Issue event types

In this article

Issue event object common properties
added_to_project
assigned
automatic_base_change_failed
automatic_base_change_succeeded
base_ref_changed
closed
commented
committed
connected
convert_to_draft
converted_note_to_issue
converted_to_discussion
cross-referenced
demilestoned
deployed
deployment_environment_changed
disconnected
head_ref_deleted
head_ref_restored
head_ref_force_pushed
labeled
locked
mentioned
marked_as_duplicate
merged
milestoned
moved_columns_in_project
pinned
ready_for_review
referenced
removed_from_project
renamed
reopened
review_dismissed
review_requested
review_request_removed
reviewed
subscribed
transferred
unassigned
unlabeled
unlocked

unmarked_as_duplicate

unpinned

unsubscribed

user_blocked

For the REST APIs for issue events and timeline events, learn about each event type, the triggering action on GitHub, and each event's unique properties.

Issue events are triggered by activity in issues and pull requests and are available in the REST API for Issue events and Timeline events. Each event type specifies whether the event is available in the REST API for issue events or timeline events.

GitHub's REST API considers every pull request to be an issue, but not every issue is a pull request. For this reason, the Issue Events and Timeline Events endpoints may return both issues and pull requests in the response. Pull requests have a `pull_request` property in the `issue` object. Because pull requests are issues, issue and pull request numbers do not overlap in a repository. For example, if you open your first issue in a repository, the number will be 1. If you then open a pull request, the number will be 2. Each event type specifies if the event occurs in pull request, issues, or both.

Issue event object common properties

Issue events all have the same object structure, except events that are only available in the REST API for timeline events. Some events also include additional properties that provide more context about the event resources. Refer to the specific event for details about any properties that differ from this object format.

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

added_to_project

The issue or pull request was added to a project board. You only see this event in responses when project boards have been enabled in the repository.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for added_to_project

Note: Project card details are now shown in REST API responses for project-related issue and timeline events. This feature is now available for developers to preview. For details, see the blog post.

To receive the `project_card` attribute, project boards must be enabled for a repository, and you must provide a custom media type in the `Accept` header:

```
application/vnd.github.starfox-preview+json
```

Warning: The API may change without advance notice during the preview period. Preview features are not supported for production use. If you experience any issues, contact us through the GitHub Support portal.

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`project_card`	`object`	The project card object.
`project_id`	`integer`	The unique identifier of the

`project_id`	`integer`	The unique identifier of the project card.
`project_url`	`string`	The REST API URL to retrieve the project card.
`column_name`	`string`	The name of the column the issue is in.

assigned

The issue or pull request was assigned to a user.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for assigned

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`assignee`	`object`	The person assigned to this issue.
`assigner`	`object`	The person who performed the assignment for this issue. This field is available in the REST API for issue events but not the REST API for timeline events.

automatic_base_change_failed

GitHub unsuccessfully attempted to automatically change the base branch of the pull request.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	×

Properties for automatic_base_change_failed

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

automatic_base_change_succeeded

GitHub successfully attempted to automatically change the base branch of the pull request.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	×

Properties for automatic_base_change_succeeded

Name	Type	Description
`id`	`integer`	The unique identifier of the

`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

base_ref_changed

The base reference branch of the pull request changed.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	×

Properties for base_ref_changed

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.

`created_at`	`string`	The timestamp indicating when the event occurred.

closed

The issue or pull request was closed. When the `commit_id` is present, it identifies the commit that closed the issue using "closes / fixes" syntax. For more information about the syntax, see "Linking a pull request to an issue".

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for closed

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

commented

A comment was added to the issue or pull request.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	×	✓

Pull requests	×	✓

Properties for commented

This events properties do not use the same structure as the issue event object common properties. The following properties are returned for this event.

