Minisymposia “Étude de la qualité numérique de code de calcul industriels”

• Christophe Denis (EDF R&D)

Etude de la qualité numérique de codes de calculs industriels : problématique, premiers résultats et perspectives

• Jean-Luc Lamotte (UPMC-LIP6)
L’approche probabiliste pour la validation de logiciels numériques

• Philippe Langlois (DALI –UPVD,LIRMM)
Performance des algorithmes précis

• Sethy Montan (EDF R&D)
Implémentation efficace de CADNA dans les bibliothèques de calcul et de communications

• Questions aux orateurs

The scientific community should always be able to verify that a published program will produce correct results, or that a published calculation is correct, in the same way that it can check the truth of scientific theorems or experimental measurements

K. Roberts, **1969**

CHANGER L'ÉNERGIE ENSEMBLE

The scientific community should always be able to verify that a published program will produce correct results, or that a published calculation is correct, in the same way that it can check the truth of scientific theorems or experimental measurements

K. Roberts, **1969**

Minisymposia “Étude de la qualité numérique de code de calcul industriels”

- Christophe Denis (EDF R&D)

Etude de la qualité numérique de codes de calculs industriels : problématique, premiers résultats et perspectives

Outline

1.Introduction

2. Numerical debugging

2. Numerical health check of dot product

3. The Xd+p approach

4. Implementation of CADNA in some communication and scientific libraries

5. Main objective in 2011

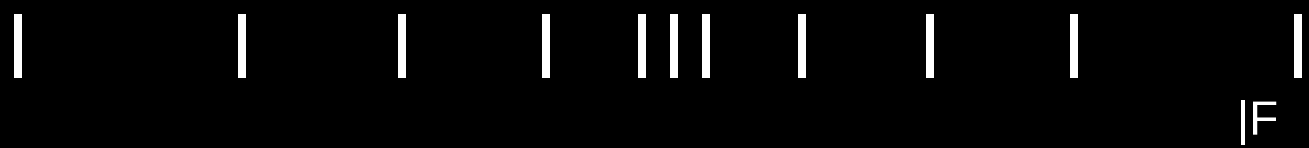
Introduction

The dark side of numerical computing

« *When you look at the dark side, careful you must be* », Yoda

A numerical algorithm generally designed for real numbers …

.. is run on computers with floating point numbers !



For example, the floating point summation is no longer associative !

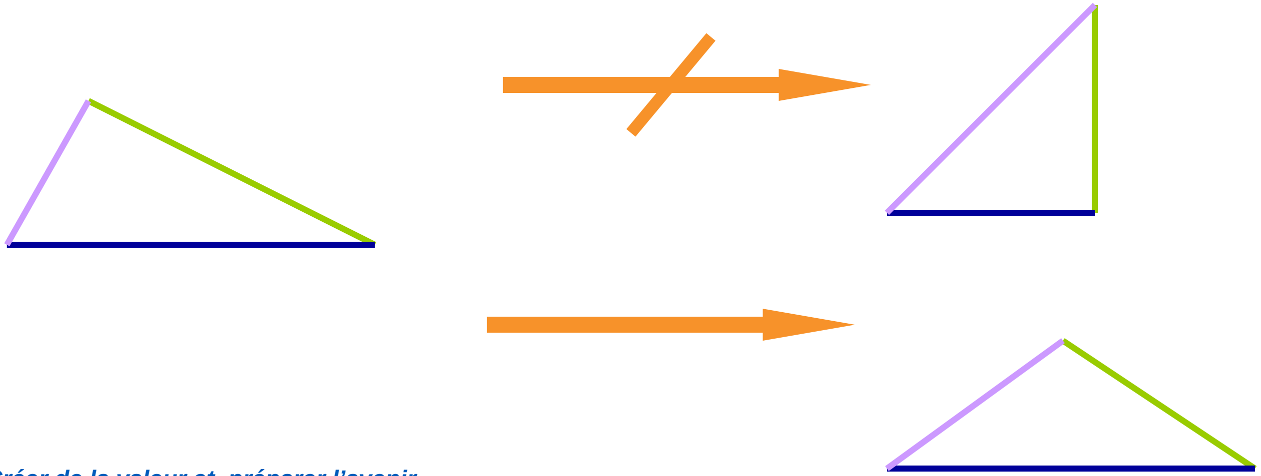
A new hope : simulate accurately and .. efficiently

« *May the force be with you* », Han Solo

- **“I have little doubt** that about 80 per cent of all the results printed from the computer **are in error to a much greater extent than the user would believe”,** Leslie Fox, 1971

- In 2011, still valid today …and exacerbated in a supercomputing environment
  →trillions of floating-point operations may be performed every second !
  →possible heterogeneous computer resources (CPU,GPU,…) !

- Our goal : Improve and validate the accuracy of numerical algorithms …but without penalizing the running-time performances !



Accuracy
Performance
Code lisibility

Foreword : Precision versus accuracy

Physical phenomenon

Modelisation error

Mathematical model

Discretisation error

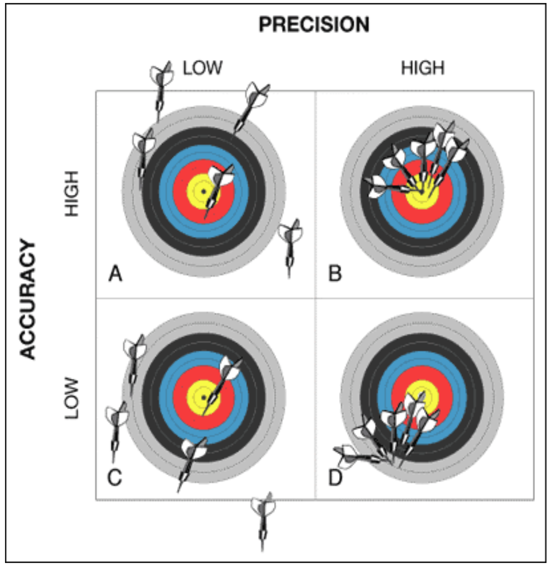
Numerical algorithm

Computing error

Precision

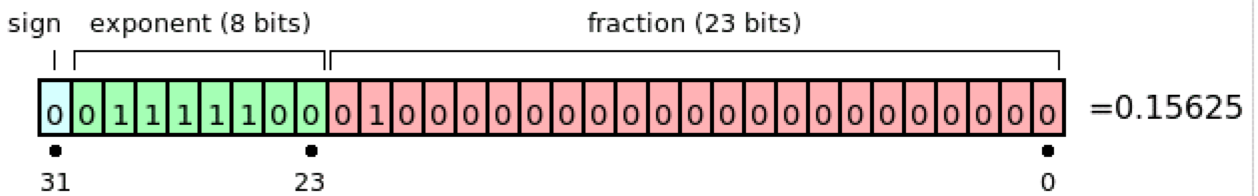
Accuracy

Computed solution using floating point arithmetic

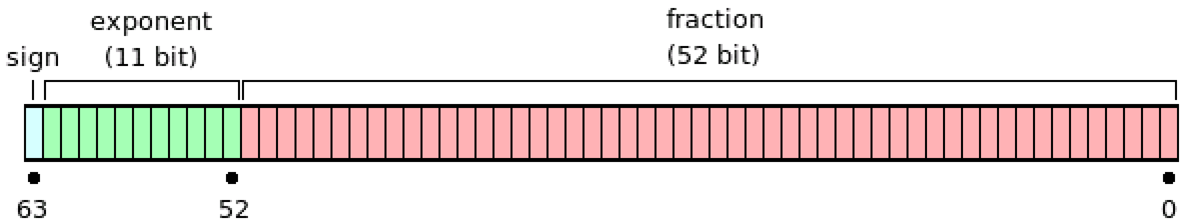


The IEEE 754 floating point number formats

- There are two primary formats:
  - 32 bit single precision and 64 bit double precision.
- Single precision (32 bits) consists of:



