watermelon software

Onboarding Guide

Confidential. Information proprietary to Watermelon Software Inc. Do not copy or distribute.

Getting Started with WM-Meta

Initiating your journey with Watermelon and unlocking its full suite of features starts with setting up the foundational metadata about your organization and its applications. Watermelon seamlessly integrates with a wide array of third-party tools spanning the Software Development Life Cycle (SDLC) that your organization might already be utilizing.

Watermelon offers a straightforward, self-service portal for the creation and management of this crucial metadata. This metadata forms the backbone of all Watermelon modules, enabling a cohesive and efficient experience across the platform for your organization.

Upon logging into Watermelon, direct your attention to the WM-Meta section on the landing page. This area acts as your portal to beginning the onboarding process and discovering the capabilities that Watermelon brings to the table. By navigating through WM-Meta, you'll set the stage for deploying Watermelon's solutions, tailored to your organizational needs and integrated with your existing SDLC tools, thereby maximizing the platform's benefits and impact.

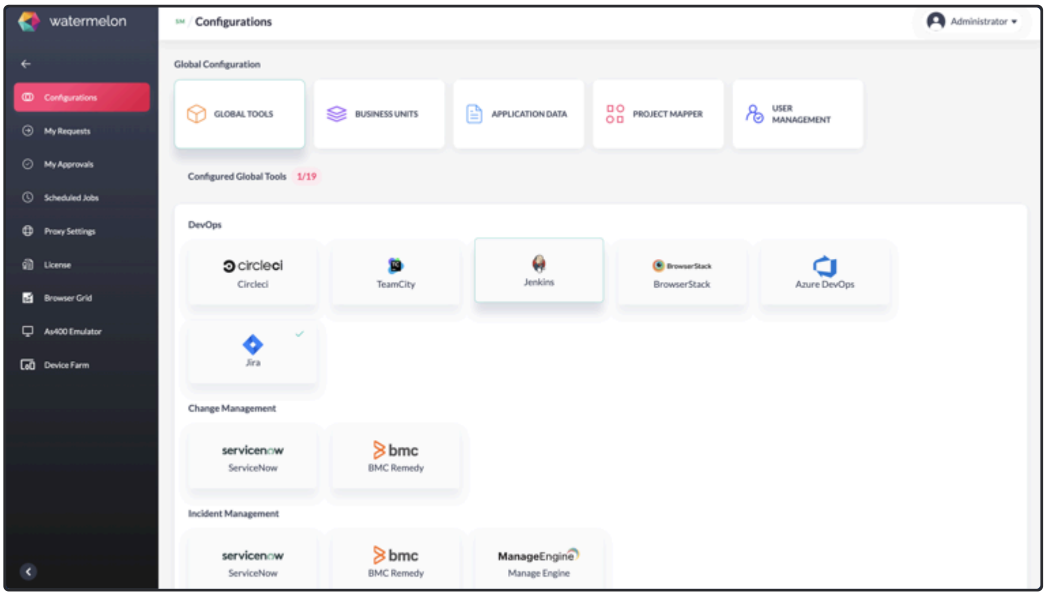
Upon accessing WM-Meta within Watermelon, you'll be greeted with a screen that offers a range of actions and configurations tailored to suit your organizational and application-specific needs. It's important to note that access to various functionalities within this section is governed by Role-Based Access Control (RBAC), ensuring that sensitive operations are safeguarded and only accessible to authorized personnel.

Key points to remember about access and actions within WM-Meta include:

- **SYSADMIN Role Restrictions**: The majority of global actions within WM-Meta are reserved for users with the SYSADMIN role. This role is designated during the initial setup phase, and it's recommended that the credentials for this high-level access are securely stored in a vault.

- **Application-Specific Permissions**: Certain actions within WM-Meta are specifically authorized for Application Owners and Application Administrators, providing them with the necessary permissions to manage their respective applications without compromising overall system security.

For those looking to delve deeper into how RBAC is implemented and functions within Watermelon, a comprehensive guide and detailed explanations are available in the designated section of the Watermelon documentation, accessible at Role Based Access Control.



This structured approach to access control ensures that the platform remains secure while offering flexibility and autonomy to users in managing their applications, all of which are detailed in their respective sections within the WM-Meta interface.

Configurations

This section serves as a comprehensive guide to the various configurations available in Watermelon under WM-Meta. It is designed to provide users with a self-service capability for managing ecosystem integrations across different Watermelon modules. The key configurations covered in this section include:

1. Global Tools Configuration: Users can configure global tools such as CI/CD tools, ALM tools, Device Farms and User Management tools that are utilized across multiple modules within Watermelon. This ensures seamless integration and interoperability between different tools and platforms.
2. Creating Business Units and Applications: Users can create and manage Business Units (BUs) and applications within Watermelon to organize and categorize their systems and applications effectively. This allows for clear access control and segregation of duties based on organizational structure.
3. Mapping Projects from ALM Systems: Integration with Application Lifecycle Management (ALM) systems allows users to map projects and artifacts from their ALM systems to corresponding applications within Watermelon. This facilitates traceability and collaboration across different stages of the software development lifecycle.
4. Data Sources Integration: Watermelon supports integration with various data sources, including APM tools, log management tools, proprietary databases, and cloud technologies. Users can configure data sources to ingest relevant data into Watermelon for monitoring, analysis, and reporting purposes.
5. User Access Management (RBAC): Role-Based Access Control (RBAC) allows users to define and manage roles and permissions within Watermelon. This ensures that users have appropriate access levels based on their roles and responsibilities within the organization.
6. License Management: Users can manage licenses for Watermelon modules and features, ensuring compliance and optimal utilization of resources.

By providing a centralized platform for configuring and managing these settings, WM-Meta streamlines the administration and maintenance of Watermelon deployments, enabling it to be seamlessly integrated within your organization landscape/ecosystem.

Global Tools

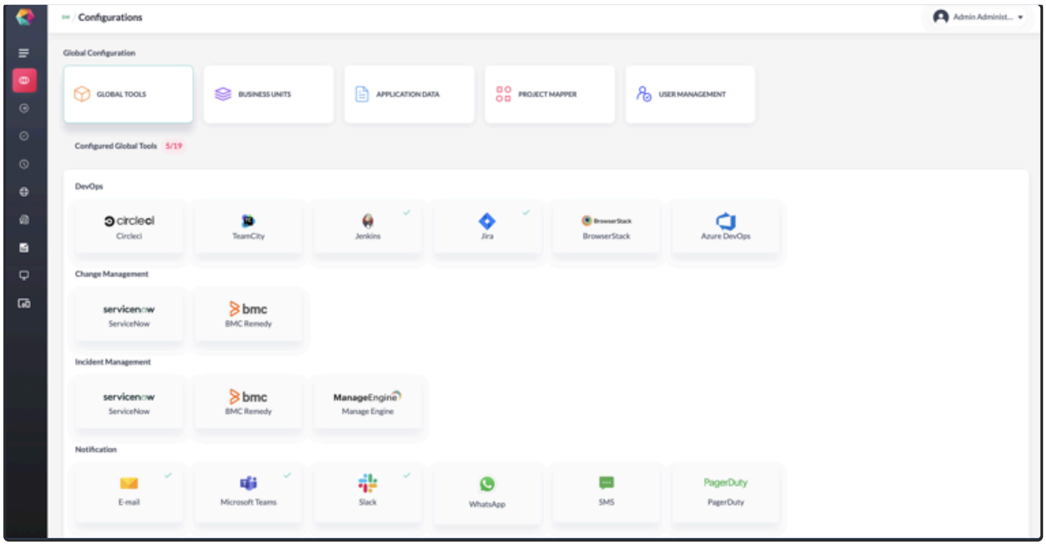
This section outlines the platform-wide integrations you can set up in Watermelon through WM-Meta, which, once configured, become accessible across all Watermelon modules. These integrations are designed to be global in scope, ensuring a unified application of tools and services throughout the platform. Given the breadth of access and control these integrations provide, configuring them typically requires elevated privileges, specific to each tool's requirements. Detailed guidance on setting up and managing these integrations for supported tools is provided further in this documentation.

Global Configurations Overview

Access to the global configurations page is primarily restricted to users holding the Watermelon SYSADMIN role. This role is equivalent to a root user, endowed with comprehensive access and control over the platform's configurations and settings. This centralized approach to integration management ensures that SYSADMIN users can efficiently oversee and adjust global settings as needed, promoting consistency and security across the platform.

For more intricate details about user access levels and the Role-Based Access Control (RBAC) system employed by Watermelon, refer to the dedicated section on RBAC in the Watermelon documentation. This resource offers an in-depth look at how roles and permissions are structured within Watermelon, enabling a secure and organized management framework for all users.

The implementation of these global integrations and the RBAC framework underscores Watermelon's commitment to providing a secure, scalable, and efficient environment for managing service level objectives and other critical operations.



Global configurations view

User Management Tools

This section outlines how Watermelon can integrate with your organization's Active Directory or other Identity Provider (IDP) providers. Watermelon utilizes Keycloak for authentication, which seamlessly integrates with numerous IDP providers supporting SAML 2.0, OIDC 1.0, and OAuth 2.0 protocols.

More documentation on Keycloak integration with IDP providers can be found at Server Administration Guide (subject to change without notice). If your organization requires Watermelon to integrate with other IDP provides (for example, Google), please reach out to your Watermelon representative or support.

Active Directory Integration

Overview:

Watermelon internally is deployed with KeyCloak with an integration capability with Active Directory for user federation and role based management. This allows users to authenticate using their AD credentials. Below is a concise write-up describing the integration process.

Integration Steps:

1. Setup Keycloak Server:

- Before integration, ensure that the Keycloak server is properly installed and configured. This is done automatically (by default) as part of the Watermelon installation.

2. Create a Realm:

- Log in to the Keycloak Admin Console and create a new Realm or use the existing one, depending on your needs. This is done automatically (by default) as part of the Watermelon installation.

3. Configure User Federation:

- Under Watermelon Meta in the Global Tools section, the following mandatory fields can be submitted to configure the integration.
    - **Connection URL:** `ldap://ad.example.com` or `ldaps://ad.example.com` for secure LDAP.
    - **Users DN:** Distinguished name of the tree containing users, e.g., `CN=Users,DC=example,DC=com`.
    - **Bind DN:** Distinguished name of the user to connect to AD, e.g., `CN=binduser,CN=Users,DC=example,DC=com`.
    - **Bind Credential:** Password for the bind user.

4. Test the Connection:

- After saving the configuration, test the connection to ensure that Keycloak can communicate with the Active Directory server.
- You may use the “Test connection” and “Test authentication” options in Keycloak User Federation settings to validate the setup or try logging in as an Active Directory user into the Watermelon UI.

It's important to be aware that integration with Active Directory is not facilitated through the WM-Meta user interface but is instead configured via the Keycloak interface during the initial setup phase. The integration process is designed to ensure seamless authentication and authorization across the Watermelon platform by leveraging your organization's Active Directory infrastructure.

During the initial setup phase of the Watermelon platform, the Watermelon support teams are available to assist with the integration process, ensuring a smooth and secure connection to your Active Directory. Once this initial setup is complete, the maintenance and any further configuration adjustments fall under the responsibility of your organization's Watermelon SYSADMIN. This role has the necessary privileges to manage the integration settings, ensuring that your organization can efficiently control access and permissions within the Watermelon platform, in alignment with your internal security policies and user management practices.

Okta integration

Integrating Okta with Keycloak for authentication in the Watermelon platform offers several advantages, including enhanced security and streamlined user access management. Here are the steps to seamlessly integrate Okta with Keycloak:

Add SAML identity provider in Watermelon Keycloak

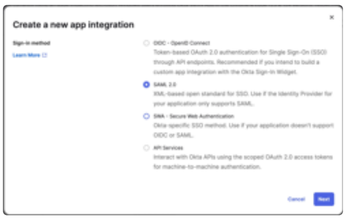
Open Watermelon Keycloak admin page, click **Identity Providers** and select **SAML v2.**0 provider from the list of providers.

Configure SAML provider in Watermelon Keycloak

Enter the **Alias**. Notice that it is part of **Redirect URI.**

Add SAML application in Okta

Select "SAML 2.0" and click Create.

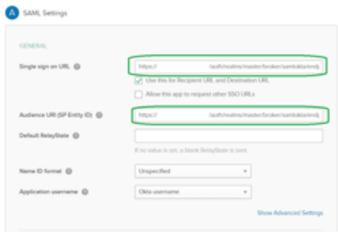


Provide the application name

Enter the application name and click Next

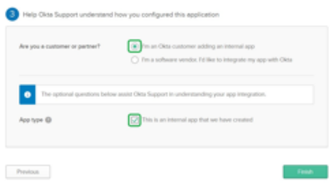
Configure SAML Settings

Configure SAML Settings by copying the Keycloak's **Redirect URI** from the SAML v2.0 provider page of Keycloak to **Single sign on URL** and **Audience URI (SP Entity ID)** settings. Click **Next**.



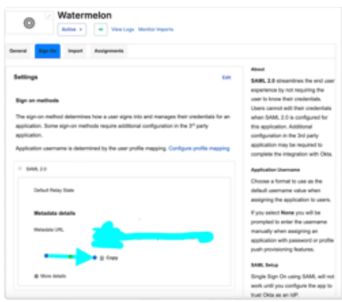
Configure the application type

Configure the application type by completing the fields as indicated below. Click **Finish**



Copy the metadata link

From the menu, click Sign On configuration for the application you are working on. Hover over Identity Provider metadata link, right-click and select from the menu Copy link.

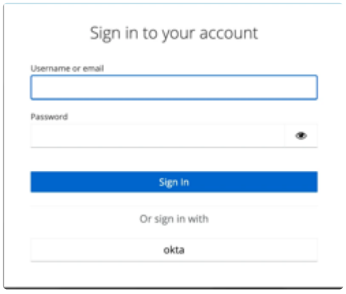


Import Okta Identity Provider Metadata into Watermelon Keycloak

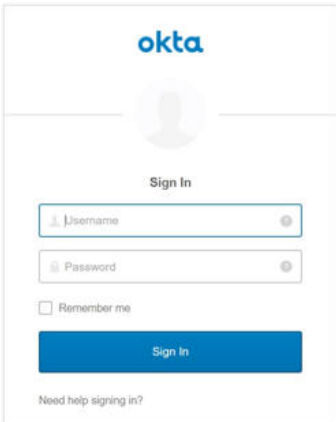
Open the **Identity Providers** configuration and paste the metadata link value into the **SAML entity descriptor** area and click the **Add** button.

Login using Okta SAML

You open the Watermelon login page. We have the additional button that allows us to login to Watermelon using the Okta SAML provider:



Click on the button and you will be redirected to the Okta SAML provider for the authentication.



After the successful authentication you will be redirected back to Watermelon.

As Watermelon leverages Keycloak, it provides capabilities to seamlessly integrate with other IDP providers. More documentation on Keycloak integration with IDP providers can be found at Server Administration Guide (subject to change without notice). If your organization requires Watermelon to integrate with other IDP provides (for example, Google), please reach out to your Watermelon representative or support.

Watermelon Software Inc.

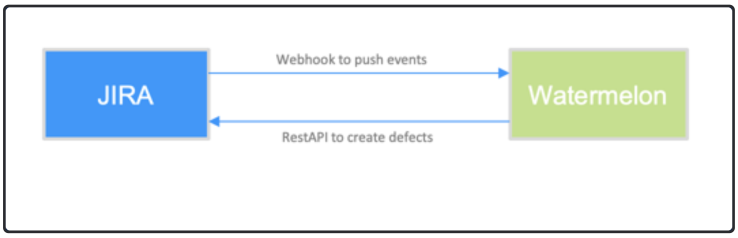
DevOps Tools

Confidential. Information proprietary to Watermelon Software Inc. Do not copy or distribute.

JIRA Integration

Overview

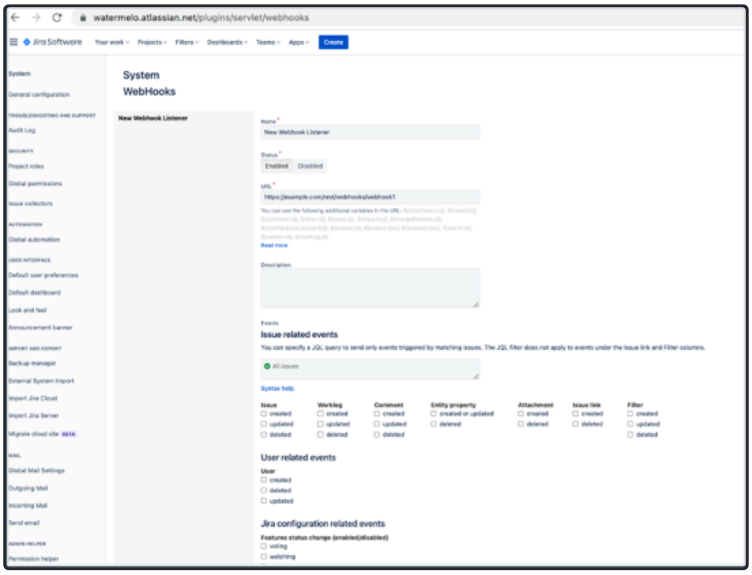
- Watermelon supports a two-way integration with JIRA. Events from JIRA flow into Watermelon in near-real time Webhooks.
- Reverse integration to JIRA is done via RestAPI to create Defects and populate fields within JIRA based on customer needs.



Webhook Setup

- We leverage standard JIRA webhooks setup to send events back to Watermelon. There is no specific customization here and it’s an out of the box JIRA feature.

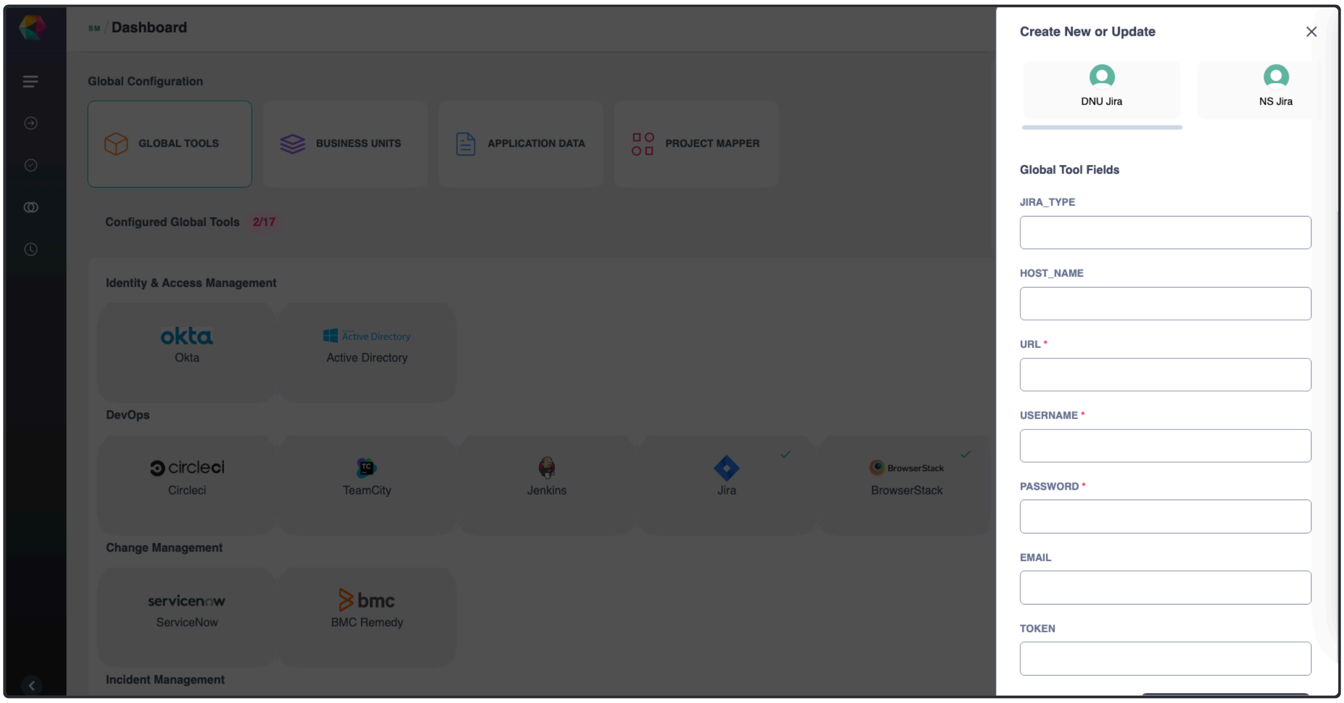
Official documentation: Webhooks



Configuring JIRA to interact with Watermelon

- JIRA can be configured under Global Tools (Watermelon Meta). Multiple JIRA instances can be configured as there could be multiple instances within an organization/BU and Watermelon provides this flexibility.

- You will need to enter the details given in the screenshot below to configure a JIRA integration.
- The URL, username and password are compulsory fields to have a valid integration.



- Watermelon accommodates the complexity and diversity of organizational structures by supporting the configuration of multiple Jira instances. This feature recognizes that different entities or departments within an organization may operate their own Jira instances for project management and tracking.

Configuring a new JIRA instance within Watermelon

1. Click the “New” button in the relevant section.
2. Fill in the details as instructed in the documentation to complete the configuration.

- These individually configured Jira instances can then be precisely associated with specific applications during the application onboarding process in Watermelon. It's important to note that each application can be linked to only one Jira instance at any given time, ensuring clear and manageable integration points.
- For comprehensive guidance on the application onboarding process and how to link Jira instances to your applications within Watermelon, please visit the dedicated section in the Watermelon documentation Onboarding Applications.
- Additionally, Watermelon's integration capabilities extend to automatically discovering all projects within your Jira instance(s). These projects can then be systematically associated

with specific applications through Watermelon's Project Mapper feature. This allows for a seamless alignment between your project management activities in Jira and the corresponding applications in Watermelon, enhancing the coherence and effectiveness of your SLO management processes.

