IETF ADMINISTRATION LLC

Request for Proposals

Datatracker Meeting Application Improvements

Date of Issuance: February 28, 2019
Proposal Submission Deadline: March 15, 2019, 5:00 P.M. ET

IETF LLC Request for Proposals

Datatracker Meeting Application Improvements

The IETF Administration LLC (IETF LLC) is soliciting proposals ("Proposals") to improve parts of the IETF Datatracker.

Proposals from any commercial or non-commercial vendor are welcome.

Timeline
28 Feb: RFP Issued
06 Mar: Questions and Inquiries deadline
08 Mar: Answers to questions issued, RFP Addenda and Update issued
15 Mar: Proposals due
29 Mar: Selection made, negotiations begin
02 Apr: Contract execution
08 Apr: Work begins

This is the process for the Request for Bids:

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

2. Any questions about the Work must be submitted by 6 March 2019. A response to all parties shall be provided by 8 March 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 15 March 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 29 March 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

8. Additionally, bidders for the Datatracker Meeting Application Improvements also have the option to be considered as a qualified software vendor for Indefinite Delivery/Indefinite Quantity (IDIQ) Master Services Agreement (MSA) purposes. Please indicate your interest. IDIQ software vendors will be eligible for future software development projects to improve and expand to the existing IETF toolset, which has been substantially developed in Python using the Django framework.

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

Datatracker Meeting Application Improvements: Statement of Work

Overview

Eight areas of improvement to the parts of the IETF Datatracker that assist with building and displaying the IETF meeting agendas have been identified. Implementing these improvements will require some redesign and extension of the models in the meeting application and significant work on the views, particularly the primary schedule building view. The project will include minor yet pervasive changes to the conceptual models used by the application. The contractor shall work with the Tools Team on each deliverable by engaging in design change discussions, helping to identify and prioritize areas of highest implementation risk, prototyping changes for review with the IETF Secretariat staff as necessary, and completing the implementation of each deliverable. It is expected that the contractor will frequently commit deployable units working towards the deliverables. These may be part of several production releases over the duration of the project. The contractor shall provide automated tests to exercise the new functionality in all changed and added code. In particular, javascript created or modified as part of this project will come with automated tests (the current automated test coverage of the project's javascript is poor). The contractor shall follow the instructions at https://trac.tools.ietf.org/tools/ietfdb/wiki/ContractorInstructions?version=26. This document refers to several existing datatracker tickets as #nnnn. These can be found at https://trac.tools.ietf.org/tools/ietfdb/ticket/nnnn.

List of Deliverables

Improve the underlying application models

Improve the view for graphically building a schedule

Improve the ability to manage the set of possible schedules for a meeting

Improve setting up a new meeting

Improve the ability to schedule sessions for things that are not working groups or research groups

Improve the creation and management of session requests

Improve display of meeting statistics

Improve the view of the official schedule for a meeting

Background

This section provides a high-level overview of parts of the current meeting application models and functionality. The current implementation of the application can be found at https://trac.tools.ietf.org/tools/ietfdb/browser/trunk#ietf/meeting. The root model of the application is Meeting. There are currently two types of meeting objects: 'ietf' and 'interim'. This project will focus primarily on 'ietf' meetings, but the possible application of the work to 'interim' meetings should be kept in mind during implementation. Early in the lifecycle of a Meeting object, it is given a set of TimeSlot objects that correspond to times in a physical room that a meeting session may be scheduled. Currently, the start and stop times of TimeSlots line up across all rooms. A Meeting will have multiple associated Session objects. These represent a meeting (or an assigned activity, like registration) that can be scheduled into a TimeSlot. A Meeting can have one or more (and typically has many) Schedule objects. Schedules associate Sessions with TimeSlots. Near the beginning of its lifecycle, a Meeting gets an "empty" schedule. This is empty only in the sense that it has no associations for working group or research group sessions. It will have sessions corresponding to the registration activities and breaks populated, currently by copying these forward from previous meetings, and manually adjusting them as needed. This "empty" schedule is copied to create new Schedule objects, which copies the break and registration sessions forward. Several weeks before an 'ietf' meeting, the IETF Secretariat prepares one or more Schedules placing all working and research group Session objects in TimeSlots. This is done with a drag-and-drop graphical editing view, which highlights self-declared 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 optimize conflicts in a way that will likely produce the most effective meeting. The result is declared the "official" schedule (or agenda) for a meeting, at which point it appears on the community's view of the meeting agenda (/meeting/agenda or /meeting/<num>/agenda). Changes to the official schedule may continue to occur up to or during the meeting. Changes after a schedule becomes official are treated differently than changes made before that point. For instance, artifacts are left in the original TimeSlot if a session is Canceled or moved to another TimeSlot. Note that placing these artifacts currently often require manual manipulation of the meeting data. In addition to the previously mentioned break and agenda session types, the IETF Secretariat also manually schedules sessions for tutorials, lunches, and other non-working-or-research group sessions, using a different interface than the above-mentioned drag-and-drop tool. These may or may not appear on the public agenda.

