Kofax Kapow 10.3 Training and Certification

Module 9 – The Development Database and Management Console

Storing Data/Intro to Kapow Administration

Kofax Kapow™

KOFAX

Kofax Kapow 10.3 Module Overview

- An overview of the Development Database
- Starting the Development Database
- Mapping Output to Database Table
- Storing Data in the Database
- Disabling Steps for Testing
- Uploading Robot to the Management Console
- The Management Console: An Overview
- Running a Robot in Management Console
- Scheduling Robots
- Viewing Data in Management Console
- Performance Graphs
- Licensing

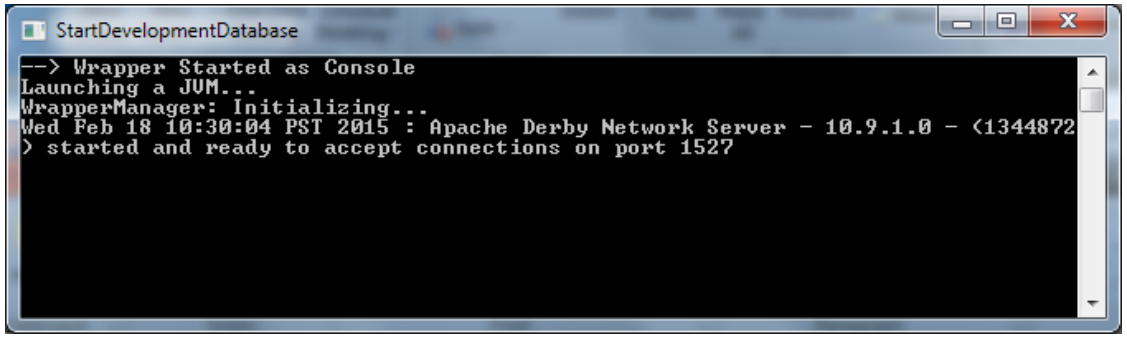
The Development Database

- Kofax Kapow includes an Apache Derby Development Database
  - This database is used for training purposes when learning about Kapow
  - It may also be used for testing, but is not a substitute for a full SQL, Oracle, JDBC etc. database in actual daily use.
  - IMPORTANT! Any database other than the Development Database requires that a JDBC driver be installed to use it.
- To use the Development Database, it must first be started.
- Output variables in your Robot must be mapped to the database.

Starting the Development Database

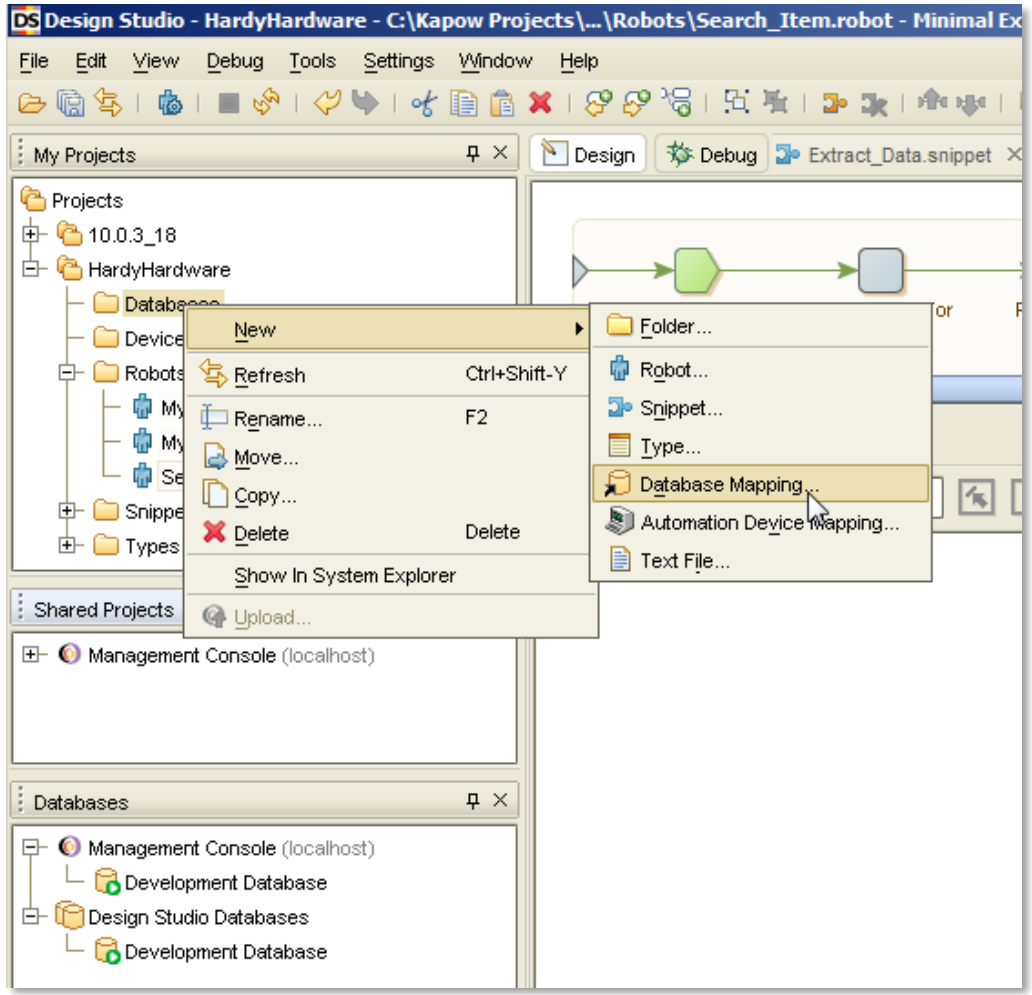
- There are shortcuts in both Windows and Linux versions to start the development database.
  - On Windows systems, it is in the Kapow 10.x.x\Development Database program group
  - On Linux systems, it is in the Kapow/bin folder
- The Development Database connects using port 1527 on both systems
- You must start the Development database before you can read from or write to it.

IMPORTANT: Do not close command window or you'll close the connection.

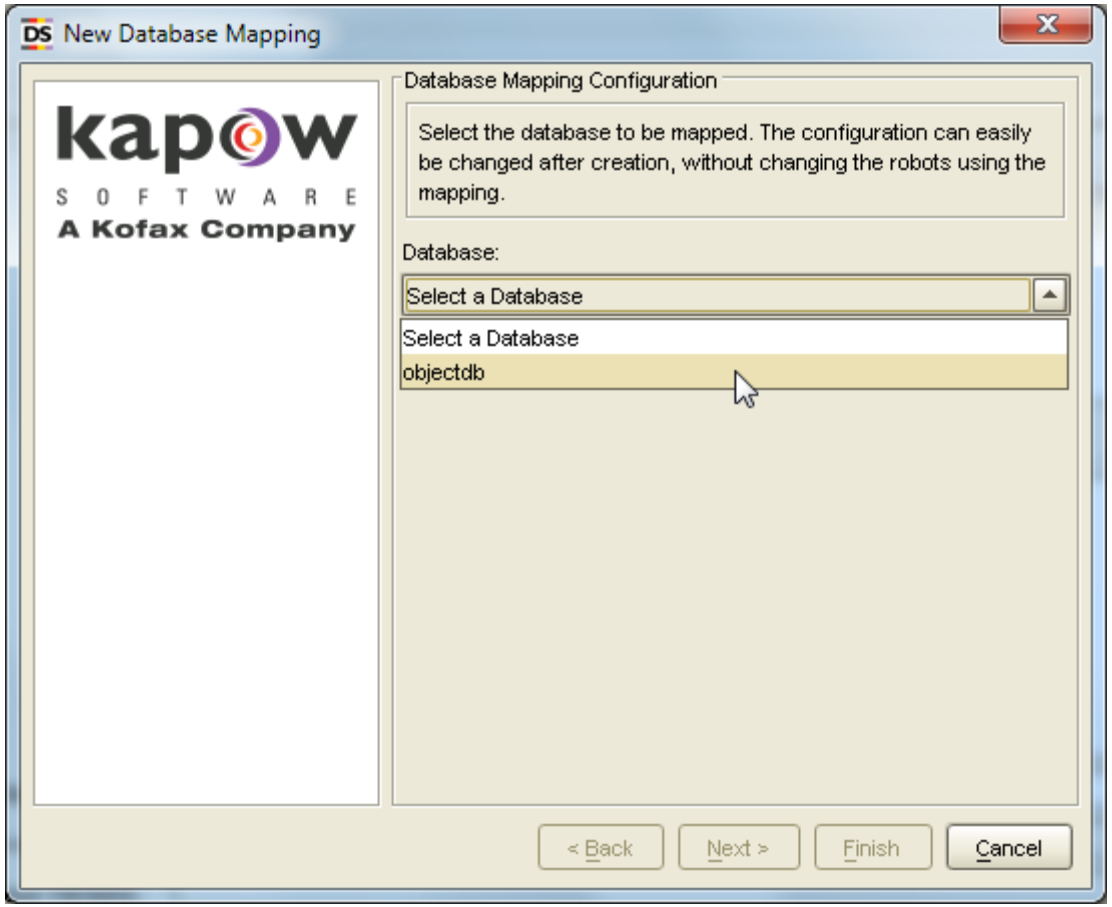


Storing Data in the Database

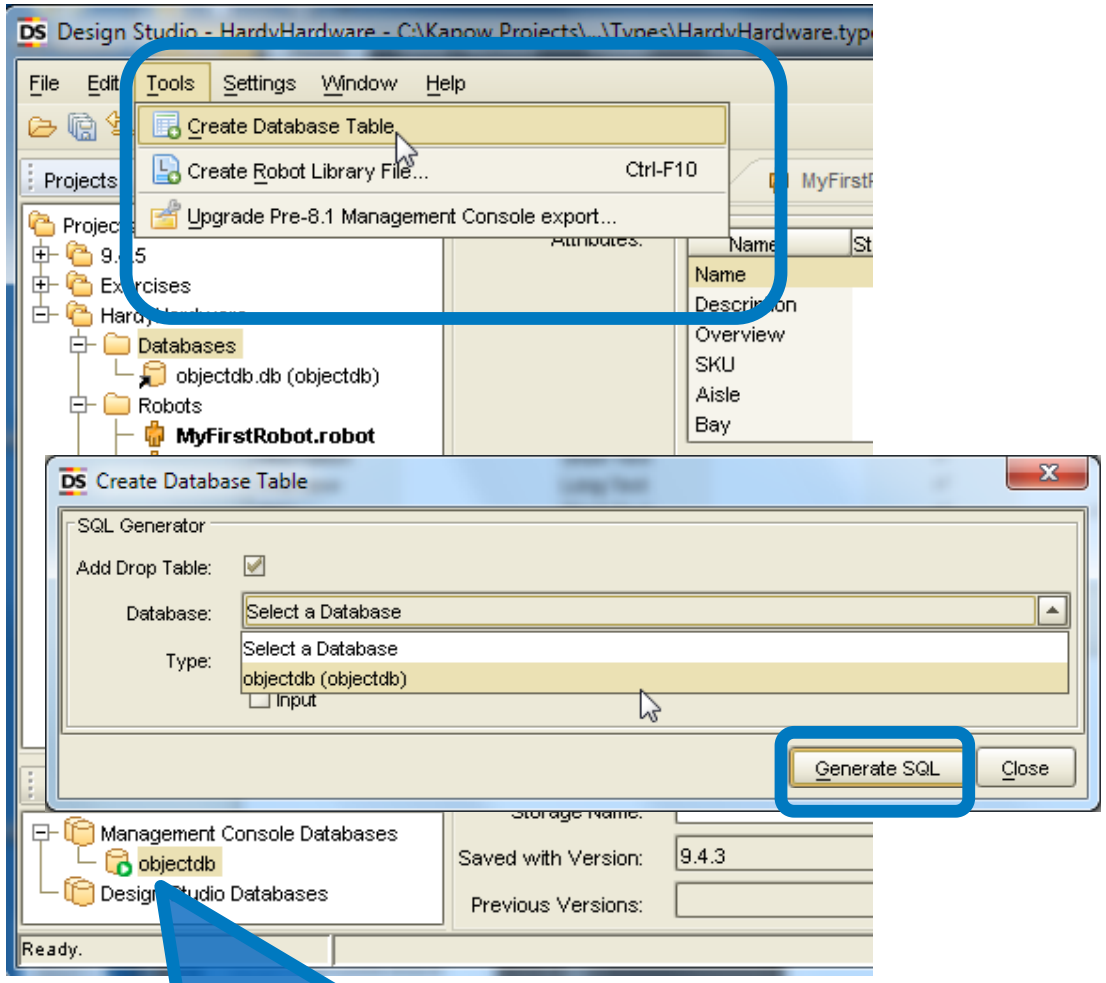
- Currently we've started RoboServer and the Management Console. Remember, the Management Console provides our license.
- And we've started the Development Database.
- Both command windows are still open but may be minimized.
- Let's go back to our last Project in Design Studio and map the attributes in our HardyHardware Type to a table in the Development Database

