Romanian Reports in Physics, Vol. 67, No. 3, P. 819–830, 2015

ELEMENTARY PARTICLE PHYSICS

DEVELOPMENT OF AN INTEGRATED ON-LINE SYSTEM FOR THE STUDY OF THE HADRONIC INTERACTIONS IN HIGH ENERGY PHYSICS

S. CIORANU[1], A. JIPA[1], P.M. POTLOG[2]

[1]University of Bucharest, Faculty of Physics, Bucharest, Romania
E-mails: silviu.cioranu@gmail.com, alexandru.jipa@gmail.com
[2]Institute of Space Science, P.O. BOX MG-23, RO-77125, Magurele-Bucharest, Romania
E-mail: mihaipotlog@spacescience.ro

Received November 7, 2013

Abstract. A WWW interface to an integrated system for on-line Monte Carlo simulations in High Energy Physics named YaPT is described. The aim of the system is to allow users to perform on-line simulations and analysis using codes like UrQMD or AMPT and ROOT framework. The interface is structured in several modules for user authentication, interface with the simulation programs, output analysis, communication among users, etc. The output results of the simulations can be analyzed using a set o predefined macros. All simulations performed in the system are stored in a database and can be later accessed by all the users. We present the structure of the web interface and discuss a specific example of physics results calculated using the system.

Key words: on-line integrated system, Monte Carlo simulations, High Energy Physics.

1. INTRODUCTION

In the Computing Laboratory of the Research Center *Nuclear Matter in Extreme Conditions* (NMEC), at the Faculty of Physics, University of Bucharest, we implemented an integrated system for on-line Monte Carlo (MC) simulations in High Energy Physics. The system was named ***YaPT*** – ***Y****et* ***a****nother High Energy* ***P****hysics* ***T****ool* [1]. Several theoretical models for relativistic heavy ion collisions, and their associated simulation codes, like UrQMD [2] and AMPT [3], are implemented and can be accessed via an easy to use web interface.

Due to its flexibility, the system we developed can be used both for research and teaching activities. We will present the structure of the web interface and discuss a specific example of Physics results calculated using this system.

In the World Wide Web (WWW) there are some dedicated servers for on-line calculations in Particle Physics or Nuclear Physics. The most advanced one is a HepWeb server [4] at the Joint Institute for Nuclear Research (JINR Dubna, Russia), namely: *HepWeb*. It allows users to perform the most popular calculations in High Energy Physics – calculations of hadron-hadron, hadron-nucleus and nucleus-nucleus interaction cross sections, as well as calculations of secondary particles characteristics in the interactions using Monte Carlo event generators. Another tool, JetWeb-server [5] – a WWW interface and database for Monte Carlo tuning and validation – has the aim "to allow rapid and reproducible comparisons to be made between detailed measurements at High Energy Physics colliders and general Physics simulation packages" [5]. Along with the Monte-Carlo generators available, such as PYTHIA and HERWIG, this server also includes data from the HERA, LEP and Tevatron experiments.

In the following – section 2 – we present the motivation for the system and an outline of the design and technology used in the YaPT server. Section 3 contains a decomposition of the modules as they were implemented, Section 4 present the data flow of the YaPT system, while section 5 cover a use case which demonstrates the YaPT performances.

2. ARCHITECTURE OF WEB-SERVICE YaPT

2.1. SERVICE REQUIREMENTS

With the increasing complexity of scientific experiments of recent years has increased the need to make better predictions, both theoretical and experimental, using Monte Carlo simulations codes. This opens new perspectives for the creation of new tools to facilitate its use, both in research – to obtain descriptions of the phenomena of nature as close to reality – and in education – to familiarize students with advanced tools for scientific research. The goal of YaPT system is to make a small part of the high energy simulation codes available *via* a www interface.

Analysis of the similar existing online servers using Monte Carlo codes to describe nuclear interactions has lead to the conclusion that the YaPT service has to meet the following requirements:

- The architecture of the YaPT service should be WEB-oriented
- the YaPT service should be modularized and easy to install, allowing modifications of various parts – esspecially upgrades of simulators – without remastering the whole service
- The input for the simulators should be standardized, therefore should not have any special settings necessary only for the YaPT
- A database with simulations results – with specific parameters – should be created because repeating some calculations would consume a lot of computational time.

Besides this there are another two requirements which come from the planned infrastructure of the system: because we intend to develop an online solution, the requirements from the users – both on hardware and software levels – should be minimal; also, the simulations results are planned to be evaluate using

CERN's scientific libraries ROOT [6], tool which means that an online convertor should be developed.

2.2. SYSTEM ARCHITECTURE

The YaPT system has three major parts which will be described extensively in Section 3. This includes the **Web interface** which allows the user to enter data, view results, communicate with other users, authenticate and consult the system documentation, the **Simulations module** which consists of several Monte-Carlo simulation codes for the study of complex phenomena in Physics, being treated differently depending on their specific needs and **Analysis module** which consists of data analysis programs needed to obtain scientific results from simulated or experimental data. These modules are connected through **Simulation Service** which is the set of PHP scripts that handle the simulations and analysis. This makes the connection between the GUI and MC codes or ROOT.

