

## Automating the Request Process for the Deep Space Network

Mark Johnston, James Mason, Gigi Lucena, Paul Christman, Alysia Wong, David Watson, Kade Peich

*Jet Propulsion Laboratory/California Institute of Technology, 4800 Oak Grove Drive, Pasadena, California 91109*

### Abstract

NASA's Deep Space Network (DSN) is responsible for tracking and communications with over 50 missions using 14 large antennas across three global sites. The scheduling process for DSN is intricate, requiring extensive manual input and negotiation among multiple stakeholders. To improve efficiency and reduce delays, the DSN Intelligent Virtual Assistant (DIVA) has been developed to provide assistance to DSN schedulers in multiple ways. DIVA is designed to automate scheduling request generation, and change proposal monitoring and response handling. This paper describes the architecture, design, and benefits of DIVA, along with opportunities for future enhancements.

**Keywords:** Deep Space Network; scheduling;

### 1. Introduction

NASA's Deep Space Network (DSN) serves as the primary communication infrastructure for interplanetary missions, providing critical tracking, telemetry, and command support for spacecraft operating beyond Earth orbit. Comprised of 14 large antennas distributed across three global facilities — located in Goldstone, California; Madrid, Spain; and Canberra, Australia (Figure 1) — the DSN ensures continuous communication coverage through its global reach and scheduling. Currently, the network supports about 50 active spacecraft, each with varying and often complex communication requirements.

The scheduling process for the DSN is inherently complex (see, e.g., [2–5]). It is conducted on a weekly basis, with preparations beginning more than six months in advance to accommodate mission-specific needs, antenna availability,



Fig. 1 The Deep Space Network complex at Canberra, Australia, one of three such complexes that make up the DSN[1]

and network constraints. Each scheduling week processing cycle requires mission schedulers to submit detailed request specifications that define the timing, duration, constraints, and dependencies of tracking events. These requests must be entered into the Service Scheduling Software (SSS), a task that can consume significant time and resources. For instance, it would typically take the Mars Odyssey mission scheduler approximately 45 minutes per week to input recurring request data, despite the repetitive nature of the mission's schedule. For many missions, requirements consist of a mix of repetitive and one-off requests, depending on phase of the mission and external events. For example, the Juno spacecraft is in an orbit around Jupiter, and every 33 days has a special set of activities associated with perijove, the closest approach to the planet.

Once initial requests are submitted and deconflicted, the DSN scheduling process moves into a negotiation phase wherein schedulers from different missions must coordinate changes via a proposal and counterproposal mechanism. This collaborative process, while necessary to accommodate shared resource constraint resolution, is susceptible to bottlenecks. At any given time, only one proposal can be active for a given week, and delays in responses from any affected party can halt further progress on changes to the schedule. This serialized nature of negotiation introduces latency into the process and demands constant vigilance from mission schedulers.

In addition to being time-intensive, the process imposes significant cognitive load on schedulers, who must track changes across multiple scheduling weeks simultaneously and maintain situational awareness of their mission's evolving priorities and constraints. The result is a system that is increasingly strained by the growing number of missions and the complexity of modern operations.

To address these challenges, we have developed the DSN Intelligent Virtual Assistant (DIVA), an intelligent automation tool designed to streamline routine aspects of the DSN scheduling process. DIVA reduces the manual burden on schedulers by automating the generation of tracking requests, monitoring scheduling changes, and responding to change proposals within predefined rule sets. By handling these tasks, DIVA enables mission schedulers to focus their efforts on more strategic decision-making and coordination, ultimately improving the responsiveness and efficiency of the DSN scheduling process.

In this paper, we present an overview of DIVA's functionality and architecture. We describe how DIVA integrates into the existing scheduling workflow, examine the approach to automating aspects of collaborative scheduling, and outline opportunities for future enhancements, including the application of intelligent decision-making techniques to further improve automation and adaptability.

## **2. Overview of the DSN Scheduling Process**

The Deep Space Network (DSN) operates as global infrastructure designed to maintain uninterrupted communications with spacecraft throughout and beyond the solar system. To ensure mission success and maximize scientific return, the network must carefully coordinate the use of its limited antenna resources across all supported missions. This coordination is achieved through an iterative scheduling process, governed by a structured weekly planning cycle and collaborative negotiation among users.