  - A single sign bit, 0 for positive and 1 for negative;
  - An 8 bit base-2 excess-127 exponent
  - A 23 bit base-2 significand, with a hidden bit giving a precision of 24 bits (i.e. $1.d_1d_2...d_{23}$)
- Double precision



The IEEE754 norm : four rounding modes

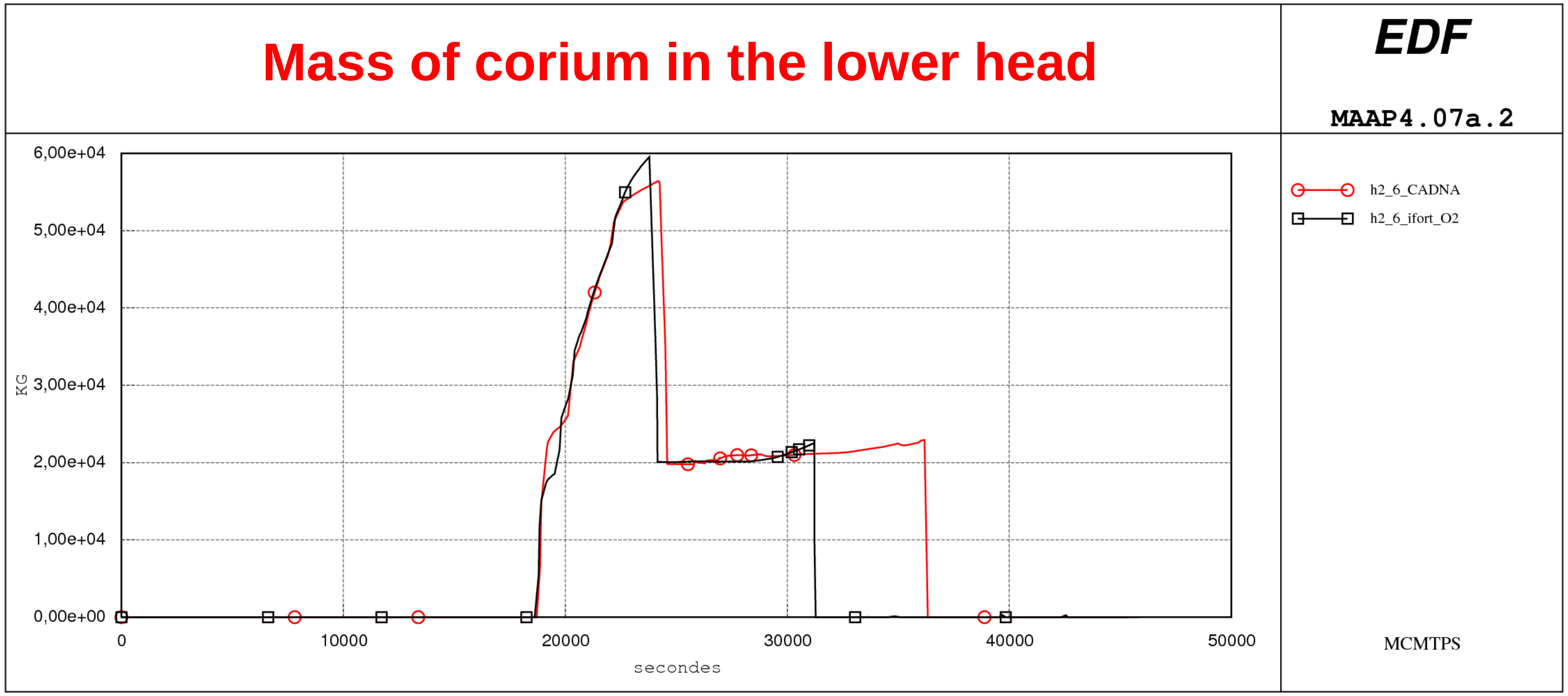
- Round to nearest (by default).
- Round toward 0 (also called truncation).
- Round toward + infinity.
- Round toward -infinity.

Numerical debugging on the MAAP code

Numerical health check of the MAAP code (1) Modular Accident Analysis Program

- MAAP is written by FAI (Fauske & Associates, LLC) to simulate the response of nuclear power plants during severe accident sequences
- It is a Fortran 77 code and uses common block, implicit declaration, …
  - MAAP4.07b/ 679 subroutines, 319 806 lines including 118274 lines of comments
- It is subject to numerical instabilities as it could provide slightly different results depending on the compiler (or the compiler options , -O0, -O2..)
- In the context of the PAGODES project
  - The CADNA library has been manually implemented on MAAP5 in 2010
  - To avoid this laborious job, a translator source tool has been designed in python to automatically implement CADNA in a Fortran 77/Fortran 90 code
    - Developed by INCKA with support of GN and CD
    - Some minor modifications have to be done after using the tool
  - This translator tool will be available for other projects

Impact of instability on results



Time of extensive failure :
-with the correction : 36 200 s
- wthout the correction : 31220 s

Difference of 16 %

Numerical health check of dot product

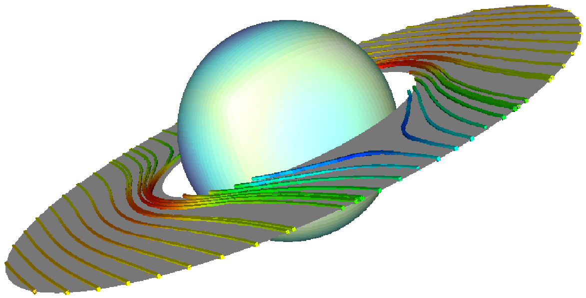
$$\mathbf{a}\cdot\mathbf{b}=\sum_{i=1}^{n}a_ib_i=a_1b_1+a_2b_2+\cdots+a_nb_n$$

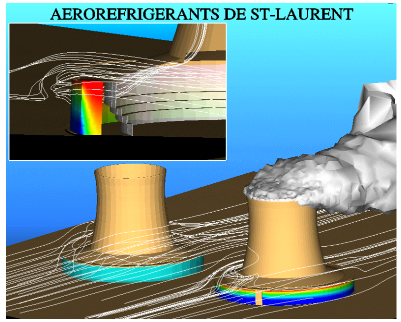
Interesting for Code_Saturne but certainly for other codes ..