- To dive deeper into how to leverage the Project Mapper for aligning Jira projects with Watermelon applications, please consult the Project Mapper section of the documentation Project Mapper
- This multi-instance support and project mapping functionality underscore Watermelon's flexibility and adaptability, enabling organizations to tailor the platform's integrations to fit their unique operational landscape and project management needs.

Integration Capabilities

- Two-way synchronization for JIRA issue types including Epics, Stories (User Stories), Tasks, Bugs (Defects), and Sub-tasks.
- Ability to create JIRA stories directly from WM for streamlined issue creation.
- Ability to log defects directly to JIRA from Watermelon.
- Updates and status changes performed within JIRA are automatically reflected in WM to maintain consistent information.
- One-to-one project mapping between a WM project and a JIRA project ensures clear data association.
- Watermelon recommends managing the full issue lifecycle within JIRA, with WM reflecting synchronized updates to keep both systems aligned.

Integration Constraints

- After creation from WM, JIRA issues cannot be edited within WM; modifications must be made directly in JIRA. User Stories synchronized from JIRA must also be edited only in JIRA as it is the source of truth.
- Test cases cannot be imported or exported between JIRA and WM at this time.
- Synchronization covers standard JIRA issue types and fields. Any custom issue types, which are user-defined issue categories in JIRA, will also be supported provided they are configured correctly within the integration. This allows flexibility to include unique or organization-specific item types beyond the default set.

JIRA Issue Types Included

- Epic: Large workstreams made up of multiple smaller issues.
- Story (User Story): Features or requirements from an end-user perspective.

- Task: Discrete, standalone work items or supporting activities.
- Bug (Defect): Issues identifying problems or errors needing resolution.
- Sub-task: Detailed, smaller parts of larger issues.
- Custom Issue Types: Organization-specific issue categories created within JIRA to tailor tracking to unique workflows or project needs.

This integration ensures flexibility by leveraging JIRA's built-in issue hierarchy with support for custom extensions, enabling effective and comprehensive project tracking within Watermelon.

Confidential. Information proprietary to Watermelon Software Inc. Do not copy or distribute.

Watermelon Software Inc.

Notification Tools

Confidential. Information proprietary to Watermelon Software Inc. Do not copy or distribute.

Email Integration

Simple Mail Transfer Protocol (SMTP) integration is generally used to send emails programmatically.

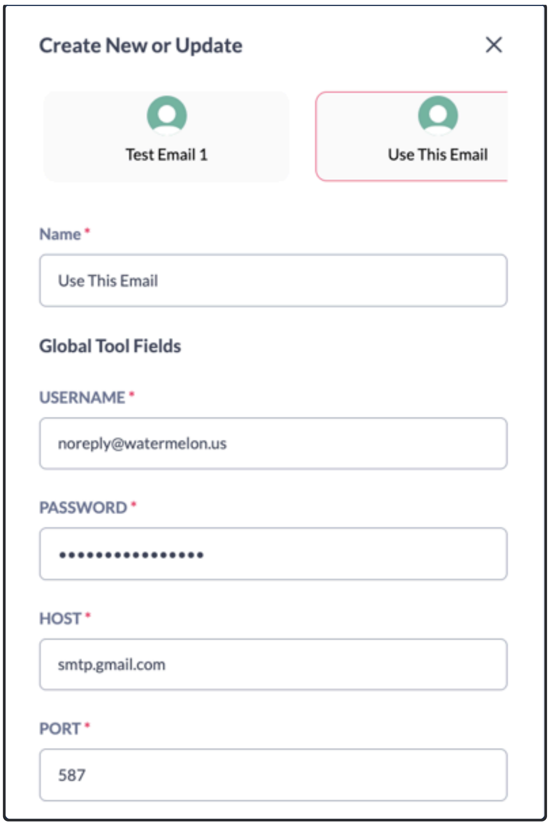
Watermelon supports configuring SMTP servers for **Gmail, Office365, SendGrid, and other SMTP providers**, ensuring flexibility for various email delivery needs. This allows seamless integration with different email services based on organisational requirements.

When integrating SMTP with an application or a system, several pieces of information (fields) are typically required. Here are the general fields you would need to set up SMTP integration:

1. **SMTP Server Address (or Host):**
   - This is the address of the SMTP server that will be used to send emails. For example, `smtp.example.com`.
2. **SMTP Username:**
   - This is the username required to authenticate with the SMTP server. Often, it is an email address.
3. **SMTP Password/Token:**
   - This is the password/token required to authenticate with the SMTP server. It is associated with the username.
4. **SMTP Port:**
   - This is the port number used to connect to the SMTP server. Common ports include 25 (unencrypted), 587 (encrypted for client-to-server communication), and 465 (SSL/TLS).

SMTP can be configured under Global Tools under Watermelon Meta by providing the necessary fields to enable seamless email notifications from the platform.

1. **Navigate to WM-Meta and Email under Global Tools:** This is where you'll be registering the SMTP servers. Please note that more than one SMTP server can be configured by clicking the “New” button.
2. **Application Mapping:** These registered SMTP servers can then be mapped to specific applications as part of the onboarding process. Watermelon will attempt email delivery via these SMTP servers mapped against each application. For more details on application onboarding, please refer to Onboarding Applications.


Create New or Update
Test Email 1
Use This Email
Name *
Use This Email
Global Tool Fields
USERNAME *
noreply@watermelon.us
PASSWORD *
HOST *
smtp.gmail.com
PORT *
587

Slack Integration

To integrate Slack with Watermelon at a granular level through the configuration of distinct webhooks, you'll follow a two-part process: setting up an application on Slack and then registering those webhooks within Watermelon. Here's a step-by-step guide to navigate through both parts:low.

Actions on Slack

1. Create a Slack App

- **Go to Your Apps**: Navigate to https://api.slack.com/apps and sign in if necessary.
- **Create New App**: Click on "Create New App", choose "From scratch", and give your app a name.
- **Select Your Workspace**: Choose the Slack workspace where you want to install your app and click "Create App".

2. Enable Incoming Webhooks

- **Go to Incoming Webhooks**: In your app settings, find the "Incoming Webhooks" feature.
- **Activate Incoming Webhooks**: Switch the toggle to activate Incoming Webhooks.

3. Create an Incoming Webhook

- **Add New Webhook**: Click on "Add New Webhook to Workspace".
- **Pick a Channel**: Choose the channel or user you want the notifications to be sent to and click "Allow". This will create a Webhook URL for you. Note this URL as you will use it to send notifications.

4. Install App to Workspace

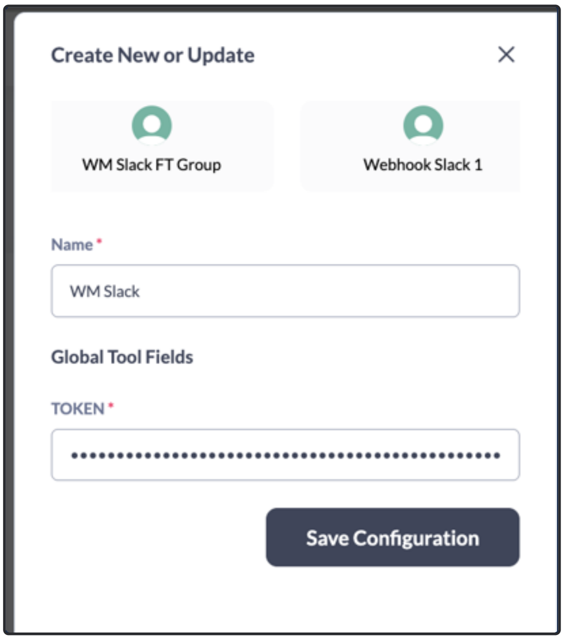
- **Install Your App**: To start sending notifications, you might need to install your app to your workspace. Go to "OAuth & Permissions" in the app settings and click "Install to Workspace".

Please note these steps are subject to change so please refer to the official documentation for Slack in case of any deviations.

Actions on Watermelon

After setting up the Slack side of things, you'll move to integrate these webhooks within Watermelon:

1. **Navigate to WM-Meta and Slack under Global Tools:** This is where you'll be registering the webhooks you created on Slack.
2. **Register Webhooks:** Assign each webhook a unique name for easy identification.
3. **Application Mapping:** These registered webhooks can then be mapped to specific applications as part of the onboarding process. For more details on application onboarding, please refer to Onboarding Applications.



This approach ensures a granular level of notification control via Slack for various applications managed within Watermelon. By meticulously following these steps, users can achieve a seamless integration between Slack for communication and notifications, and Watermelon for application management and operational efficiency.

MS Teams Integration

Integrating notifications into Microsoft Teams via webhooks involves creating an incoming webhook in a Teams channel and then using that webhook to send messages from Watermelon. Here’s a step-by-step guide to achieve this:

Actions on MS Teams

Create an Incoming Webhook in Microsoft Teams

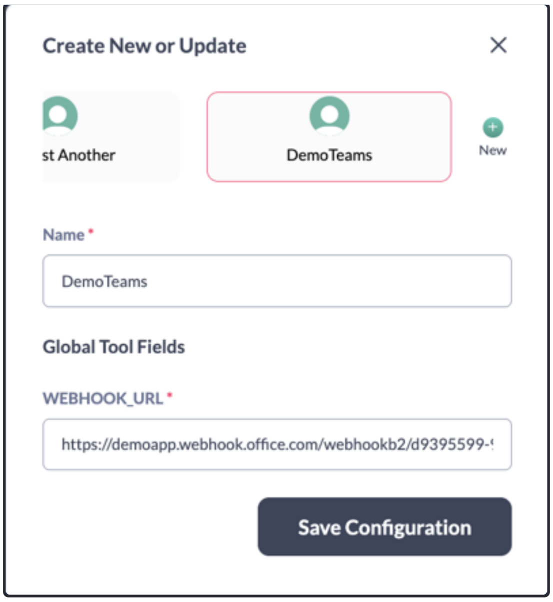
1. **Open Microsoft Teams**: Navigate to the team and channel where you want to add the webhook.
2. **Add a Connector**:
   - Click on the three dots next to the team name, then choose **More options** > **Connectors**.
   - Scroll through the list of connectors to **Incoming Webhook**, and click **Add**.
3. **Configure the Webhook**:
   - Give your webhook a name, and optionally, upload an image to help identify it.
   - Click **Create**. Teams will generate a URL for your webhook; make sure to copy and save this URL as it is needed to send messages to this channel.
4. **Finish Setup**:
   - After copying the URL, click **Done**. Your incoming webhook is now configured and ready to receive messages.

Please note these steps are subject to change so please refer to the official documentation for MS Teams in case of any deviations.

Actions on Watermelon

After setting up the MS Teams side of things, you'll move to integrate these webhooks within Watermelon:

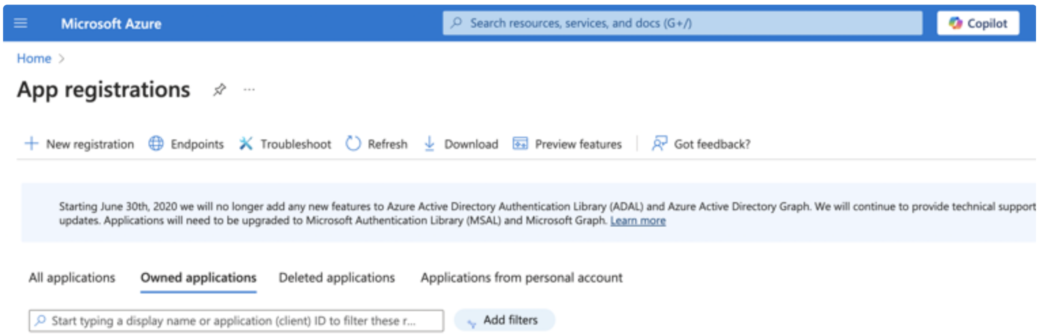
1. **Navigate to WM-Meta and Teams under Global Tools:** This is where you'll be registering the webhooks you created on Teams.
2. **Register Webhooks:** Assign each webhook a unique name for easy identification.
3. **Application Mapping:** These registered webhooks can then be mapped to specific applications as part of the onboarding process. For more details on application onboarding, please refer to Onboarding Applications.


Create New or Update
st Another
DemoTeams
New
Name *
DemoTeams
Global Tool Fields
WEBHOOK_URL *
https://demoapp.webhook.office.com/webhookb2/d9395599-
Save Configuration

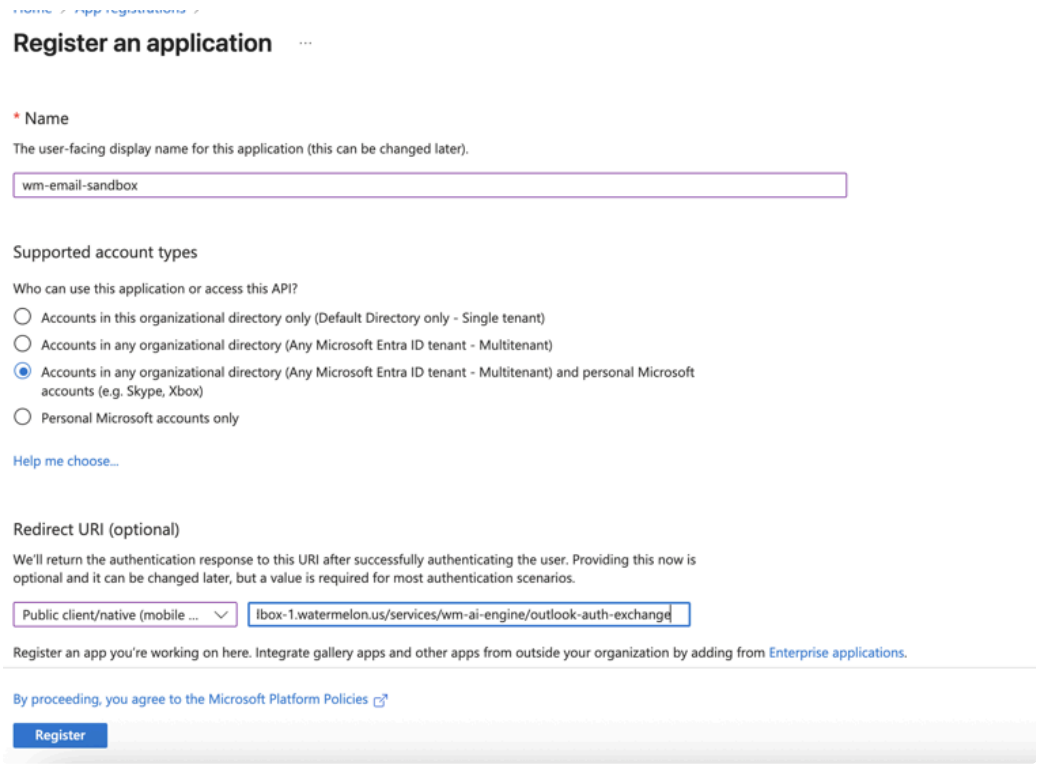
Azure Mail Integration

Configure Azure App Registration

- Login to Azure portal and search for “**App registrations**”
- Here we need to create a new app registration for “**Watermelon email**” and click on “**New registration**”

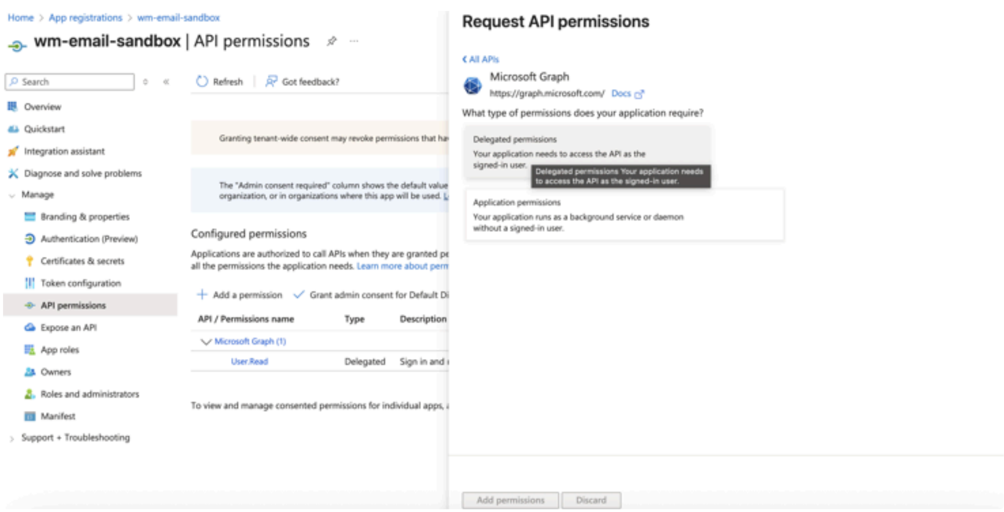


- Fill out the necessary details as below and click on “**Register**”.
  - **Name**: User-facing display name for this application e.g. Watermelon
  - **Supported account types**: Accounts in Any organisational directory (Multitenant) and personal microsoft accounts
  - **Redirect URI**: Choose “Public client/native (mobile & native”) and add URI field as “<watermelon_url>/services/wm-ai-engine/outlook-auth-exchange”

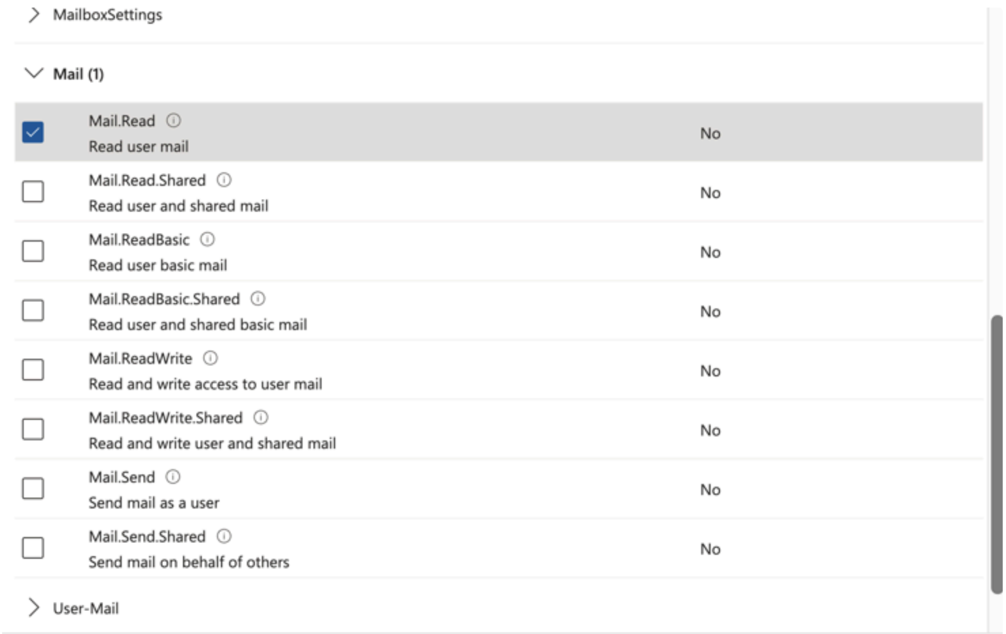


- On the app registration details page, note the of the **Application (client) ID** as this will be required in Watermelon read outlook email configuration.

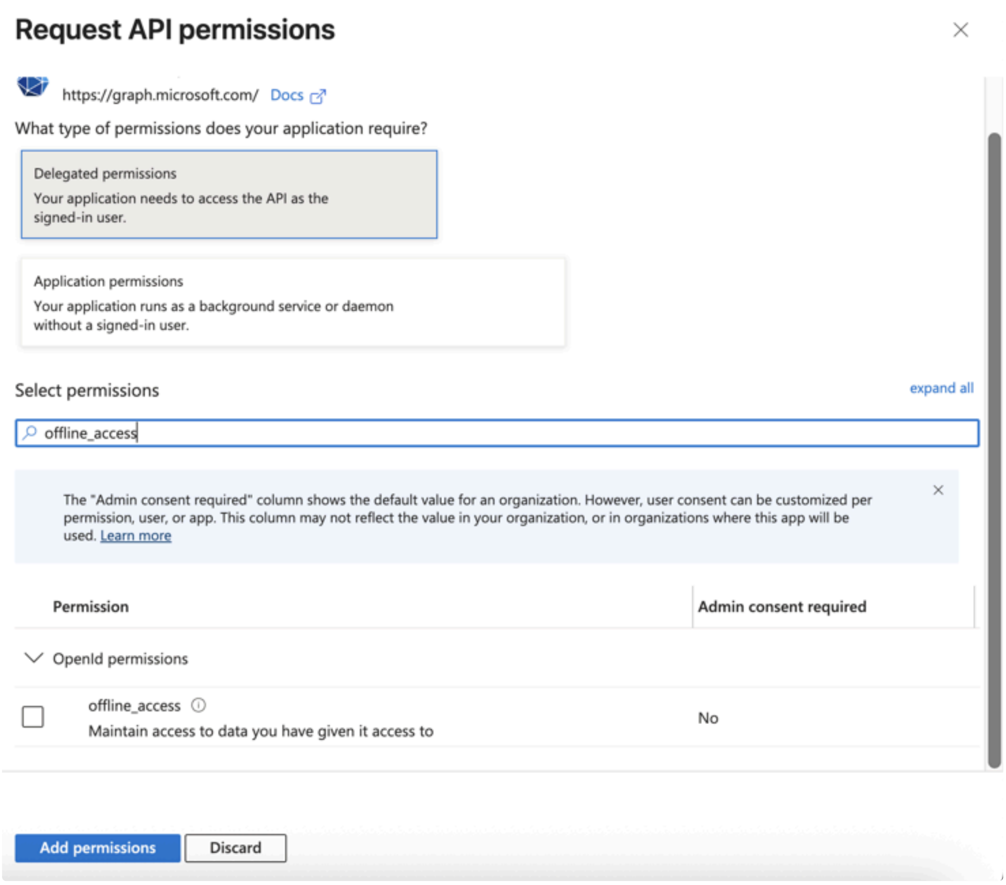
- Go to Manage > API Permissions > Add a permission and select microsoft graph Choose Delegated permissions



- Search for Mail and add Mail.Read permission



- Search for Offline and add offline_access permission



- Now you can use your block to read emails with the registered app client id

Confidential. Information proprietary to Watermelon Software Inc. Do not copy or distribute.

