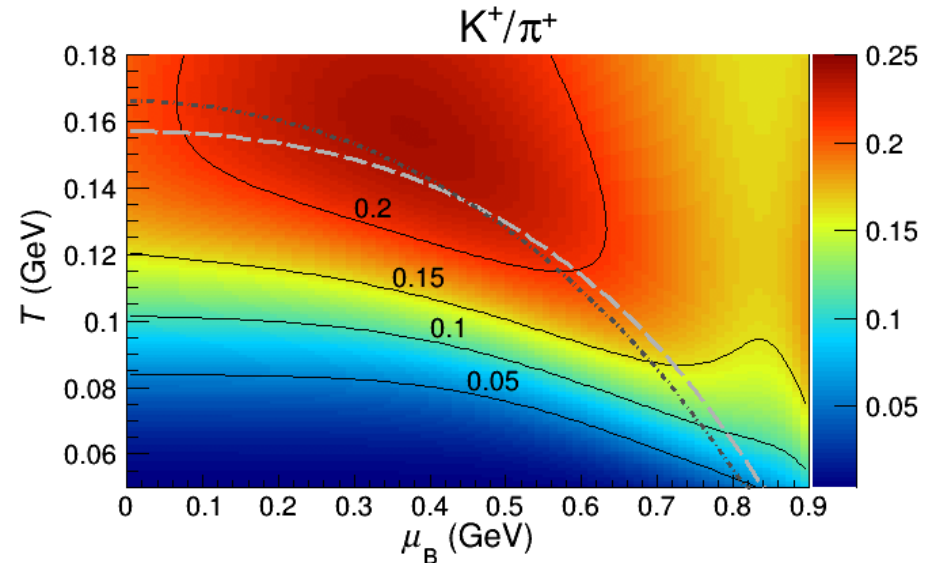
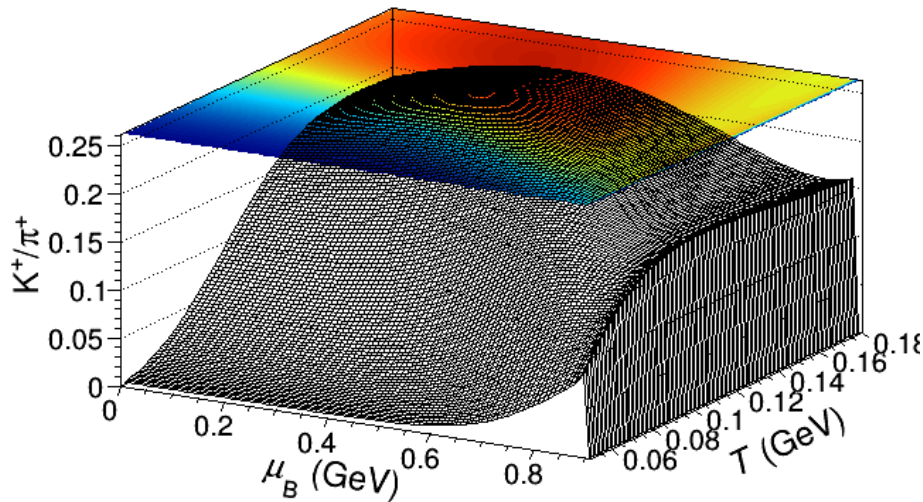


## Modelling particle production with THERMUS (v4?)



## OUTLINE

- THERMUS (A Thermal Model Package for ROOT)
  - ➔ Philosophy and historical considerations
  - ➔ Current version and caveats
  - ➔ New version proposed and availability
  - ➔ Cross-checks and examples
  - ➔ Near- and middle-term prospects
  - ➔ Summary



Prometheus logo/image  
used by Spencer Wheaton

# THERMUS: Content and Philosophy

- THERMUS -- A Thermal Model Package for ROOT

[Spencer Wheaton](#), [Jean Cleymans](#) and [Michael Hauer](#)

(Submitted on 15 Jul 2004 (v1), last revised 15 Aug 2008 (this version, v2))



**THERMUS is a package of C++ classes and functions allowing statistical-thermal model analyses of particle production in relativistic heavy-ion collisions to be performed within the ROOT framework of analysis.**

Calculations are possible within three statistical ensembles; a grand-canonical treatment of the conserved charges B, S and Q, a fully canonical treatment of the conserved charges, and a mixed-canonical ensemble combining a canonical treatment of strangeness with a grand-canonical treatment of baryon number and electric charge.