The structure accepted and implemented is in full accordance with the given above requirements:

- The YaPT service is based on the "client-server" architecture.
- The YaPT service is implemented in PHP, and uses the AJAX technology. User can apply any web-browser (IE, Mozilla, Opera etc.) for their viewing.
- The output of YaPT simulators are standardized with output data written in a ASCII file. Additionally a ROOT-file of the histograms could be also created.
- The results of simulations and analysis are stored in the database.

All simulators are implemented on the server as executable modules (the source files of the codes are unavailable for users). Of course, this simulators can be also started independently of the YaPT. This allows us to upgrade simulators without reboot of the server and add new simulators specifying only the path to the corresponding executable modules in a configuration file.

The output data from simulators can be downloaded for further independent analysis or can be converted by a special method created by us into .root type file which allow furthermore to extract the histograms. The histograms are converted into graphic files (png-files) using the standard Java-technology.

The MySQL database was chosen to store the results because it is free, and its bandwidth of the read/write data is enough for the YaPT service. An interaction scheme between the components of the YaPT system is presented in Fig. 1.

Before starting performing any simulation or analysis, each user should connect to the system using an username and a password previously assigned by the administrators. The actions that a user can perform depend on the level of access which range from Level: 5 – Common User; 4 – Account Disabled; 3 – Administrator / Technical Support; 2 – Guest; to Level 1 – Professor.

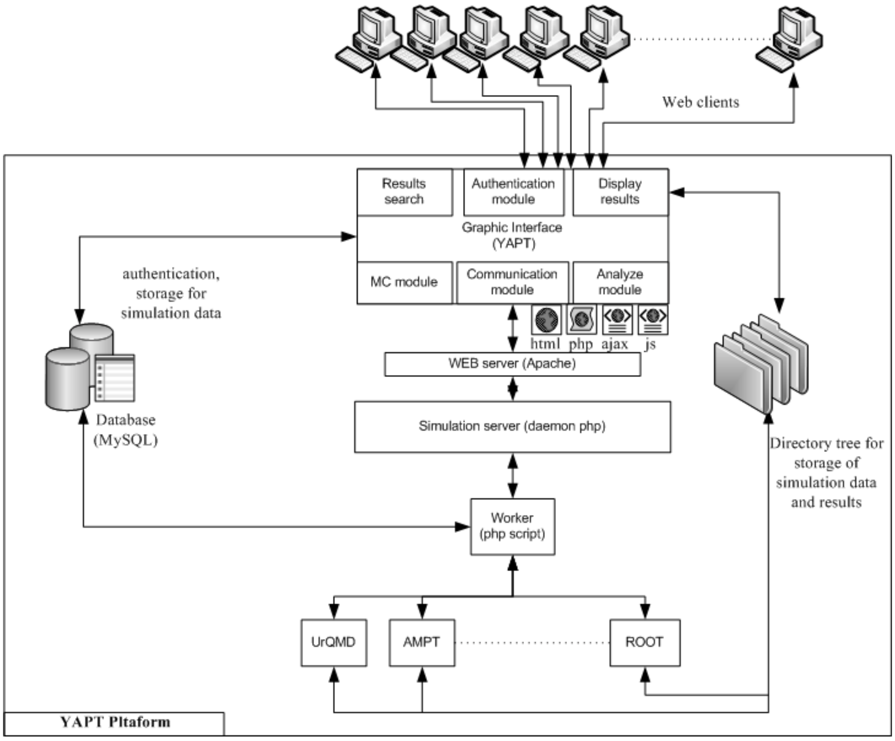


Fig. 1 – Architecture of the YaPT system emphasizing the implemented modules.

Every user – web client – initiates a simulation using an input form which is built in PHP form. Each simulator – UrQMD, AMPT, etc. – uses its own PHP input page with an appropriate interface imposed by specific input requirements. Once the simulation is intiated, the server generate an answer to the query by means of an application based on Ajax technology. The response is conditioned by the correctness of the parameters of the query that server checks. For example, it is checked that the energy of NN collision in the centre-of-mass system is larger than the sum of nucleon masses, or that the range of the impact parameters does not extend to negative values, etc.

If the parameters of the query are eligible, the correct requests are stored in the database. The parameters of each request are compared with the ones existing in the database; if there is an inquire with the same parameters already stored in the database, the simulations aren't performed anymore and the server generate the result page and sends it to the user. If the results for the request are not presented, the server start the simulation and a progress bar appear showing the estimated completion time.

The YaPT service is planned to use distributed computing system developed in NMEC Research Center (NMEC-Grid project) for the simulations. For the moment all calculations are performed by one server which means that all requests for the same simulator will be executed in consecutive order. Therefore, the user can encounter the message "The UrQMD simulator runs 5 simulations" which means that 5 simulations are in progress for the UrQMD simulator or "The AMPT simulator is not used at this time" which means that the AMPT simulator is ready for a new simulation. Consequently, if there is a large number of active simulations, the time of their completion will increase accordingly.

