Main Features

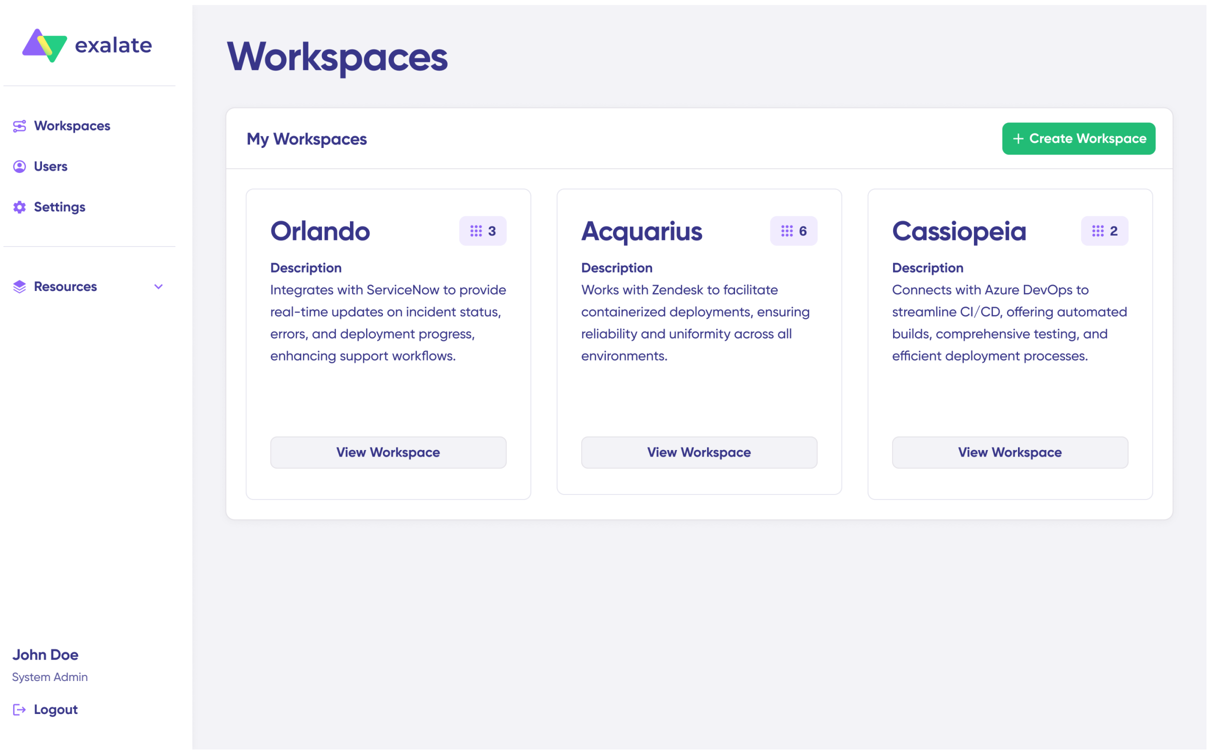
Last Modified on 12/18/2025 3:50 am EST

Formerly SyncRoom - now part of the Exalate Console Early Access.

Exalate Console provides a range of features to help streamline connection and node management:

Create Workspaces

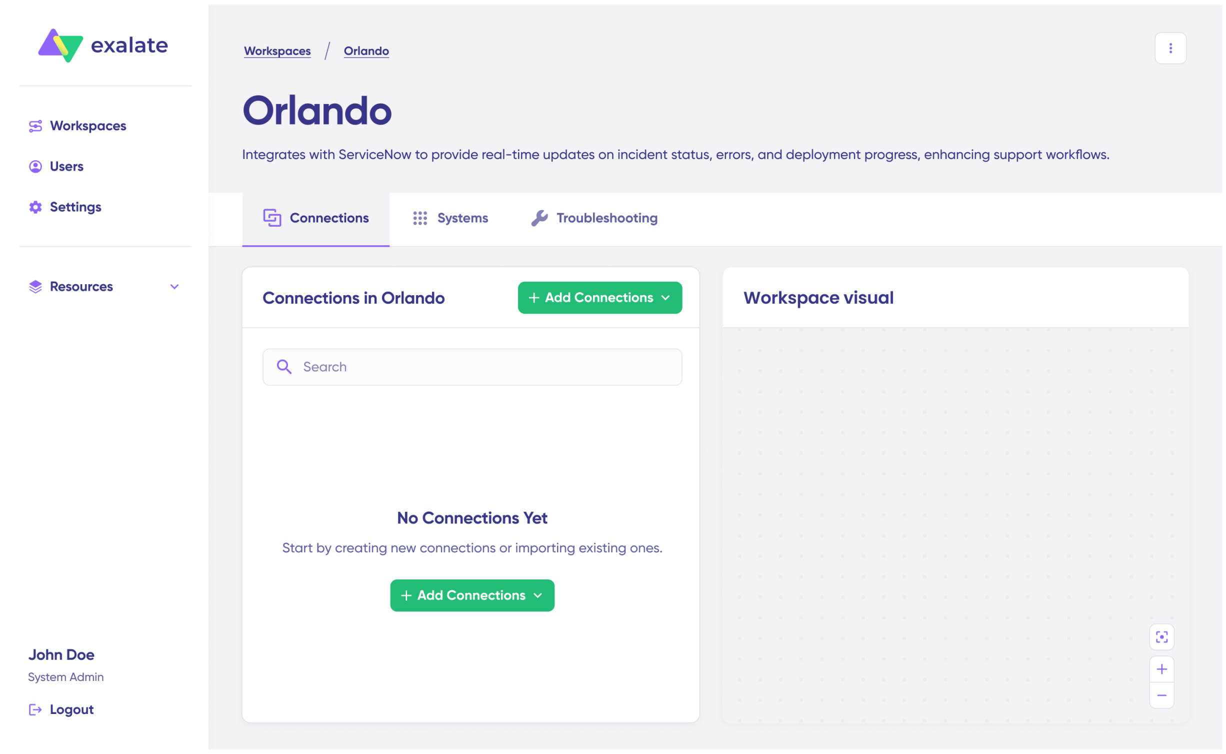
You can create one or more workspaces to organize and manage your Exalate nodes and connections efficiently. The same nodes can belong to multiple workspaces, but connections are restricted within a single workspace. A connection cannot exist in more than one workspace.



exalate
Workspaces
Users
Settings
Resources
John Doe
System Admin
Logout
Workspaces
My Workspaces
+ Create Workspace
Orlando
3
Description
Integrates with ServiceNow to provide real-time updates on incident status, errors, and deployment progress, enhancing support workflows.
View Workspace
Acquarius
6
Description
Works with Zendesk to facilitate containerized deployments, ensuring reliability and uniformity across all environments.
View Workspace
Cassiopeia
2
Description
Connects with Azure DevOps to streamline CI/CD, offering automated builds, comprehensive testing, and efficient deployment processes.
View Workspace

Import & Manage Connections

You can choose which connections to manage within your workspace. It is also possible to temporarily unmanage a connection in the new Exalate Console, returning its management to Exalate without removing it from the workspace. Similarly, you can regain management of the connection in the new Exalate console when needed.