### *2.1. DSN Infrastructure and Mission Needs*

The DSN consists of three major facilities located approximately 120 degrees apart in longitude—Goldstone (USA), Madrid (Spain), and Canberra (Australia) — ensuring global visibility for deep space missions. Each complex houses multiple antennas of varying sizes (one 70-meter and multiple 34-meter dishes) capable of supporting different communication requirements. Missions span a wide range of capabilities and objectives, from planetary science (e.g., Mars rovers and orbiters) to heliophysics, astrophysics, and others. The DSN also supports ground-based science, including planetary radar and radio astronomy.

Each mission user has unique operational constraints and communication demands. These include specific data return windows, command uplink requirements, and constraints based on spacecraft visibility, antenna compatibility, geometry, spacecraft power availability, and other factors. The scheduling system must accommodate this diversity while maintaining fairness and efficiency across all users.

### *2.2. Scheduling Workflow: Request Submission and Approval*

The DSN scheduling process begins about six months prior to the target week, with the preparation and submission of user requests. These requests are formally defined inputs that describe each mission's required communication sessions for a given week. Key components of a request include:

- Start and end times (absolute or relative to events)
- Antenna constraints (e.g., specific antennas or regions)
- Track duration and frequency (e.g. daily, multiple per week, etc.)
- Setup and teardown times
- Splittability (whether a request can be satisfied by multiple tracks)
- Relationships to other requests (e.g., dependencies or sequencing, minimum or maximum gaps, etc.)

Requests must be interactively entered into the DSN's Service Scheduling Software (SSS), a centralized system that facilitates generation and evaluation of candidate schedules. Despite the repetitive nature of many missions' weekly schedules, this manual entry process is time-consuming and prone to typos or transcription errors.

After initial request submission, a series of automated and interactive processes are executed to generate a preliminary candidate schedule. This schedule is then refined through human negotiation.

### *2.3. Manual Effort in Proposal Negotiation*

Once a preliminary schedule is generated, a collaborative refinement phase begins. During this phase, users review conflicts or inefficiencies and propose adjustments to better meet mission requirements. The negotiation process follows a structured mechanism:

- 1) Proposals are initiated by one mission, suggesting changes to the schedule (e.g., moving a track, changing duration, altering antenna assignments).
- 2) Impacted users — those whose schedules would be affected — must review and either approve or counter the proposal.
- 3) Only one proposal can be active for a given week at any time. This serialized constraint is necessary to prevent confusing and conflicting edits.
- 4) The process repeats until all conflicts are resolved and a baseline schedule is reached.

This method places a heavy coordination burden on users, who must continuously monitor for new proposals, assess their impact, and respond in a timely manner. Because of the long duration pipeline of schedules, this typically applies to about 18 different weeks at one time. Delays at any stage can cascade throughout the process, increasing the risk of missed deadlines or suboptimal resource allocation.

### *2.4. Impact of Delays on Mission Operations*

Because of the serialized and interdependent nature of proposals, even small delays can have outsized effects. If a mission representative is unavailable or delayed in responding to a proposal, all subsequent proposals are blocked for that week. This can stall the schedule finalization process, jeopardizing the timely delivery of the baseline schedule to operations teams.

Moreover, the need to monitor multiple scheduling weeks in parallel — each potentially at a different stage in the negotiation process — adds to the cognitive and procedural complexity faced by users. Schedulers must balance competing priorities across multiple time horizons, manage shifting dependencies, and ensure continuity of communication support for their missions.

These challenges were part of the motivation for improved tooling to reduce manual burdens, automate routine tasks, and enhance responsiveness in the scheduling process. The DSN Intelligent Virtual Assistant (DIVA) was designed specifically to address these issues, as detailed in the following sections.

## **3. The DSN Intelligent Virtual Assistant (DIVA)**

The DSN Intelligent Virtual Assistant (DIVA) is an automation framework developed to assist in the complex and repetitive tasks associated with DSN scheduling. DIVA is designed to reduce the manual workload of mission schedulers, accelerate the resolution of scheduling proposals, and ultimately to reduce the cost of DSN scheduling and enable more efficient use of DSN resources. By automating routine processes and integrating seamlessly into existing workflows, DIVA empowers users to focus on mission-specific strategy rather than administrative overhead.

