Follow-up Queries on Octane API Behavior and Background Task Tracking

Dear Octane Support Team,

Thank you for your detailed responses and guidance on my previous queries. I would like to follow up with some clarifications and further questions based on your feedback:

# 1. Workspace Lookup by JIRA SPK

While trying to retrieve the Octane ID using the JIRA SPK, is it possible to achieve this without explicitly passing the space\_id and workspace\_id in the endpoint URL? Currently, we are iterating over different workspace IDs dynamically by updating the API URL per workspace. Please confirm if there is any supported approach to search across all workspaces without specifying them individually.

# 2. Tracking Test Result Upload Tasks

Earlier, we were using the Task API available through the Octane interactive client, which led to the 'Task not found' issue. After switching to the endpoint you suggested, we are now able to successfully track the task status. Thank you for your guidance.

# 3. Mismatch in Uploaded Task Counts

We are uploading test results via the /test-results API (XML upload). After ensuring that all upload tasks are completed successfully, we query the uploaded automated test runs using the following approach:

- Use the /automated\_runs API.  
- Pass the suite\_run\_id as a filter to retrieve only relevant test runs.  
- Set limit=1000 per API call.  
- After each call, increment the offset by the limit value to fetch subsequent batches of results.  
  
Despite properly handling pagination, we observe that the total number of automated test runs retrieved is lower than the number of test cases we uploaded. Could you please advise if there are any known factors, limitations, or additional parameters that could explain this mismatch?

# 4. Tracking Backlog Coverage Updates

We attempted linking JIRA stories to 7,424 uploaded test runs. We are updating 500 test runs per request (also tried 1000 runs per request) using the /automated\_runs update API. While the API responds quickly with success, we notice that the backlog coverage updates take approximately 7 to 14 minutes to fully reflect in Octane and JIRA.

Additionally, we explored setting backlog links directly within the test result XML during upload. However, we observed that this approach overwrites any previously linked JIRA stories instead of appending to the existing links.

We also have a scenario where we need to upload and link backlog coverage for 84,000 tests, and this number could potentially double in the future.

- Is there a way to link backlog coverage during upload itself without overwriting existing linked JIRAs?  
- If not, could you suggest an alternative recommended method to efficiently append backlog coverage during result upload, especially at this scale?

I appreciate your continued support in helping us optimize our integration with Octane.

Looking forward to your guidance on the above points.

Best regards,  
Dhatchanamoorthy  
[Your Team / Department]  
[Your Contact Information]