IETF Automatic Schedule Builder RFP

2019-09-13

IETF Executive Director
Exec-director@ietf.org

Overview

The IETF Administration LLC (IETF LLC) is soliciting proposals ("Proposals") to build an Automatic Scheduler for IETF and IRTF.

Timeline

13 Sep: RFP Issued
20 Sep: Questions and Inquiries deadline
23 Sep: Answers to questions issued, RFP Addenda and Update issued
30 Sep: Proposals due
07 Oct: Selection made, negotiations begin
18 Oct: Contract execution
28 Oct: Work begins

Specifications

This is the process for the Request for Bids:

1. The Statement of Work (SOW) is attached.

2. Any questions about the Work must be submitted by 20 September 2019. A response to all parties shall be provided by 23 September 2019. The response will include the questions asked and the answers, but will not identify the company asking the question.

3. Bids are due by 30 September 2019. The bid must provide a not-to-exceed price, the expected start date, the expected completion date, any assumptions, and a description of any dependencies that might cause delays in the schedule.

4. The IETF LLC will discuss the Bids and may ask questions by email and/or conference call.

5. Once the answers are received a decision will be made to select the bidder to perform the work and a Work Order will be prepared for execution. We anticipate an award on or before 07 October 2019.

6. This is the Bid format:
    a.   Executive Summary

    b.   Project Approach & Plan

    c.   Schedule - When the work will begin and end, as well as dependencies and other milestones.

    d.   Test Plan

    e.   Cost & Payment Schedule

    f.   Warranty & Late Delivery Consequence

    g.   Technical Support & Maintenance

    h.   Miscellaneous

7. Instructions for IETF Software Development Contractors will apply. See
    https://trac.tools.ietf.org/tools/ietfdb/wiki/ContractorInstructions?version=26

Please reply with questions, if any, and a bid if you are interested in pursuing this opportunity to ietf-rfps@ietf.org.

Thanks in advance.

Portia Wenze-Danley

Automatic Schedule Builder: Statement of Work

About the IETF

The Internet Engineering Task Force ([IETF](http://www.ietf.org)[1]) is the premier Internet standards body, developing open standards through open processes. The IETF is a large open international community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet. The technical work of the IETF is done in [Working Groups](https://www.ietf.org/how/wgs/)[2] with much of the work handled via [mailing lists](https://www.ietf.org/how/lists/)[3]. In addition to IETF [meetings](https://www.ietf.org/how/meetings/)[4] held three times per year, individual Working Groups may hold interim meetings to advance their work.

The IETF Working Groups are organized into Areas and managed by Area Directors, who form the Internet Engineering Steering Group ([IESG](https://www.ietf.org/about/groups/iesg/)[5]). The IESG is responsible for the IETF standards process, including final approval of specifications as Internet Standards. IETF Areas include Applications and Real-Time (ART), Internet (INT), Operations and Management (OPS), Routing (RTG), Security (SEC), and Transport (TSV). The IETF Chair is the Area Director for the General Area (GEN) and chairs the IESG.

About the IRTF

The Internet Research Task Force ([IRTF](https://irtf.org/)[6]) focuses on longer term research issues related to the Internet while the IETF focuses on the shorter term issues of engineering and standards making.

The IRTF is comprised of a number of focused and long-term [Research Groups](https://irtf.org/groups)[7]. These groups work on topics related to Internet protocols, applications, architecture and technology. Research groups have the

---

1 http://www.ietf.org

2 https://www.ietf.org/how/wgs/

3 https://www.ietf.org/how/lists/

4 https://www.ietf.org/how/meetings/

5 https://www.ietf.org/about/groups/iesg/

6 https://irtf.org/

7 https://irtf.org/groups

stable long-term membership needed to promote the development of research collaboration and teamwork in exploring research issues.

Background

There are roughly 130 Working Groups and Research Groups focusing on different topics. Three times a year, around 100 of these working groups meet in person for a week. The IETF Secretariat develops a schedule[8] for these groups to meet in, ensuring the groups have rooms of appropriate size, the right resources, and with appropriate meeting duration. The schedule development process minimizes conflicts between the meeting groups.

The IETF Datatracker is a software tool used by the IETF and IRTF to facilitate the ongoing development of their work. It keeps track of documents developed by groups, allows groups to request sessions at future meetings, and displays the schedule developed for each meeting.