ITSM Tools

Confidential. Information proprietary to Watermelon Software Inc. Do not copy or distribute.

ServiceNow Integration

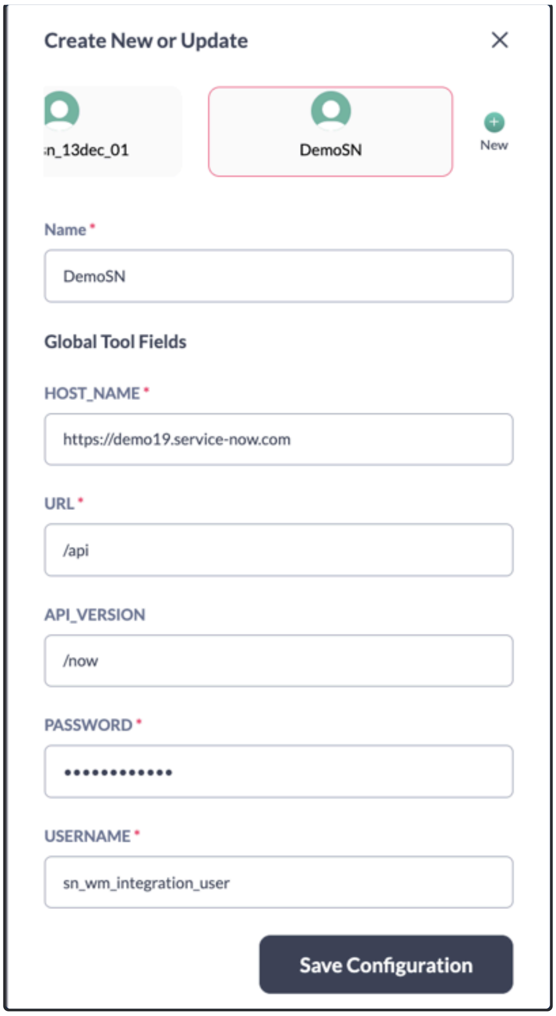
Integrating ServiceNow for Incident and Change Management into Watermelon involves setting up service accounts, configuring integration details in Watermelon, and mapping these configurations to specific applications. Here's a more detailed walkthrough based on the provided outline:

1. Prepare ServiceNow

- **Service Account Setup**: In ServiceNow, create a service account with permissions required for incident and change management operations. This typically includes read, write, and update permissions on relevant tables and records.
- **Note ServiceNow Details**: Ensure you have the following details handy for the integration setup:
  - **ServiceNow Hostname**: The base URL of your ServiceNow instance (e.g., `yourcompany.service-now.com`).
  - **API Version**: The version of the ServiceNow API you'll be using for integration.
  - **User ID**: The username of the service account.
  - **Password/Token**: The password or API token associated with the service account for authentication.

2. Configure ServiceNow Integration in Watermelon

- **Navigate to WM-Meta and Teams under Global Tools**: Access the Watermelon platform and locate the section for global tools where you can manage integrations.
- **Register ServiceNow Configuration**:
  a. **Add New Configuration**: Start by adding a new integration configuration for ServiceNow. You may add multiple instances as required.
  b. **Configuration Details**:
     - **Unique Name**: Assign a unique name to each configuration to easily identify them.
     - **ServiceNow Hostname**: Enter the ServiceNow hostname.
     - **URL**: Provide the full URL to the API endpoint if it's different from the hostname, including the API version if necessary.
     - **API Version**: Specify the API version being used.
     - **User ID**: Enter the service account's User ID.
     - **Password/Token**: Provide the password or token for authentication.



3. Application Mapping

- **Map to Applications**: With your ServiceNow instances configured, the next step is to map these configurations to specific applications within Watermelon. This mapping ensures that incidents and changes for each application are managed through the appropriate ServiceNow instance. Please refer to the application onboarding section Onboarding Applications for more details.

Watermelon Software Inc.

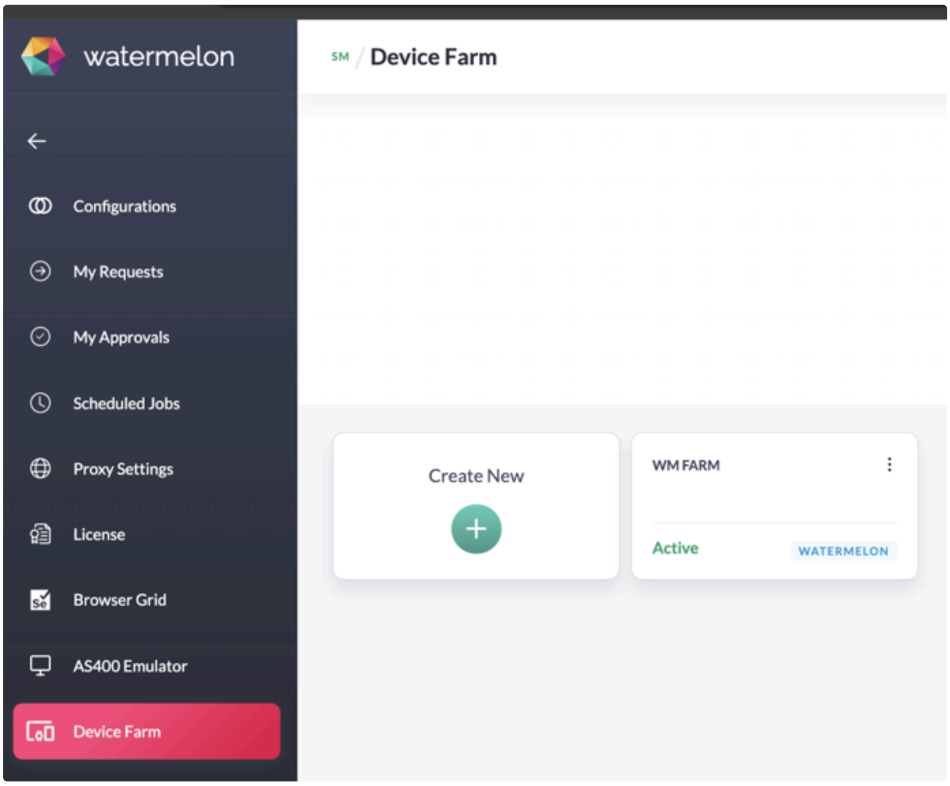
Device Farms/Browser Grids

Confidential. Information proprietary to Watermelon Software Inc. Do not copy or distribute.

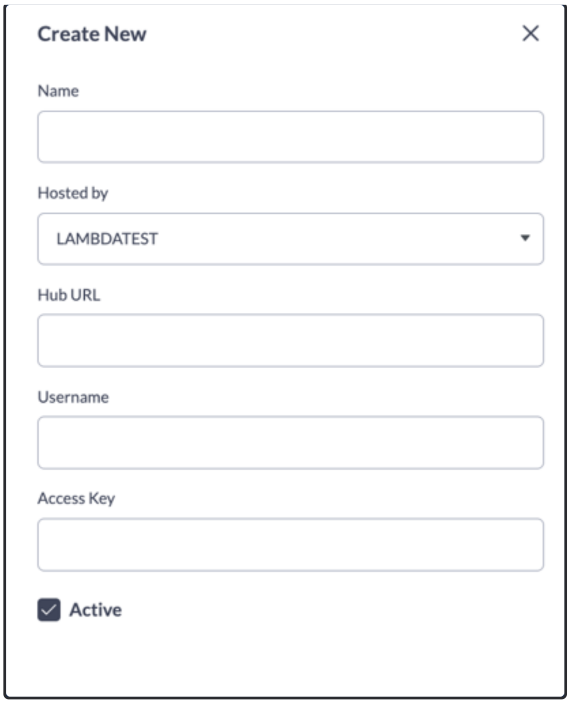
Device Farms (Mobiles and Browsers)

Watermelon supports commonly used Device Farms such as Browserstack. Lamdbatest and PCloudy. Most of these tools following a similar integration methodology and details such as Hub URL, User ID and Access Key/Token would be required.

1. **Navigate to Device Farms:** Access the Device Farms section under the left pane of WM-Meta.

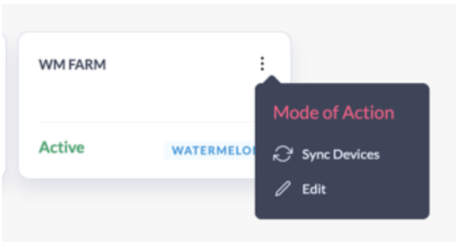


2. **Create New Integration:** Click on the "Create New" button to initiate the integration process.
3. **Select Device Farm:** From the "Hosted by" dropdown menu, choose the Device Farm you want to integrate with. Options may include Browserstack, LambdaTest, or pCloudy.
4. **Provide Necessary Details:** Enter the required details such as the Hub URL, User ID, and Access Key/Token for the selected Device Farm. Ensure that the provided credentials have a valid subscription on the Device Farm.



Device Farm configuration

5. **Complete Integration:** Once all necessary details are entered, proceed to complete the integration process. Ensure that the information provided is accurate and up-to-date to enable seamless integration with the chosen Device Farm.
6. Click the kebab menu on the configured device farm seen in the Device Farm repository and click “Sync Devices” to ensure the connectivity is tested and the available devices are visible to Watermelon.

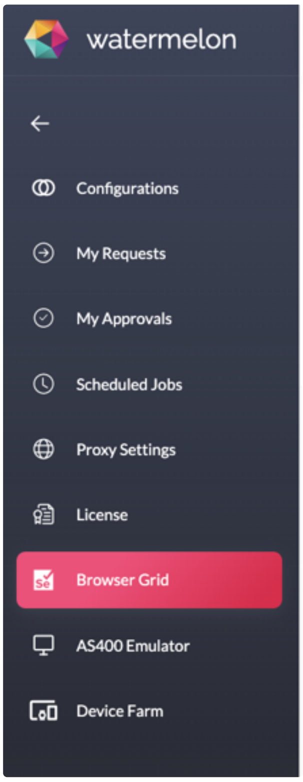


By following these steps, you can successfully integrate commonly used Device Farms with Watermelon, allowing for efficient testing and monitoring of applications across various devices and platforms.

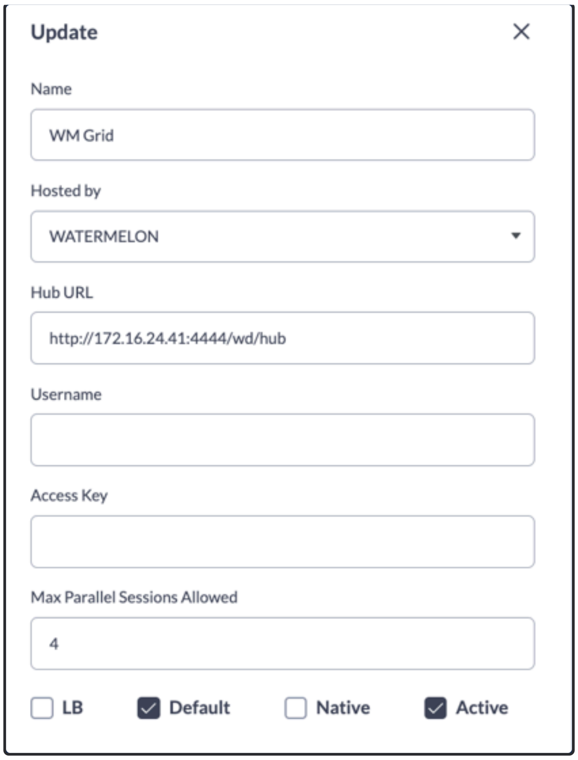
Watermelon Grid (Browsers)

Watermelon provides an out-of-the-box browser grid as part of its deployment. The Watermelon team can help you configure this grid during the setup phase. However, the instructions are shared below for your reference.

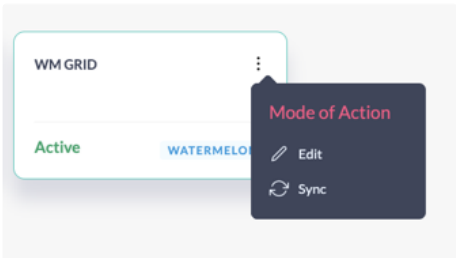
1. **Navigate to Browser Grid:** Access the Browser Grid section under the left pane of WM-Meta.



2. **Create New Integration:** Click on the "Create New" button to initiate the integration process.
3. **Select Device Farm:** From the "Hosted by" dropdown menu, select WATERMELON.
4. **Provide Necessary Details:** Enter the required details such as the Hub URL, User ID, and Access Key/Token for the selected Device Farm. You may also control the number of parallel sessions permitted. To increase this value beyond your initial configuration, please consult with Watermelon support and there may be infrastructure capacity requirements to be met prior to the increase.



5. **Complete Integration:** Once all necessary details are entered, proceed to complete the integration process. Ensure that the information provided is accurate and up-to-date to enable seamless integration with the chosen Browser Grid.
6. Click the kebab menu on the configured device farm seen in the Device Farm repository and click "Sync" to ensure the connectivity is tested and the available devices are visible to Watermelon.

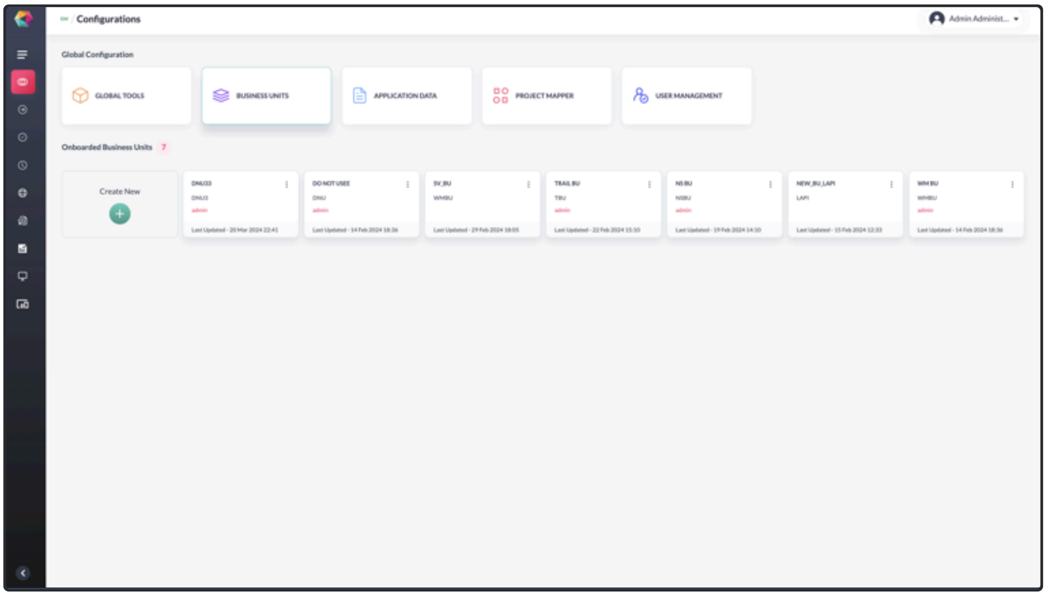


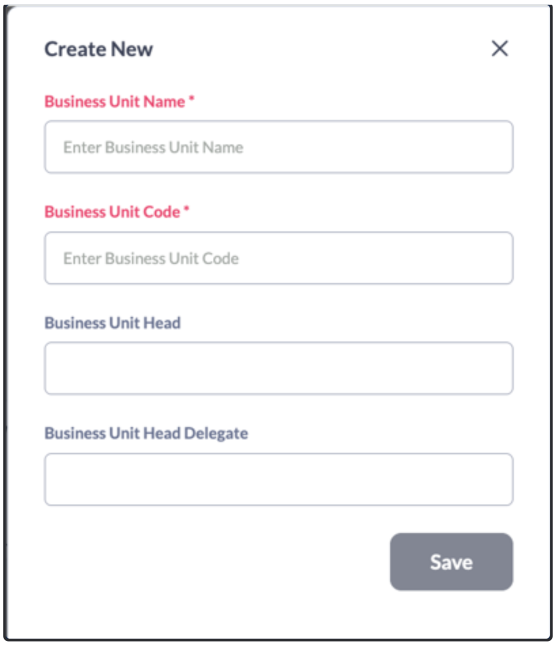
By following these steps, you can successfully integrate Watermelon's browser grid, allowing for efficient testing and monitoring of Web applications across various browser types.

Creating Business Units

Watermelon's organization system employs a Business Unit and Application combination, facilitating clear access control and segregation of duties among users. This structure allows for efficient management of application-based licensing as well. Business Units in Watermelon represent entities or groups within an organization, akin to departments or specific locations for global enterprises.

Creating a Business Unit is straightforward: navigate to Configurations → Business Units under WM-Meta, usually performed by the SYSADMIN role. Upon selecting "Create New," a form prompts for input:





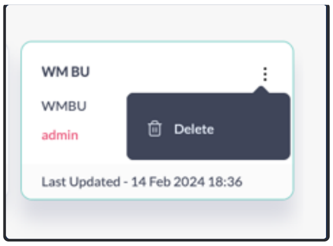
- Business Unit name: Unique name hosting applications to be onboarded to Watermelon, changeable if needed.
- Business Unit code: Short, unique code representing the Business Unit, unalterable once created.
- Business Unit Head: Oversees the entire Business Unit, often the Technology Business Unit head.
- Business Unit Delegate: Represents the Business Unit head, typically a delegate to the Technology Business Unit head.

These roles can be associated with the EXECUTIVE role in Watermelon to access reports and insights across its modules. For more details on Watermelon roles, refer to the Role Based Access Control section.

It's important to note that integrating User Management tools is a prerequisite for automatic user discovery from the directory. Additionally, both the BU Head and Delegate can be added after the Business Unit is created.

Modifying or Deleting a Business Unit

To modify a Business Unit, simply click on the Business Unit name displayed on the card. Conversely, to remove a Business Unit, access the kebab menu on the respective card and choose the Delete option.


WM BU
WMBU
admin
Delete
Last Updated - 14 Feb 2024 18:36

Onboarding Applications

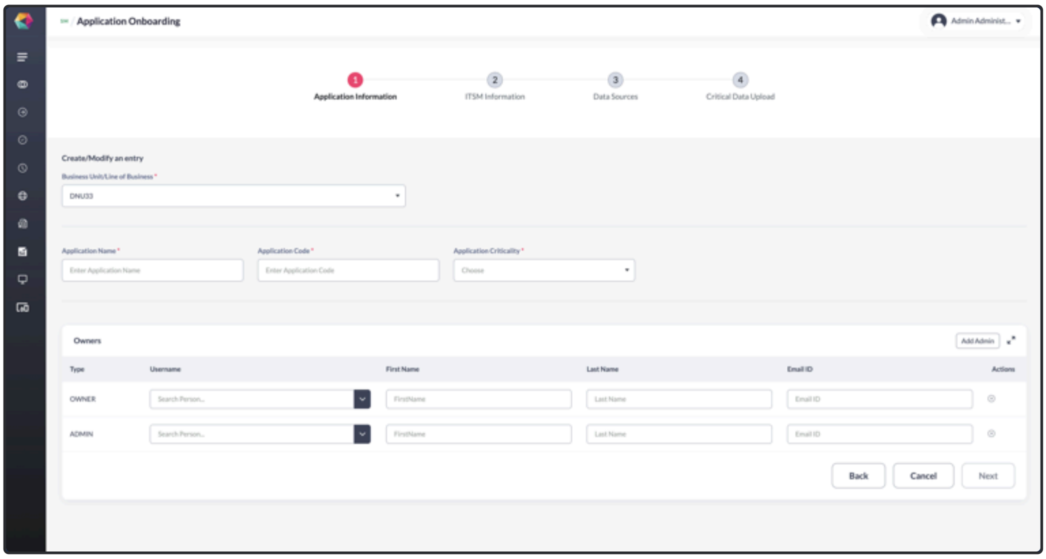
Watermelon's organization system employs a Business Unit and Application combination, facilitating clear access control and segregation of duties among users. This structure allows for efficient management of application-based licensing as well. Business Units in Watermelon represent entities or groups within an organization, akin to departments or specific locations for global enterprises.

Creating or onboarding an application is a straightforward process within Watermelon. As a SYSADMIN role, navigate to Configurations → Application Data under WM-Meta to begin. Clicking "Create New" initiates a 4-step application onboarding journey.

Step 1: This step involves providing basic application metadata, including the Business Unit (BU) the application belongs to, Application Name, Application code (a short internal reference), and its Criticality (e.g., CRITICAL, HIGH, MEDIUM, LOW).

Assign an Application Owner, typically the Technology Manager, and one or more Administrators for the application. It's important to note that the Application Administrator holds all privileges on the application and its assets within Watermelon. The Owner and Administrators can be updated in case of any changes using the same steps as performed during the Onboarding phase.