Important to be validated/improved as the dot product is used to compute norm..

Impact on the convergence of iterative methods

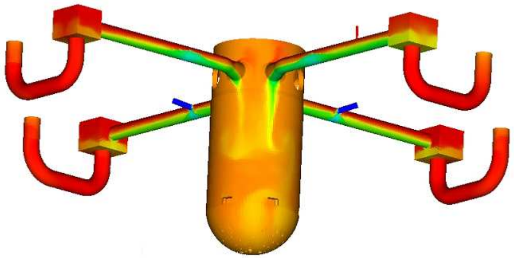
Code_Saturne

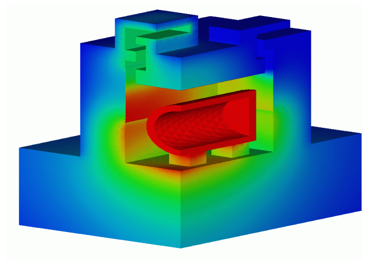




Physical modelling

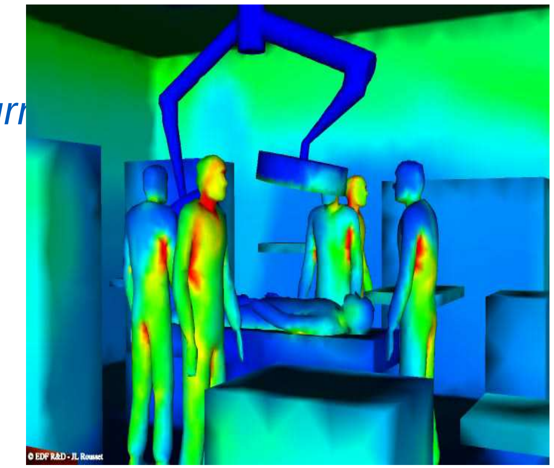
- Single-phase laminar and **turbulent flows**: k-ε, k-ω SST, v2f, RSM, LES
- **Radiative** heat transfer (DOM, P-1),**Combustion** coal, heavy fuel oil, gas (EBU, pdf, LWP)
- **Electric** arc and Joule effect
- **Lagrangian** module for dispersed particle tracking
- **Compressible** flow,
- **ALE** method for deformable meshes
- **Conjugate heat transfer** (SYRTHES & 1D)
- Specific **engineering modules** for nuclear waste surface s
- Derived version for **atmospheric flows** (*Mercure_Saturne*),





Flexibility

- **Portability** (UNIX and Linux), **GUI** (Python TkTix, Xml form
- **Parallel** on distributed memory machines
- **Periodic boundaries** (parallel, arbitrary interfaces)
- Wide range of **unstructured meshes** with **arbitrary interfaces**
- **Code coupling** capabilities (*Code_Saturne/Code_Saturn* *Code_Saturne/Code_Aster*, ...)

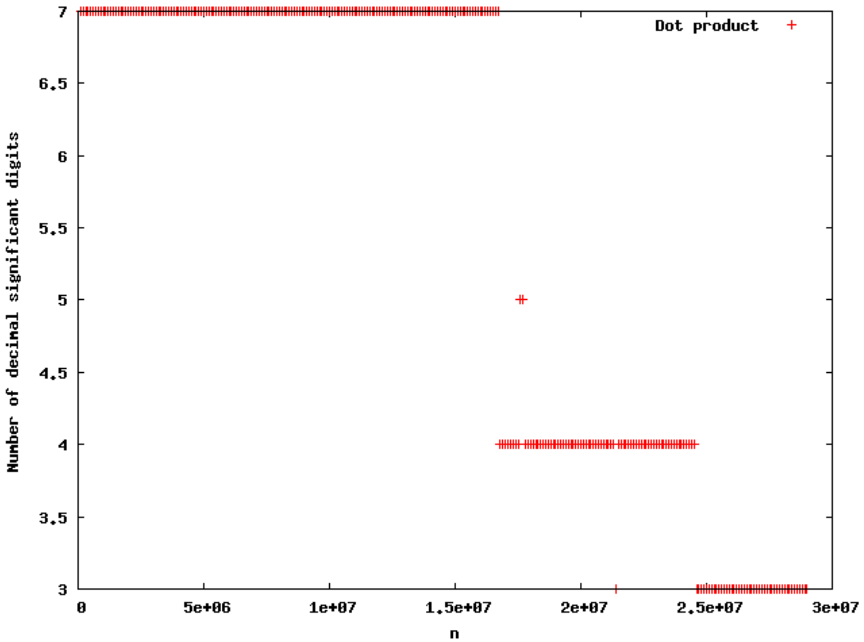


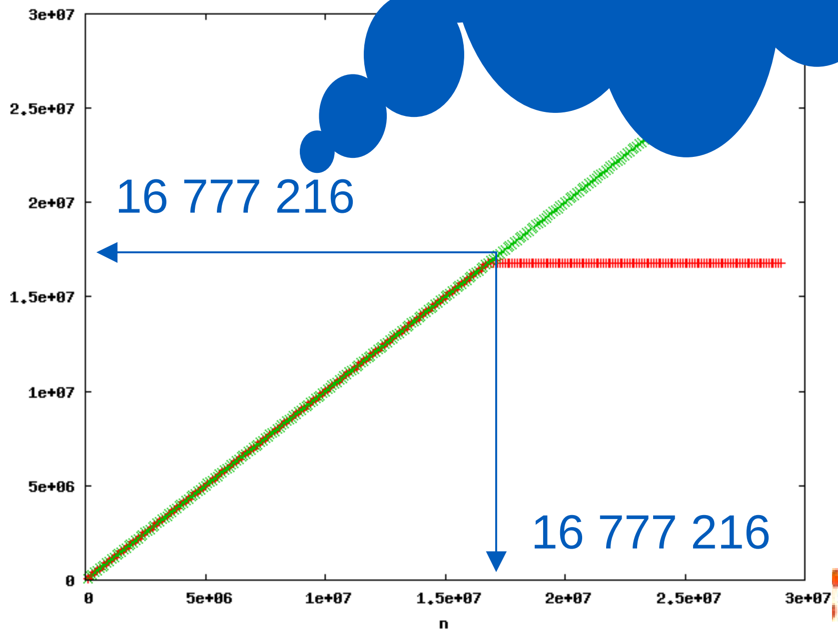
With the courtesy
Of M. Barrault

Example 1 : The dot product in single precision

$$P_{simple\_precision} = \begin{pmatrix} 1.0 \\ 1.0 \\ \vdots \\ 1.0 \\ 1.0 \end{pmatrix} \begin{pmatrix} 1.0 \\ 1.0 \\ \vdots \\ 1.0 \\ 1.0 \end{pmatrix} = 1.0 \times n$$