### 3.1. Origins and Motivation for DIVA

DIVA was originally conceived not only as a productivity enhancer for experienced DSN users, but as an enabling tool for missions that do not have the resources to maintain full-time scheduling staff. Small missions, particularly those with limited funding or operated by academic partners, may lack the in-house expertise or staffing levels required to engage fully in the DSN scheduling process. These missions face significant barriers to entry: the SSS interface is complex, the negotiation process requires continuous engagement, and the timing constraints for proposal responses are strict.

Recognizing these constraints, a project was kicked off in mid-2020 to explore the feasibility of a supplementary scheduling support tool that supplements SSS. This project was called ROBOSS, for “ROBot Scheduling Services”. The project was conducted as an agile development effort jointly between the JPL scheduling team (Multimission Resource Scheduling Services, or MRSS) and the DSN scheduling software development personnel. The project aimed to demonstrate feasibility and return on investment by building up a set of requirements, process documentation and improvement plans, and a software prototype to meet the agreed upon highest priority capabilities. The features that were demonstrated at the end of the ROBOSS project included:

- A compact mission-oriented dashboard showing conflicts, negotiation status, schedule changes, and quick links to appropriate information systems.
- A means to enter repetitive requirements in ROBOSS, then automatically upload to SSS on a weekly basis.
- A notification subscription process that sent alerts to Slack and over email of new conflicts or proposals ready for review.
- Automated conflict resolution based on a beam search for resolving antenna conflicts by adjusting track start times within requirements limits, recommending up to 3 different options if no solution is found.

By encapsulating best practices and offering configurable automation rules, ROBOSS would allow low-cost missions to participate in DSN operations effectively and reliably, without needing to maintain deep scheduling domain expertise, or dedicate personnel to full-time scheduling.

As a follow on to ROBOSS, DIVA was developed as a means to fully flesh out the concepts that ROBOSS pioneered. It became clear that the benefits provided by DIVA would apply not just to small independent DSN users and teams, but

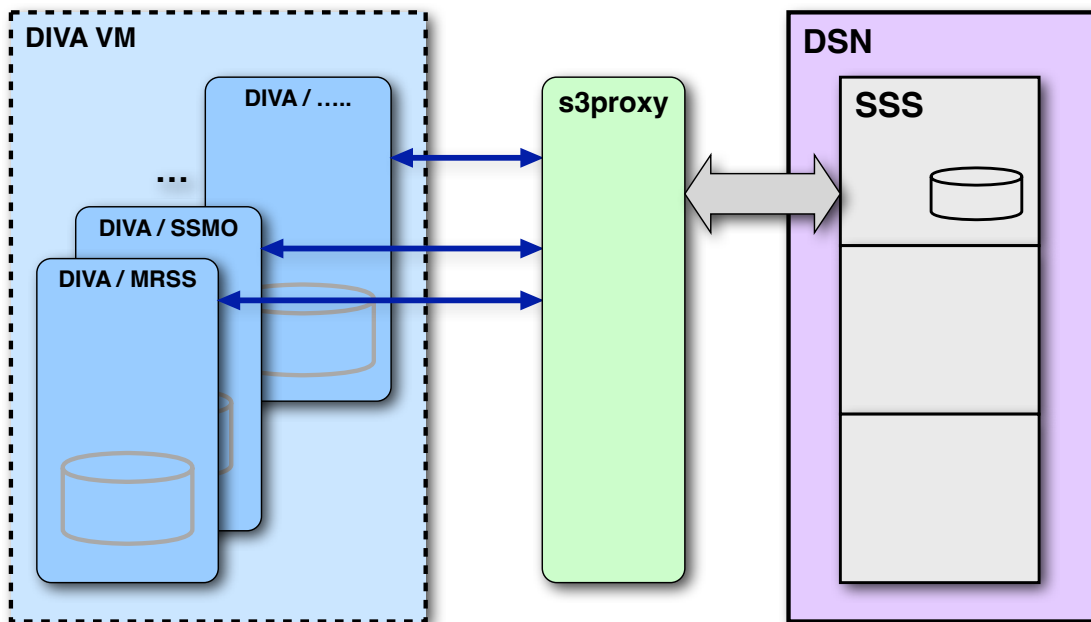


Fig. 2 DIVA architecture diagram. Each DIVA instance (web application and mongodb database) runs inside a DIVA VM hosted at JPL, and connects to S3Proxy for scheduling information. S3Proxy in turn mediates interaction with the formal DSN scheduling system, SSS.