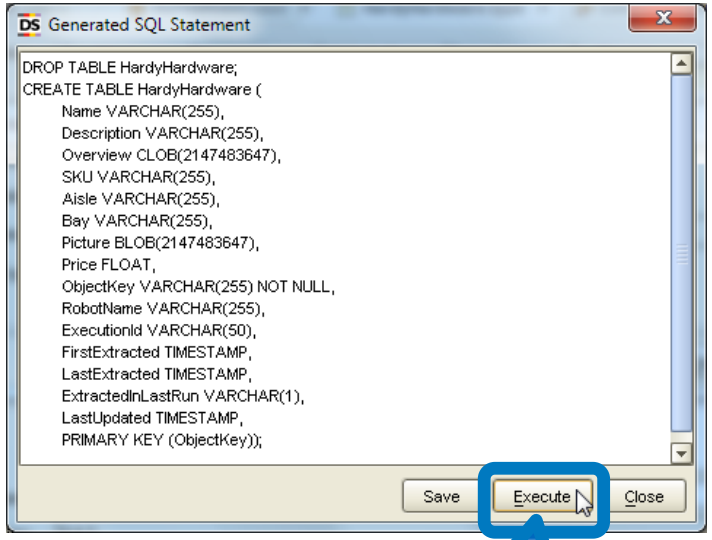


Select a Database



Create Table

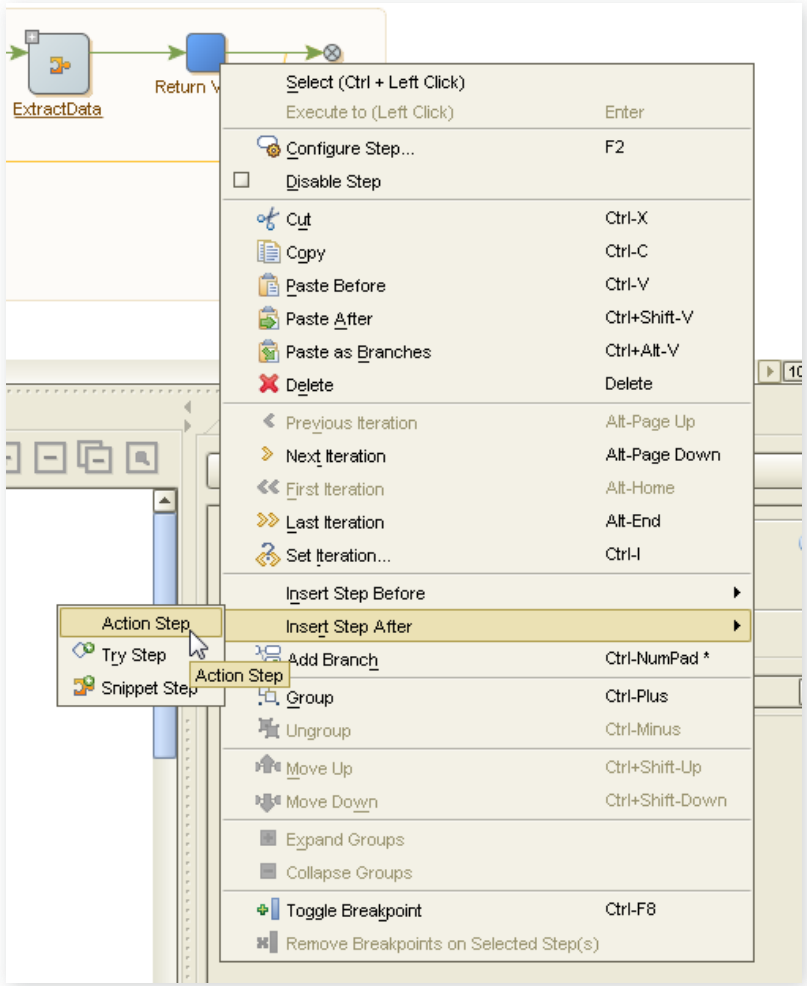




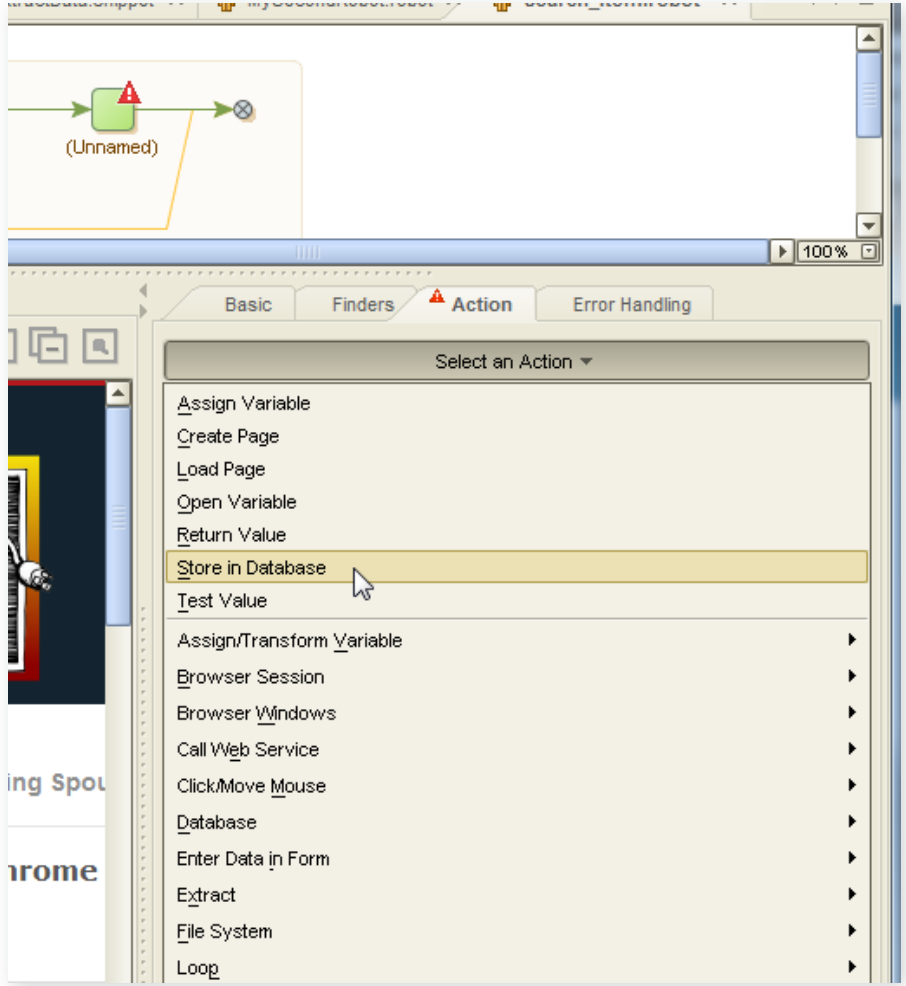
Notice green arrow: Indicates database is running and we're connected

The [Execute] button executes the SQL statement shown in the box presented and creates the columns in the HardyHardware table.

Create New Action Step to Store Data in Database



Use "Insert Step After" and Action Step

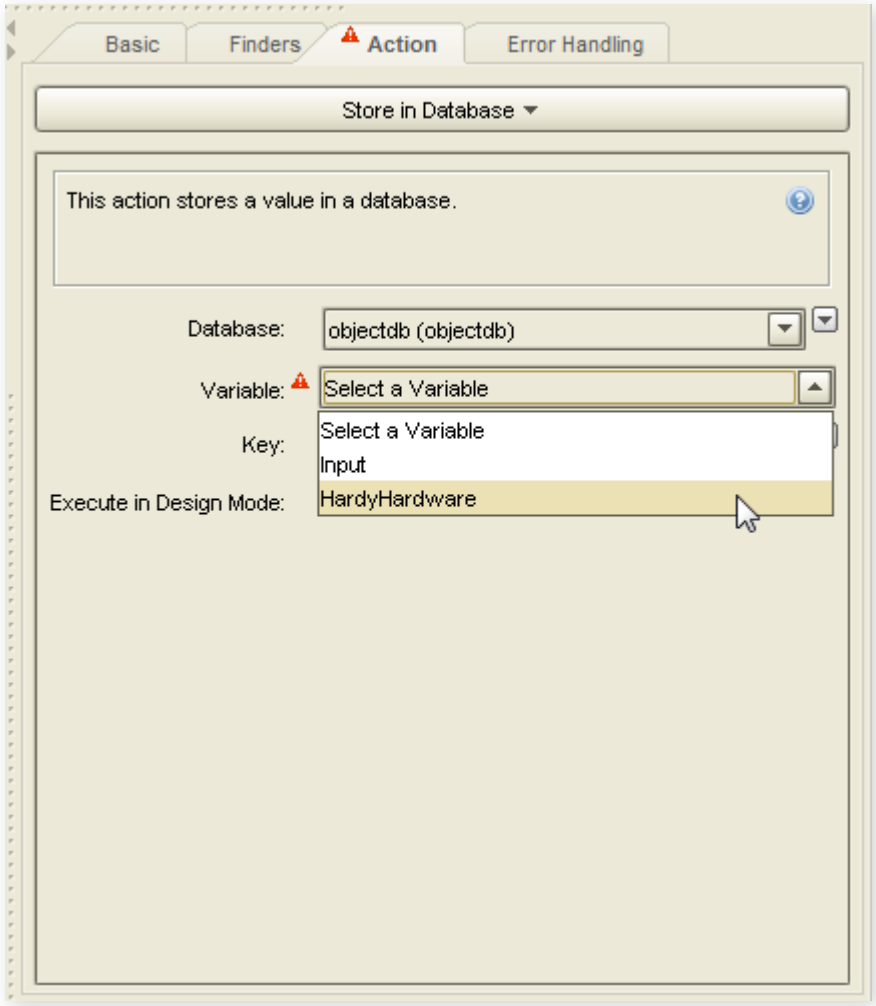


and select "Store in Database"

