

# API Potential Breaking Changes Related to the "Branches" Feature Release - February 25, 2022

Beginning February 25, 2022, we will be gradually rolling out our "Branches" feature to all customers, which will be plan-based. For more information, you can refer to our [Branches](#) documentation.

There will be no change in the response of Content Delivery API (CDA) GET calls, however this feature release will introduce a change in the API responses returned for several CDA/CDN and [Content Management API \(CMA\)](#) requests. To facilitate a smooth transition when you start using branches, we have listed below a few potential breaking changes in the API that you might need to update within your backend code.

## [Potential Breaking Change] Additional "branch" Key in Webhook-related API Responses

If the "Branches" feature is part of your plan, all webhook-related API requests that hit the Content Management API (CMA) will return the `branch` key in the response body. This key specifies the unique ID of the branch on which the concerned webhook was triggered. Additionally, it also highlights the unique ID for any alias assigned to the branch and the source branch from which it inherits data.

Here's a sample of the webhook data returned along with the `branch` key when you run the [Get webhook executions](#) API request:

```
{
  "webhooks": [
    {
      "uid": "bltf00ec3b37082288b",
      "channel": [
        "content_types.create"
      ],
      "created_at": "2021-09-21T07:24:48.757Z",
      "event_data": {
        "module": "content_type",

```



```
"api_key":"blt29ec365eaa0c8d67",
"event":"create",
"data":{
  "uid":"new_content_type",
  "title":"New Content Type",
  "branch":{
    "alias":[
      "test"
    ],
    "uid":"development",
    "source":"main"
  }
},
"event_headers":{
  "Content-Type":"application/json",
  "User-Agent":"Contentstack",
  "X-Contentstack-Signature":"d7b6d307f35086885f64a8a8sd7s878d7s8"
},
"org_uid":"blteac54a27425b3b0e",
"request_details":[
  {
    ...
    "body":{
      "triggered_at":"2021-09-21T07:24:43.734Z",
      "module":"content_type",
      "api_key":"blt29ec365eaa0c8d67",
      "event":"create",
      "data":{
        "uid":"new_content_type",
        "title":"New Content Type",
        "branch":{
          "alias":[
            "test"
          ],
          "uid":"development",
          "source":"main"
        }
      }
    }
  },
  ...
],
```

```
        "headers":{
            ...
        },
        "json":true,
        "resolveWithFullResponse":true,
        "timeout":30000
    },
    "response":{
        "statusCode":500,
        "message":"Unable to connect to host.",
        "request":{
            "uri":{

href":"https://webhook.gatsby.com/hooks/data_source/12345"
            },
            "method":"POST",
            "headers":{
                "Content-Type":"application/json",
                "User-Agent":"Contentstack",

"X-Contentstack-Signature":"d7b6d307f35086885f64a8a2a47ef2e37483
7"
            }
        }
    },
    "created_at":"2021-09-21T07:24:48.757Z"
}
],
"retry_count":1,
"status":500,
"updated_at":"2021-09-21T07:25:28.860Z",
"webhooks":[
    "blt6b256750460b99c0"
],
"destination":{}
}
]
}
```

## Additional "branches" and "branch\_aliases" Keys to Specify Branch/Alias Scope in Responses

If the "Branches" feature is part of your plan, you will see additional keys named `branches` and `branch_aliases` in the API responses in some of the Contentstack modules (listed below). The `branches` and `branch_aliases` keys specify the unique IDs of the branches and aliases selected within the scope for the concerned modules.

- **Webhooks**-related API requests that hit the Content Management API (CMA) will return a top-level `branches` key in the response body. This key specifies the unique ID of the branch to which the webhook is applicable.