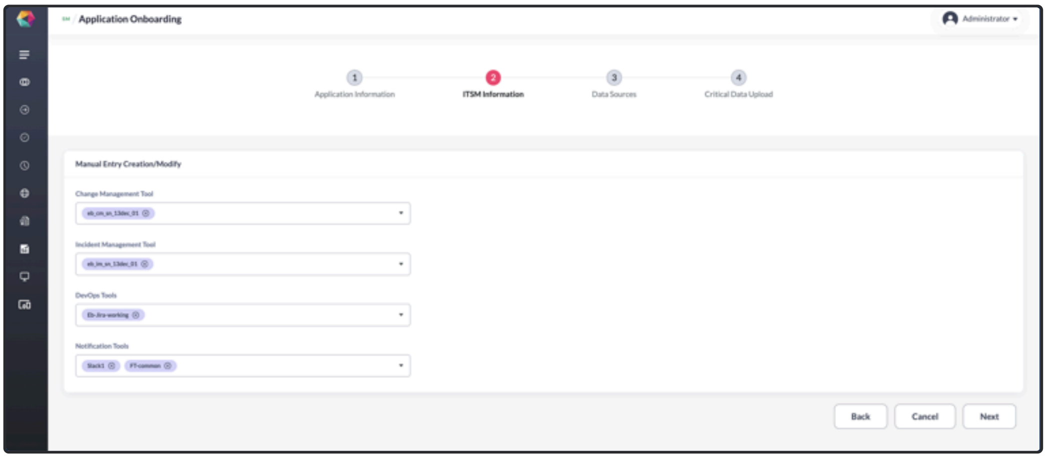
More information on role-based access control (RBAC) can be found in the documentation.



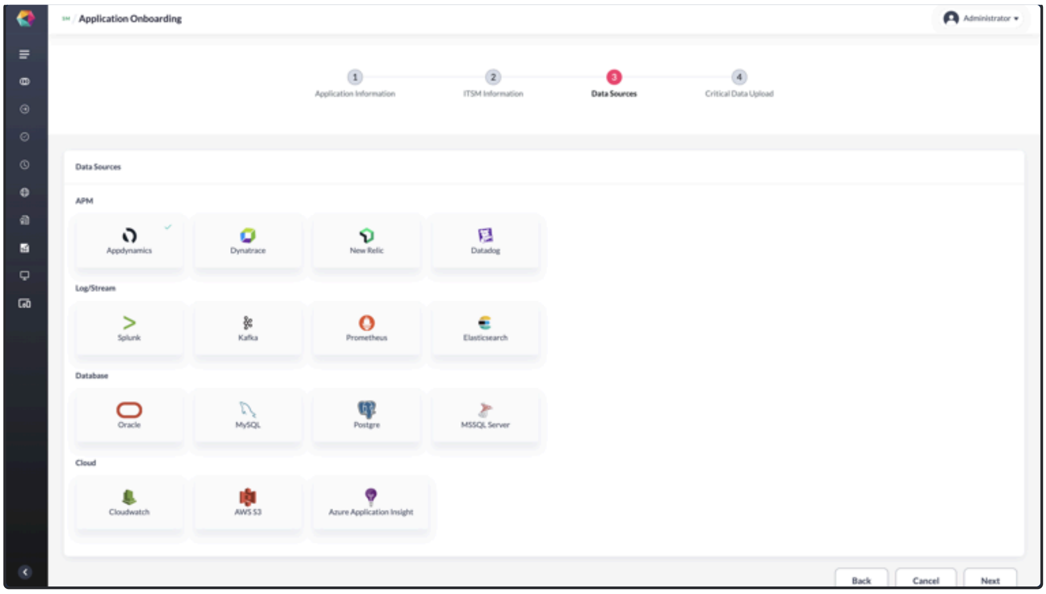
Step 2: This step involves associating the application with Change Management, Incident Management, DevOps, and Notification tools it may utilize. These tools are typically pre-

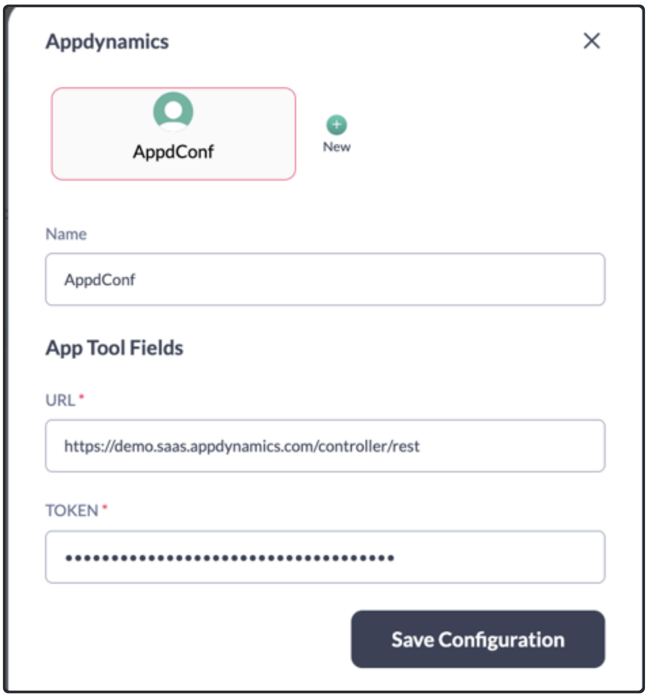
configured under WM-Meta → Configurations → Global Tools and will be listed for association. Watermelon offers the flexibility to configure multiple instances of tools under Global Tools, allowing different applications to use the instances they are hosted on. This flexibility is designed to accommodate organizational nuances where such flexibility may be desired.

For a comprehensive list of supported tools and instructions on configuring them, please refer to the Global Tools section on the Watermelon documentation.



Step 3: This step involves associating various data sources, including APM tools, Log Management tools, and Databases, which may be utilized by various Watermelon modules such as SLO Manager and Chaos Engineering for analytics, correlation, and insights reporting. Unlike Global Tools, these data source tools are not configured centrally, as they could vary from application to application. Watermelon provides the flexibility to configure multiple instances of data sources, enabling applications to use different instances as required. This flexibility is designed to accommodate organizational nuances where such flexibility may be desired. This is not mandatory to register data sources at time of onboarding and can be done eventually using the modify option by clicking on the application name in the Application Data repository.

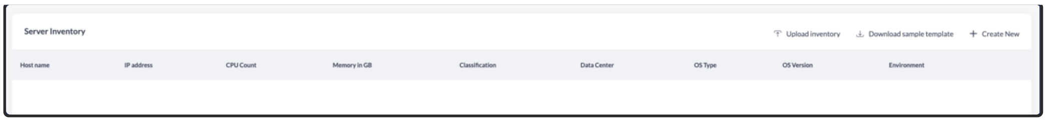




Sample integration with AppDynamics

You may refer to the details for configuring each of the supported data sources in Data Sources Integration section.

Step 4: This step involves uploading infrastructure inventory mainly bare metal and virtual machine servers that can be consumed by Watermelon modules such as Chaos Engineering and SLO Manager. The template is available for download and can be uploaded or entries can be made manually. This is not a mandatory step and can be done eventually using the modify option.



Confidential. Information proprietary to Watermelon Software Inc. Do not copy or distribute.

Project Mapper

Watermelon supports integration with Application Lifecycle Management (ALM) tools such as Jira for multiple reasons such as:

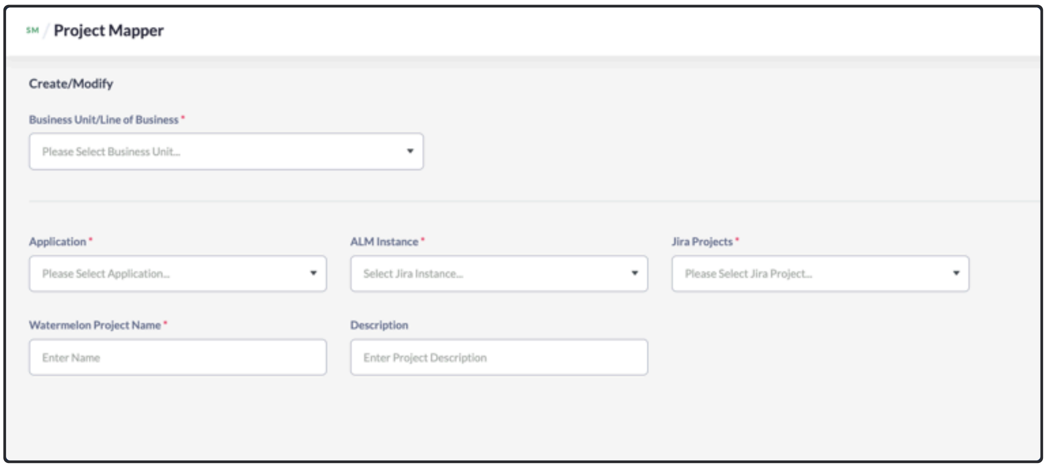
1. Mapping business requirements against test cases from a testing perspective
2. Raising bugs/defects from a testing perspective
3. Raising observations as either defects or requirements from a CHaos engineering perspective
4. Creating tickets from an SLO management perspective

These use cases require a mapping between projects in Jira and Watermelon and this is facilitated by WM-Meta.

To map projects or create new projects within Watermelon, follow these steps:

1. Select the Business Unit, which will provide a dropdown list of associated applications.
2. Choose the Application for which you need to map the project.
3. From the list of associated ALM instances (configured during application onboarding), select one on which the project is created. If no ALM instance is available or if you prefer not to use one, select WATERMELON, which will create a project within Watermelon itself.
4. Provide a name for the project within Watermelon and type a description for future reference.

With these steps, project mapping or creation within Watermelon is complete.



Role Based Access Control

Watermelon's Role-Based Access Control (RBAC) facilitates a streamlined, self-service approach to managing user roles and permissions through its WM-Meta module. Understanding how to configure roles within Watermelon requires a grasp of its RBAC mechanism and the roles available. Let's go over the prerequisites for setting up RBAC in Watermelon:

Prerequisites for RBAC Configuration in Watermelon:

1. **Integration with User Management Tools**: The first step is integrating Watermelon with your existing user management system, such as Active Directory, Okta, or other SAML/OIDC-based tools, via Watermelon's KeyCloak integration. Detailed instructions for these integrations are available in the documentation section of Watermelon User Management Tools. Using this approach users of Watermelon are aligned to your organizations User Management/SSO standards and this reduces the overheads of maintaining users specifically within Watermelon.
2. **User Creation in KeyCloak**: If you opt not to integrate with an external user management tool, you can use Watermelon's KeyCloak for user creation. Follow the KeyCloak user manual to create user IDs, specifying details like User ID, First Name, Last Name, Email, and an initial password. The assignment of users to groups will be managed automatically through the "User Access Management" feature within WM-Meta. Do note that while this is a supported approach, the user access management will need to be handled by your Watermelon System Administrator as direct access to KeyCloak would be restricted. For the most current instructions on user management within KeyCloak, refer to the official KeyCloak documentation: KeyCloak User Management.

By following these prerequisites, you set the foundation for effective RBAC within Watermelon, ensuring that access and permissions are configured accurately and securely across your organization. This setup not only enhances the security of your Watermelon deployment but also aligns with best practices for user access management and control.

Roles

Watermelon delineates various roles within its system to ensure a granular level of access control and operational efficiency. Here's an overview of the roles available within Watermelon and their intended purposes:

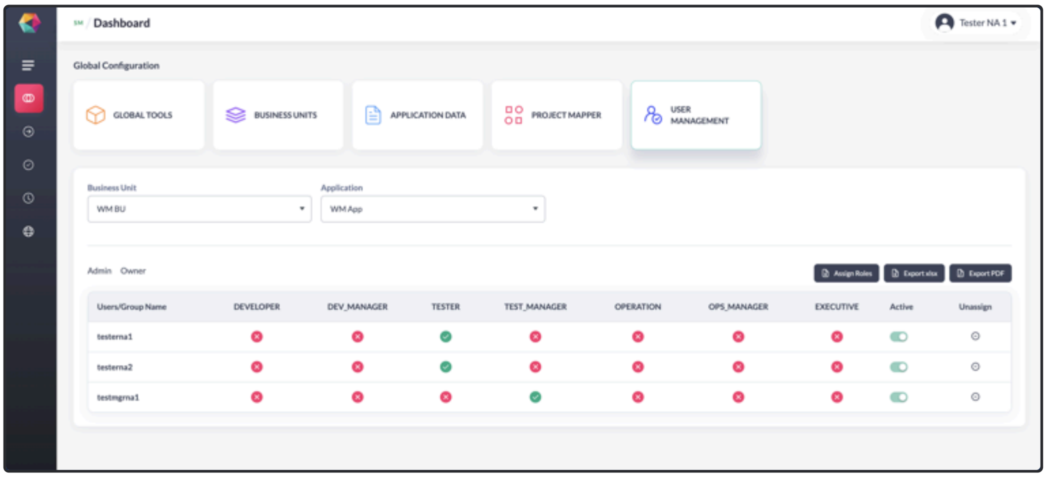
- **SYSADMIN**: This role is established during the setup phase and is created locally within KeyCloak. It serves as a break-glass account, with its credentials needing secure storage. It's advisable to add more members to this role via KeyCloak after integrating User Management tools.
- **OWNER**: Assigned during the Application Onboarding phase under WM-Meta, this role represents the application owner, responsible for the overarching management of the application.
- **ADMIN**: Different from the SYSADMIN, the Admin is designated to each application, possessing extensive control and access over all assets within a specific Business Unit (BU)/Application combination.
- **DEVELOPER**: Developers have access to perform a broad range of actions, given their involvement from the early stages of the system life cycle for unit testing, Chaos Engineering, among other activities. This role is considered a Maker role and lacks approval privileges.
- **DEVELOPMENT MANAGER**: This role, assigned to a development manager, allows access to assets within their BU/Application combination. As a Checker role, it grants approval rights for requests initiated by the DEVELOPER role. Note that individuals in a DEVELOPER role cannot concurrently hold a DEVELOPMENT MANAGER or TEST MANAGER role.
- **TESTER**: Testers, akin to developers, can perform most actions and use Watermelon for various testing stages and Chaos Engineering. This Checker role, however, does not hold any approval privileges.
- **TEST MANAGER**: Test Managers can access assets related to their BU/Application combination, with the authority to approve requests made by the TESTER role. Like with DEVELOPMENT MANAGER, a TESTER cannot also be a DEVELOPMENT MANAGER or TEST MANAGER.
- **OPS**: Targeted at SRE/production support personnel, this role focuses on the production modules of Watermelon, such as SLO Management. It's a Maker role with the ability to propose changes to SLO configurations, though it lacks approval capabilities.
- **OPS MANAGER**: The SRE/Production Manager role encompasses access to assets within the BU/Application combo and can approve changes requested by the OPS role. An individual in an OPS role cannot be assigned as an OPS MANAGER.
- **EXECUTIVE**: This role in Watermelon is designed as a primarily read-only access level, tailored to accommodate the needs of various high-level personas within an organization. This role is ideal for C-suite executives (CxO), and professionals in Risk and Compliance, Audit functions, and similar positions who require a broad overview of operational insights without the necessity for direct interaction with the system's functional controls.

This structure facilitates precise role assignment, ensuring that each team member has the appropriate level of access and responsibilities, thereby promoting efficient collaboration and secure management of applications and services within Watermelon.

Using the User Management capabilities

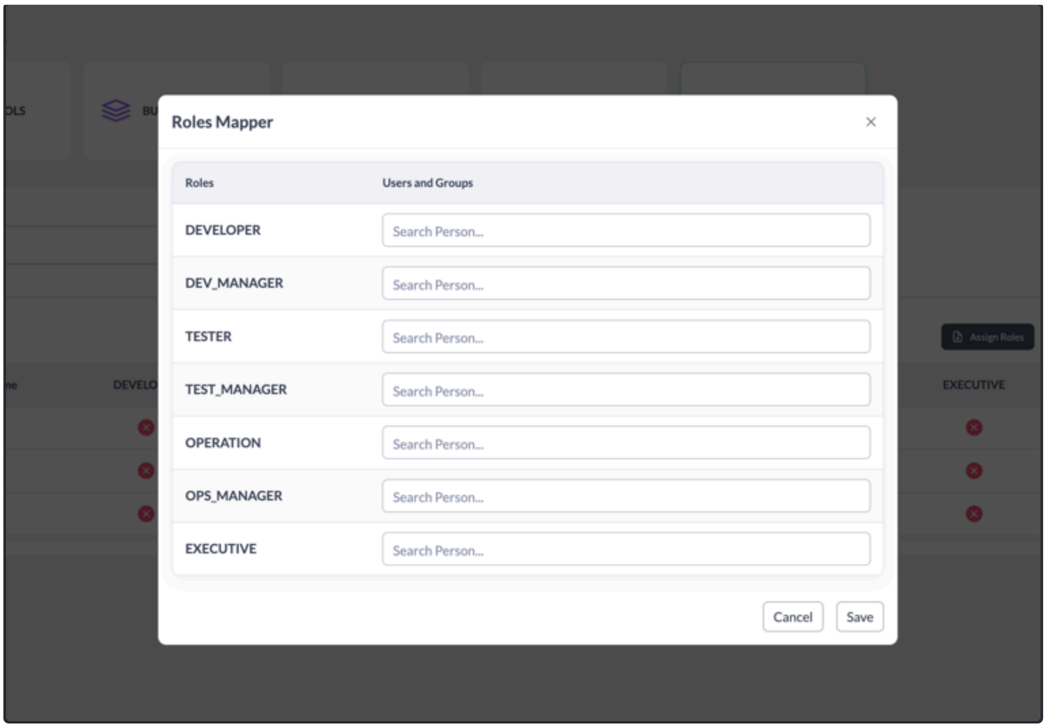
Utilizing the User Management functionalities within Watermelon, especially under the WM-Meta section, is straightforward and offers administrators the flexibility to assign or unassign roles to users within specific Business Units (BUs) and Applications. Here's a concise guide on how to use these capabilities effectively:

1. **Navigate to User Management**: As an ADMIN, go to the User Management option located under Configurations in the WM-Meta dashboard.



User Management

2. **Select BU and Application**: Choose the relevant Business Unit and Application for which you intend to manage user roles.



Role Mapper

3. **Assign Roles**: Click on the “Assign Roles” button to open the Role Mapper. Here, you can search for users by their IDs. The search results will automatically populate based on the data from your integrated user management source.
4. **Select/Deselect Roles**: In the list view showing the search results, you can easily select or deselect roles for each user by checking or unchecking the boxes next to their names.
5. **Mark a User as Inactive**: If necessary, you can deactivate a user account by toggling the switch under 'Active'. This action locks the user out of Keycloak, preventing them from logging into Watermelon. Given the significant impact of this action, it should be used with caution.
6. **Unassign Roles**: Should you need to remove a user from a specific role, simply click the “Unassign” button next to the relevant role and user.

Data Sources Integration

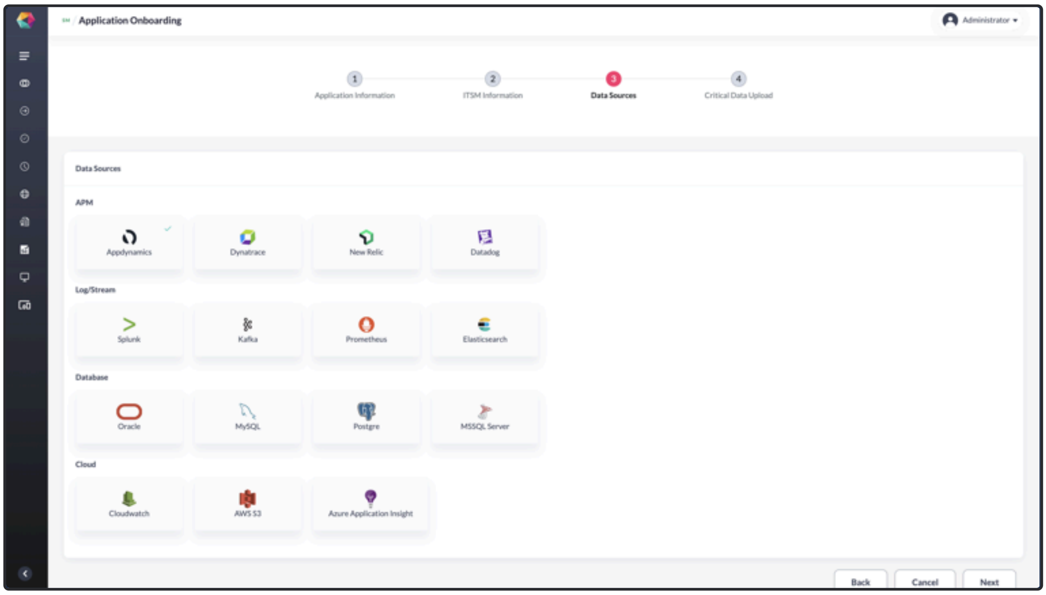
This section provides an overview of the various data sources that can be seamlessly integrated with Watermelon. These data sources encompass a wide range of tools and technologies commonly used in modern IT environments. The supported data sources include:

1. APM Tools: Application Performance Monitoring tools, which offer insights into the performance and behavior of applications, are supported by Watermelon. These tools help monitor application health, diagnose performance issues, and optimize application performance.
2. Log-Based Ingestion Tools: Log management and analysis tools enable the collection, aggregation, and analysis of log data generated by applications and systems. Watermelon supports integration with various log management tools to provide comprehensive visibility into system logs and facilitate troubleshooting and analysis.
3. Proprietary Databases: Watermelon seamlessly integrates with proprietary databases, allowing organizations to leverage data stored in their databases for monitoring, analysis, and reporting purposes. This enables organizations to gain insights into their data and make informed decisions based on real-time information.
4. Cloud Technologies: Watermelon supports integration with various cloud data sources such as AWS Cloudwatch and Azure Application Insights. This enables organizations to monitor and manage their cloud infrastructure and applications effectively.