THERMUS allows for the assignment of decay chains and detector efficiencies specific to each particle yield, which enables sensible fitting of model parameters to experimental data.

Journal reference: Comput.Phys.Commun.180:84-106,2009

DOI: [10.1016/j.cpc.2008.08.001](https://doi.org/10.1016/j.cpc.2008.08.001)

Cite as: [arXiv:hep-ph/0407174](https://arxiv.org/abs/hep-ph/0407174) (or [arXiv:hep-ph/0407174v2](https://arxiv.org/abs/hep-ph/0407174v2) for this version)

- ➔ open source code interfaced with ROOT (both strength and weakness)
- ➔ commonly used: many publications and proceedings (>200)
- ➔ pedagogical aspects

## THERMUS: Historical considerations

v1.0	(15 Jul. 2004)	first version (Spencer Wheaton and Jean Cleymans)
v1.1	(28 Apr. 2005)	minor corrections for use with newer versions of ROOT (SW)
v2.0	(18 Aug. 2005)	additional constraints e.g. excluded volume corrections (SW)**
/	(5 Mar. 2007)	minor corrections for use with newer versions of ROOT (SW)
v2.1	(23 Oct. 2007)	additional constraints e.g. percolation (SW and Michael Hauer)**
v2.2	(4 Mar. 2011)	contains the framework to treat charm and beauty grand-canonically
/	(6 Jun. 2011)	Extended particle set including hadrons containing c and b quarks (Dawit Worku and Jean Cleymans)
/	(7 Nov. 2011)	macosx makefiles and support (B.H.)
v2.3	(5 Dec. 2013)	couple of minor bug fixes (B.H.)
v3.0	(28 Nov. 2014)	ROOT 6 compatible together with a particle list containing charm, beauty and hypernuclei (B.H.)

info from Spencer Wheaton's THERMUS [webpage](#) at University of Cap Town

\*\* deduced from the code (my bad if any mistake 🤓)

## Current version and caveats

- **Nice and fast Statistical Thermal Model**

- ➔ interfaced with ROOT (so ROOT user friendly)
  - (caveat: major ROOT change to be checked e.g. integrals v5→v6)
- ➔ extensive use of TMinuit minimisation package
  - (translated in C++ from Fred James's fortran 1994 version)
- ➔ constraining functions use "Numerical Recipes in C" code<sup>©</sup>
  - (caveat: user must have his/her own NRC paid copies)
  - (otherwise self-consistent *tar.gz* package distributed at UCT)
- ➔ fully open source code with simple (.txt) particle list and decays
- ➔ shared object libraries (.so) built with make/makefiles
  - (loaded once and for all when opening ROOT)

## Recent developments

- Initially part of a project defined with **Helmut Oeschler**
  - ➔ chaire Gutenberg 2017 (Strasbourg) awarded to Helmut
- **Remove “Numerical Recipes in C” dependence**
  - ➔ replace NRC (mainly Brent and Broyden) routines used for solving non linear systems and constraining parameters
    - (see THERMUS, Comp. Physics Comm. 180 (2009), p100)
    - but avoid any “additional” dependence
  - ➔ ROOT v6 includes “mathmore” & Gnu Scientific Library (GSL)
  - ➔ **GSL interface implemented by Yves Schutz (end of 2017)**
    - mathmore needs GSL by default
    - ROOT feature “gsl\_shared” needed by THERMUS
    - no additional dependence

## Recent developments

- **CMake for building the shared object libraries**
  - ➔ cross-platform (linux flavours and macosx)
  - ➔ checking dependences and locations (ROOT and GSL)
  - ➔ options and flags propagated from ROOT to THERMUS
    - (make/makefile provided but not supported anymore)
  - ➔ single step configuration and compilation (B.H.)
    - simple script (provided in the repository): *cmakethermus.sh*
    - build dir (independent from source) can be deleted anytime
    - (loaded once and for all when opening ROOT in *rootlogon.C*)