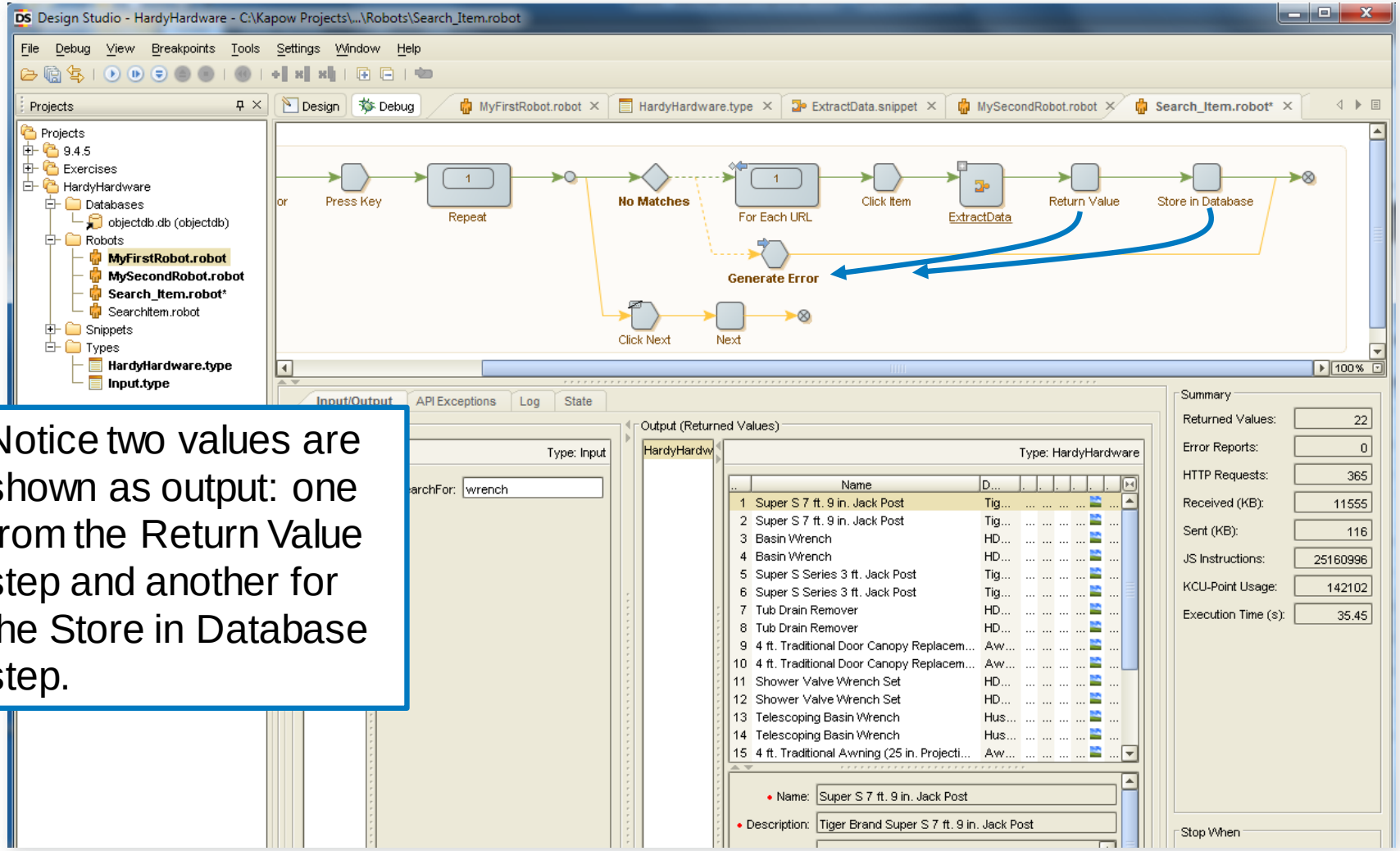
Select a Variable for Storage

- On the Action tab, you must select a Variable for output. Here, we've selected HardyHardware from the dropdown.

- We're not going to output the Input variable.



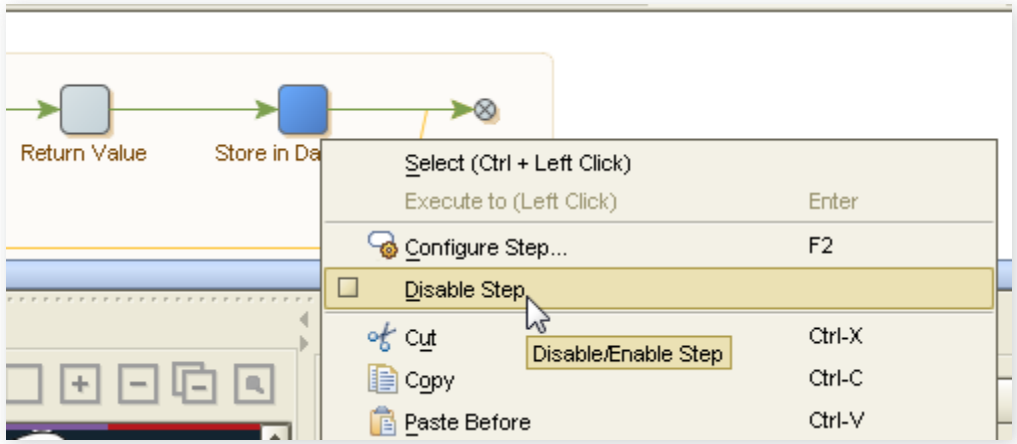
Testing in Debug Mode



Notice two values are shown as output: one from the Return Value step and another for the Store in Database step.

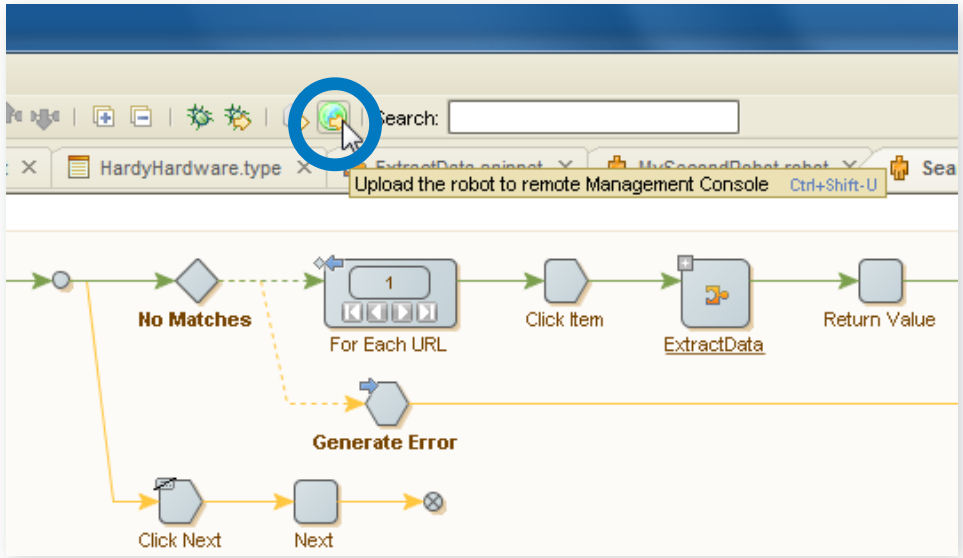
Disabling Steps for Testing

- If you test the Robot through the end step or if you run it in Debug mode, you will be writing data to the database. You may not want to do that.
- You may disable this step (or any other step) by right mouse-clicking on the step and selecting "Disable Step" from the context menu. Remember to re-enable it when you want to store the data. *At that point, you might want to delete the Return Value step if it's no longer needed.*

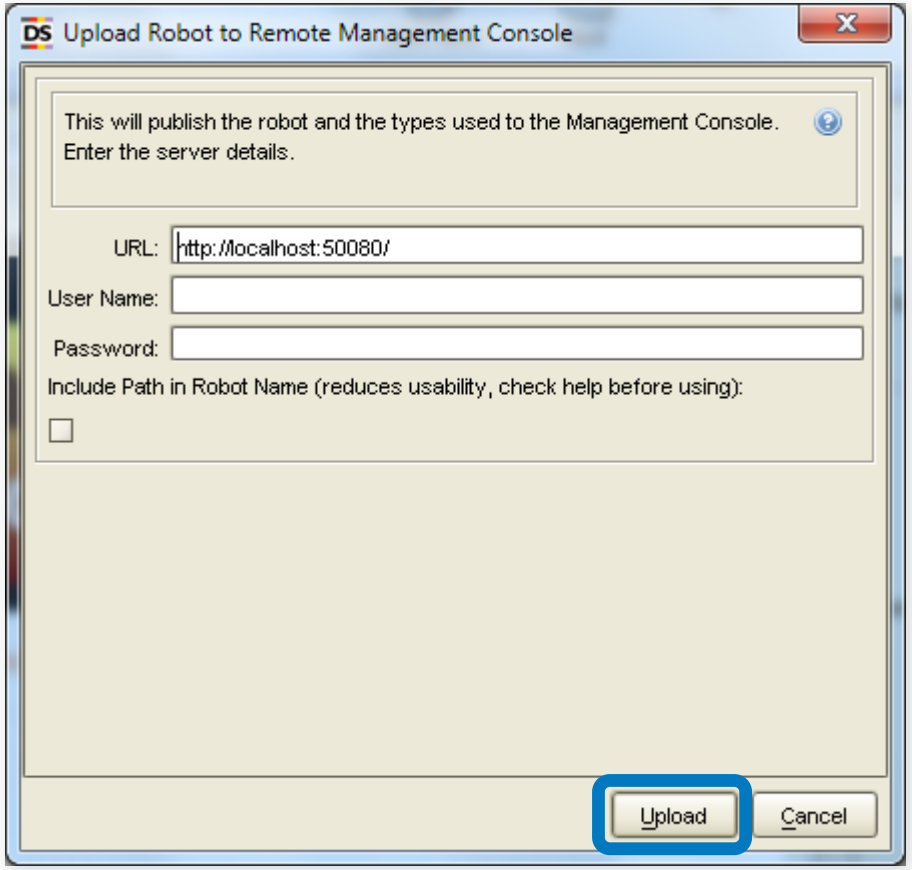


Uploading Robot to the Management Console

- Uploading the Robot to the Management Console "publishes" the Robot making it available for use
  - You may run the Robot manually in the Management Console
  - You may set the Robot to run on a scheduled bases
  - You may create a "Kapplet" to allow users to run it from their workstation (more about Kapplets later).

