

Tidal Repository

Manage all workload automation schedules across instances from a central location.

KEY BENEFITS

Maintains a central record of all workload automation schedules and calendars

Enables versioning of schedule changes to simplify management of updates

Integrates with DevOps processes and accelerates implementation of new application and schedule updates

PRODUCT OVERVIEW

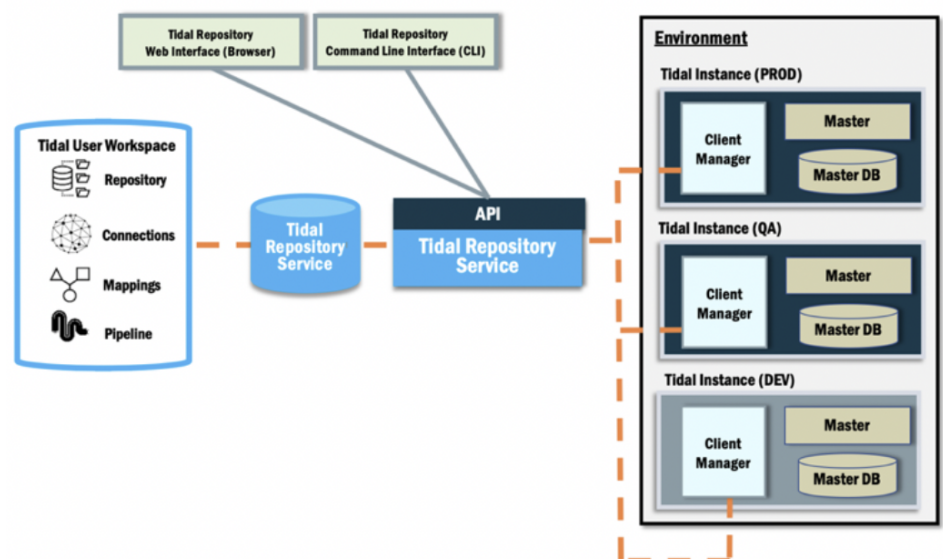
Tidal Repository™ provides centralized management for workload automation schedules and calendars. Designed for use by administrators and developers, the Repository application offers advanced capabilities for storing, organizing, sharing and using job data across multiple Tidal instances. It supports best practices for change management and maintains a complete history and auditable record of revisions.

BUSINESS CHALLENGES

Modern workload automation environments incorporate multiple instances of Tidal Automation – development, QA, production, etc. – with a multitude of job definition configurations. It is resource-intensive to manage and synchronize schedule and calendar updates, or to replicate jobs across instances while ensuring integrity and security of workloads. Additionally, with ongoing application development and updates, DevOps processes can also be slowed while coordinating associated schedule updates to be tested and implemented.

Centralize and lock down control of your schedules

Tidal Repository moves and shares data across multiple Tidal Automation instances.





Tidal Repository offers advanced capabilities for administrators and developers.

FOR ADMINISTRATORS

Centralized management of schedules and calendars. Create or update a job or set of jobs once and replicate the changes easily and seamlessly across multiple instances. Even revert to a prior version in the event of misconfiguration. Repository captures and retains a history of who made what changes and when, supporting change management best practices.

Controlled access and security. Manage user access effectively and securely. Additionally, workspaces are created – either public or private – where users or workgroups can access and manipulate their own schedules.

Support for DevOps processes. Reduce wait times and bottlenecks for new application updates when Administrators give developers the ability to edit and test their own jobs.

Single source of truth for schedule change management. The comprehensive history maintained in the Repository is a valuable resource for satisfying regulatory and compliance audit requirements.

Export/import capability. Easily export a schedule into XML and import into other systems or applications.

FOR DEVELOPERS

Integration with application development. Synchronize application development and associated schedule changes for more efficient continuous delivery. These new efficiencies can drive an increase in development velocity.

Self-service without risk. Test and run “what-if” scenarios in a dedicated workspace with data that mirrors the Tidal Master environment, reducing risk to the integrity of production schedules and calendars.

REPOSITORY vs. TRANSPORTER

Customers currently using Transporter may wonder what’s different with Repository.

	Repository	Transporter
Access	Resides centrally on a server and accessed via web-based browser.	Resides on a workstation so access is limited to individual user(s).
Movement of Data	Retains central record of all scheduling data changes across all instances.	Acts as a pipe to move scheduling and calendar data. Once moved, the data no longer exists in Transporter.
Making changes	Changes made in workspaces that mirror the Master schedule; all changes versioned with a complete log record.	Moves the changes but does not retain any changes.

Repository is the successor to Transporter.

