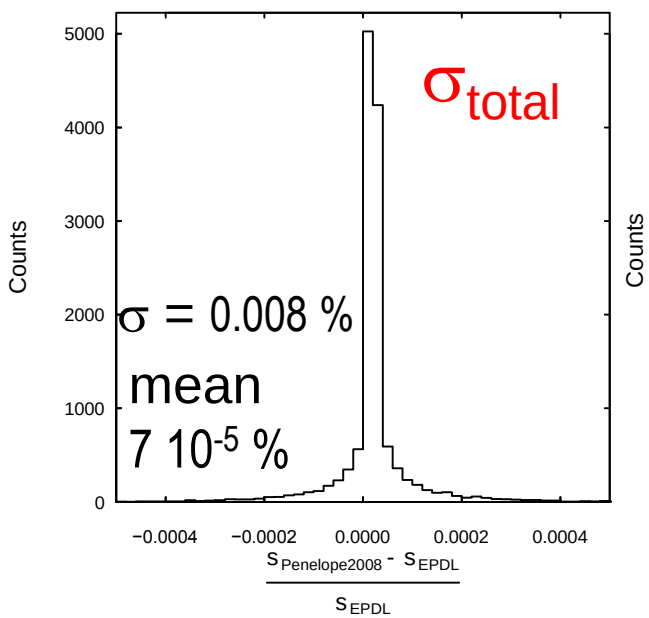
Prune

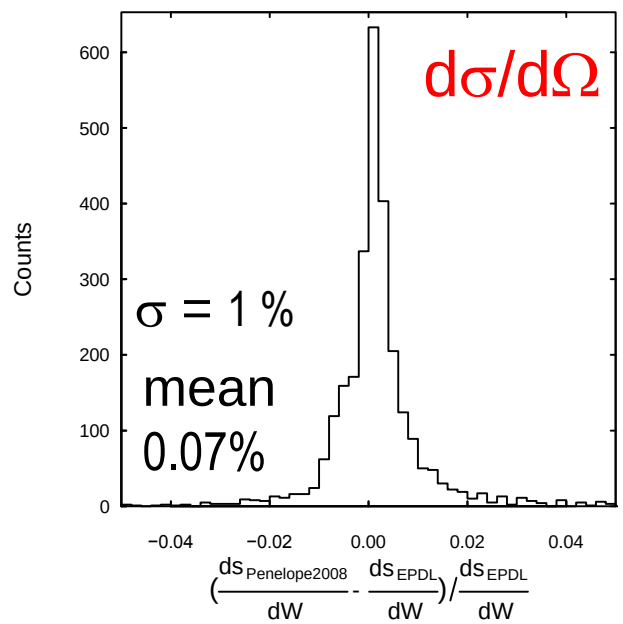
Number one in the stink parade is duplicated ~~code~~ physics

M. Fowler, *Refactoring*

Objective quantification of smell

Two Geant4 models, identical underlying physics content *(it used to be different)*





Efficiency w.r.t. experiment

"Livermore"	Penelope
EPDL97	EPDL97
0.38±0.06	0.38±0.06

Code bloat

Burden on

- Software design
- Maintenance
- User support

Unnecessary complexity

Maria Grazia Pia, *INFN Genova*

Bremsstrahlung, evaporation, proton elastic scattering etc.

Trash
and redo

Number one in the stink parade is duplicated ~~code~~ numbers

1. Bearden & Burr (1967)
2. Carlson
3. EADL
4. Sevier
5. ToI 1978 (Shirley)
6. ToI 1996 (Larkins)
7. Williams

Atomic binding energies

Geant 4 { Carlson + Williams / EADL (*Carlson* / *Shirley*)