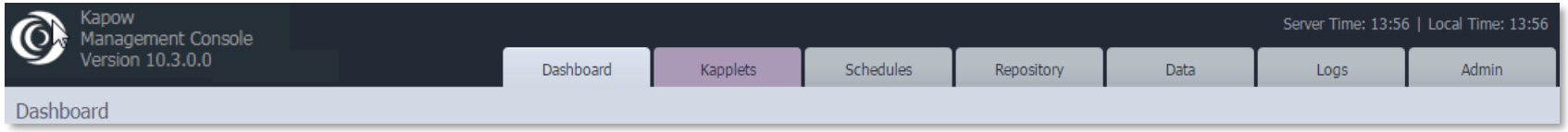


Specify the URL and Logon Credentials



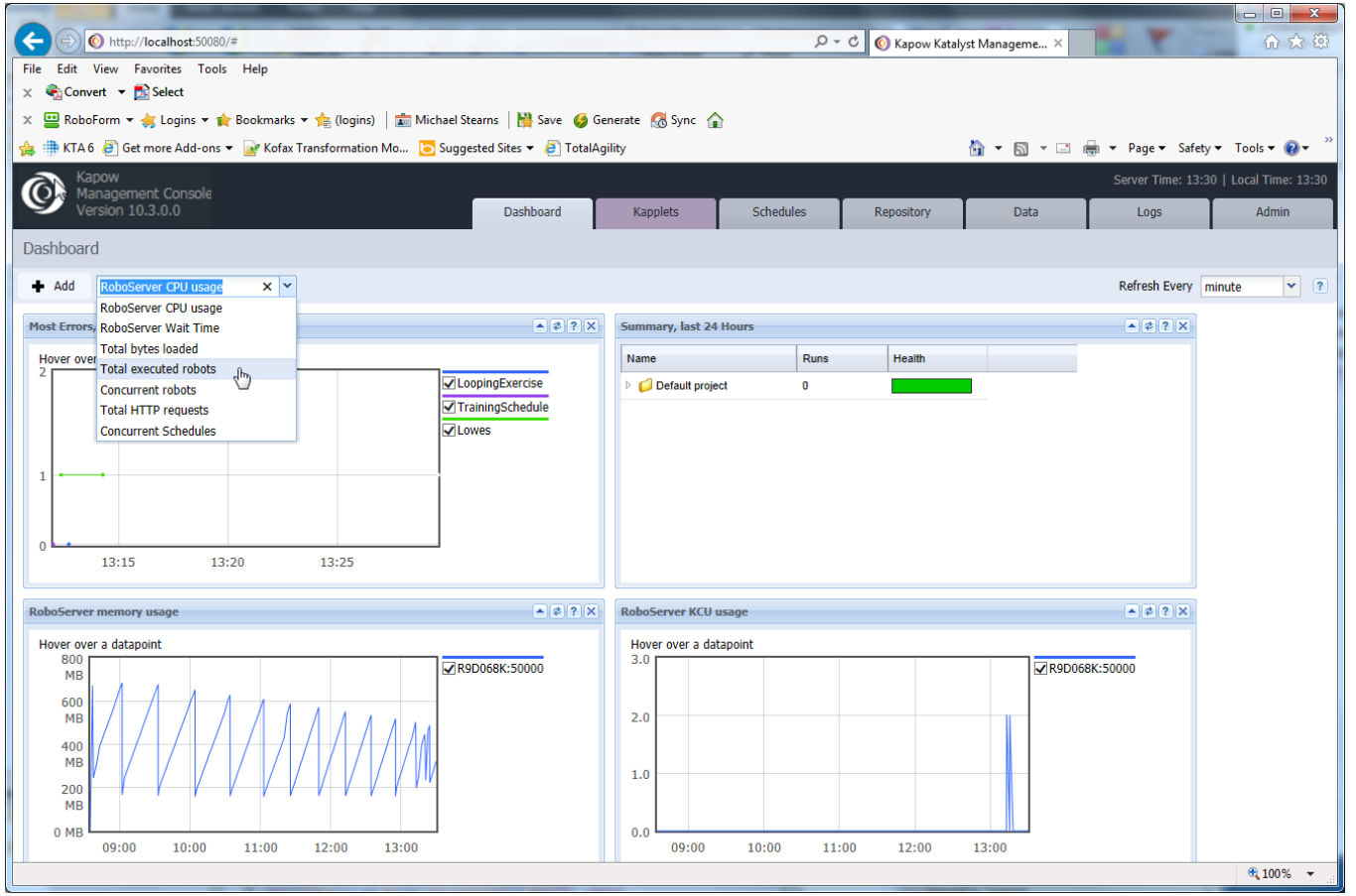
The Management Console: An Overview

- The Management Console is displayed in a web interface (http://URL:50080)
- It is divided into several functional areas accessed by tabs.
  - Dashboard
  - Kapplets
  - Schedules
  - Repository
  - Data
  - Logs
  - Admin



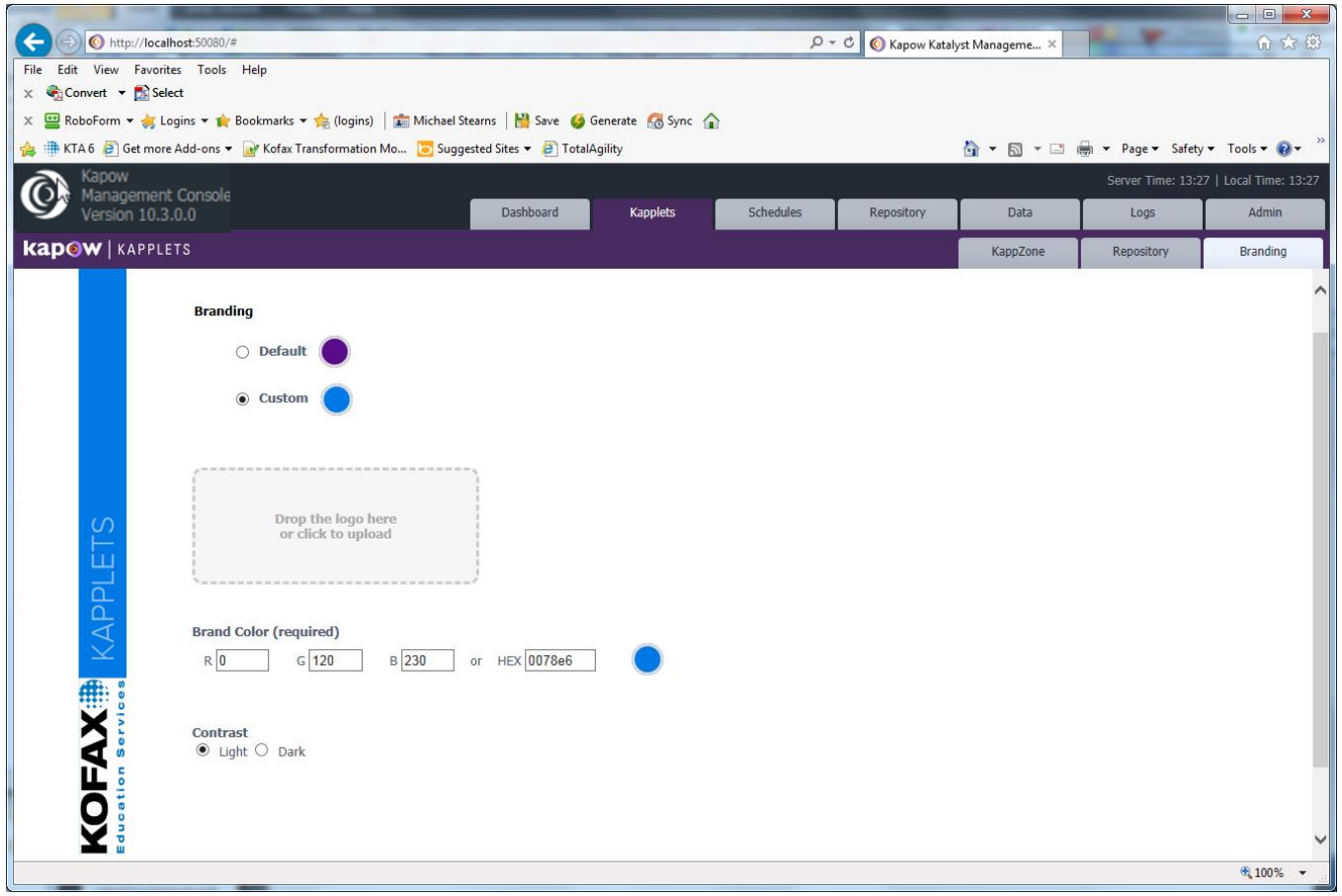
Dashboards

- Dashboards give you a quick overview of the Management Console. The information is presented through portlets, which shows the health of your RoboServers, Schedules, and Robots.



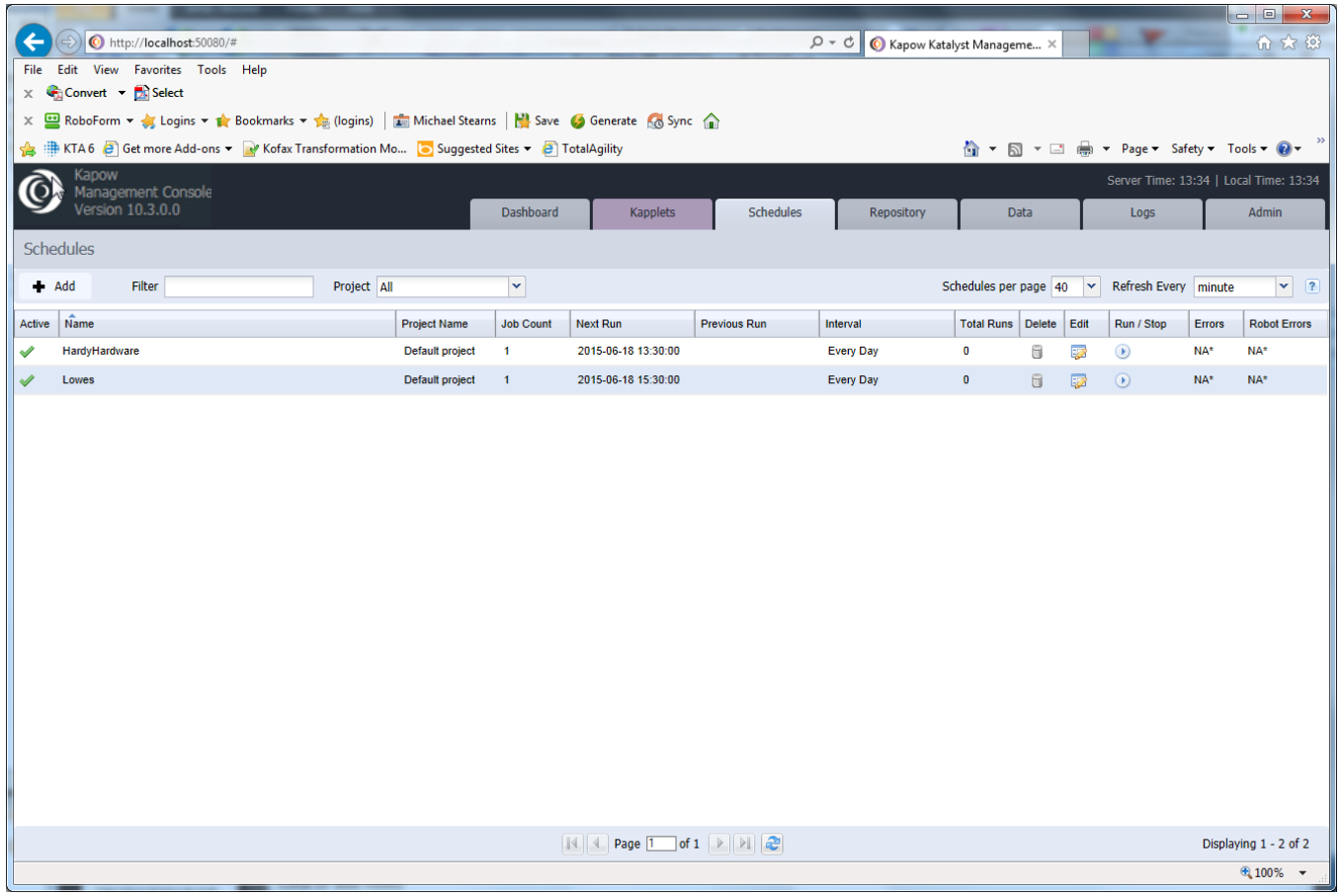
Kapplets

- A Kapplet is a web-based user interface that allows you to run a robot without any programming.



