

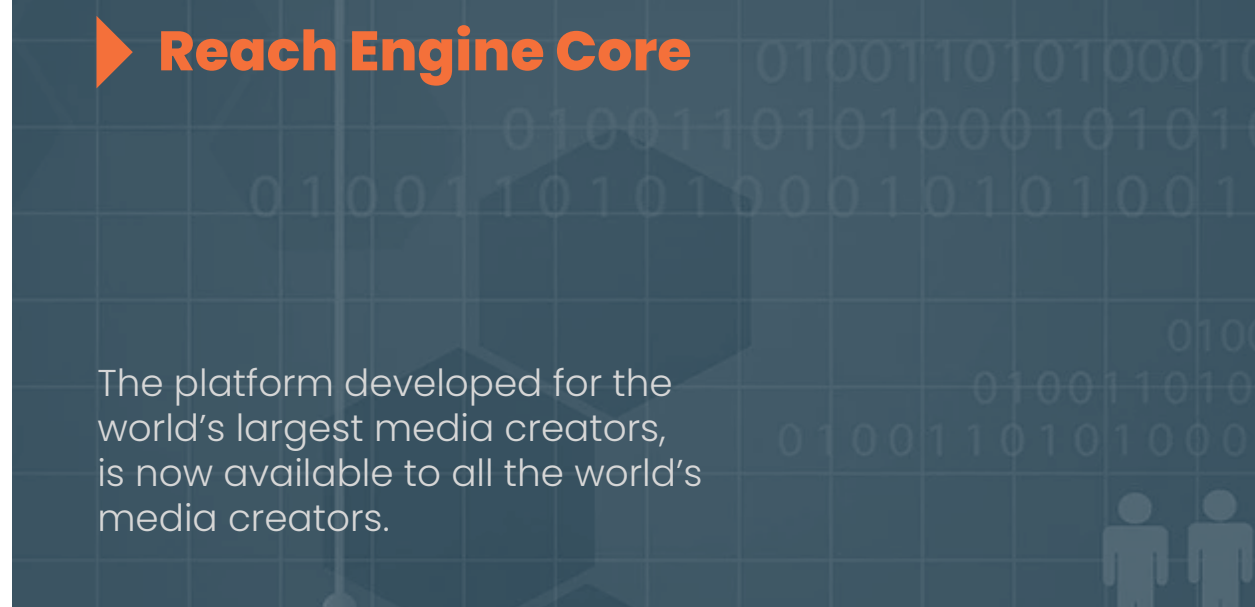
## Version 2.15.0

July 30, 2021 v5



### ► Reach Engine Core

The platform developed for the world's largest media creators, is now available to all the world's media creators.



# Table of Contents

|                                  |    |
|----------------------------------|----|
| <b>Performance Improvements</b>  | 01 |
| <b>Reach Engine Core</b>         | 03 |
| Multi-Subject Workflows          | 03 |
| Multi-File Workflow Inputs       | 05 |
| Workflow Scheduling Enhancements | 06 |
| Corrected Bugs                   | 07 |