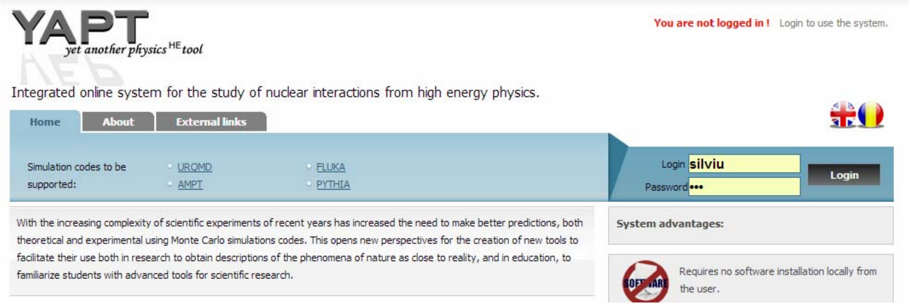
The infrastructure of NMEC Research Center will be developed for starting tasks on several computers which will allow us to perform several queries simultaneously.

3. YaPT MODULES IMPLEMENTATION

The **web interface** represents the YaPT application's page, which can be accessed through any web browser; it links the other modules and facilitates access to the system simultaneously to multiple users. Open Source technologies such as: Linux as Operating System, Apache as web server, MySQL as database server, PHP language for server-side scripting and AJAX for the client side scripts were used for development of this interface. By means of AJAX, our web application can send data to and retrieve data from a server asynchronously (in the background) without interfering with the display and behavior of the existing page.

Using these technologies, the graphical interface of the system will be easy to use and facilitate the user authentication, especially prerequisite for the use of e-learning tool. For securely authenticating users against MySQL database and allow users to perform the simulations, search the database and analyze data we implemented an *authentication system* based on PHP script. The whole interface with the authentication part active is shown in Fig. 2.

The time used by the user to write programs for simulation and analysis will be used only to establish the parameters of the simulation or analysis. The www interface contains, also, a well suited documentation, accordingly with the page displayed which guides the user through various steps of simulation or analysis.



You are not logged in ! Login to use the system.

Integrated online system for the study of nuclear interactions from high energy physics.

Simulation codes to be supported:

Login silviu

Password •••

With the increasing complexity of scientific experiments of recent years has increased the need to make better predictions, both theoretical and experimental using Monte Carlo simulations codes. This opens new perspectives for the creation of new tools to facilitate their use both in research to obtain descriptions of the phenomena of nature as close to reality, and in education, to familiarize students with advanced tools for scientific research.

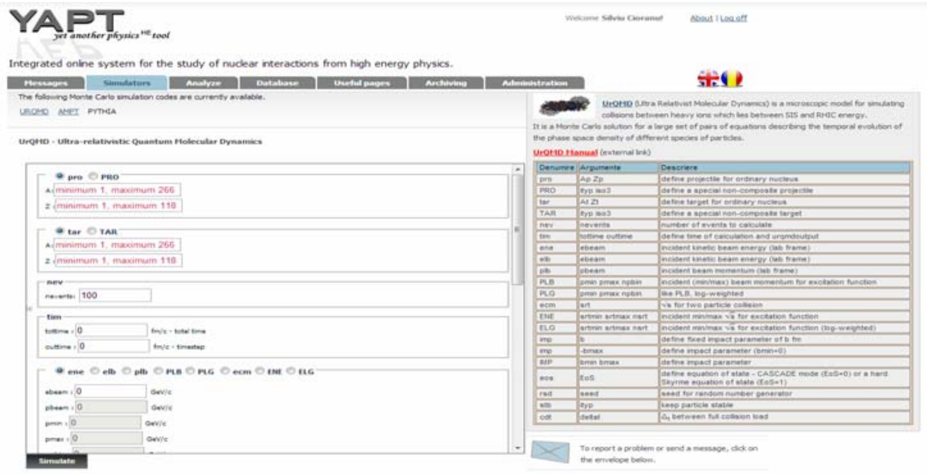
System advantages:

Requires no software installation locally from the user.

Fig. 2 – Web interface of YaPT system.

Once the users has accessed their account the interface will change accordingly with their level of access; for example, an user with low access level (level 5 – common user) will have only the possibility of starting a simulation, saving data on personal computer, make general analysis while an user with higher access level (level 1 – professor) will have, among others, possibility to erase an account or a database entrance, to introduce new analysis macros or to evaluate simulations of other users.

The **Simulations Module,** which is the core of YaPT system, consists of several Monte-Carlo simulation codes (UrQMD, AMPT, Fluka, etc.) implemented and is basically divided into 3 independent parts. Firstly, we have the ***data input form*** which is different for each simulator, adapted accordingly to the codes requirements (Fig. 3). After completion, each form is submitted to the corresponding executable and ***the simulation*** begins; this is the part which is the most time consuming and the one which cannot be controlled by the user.



Integrated online system for the study of nuclear interactions from high energy physics.

The following Monte Carlo simulation codes are currently available.

URQMD AMPT PYTHIA

UrQMD - Ultra-relativistic Quantum Molecular Dynamics

● pro ○ PRO
A: minimum 1, maximum 266
Z: minimum 1, maximum 118

● tar ○ TAR
A: minimum 1, maximum 266
Z: minimum 1, maximum 118

nev
nevents: 100

tim
tottime : 0 fm/c - total time
outtime : 0 fm/c - timestep

● ene ○ elb ○ plb ○ PLB ○ PLG ○ ecm ○ ENE ○ ELG
ebeam : 0 GeV/c
pbeam : 0 GeV/c
pmin : 0 GeV/c
pmax : 0 GeV/c

UrQMD (Ultra Relativist Molecular Dynamics) is a microscopic model for simulating collisions between heavy ions which lies between SIS and RHIC energy.
It is a Monte Carlo solution for a large set of pairs of equations describing the temporal evolution of the phase space density of different species of particles.

