Computing for the next generation flavour factories

F Bianchi¹, D Brown², M Corvo³, A Di Simone⁴, A Fella⁵, A Gianoli⁶, E Luppi⁶, M Morandin⁷, E Paoloni⁵, M Rama⁸ and L Tomassetti⁶

¹INFN Torino, Torin, Italy
²Lawrence Berkeley National Laboratory, Berkeley, USA
³University of Padua and INFN, Padua, Italy
⁴University of Rome Tor Vergata, Rome, Italy
⁵INFN Pisa, Pisa, Italy
⁶University of Ferrara, Physics Department and INFN, Ferrara, Italy
⁷INFN Padua, Italy
⁸INFN Frascati National Laboratory, Frascati, Italy

E-mail: armando.fella@pi.infn.it

Abstract. The next generation of Super Flavor Factories, like SuperB and SuperKEKB, present significant computing challenges. Extrapolating the BaBar and Belle experience to the SuperB nominal luminosity of $10^{36}\text{cm}^{-2}\text{s}^{-1}$, we estimate that the data size collected after a few years of operation is 200 PB and the amount of CPU required to process them of the order of 2000 KHeR-Spec06. Already in the current phase of detector design, the amount of simulated events needed for estimating the impact on very rare benchmark channels is huge and has required the development of new simulation tools and the deployment of a worldwide production distributed system.

With the collider in operation, very large data set have to be managed and new technologies with potential large impact on the computational models, like the many core CPUs, need to be effectively exploited. In addition SuperB, like the LHC experiments, will have to make use of distributed computing resources accessible via the Grid infrastructures while providing an efficient and reliable data access model to its final users. To explore the key issues, a dedicated R&D program has been launched and is now in progress. A description of the R&D goals and the status of ongoing activities is presented.

1. Introduction

The computing models of the BaBar [2] and Belle [3] experiments have proven to be quite successful for a flavor factory in the $\mathcal{L} = 10^{34}\text{cm}^{-2}\text{s}^{-1}$ luminosity regime. A similar computing model can also work for a super flavor factory at a luminosity of $\mathcal{L} = 10^{36}\text{cm}^{-2}\text{s}^{-1}$. Data volumes will be much larger, comparable in fact to those expected for the first running periods of the ATLAS [4] and CMS [5] experiments at the LHC, but predictable progress in the computing industry will provide much of the performance increase needed to cope with them. In addition, effective exploitation of computing resources on the Grid, that has become well established in the LHC era, will enable SuperB to access a much larger set of resources than were available to BaBar.

To illustrate the scale of the computing problem and how the SuperB computing group envisage to attack it, the first part of this document contains an overview of the current BaBar-inspired computing baseline model with an estimate of the extrapolated SuperB computing...
requirements, followed by a description of the current development and implementation timeline. In the current view, the design phase of the SuperB computing model is planned to start with a dedicated R&D program in the first year of the project and to finish with the completion of the Computing Technical Design Report (TDR) by the end of the second year.

So far, the main effort of the computing group has been devoted to the development and the support of the simulation software tools and the computing production infrastructure needed for carrying out the detector design and performance evaluation studies for the Detector TDR. Quite sophisticated and extended detector and physics studies can now be performed thanks to:

- the development of a detailed Geant4-based Monte Carlo [25] simulation (called Bruno or referenced as FullSim) and of a much faster parametric fast simulation (FastSim) which can directly leverage the existing BaBar analysis code base;
- the implementation of a production system for managing very large productions that can parasitically exploit the computing resources available on the European and US Grids.

2. Computing requirement evaluation

The SuperB computing requirements can be estimated using as a basis the present experience with BaBar and applying a scaling of about two orders of magnitude. Fortunately, much of this scaling exercise is quite straightforward.

As a baseline, all rates are simply scaled linearly with luminosity. Only a few parameters have been modified to keep into account improved efficiency of utilization of the computing resources that are likely to be obtained with SuperB, i.e.:

- the skimmed data storage requirements have been reduced (by $\sim 40\%$), assuming a more aggressive use of event indexing techniques;
- the CPU requirements for physics analysis are reduced by a factor of two as a result of more stringent optimization goals that can be achieved in SuperB;
- the duration of the reprocessing and simulation re-generation cycle, expected to take place once significant improvements of the reconstruction code physics performance have been obtained, has been set to two years instead of one, as it was in BaBar, in view of the larger expected cost-to-benefit ratio;

The total computing resources needed for one year of data taking at nominal luminosity are of the same order as the corresponding figures estimated, in the spring of 2010, by the Atlas and CMS experiments for the 2011 running period. However SuperB will profit from the technological advances that will take place over a period of approximately 10 years, and make extensive use of distributed computing resources accessible via the Grid infrastructures.