# Performance Improvements

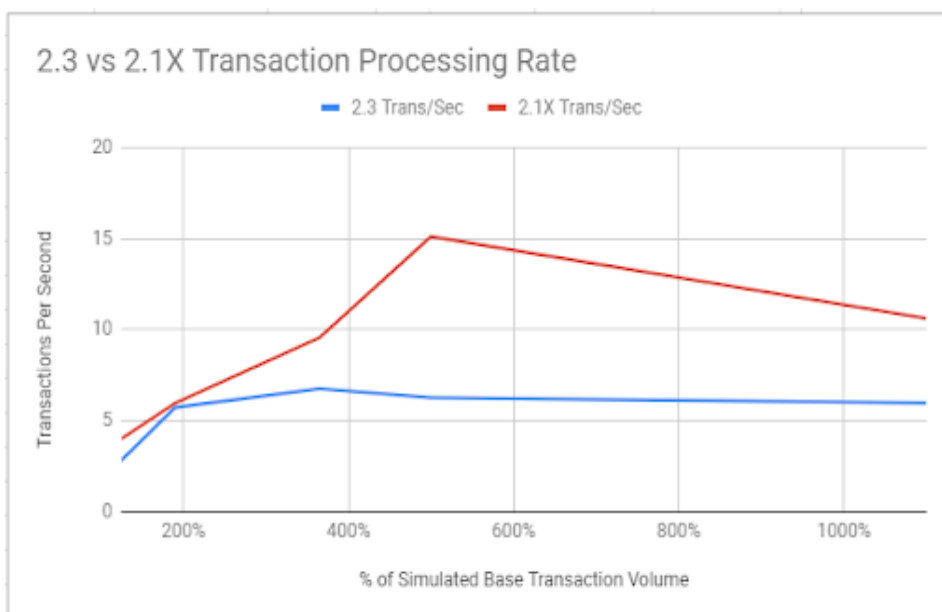
## Performance Testing Results

The introduction of the option to scale via additional workflow nodes has provided us the opportunity to take REACH ENGINE in to the lab and see just how far we've come since 2.3. From 2.10 onwards, the REACH ENGINE workflow engine can support multiple nodes in parallel allowing horizontal scaling in workflow executions. On top of that, we've also worked hard to increase the performance of the workflow engine to reduce the total processing time of key workflows as well as improving individual transaction time of targeted processes.

The results are really visible and we're excited to share them with you below.

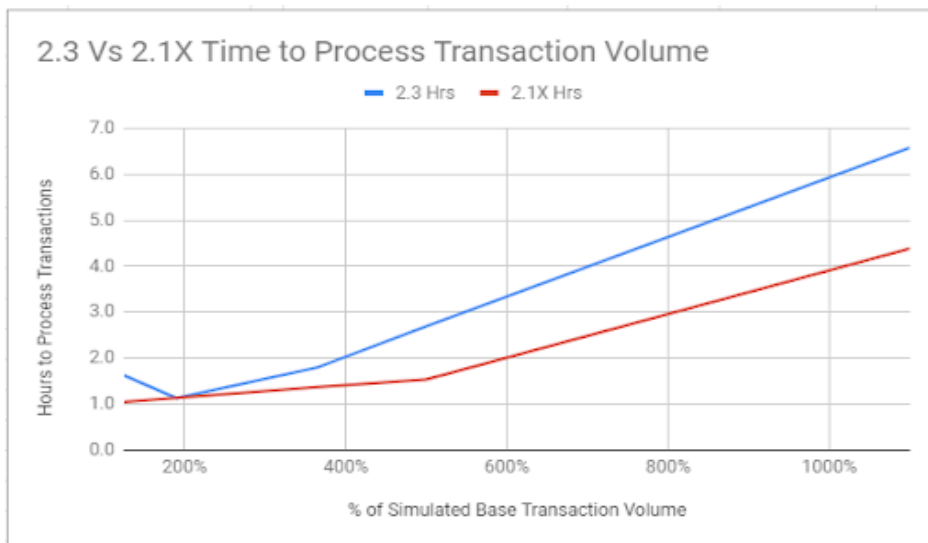
### 2.3 vs. 2.1X Transaction Processing Rate

- 2.1x outperformed 2.3 in every run for transactions executed per second
- 2.1x performed at a transactions per second average that was 59% better than the 2.3 average across all tests.
- At its peak, 2.3 was seen to perform at 6.8 transactions per second. 2.1x was seen to perform at 223% higher of a peak
- At the upper volumes of transaction testing (1000%), we were able to start seeing degradation to the database resources - this is exactly where we want to see the bottleneck pushed when tuned appropriately. Anticipation is that if we hit this point when live, increasing resources on the DB server will allow us to continue to scale



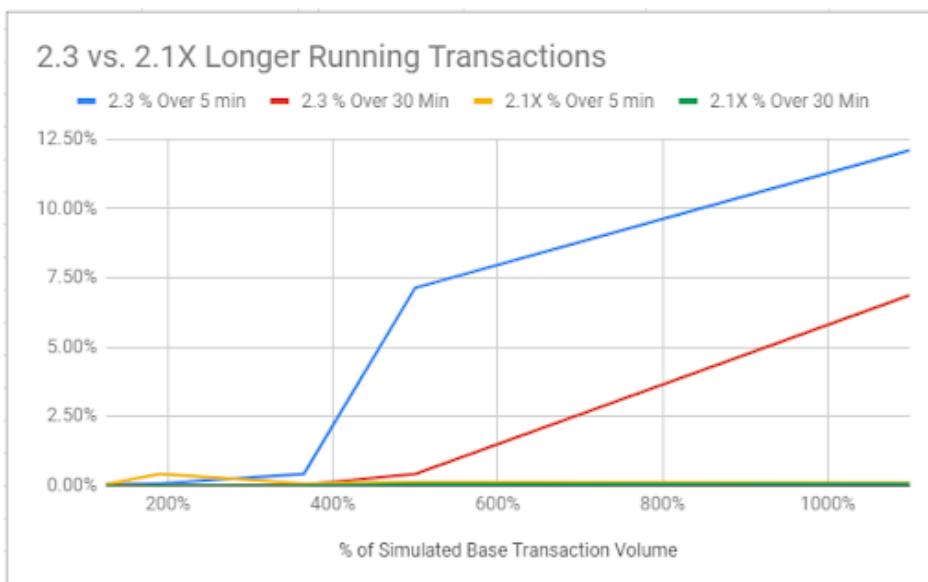
## 2.3 vs. 2.1X Time to Process Transaction Volume