UrQMD Manual (external link)

Denumire	Argumente	Descriere
pro	Ap Zp	define projectile for ordinary nucleus
PRO	ityp iso3	define a special non-composite projectile
tar	At Zt	define target for ordinary nucleus
TAR	ityp iso3	define a special non-composite target
nev	nevents	number of events to calculate
tim	tottime outtime	define time of calculation and urqmdoutput
ene	ebeam	incident kinetic beam energy (lab frame)
elb	ebeam	incident kinetic beam energy (lab frame)
plb	pbeam	incident beam momentum (lab frame)
PLB	pmin pmax npbin	incident (min/max) beam momentum for excitation function
PLG	pmin pmax npbin	like PLB, log-weighted
ecm	srt	√s for two particle collision
ENE	srtmin srtmax nsrt	incident min/max √s for excitation function
ELG	srtmin srtmax nsrt	incident min/max √s for excitation function (log-weighted)
imp	b	define fixed impact parameter of b fm
imp	-bmax	define impact parameter (bmin=0)
IMP	bmin bmax	define impact parameter
eos	EoS	define equation of state - CASCADE mode (EoS=0) or a hard Skyrme equation of state (EoS=1)
rsd	seed	seed for random number generator
stb	ityp	keep particle stable
cdt	deltat	Δ_t between full collision load

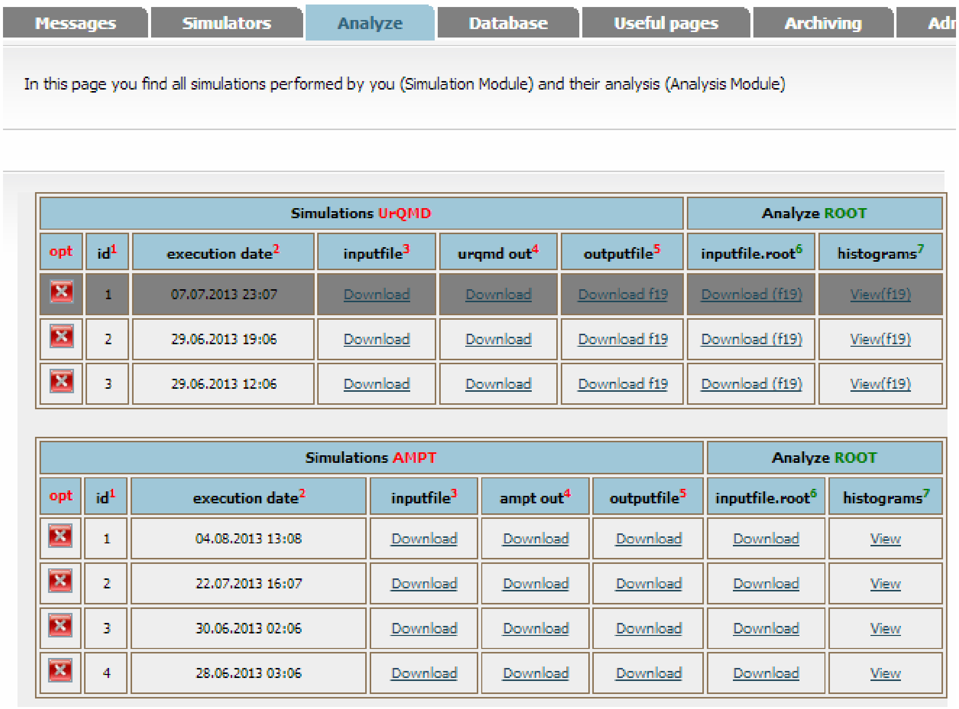
To report a problem or send a message, click on the envelope below.

Fig. 3 – Interface of Simulation module of YaPT system presenting UrQMD input parameters and appropriate instructions for filling.

Finally, after simulation is done, the data are written in the database and the user receives the ***output data table.*** The entrances in this table are shown in Fig. 4 and are important, because they connect the simulation with the analysis module:

- an identification number for simulation (id) - the last simulations will always be displayed first;
- a field with a timestamp of simulation date (execution date);
- an input data file with data introduced by user (*inputfile*);
- a .txt file with the output from the executable which does not contains simulation's results but only information about simulation process (*urqmd_out* or *ampt_out*);

- an ASCII file with simulation results in the standard format of codes.