Indication
16 777 216 = 2^{24}



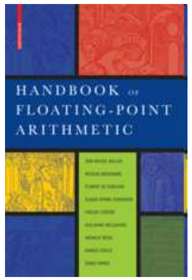


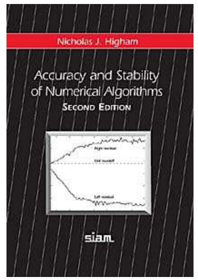
Preliminary results on the compensated summation

- Kahan's compensated summation method $s = \Sigma x(i)$
  - $s \leftarrow 0.0$; $e \leftarrow 0.0$
  - For i=1 to n
    - tmp $\leftarrow$ s
    - Y = x(i) + e
    - s= tmp + y
    - e = (tmp –s) +y
  - End For

compensated term e computed at iteration i and added to s at iteration i+1

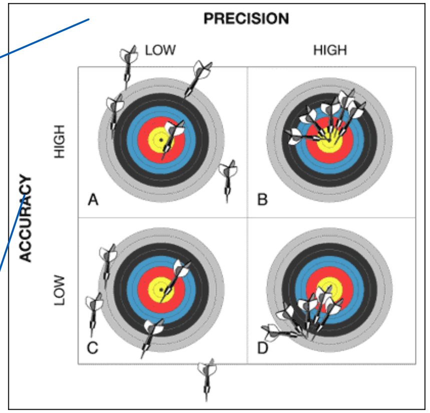
- But it exists other compensated summation method (double compensated method, etc.) to be implemented and compared in terms of accuracy and performance
  - N.J Higham, Accuracy and Stability of Numerical Algorithms, SIAM.
  - J.-M Muller et all, Handbook of Floating-Point Arithmetic, Birkhäuser Boston

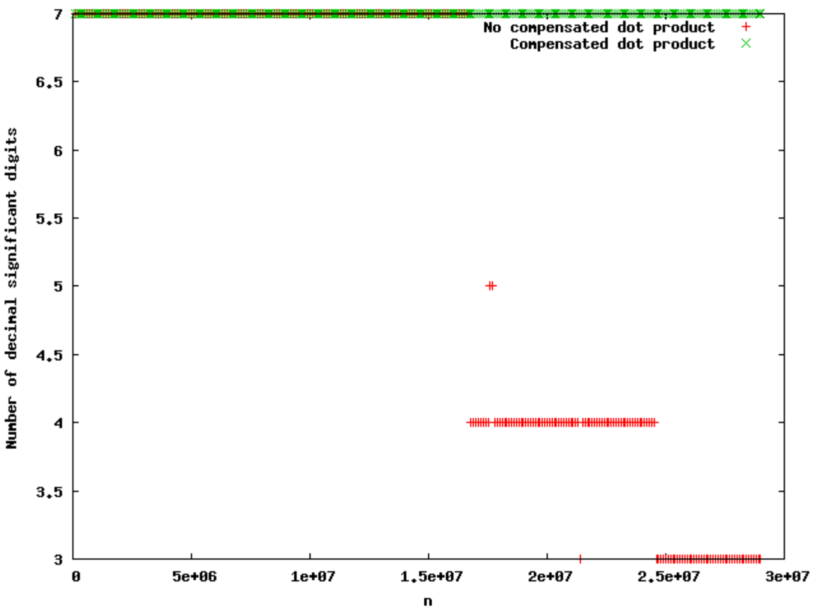


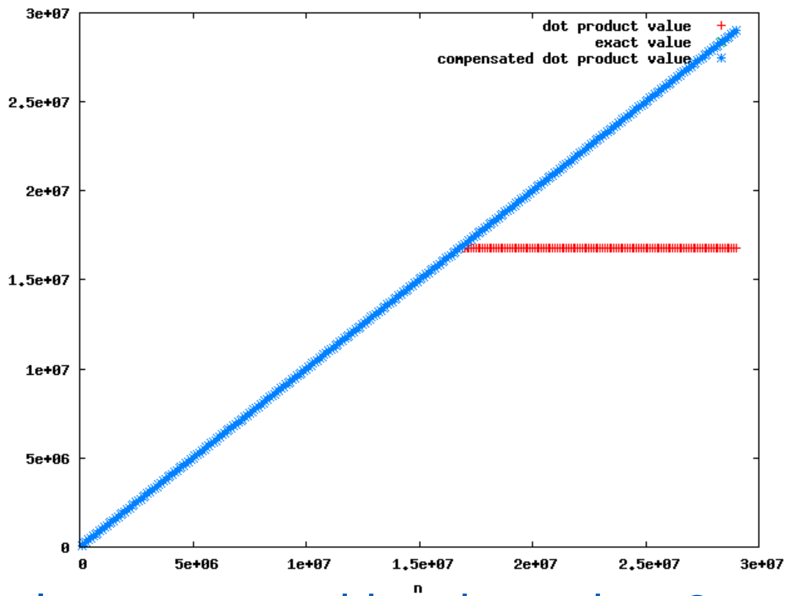


Return at the dot product in single precision

$$P_{simple\_precision} = \begin{pmatrix} 1.0 \\ 1.0 \\ \vdots \\ 1.0 \\ 1.0 \end{pmatrix} \begin{pmatrix} 1.0 \\ 1.0 \\ \vdots \\ 1.0 \\ 1.0 \end{pmatrix}$$

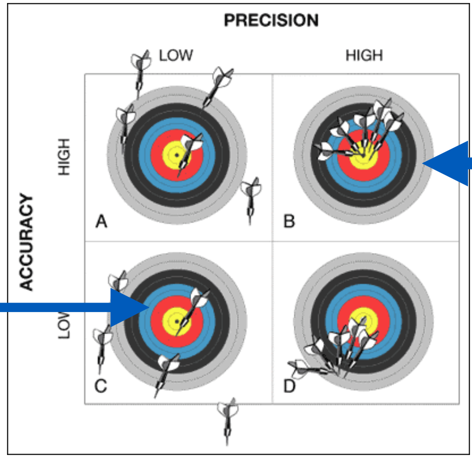






Timing : overhead 40% : Reproducible value or too machine dependent ?

Timing ....



Compensated dot product
(0.556 s)

Original dot product
(0.396 s)

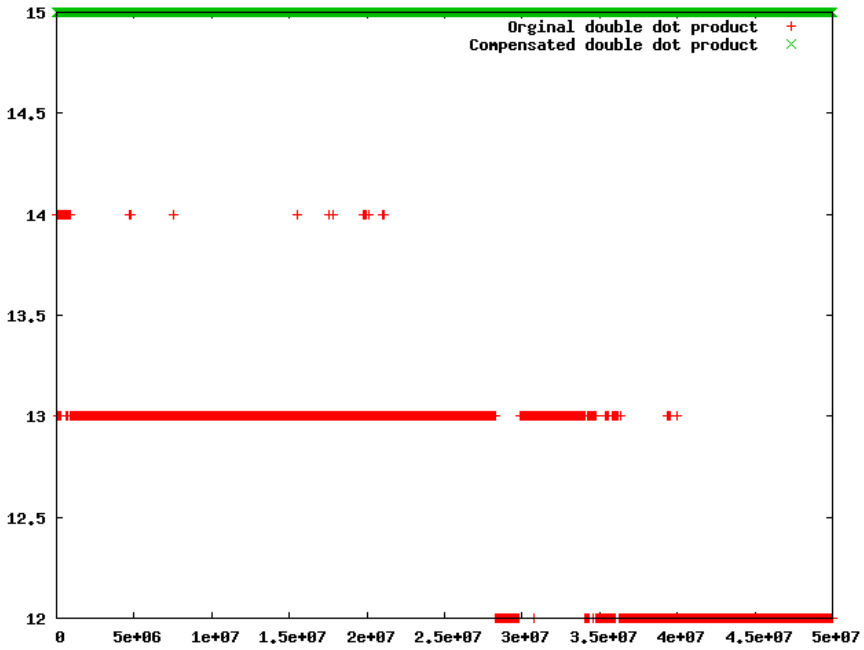
Overhead 40% : Reproducible value or too machine dependent ?

