

Boost Summer of Code 2016: Funding Proposal

Author: Bryce Adelstein Lelbach <bryce@cppnow.org>

0.) Introduction

Since 2006, Boost has participated in Google's Summer of Code (SoC) initiative, an annual program that provides student developers with sponsorships to contribute to an open source project during the summer. Over the years, Boost's participation in this program has yielded a large volume of contributions to Boost (detailed in Section 1). **~32%** of the 65 Boost SoC projects have been **accepted** into Boost and an additional **~20%** are either **in use by the community** or **under review** for adoption into Boost. Additionally, the program has helped recruit young developers to our community (detailed in Section 2).

This year, Boost's application to Google's SoC program was rejected. Boost's SoC application was also rejected in 2012. In both cases, the application was not rejected due to merit. The Google SoC administration has informed us that organizations which have participated in Google SoC for multiple years are sometimes rejected to increase the number of new organizations participating in Google SoC each year. We anticipate that this pattern will continue going forwards; typically, Boost's application will be accepted, but sometimes we will be rotated out of the program for a year.

Due to the lack of funding from Google this year, we would like to request that the Boost Steering Committee authorize **\$18,000 USD** to fund this year's Boost Summer of Code independent of Google. To minimize the investment risk to Boost, we have solicited three proposals from "veteran" students and mentors who have participated in Boost SoC's program in previous years. Our plan for the administration of this program is described in Section 3. A breakdown of the budget is provided in Section 4. Section 5 presents the three solicited proposals for this year.

1.) Contributions to Boost

Boost's SoC program has typically received funding for 5 to 10 projects each year. 65 individual projects have been funded during our 10 years of participation in Google's program. Typically, proposals are either new libraries that are intended as candidates for inclusion into Boost or contributions to existing Boost libraries. Out of the 65 SoC projects to date:

- **21 projects (~32%)** have been **accepted into Boost**.
- **13 projects (~20%)** have not been accepted, but are either **in use by the community** or **undergoing review**.

The following Boost libraries and candidate libraries started as Boost SoC projects:

- Boost.Heaps (accepted)
- Boost.Lockfree (accepted)
- Boost.Hana (accepted)
- Boost.AFIO (pending)
- Boost.Process (submitted, rejected once before)
- Boost.HTTP (rejected)

Additionally, the following Boost libraries have accepted contributions developed during SoC projects:

- Boost.Math, 4 projects (special functions, multi-core/SIMD optimizations)
- Boost.Graph, 4 projects (mincut/maxflow, user-friendly graphs, graph partitioning)
- Boost.Odeint, 2 projects (preparation for review, parallelization)
- Boost.Compute, 2 project (various improvements)
- Boost.Fusion, 2 projects (C++11 port)
- Boost.Phoenix, 1 project (v3 rewrite)
- Boost.Geometry, 1 project (spatial indexing)
- Boost.Polygon, 1 project (sweep-line algorithm)
- Boost.StringAlgo, 1 project (new algorithms)
- Boost.Python, 1 project (py3k support)

A majority of Boost SoC projects are mostly completed (70-90%) during the course of the program, even if they are not accepted or proposed for inclusion in Boost. Despite the lack of adoption by Boost, these projects should not necessarily be considered failures. For example, a derivative of the original candidate Boost.Coroutine library (developed by Giovanni Deretta during Boost SoC 2016, distinct from the Boost.Coroutine library by Oliver Kowalke) is in use today in the HPX parallel runtime system, a large open-source C++ project.

In summary, the Boost SoC program has been successful at contributing to Boost. **1 out of 2** SoC projects have **made a contribution to Boost**.

2.) Community Building and Recruitment

The adoption of a SoC project into Boost is not necessarily a measure of the project's success. The community-building and recruiting aspects of SoC should not be discounted. Boost is about 16 years old now. Many original Boost authors are sadly no longer active in the community. Maintainer turn-over has become a significant impediment for Boost. The Community Maintenance Team was created to help address this issue, but as a long-term solution we need to bring new developers into our community. SoC is an excellent way for us to recruit and support young Boost developers.