In this page you find all simulations performed by you (Simulation Module) and their analysis (Analysis Module)

Simulations UrQMD						Analyze ROOT	
opt	id[1]	execution date[2]	inputfile[3]	urqmd out[4]	outputfile[5]	inputfile.root[6]	histograms[7]
☒	1	07.07.2013 23:07	Download	Download	Download f19	Download (f19)	View(f19)
☒	2	29.06.2013 19:06	Download	Download	Download f19	Download (f19)	View(f19)
☒	3	29.06.2013 12:06	Download	Download	Download f19	Download (f19)	View(f19)

Simulations AMPT						Analyze ROOT	
opt	id[1]	execution date[2]	inputfile[3]	ampt out[4]	outputfile[5]	inputfile.root[6]	histograms[7]
☒	1	04.08.2013 13:08	Download	Download	Download	Download	View
☒	2	22.07.2013 16:07	Download	Download	Download	Download	View
☒	3	30.06.2013 02:06	Download	Download	Download	Download	View
☒	4	28.06.2013 03:06	Download	Download	Download	Download	View

Fig. 4 – Interface of Analysis module of YaPT system with example of simulation results from UrQMD and AMPT.

The **Analysis Module** allows users to evaluate the simulations results using various analysis macros. Firstly, users are given the possibility to download the ASCII files generated by the simulations or to convert them into Root type. If the user decides to convert the ASCII file, he/she is connected to the analysis module and the *outputfile.txt* from simulations become *inputfile.root* (Fig. 4). This will be done by means of a script for analysis (analysis macros written using ROOT) that allows obtaining targeted physical quantities as a result of the evaluation file *inputfile.root*. Depending on the access level, users can upload their own analysis macros for specific analysis. For the common users, lists with macros with general analysis are provided.

Once the analysis script is applied, users get a new.root type file *outputfile.root* where the histograms with physical quantities are stored. Using an advanced application developed specially for YaPT, histograms are converted into .png file and can be displayed online.

The ROOT files created from results generated by the Monte-Carlo simulator have a tree structure with each branch representing a collection of similar data – TClonesArray (Fig. 5). Inside the file there are a TTree and a folder where histograms could be saved (divided in folders). If a branch contains a single entry per event, the branch is then represented by a TClonesArray with only one entry.

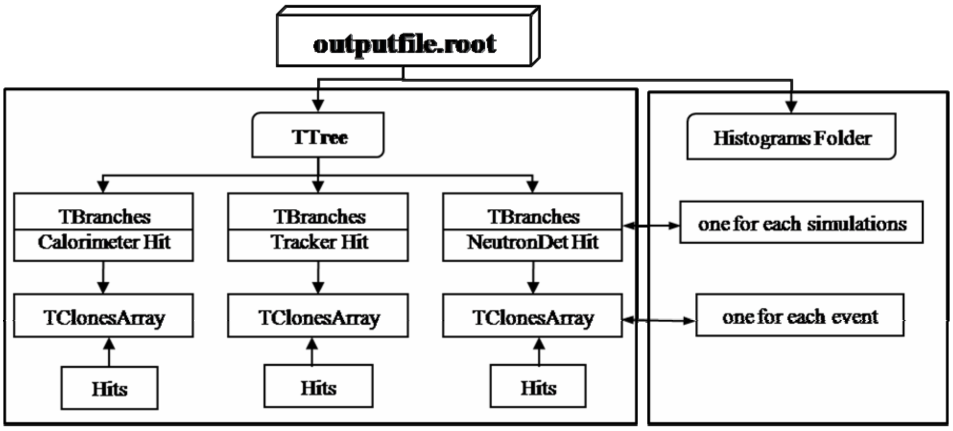


Fig. 5 – Structure of *outputfile.root* containing results from simulations.

This complex structure is made to be able to store and recover for the analysis, the typical event structure in simulations. Each TClonesArray object contains the Hits which have been generated in an event. So, many Hits could be created in an event and the Hits are stored in an array (the TClonesArray) for each event. Then, the TTree stores each TClonesArray for each event. All the information is compressed and stored (persistency) in a ROOT file.

To analyze the data, the ROOT file should be opened and the events reconstructed one by one. For this purpose we perform an "event loop", getting from the file the TClonesArray objects stored per event. Then, we can iterate on the TClonesArray for obtaining the Hits in each event. A sample code which reads the TTree and loops on the events is available.

4. DATA FLOW

A simplified data flow diagram for YaPT service is shown in Fig. 6. Web-oriented structure of YaPT allows permanent interaction with users. Modularized structure of YaPT allows to access data from different file formats. There are the levels that control the data flow: *input level* where all the parameters for simulations are set, *application level* where simulations are performed and *output level* where data can be saved and analyzed. All these levels are connected: when any of the conditions is changed the other levels are automatically updated.