- 2.1x was able to process transaction volumes against clock time at an average of 33% faster than 2.3 across all tests
- 2.1x was able to process nearly 2 times what has been seen to be an enterprise client transaction volume within an hour testing period



## 2.3 vs. 2.1X Longer Running Transactions

- Across all tests, 2.1x was consistently processing transactions within reasonable individual time frames- it never had more that .5% of transactions over 5 minutes
- As transaction volume increased, 2.3 saw an increasing number of transactions that were taking an extended amount of time



## REACH ENGINE Core

### Multi-Subject Workflows

A multi-subject workflow is the ability to run a single workflow for a batch of different assets. The main advantage is to have an easier tracking of all the workflow executions viewed together as one workflow in the system, while still maintaining the ability to track the individual processes on each individual asset as well.

#### What's New?

- New workflow header in XML to enable multiple subjects
- Allow a user to select multiple assets in the UI and submit them to a multi-subject workflow and result in one workflow execution

#### Why is it Valuable?

- Process assets in a single workflow and retain their related context
- Manage one request as a single execution for ease of management and troubleshooting
- Backwards compatibility ensures no migration or workflow rewrites are required and appropriate use cases can be implemented with Multi-Subject at the right time

#### Important Notes

This can be done on a per-workflow basis with a new workflow header attribute ***multiSubjectEnabled***.

```
id="multiSubjectWorkflow"
name="Multi Subject Workflow"
executionLabelExpression="Received ${subjects.size()} subjects"
subjectDOClassName="VideoAssetMaster"
multiSubjectEnabled="true"
```

One very important detail to note is that you cannot simply change a single subject workflow into a multi-subject workflow without any refactoring. This is specifically due to the unique syntax required to access subjects in a multi-subject execution. There is a new keyword used to access subjects called, as you may have guessed, `subjects`, however this is not directly an array of the target `DataObjects`. Due to the way the association is stored in the database the `subjects` member of an execution is actually an array of a new object type called `WorkflowExecutionSubject`, which contains a member, `subject`, which is the actual subject `Data Object`.

Here's an example `dataDef` accepting multiple subjects:

```
<contextDataDef name="allTheSubjects" dataType="Data Object"
multiple="true" defaultDataExpression="${subjects.![subject]}"/>
```

And here's an example of sending multiple subjects to a subflow which will result in a subflow execution for each subject.

```
<executeSubflowStep
  name="process multiple asset masters"
  targetWorkflowId="_processSingleAssetMaster"
  subjectChangePath="${workflow.subjects.![subject]}"/>
```

## Client Story

- Multiple different assets that are related to each other need to be processed together but don't want to waste time putting into a collection to process
- Select a few clips and have them stitched together in one workflow
- Select a long form video, a short form video and an image to distribute as a singular package

## Multi-file Workflow Inputs

A user running a File ingest workflow from the UI can select multiple files from either their local system or the "remote" browser and ingest them all at once in a single operation. The user also has the opportunity to enter metadata (from a metadata form) and/or define the collection where the files should be ingested.

### What's New?

- When a workflow FileInput has "multiple=true" then the UI modal will allow the user to select multiple files from the remote and local file input picker
- Users can mix local files and remote files in the same input
- A user can use the Shift or CMD/CNTRL keys on their keyboard to select multiple local files as the inputs
- File input modals will persist with one or more files selected so that files in different locations can be uploaded to the same modal
- Selected assets will upload and be displayed & managed individually
- Individual files can be removed after upload

### Why is it Valuable?

- Get more work done, faster, by ingesting batches of files
- Easily apply metadata and orchestrations to multiple files at point of ingest

**Ingest Asset** [X]

Collections [v]

File To Ingest

MY COMPUTER

SHARED STORAGE

Selected

|      |      |   |
|------|------|---|
| test | 100% | X |
| test | 100% | X |
| test | 100% | X |
| test | 100% | X |

Metadata Form [v]

RESET RUN WORKFLOW

## Workflow Scheduling Enhancements

Improvements to the Workflow Engine and the workflow scheduling logic to improve the efficiency of the platform, simplify custom workflow design/architecture and associated maintenance.