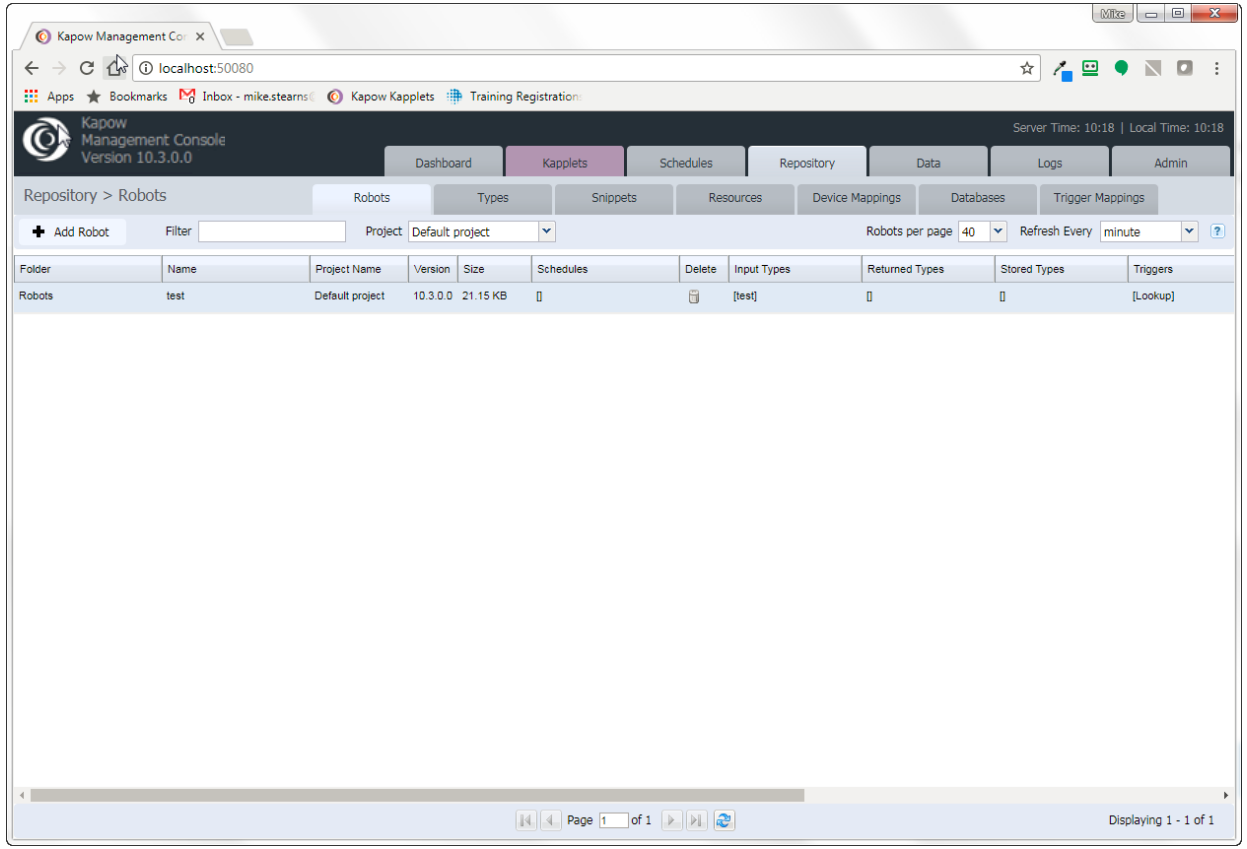
Schedules

- A schedule is a plan for running one or more robots, typically at pre-planned points in time and in a repeating fashion. A schedule does not run robots itself; it merely provides the plan for when the robots should be run, which is done by passing them to the configured servers.



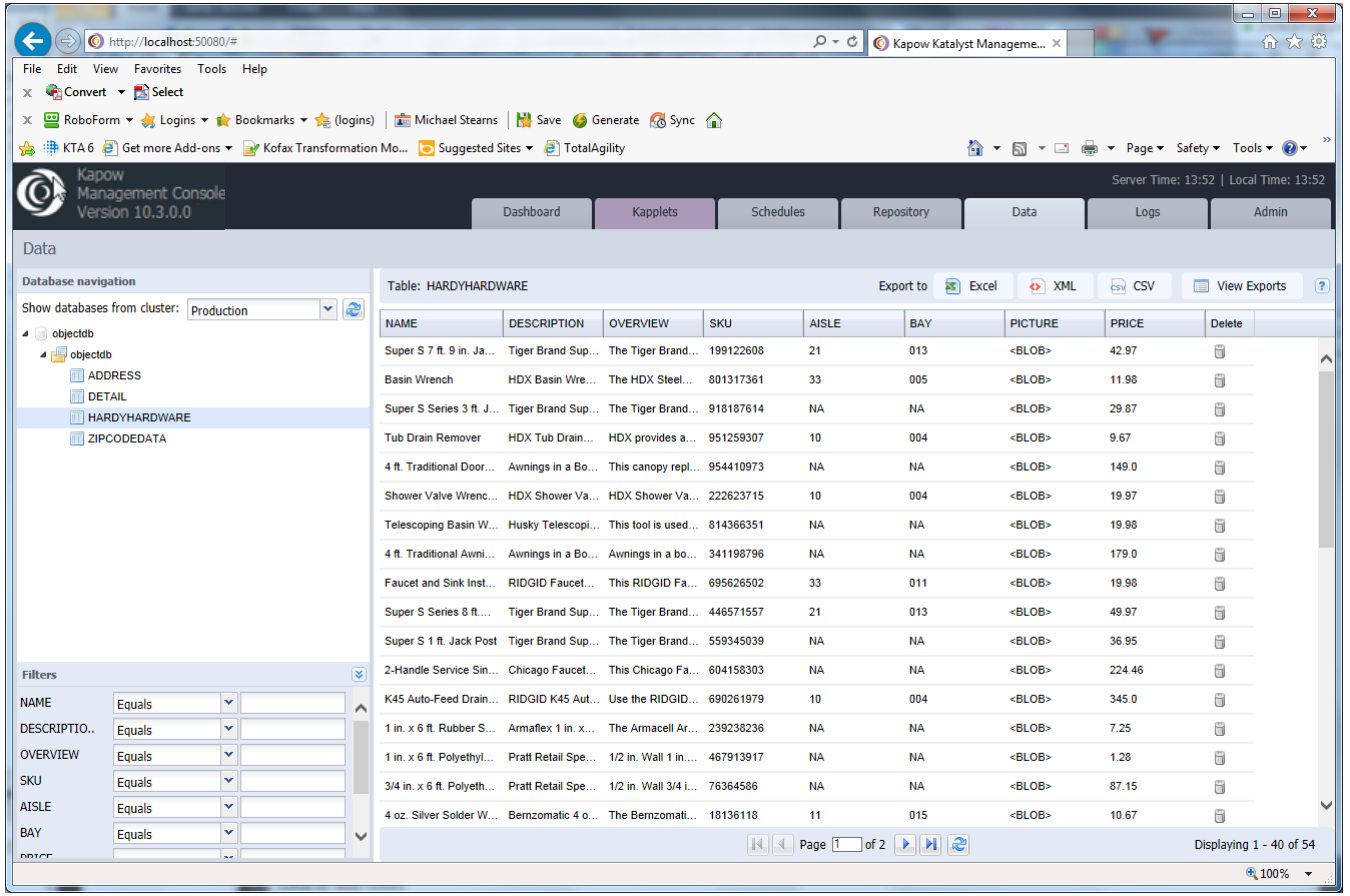
Repository

- Robots, type definitions and resources can be uploaded from Design Studio to the Management Console repository or can be uploaded manually through the web interface of the Management Console. Uploaded robots can be executed as part of a Schedule or through client code that executes robots using the Kapow Java or C# APIs. You can also use the APIs to programmatically query or update the repository.



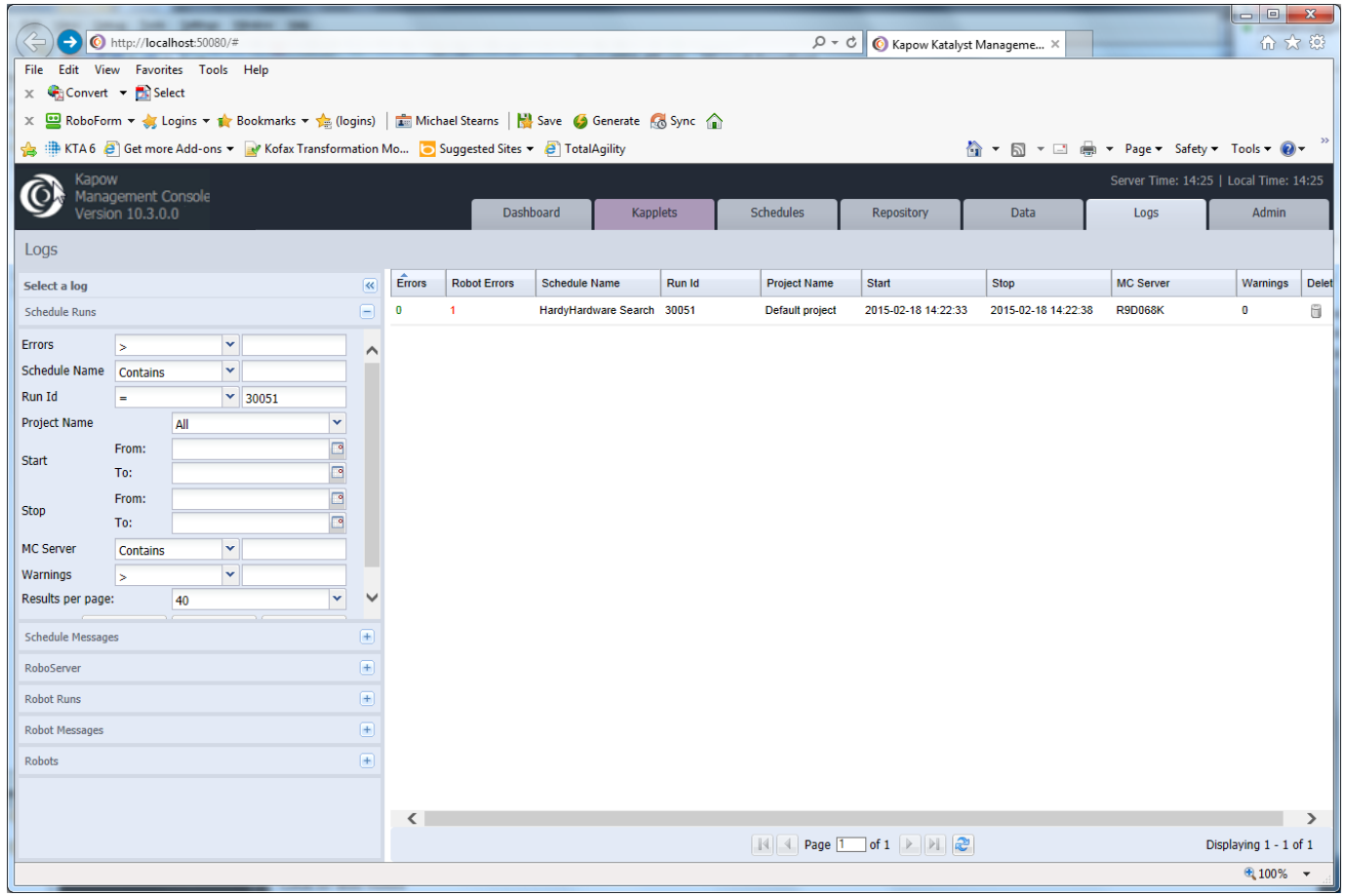
Data View

- The Data View allows you to see the data your robots have stored in databases. The data view also allows you to export this data to Excel or XML.