Detailed Description of Deliverables

Improve the underlying application models

The current code has a mix of the same things named “schedule” and “agenda”. The code should be changed to use “schedule” wherever reasonable. Note that Sessions point to an agenda document – this use of agenda in the code will remain unchanged. The Meeting “agenda” variable should change to “schedule”, but the views will still speak in terms of agenda (at, for instance, /meeting/agenda).

Session objects currently have a “scheduled” DateTimeField. This will be replaced with a relation to a new event class “SchedulingEvent” that captures all scheduling status transitions for a session.

There is dissonance in the code and views around a Session. The original conceptual model was that a Session was a working group or research group session, but that has been extended over time to include things the interface labels “non-session sessions”. The code, views, and data (particularly TimeSlotTypeName) will be adjusted to remove the idea of a “non-session session” and refer to those things by a useful descriptive name. Examples of “non-session sessions” in recent meetings include lunches, private meetings, tutorials, workshop meetings (ANRW), and Area Director office hours. Not all sessions may appear on the public official agenda. Care must be taken when deciding which sessions are publically shown.

Improve the view for graphically building a schedule

Performance Improvements

The current implementation of the graphical drag-drop schedule editor makes several hundred requests for information about individual sessions, groups, their needed resources, and their conflicts. This will be redesigned to make a single combined request (or perhaps a much smaller number of combined requests) for the initial load of the view.

View and Template improvements

The schedule editor view is based on an earlier and now abandoned base view framework. It has custom notions of changing whether menubars appear that are no longer relevant. The view will be updated to use the site’s primary base template (which is based on bootstrap) and the no longer relevant menubar manipulation code will be removed. See ticket #1387.

If you are not the owner of a schedule being viewed, there is a large box presented telling you must save the schedule to edit it. If you are the owner, there is a large box offering “Save as”. These boxes take up too much screen space, and will be redesigned to be less intrusive. See ticket #1429.

The current view offers the ability to add whitespace between days using small handles near the header for each day. This functionality has not turned out to be useful, and will be removed.

The ability to hide a room or a day (a row or set of columns on the displayed grid) has not seen use, and will be removed.

The HTML produced for the schedule editor view is currently invalid. This project will ensure that the view passes HTML validator checks.

The javascript in the current implementation sends a large amount of debugging information to the browser console. Most of this will be removed. Particularly useful debugging information points may be preserved, but will be guarded with site-wide debug level settings.

View functionality improvements

The project will make Area Director (AD) conflicts visible on the editing grid. An AD conflict is when two or more groups with the same responsible Area Director are scheduled at the same time.

The project will make "must be present" conflicts visible on the grid page. These must not be presented as intrusively as group conflicts and AD conflicts. Currently, the quality of the self-provided "must be present" data is suspect, and the IETF Secretariat is not comfortable evaluating the severity of the conflict, so for the near future, the IETF Secretariat will likely ignore this conflict information. Thus, again, it should not be presented as intrusively as other conflict types. We anticipate improvements to the quality of this data in the future.

While building a proposed schedule, the IETF Secretariat often finds swapping all the current scheduled groups on one day with those on another. Currently, this is an arduous and error prone task involving moving all the sessions for one day back into the unscheduled state, moving the sessions for the second day into the first day, and then moving the sessions that had been in the first day from the unscheduled state into the second day. This project will make it possible to exchange days with a simpler operation, preferably a single drag and drop.

The project will make groups that are scheduled in timeslots with a different length than their requested time visually distinct. See ticket #1428.

The project will make it possible to grey out sessions by area whether or not the user owns the agenda being viewed. See ticket #1584,

If a session is moved (rescheduled – assigned to a different TimeSlot) after the agenda is published as official, a new Session object will be created and associated with the original timeslot. That Session will have a new status of "rescheduled"; see SessionStatusName. This will display distinctly on the agenda view letting users know into which timeslot the session has been rescheduled. Note that this applies to both sessions that appear on the publically visible agenda and those that are reserved for private role-based views.

Improve the ability to manage the set of possible schedules for a meeting

The typical workflow for the IETF Secretariat is to copy a schedule and make a small number of changes. The project will add the ability to track those changes. This might be implemented by adding a history to a given schedule. If possible, a description of the difference between two schedules should be provided.