When creating or updating a webhook, you need to specify the branch scope through the following schema in the request body:

```
"branches": [
  "main"
]
```

For instance, here's the response that you will get for the [Create a webhook](#) API request:

```
{
  "notice": "The webhook was created successfully",
  "webhook": {
    "uid": "blta4c0a90071d12f38",
    "channels": [
      "assets.create"
    ],
    "branches": [
      "main"
    ],
    "disabled": false,
    "concise_payload": true,
    "name": "Test",
    "destinations": [
      {
        "target_url": "http://example.com",
        "http_basic_auth": "basic",
        "http_basic_password": "test",
```

```
        "custom_header": [
          {
            "header_name": "Custom",
            "value": "testing"
          }
        ]
      }
    ],
    "retry_policy": "manual",
    "updated_by": "blta7eaf6883dd73a0b",
    "created_by": "blta7eaf6883dd73a0b",
    "created_at": "2021-09-20T18:54:01.105Z",
    "updated_at": "2021-09-20T18:54:01.105Z"
  }
}
```

- **Workflows**-related API requests that hit the Content Management API (CMA) will return a top-level `branches` key in the response body. This key specifies the unique IDs of the branches to which the workflow is applicable.

When creating or updating a workflow, you need to specify the branch scope through the following schema in the request body:

```
"branches": [
  "main"
]
```

For instance, here's the response that you will get for the [Create a workflow](#) API request:

```
{
  "notice": "Workflow created successfully.",
  "workflow": {
    "workflow_stages": [
      {
        "color": "#74ba76",
        "SYS_ACL": {
          "others": {
            "read": true,
            "write": true,
            "transit": false
          }
        }
      }
    ]
  }
}
```

```
    },
    "users":{
        "uids":[
            "$all"
        ],
        "read":true,
        "write":true,
        "transit":true
    },
    "roles":{
        "uids":[

        ],
        "read":true,
        "write":true,
        "transit":true
    }
},
"next_available_stages":[
    "$all"
],
"allStages":true,
"allUsers":true,
"specificStages":false,
"specificUsers":false,
"name":"Review",
"uid":"blt7336c2c86ede8b9a"
},
{
    "color":"#2196f3",
    "SYS_ACL":{
        "others":{
            "read":true,
            "write":true,
            "transit":false
        },
        "users":{
            "uids":[
                "$all"
            ],
            "read":true,
```

```
        "write":true,
        "transit":true
    },
    "roles":{
        "uids":[

            ],
        "read":true,
        "write":true,
        "transit":true
    }
},
"allStages":true,
"allUsers":true,
"specificStages":false,
"specificUsers":false,
"next_available_stages":[
    "$all"
],
"name":"Complete",
"uid":"bltace4f008c3156e97"
}
],
"admin_users":{
    "users":[

    ],
    "roles":[
        "bltc5412bb640c8cce1"
    ]
},
"name":"Sample Workflow",
"enabled":true,
"branches":[
    "main"
],
"content_types":[
    "author",
    "header"
],
"uid":"bltf502c1bf4b742857",
```

```
    "api_key": "blt29ec365eaa0c8d67",
    "org_uid": "blteac54a27425b3b0e",
    "created_by": "blt1fd0057b90905592",
    "created_at": "2021-09-22T14:05:54.454Z",
    "deleted_at": false
  }
}
```

- **Publish Rules**-related API requests that hit the Content Management API (CMA) will return a top-level `branches` key in the response body. This key specifies the unique IDs of the branches for which the publishing rule is applicable.

When creating or updating a publishing rule, you need to specify the branch scope through the following schema in the request body:

```
"branches": [
  "main"
]
```

For instance, here's the response that you will get for the [Create publish rules](#) API request:

```
{
  "notice": "Publish rule created successfully.",
  "workflow": {
    "status": {
      "publishing_rules": true
    },
    "publishing_rules": [
      {
        "uid": "blta7e8171bdaf013cd",
        "actions": [],
        "environment": "blta6fa90d8fc8a35b3",
        "branches": [
          "main"
        ],
        "content_types": [
          "$all"
        ],
        "locales": [
          "en-us"
        ]
      }
    ]
  }
}
```



```
    ],
    "approvers": [
      "blt1fd0057b90905592"
    ],
    "status": true
  }
]
}
}
```

- **Delivery Token and Management Token**-related API requests that hit the Content Management API (CMA) will return the `branches` and `branch_aliases` keys under the `scope` section in the response body. These keys specify the unique IDs of the branches and branch aliases for which a delivery or management token is applicable.