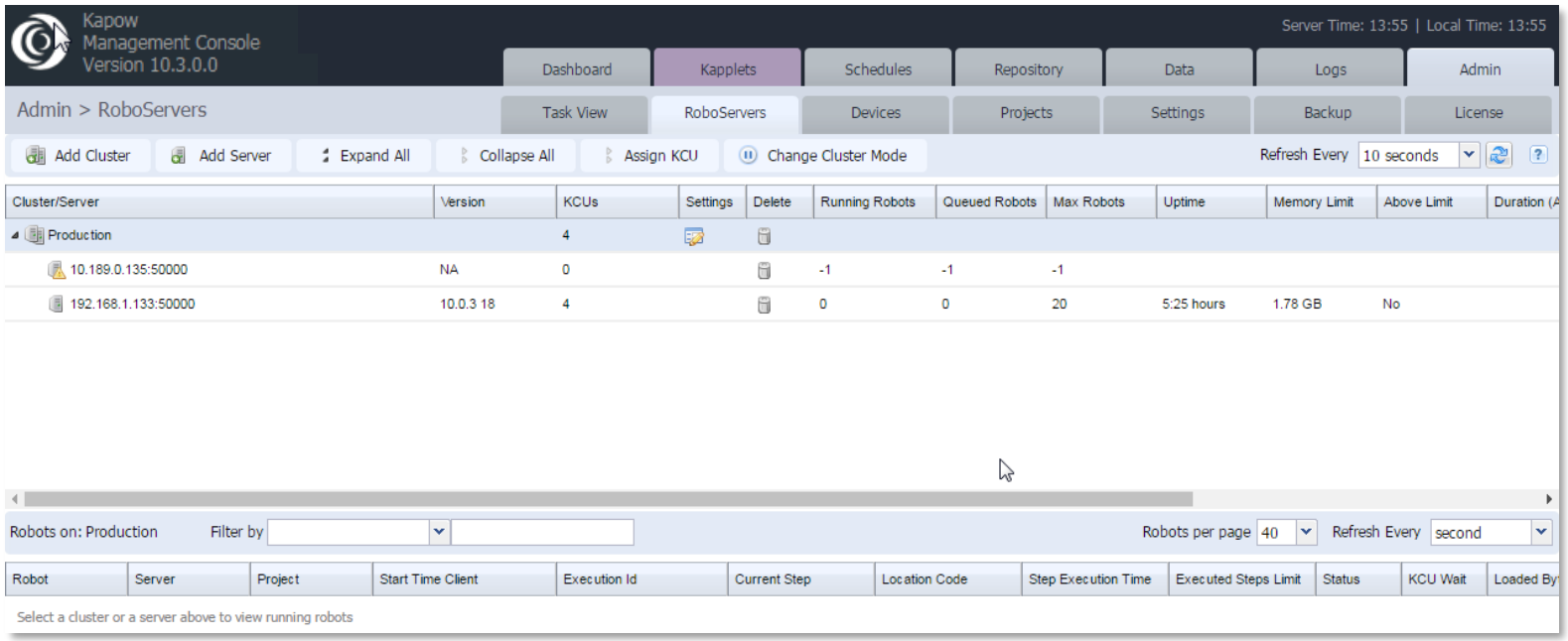
Logs

- Here you can view the execution history of your schedules. If database logging is enabled, you can also view the RoboServer logs which contains details of every robot execution.

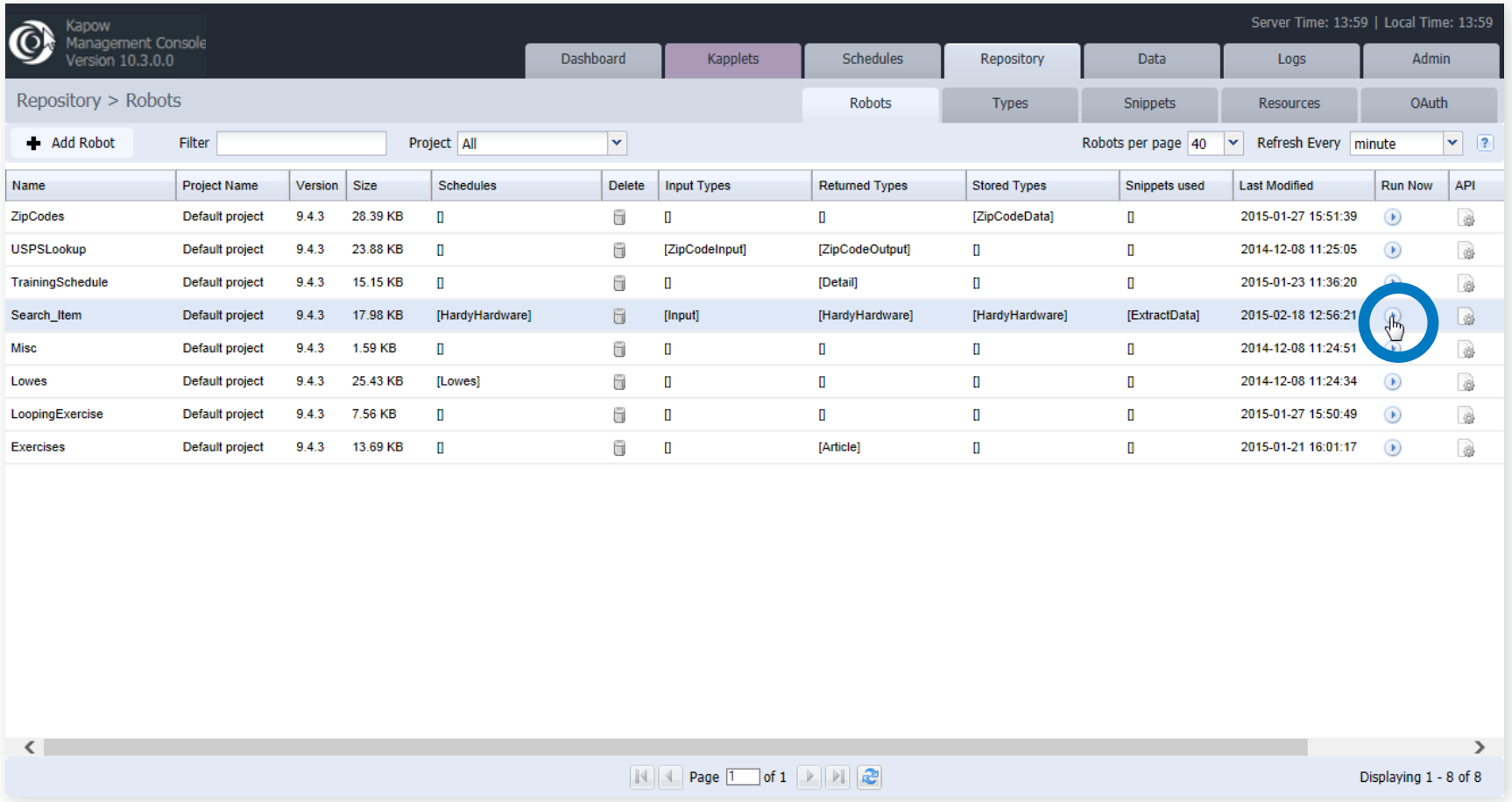


Admin

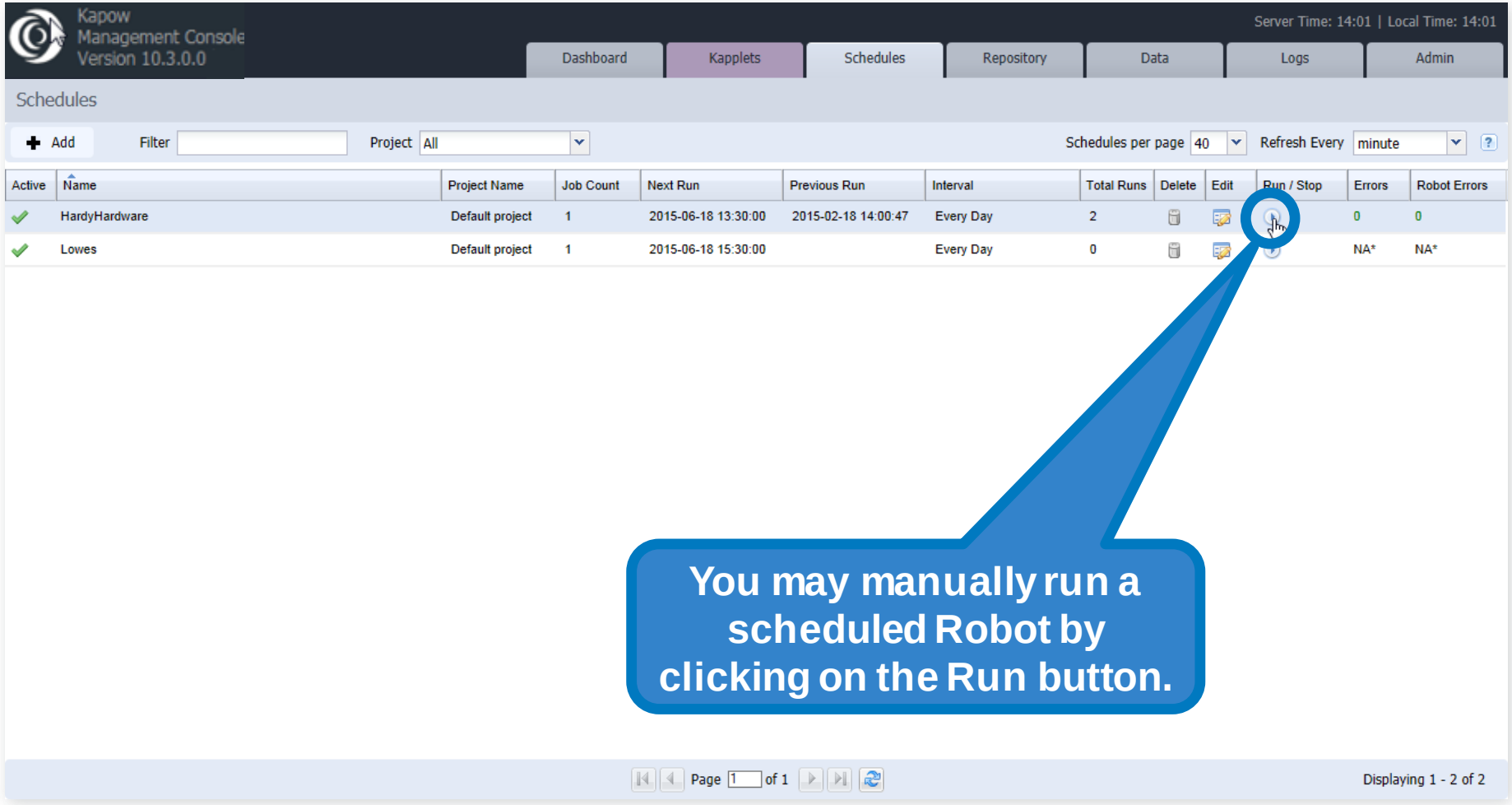
- In the Admin section you can configure settings for the Management Console. It also enables you to manage the clusters of RoboServers and their settings, as well as manage projects and permissions. This is also where you configure the license and create/restore backups.
- **Note:** Some features, such as High Availability may not be available, depending on your license key



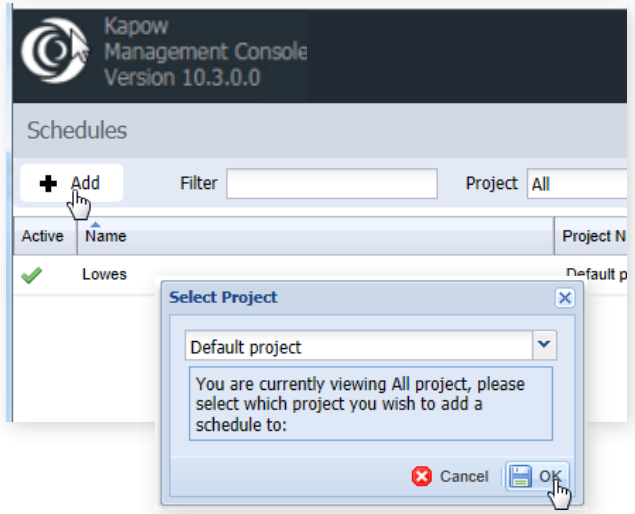
You May Manually Run a Robot from Repository/Robots Tab