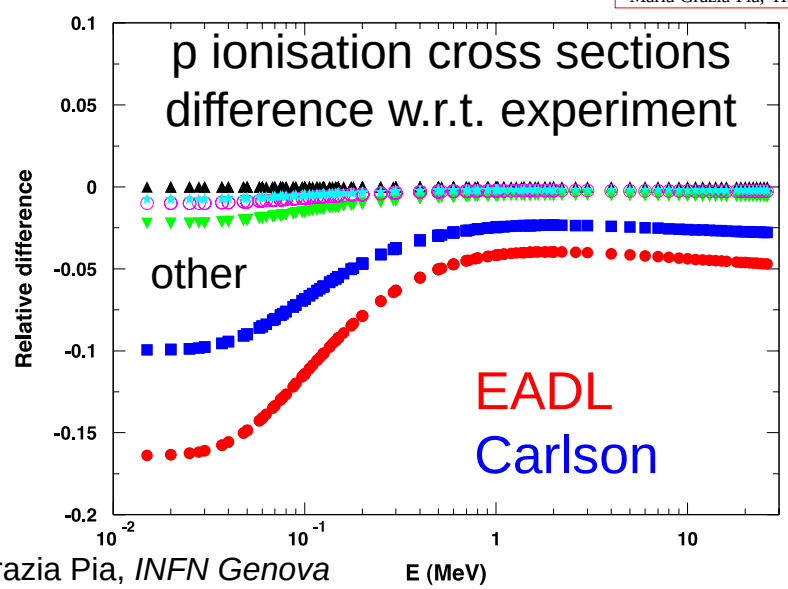
Source of epistemic uncertainties?

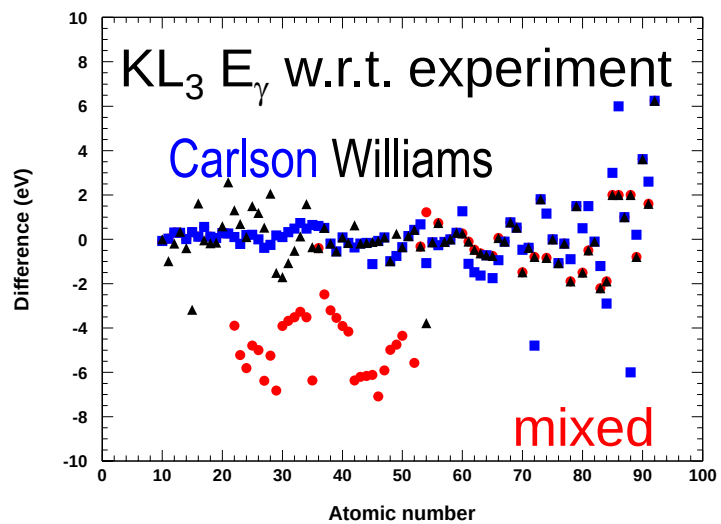
3246 IEEE TRANSACTIONS ON NUCLEAR SCIENCE, VOL. 58, NO. 6, DECEMBER 2011

Evaluation of Atomic Electron Binding Energies for Monte Carlo Particle Transport

Maria Grazia Pia, Hee Seo, Matej Batic, Marcia Begalli, Chan Hyeong Kim, Lina Quintieri, and Paolo Saracco

23 pages





Algorithms

Popular belief

Physics model X is intrinsically slow

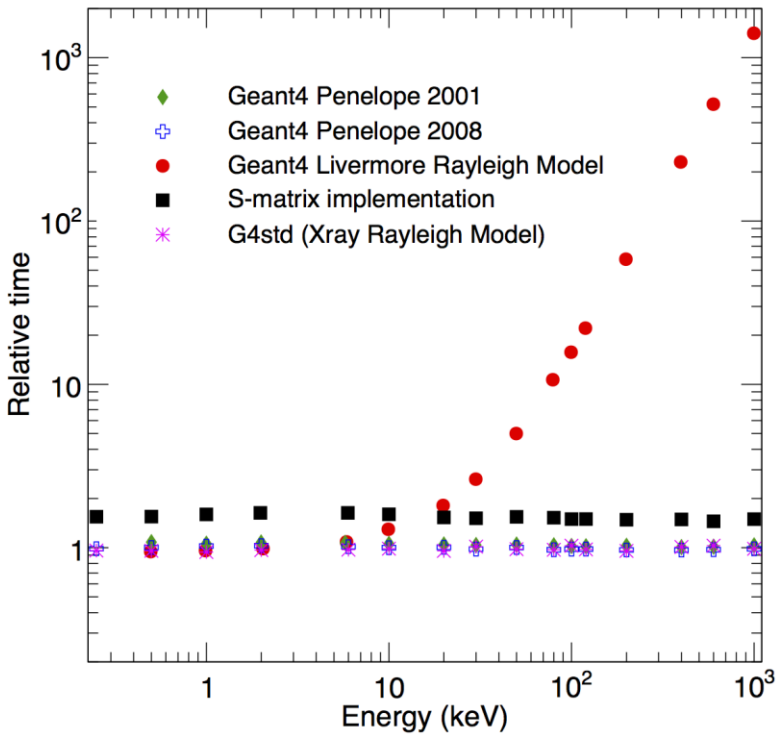
Baroque methods to combine it with "faster" lower precision models and limit its use to cases where one is willing to pay for higher precision