3. FastSim: the parametrized Monte Carlo simulation tool

A detailed simulation of the SuperB detector, with its various options, with sufficient statistical precision for a relevant physics result, is beyond the capability of the current SuperB computing effort. In order to measure the impact of various detector design options on final physics results, and to estimate the physics reach of SuperB in key benchmark channels, a fast simulation (FastSim) program has been developed. FastSim relies on simplified models of the detector geometry, materials, response, and reconstruction to achieve an event generation rate a few orders of magnitude faster than is possible with a Geant4-based detailed simulation, but with sufficient detail to allow realistic physics analysis. In order to produce more realistic results, FastSim incorporates the effects of expected machine and detector backgrounds. FastSim is configurable at runtime, and is compatible with the BaBar analysis framework, allowing sophisticated analysis to be performed with minimal software development. FastSim uses the same event generation tools used by BaBar, which can generate $e^+e^- \rightarrow \Upsilon(4S) \rightarrow B\bar{B}$ events,
continuum \( e^+ e^- \rightarrow q\bar{q} \) events \((q = u, d, s, c)\), and \( e^+ e^- \rightarrow \tau^+ \tau^- \) with polarized beams. The beam energies can be adjusted within the range of the Super\( B \) accelerator.

FastSim models the Super\( B \) detector as a collection of detector elements, with 2-dimensional structure and given material properties. The overall detector geometry is assumed to be cylindrical. Simplified cross sections and standard formulas are used to model particle-material interactions. Unstable particles are allowed to decay during their traversal of the detector, using EvtGen[23] code to describe the decays. Sensitive components are modeled by optionally adding a measurement type to an element. The geometry and properties of detector element materials and measurement types are defined through a set of XML files using a dedicated schema.

All measurement types relevant to Super\( B \) are implemented in FastSim. Tracking detectors such as Si strip and pixel trackers, wire chambers with axial and stereo layers, with position and ionization measurements \((dE/dx)\) are implemented. In the EM calorimeter the energy deposits are cluster to account for shower fluctuations and crystals energy resolution. In the muon detector (IFR) the hadronic shower profile is to generate scintillator hits. Cherenkov rings from the DIRC are simulated using a lookup table based on BaBar data. Timing measurements are modeled based on their intrinsic resolution. FastSim reconstructs high-level detector objects (tracks and clusters) from simulated low-level detector objects (hits and energy deposits), using the simulation truth to associate detector objects. Reconstruction errors are modeled by perturbing the truth-based association. Tracking hits from nearby particles are merged, or mis-associated to model pattern recognition errors. Calorimeter signals from different particles are summed across a grid representing the crystals, and grid cells are clustered to define cluster energy, position, and shape. IFR hits are clustered and fit to compute the number of interaction lengths crossed by the particle.

Machine backgrounds are generated in dedicated FastSim or full simulation (Geant4) runs, stored as \( TParticles[24] \). Multiple background events are overlaid on each physics event using Poisson sampling. Particles from background events are simulated exactly as those from the physics event, modulated by a randomly assigned background time. Hit-merging, pattern recognition confusion, and cluster merging are performed after background overlay, to model the effect of backgrounds on resolution.

FastSim is compatible with the BaBar analysis framework, which allows existing BaBar analysis to be run. This allows optimizing the Super\( B \) design by studying the performance of different detector layouts in terms of the physics reach in some benchmark channels, for example, the rare decay \( B \rightarrow K^* \nu\bar{\nu} \).

Thus FastSim allows a quantitative comparison of these options in terms of the equivalent luminosity to achieve a given sensitivity.

Typical per event FastSim processing times on a dual quad core Intel(R) Xeon(R) CPU E5520 @ 2.27GHz architecture are 1 ms for particle generation, 10 ms for propagation of particles through the detector, 100 ms for reconstruction, between 100 and 1000 ms for composites selection depending on the event complexity. A public version of FastSim is planned to be released in the near future.