Name	Type	Description
`url`	`string`	The REST API URL to retrieve the issue comment.
`html_url`	`string`	The HTML URL of the issue comment.
`issue_url`	`string`	The HTML URL of the issue.
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`user`	`object`	The person who commented on the issue.
`created_at`	`string`	The timestamp indicating when the comment was added.
`updated_at`	`string`	The timestamp indicating when the comment was updated or created, if the comment is never updated.
`author_association`	`string`	The permissions the user has in the issue's repository. For example, the value would be `"OWNER"` if the owner of repository created a comment.
`body`	`string`	The comment body text.
`event`	`string`	The event value is `"commented"`.
`actor`	`object`	The person who generated the event.

committed

A commit was added to the pull request's `HEAD` branch.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	×	✓

Properties for committed

This events properties do not use the same structure as the issue event object common properties. The following properties are returned for this event.

Name	Type	Description
`sha`	`string`	The SHA of the commit in the pull request.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL to retrieve the commit.
`html_url`	`string`	The HTML URL of the commit.
`author`	`object`	The person who authored the commit.
`committer`	`object`	The person who committed the commit on behalf of the author.
`tree`	`object`	The Git tree of the commit.
`message`	`string`	The commit message.
`parents`	`array of objects`	A list of parent commits.
`verification`	`object`	The result of verifying the commit's signature. For more information, see "Git database."
`event`	`string`	The event value is `"committed"`.

connected

The issue or pull request was linked to another issue or pull request. For more information, see "Linking a pull request to an issue".

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for connected

Name	Type	Description
`id`	`integer`	The unique identifier of the event.

`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

convert_to_draft

The pull request was converted to draft mode.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	✓

Properties for convert_to_draft

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

converted_note_to_issue

The issue was created by converting a note in a project board to an issue. You only see this event in responses when project boards have been enabled in the repository.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓

Properties for converted_note_to_issue

Note: Project card details are now shown in REST API responses for project-related issue and timeline events. This feature is now available for developers to preview. For details, see the blog post.

To receive the `project_card` attribute, project boards must be enabled for a repository, and you must provide a custom media type in the `Accept` header:

```
application/vnd.github.starfox-preview+json
```

Warning: The API may change without advance notice during the preview period. Preview features are not supported for production use. If you experience any issues, contact us through the GitHub Support portal.

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`project_card`	`object`	The project card object.
`project_id`	`integer`	The unique identifier of the

`project_id`	`integer`	The unique identifier of the project card.
`project_url`	`string`	The REST API URL to retrieve the project card.
`column_name`	`string`	The name of the column the issue is in.

converted_to_discussion

The issue was closed and converted to a discussion.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	×

Properties for converted_to_discussion

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

cross-referenced

The issue or pull request was referenced from another issue or pull request.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	×	✓

Pull requests	✗	✓

Properties for cross-referenced

This events properties do not use the same structure as the issue event object common properties. The following properties are returned for this event.

Name	Type	Description
`actor`	`object`	The person who generated the event.
`created_at`	`string`	The timestamp indicating when the cross-reference was added.
`updated_at`	`string`	The timestamp indicating when the cross-reference was updated or created, if the cross-reference is never updated.
`source`	`object`	The issue or pull request that added a cross-reference.
`source[type]`	`string`	This value will always be `"issue"` because pull requests are of type issue. Only cross-reference events triggered by issues or pull requests are returned in the REST API for timeline events. To determine if the issue that triggered the event is a pull request, you can check if the `source[issue][pull_request]` object exists.
`source[issue]`	`object`	The `issue` object that added the cross-reference.
`event`	`string`	The event value is `"cross-referenced"`.

demilestoned

The issue or pull request was removed from a milestone.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for demilestoned

Name	Type	Description

`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`milestone`	`object`	The milestone object.
`milestone[title]`	`string`	The title of the milestone.

deployed

The pull request was deployed.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	✓

Properties for deployed

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that

`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

deployment_environment_changed

The pull request deployment environment was changed.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	×

Properties for deployment_environment_changed

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

disconnected

The issue or pull request was unlinked from another issue or pull request. For more information, see "Linking a pull request to an issue".