- Use the makefiles in "\$(THERMUS)/functions", "\$(THERMUS)/nrc" and "\$(THERMUS)/main" to build the libFunctions.so, libNRCFunctions.so and libTHERMUS.so shared object files respectively:

```
[$(THERMUS)/functions]$ make all
```

```
[$(THERMUS)/nrc]$ make all
```

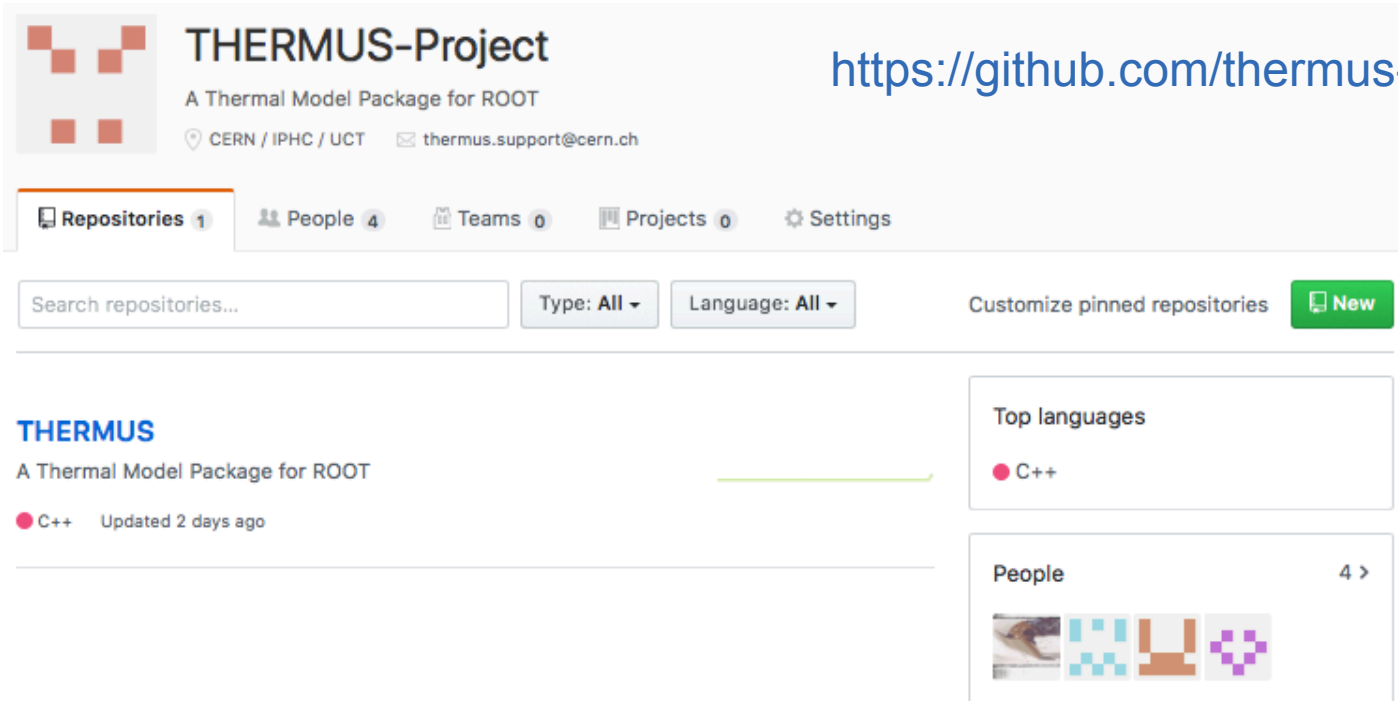
```
[$(THERMUS)/main]$ make all
```

Former method for building THERMUS

## Recent developments

- **Distribution of versions via GitHub (last week, B.H.)**

- ➔ use GitHub for developing and distributing the code source



**THERMUS-Project**  
A Thermal Model Package for ROOT  
CERN / IPHC / UCT | thermus.support@cern.ch

<https://github.com/thermus-project>

Repositories 1 | People 4 | Teams 0 | Projects 0 | Settings

Search repositories... | Type: All | Language: All | Customize pinned repositories | New