When creating or updating a delivery or management token, you need to specify the branch and alias scope through the following schema in the request body:

```
{
  "module": "branch",
  "branches": [
    "main",
    "development"
  ],
  "acl": {
    "read": true
  }
},
{
  "module": "branch_alias",
  "branch_aliases": [
    "deploy",
    "release"
  ],
  "acl": {
    "read": true
  }
}
```

For instance, here's the response that you will get for the [Create a delivery token](#)

## API request:

```
{
  "notice": "Delivery Token created successfully.",
  "token": {
    "name": "Sample Delivery Token",
    "description": "Sample token.",
    "scope": [
      {
        "environments": [
          {
            "name": "staging",
            "servers": [
              {
                "name": "default"
              }
            ],
            "urls": [
              {
                "locale": "en-us",
                "url":
"http://example.com/"
              }
            ],
            "deploy_content": true,
            "app_user_object_uid": "system",
            "uid": "blta6fa90d8fc8a35b3",
            "created_by":
"blta7eaf6883dd73a0b",
            "updated_by":
"blta7eaf6883dd73a0b",
            "created_at":
"2021-09-16T19:27:18.898Z",
            "updated_at":
"2021-09-16T19:27:18.898Z",
            "ACL": [],
            "_version": 1,
            "tags": []
          }
        ],
        "module": "environment",

```

```
      "acl": {
        "read": true
      },
      "_metadata": {
        "uid": "csfa77cbd080f5afbd"
      }
    },
    {
      "module": "branch",
      "acl": {
        "read": true
      },
      "branches": [
        "main",
        "development"
      ],
      "_metadata": {
        "uid": "cs766df728fb56e697"
      }
    },
    {
      "module": "branch_alias",
      "acl": {
        "read": true
      },
      "branch_aliases": [
        "deploy",
        "release"
      ],
      "_metadata": {
        "uid": "cs27a40bf57db84414"
      }
    }
  ],
  "uid": "blt9d62d10720189b2e",
  "created_by": "blt1fd0057b90905592",
  "updated_by": "blt1fd0057b90905592",
  "created_at": "2021-09-22T16:01:34.410Z",
  "updated_at": "2021-09-22T16:01:34.410Z",
  "token": "cs93d9e3bb4bf8a0612d8bdb77",
  "type": "delivery",
```

```
        "_metadata": {
            "references": []
        }
    }
}
```

- **User Roles**-related API requests that hit the Content Management API (CMA) will return the `branches` and `branch_aliases` keys under the `rules` section in the response body. These keys specify the unique IDs of the branches and branch aliases of which a particular user role can access data.

When creating or updating a user role, you need to specify the branch and alias scope through the following schema in the request body:

```
{
  "module": "branch",
  "branches": [
    "main"
  ],
  "acl": {
    "read": true
  }
},
{
  "module": "branch_alias",
  "branch_aliases": [
    "deploy"
  ],
  "acl": {
    "read": true
  }
}
```

For instance, here's the response that you will get for the [Create a role](#) API request:

```
{
  "notice": "The role created successfully.",
  "role": {
    "name": "Demo role",
    "description": "Demo role."
  }
}
```

```
"rules":[
  {
    "module":"content_type",
    "content_types":[
      "author"
    ],
    "acl":{
      "read":true,
      "sub_acl":{
        "read":true,
        "create":true,
        "update":true,
        "publish":true,
        "delete":true
      }
    }
  },
  {
    "module":"asset",
    "assets":[
      "$all"
    ],
    "acl":{
      "read":true,
      "create":true,
      "update":true,
      "publish":true,
      "delete":true
    }
  },
  {
    "module":"environment",
    "environments":[
      "blta6fa90d8fc8a35b3"
    ],
    "acl":{
      "read":true
    }
  },
  {
    "module":"locale",
```

```
        "locales": [
            "blt37255d0c5a6d20b1"
        ],
        "acl": {
            "read": true
        }
    },
    {
        "module": "branch",
        "branches": [
            "main"
        ],
        "acl": {
            "read": true
        }
    },
    {
        "module": "branch_alias",
        "branch_aliases": [
            "deploy"
        ],
        "acl": {
            "read": true
        }
    }
],
"users": [

],
"uid": "bltc590a9b813590e8e",
"org_uid": "blteac54a27425b3b0e",
"api_key": "blt29ec365eaa0c8d67",
"created_by": "blt1fd0057b90905592",
"updated_by": "blt1fd0057b90905592",
"created_at": "2021-09-23T08:09:30.971Z",
"updated_at": "2021-09-23T08:09:30.971Z"
}
}
```