### What's New?

#### 1. SCHEDULING OF WORKFLOW SUBFLOWS

Ability to schedule workflow subflows in the REACH ENGINE scheduler. This allows the engine to run subflows independently of each other and be prioritized for more efficiency and flexibility in the workflow execution.

##### What does it mean, technically?

- A new property "scheduled" has been added to the "ExecuteSubflowsStep" to enable the feature.
- Scheduled subflows will be processed first by priority then by date of creation.

#### 2. SCHEDULED EXECUTION LIMITS

Allow to define a limit for workflows that interact with limited technical resources. Technically this will limit the number of workflow instances that can be executed in parallel to avoid any resource overflow.

##### Additional Information:

- A new "scheduledExecutionLimit" header has been added to the workflow XML to support the feature
- Synchronous subflows will not adhere to the limit

### Why is it Valuable?

In a scenario where an underlying system cannot handle the load being processed by Reach Engine, for example a transcoder that can't handle more than 10 jobs and isn't utilized using a native workflow step, the limit can be implemented at the workflow level so that an infinite number of transcode requests can be made by users or other workflows, but only the limit will be in an executing state in parallel. This feature can be combined with workflow level prioritization to ensure that scheduled workflows that exceed the limit will process in priority order.



## Corrected Bugs

| ID    | Issue Request/Reported   | Issue Resolution   |
|-------|--|--|
| 88276 | Sharing multiple clips with an external user causes stalled Collaborate workflows                      | Resolves an issue in Collaborate where selecting multiple clips to share with a user resulted in a stall.  |
| 90525 | Non-admins cannot start workflow via api using workflowkey   | Resolves an issue where non admins canuse the key to execute a/start call of a wovia API.  |
| 80567 | Marker Collision with other Asset-Types  | Resolves an issue where the creation or of a Marker would indicate that an asset different type but same numerical ID was updated, even though it wasn't.  |
| 88591 | Collection dataObjectSearch in workflow not adhering to set permissions during collection modification | Resolves an issue where a user may under certain conditions may be able to update a permission restricted collection within a workflow-modal.  |
| 88613 | Collections with large amounts of assets take longer to load in collection drop down option            | Resolves an issue where a Workflow modal presenting a dropdown list of Collections was decreasingly performant in relation to number of assets in any collection.  |
| 88633 | DataObjectSearch -- Result Set not aligning data types   | Resolves an issue where Workflow modal would not display when there was a ColledataObject query populating a picklist in forms and any item in the collections returned the query contained anon-integer audiocodec. |
| 88636 | Workflows can not create Timeline track names greater than 15 characters                               | Resolves an issue where Timeline track names were limited to 15 characters when being added via a workflow   |
| 81268 | Update resource paths for XSDs served by reach-engine  | Resolves an issue where the workflow's resources were not returned with NGINX proxying API requests.   |
| 59311 | Cannot Delete Role if Facet Group associated to it was deleted prior                                   | Resolves an issue where a role could not be deleted when it still contained a phantom association with a metadata form.  |
| 88558 | Static Workflow Runtime LoadService does not work asintended   | Resolves an issue where workflows are nas capacity frees and the value of max.running.workflow.count would not enforced when the value of running work-could not be accurately calculated.                           |

| ID    | Issue Request/Reported  | Issue Resolution   |
|-------|---|--|
| 89576 | maxConcurrent no longer limitsthe number of executions run bywatch-folder triggers            | Resolves an issue where the value input watch folder's "maxConcurrent" configuration would not limit the number of files and subsequent workflow executions sent to Workflow Engine's Scheduler service. |
| 89558 | Workflow steps will not autocom-plete with correctlyconfigured studio xsd in workflow headers | Resolved an issue with workflow XSD autocompletion when using a server's dir hosted schemas.   |
| 89588 | Media Profile file extension validation does not work against object storage                  | Resolved an issue where file extensions were not properly pulled from files on object storage. The solution involves a wrapper script atop MediaInfo.  |
| 90734 | GCS Filesystem Credentials Path is not unencrypted before being used                          | Resolved an issue where Google Cloud Storage credentials were not being encrypted and decrypted properly when the integration was setup or modified by the UI.   |