Measuring the computing time of summation algorithms in a high-level language on today's architectures is more of a hazard than scientific research.
S.M. Rump
(SISC, 2009)

Example 2 : The dot product in double precision

$$P_{double\_precision} = \begin{pmatrix} a_1 \\ \vdots \\ a_i \\ \vdots \\ a_n \end{pmatrix} \begin{pmatrix} b_1 \\ \vdots \\ b_i \\ \vdots \\ b_n \end{pmatrix}$$

Randomised positive values between 10^{-1} and 10^{1}

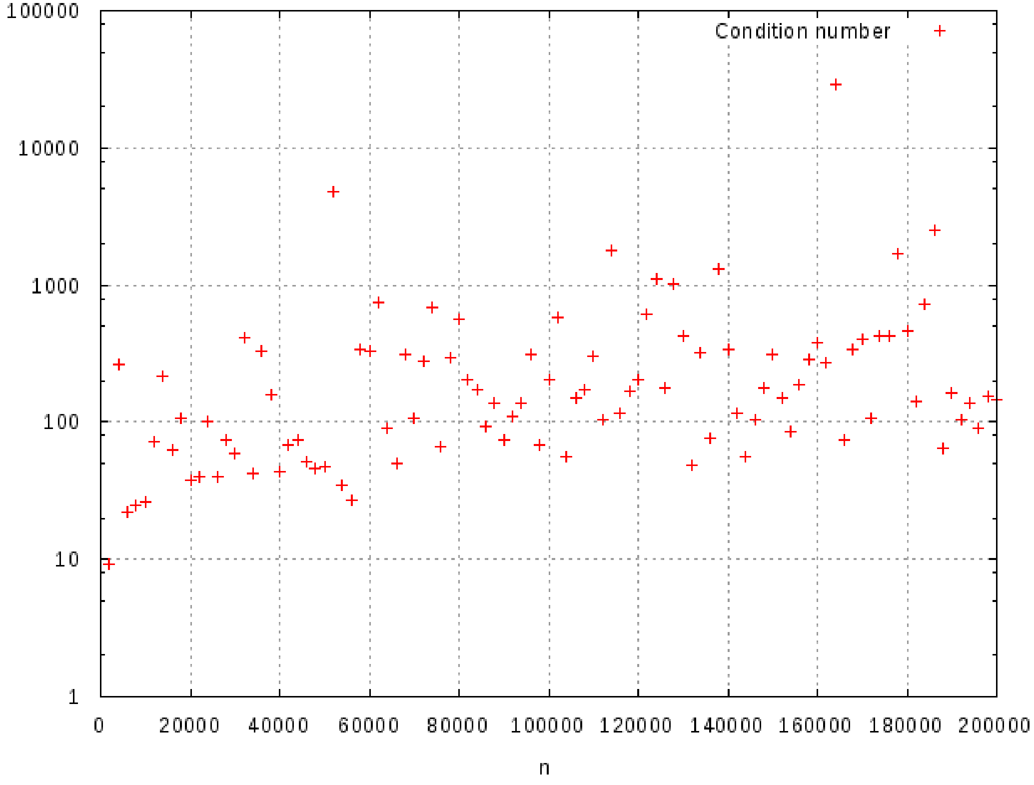


Example 3 : The summation in double precision (condition number)