## [Potential Breaking Change] Updated Asset Download URL Format as per Branch

If the "Branches" feature is part of your plan, you will see an updated download URL format for all assets part of any branch, **except the default `main` branch**. The `branch={branch_UID}` query parameter is attached to the download URL.

For instance, if you upload an asset to the `development` branch, Contentstack generates the following asset download URL:

```
https://assets.contentstack.com/v3/assets/blt252b550c5ccdae26/bltc56f275260214c0b/61280d9625aca65092ce7f15/Sample_Image.jpg?branch=development
```

For any asset that you upload to the `main` branch, Contentstack will continue to generate asset download URLs in the earlier format:

```
https://assets.contentstack.com/v3/assets/blt252b550c5ccdae26/bltc56f275260214c0b/61280d9625aca65092ce7f15/Sample_Image.jpg
```

## Support "include\_branch" as Query Parameter in CMA and CDA API Requests

If the "Branches" feature is part of your plan, you can pass the `include_branch` query parameter while running [Content Delivery API \(CDA\)](#) / CDN and [Content Management API \(CMA\)](#) requests. Set the `include_branch` key to `true` to allow the API to return the `_branch` top-level key in the response. This key specifies the unique ID of the branch and helps identify where the concerned Contentstack module resides.

Here's the list of the components that support the `include_branch` query parameter:

- Content Types [CDA / CMA]
- Global fields [CDA / CMA]
- Assets [CDA / CMA]
- Entries [CDA / CMA]
- Extensions
- Languages
- Releases

- Audit Log
- Publish Queue

**Note:** For Publish Queue and Audit Log related API requests, instead of `"_branch": "{branch_ID}"`, you will see the `"branch": "{branch_ID}"` top-level key in the response.

Let us look at an example to understand how the response would look when you pass the `include_branch` query parameter. For instance, here's the response that you will get for the [Get all content types](#) API request:

```
{
  "content_types": [
    {
      ...,
      "schema": [
        {
          "display_name": "Title",
          ...
        },
        {
          "display_name": "URL",
          ...
        },
      ],
      "last_activity": {},
      "maintain_revisions": true,
      "description": "",
      "DEFAULT_ACL": {
        "others": {
          "read": false,
          "create": false
        },
      },
      "users": [
        {
          "read": true,
          "sub_acl": {
            "read": true
          },
          "uid": "blt690a3fed2a593445"
        }
      ]
    }
  ]
}
```



```
    ],
    "management_token": {
      "read": true
    }
  },
  "SYS_ACL": {
    "roles": [
      {
        "uid": "bltdc9d89cd41565777",
        "read": true,
        "sub_acl": {
          ...
        },
        "update": true,
        "delete": true
      },
      {
        ...
      },
      {
        ...
      },
      ...
    ]
  },
  "others": {
    ...
  }
},
"_branch": "development",
"options": {
  "title": "title",
  "publishable": true,
  "is_page": true,
  "singleton": false,
  "sub_title": [
    "url"
  ],
  "url_pattern": "[:title]",

```

```
        "url_prefix": "/"
    },
    "abilities": {
        "get_one_object": true,
        "get_all_objects": true,
        "create_object": true,
        "update_object": true,
        "delete_object": true,
        "delete_all_objects": true
    }
}
]
```

**Note:** The “branches” feature is available in Contentstack only for the new interface.