This design introduces an additional computational burden due to the effects of inheritance and the combination algorithms themselves

Truth

Physics model X is intrinsically fast

But its computationally fast physics functionality is spoiled by an inefficient sampling algorithm



- **No code smell**
- Spotted through in-depth **code review** in the course of **software validation**

Change the sampling algorithm!

The fastest algorithm

no algorithm at all

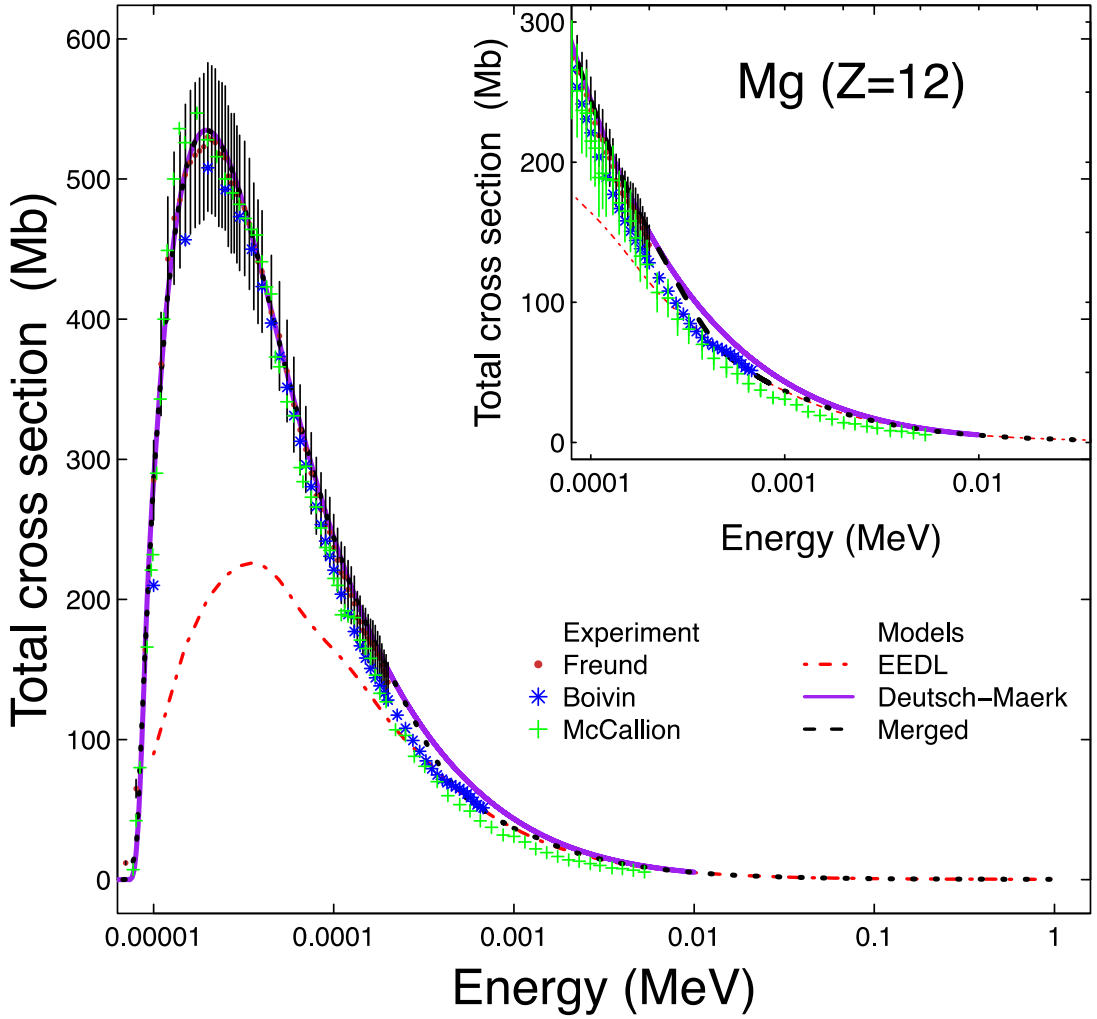
Shift modeling from algorithms to data

Merging models

Smoothing data

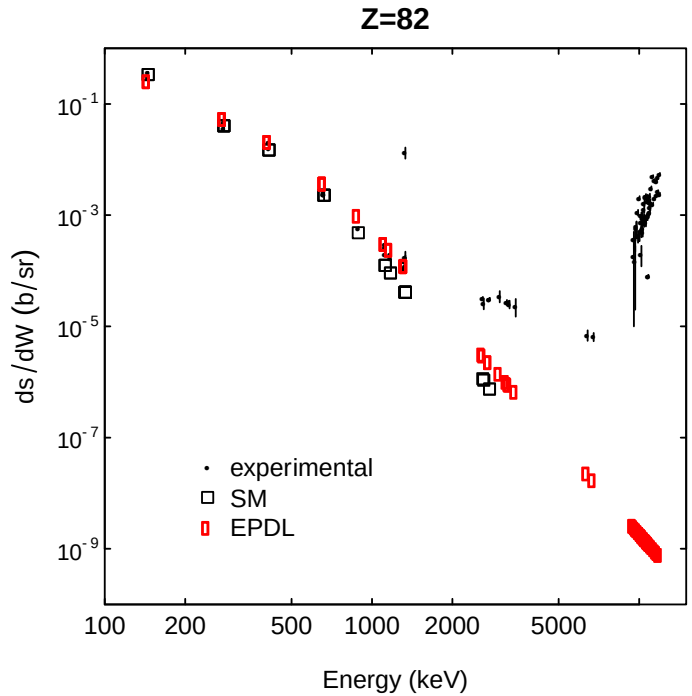
Guidance from experimental data *(when available)*

Electron impact ionisation cross sections



Example: LOESS local polynomial regression fitting
Beware: not optimized! Mathematical support in progress

…no silver bullet



Photon elastic scattering cross section at 90°

Experimental data by Schumacher et al., Starr et al., Jackson et al., Moreh et al. etc., complete list in a forthcoming publication

Smoothing models (or data) that exhibit significantly different behaviour is not physically justified

Here our understanding of the underlying physics phenomena fails

Experimental data themselves must be analyzed for inconsistencies and possible systematics

How to solve this kind of problems?

Better theoretical calculations
New, dedicated experimental data

No easy solution

Data libraries

Experimentally **validated**
Quantified accuracy
State of the art

- PIXE (proton ionisation cross sections)
- BEB – DM (electron ionisation cross sections)
- Photon elastic
  - RTAB (Lynn Kissel) trimmed and reformatted for MC use

