

# Breaking changes

## In this article

- About breaking changes
- Changes scheduled for 2023-10-01
- Changes scheduled for 2023-07-01
- Changes scheduled for 2023-04-01
- Changes scheduled for 2023-01-01
- Changes scheduled for 2022-10-01
- Changes scheduled for 2022-07-01
- Changes scheduled for 2021-10-01
- Changes scheduled for 2021-06-21
- Changes scheduled for 2021-01-01
- Changes scheduled for 2020-10-01
- Changes scheduled for 2020-01-01
- Changes scheduled for 2019-04-01

Learn about recent and upcoming breaking changes to the GitHub GraphQL API.

## About breaking changes

Breaking changes are any changes that might require action from our integrators. We divide these changes into two categories:

- **Breaking:** Changes that will break existing queries to the GraphQL API. For example, removing a field would be a breaking change.
- **Dangerous:** Changes that won't break existing queries but could affect the runtime behavior of clients. Adding an enum value is an example of a dangerous change.

We strive to provide stable APIs for our integrators. When a new feature is still evolving, we release it behind a [schema preview](#).

We'll announce upcoming breaking changes at least three months before making changes to the GraphQL schema, to give integrators time to make the necessary adjustments. Changes go into effect on the first day of a quarter (January 1st, April 1st, July 1st, or October 1st). For example, if we announce a change on January 15th, it will be made on July 1st.

## Changes scheduled for 2023-10-01

- **Breaking** A change will be made to `PullRequestReviewComment.position`.

### Description:

`position` will be removed. Use the `line` and `startLine` fields instead, which are file line numbers instead of diff line numbers

**Reason:**

We are phasing out diff-relative positioning for PR comments

- **Breaking** A change will be made to `PullRequestReviewComment.originalPosition` .

**Description:**

`originalPosition` will be removed.

**Reason:**

We are phasing out diff-relative positioning for PR comments

- **Breaking** A change will be made to `AddPullRequestReviewInput.comments` .

**Description:**

`comments` will be removed. use the `threads` argument instead

**Reason:**

We are deprecating comment fields that use diff-relative positioning

- **Breaking** A change will be made to `AddPullRequestReviewCommentInput.pullRequestId` .

**Description:**

`pullRequestId` will be removed. use `addPullRequestReviewThread` or `addPullRequestReviewThreadReply` instead

**Reason:**

We are deprecating the `addPullRequestReviewComment` mutation

- **Breaking** A change will be made to `AddPullRequestReviewCommentInput.pullRequestId` .

**Description:**

`pullRequestId` will be removed. use `addPullRequestReviewThread` or `addPullRequestReviewThreadReply` instead

**Reason:**

We are deprecating the `addPullRequestReviewComment` mutation

- **Breaking** A change will be made to `AddPullRequestReviewCommentInput.position` .

**Description:**

`position` will be removed. use `addPullRequestReviewThread` or `addPullRequestReviewThreadReply` instead

**Reason:**

We are deprecating the `addPullRequestReviewComment` mutation

- **Breaking** A change will be made to `AddPullRequestReviewCommentInput.path` .

**Description:**

`path` will be removed. use `addPullRequestReviewThread` or `addPullRequestReviewThreadReply` instead

**Reason:**

We are deprecating the `addPullRequestReviewComment` mutation

- **Breaking** A change will be made to `AddPullRequestReviewCommentInput.inReplyTo` .

**Description:**

`inReplyTo` will be removed. use `addPullRequestReviewThread` or `addPullRequestReviewThreadReply` instead

**Reason:**

We are deprecating the `addPullRequestReviewComment` mutation

- **Breaking** A change will be made to `AddPullRequestReviewCommentInput.commitOID` .

**Description:**

`commitOID` will be removed. use `addPullRequestReviewThread` or `addPullRequestReviewThreadReply` instead

**Reason:**

We are deprecating the `addPullRequestReviewComment` mutation

- **Breaking** A change will be made to `AddPullRequestReviewCommentInput.body`.

**Description:**

`body` will be removed. use `addPullRequestReviewThread` or `addPullRequestReviewThreadReply` instead

**Reason:**

We are deprecating the `addPullRequestReviewComment` mutation

## Changes scheduled for 2023-07-01 [🔗](#)

---

- **Breaking** A change will be made to `ProjectV2ItemFieldGroup.field`.

**Description:**

`field` will be removed. Check out the `ProjectV2ItemFieldGroup#groupByField` API as an example for the more capable alternative.

**Reason:**

The `ProjectV2ItemFieldGroup#field` API is deprecated in favour of the more capable `ProjectV2ItemFieldGroup#groupByField` API.

- **Breaking** A change will be made to `Commit.pushedDate`.

**Description:**

`pushedDate` will be removed.

**Reason:**

`pushedDate` is no longer supported.

## Changes scheduled for 2023-04-01 [🔗](#)

---

- **Breaking** A change will be made to `Repository.squashPrTitleUsedAsDefault`.

