

Jira Query Language (JQL) Guide

Extracting Story Points by Stream, Epic, and User Story with Export Workflows

1. Overview & Key JQL Fields

Jira Query Language (JQL) allows you to filter and aggregate issues to track workload, capacity, and estimation metrics across various hierarchy levels. Below are the primary fields utilized for querying story point metrics:

Field Name	JQL Field Identifier	Description & Usage
Story Points	'Story Points' or 'Story Point Estimate'	Numerical estimation field assigned to user stories, tasks, or bugs.
Epic / Parent	'Epic Link', parent, or parentEpic	Associates child user stories to their parent Epic or Initiative.
Stream / Tribe	Stream, 'Component[s]', or 'Labels'	Custom field or component/label representing business or feature streams.
Issue Type	issuetype	Filters by work item hierarchy (e.g., Epic, Story, Task, Bug, Sub-task).
Status / Resolution	status, statusCategory	Tracks progress states (e.g., 'To Do', 'In Progress', 'Done').

IMPORTANT JQL CONSIDERATIONS

Custom Field Names: In many Jira instances, 'Story Points' or 'Stream' are custom fields. You can use exact field IDs if names conflict (e.g., cf[10002] > 0).

Aggregation Note: Standard JQL filters return matching issues. Total story point summation per stream/epic is performed via Jira Dashboards, Advanced Roadmaps, or

post-CSV export.

2. JQL Queries by Granularity Level

Level 1: Story Points per Stream / Workstream

Retrieve all stories, tasks, and bugs belonging to specific feature streams or business components:

All Active Items in Stream A:

```
project = "PROJ" AND Stream = "Stream A" AND "Story Points" EMPTY ORDER BY priority DESC
```

Completed Points in Stream B:

```
project = "PROJ" AND Component in ("Frontend", "API") AND statusCategory = Done AND "Story Points" > 0
```

Multi-Stream Summary:

```
project = "PROJ" AND Stream in ("Stream A", "Stream B", "Stream C") AND issuetype in (Story, Task)
```

Level 2: Story Points per Epic

Extract all child issues under specific Epics to evaluate Epic progress and sizing:

Points under Specific Epic:

```
"Epic Link" = PROJ-101 OR parent = PROJ-101 ORDER BY key ASC
```

All Open Stories Across Epics:

```
project = "PROJ" AND issuetype = Story AND "Epic Link" IS NOT EMPTY AND status != Done
```

Epics in Current Sprint:

```
project = "PROJ" AND sprint in openSprints() AND issuetype = Epic
```

Level 3: Story Points per User Story / Individual Work Item

Detailed query to audit story point allocations across individual user stories and sub-tasks:

Unestimated Stories (Needs Sizing):

```
project = "PROJ" AND issuetype = Story AND ("Story Points" IS EMPTY OR "Story Points" = 0) AND statusCategory != Done
```

High-Value Stories (> 8 Points):

```
project = "PROJ" AND issuetype = Story AND "Story Points" >= 8 ORDER BY "Story Points" DESC
```

Assignee Workload in Active Sprint:

```
project = "PROJ" AND sprint in openSprints() AND assignee = currentUser() AND "Story Points" IS NOT EMPTY
```

3. Sample CSV Extract & Data Format

When exporting JQL query results from Jira (Filters > Export > Export Excel / CSV), the output layout provides structured row items that can be summarized using Pivot Tables in Excel or Google Sheets:

Issue Key	Issue Type	Summary	Stream	Epic Link	Assignee	Status	Story Points
PROJ-101	Epic	Core Payment Gateway	Checkout	—	Sarah T.	In Progress	21
PROJ-102	Story	Integrate Stripe API	Checkout	PROJ-101	Alex M.	In Progress	5
PROJ-103	Story	Apple Pay Support	Checkout	PROJ-101	David K.	To Do	8
PROJ-104	Story	PCI Compliance Audit	Security	PROJ-105	Elena R.	Done	3
PROJ-105	Epic	Security & Hardening	Security	—	Elena R.	In Progress	13
PROJ-106	Story	User Auth Refactoring	Identity	PROJ-107	James P.	In Progress	5

4. Exporting and Summarizing Story Points

Step-by-Step CSV Export Instructions:

1. Navigate to Filters > Search Issues in Jira.
2. Paste your desired JQL query (e.g., project = "PROJ" AND issuetype in (Story, Epic)).
3. Click Switch to JQL (if in basic mode) and press Search.
4. Click the Export icon in the top right corner.
5. Select Export Excel CSV (my fields) or Export Excel CSV (all fields).
6. Open the downloaded CSV file in Microsoft Excel or Google Sheets.

Creating a Pivot Table Summary:

- Select all exported CSV data (Ctrl + A / Cmd + A).
- Go to Insert > Pivot Table.
- Set Rows: Add Stream and then Epic Link (or Epic Name).
- Set Values: Add Story Points and set summary function to SUM.
- Set Columns / Filters (Optional): Add Status or Assignee to break down points by completion state.

Useful JQL Pattern

Purpose	JQL
All stories	issuetype = Story
Stories in a project	project = ABC AND issuetype = Story
Stories for an Epic	"Epic Link" = ABC-123 AND issuetype = Story
Stories for several Epics	"Epic Link" IN (ABC-123, ABC-456) AND issuetype = Story
Stories in an Epic/parent hierarchy	parent = ABC-123 AND issuetype = Story
Stories assigned to a user	assignee = jsmith AND issuetype = Story
Stories in a stream	"Stream" = "Stream A" AND issuetype = Story
Stories in several streams	"Stream" IN ("Stream A", "Stream B") AND issuetype = Story
Stories in a sprint	sprint = "Sprint 25" AND issuetype = Story
Stories in a date range	project = ABC AND issuetype = Story AND created >= "2026-01-01" AND created < "2027-01-01"
Stories with story points	project = ABC AND issuetype = Story AND "Story Points" IS NOT EMPTY
Completed stories	project = ABC AND issuetype = Story AND statusCategory = Done

Stream-level Summary (Aggregation)

Stream	Epic	User Story	Assignee	Story Points
Stream A	Epic 1	ABC-101	User A	5
Stream A	Epic 1	ABC-102	User B	8
Stream A	Epic 2	ABC-103	User A	3
Stream B	Epic 3	ABC-104	User C	13

Epic-level Aggregation

Stream	Epic	Number of Stories	Total Story Points
Stream A	Epic 1	2	13
Stream A	Epic 2	1	3
Stream B	Epic 3	1	13

Stream-level Aggregation

Stream	Number of Epics	Number of Stories	Total Story Points
Stream A	2	3	16
Stream B	1	1	13