R&D in progress

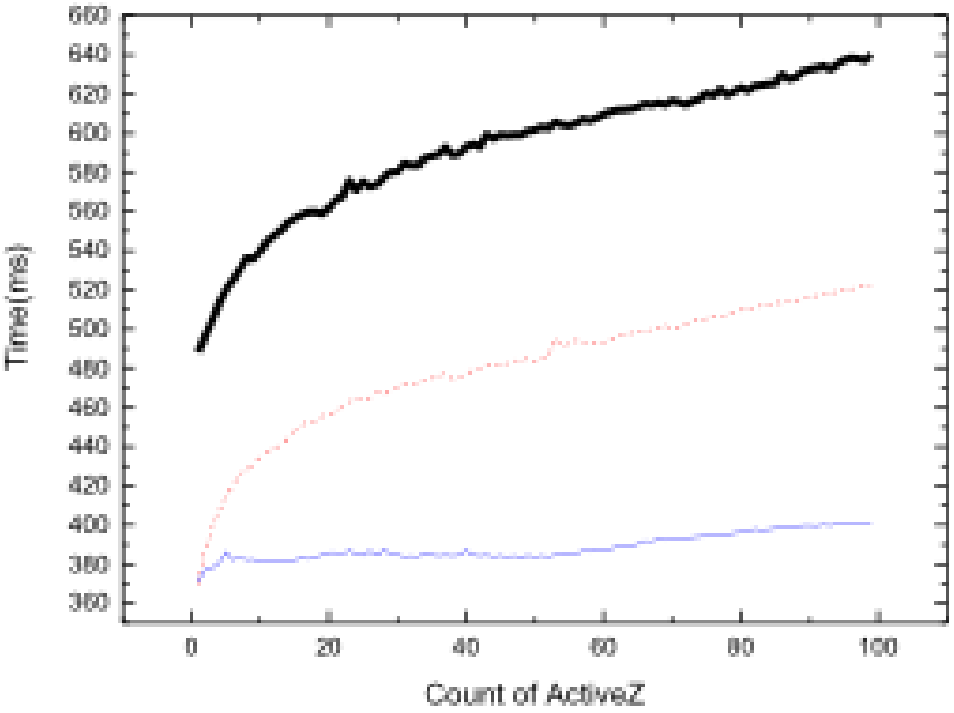
- Interpolation algorithms
- Smoothing techniques
- Data management methods

Mathematical expertise

Refactoring data management

- Today's technology
  - …keeping an eye on the new C++ Standard

- Optimal container
- Pruning data
- Splitting files
- Software design

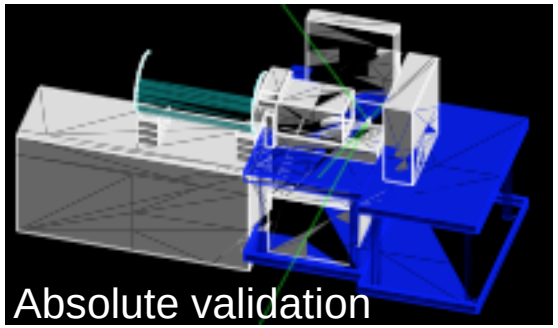


Big refactoring

Geant4 Radioactive Decay

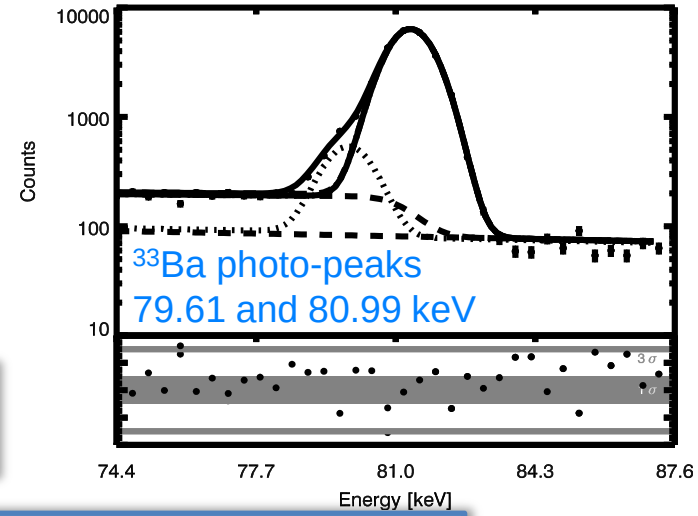
Well defined
responsibilities
and **interactions**