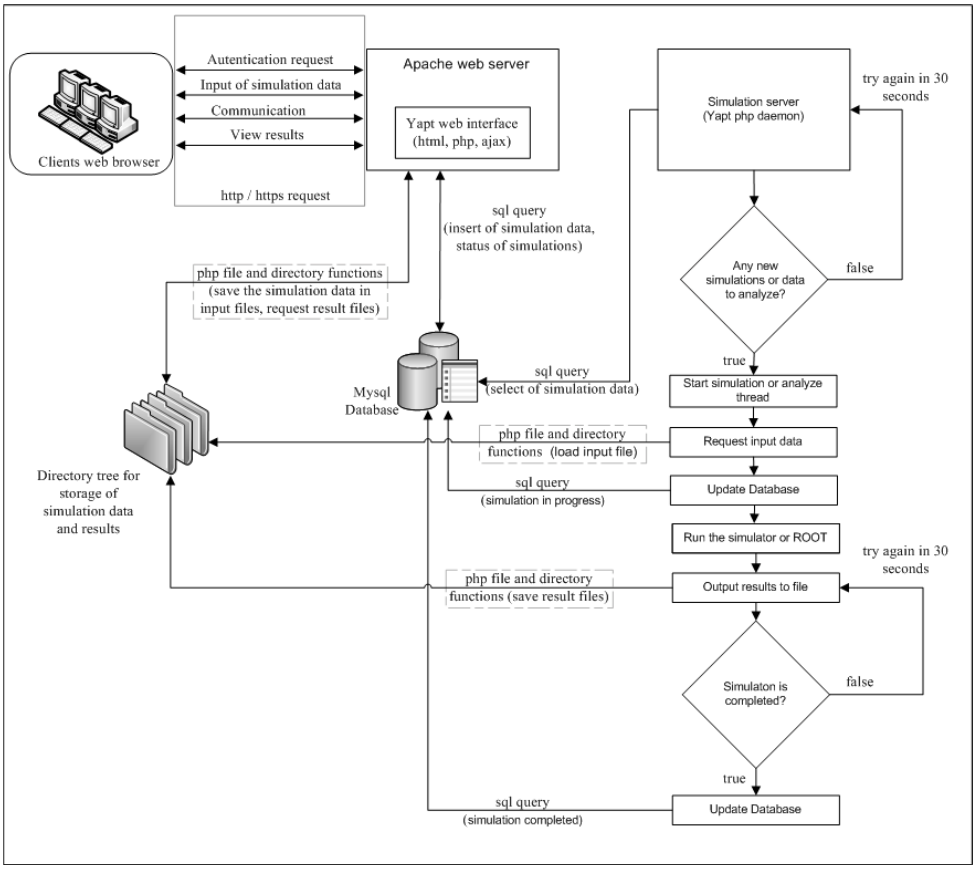

Autentication request
Input of simulation data
Communication
View results
Clients web browser
http / https request
Apache web server
Yapt web interface (html, php, ajax)
Simulation server (Yapt php daemon)
try again in 30 seconds
sql query (insert of simulation data, status of simulations)
php file and directory functions (save the simulation data in input files, request result files)
Any new simulations or data to analyze?
false
true
Mysql Database
sql query (select of simulation data)
Start simulation or analyze thread
Directory tree for storage of simulation data and results
php file and directory functions (load input file)
Request input data
sql query (simulation in progress)
Update Database
Run the simulator or ROOT
try again in 30 seconds
php file and directory functions (save result files)
Output results to file
Simulaton is completed?
false
true
sql query (simulation completed)
Update Database

Fig. 6 – Flow chart describing the YaPT workflow.

5. USE CASE

As we previously mentioned there are two ways that this system can be used: for research and as educational software.

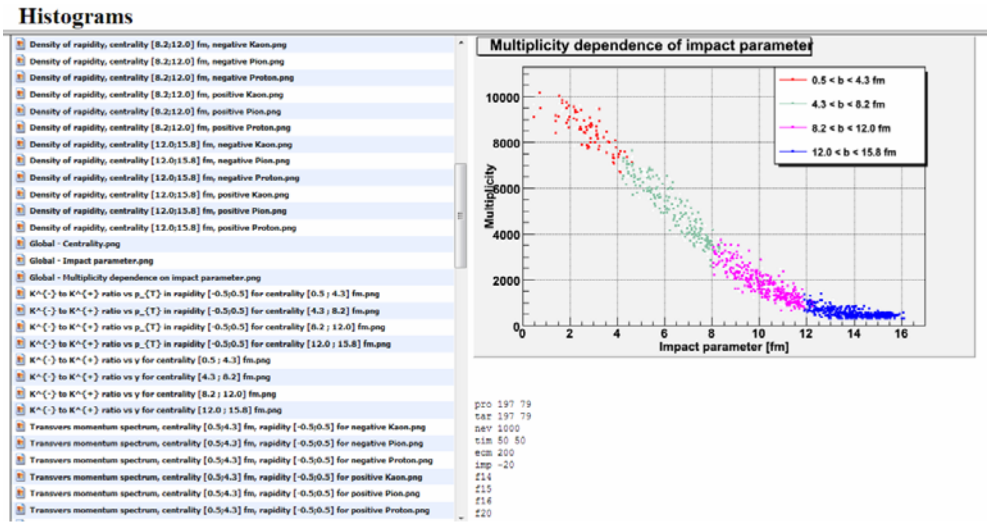
In order to illustrate the YaPT performances with concrete example we will present in the following a use case. In this example, the study of centrality dependence of the particles generated in Au+Au collisions at 200 GeV will be represented using UrQMD and AMPT codes. This example use macros for analysis developed solely for YaPT application and included into common analysis macros of YaPT and are meant to be easy to understand.

To perform the simulations we follow several steps:

- introduce the collision parameters into YaPT input form: A and Z of projectile and target, number of events to calculate, impact parameter (b_{min} and b_{max}), collision energy in center of mass frame;

- run the command to perform the simulation (Simulate);
- convert output .txt file into .root type file;
- apply macros for analysis. This macros generated the histograms with specific physical quantities. As a result, the main page looks as it is presented in Fig. 7.