**THERMUS**  
A Thermal Model Package for ROOT  
C++ Updated 2 days ago

Top languages  
C++

People 4 >

- ➔ people (so far): Jean Cleymans, B.H., Yves Schutz, Spencer Wheaton

- ➔ *git clone https://github.com/thermus-project/THERMUS.git THERMUS*

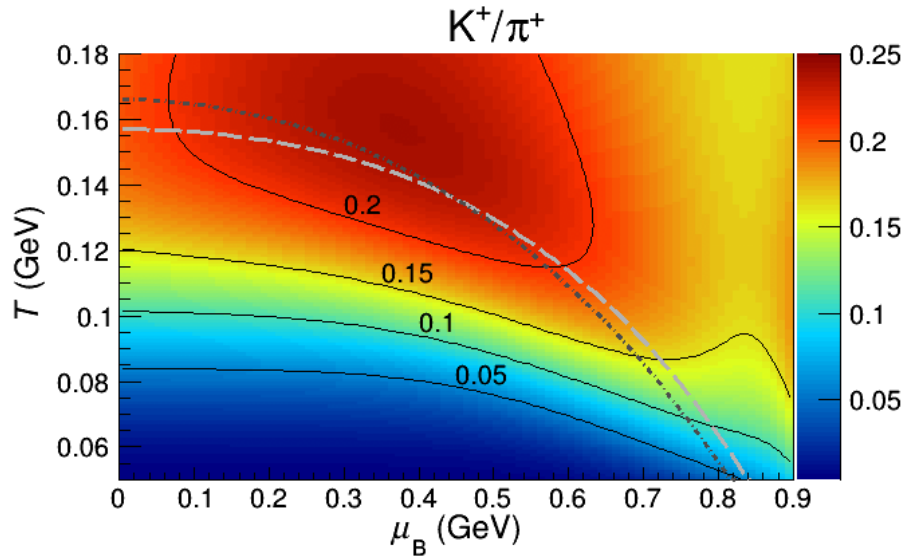
# Cross-checks and examples

- full benchmark ongoing

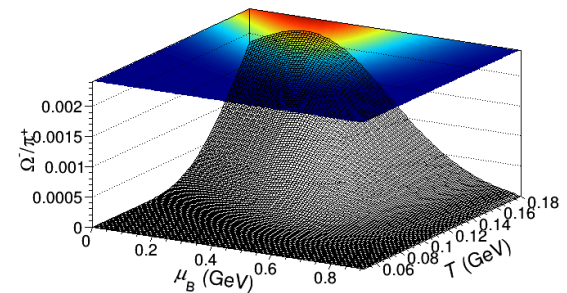
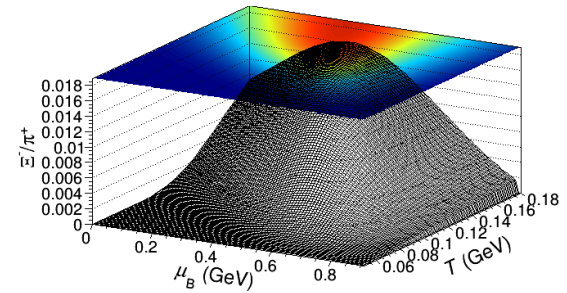
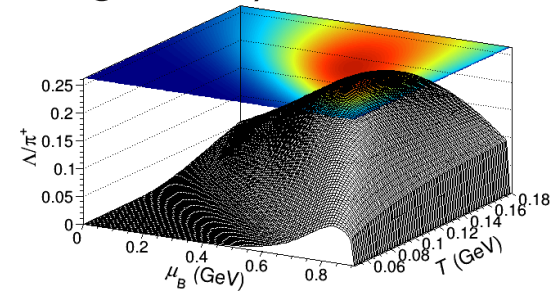
➔ predictions: no difference found (Jihye Song, B.H.) between

- root-v5 + THERMUS-v3 (NRC)
- root-v6 + THERMUS-v4? (GSL)

➔ example: H. Oeschler *et al*, Eur. Phys. J. C77 (2017) no.9, 584



produced with THERMUS new version  
(to be compared with Workshop poster 🙌)