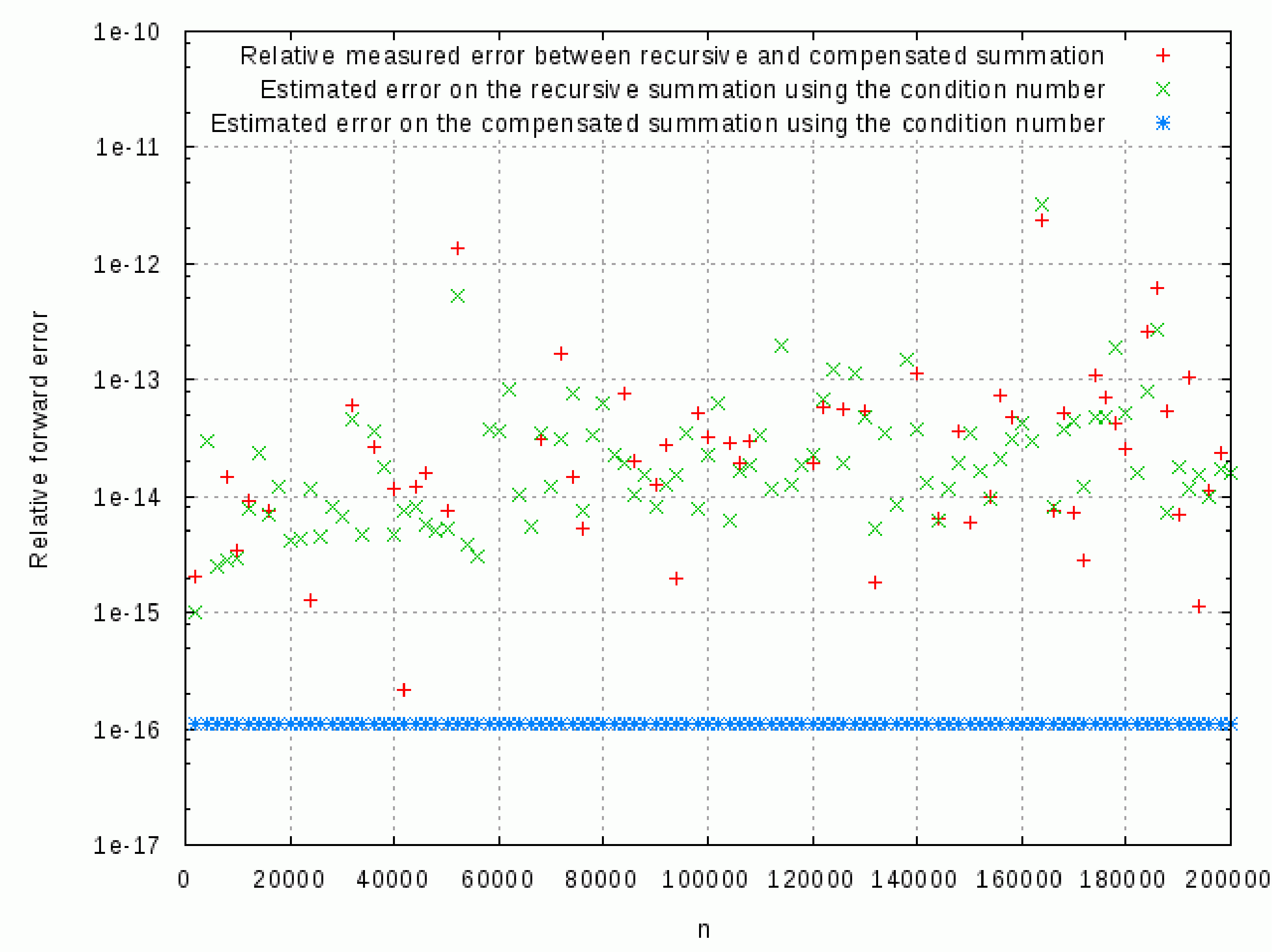
$$S_n = \sum_n p_i$$

$$CN = \frac{\sum |p_i|}{|\sum p_i|}$$

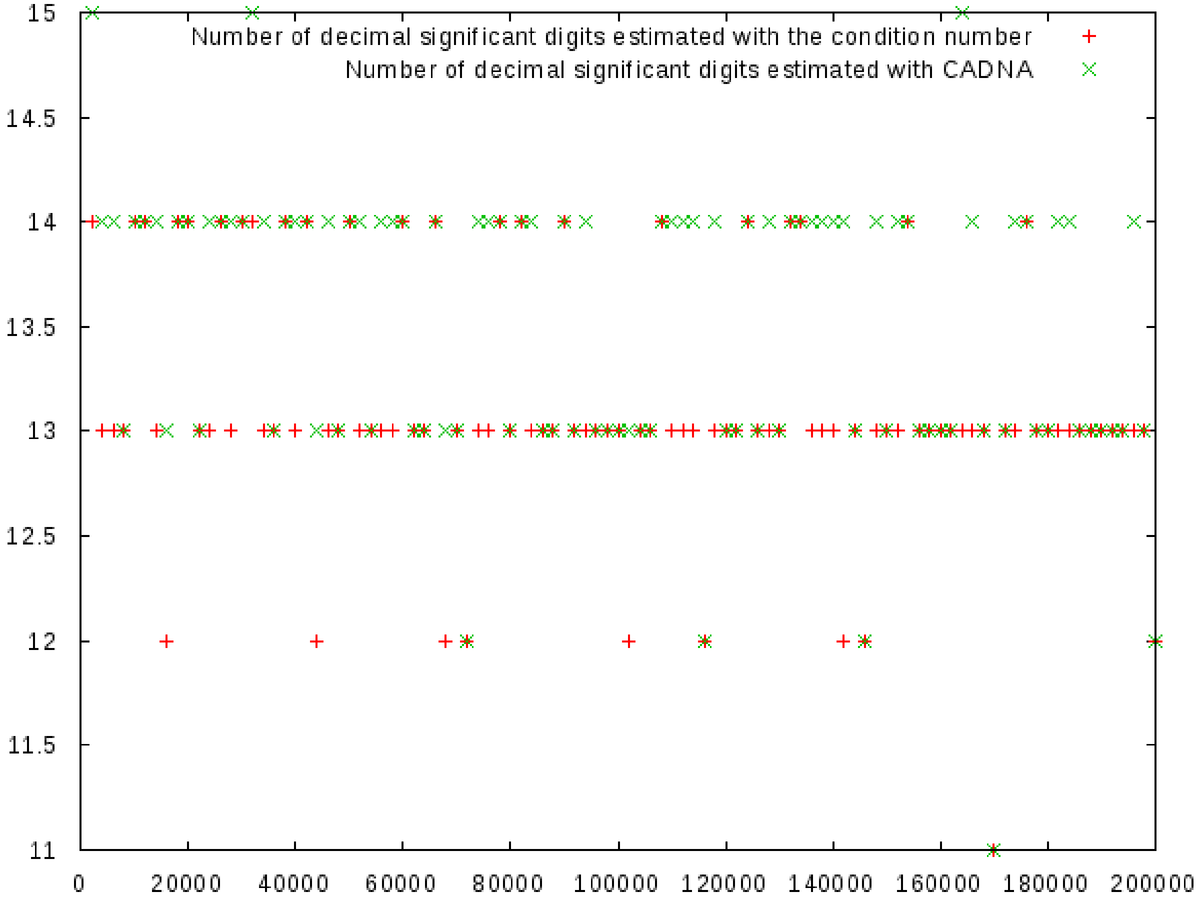
p_i : randomised positive values between -10^1 and 10^1



Relative forward errors



Number of decimal significant digits measured (with CADNA) and estimated (with the condition number)



The xD+p approach

The Telemac system

The Telemac system

- Several CFD programs for free surface flows (hydrodynamics, sediment transport, water quality, groundwater flows, waves)
- Since 1987, co-development (EDF R&D – LNHE and scientific partners)
- More than 200 commercial licences around the world
- Telemac-2D: Open Source in 2010

Telemac-2D and Telemac-3D

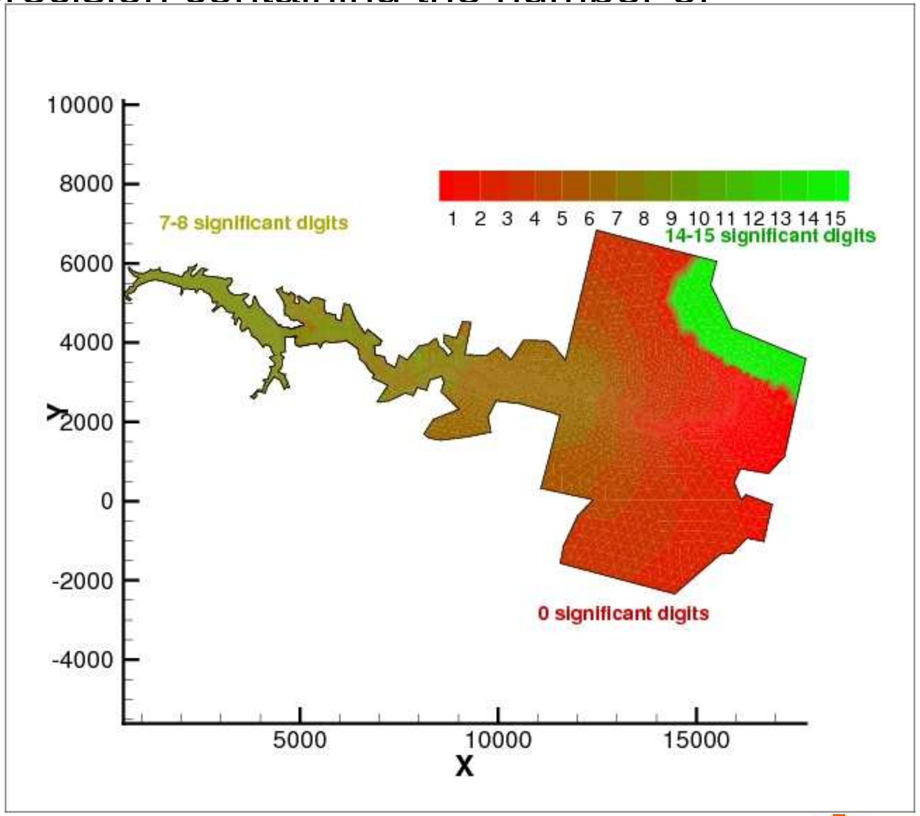
- Telemac-2D: Saint-Venant or Shallow water equations
- Telemac-3D: Navier-Stokes equations
- Based on finite element method
- Fluvial, estuarian, lacustrine and coastal flows, particularly tidal flows