4. Bruno: the Super\( B \) full simulation tool

The availability of reliable tools for full simulation is crucial in the present phase of the design of both the accelerator and the detector. For example, the background rate at the sub-detectors needs to be carefully assessed for each modification in the accelerator design and, for a given background scenario, the sub-detectors’ design must be optimized. The full simulation tool can be used to improve the results of the FastSim in some particular cases, as discussed in the following. The choice was made to re-write from scratch the core simulation software, aiming at having more freedom to better profit from both the BaBar legacy and the experience gained in the development of the full simulation for the LHC experiments. Geant4 and the C++
programming language were therefore the natural choices as underlying technology.

4.1. Geometry description
The need to re-use as much as possible the existing geometrical description of the BaBar full simulation called for some application-independent interchange format to store the information concerning the geometry and materials of the sub-detectors. Among the formats currently used in High Energy Physics applications, the Geometry Description Markup Language (GDML) was chosen because of the availability of native interfaces in Geant4 and ROOT, and the easiness of human inspection and editing provided by the XML-based structure.

4.2. Simulation input: Event generators
The event generator code can be run either inside the Bruno executable or as a separate process. In the latter case the results are saved in an intermediate file, which is then read by the full simulation job. Bruno currently supports two interchange formats: a plain text file and one in ROOT format.

4.3. Interplay with FastSim
The event snapshot at a specific sub-detector boundary can also be read by FastSim, allowing a very powerful hybrid simulation approach. For instance the design of the interaction region, which strongly influences the background rates in the detector, cannot be described with the required level of detail in FastSim, while full simulation is not fast enough to generate the high statistics needed for physics studies. By using Bruno to simulate background events up to and including the interaction region, and saving a snapshot of the event without running the entire simulation, one obtains a set of background frames, which can be read back in FastSim, that then propagates particles through the simplified detector geometry and adds the resulting hits to the ones coming from signal events.

Another aspect where the interplay between fast and full simulation is needed is the evaluation of the neutron background. The concept is to have Bruno, in addition to handling all particle interactions within the interaction region, as explained above, also track neutrons in the whole detector until they interact or decay, saving the products as part of the background frame used by FastSim. All these functionalities are currently implemented, and have been used in recent production runs.

5. The distributed production system
To design the detector and to extract statistically significant results from the data analysis, a huge number of Monte Carlo simulated events is needed. Such a production is way beyond the capacity of a single computing farm so it was decided to design, even to support the detector TDR studies, a distributed model capable of fully exploiting the existing HEP world wide Grid computing infrastructure [9, 10, 11, 12, 13].

The LHC Computing Grid (LCG) architecture [18] was adopted to provide the minimum set of services and applications upon which the SuperB distributed model could be built, and the INFN Tier1 site located at CNAF (Bologna) was chosen as the central site where job submission management, the Bookkeeping Data Base, and the data repository would reside. The experiment distributed computing infrastructure spans over several sites in Europe and North America and we need to manage transparently the different Grid middleware flavors depending on the sites geographical position.

A web-based user interface has been developed which takes care of the bookkeeping database interactions and the job preparation; is also provides basic monitoring functionalities. Moreover, a simple automatic submission procedure for non-expert users has been prepared in order to
maintain data consistency, and to optimize the production cycle. It has two different sections for Full Simulation and Fast Simulation, both taking care of job submission and monitoring [7, 8].

6. Conclusions and future plans
In 2001, just at the time the PEP-II/BABAR and KEKB/Belle projects were publishing their first results on the CP-violating asymmetries in B0 meson decay, it became clear that the sensitivity to measure the effects of physics beyond the Standard Model in heavy quark and heavy lepton decays required a two-order-of-magnitude increase in collider luminosity.

The first approaches were based on straightforward extrapolations of the parameters of the existing PEP-II and KEKB colliders. The resulting regime, with more beam bunches in the ring and more particles per beam bunch, was studied for several years. Nowadays SuperB and the competitor SuperKEK experiments are focusing the effort in Technical Design Report (TDR) definition.