If you have a data source that's not listed in this section, please don't hesitate to reach out to Watermelon support for further assistance. Our team can evaluate the feasibility of integrating your specific data source with Watermelon to meet your organization's needs effectively.

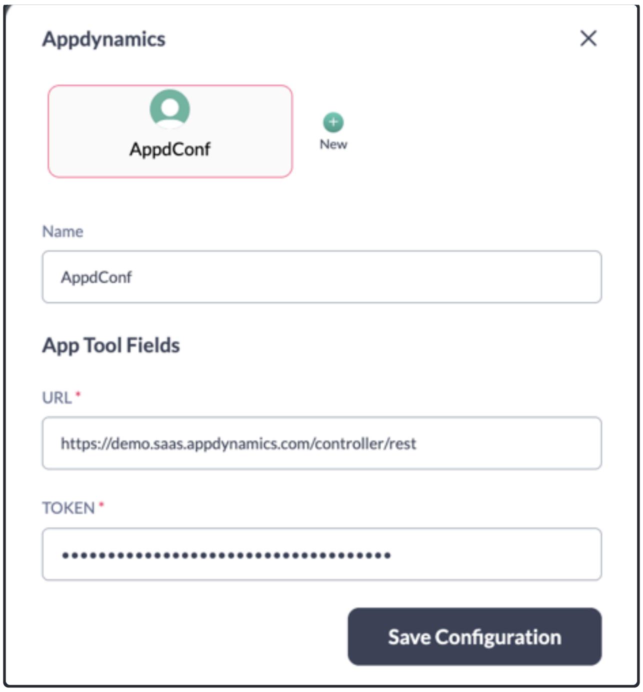
AppDynamics Data Source

To configure AppDynamics as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on AppDynamics with the required privileges to connect and query data through APIs. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



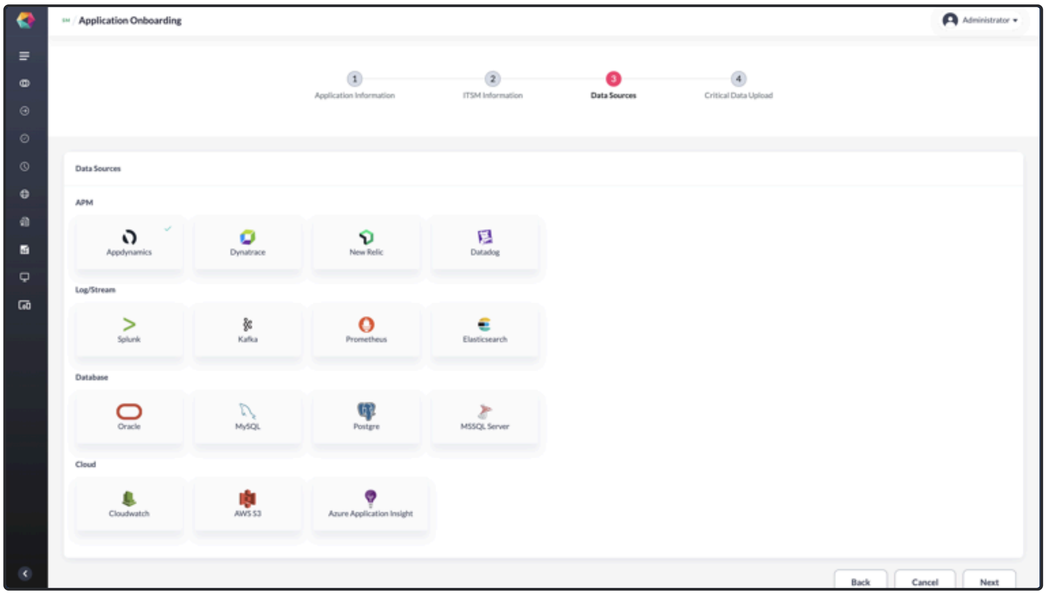
Step 3 of the Application Onboarding journey

Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the AppDynamics URL endpoint and the Token for the user account with the necessary privileges on the relevant applications. Enter these details accordingly to complete the configuration process as shown in the illustration below.


Appdynamics
AppdConf
New
Name
AppdConf
App Tool Fields
URL *
https://demo.saas.appdynamics.com/controller/rest
TOKEN *
Save Configuration

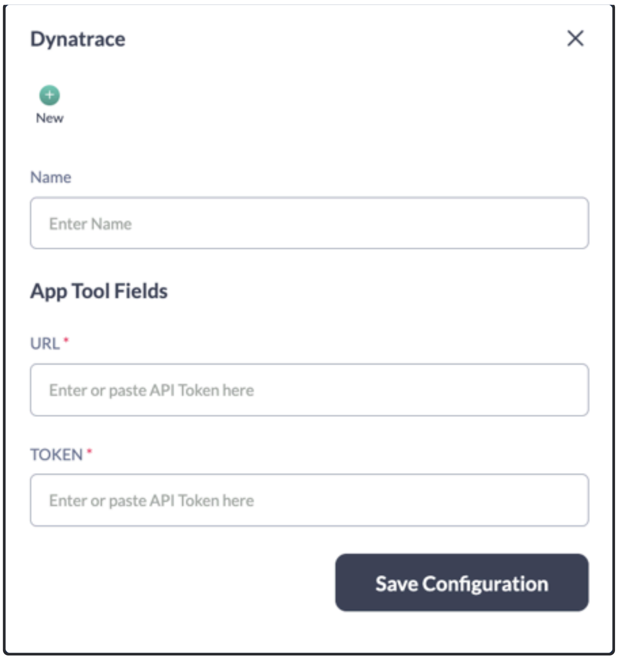
Dynatrace Data Source

To configure Dynatrace as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on Dynatrace with the required privileges to connect and query data through APIs. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



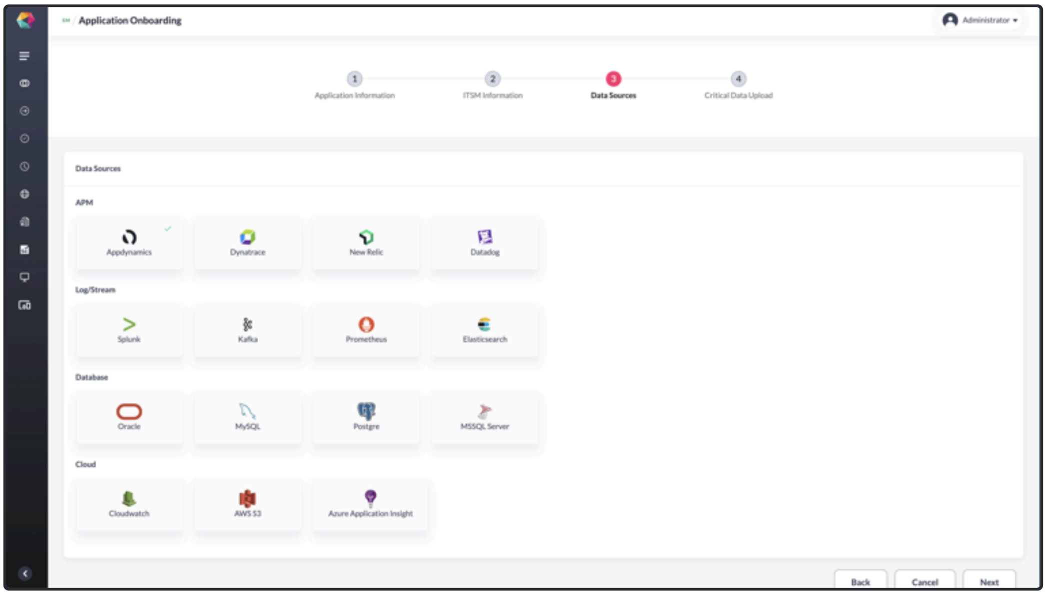
Step 3 of the Application Onboarding journey

Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the Dynatrace URL endpoint and the Token for the user account with the necessary privileges on the relevant applications. Enter these details accordingly to complete the configuration process as shown in the illustration below.


Dynatrace
New
Name
Enter Name
App Tool Fields
URL *
Enter or paste API Token here
TOKEN *
Enter or paste API Token here
Save Configuration

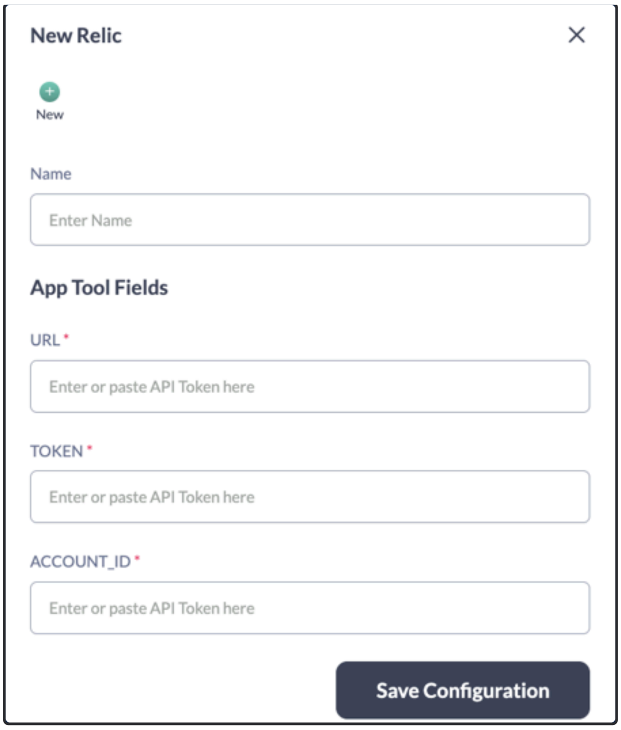
New Relic Data Source

To configure New Relic as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on New Relic with the required privileges to connect and query data through APIs. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



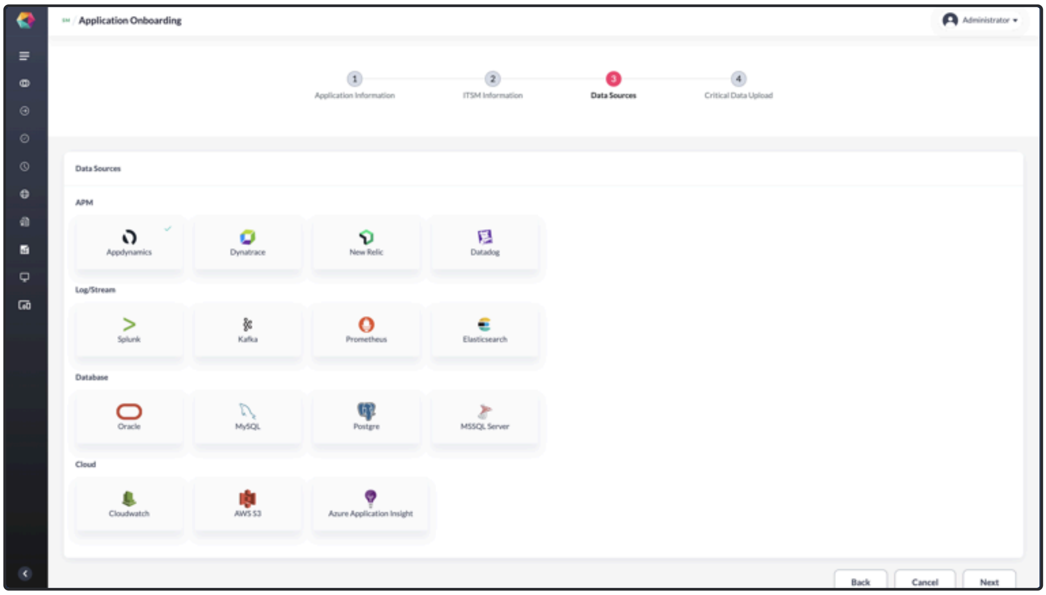
Step 3 of the Application Onboarding journey

Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the New Relic URL endpoint , the Token for the user account with the necessary privileges on the relevant applications and the hosted Account ID. Enter these details accordingly to complete the configuration process as shown in the illustration below.


New Relic
New
Name
Enter Name
App Tool Fields
URL *
Enter or paste API Token here
TOKEN *
Enter or paste API Token here
ACCOUNT_ID *
Enter or paste API Token here
Save Configuration

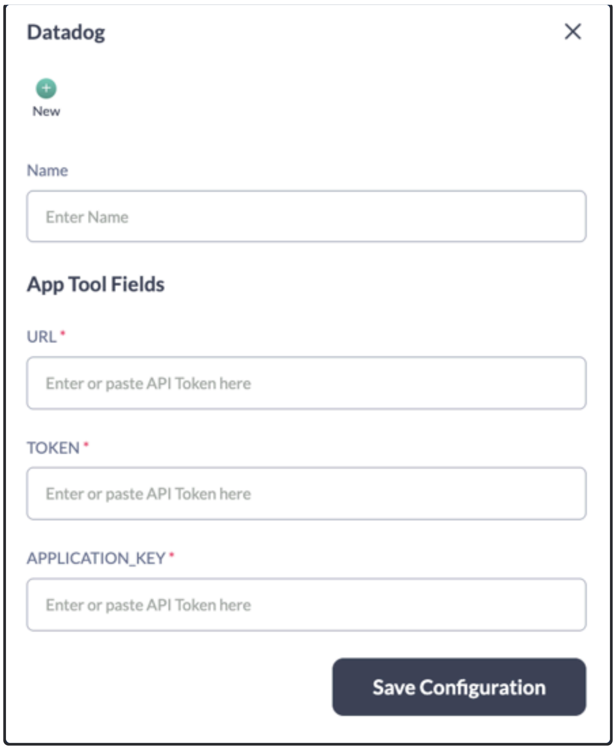
Datadog Data Source

To configure Datadog as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on Datadog with the required privileges to connect and query data through APIs. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



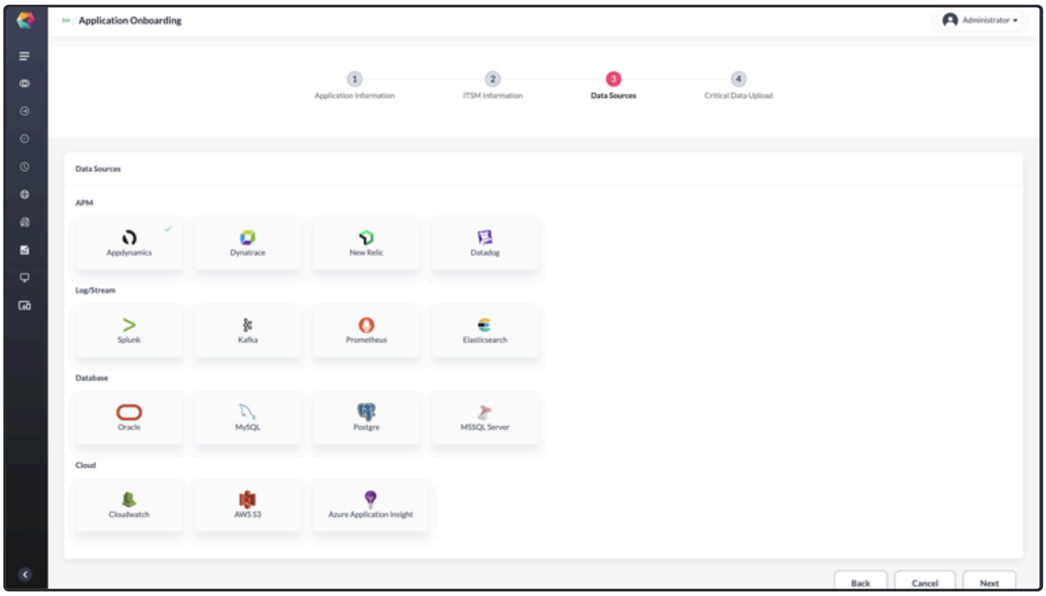
Step 3 of the Application Onboarding journey

Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the Datadog URL endpoint , the Token for the user account with the necessary privileges on the relevant applications and the hosted Application Key. Enter these details accordingly to complete the configuration process as shown in the illustration below.


Datadog
New
Name
Enter Name
App Tool Fields
URL *
Enter or paste API Token here
TOKEN *
Enter or paste API Token here
APPLICATION_KEY *
Enter or paste API Token here
Save Configuration

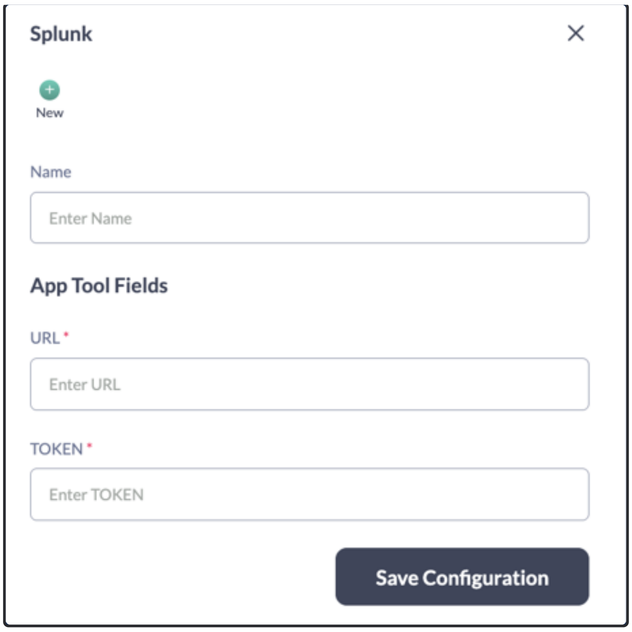
Splunk Data Source

To configure Splunk as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on Splunk with the required privileges to connect and query data through APIs. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



Step 3 of the Application Onboarding journey

Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the Splunk URL endpoint and the Token for the user account with the necessary privileges on the relevant applications/indexes. Enter these details accordingly to complete the configuration process as shown in the illustration below.

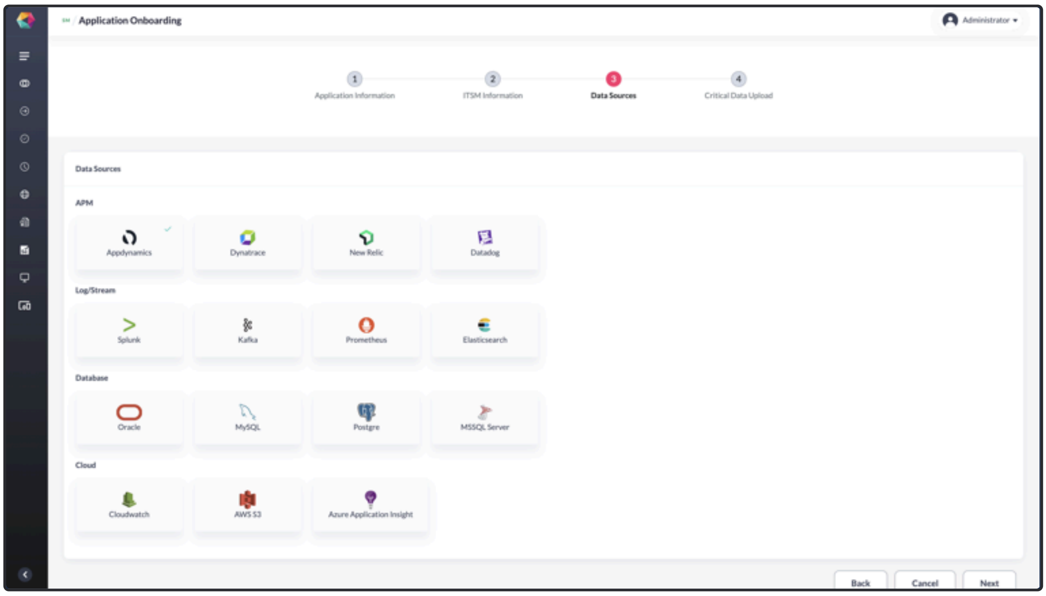


Splunk configuration

Confidential. Information proprietary to Watermelon Software Inc. Do not copy or distribute.