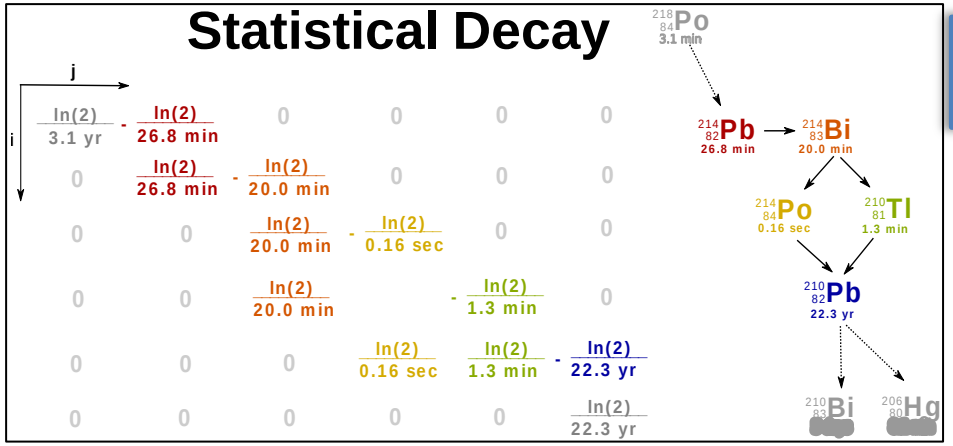
Maria Grazia Pia, *INFN Genova*



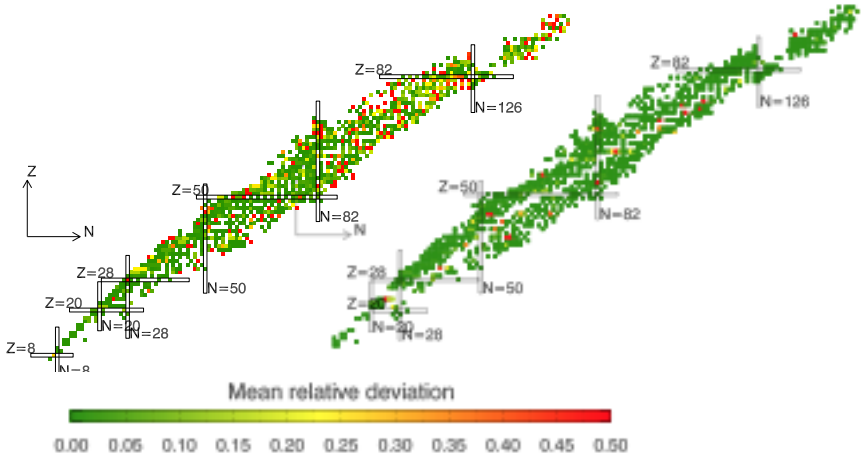
Experiment: Z. W. Bell *(ORNL)*

Validation

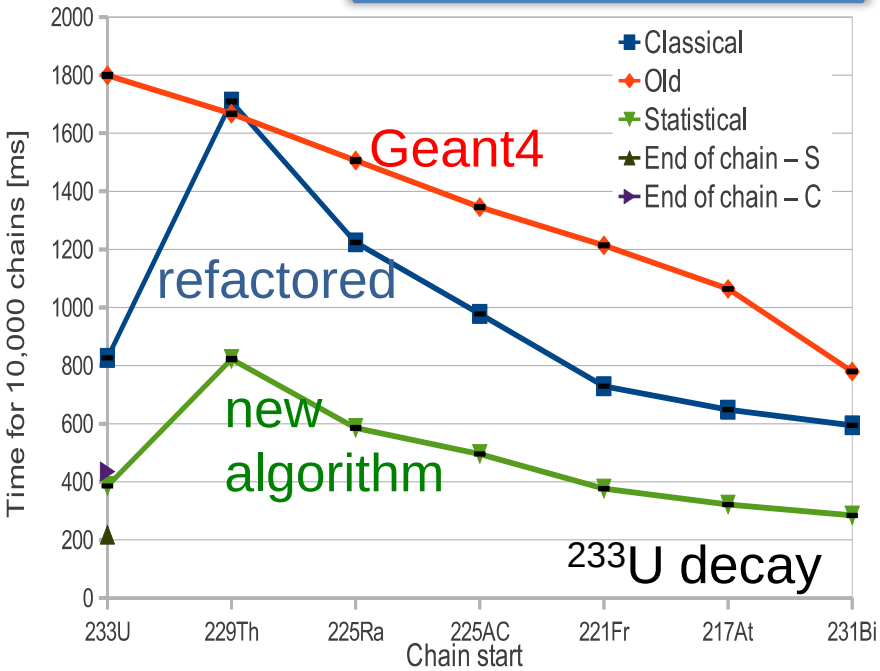




New algorithm



Performance



Conclusions

A quantitatively validated, state-of-the-art Monte Carlo code is at reach

a tremendous challenge
an inspiring motivation

Large effort
Supported by software design

Refactoring techniques and reengineering patterns contribute to improve **computational performance** and facilitate **validation**

Physics insight is the key