Workspaces / Orlando

Orlando

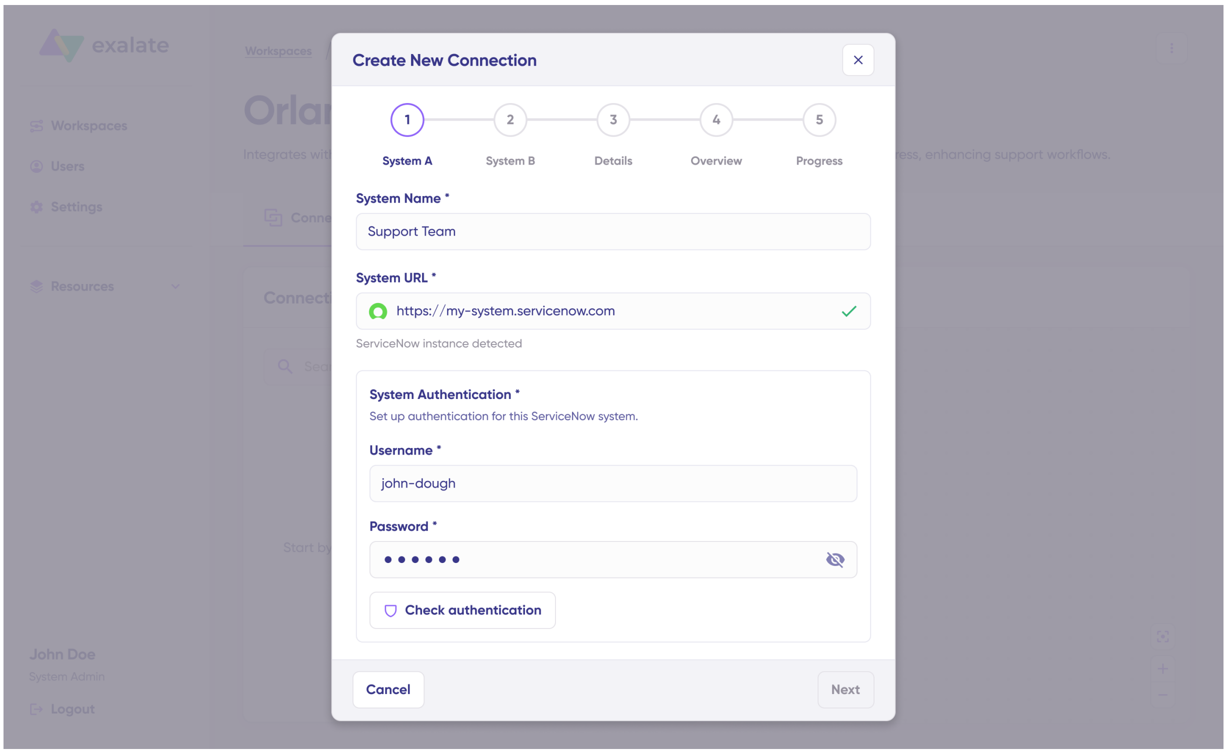
Integrates with ServiceNow to provide real-time updates on incident status, errors, and deployment progress, enhancing support workflows.

No Connections Yet

Start by creating new connections or importing existing ones.

Streamlined Connection Creation

Create connections between systems without manual node installation or accessing each system individually. A smooth, guided wizard experience connects two systems in minutes.