SuperB experiment has been approved by Italian government on December 2010. The computing group is concentrating on support for SuperB detector TDR: the simulation of physics, detector behaviors and background studies is the first core goal at present time. The SuperB computing TDR planned to be released one year after detector TDR definition (second half of 2012) is in its start design phase. It will include the description of definitive experiment computing model. The outline of an R&D program to be carried out in next two years has been defined and the works on key subjects has started. In the design of the final experiment computing model we will take into account new CPU architecture, software architecture and framework. This is related, for example, to the multi-core and many-core revolution, which permits a significant increase in available execution cycles within a constant processor power envelope. The multi-core processors have already permitted to benefit from improvements in overall performance, without a corresponding increase in processor power consumption, and many-core processors or GPUs are to be investigated.

As almost all HEP programmes (simulation, reconstruction, data analysis, etc.) are written by high energy physicists, it is crucial to understand which modifications in the code can provide the most benefit from a multi-core or many-core architecture. Another crucial point is the strategic deployment of the experiment database applications in a distributed computing environment, together with the model of efficient data handling and data storage systems.

The TDR definition phase will be followed by the R&D results integration into complete SuperB software architecture converging on final full-scale system. The experiment commissioning has been scheduled for 2016.

References
[1] The SuperB Collaboration, SuperB Progress Report, Detector, http://arxiv.org/abs/1007.4241v1
[2] B. Aubert et al., Nucl. Instr. Meth. Phys. Res., A 479, 1 (2002).
[3] The Belle Collaboration, The Belle Detector, Nucl. Instrum. Methods Phys. Res., Sect. A 479, 117 (2002).
[4] The ATLAS Collaboration, ATLAS Detector and Physics Performance Technical Design Report, http://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/TDR/access.html.
[5] The CMS Collaboration, CMS Detector Technical Design Report, http://cmsdoc.cern.ch/cms/cpt/tdr/.
[6] R. Andreassen et al, FastSim: fast simulation of the SuperB detector, Proceeding of conference IEEE NSS-MIC 2010, Knoxville, TN, USA (2010) A. Di Simone, I. Gaponenko, E. Manoni, A. Perez, M. Rama, D. Roberts, M. Rotondo, G. Sini, M. Sokoloff, A. Suzuki, J. Walsh, FastSim: fast simulation of the SuperB detector, Proceeding of conference IEEE NSS-MIC 2010, Knoxville, TN, USA (2010)
[7] D. Brown, M. Corvo, A. Di Simone, A. Fella, E. Luppi, E. Paoloni, R. Stroili, L. Tomassetti, The Distributed Production System of the SuperB Project: Description and Results. Proceeding of conference CHEP 2010, Computing for High Energy Physics, Taipei, Taiwan, 18-22 October 2010
[8] D. Brown, M. Corvo, A. Di Simone, A. Fella, E. Luppi, E. Paoloni, R. Stroili, L. Tomassetti First Results from the SuperB Simulation Production System. Proceeding of conference IEEE 2010, NSS-MIC 2010, Knoxville, TN, USA (2010)
[9] http://www.egi.eu
[10] http://www.opensciencegrid.org
[11] http://www.nordugrid.org
[12] http://www.westgrid.ca
[13] The LCG TDR Editorial Board, LHC Computing Grid, Technical Design Report LCG-TDR-001 CERN-LHCC-2005-024 (2005)
[14] http://ganga.web.cern.ch/ganga
[15] http://glite.web.cern.ch/glite
[16] http://hep-project-grid-scg.web.cern.ch/hep-project-grid-scg/voms.html
[17] https://twiki.cern.ch/twiki/bin/view/LCG/LfcAdminGuide
[18] http://lcg.web.cern.ch/LCG
[19] http://sdm.lbl.gov/srm-ug/doc/SRM.v2.2.html
[20] A Corso et al. StoRM, an SRM Implementation for LHC Analysis Farms Computing in High Energy Physics (CHEP 2006), Mumbai, India, Feb. 13-17
[21] http://hadoop.apache.org/
[22] Fielding R.T., Architectural Styles and The Design of Network-based Software Architectures, PhD Thesis, University of California Irvine, 2000
[23] http://robbep.home.cern.ch/robbep/EvtGen/GuideEvtGen.pdf
[24] http://root.cern.ch/
[25] http://geant4.web.cern.ch/geant4/