This example was chosen because the study of centrality dependence of the characteristics of nucleus-nucleus interactions is an important experimental way for obtaining information on phases of strongly interacting matter formed during the collision and in the subsequently evolution at the fireball at different energies. On the other hand the centrality of collisions can not be defined directly in the experiment, therefore the need for simulations. In experiments the values of the centrality are defined as a number of identified projectiles' and targets' fragments, as the energy flow of the particles with emission angles θ ≈ 0° or with θ ≈ 90°.



Histograms

Density of rapidity, centrality [8.2;12.0] fm, negative Kaon.png
Density of rapidity, centrality [8.2;12.0] fm, negative Pion.png
Density of rapidity, centrality [8.2;12.0] fm, negative Proton.png
Density of rapidity, centrality [8.2;12.0] fm, positive Kaon.png
Density of rapidity, centrality [8.2;12.0] fm, positive Pion.png
Density of rapidity, centrality [8.2;12.0] fm, positive Proton.png
Density of rapidity, centrality [12.0;15.8] fm, negative Kaon.png
Density of rapidity, centrality [12.0;15.8] fm, negative Pion.png
Density of rapidity, centrality [12.0;15.8] fm, negative Proton.png
Density of rapidity, centrality [12.0;15.8] fm, positive Kaon.png
Density of rapidity, centrality [12.0;15.8] fm, positive Pion.png
Density of rapidity, centrality [12.0;15.8] fm, positive Proton.png
Global - Centrality.png
Global - Impact parameter.png
Global - Multiplicity dependence on impact parameter.png
K^{-} to K^{+} ratio vs p_{T} in rapidity [-0.5;0.5] for centrality [0.5 ; 4.3] fm.png
K^{-} to K^{+} ratio vs p_{T} in rapidity [-0.5;0.5] for centrality [4.3 ; 8.2] fm.png
K^{-} to K^{+} ratio vs p_{T} in rapidity [-0.5;0.5] for centrality [8.2 ; 12.0] fm.png
K^{-} to K^{+} ratio vs p_{T} in rapidity [-0.5;0.5] for centrality [12.0 ; 15.8] fm.png
K^{-} to K^{+} ratio vs y for centrality [0.5 ; 4.3] fm.png
K^{-} to K^{+} ratio vs y for centrality [4.3 ; 8.2] fm.png
K^{-} to K^{+} ratio vs y for centrality [8.2 ; 12.0] fm.png
K^{-} to K^{+} ratio vs y for centrality [12.0 ; 15.8] fm.png
Transvers momentum spectrum, centrality [0.5;4.3] fm, rapidity [-0.5;0.5] for negative Kaon.png
Transvers momentum spectrum, centrality [0.5;4.3] fm, rapidity [-0.5;0.5] for negative Pion.png
Transvers momentum spectrum, centrality [0.5;4.3] fm, rapidity [-0.5;0.5] for negative Proton.png
Transvers momentum spectrum, centrality [0.5;4.3] fm, rapidity [-0.5;0.5] for positive Kaon.png
Transvers momentum spectrum, centrality [0.5;4.3] fm, rapidity [-0.5;0.5] for positive Pion.png
Transvers momentum spectrum, centrality [0.5;4.3] fm, rapidity [-0.5;0.5] for positive Proton.png

```
pro 197 79
tar 197 79
nev 1000
tim 50 50
ecm 200
imp -20
f14
f15
f16
f20
```

Fig. 7 – General view of the results page indicates the files storing the histograms (left), selected histogram (top right) and file with input parameters (bottom right).

By means of macros already implemented we can evaluate the following histograms which show the dependencies of various physical quantities on collision centrality and can be further compared with data from literature:

1. Particle multiplicities gives calculations of multiplicities of π-mesons, K-mesons, protons, anti-protons and other particles;

2. Particle multiplicity ratios dependences of relative multiplicities of p-/p, K-/K+, π -/π+ etc. of rapidity and transverse momentum;

3. Multiplicity distributions and correlations calculations of particle multiplicity distributions and correlation dependencies between particle multiplicities such as: particle multiplicity dependence on the impact parameter in

nucleus-nucleus interactions, correlations between multiplicities of particles produced in different phase-space regions, correlations between energies of neutral and charged particles etc.;

4. Rapidity distributions – the rapidity distributions for K-, K+, pi-, pi+, p- and p for four centrality classes: [0.5; 4.3], [4.3; 8.2], [8.2; 12.0] and [12.0; 15.8] in UrQMD and [0.9; 5.1], [5.1; 9.2], [9.2; 13.3] and [13.3; 17.5] in AMPT.

5. P_T-distributions represented for K-, K+, pi-, pi+, p- and p for the same centrality classes as in rapidity distributions case and for rapidity range from –0.5 to 0.5. Figure 8 presents the P_T-distributions for [0.5; 4.3] centrality obtained using UrQMD (left), and for [0.9; 5.1] centrality using AMPT (right).