One application at EDF R&D : the "xD+p" approach
Example on the Malpasset dam break

Example : computed results on the water level:
there is no idea about the numerical quality of the results !

•Idea adding a dimension called "p" for precision containing the number of significant digits

•Visualisation of the effect of the round-off error propagation of the results (interesting also to study mesh sensitivity)

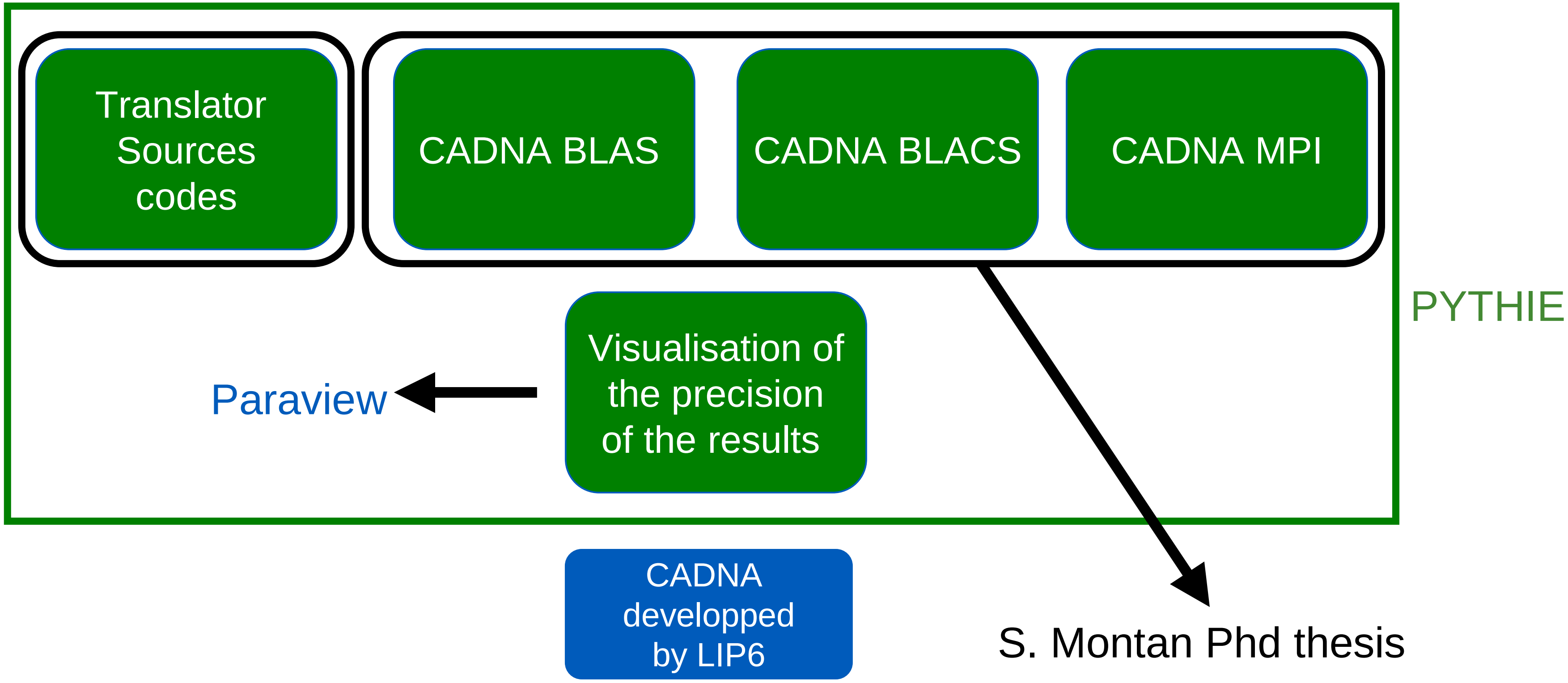


In this case the 0 significant digits is not a real problem as it concerns tiny values

Implementation of CADNA in some communication and scientific libraries

S. Montan, PhD student

A framework to pool this round-off error analysis activity

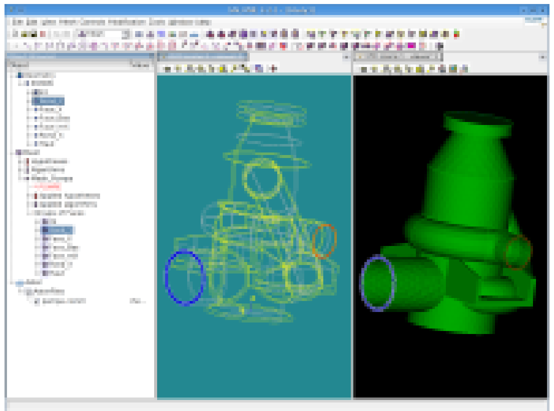


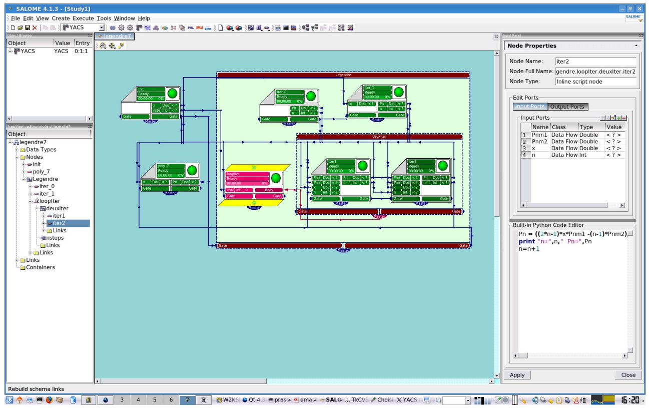
The development of a SALOME module dedicated to round-off error analysis could strongly improve the pooling.