## Cross-checks and examples

- full benchmark ongoing
  - ➔ fits: no difference found (B.H.) but 8% slower w/ EVC
    - ALICE Pb-Pb results at 2.76 TeV and 5.02 TeV
    - ALICE p-Pb results at 5.02 TeV
  - ➔ examples: simple root macros (scripts) to be provided
    - GitHub: [thermus-project/THERMUS/tests](https://github.com/thermus-project/THERMUS/tests)
  - ➔ prompting “usual suspect” users for additional evaluations
    - in ALICE, in HADES and in STAR
    - for instance, next speaker 🤔
  - ➔ Remark: significant changes in the routines so consistency checks with former version(s) before adding new features...

## Near-term and middle-term prospects

- “intermediate” counting of decaying hadrons
  - ➔ feature several times mentioned by users (as confusing)
    - two-step calculation needed for extracting resonance to stable ratios
    - fix already available with one extra loop (**V.Vovchenko**, 2016)

- internal (light) database for hadron list and branching ratios



- SQLite DB already tried
- implemented by **Yves Schutz**
- input from PDG and ROOT

example for (hyper-) nuclei

- Qt version and web version(s)

id ▲	name	pdg	spin	statistic	mass	s	baryon	charge
1	Deuteron	1000010020	3	-1	1,87561	0	2	1
2	Deuteron_bar	-1000010020	3	-1	1,87561	0	-2	-1
3	Triton	1000010030	2	1	2,80892	0	3	1
4	Triton_bar	-1000010030	2	1	2,80892	0	-3	-1
5	HE3	1000020030	2	1	2,80839	0	3	2
6	HE3_bar	-1000020030	2	1	2,80839	0	-3	-2
7	Alpha	1000020040	1	-1	3,72738	0	4	2
8	Alpha_bar	-1000020040	1	-1	3,72738	0	-4	-2
9	LambdaNeutron	1010000020	3	-1	2,057	-1	2	0
10	LambdaNeutron_bar	-1010000020	3	-1	2,057	1	-2	0
11	LambdaProton	1010010020	3	-1	2,255	-1	2	1
12	LambdaProton_bar	-1010010020	3	-1	2,255	1	-2	-1
13	DiLambda	1020000020	1	-1	2,231	-2	2	0
14	DiLambda_bar	-1020000020	1	-1	2,231	2	-2	0
15	HyperTriton	1010010030	2	1	2,99114	-1	3	1
16	HyperTriton_bar	-1010010030	2	1	2,99114	1	-3	-1
17	HyperHydrogen	1010010040	1	-1	3,9225	-1	4	1
18	HyperHydrogen_bar	-1010010040	1	-1	3,9225	1	-4	-1
19	HyperHelium	1010020040	1	-1	3,92119	-1	4	2
20	HyperHelium_bar	-1010020040	1	-1	3,92119	1	-4	-2
21	HyperHelium5	1010020050	2	1	4,8399	-1	5	2
22	HyperHelium5_bar	-1010020050	2	1	4,8399	1	-5	-2

## Summary

- New version of THERMUS (v4?) is available
  - ➔ no dependence on Numerical Recipe in C anymore
    - GNU Scientific Library (interfaced with ROOT)
  - ➔ Cross-platform (Cmake) configuration and building w/ ROOT v6
  - ➔ open-source project on GitHub
    - <https://github.com/thermus-project>
    - contributors: J.Cleymans, B.H., Y.Schutz, S.Wheaton
    - git clone <https://github.com/thermus-project/THERMUS.git>
  - ➔ currently performing cross-checks
  - ➔ new developments to be even more user-friendly are ongoing

## Summary

- New version of THERMUS (v4?) is available
  - ➔ no dependence on Numerical Recipe in C anymore
    - GNU Scientific Library (interfaced with ROOT)
  - ➔ Cross-platform (Cmake) configuration and building w/ ROOT v6
  - ➔ open-source project on GitHub
    - <https://github.com/thermus-project>
    - contributors: J.Cleymans, B.H., Y.Schutz, S.Wheaton
    - git clone <https://github.com/thermus-project/THERMUS.git>
  - ➔ currently performing cross-checks
  - ➔ new developments to be even more user-friendly are ongoing

Thanks for your attention !