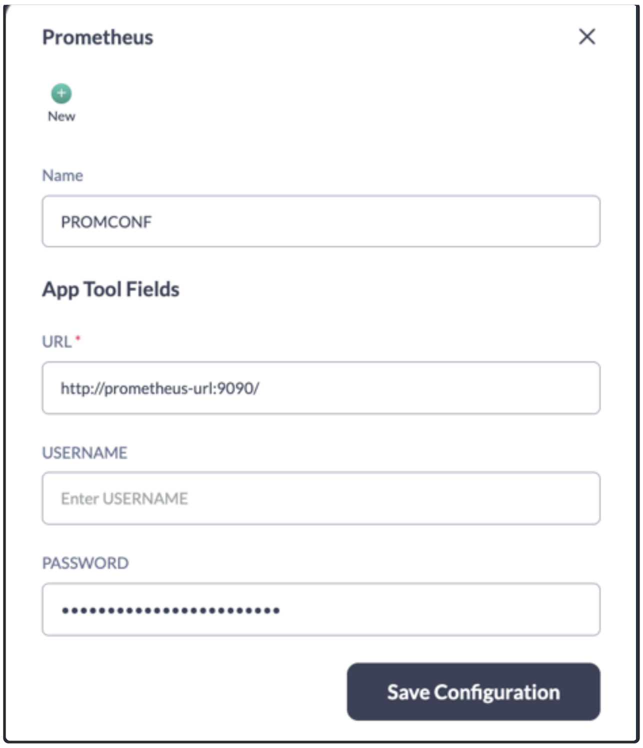
Prometheus Data Source

To configure Prometheus as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on Prometheus with the required privileges to connect and query data through APIs. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



Step 3 of the Application Onboarding journey

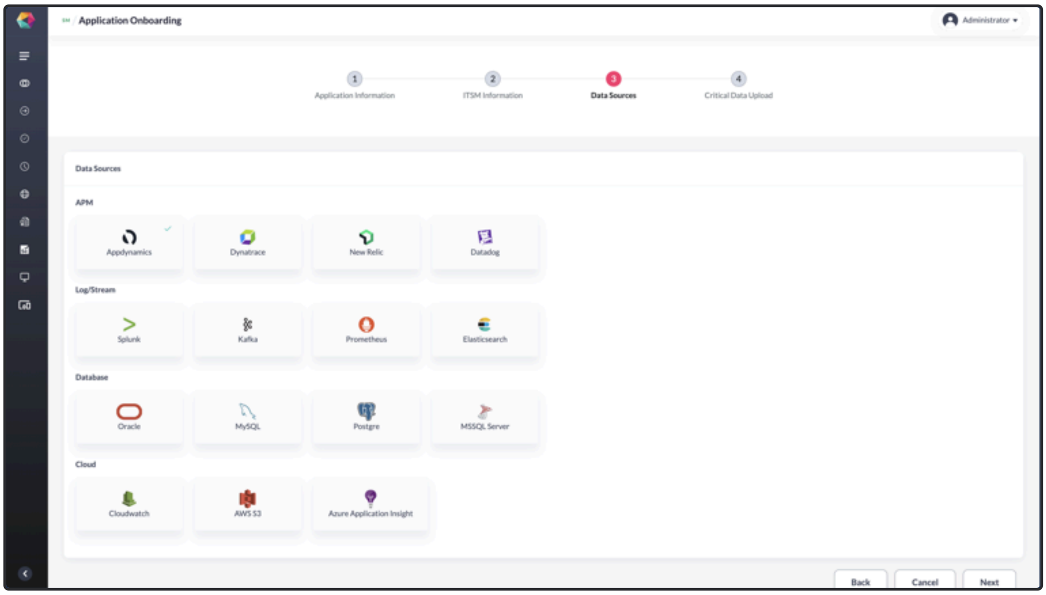
Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the Prometheus URL endpoint and the Password/Token for the user account with the necessary privileges on the relevant applications/indexes. Enter these details accordingly to complete the configuration process as shown in the illustration below.



Prometheus configuration

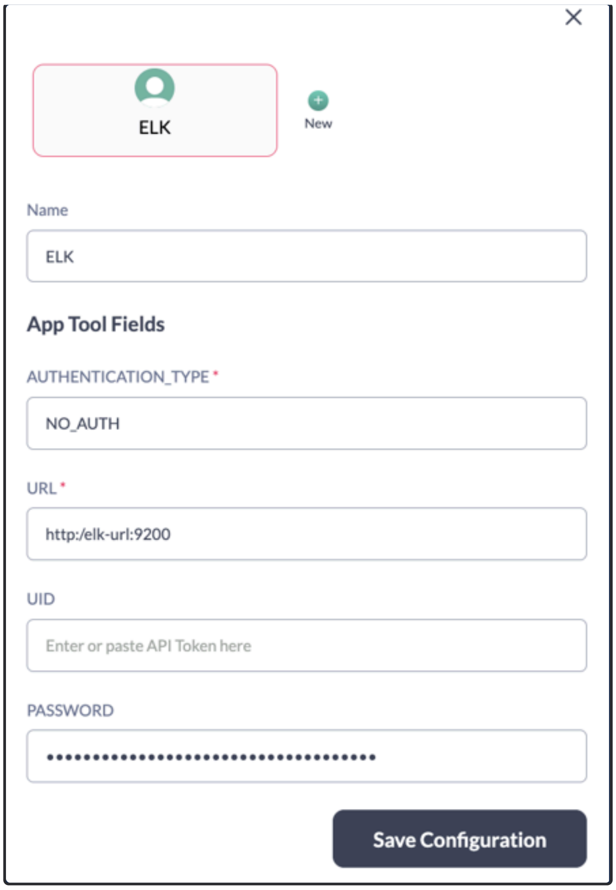
Elasticsearch (ELK) Data Source

To configure ELK as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on ELK with the required privileges to connect and query data through APIs. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



Step 3 of the Application Onboarding journey

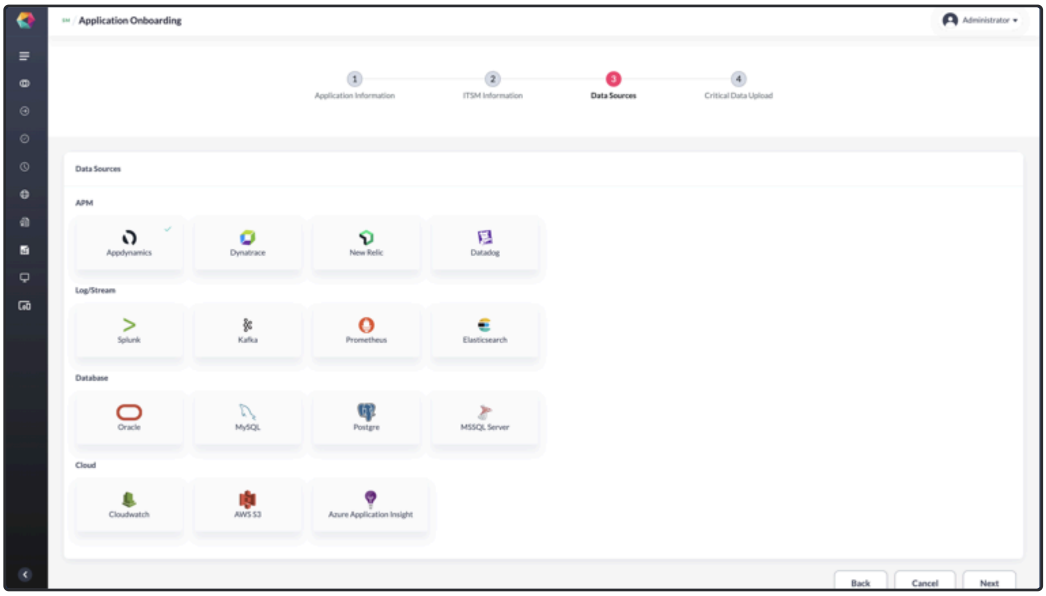
Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the ELK URL endpoint, the Token for the user account with the necessary privileges on the relevant applications/indexes and the authentication type supported by ELK. Enter these details accordingly to complete the configuration process as shown in the illustration below.



Elasticsearch configuration

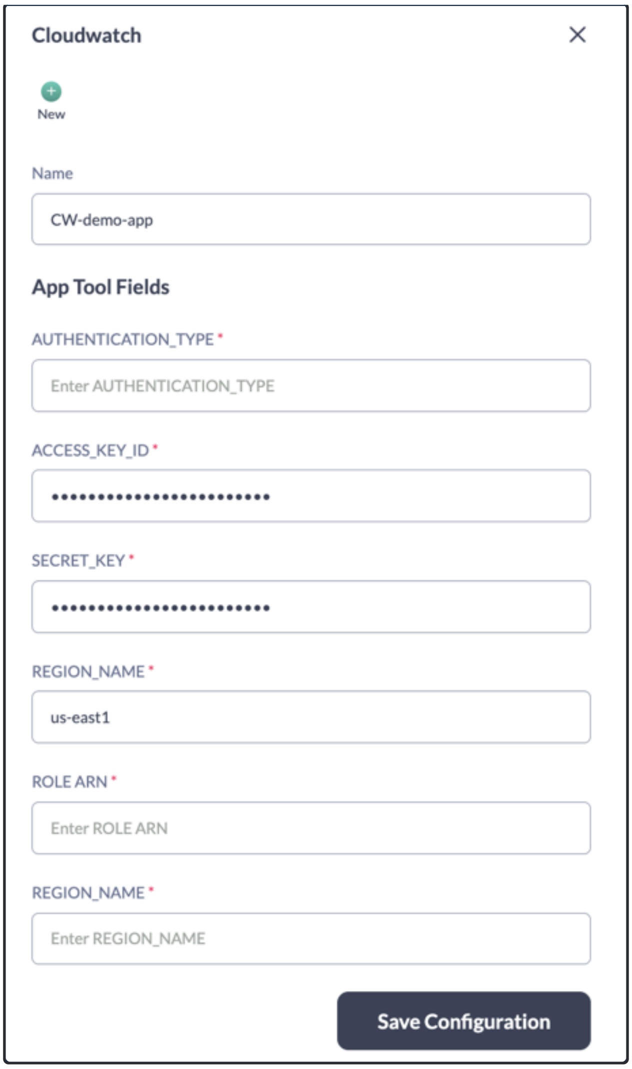
AWS Cloudwatch Data Source

To configure AWS Cloudwatch as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on Cloudwatch with the required privileges to connect and query data through APIs. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



Step 3 of the Application Onboarding journey

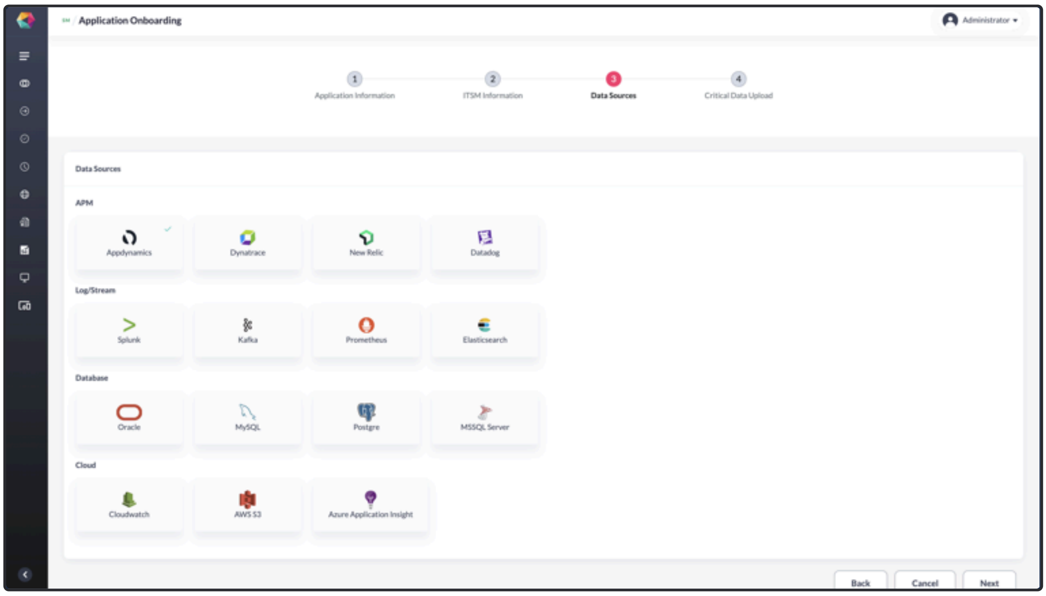
Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the AWS Cloudwatch URL endpoint and the Region. Watermelon supports integration using both Key/Secret combination as well as IAM roles. Ensure all the necessary access is configured on the AWS side. Enter these details accordingly to complete the configuration process as shown in the illustration below.



AWS Cloudwatch configuration

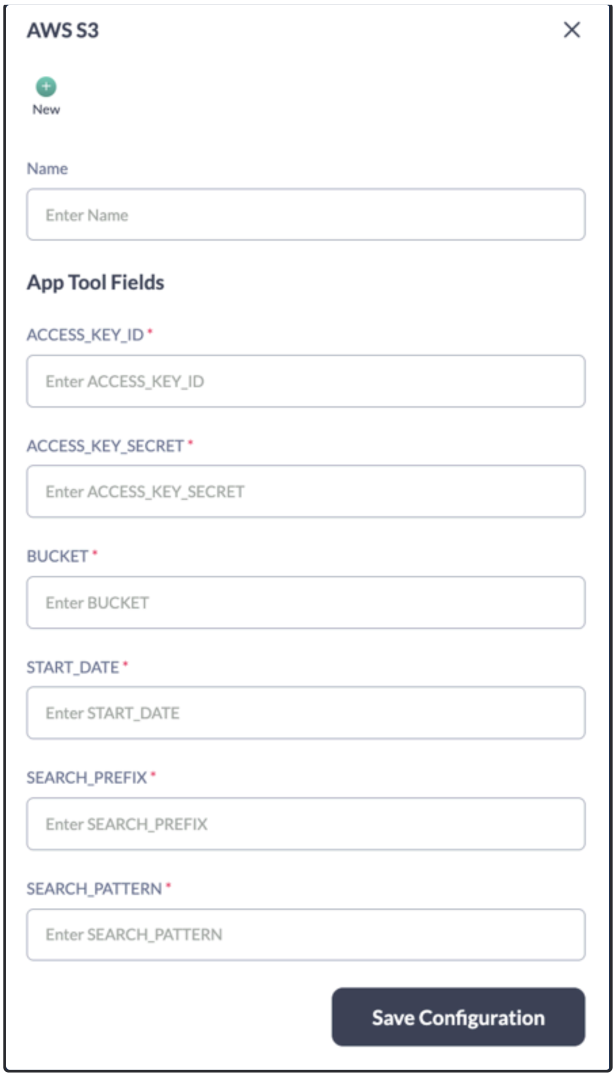
AWS S3 Data Source

To configure AWS S3 as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created with the required privileges to connect and query data through supported mechanisms. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



Step 3 of the Application Onboarding journey

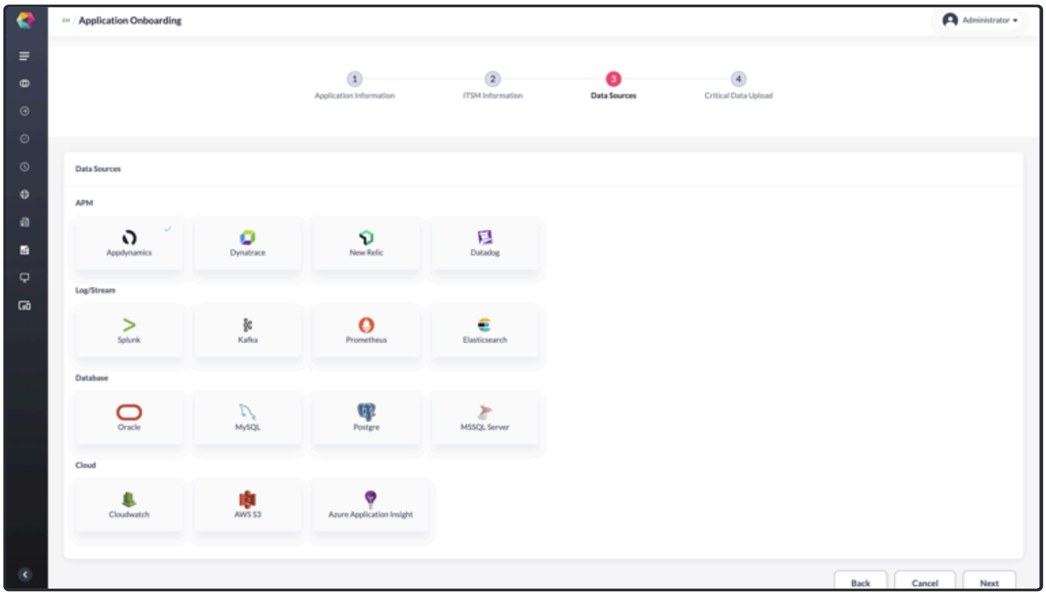
Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the AWS S3 bucket name, Search Prefix, Search Pattern, Region and the start date from which the data needs to be ingested. Watermelon supports integration using both Key/Secret combination as well as IAM roles. Ensure all the necessary access is configured on the AWS side. Enter these details accordingly to complete the configuration process as shown in the illustration below.



AWS S3 configuration

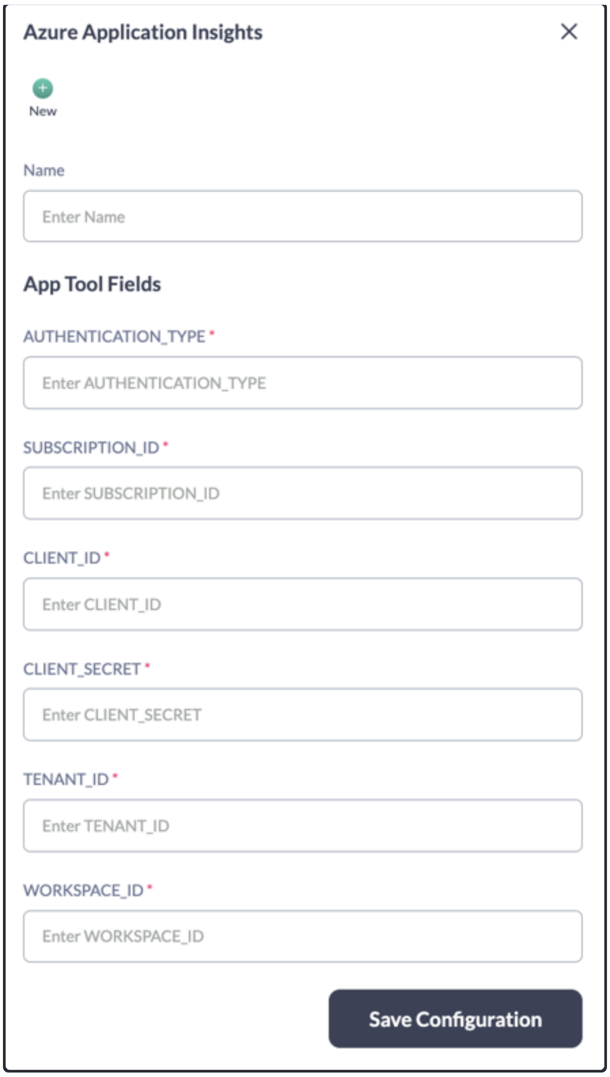
Azure App Insights Data Source

To configure Azure Application Insights as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on App Insights with the required privileges to connect and query data through supported mechanisms. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



Step 3 of the Application Onboarding journey

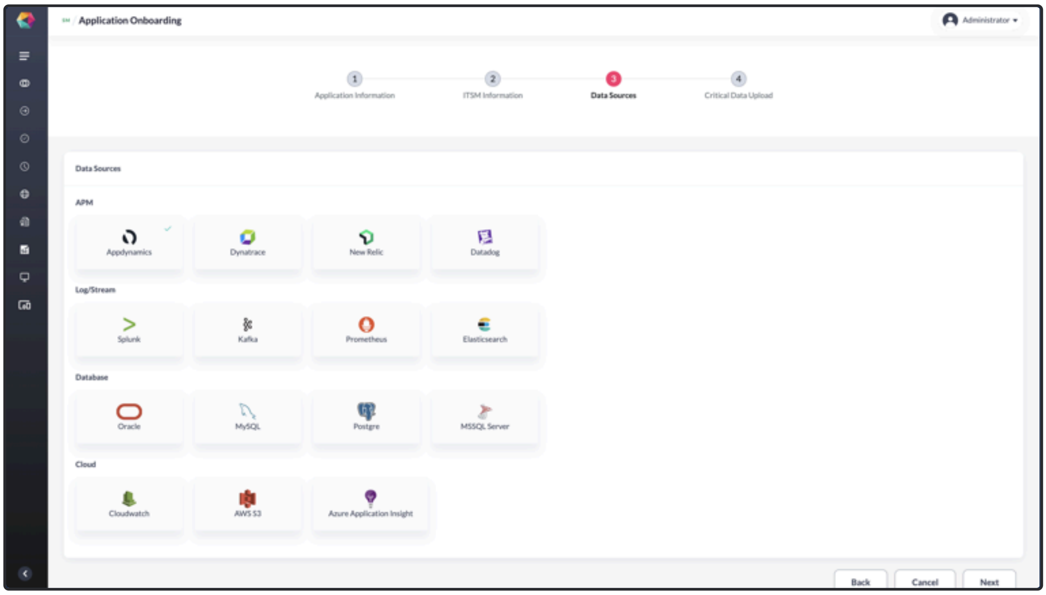
Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the Azure Tenant ID, Client ID, Subscription ID, Client Secret and Workspace ID. Ensure all the necessary access is configured on the AWS side. Enter these details accordingly to complete the configuration process as shown in the illustration below.