61 students have participated in the Boost SoC program over the years. Today, a number of these former students are involved members of our community, such as:

- Matt Calabrese, ISO C++ committee member and longtime Boost contributor
- Andrew Sutton, ISO C++ committee member, architect of Concepts Lite, Boost.Graph maintainer
- Mathias Gaunard, ISO C++ committee member, Boost.SIMD and NT2 developer
- Louis Dionne, Boost.Hana developer
- Thomas Heller, Boost.Phoenix maintainer, HPX developer
- Tim Blechmann, Boost.Atomic maintainer, Boost.Lockfree and Boost.Heap developer
- Mario Mulansky, Boost.Odeint developer
- Jeroen Habraken, Boost.Spirit maintainer, Boost.Coerce author

Without the Boost SoC program, these individuals may have not had the opportunities that led to their continued involvement in the Boost community.

3.) 2016 Program Plan

Administration Team

- Niall Douglas
- Bryce Adelstein Lebach

Proposal Solicitation

Instead of openly accepting applications, we have **solicited** detailed proposals from “**veteran**” **students who have previously worked on a Boost SoC project** for this year. It was our intention to craft a portfolio of projects that had a high chance of successful completion and minimized the investment risk for Boost. We already know the capabilities of the students in question and the students already have a connection with the community and their mentors, so we do not have to be as concerned with aptitude testing and community bonding. All the solicited proposals follow [the Google SoC proposal guidelines](#), providing background information about the project, a detailed week-by-week project plan and a clearly defined set of deliverables.

Three projects decided to submit proposals. For more information on the three proposals, see Section 5.

Timeline

We will use a timeline based on [based on Google’s SoC timeline](#):

May 23 rd	Students begin coding
June 27 th	Midterm report due
August 15 th	Beginning of final week (documentation, code cleanup)
August 23 rd	Students finish coding
August 29 th	Final report due

All reports will be submitted to the Boost SoC administration team, the Boost Steering Committee, and Software Conservancy.

Reporting Requirements

The Boost administration team will check in weekly with each project to get an informal status update and to learn of any issues or roadblocks affecting the project.

The **midterm report** should describe the progress that has been made thus far on the project. An adjusted week-by-week project plan for the second half of the summer should be included.

The **final report** should summarize the deliverables of the project, provide a self-evaluation the success of the project and discuss plans for submission to Boost.

The students will write both reports, and the mentors will review the reports and sign off on them before they are submitted upstream.

4.) Proposed 2016 Budget

Student stipend:		\$5500 USD
Mentor stipend:		\$1000 USD
Total per project:		\$6500 USD
	2 projects @ \$6500 USD	
TOTAL:	1 project @ \$5500 USD	\$18000 USD

Google's SoC program pays students a stipend of \$5500 USD, for approximately three months of work. This works out to:

- \$5500 USD / 3 months = **\$1833.33 USD per month**
- \$1833.33 USD / (40 hours/week * 4 weeks/month) = **\$11.46 USD per hour**

We propose using this rate for a Boost-funded SoC. This rate is comparable to **the typical salary of a US undergraduate student** participating in a **work-study program** or **research internship at their university** (private sector internships for software engineers tend to pay more). For reference, the US federal minimum wage is \$7.25 USD an hour.

We propose a mentor stipend of \$1000 USD, which is the amount used for the Boost-funded extension of Louis Dionne's SoC project.

We are currently exploring opportunities for corporate sponsorship of this program, so it is possible that the total cost to Boost will be reduced by donations.

5.) 2016 Project Proposals

- Boost.Document
 - anuragxel.github.io/boost-document
 - **Proposal**
 - Student: Anurag Ghosh (Boost GSoC 2015 participant)
 - Mentor: Antony Polukhin
 - Library Status: No Boost review
- Boost.Compute
 - boostorg.github.io/compute
 - **Proposal**
 - Student: Jakub Szuppe (Boost GSoC 2015 participant)
 - Mentor: Kyle Lutz
 - Library Status: Accepted
- Boost.HTTP
 - boostsoc14.github.io/boost.http
 - **Proposal**
 - Student: Vinícius dos Santos Oliveira (Boost GSoC 2014 participant)
 - Mentor: Bjorn Reese
 - Library Status: 1 Boost review (not accepted)