Create New Connection

System Name * Support Team

System URL * https://my-system.servicenow.com

ServiceNow instance detected

System Authentication *

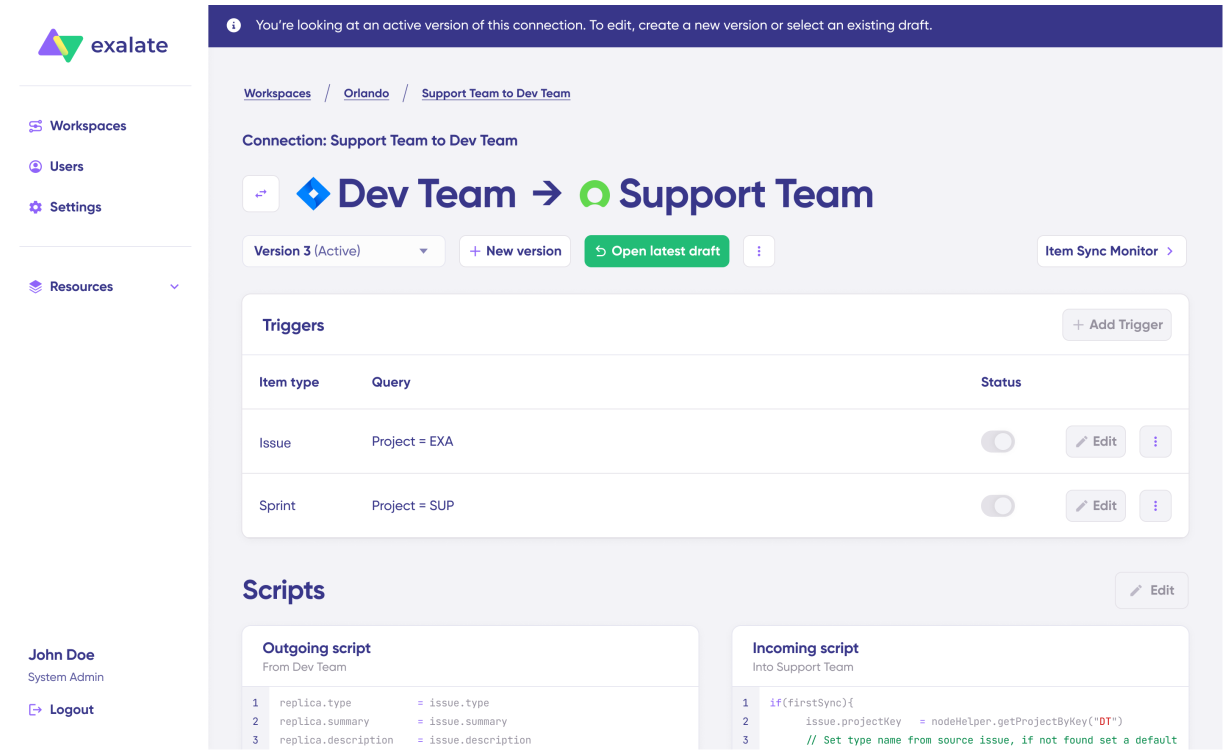
Set up authentication for this ServiceNow system.

Username * john-dough

Password * ••••••

Configure Scripts and Triggers with Versioning

You can create multiple configuration version drafts and publish a version from this Exalate Console. Each version consists of two scripts—an outgoing script from the source side and an incoming script for the destination side—along with a trigger. It is also possible to switch the data flow direction and create another version to manage the backflow separately.



You're looking at an active version of this connection. To edit, create a new version or select an existing draft.