The Datatracker is written in Python using the Django framework. It contains a meeting application focused on managing various types of meetings. The current implementation can be read using the Datatracker source browser[9]. The root model of the application is Meeting. There are currently two types of Meeting objects: 'ietf' and 'interim'. This project will focus exclusively on 'ietf' meetings.

Early in the lifecycle of an 'ietf' Meeting object, it is given a set of TimeSlot objects that correspond to a block of time in a physical room (represented as a Room object) that a meeting session may be scheduled. Before the meeting, Working Groups and Research Groups make meeting session requests, which are captured as Session objects associated with the meeting. These represent sessions that can be scheduled into a TimeSlot. Sessions have associated Constraints. For the last few years, the constraints used were those listing a set of other groups that this group must not, should not, and prefers not to conflict with (known as first, second, and third priority conflicts, respectively). They also list a set of Person objects representing the people that must be present at the session, and resources required in the room for the session. The IETF Secretariat also manually maintains a set of constraints. Some working groups have long-standing requests to only meet at a certain time, or to meet before other groups from the same Area. Groups are also able to convey constraint requests in informal text with their Session request. Variations of "Please don't schedule us on Friday" are common[10]. For IETF 106, the first, second, and third order conflicts have been relabeled as "Chair Conflict Conflict", "Technology Overlap Conflict", and "Key Participant Conflict". We expect that these will become new Constraint objects after IETF 106, returning the first, second, and third priority conflicts to their original semantic.

---

8 https://datatracker.ietf.org/meeting/105/agenda

9 https://trac.tools.ietf.org/tools/ietfdb/browser/trunk#ietf/meeting

10 https://datatracker.ietf.org/meeting/105/requests

The application allows for the creation of multiple alternative Schedules. Schedules associate Sessions with TimeSlots. Several weeks before an 'ietf' meeting, the IETF Secretariat manually prepares one or more Schedules placing all working group and research group Session objects in TimeSlots. This is done with a drag-and-drop graphical editing view, which highlights conflicts between groups meeting at the same time. The IETF Secretariat takes their best proposed Schedule into a review with the IESG (called the "conflict resolution call") using the drag-and-drop editing view to minimize conflicts in a way that will likely produce the most effective meeting.

There is also work underway to improve the meeting application generally, and specifically to improve some terms in the models. This project is unlikely to encounter code-merge conflicts with that work, but it would be useful to be aware of that project's goals[11].

Earlier work to provide this functionality was abandoned before it concluded. It can be viewed here[12] and here[13]. The design for this project is *not* constrained to follow the paths that were explored by that earlier work.

Objectives

The goal of this project is to automatically produce one or more possible Schedules for the IETF Secretariat. These Schedules will ensure that the resources required exist in the Room into which each Session is scheduled. The Schedules will also minimize the declared conflicts between groups scheduled at the same time. The relative weights of the conflicts must be configurable, with the initial configuration prioritizing 1st priority conflicts higher than 2nd priority conflicts, and 2nd priority conflicts higher than 3rd priority, and 3rd priority conflicts higher than the people who must be present. (Note the relabelling of these constraints for IETF 106. This project will be agnostic to that, simply working with the Conflict types, with the initial configuration listed above. When the new Conflict types are added, the configuration this project provides will allow their relative priorities to be configured).

This project will also provide a mechanism for the IETF Secretariat to capture the constraints mentioned above that are not yet modeled, such as "SAAG always meets in the first afternoon session on Thursday". These constraints will initially have a priority configuration stronger than 1st priority conflicts.

Each produced Schedule should show a computed value that indicates the severity of the set of remaining conflicts.

---

11 https://www.ietf.org/documents/258/IETF_MeetingApp_RFP_02-28-19.pdf

12 https://trac.tools.ietf.org/tools/ietfdb/browser/trunk/ietf/meeting/placement.py?rev=12923

13 https://trac.tools.ietf.org/tools/ietfdb/browser/trunk/ietf/meeting/management/commands/autoplace.py?rev=12923

Deliverables

- A mechanism to capture the constraints the IETF Secretariat currently enforces by hand.
- GUI for initiating and presenting the results of automatic Schedule creation.
- An algorithm for finding a set of Schedules with low (optimally lowest) severity of the set of remaining conflicts scores.

All code deliverables are required to include test-suite covering all functionality. Developers will follow the [Contractor Instructions](https://trac.tools.ietf.org/tools/ietfdb/wiki/ContractorInstructions)[14].

---

[14] https://trac.tools.ietf.org/tools/ietfdb/wiki/ContractorInstructions

I E T F®