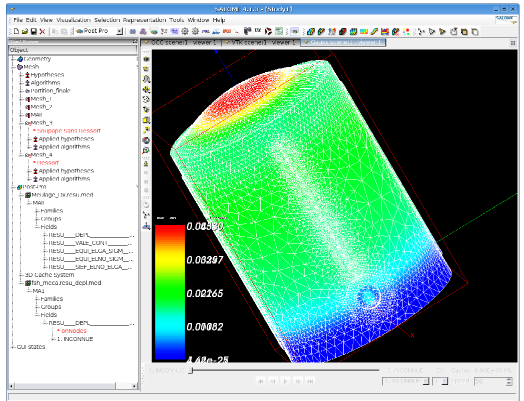
Objectives

Download: http://www.salome-platform.org

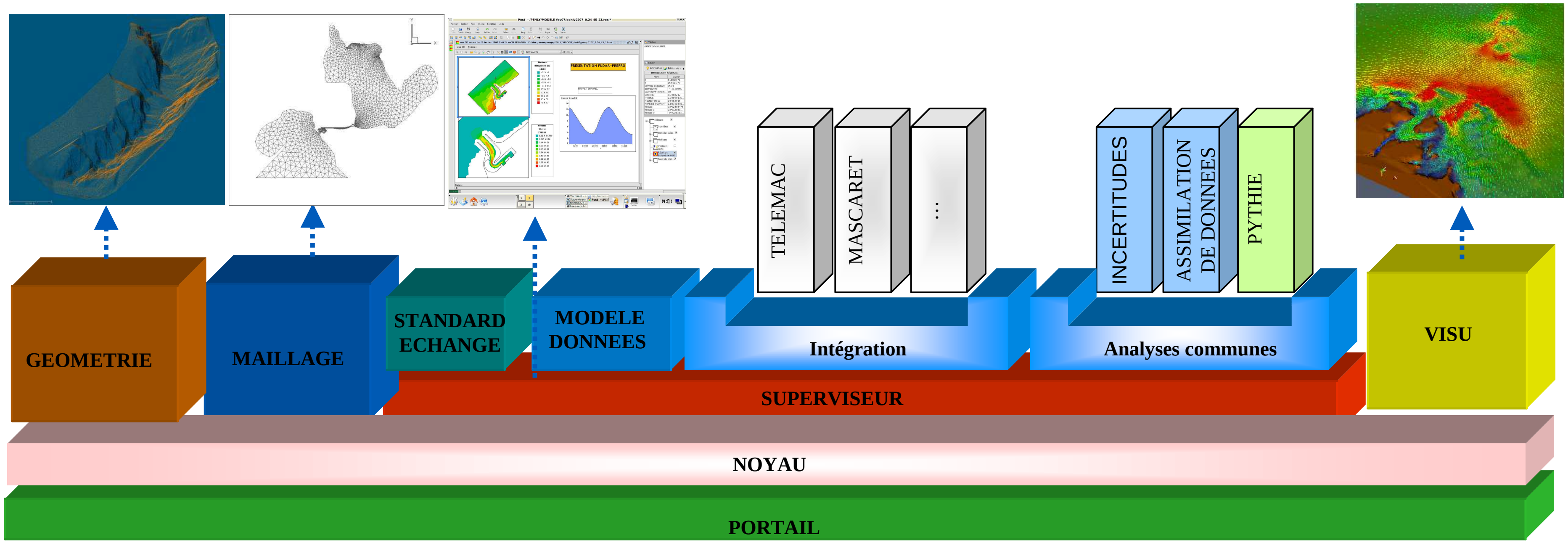
- Salome is a generic platform for **pre and post processing and code coupling** for numerical simulation with the following aims:
  - facilitate **interoperation between CAD modelling and computing codes;**
  - facilitate **implementation of coupling between computing codes** in a distributed environment ;
  - **pool production of developments** (pre and post processing, calculation distribution and supervision) in the field of numerical simulation.







The Salome platform: Mixing scales and physics with pre-post capabilities



BlueGene

CCRT

Cluster

Poste Calibre

Mur d'images

Implementation of CADNA in some communication and scientific libraries

- Currently, the CADNA library could be only used on sequential programs written in C/C++, Fortran77/Fortran90
- Unfortunately, numerical codes at EDF R&D use communication libraries (MPI, BLACS) and scientific libraries (BLAS, LAPACK)
- Implement CADNA directly in the BLAS library without redefining algorithms is quite easy but counterproductive
  - Overhead > 100 in some cases for BLAS level 3 operations !

- A PhD Thesis (Sethy Montan) has started since October 2010 to implement efficiently CADNA in MPI, BLACS, BLAS ..

Conclusion and future work

Main objective (in 2011..)

Improve and validate the accuracy of numerical algorithms but …without penalizing the running-time performances

- MAAP code
  - Finish the numerical debugging of MAAP4 by testing several benchmarks and test it in Calibre 7
  - Training and improve the collaboration with FAI
- TELEMAC
  - Test the Xd+P approach with SPARTACUS-2D
  - Implement compensated summation algorithm in the communication scheme and re-experiment
- Code_Saturne
  - Test our development of compensated algorithms with CADNA in some “real” cases provided by the Code_Saturne team
- CADNA BLAS (Sethy Montan, PhD thesis)
  - Implement efficiently CADNA into the BLAS (difficult for BLAS level 3!)
- ….. **without penalizing the running-time performances !**

  Measuring the computing time of summation algorithms in a high-level language on today's architectures is more of a hazard than scientific research. S.M. Rump (SISC, 2009)

  - Use and test the PerPI tool (Philippe Langlois and the DALI team)
    - was developed by the DALI team to measure, observe and analyze the instruction level parallelism (ILP) present in a code

Some references

- N.S. Scott, F. Jézéquel, C. Denis, J.-M. Chesneaux, ***Numerical 'health check' for scientific codes: the CADNA approach***, *Computer Physics Communications*, *Volume 176, Issue 8*, *15 April 2007*, *Pages 507-521*, ***2007***
- C. Denis, ***Numerical Health Check of Industrial Simulation Codes from HPC Environments to New Hardware Technologies***, Parallel Processing and Applied Mathematics, Lecture Notes in Computer Science, **2009.**
- C . Denis, Charles Moulinec, Jean-Michel Hervouet, Emile Razafindrakoto, Robert Barber, Dave Emerson, Xiaojun Gu, ***TELEMAC, an Efficient OpenSource HPC Hydrodynamics Suite,*** Parallel CFD 2010, **2010**, and under review in Computers and Fluids.
- C. Moulinec, C. Denis, N Durand, R. W. Barber, D. R. Emerson, X. J. Gu, E. Razafindrakoto, R. Issa and J.-M. Hervouet, ***Coupling HPC and Numerical Validation: Accurate and Efficient Simulation of Large-scale Hydrodynamic Events***, in PARENG'11, Second International Conference on Parallel, Distributed, Grid and Cloud Computing for Engineering, **2011**
- C. Denis, C. Moulinec, N Durand, R. W. Barber, D. R. Emerson, X. J. Gu, E. Razafindrakoto, R. Issa and J.-M. Hervouet***, Simulate Accurately and Efficiently Large Scale Hydrodynamic Events*****,** 34th World Congress of the International Associationfor Hydro-Environment Engineering and Research, Australia, **2011**