6. Flow-calculations of characteristics of flow in nucleus-nucleus interactions (v1, v2, px, F).

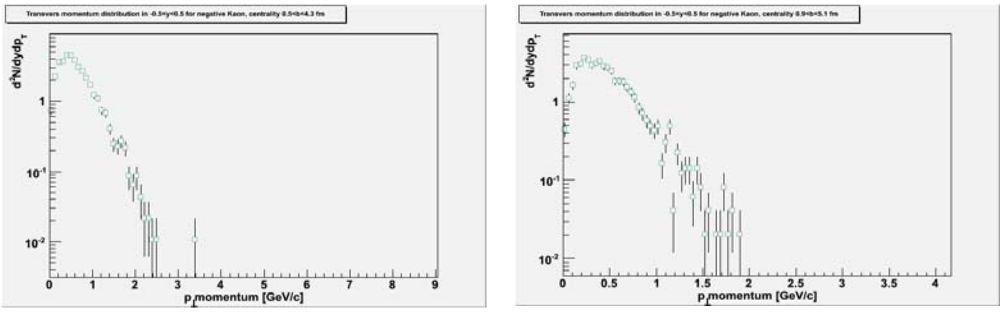


Fig. 8 – P_T-distributions for K^- with [0.5; 4.3] centrality obtained using UrQMD (left) and with [0.9; 5.1] centrality using AMPT (right).

This tool can be used also as educational software for master and PhD students. As was previously demonstrated [7], teaching Physics using computer assisted methods creates an active learning environment, which allows students to participate *via* remote laboratory techniques to real experiment activities [8].

The main advantages of the YaPT system as educational software are:

- Provide an opportunity for students to evaluate a project in a way closer to the way is done in research. Simulations are an excellent predictor for how results in real research activities looks like.
- The system is results oriented with a minimum background in physics and no programming experience required. In this context students can focus on learning physics not programming.
- Simulations activities are usually time consuming. The help of YaPT is essential because the platform allow users to start a simulation and check later for the results or to evaluate the results from an already existing simulation.
- Obtaining visual feedback, from simulations based on theoretical models, makes the models and algorithms more meaningful.

- Once students know how simulations work they can write and test their own macros for analysis.

The YaPT platform is already used in extensive tests by the students in the Physics of Atom, Nucleus, Elementary Particles, Astrophysics and Applications Master Program and PhD Program in the same field, at the Faculty of Physics, University of Bucharest, as a laboratory course. Apart of helping students to learn how to work with simulation programs, the platform will help us also to detect the flaws in the system and to improve it.

6. CONCLUSION

We discussed the first version of YaPT system, a framework designed to perform online simulation of hadron-nucleus and nucleus-nucleus interactions for high energy physics. The modular design of YaPT was presented, and we described the principles used for modeling the interactions and parameterizing the event simulations.

We showed that YaPT is able to produce online analysis and display the results of simulations. Therefore, it can be used to perform quickly realistic Physics studies without in-depth knowledge of the technicalities of a simulation process.

Acknowledgments. This paper is supported by the Sectorial Operational Programme Human Resources Development (SOP HRD), financed from the European Social Fund and by the Romanian Government under the contract number SOP HRD/107/1.5/S/ 82514.

REFERENCES

1. http://hpc.fizica.unibuc.ro
2. S.A. Bass, M. Belkacem, M. Bleicher, M. Brandstetter, L. Bravina, C. Ernst, L. Gerland, M. Hofmann, S. Hofmann, J. Konopka, G. Mao, L. Neise, S. Soff, C. Spieles, H. Weber, L.A. Winckelmann, H. Stocker, W. Greiner, Ch. Hartnack, J. Aichelin and N. Amelin, *Microscopic Models for Ultrarelativistic Heavy Ion Collisions*, Prog.Part.Nucl.Phys., **41**, 225–370 (1998), see also: http://urqmd.org/
3. Zi-Wei Lin, Che Ming Ko, Bao-An Li, Bin Zhang, Subrata Pal, *Multiphase Transport Model for Relativistic Heavy Ion Collisions*, Phys. Rev. C, **72**, 064901 (2005), see also: http://personal.ecu.edu/linz/ampt/
4. E.I. Alexandrov, V.M. Kotov, V.V. Uzhinsky, P.V. Zrelov, *WEB Portal for Monte Carlo Simulations in High Energy Physics – HEPWEB*, arXiv:1208.6439, see also: http://hepweb.jinr.ru/
5. J.M. Butterworth, S. Butterworth, *JetWeb: A WWW interface and database for Monte Carlo tuning and validation*, Computer Physics Communications, **153**, *2*, 164–178 (2003), see also: http://jetweb.hep.ucl.ac.uk/
6. Rene Brun and Fons Rademakers, *ROOT – An Object Oriented Data Analysis Framework*, Proceedings AIHENP'96 Workshop, Lausanne, Sep. 1996, Nucl. Inst. & Meth. in Phys. Res. A, **389**, 81–86 (1997), see also: http://root.cern.ch
7. C. Kuncser, A. Kuncser, G. Maftei, S. Antohe, *Computer assisted methods in teaching basics of magnetism at high school level*, Romanian Reports in Physics, **64**, *4*, P. 1119–1130 (2012).
8. S. Moraru, I. Stoica, F.F. Popescu, *Educational software applied in teaching and assessing physics in high schools*, Romanian Reports in Physics, **63**, *2*, 577–586 (2011).