to all of the scheduling teams to which mission scheduling representatives typically belong. There are six such teams, with the majority of DSN users supported by three of them: MRSS – based at JPL; Space Science Mission Operations (SSMO) – based at Goddard Space Flight Center; and “DSN Scheduling” – an internal DSN team responsible for scheduling maintenance and engineering activities within the DSN. Three other small teams represent one or two spacecraft each: one based in Berkeley for the Themis B and C Lunar orbiters; one at NOAA for the Deep Space Climate Observatory spacecraft; and one in Cambridge, MA, for the Chandra X-ray observatory.

### 3.2. Objectives and Design Philosophy

DIVA is guided by the principle of automation with oversight. Rather than seeking to replace human judgment in complex scheduling trade-offs, DIVA is designed to automate repetitive tasks, reduce the opportunity for human error, and surface actionable information to schedulers at the right time. Its main objectives include:

- **Reduce manual data entry** by generating routine requests from predefined templates
- **Decrease response latency** in the proposal process through automated evaluation and approvals for certain classes of changes
- **Notify users proactively** about important scheduling events, such as new proposals or upcoming deadlines
- **Lower the technical barrier to DSN participation**, enabling a wider range of users to contribute effectively to scheduling

Each scheduling team manages its own team assignments, such as primary and backup roles for each mission. And since a goal of DIVA is to decentralize configuration settings, the logical architecture is to provide each team with their own instance of DIVA, for them to manage as best fits their work processes. The result is illustrated in Figure 2, which shows multiple DIVA instances, each with its own database. Each instance is administered by its team.

DIVA operates as a layer above the existing SSS interface, interfacing with standard data products and workflows without requiring significant changes to DSN infrastructure. This approach minimizes the adoption cost for missions and ensures compatibility with existing scheduling practices.

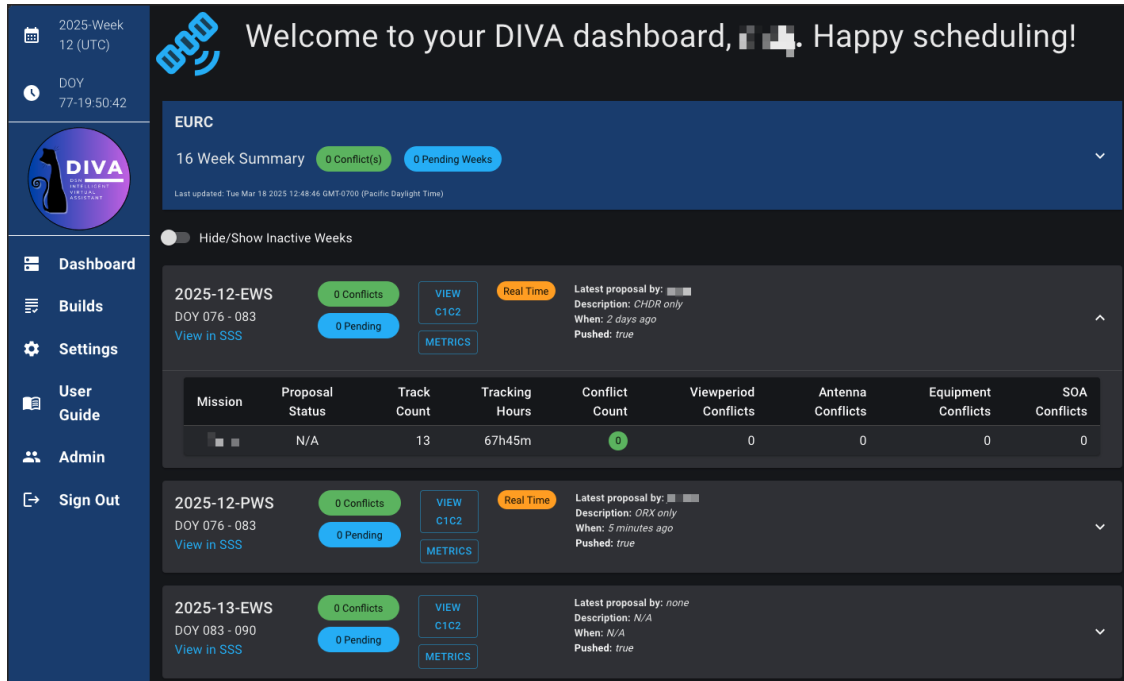
To minimize the impact of a high rate of queries to the operational SSS, a central “proxy” server was created to act as the single point of contact for any instance of DIVA. This server, designated *S3Proxy* in Figure 2, gathers and keeps in memory the latest status information from all active SSS schedules and workspaces. It provides a set of APIs used by DIVA to query for workspace status and changes. Because the *S3Proxy* data is kept in memory only, response times are fast, and the individual instances of DIVA can process the responses to handle display updates and notification generations.