Workspaces / Orlando / Support Team to Dev Team

Connection: Support Team to Dev Team

Dev Team → Support Team

Triggers

Item type	Query	Status
Issue	Project = EXA	
Sprint	Project = SUP	

Scripts

Outgoing script
From Dev Team

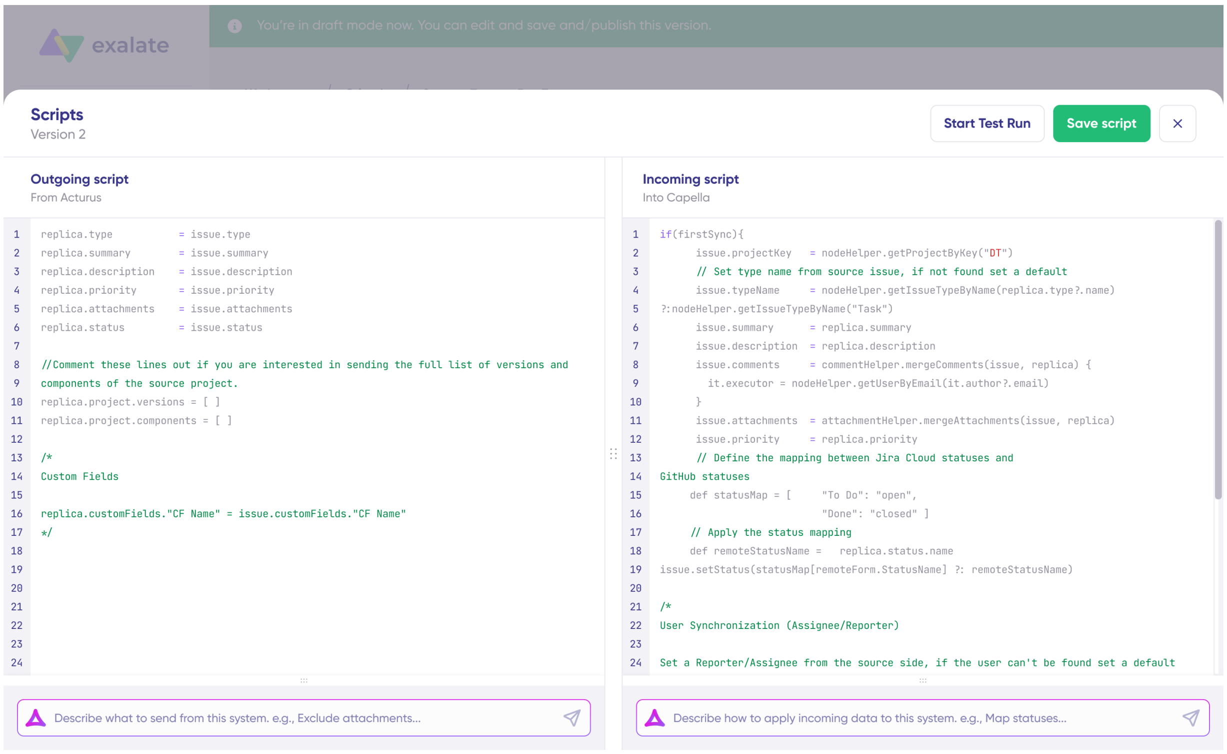
```
replica.type          = issue.type
replica.summary       = issue.summary
replica.description   = issue.description
```

Incoming script
Into Support Team

```
if(firstSync){
      issue.projectKey   = nodeHelper.getProjectByKey("DT")
      // Set type name from source issue, if not found set a default
```

Side-by-Side Script View

The outgoing script from the source side and the incoming script for the destination side are displayed side by side, making it easier to create, compare, and manage configurations.



You're in draft mode now. You can edit and save and/publish this version.

Scripts
Version 2

Outgoing script
From Acturus

```
replica.type          = issue.type
replica.summary       = issue.summary
replica.description   = issue.description
replica.priority      = issue.priority
replica.attachments   = issue.attachments
replica.status        = issue.status

//Comment these lines out if you are interested in sending the full list of versions and
components of the source project.
replica.project.versions = [ ]
replica.project.components = [ ]

/*
Custom Fields

replica.customFields."CF Name" = issue.customFields."CF Name"
*/
```

Describe what to send from this system. e.g., Exclude attachments...