Azure Application Insights

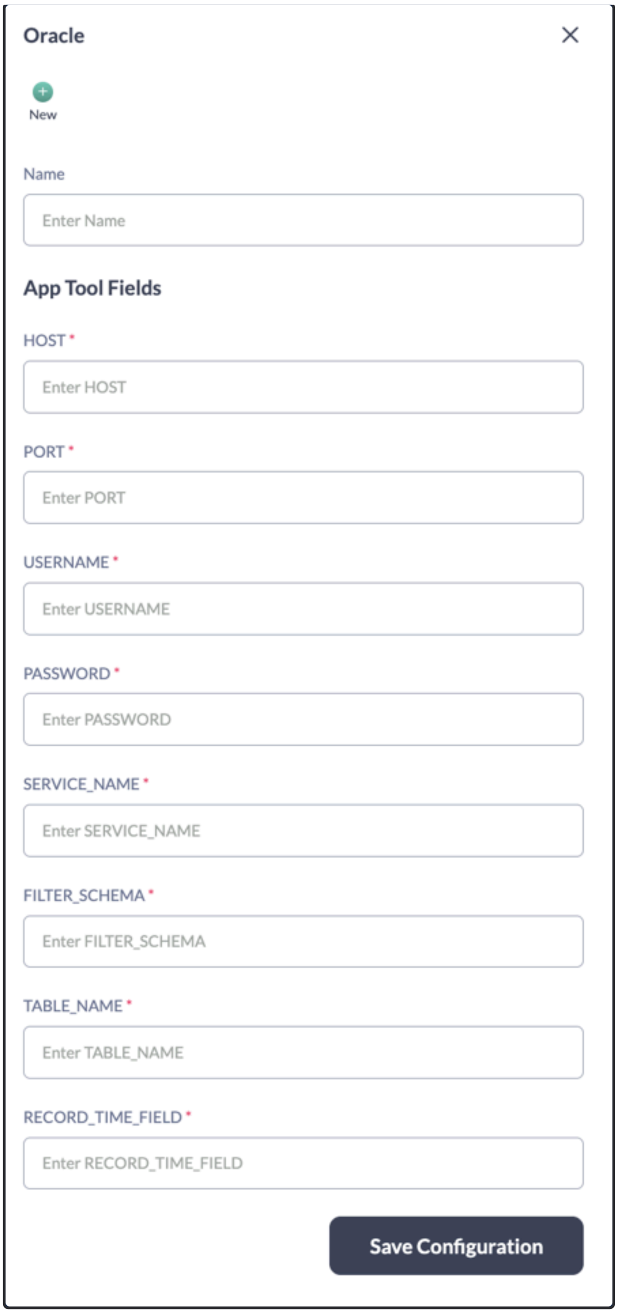
Oracle Data Source

To configure Oracle databases as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on Oracle with the required privileges to connect and query data through supported mechanisms. For more detailed instructions, refer to the Application Onboarding documentation available at [Onboarding Applications].



Step 3 of the Application Onboarding journey

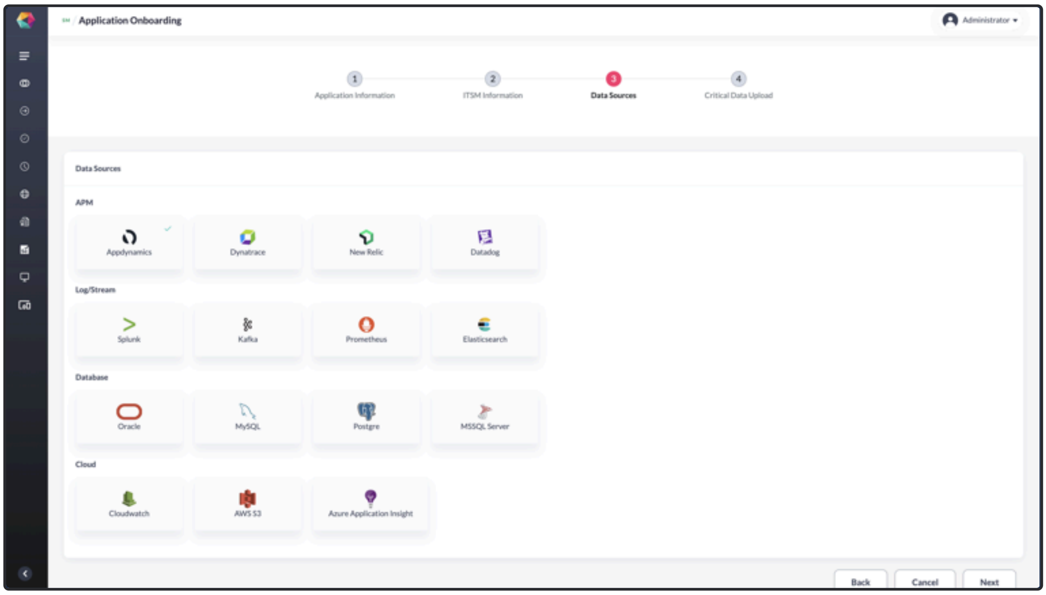
Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the Oracle host details along with the details required to construct the necessary queries for data retrieval. Enter these details accordingly to complete the configuration process as shown in the illustration below.



Oracle configuration

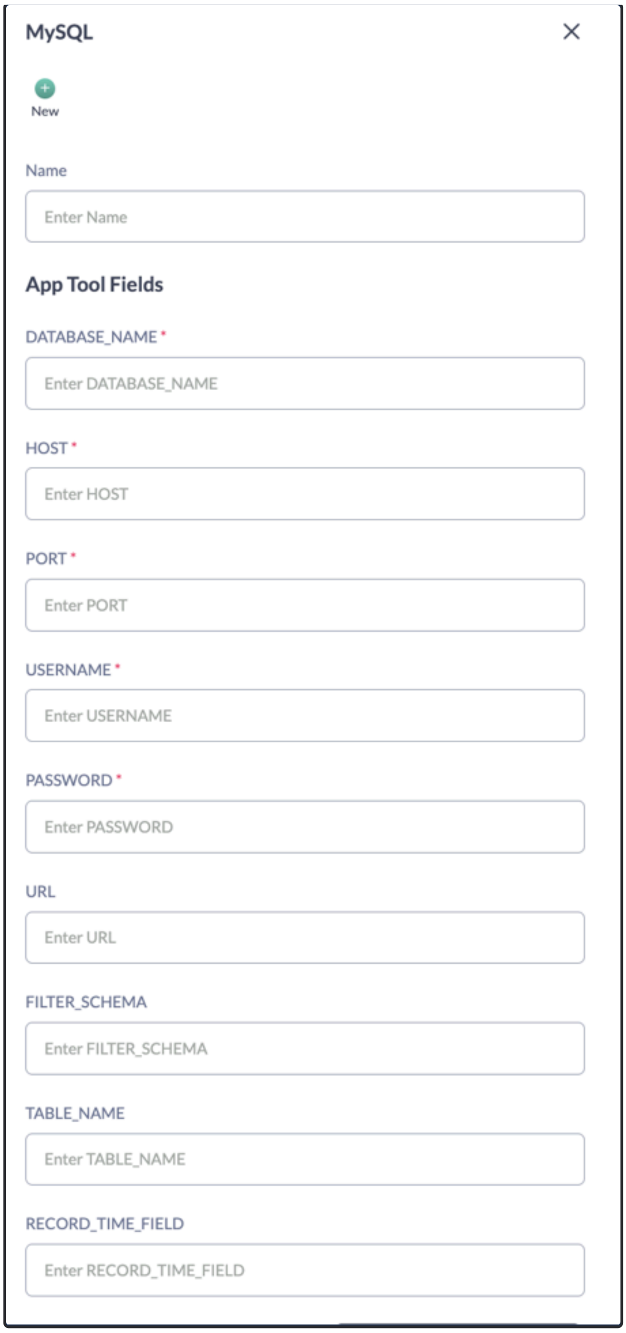
MySQL Data Source

To configure MySQL databases as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on MySQL with the required privileges to connect and query data through supported mechanisms. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



Step 3 of the Application Onboarding journey

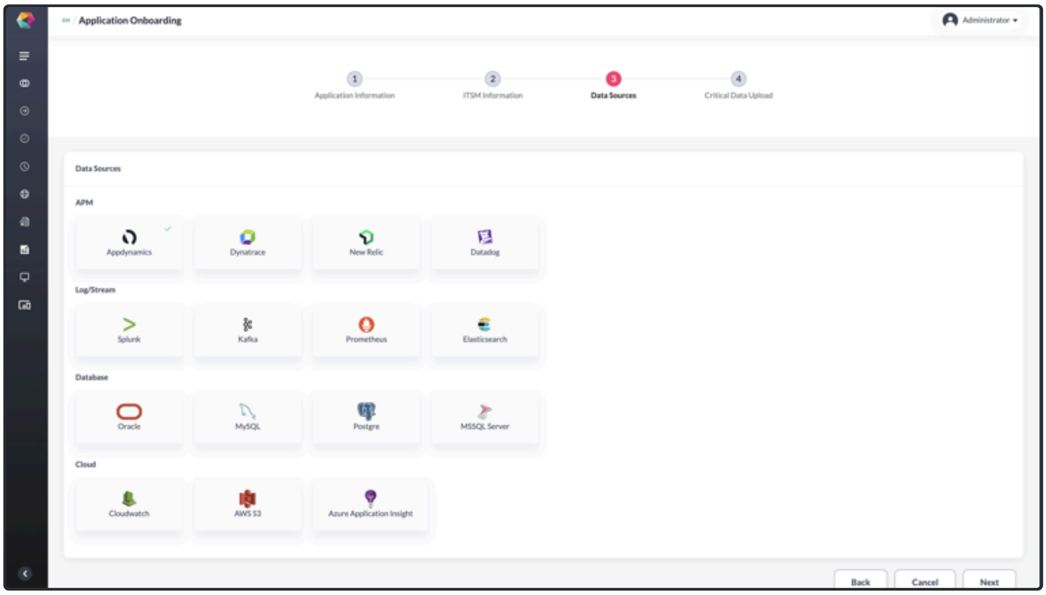
Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the MySQL host details along with the details required to construct the necessary queries for data retrieval. Enter these details accordingly to complete the configuration process as shown in the illustration below.



MySQL configuration

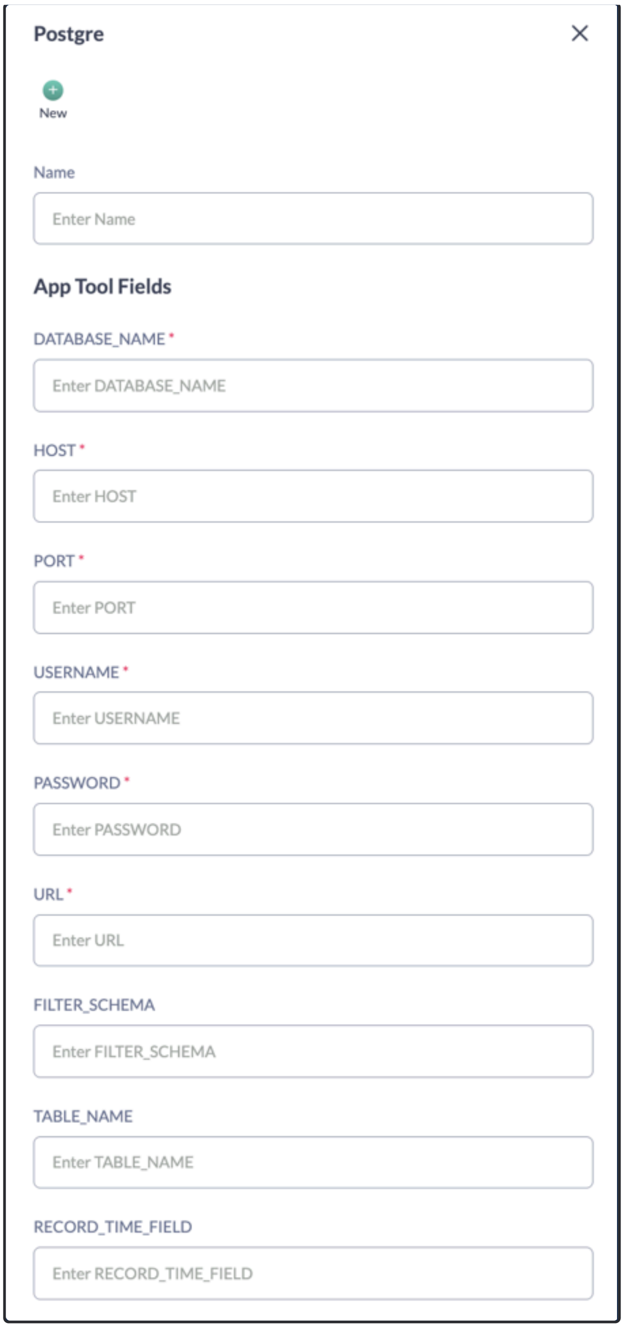
Postgres Data Source

To configure Postgres databases as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on Postgres with the required privileges to connect and query data through supported mechanisms. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



Step 3 of the Application Onboarding journey

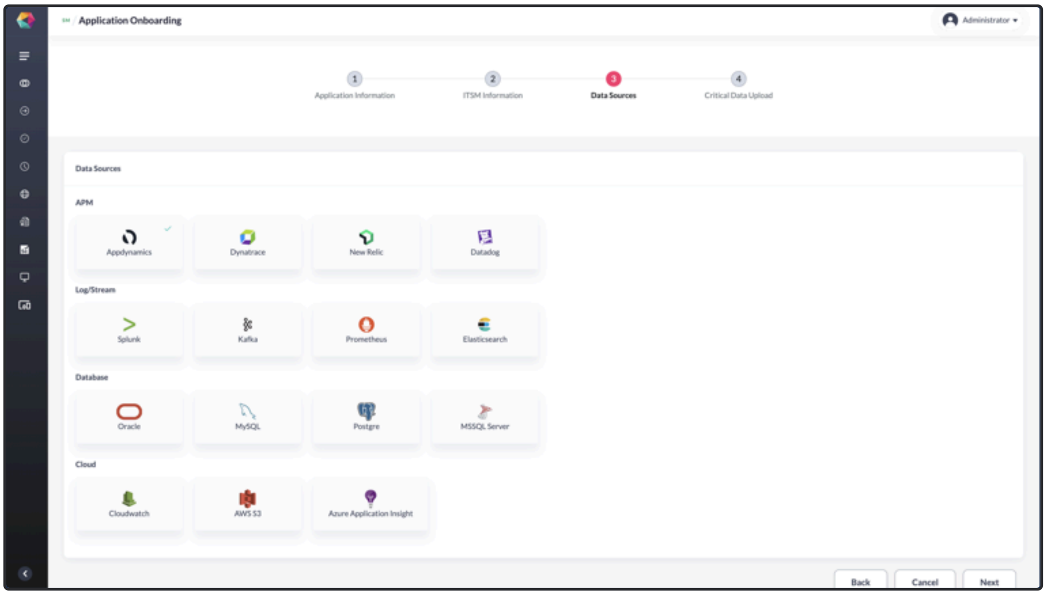
Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the Postgres host details along with the details required to construct the necessary queries for data retrieval. Enter these details accordingly to complete the configuration process as shown in the illustration below.



Postgres configuration

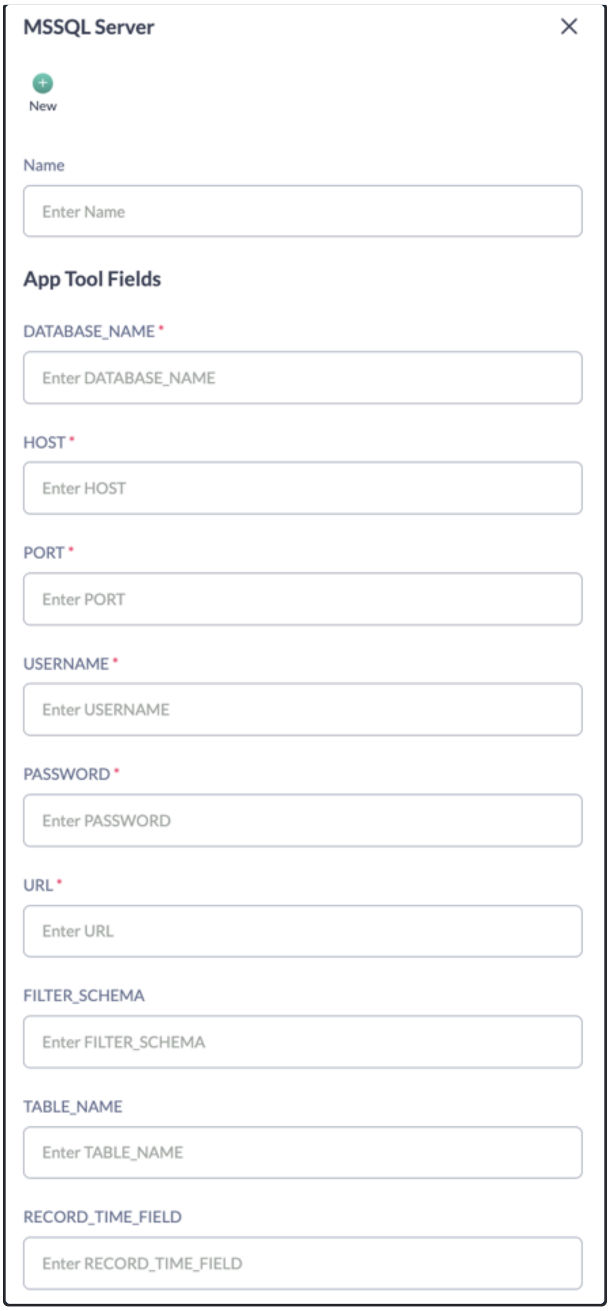
MSSQL Data Source

To configure MSSQL databases as a data source in Watermelon, follow *Step 3* of the Application Onboarding journey. Begin by ensuring that a read-only user account is created on MSSQL with the required privileges to connect and query data through supported mechanisms. For more detailed instructions, refer to the Application Onboarding documentation available at Onboarding Applications.



Step 3 of the Application Onboarding journey

Next, access the Application Onboarding section and click on "New" to add a new instance. Provide a descriptive name for the configuration that can be referenced in other Watermelon modules. For the integration, you will need the MSSQL host details along with the details required to construct the necessary queries for data retrieval. Enter these details accordingly to complete the configuration process as shown in the illustration below.



MSSQL Server

NGINX passthrough proxy configuration

Watermelon supports connectivity to endpoints for Testing via proxy configuration. This helps in reducing overheads in whitelisting and port opening and makes network management easier. Watermelon supports Squid proxy and NGINX proxy configuration in a forward proxy manner.

The below section explains the best suited NGINX configuration for Watermelon.

For RHEL and CentOS:

To install latest NGINX on Red Hat Enterprise Linux or CentOS, you can set up the official NGINX repository:

1. **Install the prerequisites:**
   sudo yum install yum-utils -y
2. **Set up the yum repository:**
   Create a file named /etc/yum.repos.d/nginx.repo with the following contents:
   [nginx-mainline]
   name=nginx mainline repo
   baseurl=http://nginx.org/packages/mainline/centos/$releasever/$basearch/
   gpgcheck=1
   enabled=1
   gpgkey=https://nginx.org/keys/nginx_signing.key
   module_hotfixes=true
3. **Install NGINX:**
   sudo yum install nginx -y
   sudo yum install -y nginx-mod-http-ssl -y
   nginx -v
4. **Download ngx_http_proxy_connect_module:**
   Based on the nginx version in step 3 download the ngx_http_proxy_connect_module from
   Index of /download/
   For example if nginx version: nginx/1.27.4 then download
   wget https://nginx.org/download/nginx-1.27.4.tar.gz
5. **Install ngx_http_proxy_connect_module:**
   $ tar -xvf nginx-1.27.4.tar.gz
   $ cd nginx-1.27.4

```
$ wget
https://github.com/chobits/ngx_http_proxy_connect_module/archive/refs/tags/v0.0.7.tar.gz
$ tar -xf v0.0.7.tar.gz
$ patch -p1 < ngx_http_proxy_connect_module-
0.0.7/patch/proxy_connect_rewrite_102101.patch
$ yum install -y redhat-rpm-config
$ yum groupinstall -y "Development Tools"
$ yum install -y pcre pcre-devel openssl openssl-devel gd gd-devel zlib patch libxml2-devel
libxslt-devel perl-devel perl-ExtUtils-Embed perl zlib-devel patch gcc-c++ gcc gcc-devel
unzip
$ ./configure --add-dynamic-module=<path>/nginx-1.27.4/ngx_http_proxy_connect_module-
0.0.7 --with-compat --with-http_ssl_module
$ make && make install
$ rm /usr/sbin/nginx
$ cp /usr/local/nginx/sbin/nginx .
$ cd /etc/nginx
$ mv nginx.conf nginx.conf.bkup
$ vim nginx.conf # add the below contents
—-------------------------------------------------------------------------------
```

```
user nginx;worker_processes auto;

load_module /usr/local/nginx/modules/ngx_http_proxy_connect_module.so;

error_log /var/log/nginx/error.log notice;

pid /var/run/nginx.pid;

events {

worker_connections 1024;

}

http {

include /etc/nginx/mime.types;

default_type application/octet-stream;

log_format main '$remote_addr - $remote_user [$time_local] "$request" '

'$status $body_bytes_sent "$http_referer" '
```

```
'"$http_user_agent" "$http_x_forwarded_for"';

access_log /var/log/nginx/access.log main;

sendfile on;

#tcp_nopush on;

keepalive_timeout 65;

#gzip on;

include /etc/nginx/conf.d/*.conf;

server {

listen 443 ssl;

# Self-signed certificate

#ssl_certificate_key /root/nginx/dev.watermelon.internal/cert.key;

#ssl_certificate /root/nginx/dev.watermelon.internal/cert.crt;

#ssl_session_cache shared:SSL:1m;

ssl_certificate_key /root/nginx/app.watermelon.us/certificate.key;

ssl_certificate /root/nginx/app.watermelon.us/certificate.crt;

ssl_session_cache shared:SSL:1m;

# DNS resolver used by forward proxying

resolver 8.8.8.8;

# Forward proxy for CONNECT request

proxy_connect;

proxy_connect_allow 443 563;

proxy_connect_connect_timeout 10s;

proxy_connect_data_timeout 10s;

# Reverse proxy for non-CONNECT requests

location / {
```

proxy_pass http://$host;

proxy_set_header Host $host;

}

}

}

——--

Start the nginx proxy:

$ systemctl start nginx

$ systemctl enable nginx

$ systemctl status nginx

nginx.service - nginx - high performance web server

Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; vendor preset: disabled)

Active: active (running) since Fri 2025-02-21 03:15:20 +08; 41min ago

Docs: nginx documentation

Main PID: 10097 (nginx)

Tasks: 9 (limit: 100624)

Memory: 11.0M

CGroup: /system.slice/nginx.service

├─10097 nginx: master process /usr/sbin/nginx -c /etc/nginx/nginx.conf

├─10098 nginx: worker process

├─10099 nginx: worker process

├─10100 nginx: worker process

├─10101 nginx: worker process

├─10102 nginx: worker process

├─10103 nginx: worker process

├─10104 nginx: worker process

└─10105 nginx: worker process

Feb 21 03:15:20 hostname systemd[1]: Starting nginx - high performance web server...