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events

Issues	✓	✓
Pull requests	✓	✓

Properties for disconnected

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

head_ref_deleted

The pull request's `HEAD` branch was deleted.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	✓

Properties for head_ref_deleted

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the

`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

head_ref_restored

The pull request's `HEAD` branch was restored to the last known commit.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	✓

head_ref_force_pushed

The pull request's HEAD branch was force pushed.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	✓

Properties for head_ref_force_pushed

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.

`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

labeled

A label was added to the issue or pull request.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for labeled

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`label`	`object`	The label removed from the issue
`label[name]`	`string`	The name of the label.
`label[color]`	`string`	The hex color code.

locked

The issue or pull request was locked.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for locked

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`lock_reason`	`string`	The reason an issue or pull request conversation was locked, if one was provided.

mentioned

The `actor` was `@mentioned` in an issue or pull request body.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for mentioned

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

marked_as_duplicate

A user with write permissions marked an issue as a duplicate of another issue, or a pull request as a duplicate of another pull request.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for marked_as_duplicate

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.

`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

merged

The pull request was merged. The `commit_id` attribute is the SHA1 of the `HEAD` commit that was merged. The `commit_repository` is always the same as the main repository.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	✓

Properties for merged

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

milestoned

The issue or pull request was added to a milestone.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for milestoned

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`milestone`	`object`	The milestone object.
`milestone[title]`	`string`	The title of the milestone.

moved_columns_in_project

The issue or pull request was moved between columns in a project board. You only see this event in responses when project boards have been enabled in the repository.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for moved_columns_in_project

Note: Project card details are now shown in REST API responses for project-related issue and timeline events. This feature is now available for developers to preview. For details, see the blog post.

To receive the `project_card` attribute, project boards must be enabled for a repository, and you must provide a custom media type in the `Accept` header:

```
application/vnd.github.starfox-preview+json
```

Warning: The API may change without advance notice during the preview period. Preview features are not supported for production use. If you experience any issues, contact us through the GitHub Support portal.

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`project_card`	`object`	The project card object.
`project_id`	`integer`	The unique identifier of the project card.
`project_url`	`string`	The REST API URL to retrieve the project card.
`column_name`	`string`	The name of the column the issue is in.
`previous_column_name`	`string`	The name of the column the issue was moved from.

pinned

The issue was pinned.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓

Properties for pinned

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

ready_for_review

A draft pull request was marked as ready for review.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	✓

Properties for ready_for_review

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.

`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

referenced

The issue was referenced from a commit message. The `commit_id` attribute is the commit SHA1 of where that happened and the commit_repository is where that commit was pushed.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for referenced

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

removed_from_project

The issue or pull request was removed from a project board. You only see this event in responses when project boards have been enabled in the repository.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for removed_from_project

Note: Project card details are now shown in REST API responses for project-related issue and timeline events. This feature is now available for developers to preview. For details, see the blog post.

To receive the `project_card` attribute, project boards must be enabled for a repository, and you must provide a custom media type in the `Accept` header:

```
application/vnd.github.starfox-preview+json
```

Warning: The API may change without advance notice during the preview period. Preview features are not supported for production use. If you experience any issues, contact us through the GitHub Support portal.

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`project_card`	`object`	The project card object.
`project_id`	`integer`	The unique identifier of the

`project_id`	`integer`	The unique identifier of the project card.
`project_url`	`string`	The REST API URL to retrieve the project card.
`column_name`	`string`	The name of the column the issue is in.

renamed

The issue or pull request title was changed.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for renamed

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`rename`	`object`	The name details.
`rename[from]`	`string`	The previous name.
`rename[to]`	`string`	The new name.

reopened

The issue or pull request was reopened.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for reopened

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

review_dismissed

The pull request review was dismissed.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	✓

Properties for review_dismissed

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the

~~node_id~~	~~string~~	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`dismissed_review`	`object`	The information for the dismissed review.
`dismissed_review[state]`	`string`	The state the pull request was in when it was dismissed. Can be one of: `commented`, `approved`, or `changes_requested`.
`dismissed_review[review_id]`	`string`	The unique identifier of the pull request's review.
`dismissed_review[dismissal_message]`	`string`	The message the user included when dismissing the review.
`dismissed_review[dismissal_commit_id]`	`string`	The unique identifier of the commit that dismissed the review if one exists.

review_requested

A pull request review was requested.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	✓

Properties for review_requested

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.