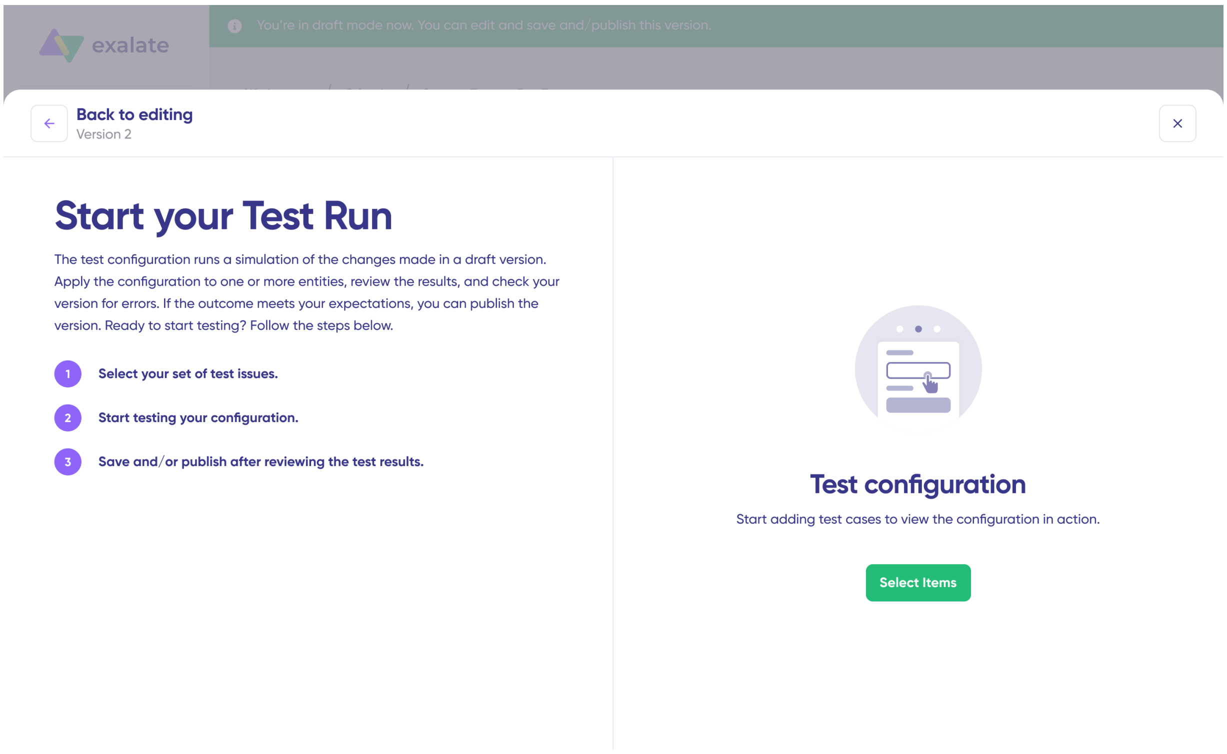
Incoming script
Into Capella

```
if(firstSync){
      issue.projectKey   = nodeHelper.getProjectByKey("DT")
      // Set type name from source issue, if not found set a default
      issue.typeName     = nodeHelper.getIssueTypeByName(replica.type?.name)
?:nodeHelper.getIssueTypeByName("Task")
      issue.summary      = replica.summary
      issue.description  = replica.description
      issue.comments     = commentHelper.mergeComments(issue, replica) {
        it.executor = nodeHelper.getUserByEmail(it.author?.email)
      }
      issue.attachments  = attachmentHelper.mergeAttachments(issue, replica)
      issue.priority     = replica.priority
      // Define the mapping between Jira Cloud statuses and
GitHub statuses
     def statusMap = [     "To Do": "open",
                           "Done": "closed" ]
     // Apply the status mapping
     def remoteStatusName =   replica.status.name
issue.setStatus(statusMap[remoteForm.StatusName] ?: remoteStatusName)

/*
User Synchronization (Assignee/Reporter)

Set a Reporter/Assignee from the source side, if the user can't be found set a default
```

Describe how to apply incoming data to this system. e.g., Map statuses...

Test Run Your Configuration

You can TestRun your scripts without creating an entity on the other side and compare the replica before making changes in production.



exalate

You're in draft mode now. You can edit and save and/or publish this version.

Back to editing
Version 2

Start your Test Run

The test configuration runs a simulation of the changes made in a draft version. Apply the configuration to one or more entities, review the results, and check your version for errors. If the outcome meets your expectations, you can publish the version. Ready to start testing? Follow the steps below.

1. Select your set of test issues.
2. Start testing your configuration.
3. Save and/or publish after reviewing the test results.