**Description:**

`squashPrTitleUsedAsDefault` will be removed. Use `Repository.squashMergeCommitTitle` instead.

**Reason:**

`squashPrTitleUsedAsDefault` will be removed.

- **Breaking** A change will be made to `ProjectV2View.verticalGroupBy`.

**Description:**

`verticalGroupBy` will be removed. Check out the `ProjectV2View#vertical_group_by_fields` API as an example for the more capable alternative.

**Reason:**

The `ProjectV2View#vertical_group_by` API is deprecated in favour of the more capable `ProjectV2View#vertical_group_by_fields` API.

- **Breaking** A change will be made to `ProjectV2View.sortBy`.

**Description:**

`sortBy` will be removed. Check out the `ProjectV2View#sort_by_fields` API as an example for the more capable alternative.

**Reason:**

The `ProjectV2View#sort_by` API is deprecated in favour of the more capable `ProjectV2View#sort_by_fields` API.

- **Breaking** A change will be made to `ProjectV2View.groupBy`.

**Description:**

`groupBy` will be removed. Check out the `ProjectV2View#group_by_fields` API as an

example for the more capable alternative.

**Reason:**

The `ProjectV2View#order_by` API is deprecated in favour of the more capable `ProjectV2View#group_by_field` API.

## Changes scheduled for 2023-01-01 [↗](#)

---

- **Breaking** A change will be made to `ProjectV2View.visibleFields`.

**Description:**

`visibleFields` will be removed. Check out the `ProjectV2View#fields` API as an example for the more capable alternative.

**Reason:**

The `ProjectV2View#visibleFields` API is deprecated in favour of the more capable `ProjectV2View#fields` API.

- **Breaking** A change will be made to `Commit.changedFiles`.

**Description:**

`changedFiles` will be removed. Use `changedFilesIfAvailable` instead.

**Reason:**

`changedFiles` will be removed.

## Changes scheduled for 2022-10-01 [↗](#)

---

- **Breaking** A change will be made to `RemovePullRequestFromMergeQueueInput.branch`.

**Description:**

`branch` will be removed.

**Reason:**

PRs are removed from the merge queue for the base branch, the `branch` argument is now a no-op

- **Breaking** A change will be made to `DependencyGraphDependency.packageLabel`.

**Description:**

`packageLabel` will be removed. Use normalized `packageName` field instead.

**Reason:**

`packageLabel` will be removed.

## Changes scheduled for 2022-07-01 [↗](#)

---

- **Breaking** A change will be made to `AddPullRequestToMergeQueueInput.branch`.

**Description:**

`branch` will be removed.

**Reason:**

PRs are added to the merge queue for the base branch, the `branch` argument is now a no-op

## Changes scheduled for 2021-10-01 [↗](#)

---

- **Breaking** A change will be made to `ReactionGroup.users`.

**Description:**

`users` will be removed. Use the `reactors` field instead.

**Reason:**

Reactors can now be mannequins, bots, and organizations.

## Changes scheduled for 2021-06-21 [🔗](#)

---

- **Breaking** A change will be made to `PackageType.DOCKER`.

**Description:**

`DOCKER` will be removed.

**Reason:**

DOCKER will be removed from this enum as this type will be migrated to only be used by the Packages REST API.

## Changes scheduled for 2021-01-01 [🔗](#)

---

- **Breaking** A change will be made to `MergeStateStatus.DRAFT`.

**Description:**

`DRAFT` will be removed. Use `PullRequest.isDraft` instead.

**Reason:**

DRAFT state will be removed from this enum and `isDraft` should be used instead

## Changes scheduled for 2020-10-01 [🔗](#)

---

- **Breaking** A change will be made to `PullRequest.timeline`.

**Description:**

`timeline` will be removed. Use `PullRequest.timelineItems` instead.

**Reason:**

`timeline` will be removed

- **Breaking** A change will be made to `Issue.timeline`.

**Description:**

`timeline` will be removed. Use `Issue.timelineItems` instead.

**Reason:**

`timeline` will be removed

## Changes scheduled for 2020-01-01 [🔗](#)

---

- **Breaking** A change will be made to `UnassignedEvent.user`.

**Description:**

`user` will be removed. Use the `assignee` field instead.

**Reason:**

Assignees can now be mannequins.

- **Breaking** A change will be made to `AssignedEvent.user`.

**Description:**

`user` will be removed. Use the `assignee` field instead.

**Reason:**

Assignees can now be mannequins.

## Changes scheduled for 2019-04-01 [🔗](#)

---

- **Breaking** A change will be made to `LegacyMigration.uploadUrlTemplate`.

**Description:**

`uploadUrlTemplate` will be removed. Use `uploadUrl` instead.

**Reason:**

`uploadUrlTemplate` is being removed because it is not a standard URL and adds an extra user step.

**Legal**

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)