The project will add a Notes field to Schedule, allowing the IETF Secretariat to capture a description of the changes to that Schedule.

The project will make the page at /meeting/nnn/agendas/list more intuitive to use. One possible improvement is grouping agendas by the visible and public attributes. The edit button currently takes the user to a place to change the name of the schedule, possibly delete it, or change other properties instead of the place to edit the schedule. This will be adjusted to be more self-documenting. This page will show the new Notes field for each schedule.

Improve setting up a new meeting

The project will replace the current notion of an "empty" schedule with a base schedule object that all schedules for a meeting inherit. (Currently changes to the "empty" schedule are not propagated to any copies that have been made of that schedule). Sessions that are expected to be scheduled in the same TimeSlot across all proposed Schedules (such as breaks and registration) will be scheduled in the base schedule object. Changes to the base schedule will be immediately reflected in the view of any schedule. Sessions scheduled in the base schedule will be made visually distinct when viewing pages that show session schedules graphically, such as /meeting/nnn/room-view.

Improve the ability to schedule sessions for things that are not working groups or research groups

Improve the flexibility of the TimeSlot grid

Rooms that are used for sessions other than working or research groups should be allowed to have timeslots with different start and stop times, rather than adhering to the current convention which requires TimeSlots to start and stop at the same time across all rooms. It is currently expected that all of the rooms for working group or research group meetings would have TimeSlots that start and stop at the same time across rooms. Thus, the current drag-and-drop scheduling page does not currently need to be changed to allow for rooms with varying TimeSlot start and stop times. However, care should be taken with the code to not make it harder than it currently is to move to a convention where the working and research group sessions could be scheduled across rooms with different TimeSlot regimes.

The /meeting/nnn/timeslots/edit page will need to be adjusted to accommodate rooms with different TimeSlot regimes. One possible approach would be to group rooms sharing a regime vertically.

Allow graphical scheduling of all Sessions

While the current ability to focus on graphically scheduling only working and research group sessions must be preserved, the project will add graphical scheduling (using different views) for other types of sessions. Simply reusing the graphical editing code in another view is one possibility, but the conflict display mechanics are not needed. An alternative would be to add a simpler drag-drop scheduling mechanic to /meeting/nnn/room-view, or a non drag-drop mechanic to pages like /meeting/nnn/agenda/by-type, or /meetings/nnn/agenda/by-room.

Ensure private meetings don't show on public agendas

All modifications to facilitate graphical scheduling of non working group or research group sessions should take into account whether the Session being scheduled is intended to show on the public agenda. It should be easy to control which Session types are intended to show publically. The project will also make it easy to graphically view private sessions by type, perhaps by reusing the customized agenda view when logged in as a person with an appropriate role to see a private meeting.

Improve the creation and management of session requests

Currently, the group chairs requesting sessions provide the group conflict lists. The datatracker makes it easy to copy the conflict lists used at the last meeting forward. The long-term result has been that several groups have poorly curated conflict lists. At IETF 103, there were three groups that listed themselves as a first order conflict. This project will improve the session request form validation to disallow listing a group as conflicting with itself (at any level of priority), and to disallow listing groups or BoFs that have been closed. Such conflicts will be automatically removed when copying a previous meeting's conflict list forward.

Improve the view of the official schedule for a meeting

Improve the custom week-view

The agenda view at /meeting/nnn/agenda includes a javascript driven graphical display of meetings selected by the user. This project will enhance that view.

The project will make it possible to select and de-select individual non working or research group events, such as the codesprint, the systers lunch, and other non-session sessions.

The project will add the ability to select "All BoFs", "All Newcomer", and "All Running Code" sessions, and allow selecting or de-selecting individual sessions under those categories. See ticket #2314.

The project will modify the week view custom urls to use a query string rather than a fragment list. These changes will also be made to the views that present customized calendar subscriptions.

The project will make it possible to store the customization selected (what appears in the new query string) as a cookie in the user's profile for retrieval at future meetings. See ticket #1991.

Improve calendar subscriptions

The project will add an important date ics view similar to the existing view at /meeting/important-dates.

The project will reduce the granularity of how IETF meetings are represented in the current "Upcoming meetings" ics view. Instead of including all of the sessions for that meeting, the calendar should have one time blocked out for the entire meeting, and show important dates. It would also be nice to allow filtering upcoming meetings to those with Sessions scheduled for a given set of groups (reusing the new query-string from the customized week view would be a reasonable approach). See tickets #2605 and #2216.