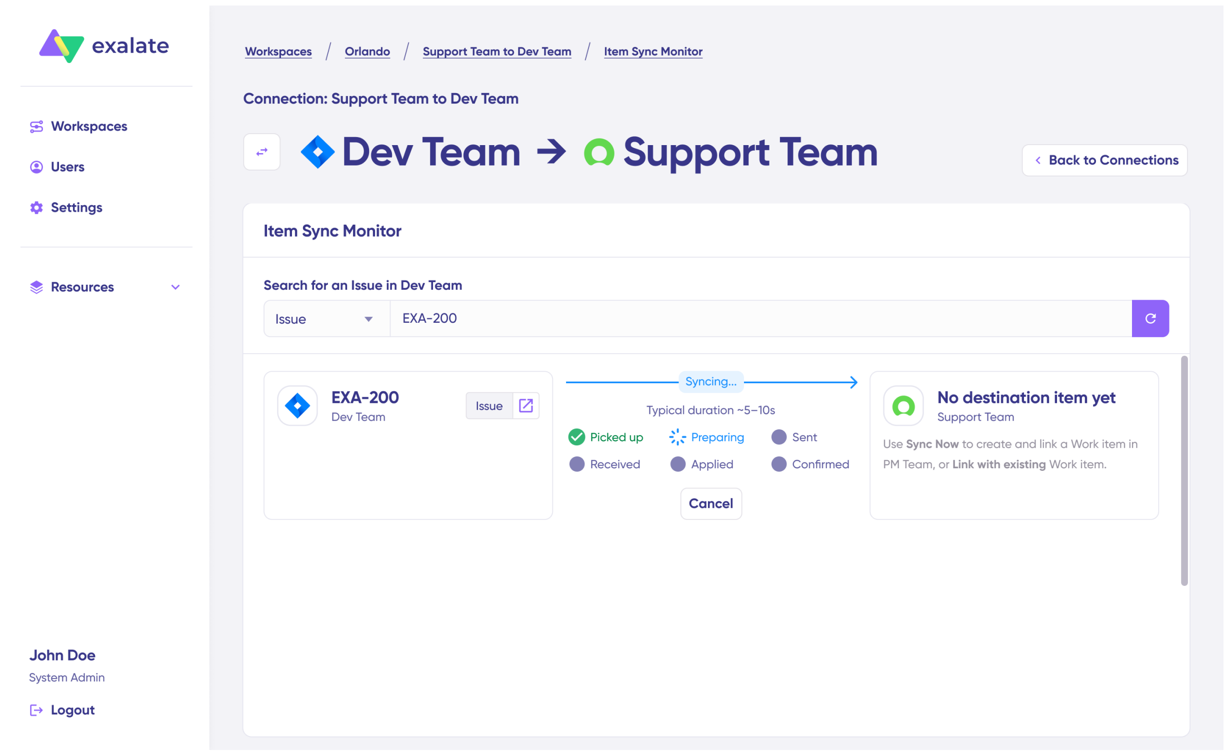
Test configuration

Start adding test cases to view the configuration in action.

Select Items

Monitor Sync Status and synchronize manually

Check item sync status, Synchonize or Connect entities manually through the Item Sync Monitor.