Feb 21 03:15:20 hostname systemd[1]: Started nginx - high performance web server.

$ nginx -V

nginx version: nginx/1.27.4

built by gcc 8.5.0 20210514 (Red Hat 8.5.0-23) (GCC)
built with OpenSSL 1.1.1k FIPS 25 Mar 2021

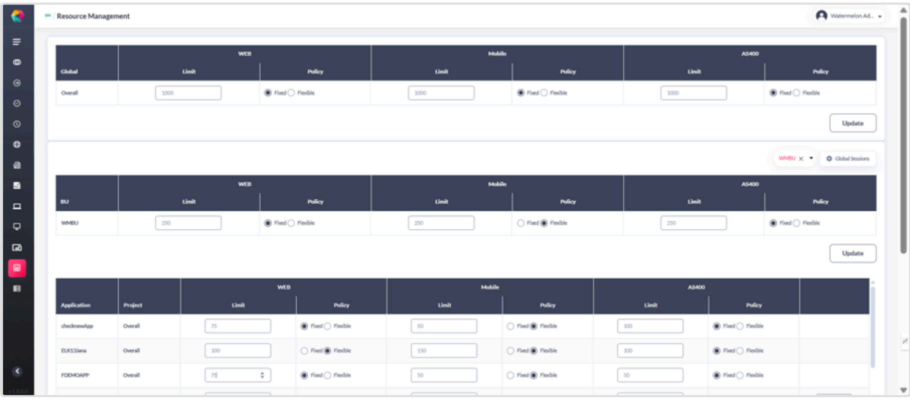
TLS SNI support enabled
configure arguments:

--add-dynamic-module=/root/nginx/nginx-1.27.4/ngx_http_proxy_connect_module-0.0.7 --with-compat --with-http_ssl_module

Reference: GitHub - chobits/ngx_http_proxy_connect_module: A forward proxy module for CONNECT request handling

Resource Management

This article explains how to configure the **Resource Management** module so that Admin users can control concurrent test executions across the organization.



1. Purpose and scope

Resource Management lets Admin users set **concurrency limits** for automated runs across three channels:

- **Web**
- **Mobile**
- **Mainframe (AS400)**

Limits are managed in a hierarchy:

> ***Global → Business Unit (BU) → Application → Project → Group → Sub-group***

Each level can have its own limits, but every configuration must stay within the capacity granted by its parent level.

Only users with Admin rights can view and edit this module.

---

2. Key terms

Limit

For each level and channel, **Limit** is the **maximum number of concurrent sessions that level is allowed to use**.

Examples:

- Global Web Limit = 100
- BU “Retail Banking” Web Limit = 60
- Application “Online Banking” Web Limit = 30

Policy

Every Limit has a **Policy**:

- **Fixed** – a reserved share of the parent’s capacity for that level and its children.
- **Flexible** – a share of the parent’s capacity reserved to be **used flexibly by that level’s children**.

Fixed and Flexible are two ways of carving out portions of the parent's Limit. A level may also have **no policy selected** (see section 5.6).

---

3. Core configuration rule

For any parent level and channel, Resource Management enforces this rule at save time:

> ***The sum of all child Fixed and Flexible limits for that channel must not exceed the parent's Limit.***

If this condition is not met, saving is blocked and an error message is shown.

This rule applies at every link in the hierarchy:

- Global → BU
- BU → Application / Project
- Application → Project
- Project → Group
- Group → Sub-group

---

4. Policy inheritance

Policies are constrained by their parents.

Parent is Fixed

If a parent level has **Fixed** policy for a channel:

- Its children for that channel may be **Fixed or Flexible**, as long as the sum of their Limits (Fixed + Flexible) does not exceed the parent Limit.

Parent is Flexible

If a parent level has **Flexible** policy for a channel:

- All of its children for that channel **must also be Flexible**.
- A child cannot be set to Fixed if its parent is Flexible.

This applies throughout the hierarchy. For example:

- If Global Web is Flexible → all BUs for Web must be Flexible.
- If a BU is Flexible for Web → all Applications/Projects under that BU for Web must be Flexible, and so on.

---

5. How limits work across levels

Below, "channel" means Web, Mobile, or Mainframe.

5.1 Global level

At the top of the page:

- For each channel, you configure:
  - **Global Limit** – total maximum concurrency for that channel.
  - **Global Policy** – Fixed or Flexible.

All lower-level limits for that channel must fit within the Global Limit when their Fixed and Flexible values are added together.

5.2 Business Unit (BU) level

In the BU section:

- Each BU can have a Limit and Policy per channel.

Rules:

- For a given channel, the **sum of Fixed + Flexible limits of all BUs** must be ≤ the Global Limit.
- If Global is Flexible for a channel, all BUs for that channel must be Flexible.

Meaning:

- BU **Fixed** – part of the Global capacity reserved for that BU and its children.
- BU **Flexible** – part of the Global capacity reserved to be shared flexibly by that BU's children.

5.3 Application and Project level

In "Applications & Projects":

- Each row combines an Application and a Project.
- For each channel you set:
  - **Limit** – numeric.
  - **Policy** – Fixed or Flexible.

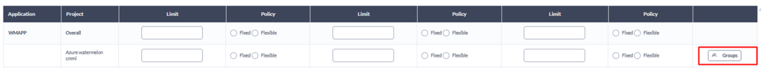
Rules:

- For a BU and channel, the **sum of Fixed + Flexible limits of all Applications/Projects in that BU** must be ≤ the BU Limit.
- If the BU is Flexible for a channel, all Applications/Projects under it must be Flexible for that channel.

Meaning:

- Application/Project **Fixed** – part of the BU's reserved capacity dedicated to that Application/Project subtree.
- Application/Project **Flexible** – part of the BU's flexible share that can be used by that Application/Project's children.

5.4 Groups and Sub-groups

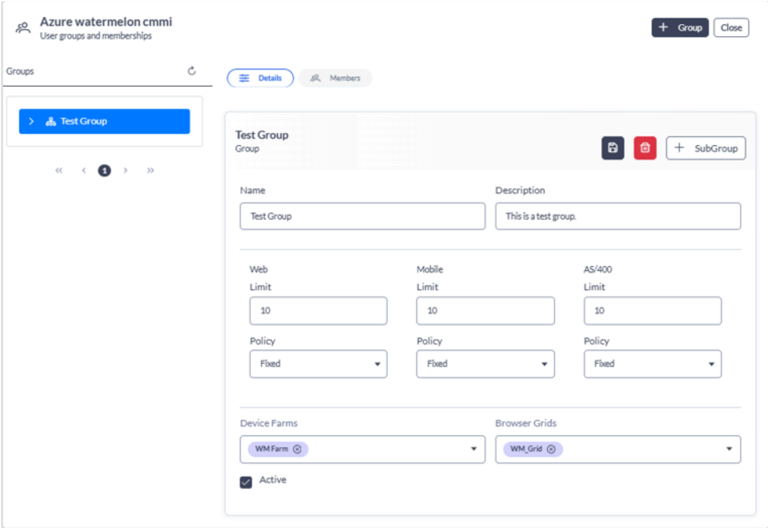


From an Application/Project row, selecting **Groups** opens the Groups panel:

- You can define **Groups** and optional **Sub-groups**.
- Each Group or Sub-group has a Limit and Policy per channel.

Rules:

- For a Project and channel, the **sum of Group Fixed + Flexible limits** must be ≤ the Project Limit.
- For a Group and channel, the **sum of Sub-group Fixed + Flexible limits** must be ≤ the Group Limit.
- If a Group is Flexible for a channel, all Sub-groups under it must also be Flexible for that channel.



5.5 User precedence between Group and Sub-group

If the **same user** is assigned both:

- to a **Group**, and
- to one of its **Sub-groups**,

then:

- The **Sub-group limit** is the effective cap for that user.
- Group-level limits still apply to users who are only in the Group.

5.6 Children with no policy configured

At any level and channel:

- If a parent has a Limit and Policy configured, and
- Some of its children **do not** have a Limit/Policy set for that channel,

then:

- Those unconfigured children are treated as having a **flexible share of the parent's remaining capacity**.
- They can still run tests, drawing from whatever capacity is left

Hope this helps you to gain deeper understanding into the Resource Management module! Please check out the worked examples in the next article for more details.

Worked Example

This example shows how to configure Resource Management for **two Business Units**, a couple of applications, projects, and two groups with different needs.

---

1. Global and Business Units

Scenario

You have:

- **BU1**
- **BU2**

Total available Web executors: **80**.

Step 1 – Global Web

- Global Web **Limit** = 80
- Global Web **Policy** = Fixed

All Web limits configured under BU1 and BU2 must fit into these 80 executors.

Step 2 – BU Web limits

You want BU1 to have more capacity than BU2.

BU	Web Limit	Policy	Meaning
BU1	60	Fixed	60 Web executors reserved for BU1 and its children.
BU2	10	Flexible	10 Web executors reserved to be shared flexibly by BU2’s children.

Check:

- BU1 (60 Fixed) + BU2 (10 Flexible) = **70 ≤ 80** → valid.
- The remaining 10 Web executors at Global can be used by any BU with no specific limits.

---

2. Applications and Projects inside BU1

Inside **BU1**, you have one application with two projects:

- **App1 / Proj1-A**
- **App1 / Proj1-B**

BU1 Web Limit: 60 (Fixed).

Step 3 – App/Project Web limits

You want Proj1-A to have more capacity than Proj1-B, and still leave some room for future work.

Application	Project	Web Limit	Policy	Meaning
App1	Proj1-A	35	Fixed	35 Web executors reserved for Proj1-A.
App1	Proj1-B	10	Flexible	10 Web executors reserved to be used flexibly under Proj1-B.

Check under BU1:

- 35 (Fixed) + 10 (Flexible) = **45 ≤ 60** → valid.
- 15 Web executors remain at BU1 level for other App/Project combinations without explicit limits.

---

3. Applications and Projects inside BU2

Inside **BU2**, you have:

- **App2 / Proj2-A**

BU2 Web Limit: 10 (Flexible).

Step 4 – App/Project Web limits for BU2

Application	Project	Web Limit	Policy	Meaning
App2	Proj2-A	8	Flexible	8 Web executors reserved to be used flexibly under Proj2-A.

Check under BU2:

- 8 (Flexible) ≤ BU2 Web Limit 10 → valid.
- 2 Web executors remain at BU2 level for any additional projects.

---

4. Groups and Sub-groups inside App1 / Proj1-A

Focus now on **App1 / Proj1-A** (Web limit 35 Fixed).

You want to distinguish **two groups**:

- **Group1** – higher priority, needs more executors regularly.
- **Group2** – lower priority, needs only a few executors occasionally.

Step 5 – Group Web limits

Group	Web Limit	Policy	Meaning
Group1	20	Fixed	20 Web executors reserved for Group1 under Proj1-A.
Group2	5	Flexible	5 Web executors reserved to be

			used flexibly by Group2.

Check under Proj1-A:

- 20 (Group1 Fixed) + 5 (Group2 Flexible) = **25 ≤ 35** → valid.
- 10 Web executors remain at Proj1-A level for any additional groups or unconfigured usage.

How this behaves:

- Group1 can use up to 20 Web executors; these are effectively its main share.
- Group2 can use up to 5 Web executors when capacity is available, but it is clearly limited and lower-capacity compared to Group1.
- Any other group added later (or with no explicit limit) will share the remaining capacity, bounded by the same 35 for Proj1-A.

This example stays fully generic (BU1/BU2, App1/App2, Proj1-A, Group1/Group2) and demonstrates:

- Different capacity for two BUs
- Different capacity for two projects under App1
- A 'larger' higher-priority group (Group1) and a “smaller” lower-priority group (Group2) inside the same project.

Taken together, this single setup shows:

- Two Business Units (BU1 and BU2) with different overall Web capacity.
- Two projects under App1 with different shares inside BU1.
- Two groups under Proj1-A where Group1 gets a bigger, fixed share and Group2 gets a smaller, flexible share used only occasionally.

Resource Utilization

Resource Utilization shows **how many executors are in use vs. configured limits**, at any level of your hierarchy, for the three platforms:

- Web
- Mobile
- Desktop

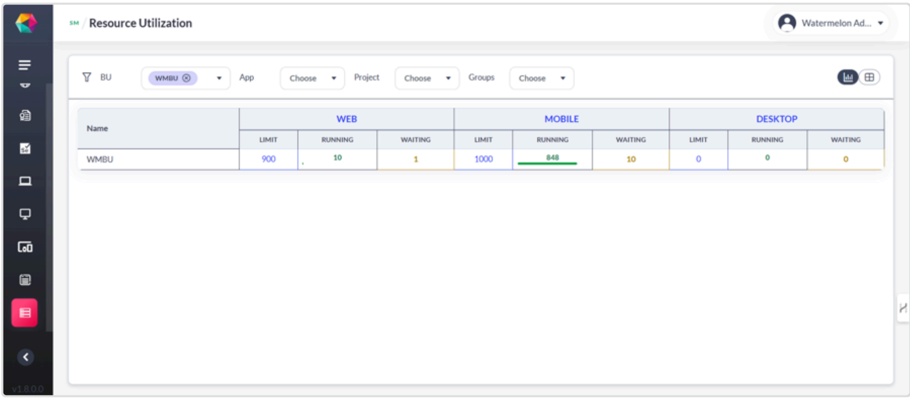
Each row represents one level (Global, BU, App/Project, or Group), depending on the filters you choose.

---

1. Columns and metrics

For each visible row you see, per platform:

- **LIMIT** – the concurrency limit configured in Resource Management at that level.
- **RUNNING** – how many sessions are currently executing at that level.
- **WAITING** – how many sessions are queued because no executor is currently available under the applicable limits.



Example interpretation for BU=WMBU:

- LIMIT=900, RUNNING=10, WAITING=1 for Web means:
  - You can run up to 900 Web sessions.
  - 10 are running now.
  - 1 is waiting for capacity to free up.

---

2. Filters and drill-down

The filter bar at the top controls **what level** you are looking at:

- **BU** – Business Unit
- **App** – Application
- **Project**
- **Groups**

Behavior:

- With **no filters**, the table shows **Global** utilization (one row).
- When you select **BU** only, the table shows utilization aggregated at **BU level**.
- When you select **BU + App**, the table shows utilization for that **App/Project inside the BU**.
- As you keep adding **Project** and **Groups**, the view drills down further to those levels.

Each time you change a filter:

- The data is refreshed.
- LIMIT/RUNNING/WAITING are recalculated for the currently selected scope.

---

3. Detailed view (runs list)

When both **BU and App** are selected, an additional **detailed view** becomes available (the list view icon).

This view shows individual runs with columns such as:

- BU
- App / Project
- Run / Task IDs
- Test Case / Platform
- Level / Policy (which level's limit is controlling the run, and whether it is Fixed or Flexible)
- State / Created time

Use this view to answer:

- "Which specific runs are using capacity right now?"
- "At which level (BU, Project, Group) is a particular run counted?"
- "Is a run governed by a Fixed or Flexible policy?"

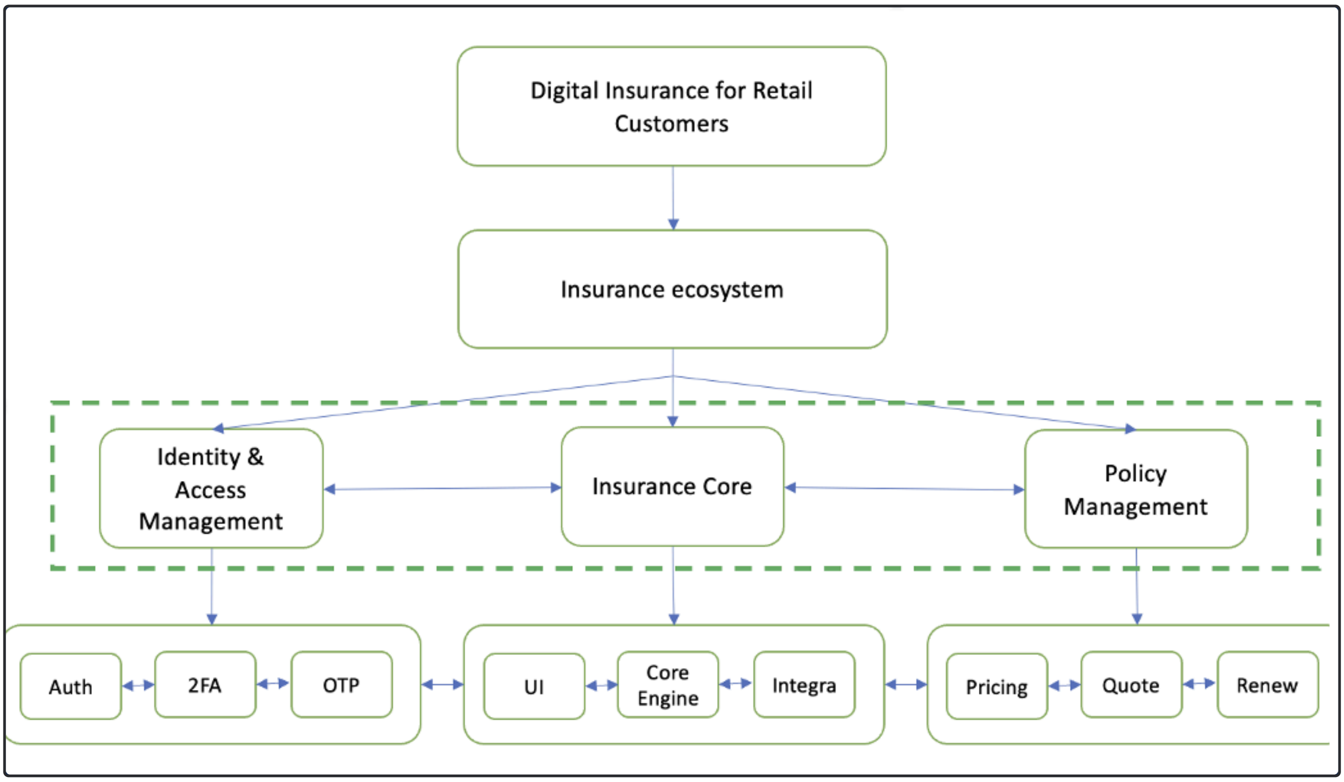
---

4. How to use Resource Utilization in practice

- **Check overall health**
  - Leave filters empty to see Global LIMIT/RUNNING/WAITING per platform.
  - If WAITING is high at Global, you are hitting overall capacity.
- **Find hotspots**
  - Filter by BU to see which BU is saturating its own limits.
  - Add App/Project to see which application or project is causing queues.
- **Relate back to Resource Management**
  - If one BU or App is always close to its LIMIT with many runs waiting, consider increasing its limit (if Global capacity allows) or adjusting Fixed vs Flexible settings in Resource Management.

Licensing

Watermelon licensing is based on a per application level providing numerous benefits to the customer. A simple illustration on the licensing approach is shared below.



Watermelon licensing is based at the tier in the **green box**. In this illustration, customer will only need to procure 3 licenses. Each of these systems need to be identified by a unique identifier such as an Application Name or Application Code (short-name).

Benefits of an application based license approach

No limits on the number of services that an application may comprise

o With a focus for most organizations to evolve towards Cloud Native, there will be more and more micro services added to their ecosystem. Our licensing makes the evolution immune to cost as the licensing happens at a higher level.

No limits on the number of users that need to access Watermelon

o Our philosophy is not only to drive reliability but also efficiency. No limits on users means organizations can fully benefit from wide and extensive usage of our enablers.

No restriction on the size of application or components

o Each application would comprise of various components such as Web components, App components, Middleware components, DB components, etc and our licensing ensures no limits on the number of components that are required for an application.

Single licensing scheme making it environment agnostic

o As Watermelon aims to make a positive impact across both non-production and production environments, we provide a single license that allows implementation of our modules across all environments.

Simple licensing scheme that is easy to understand and adopt

o Minimal to No complexity in the licensing scheme makes it easier for customers to track, manage and plan for subsequent expansion.

Highly reusable and flexible

o Watermelon licenses can be transferred internally (example: a decommissioned application's license could be used by another application) by the customers without our involvement as long as the count of applications licensed is maintained.

Type	Definition	Covered under license (Y/N) ?
Module	A module refers to a self-contained, independent component of a larger system or platform. It is designed to perform a specific function or set of functions, and can be integrated with other modules to create a complete system or platform.	Yes
Capability	A capability refers to a specific function or feature that a product or system is designed to perform. Capabilities are usually defined in	Yes

	terms of the outcomes or benefits they provide to the user or customer. They represent the specific ways in which the product or system can be used to meet user needs or solve user problems.	
Enhanceme nt	An enhancement refers to a new feature or improvement made to an existing product or service. Enhancements are designed to add value, increase functionality, or improve the overall user experience of the product. An enhancement may be made in response to customer feedback, market trends, or other factors that impact the product's performance or appeal.	Yes
Defect/Bug fix	A defect/bug refers to any deviation from the specified requirements or standards of a product that results in its failure to meet customer expectations or needs.	Yes
Black-box	Blackbox refers to a module or capability or enhancement that is tailored or customized to meet the specific needs or requirements of a particular customer or client. This is subject to Watermelon’s assessment (viable for customer, viable for Watermelon and value to broader market) and sole discretion to deliver as part of the licensing agreement with the ownership of Intellectual Property with Watermelon.	Yes