`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`requested_reviewer`	`object`	The person requested to review the pull request.
`review_requester`	`object`	The person who requested a review.

review_request_removed

A pull request review request was removed.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✓	✓

Properties for review_request_removed

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the

`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`requested_reviewer`	`object`	The person requested to review the pull request.
`review_requester`	`object`	The person who requested a review.

reviewed

The pull request was reviewed.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Pull requests	✕	✓

Properties for reviewed

This events properties do not use the same structure as the issue event object common properties. The following properties are returned for this event.

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`user`	`object`	The person who commented on the issue.
`body`	`string`	The review summary text.
`commit_id`	`string`	The SHA of the latest commit in the pull request at the time of the review.
`submitted_at`	`string`	The timestamp indicating when the review was submitted.
`state`	`string`	The state of the submitted review. Can be one of: `commented`, `changes_requested`, `approved` or `dismissed`.
`html_url`	`string`	The HTML URL of the review.
`pull_request_url`	`string`	The REST API URL to retrieve the pull request.
`author_association`	`string`	The permissions the user has in the issue's repository. For

		the issue's repository. For example, the value would be `"OWNER"` if the owner of repository created a comment.
`_links`	`object`	The `html_url` and `pull_request_url`.
`event`	`string`	The event value is `"reviewed"`.

subscribed

Someone subscribed to receive notifications for an issue or pull request.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for subscribed

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

transferred

The issue was transferred to another repository.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓

Properties for transferred

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

unassigned

A user was unassigned from the issue.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for unassigned

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.

`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`assignee`	`object`	The person unassigned from this issue.
`assigner`	`object`	The person who performed the unassignment for this issue. This field is available in the REST API for issue events but not the REST API for timeline events.

unlabeled

A label was removed from the issue.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for unlabeled

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that

`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`label`	`object`	The label removed from the issue
`label[name]`	`string`	The name of the label.
`label[color]`	`string`	The hex color code.

unlocked

The issue was unlocked.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for unlocked

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.
`lock_reason`	`string`	The reason an issue or pull

`lock_reason`	`string`	The reason an issue or pull request conversation was locked, if one was provided.

unmarked_as_duplicate

An issue that a user had previously marked as a duplicate of another issue is no longer considered a duplicate, or a pull request that a user had previously marked as a duplicate of another pull request is no longer considered a duplicate.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for unmarked_as_duplicate

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

unpinned

The issue was unpinned.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓

Properties for unpinned

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

unsubscribed

Someone unsubscribed from receiving notifications for an issue or pull request.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✕	✓
Pull requests	✕	✓

Properties for unsubscribed

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of

`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

user_blocked

An organization owner blocked a user from the organization. This was done through one of the blocked user's comments on the issue.

This event is available for the following issue types.

	REST API for issue events	REST API for timeline events
Issues	✓	✓
Pull requests	✓	✓

Properties for user_blocked

Name	Type	Description
`id`	`integer`	The unique identifier of the event.
`node_id`	`string`	The Global Node ID of the event.
`url`	`string`	The REST API URL for fetching the event.
`actor`	`object`	The person who generated the event.
`event`	`string`	Identifies the actual type of event that occurred.
`commit_id`	`string`	The SHA of the commit that referenced this issue.
`commit_url`	`string`	The GitHub REST API link to the commit that referenced this issue.
`created_at`	`string`	The timestamp indicating when the event occurred.

Legal

© 2023 GitHub, Inc. Terms Privacy Status Pricing Expert services Blog