exalate

Workspaces
Users
Settings
Resources

Workspaces / Orlando / Support Team to Dev Team / Item Sync Monitor

Connection: Support Team to Dev Team

Dev Team → Support Team

Back to Connections

Item Sync Monitor

Search for an Issue in Dev Team

Issue | EXA-200

EXA-200
Dev Team
Issue

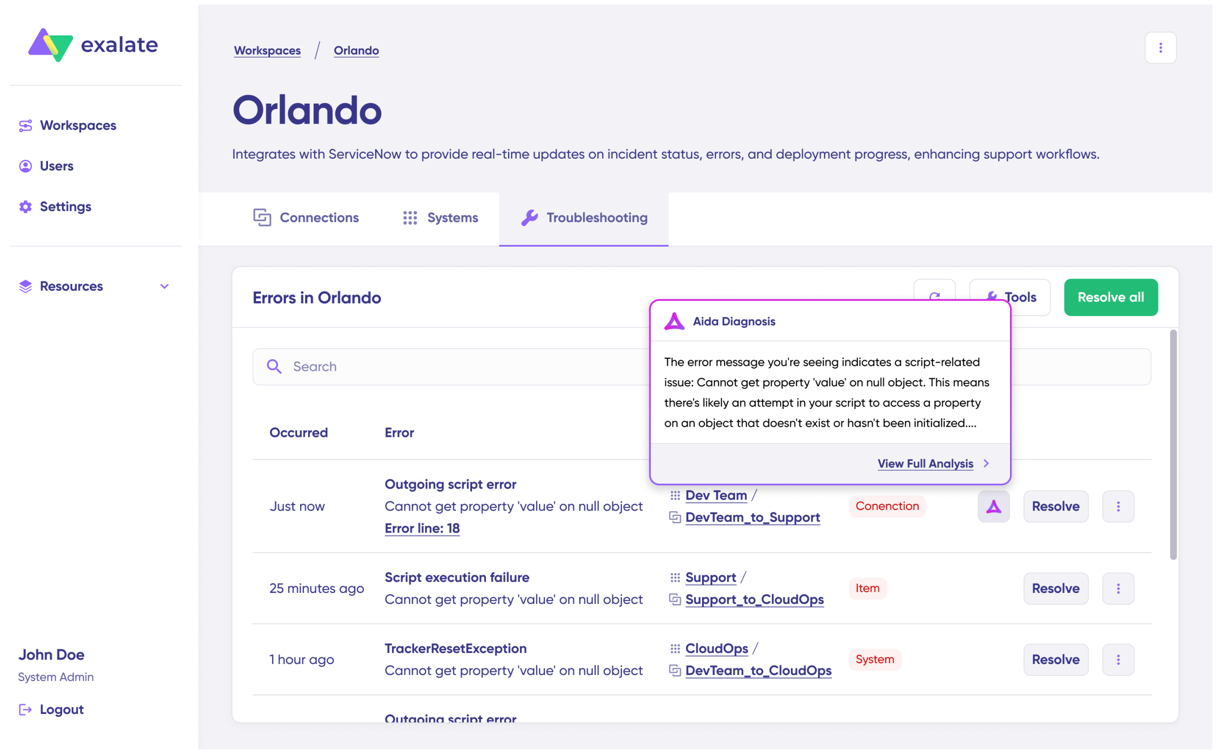
Syncing...
Typical duration ~5–10s
✓ Picked up · Preparing · Sent
Received · Applied · Confirmed
Cancel

No destination item yet
Support Team
Use **Sync Now** to create and link a Work item in PM Team, or **Link with existing** Work item.

John Doe
System Admin
Logout

AI-Powered Scripting and Troubleshooting

Get intelligent help writing sync scripts and resolving errors with AIDA, your AI assistant built into a new Exalate Console.



Orlando

Integrates with ServiceNow to provide real-time updates on incident status, errors, and deployment progress, enhancing support workflows.

Errors in Orlando

Aida Diagnosis

The error message you're seeing indicates a script-related issue: Cannot get property 'value' on null object. This means there's likely an attempt in your script to access a property on an object that doesn't exist or hasn't been initialized....

View Full Analysis

Occurred	Error				
Just now	Outgoing script error Cannot get property 'value' on null object Error line: 18	Dev Team / DevTeam_to_Support	Conenction	Resolve	⋮
25 minutes ago	Script execution failure Cannot get property 'value' on null object	Support / Support_to_CloudOps	Item	Resolve	⋮
1 hour ago	TrackerResetException Cannot get property 'value' on null object	CloudOps / DevTeam_to_CloudOps	System	Resolve	⋮
	Outgoing script error				

Next: Check the current **Limitations**.

ON THIS PAGE