### 3.3. Key Functionalities

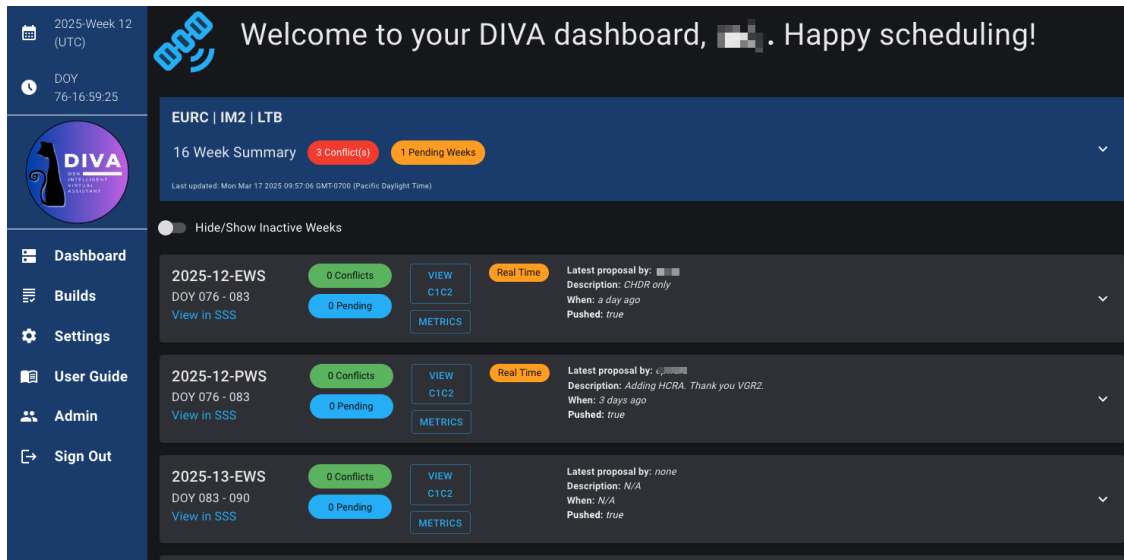
DIVA provides three major capabilities that address different aspects of the DSN scheduling process:

- **Automated Request Generation:** Instead of entering tracking requests manually each week, users define generalized request patterns called *builds*. These builds specify relative timing and constraints, which DIVA translates into concrete SSS-compatible request sets. This greatly reduces the time and effort required for recurring or templated request patterns.
- **Proposal Monitoring and Notifications:** DIVA monitors the state of each scheduling week for new proposals, responses required, and schedule changes. It notifies users when their attention is needed, reducing the chance of accidental delays in negotiation and allowing users to monitor multiple weeks concurrently with minimal effort.
- **Automated Proposal Responses:** For proposals that include routine or low-risk changes, such as adjusting setup times or track durations, DIVA is being engineered to automatically respond on behalf of the user. These responses are governed by user-defined rules, ensuring that automation aligns with mission policy while eliminating unnecessary delays.

Each of these capabilities is described in greater detail in Section 4. Together, they enable a more agile and resilient approach to DSN scheduling, supporting both expert users and those with limited scheduling capacity.



(a) The main DIVA dashboard, showing the first few weeks from realtime out through all weeks in negotiation and not yet baselined. Clicking on a week expands a section that has more details about specific missions, including total track count, tracking hours, and conflicts by type.



(b) DIVA dashboard showing a situation when there are conflicts affecting missions of interest, as well as pending change proposals awaiting action.