And You May Schedule it to Run Automatically

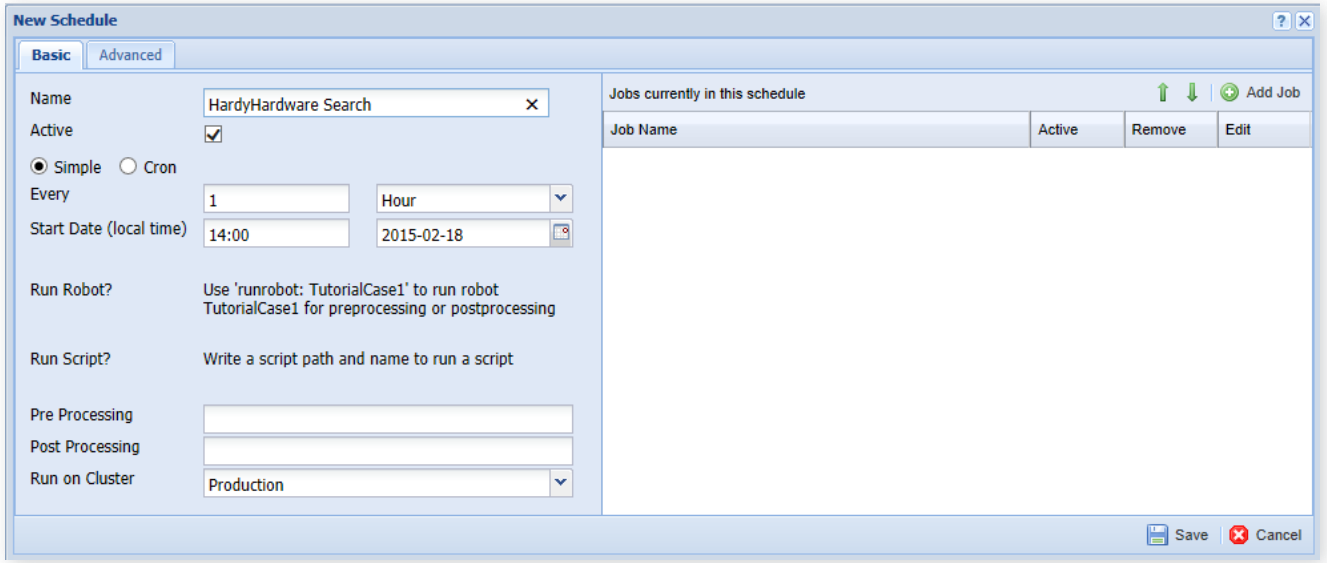


Scheduling Robots to Run Automatically

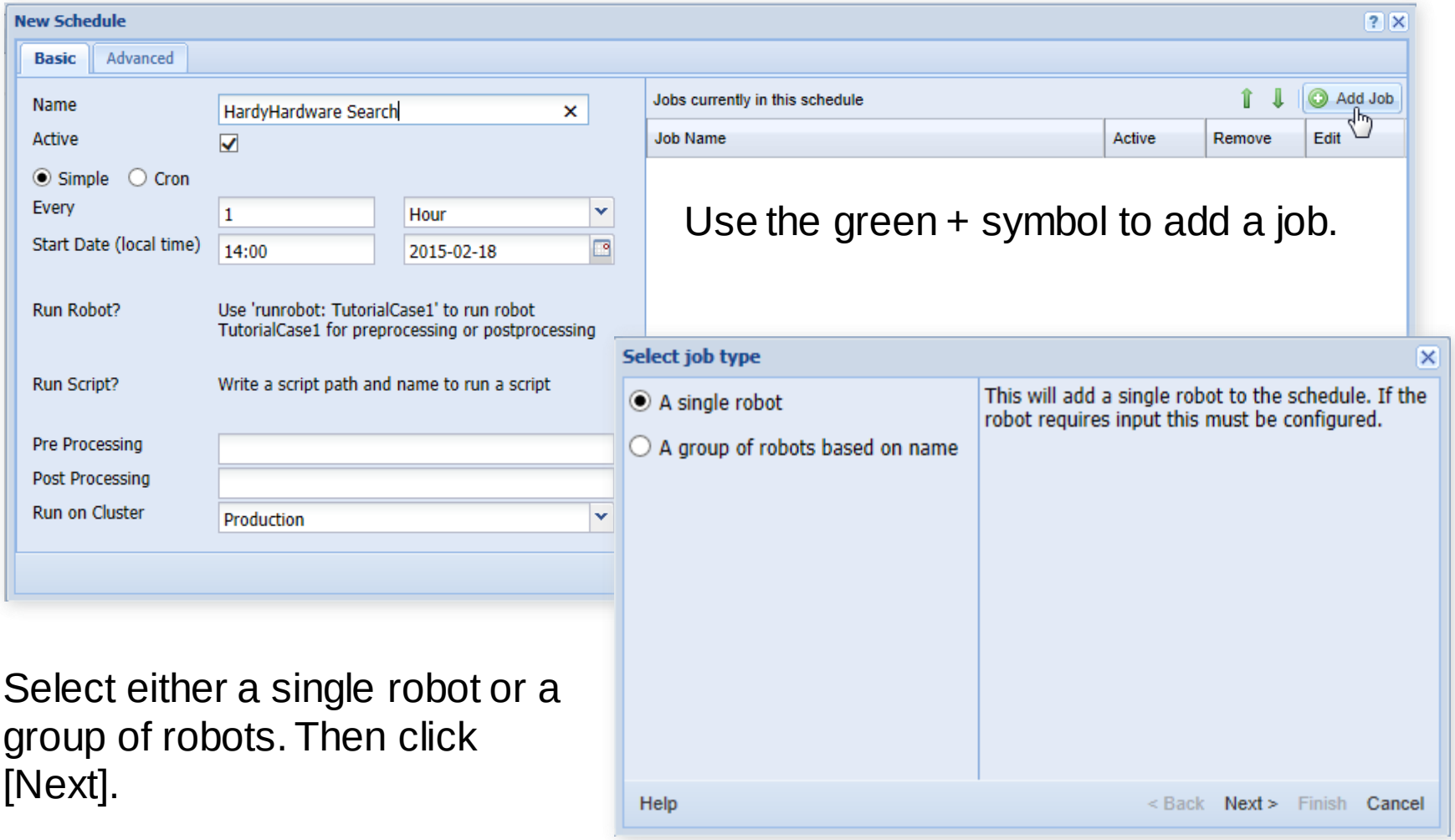


On the Schedules tab, click the Add button.

Give your schedule a name, check Active (or uncheck it if you don't want it to run according to the schedule) and set the time interval.

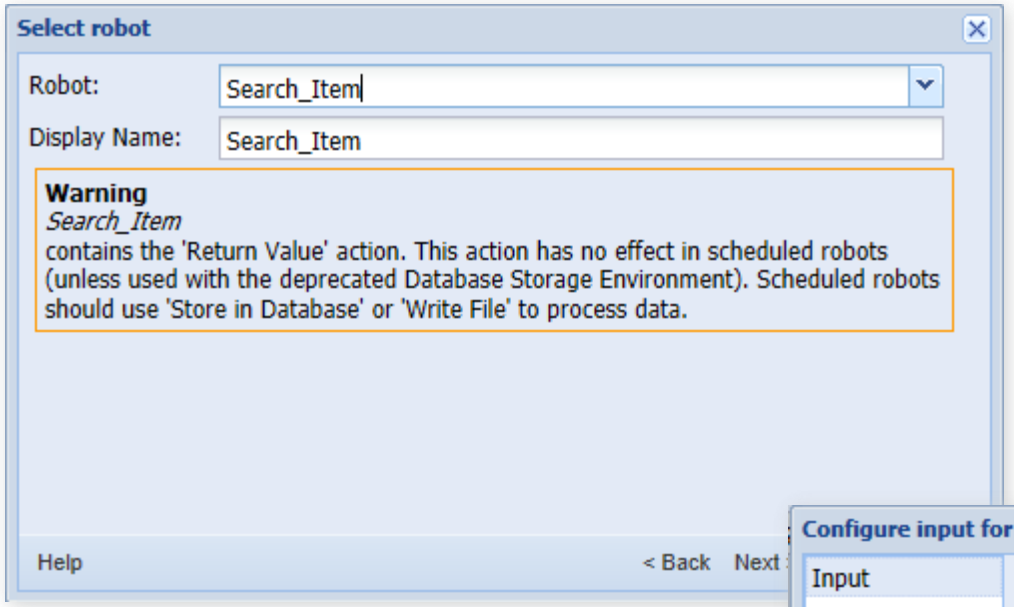


Add a Job

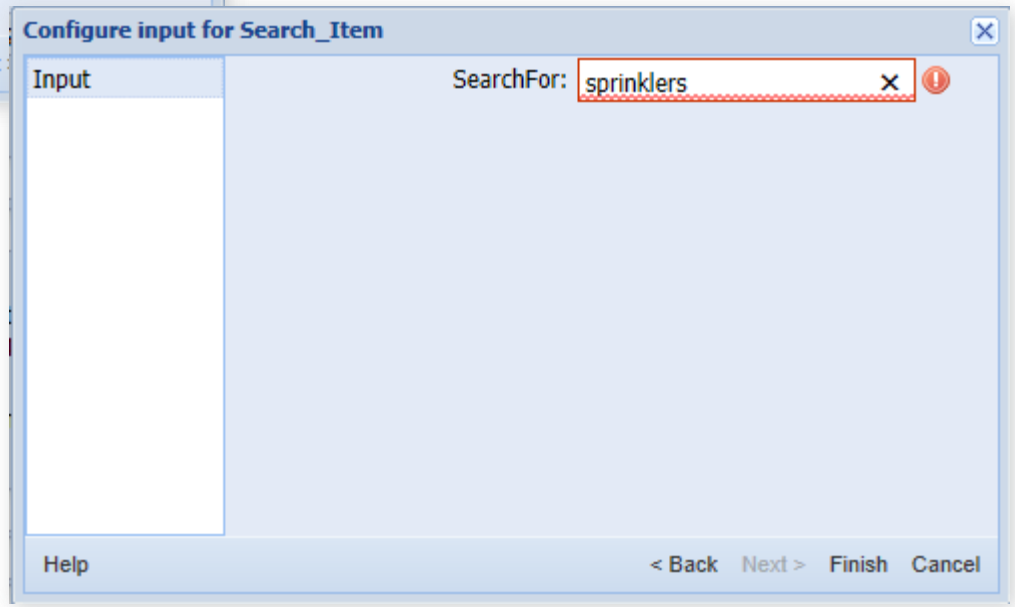


Select either a single robot or a group of robots. Then click [Next].

Select the Robot and Enter a Value for the Variable

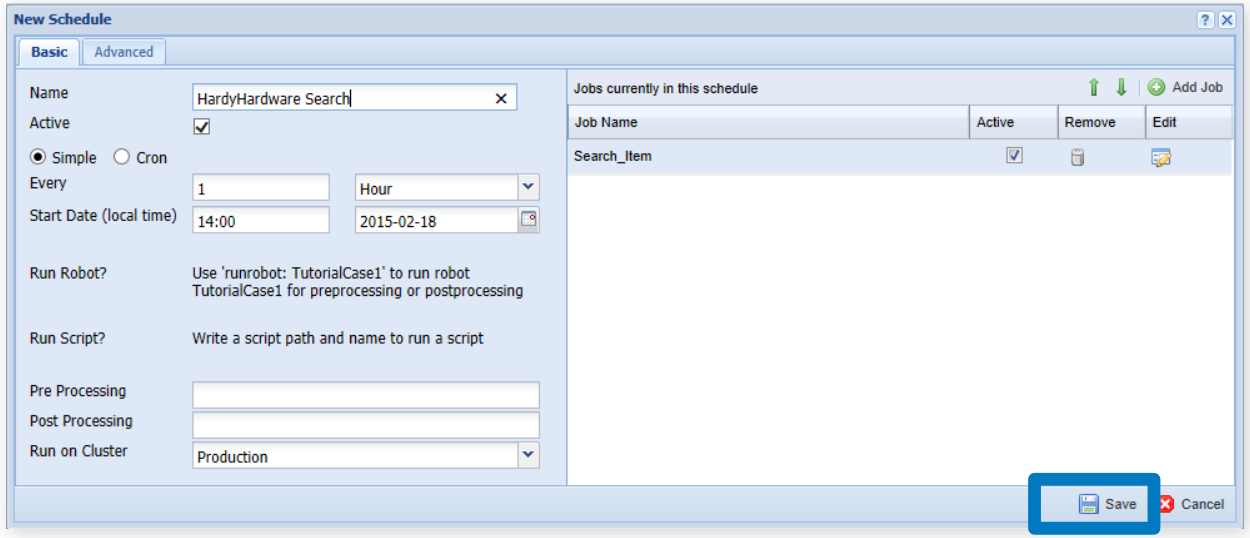


Select the robot from the dropdown list. If you want a different display name in the Management Console, you can enter it on the second line. Click [Next].