Fig. 3 DIVA dashboard views: the dashboard provides an instantly comprehensible overview of the DSN scheduling process from the perspective of one or more missions, highlighting items that require user action (such as resolving a scheduling conflict, or responding to a change proposal). Combined with proactive Slack notifications, the goal is to minimize latency in the process.

#### 4. DIVA Capabilities and Features

In this section we provide a brief tour of the DIVA user interface, highlighting capabilities that are useful to the scheduling system users.

Upon logging in, the user is greeted with a dashboard view such as that in Figure 3a. This includes summary information about each week that is currently in process, whether the current executing week, a week that has been baselined, or those that are still in negotiation. Each displayed week has highlights that display key indicators about the status of the week with respect to the missions of interest to that specific user: whether there are conflicts, change proposals pending, current proposed changes, and current metrics. Clicking on a week expands to show more detailed information (Figure 3b). There is also a link that will directly open the SSS application focused on the specified week's workspace. The dashboard automatically and seamlessly updates very frequently, so the view is very close to realtime. And because it focuses on the individual user's mission set, it consistently highlights actionable items that need attention. To further ensure that key information is always visible, the user can click a button to hide all "inactive" weeks, i.e. those without conflicts or pending change proposals.

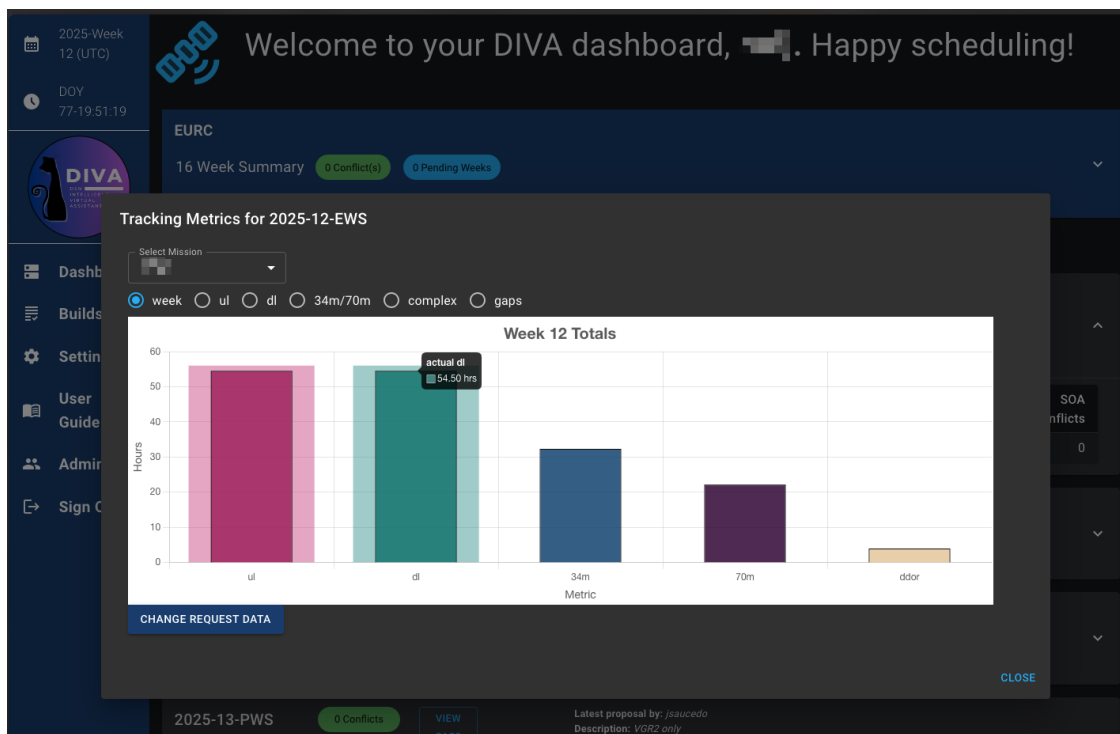


Fig. 4 DIVA metrics for one week, broken out into totals for uplink, downlink, antenna class (34- vs 70-meter), and DDOR

Figure 4 shows a view of the metrics instantly available for a mission. At a top level, the graphical view shows the number of hours scheduled for uplink, downlink, 34-meter, 70-meter, and in DDOR (Delta Differential One-Way Ranging, a key navigation input for trajectory determination). Selecting one of the radio buttons changes the view to a daily tally for uplink, downlink, 34/70-meter, DSCC (Goldstone, Canberra, or Madrid), and gaps. This allows a quick check that mission requirements for the week are in the expected ranges.

A major time-saving aspect of DIVA is provided by the "build" functionality as shown in Figure 5. This allows the user to specify a set of requirements that can be repeatedly used, week after week, and automatically specified to upload to SSS by the requirements input deadline for each week. A build can have multiple associated requests, each with a set of constraints and parameters. For example, a request would typically have:

- a distinctive name
- the service alias identifying which antenna and equipment options may be used
- a comment describing the intent or purpose of the request

- a constraining time range within the week, for example, “Monday”
- tracking duration parameters: nominal, min, max, and whether it can be split over multiple tracks, and whether the splits can overlap or must be separated
- priority and preference
- any constraining event intervals
- timing relationships relative to other requests in the week
- overrides to any default service alias parameters such as activity description, sequence identifier, or a filtered set of antennas (e.g. southern hemisphere only)
- special flags such as ‘MSPA allowed’ or schedule as a backup to a primary track

As many requests as desired can be assembled into a build and given a suggestive name. This is particularly useful for intervals of time when mission requirements are relatively uniform from one week to the next.

Once defined, builds can be associated with scheduling weeks out into the future. They may be exported as XML files that follow a standardized request specification used by DSN for uploading to SSS. While this XML file could be manually uploaded, the preferred approach is to have DIVA automatically create a new workspace in SSS, upload the requirements, and run the command to expand the requests into candidate tracks.

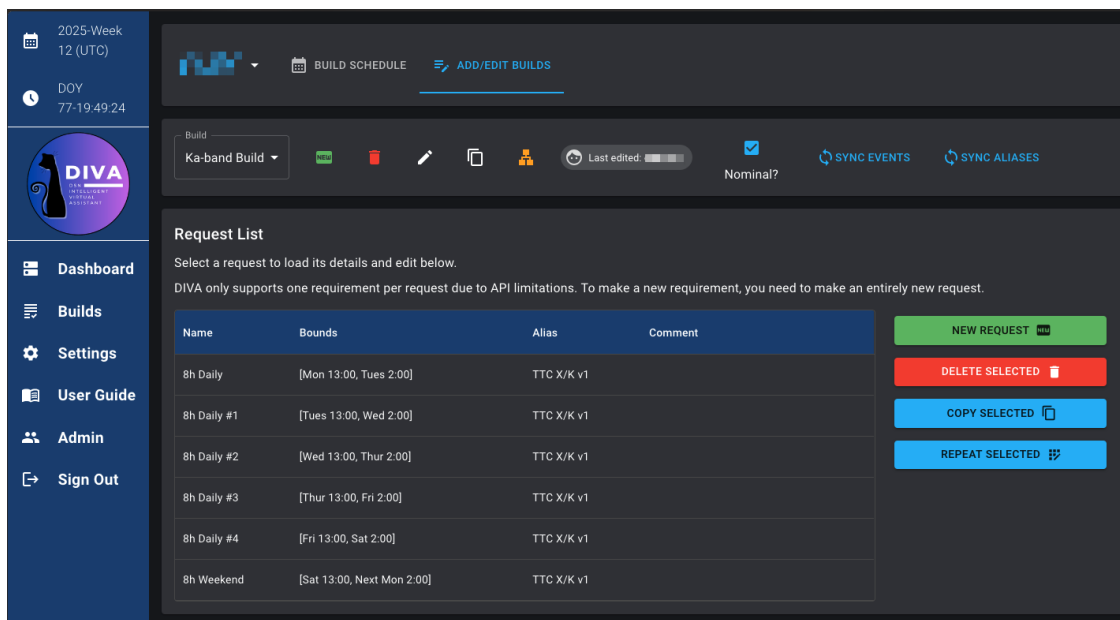


Fig. 5 DIVA build — a set of scheduling request parameters ready to submit

DIVA notifications can be configured by the user to trigger on specific situations as they occur. The rise in popularity of Slack as a team communication tool has essentially made obsolete other notification mechanisms such as email and text messages. DIVA Slack notifications can be subscribed by the individual user to suit their preferences, and include:

- new conflicts
- change proposals that need a response
- the submission of requirement builds
- a delay reminder
- a week has pushed to the Master

Notifications are not simple text notes but include salient information for the scheduler. For example, the conflict notification specifies the mission affected, the type of conflict, what workspace is affected, and include a link to click that instantly opens the SSS application on that workspace.