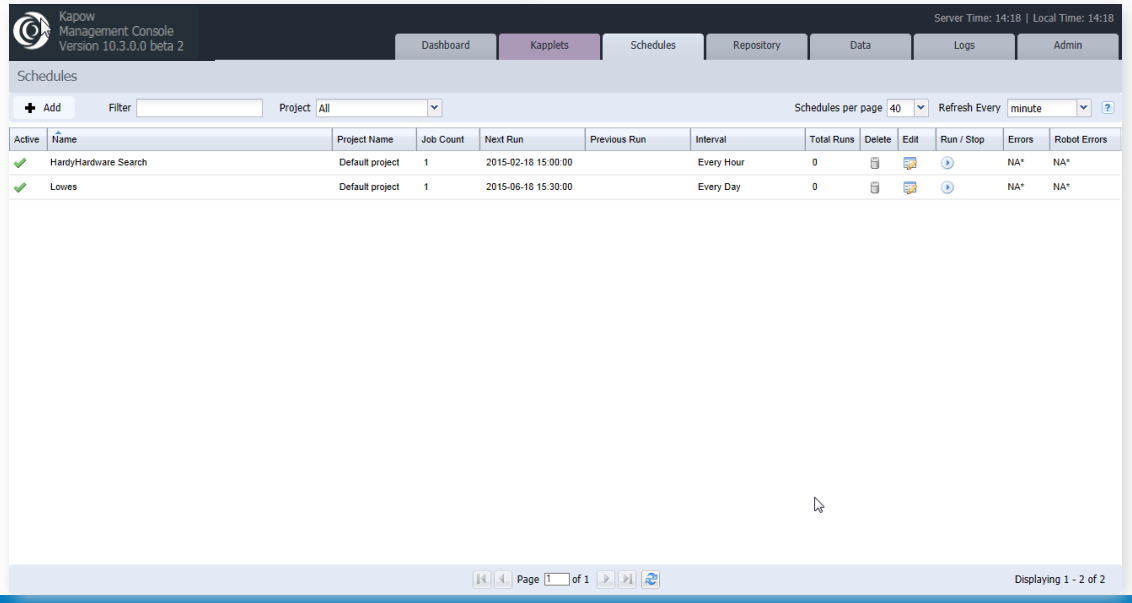
Because our robot has an input variable, you must enter a value here for it to run. NOTE: You can always come back and edit this job and provide a different value for the variable. Click [Finish].

The Job Has been Created and Added to the Schedule



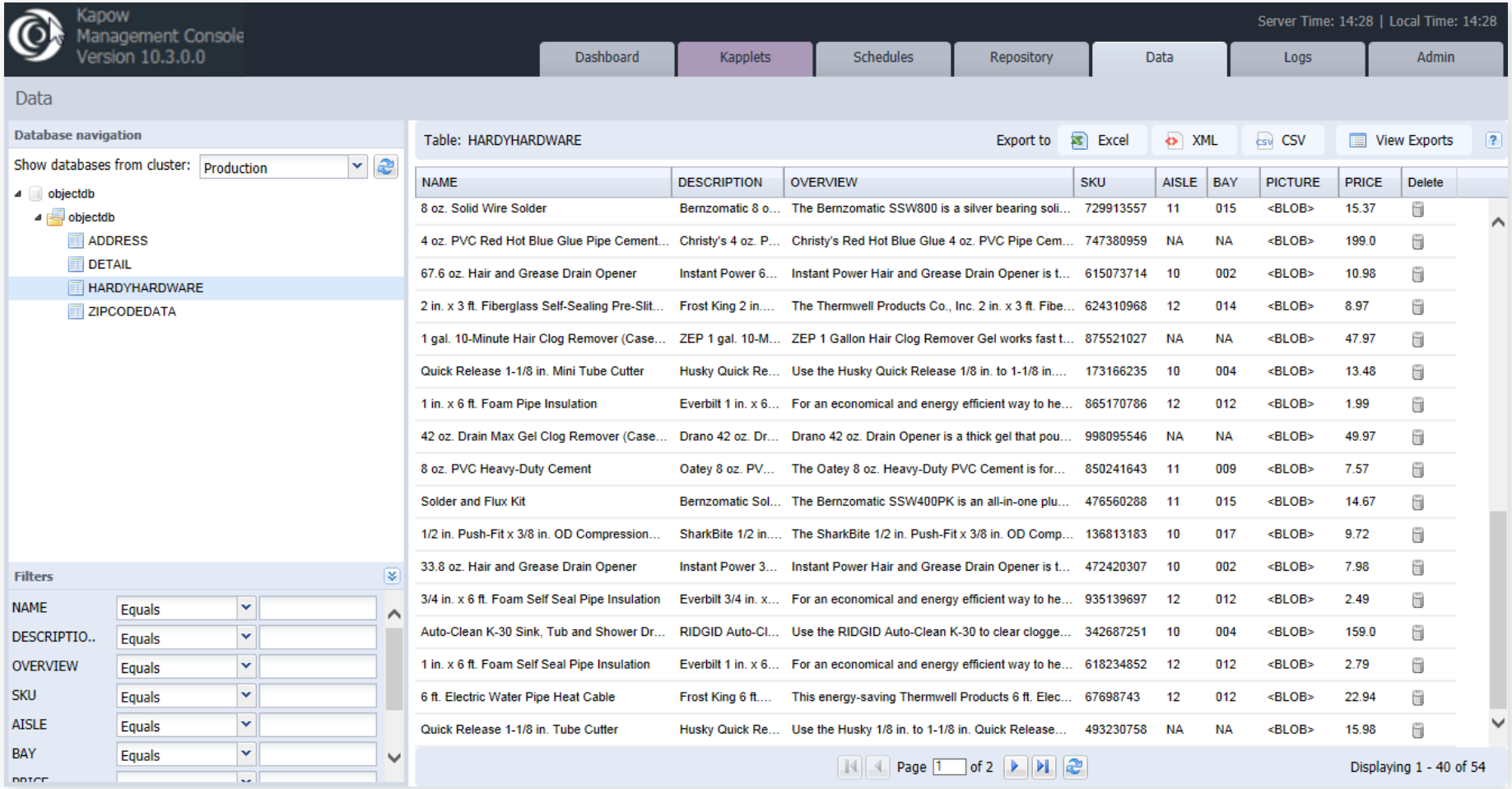
Click [Save] to save the job to your schedule list

New scheduled item has been created.

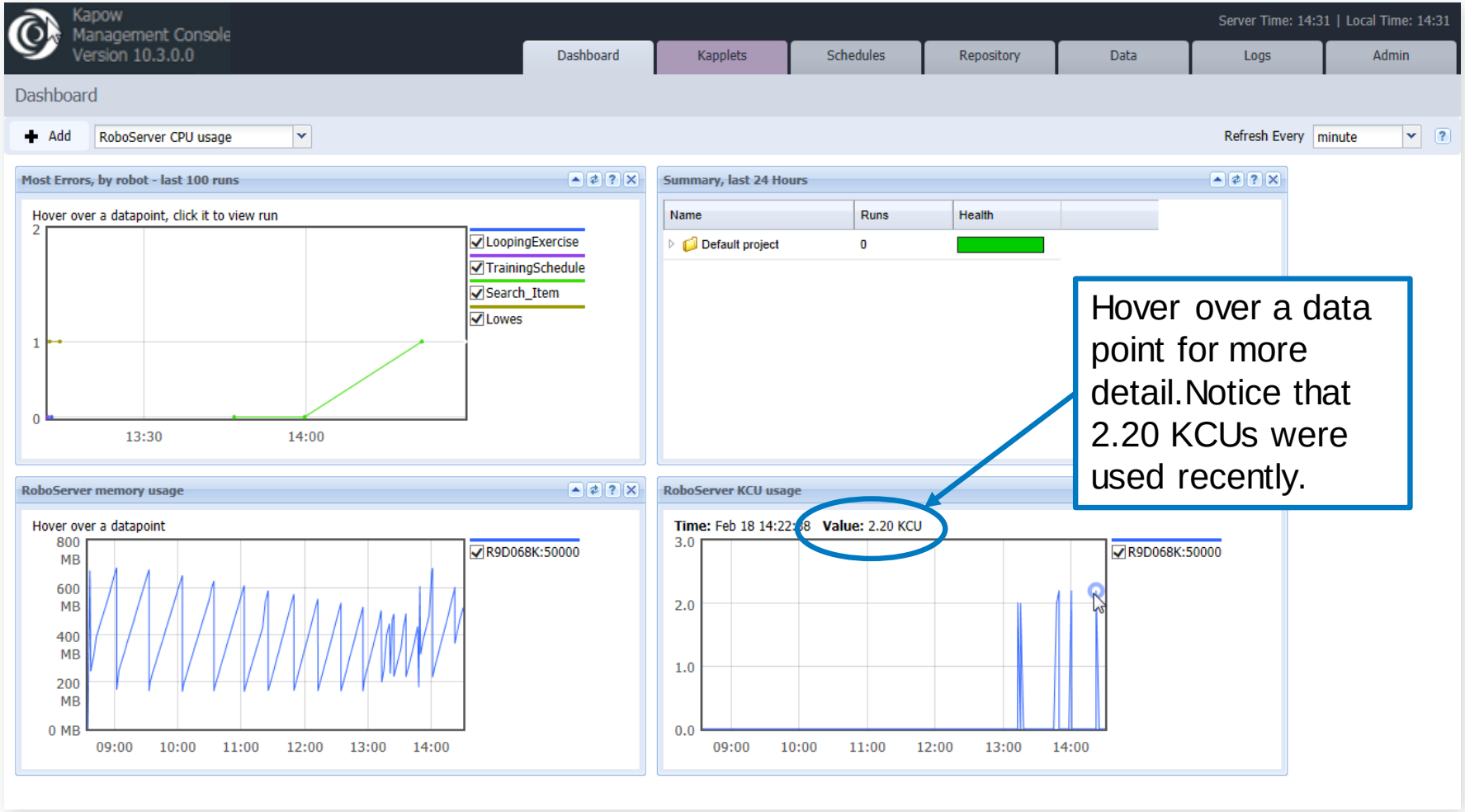


Viewing Data in Management Console

This is data that is now stored in the database. Notice the handy search filters at the bottom left of the screen.



Performance Graphs



Kapow Compute Units (KCUs) – A Review

- A KCU is a Kapow Compute Unit and is defined as a unit of measure for how many operations (or steps) a Kapow RoboServer can perform in one second (and is unrelated to underlying server capacity). Capacity-based pricing is how you license Kofax Kapow.
- A step is the smallest unit of action which can be performed within a RoboServer. Examples of steps are, loading a web page, writing a data record to a database, or performing a transformation on a data element.
- One KCU represents a total of 5000 KCU points per second. The number of Kapow steps that make up one (1) KCU depends on the type of Kapow steps involved, as each step type consumes a different amount of the KCU. The steps are divided into groups the most important groups are listed here:
  - Steps that both do I/O and execute JavaScript, 10000 KCU points E.g.: 2 page loads per second with 4 KCUs
  - Steps that do either I/O or execute JavaScript (but not both), 1000 KCU points E.g.: 20 Call REST Web Service Steps per second with 4 KCUs
  - Extraction and transformation steps 1 KCU point E.g. 40,000 extract or assign steps per second with 4 KCUs.
- In this version of Kapow, the KCU licensing model has been superseded by the CRE model based on maximum number of concurrently running robots.

KOFAX

Demo & Lab

Development Database Management Console