As noted above, each scheduling team has its own instance of DIVA, separately administered. Administration is very straightforward and is handled by designated team members who provide the following information:

- which users have access to the DIVA instance
- which missions are managed in the DIVA instance, i.e. assigned to the team
- which individual scheduler is prime for each mission
- upcoming week requirements build due dates

In addition, the admin section includes logs of significant activities, such as errors and autobuild runs.

The DIVA administration section also includes an overview of the status of all missions assigned to the team. This includes which weeks have pending change proposals for those missions, and which weeks have any of those missions in conflict.

The major new feature close to release is to have DIVA automatically concur to certain types of changes when they occur in change proposals. Normally, when one mission proposes a change that affects another mission, the affected mission scheduler has to log into SSS, navigate to the relevant workspace, check out the change, and verify that it is acceptable. Sometimes these changes are extensive and require the scheduler to consult with their project personnel to see if it will work; if not, they will have to go in and make a counter proposal, thus continuing an iterative process that eventually leads to concurrence from all parties. However, certain type of changes are much more straightforward and lend themselves to automatic concurrence. As an example, if a track is moved to a different time and/or antenna, and then moved back, SSS considers that a new version of the track and requires concurrence. In fact there is “no net change”, so this could be automatically concurred. Other cases include changes to setup and teardown only, or moving a track to another equivalent antenna (with equivalent services) but not changing the timing. A version of DIVA is in testing that can recognize these scenarios and, if the user has configured it to do so, automatically concur to the change in SSS, essentially acting as an agent on behalf of the scheduler. This has the potential to significantly streamline proposal handling since only one proposal can be active for a week at a time, so an incomplete proposal can block further necessary changes. It will allow the scheduler to focus on change proposals that really need attention.

## 5. Conclusions and Future Directions

DIVA has demonstrated how an auxiliary application used in conjunction with the main DSN scheduling tool, SSS, can provide benefit to schedulers and to teams engaged in the DSN scheduling process. Any improvements to process efficiency benefit all users who would otherwise find themselves waiting for bottlenecks to clear. While SSS is DSN mission critical and has to go through a rigorous development, testing, and release process for new capabilities to be added, DIVA has provided a complementary opportunity to be more agile and responsive to user suggestions. It can also be more readily customized to user and team preferences that would not be practical to incorporate directly into SSS.

The primary future direction for DIVA is to continue to identify and implement user- and team-driven improvements to make the scheduling process more efficient and streamlined. DIVA’s automated proposal responses operate similarly to task-focused AI agents: they interpret structured input, apply user-defined rules or policies, and take autonomous actions within a defined scope. While currently limited to certain scheduling change responses such as those noted in the previous section, this agent-like behavior offers a foundation for broader automation. In the future, DIVA could be extended to handle more complex negotiations, perform complex conflict resolution, or assist with cross-mission coordination by reasoning over multiple proposals and constraints simultaneously.

Acknowledgements: This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

## References

- [1] “Basics of Space Flight - Solar System Exploration: NASA Science,” <https://solarsystem.nasa.gov/basics/chapter18-1/>, Apr. 2021.
- [2] Johnston, M. D., Tran, D., Arroyo, B., Call, J., and Mercado, M., “Request-Driven Schedule Automation for the Deep Space Network,” *SpaceOps 2010*, Huntsville, AL, 2010.
- [3] Johnston, M. D., and Tran, D., “The DSN Scheduling Engine (DSE): Automated Scheduling of Activities for the NASA Deep Space Network,” *International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*, Sapporo, Japan, 2010.

- [4] Johnston, M. D., and Tran, D., "Automated Scheduling for NASA's Deep Space Network," *6th International Workshop on Planning and Scheduling in Space (IWPS)*, Darmstadt, Germany, 2011.
- [5] Carruth, J., Johnston, M. D., Coffman, A., Wallace, M., Arroyo, B., and Malhotra, S., "A Collaborative Scheduling Environment for NASA's Deep Space Network," *SpaceOps 2010*, Huntsville, AL, 2010.