

TRM RulesManager™ SE

Whitepaper

Al Johnson, VP of Product Development
Total Resource Management

Table of Contents

- 03 Base Maximo® Platform as a Development Environment
- 03 Maximo is Complex Software
- 03 Maximo Out-of-the Box is not Complete
- 03 Maximo Requires Customization
- 04 The RulesManager SE™ Solution
- 06 RulesManager has a Stand-Alone Capability
- 06 About Total Resource Management

To facilitate the discussion of TRM RulesManager SE™'s value proposition, it is useful to agree on a few postulates about Maximo® Software

Base Maximo® Platform as a Development Environment

You know Maximo as the leading software application for enterprise asset management. TRM knows Maximo as that, plus Maximo provides a unique software development platform. If you strip away all of the outer applications of Maximo, such as WORKORDERS and POs, it becomes clear that the Maximo MBO/Presentation model is a good base platform for building any data-related application and presenting it as a Web application to an end audience. This is not restricted to the applications Maximo presents in its suites. The base platform allows for the addition of new applications invented out of whole cloth, which can be related (or not) to their Maximo cousins and fully integrated as though they were native to the product set.

Maximo is Complex Software

As elegant as the base Maximo platform is from a technical standpoint, it is also an intensely complex piece of work. The layered business applications in WORKORDER, PO, PR, INVOICING, etc. are in and of themselves wonders of intricate business logic. They would require a great deal of related process understanding to even begin to customize their behavior. The base programming platform for customizing or extending Maximo's general capabilities is also extremely complex.

Maximo Out-of-the-box is Not Complete

Maximo is not "complete." (This is not a negative statement, by the way). The industry expertise that IBM brings to bear in building the Asset Management, Financial and IT applications using the base platform continues to evolve and meet customers' needs better with each iteration. But as long as the business needs of that customer base continues to change and grow, the product will require tweaking and modification to meet these needs. It is virtually

impossible to deliver this intricate suite of complex related business solutions in such a way that it will meet all of the customer's requirements, all of the time, out of the box.

Maximo Requires Customization

No matter how much functionality is coded into the base Maximo platform, customers will always need to modify that behavior to suit their individual needs. Sometimes the only modification needed is a simple set of requirements. Other times, the requested modifications become very substantial. Field additions, modifications, screen appearance, data validation and business rule addition/modification are chief among these frequent requirements.

RulesManager SE takes the complex nature of Maximo's many layers (see Complex Software above) and exposes all aspects of it in a unified development environment where the addition and modification of detailed business rules can be performed in a matter of minutes. Maximo's rich complexity of interrelated business entities and actions are abstracted and made available in a content-assisted, intuitive environment.

Without TRM RulesManager SE

Without RulesManager SE, this level of accessibility is primarily available through the direct extension of multiple Maximo class files. While any level of customization can be performed on Maximo at this level, the cost of this approach quickly adds up. A few scenarios:

Engaging in-house resources to design and implement customizations requires a deep level of understanding of Maximo at the API level. This expertise is acquired over a period of months, and investment in in-house resources tends to result in dedicating the resources to the Maximo implementation project and its ongoing support.

Contracting outsourced programming is a solution that has a different cost. Changes and adaptations to original project specifications, which are a routine part of a major IT implementation of this type, require a (usually) time-consuming process of change request/

rewrites and partial redesigns along the way to achieve the desired end results. In addition, after the project's initial phase is completed and the outsourced firm has disengaged, the issues which have arisen (including problems, additional functionality requests and upgrades to the Maximo system) require either a long-term service commitment with the original firm or a complete reengagement of a new firm.

Extending Maximo meaningfully to solve real functional requirements usually involves the extension of many of Maximo's base classes, and frequently can involve scores of these java classes. Managing this complexity and change cycle involves significant costs over time and increased risk for the lifecycle management of the system.

One prime example of this can be demonstrated in the upgrade path of Maximo from version 5 to version 6. Maximo changed the name of the underlying asset object from "EQUIPMENT" to "ASSET." Any customization to any java class related to EQUIPMENT (a frequent occurrence) must be repackaged and rewritten in order to function in version 6.

Inevitably, a customer implementing a complex Maximo configuration is generally driven to a limited set of choices:

Don't customize Maximo; use out-of-the-box functionality. This requires the adaptation of your entire business process to fit Maximo's structure and flow.

Use only the portions of Maximo that adhere to your process and use other systems (legacy or newly acquired) as the remainder of the enterprise solution. This presents all the challenges associated with synchronizing and integrating these disparate systems. While some level of inter-system cooperation and integration is always expected in an enterprise implementation, it is generally a good practice to minimize the number of heterogeneous interconnects in favor of more normalized data structures.

Buy third party product(s) to fill the need. The market for third party products for customization of Maximo is

fragmented. By assembling four or five products from various vendors, you only approach a small subset of functionality available in the core RulesManager SE product. Additionally, you will be required to learn the individual configuration, management, deployment and maintenance of each of those products.

The RulesManager SE™ Solution

TRM RulesManager SE is written to allow a high level of configuration capability to all aspects of Maximo in a highly accelerated and intuitive way while substantially reducing the costs, both short- and long-term, associated with the configuration and maintenance of the system.

RulesManager SE is written to allow a high level of configuration to all aspects of Maximo in a highly accelerated and intuitive way, while substantially reducing the costs associated with the configuration and maintenance of the system.

A partial list of features of the RulesManager product which specifically and directly impact the cost of deploying Maximo follow:

A fully compliant Eclipse-based Integrated Development Environment (IDE) for rule building. This tool abstracts all aspects, events and objects in Maximo. It also allows navigation of all applications, events and fields, putting powerful rule-writing at the administrator's fingertips. The administrator simply opens the IDE, connects through a login dialog to the Maximo system and begins writing rules.

RulesManager SE is implemented using an ECMA compliant JavaScript engine. Rules are available for all Maximo events, including field changes, MBO saves, status changes (any and all) and MEA data arrival/departure. Double-click the entity/event and a JavaScript editor opens for writing business rules.

Maximo is presented to the rule writer as a highly abstracted set of JavaScript objects.

Adding a WORKORDER OnSave Javascript rule will open an editor and present the rule builder with an object named WORKORDER, which represents the record that the rule is acting on. The editor provides the user with a self-documenting tutorial view of all the underlying API's rich capabilities.

These are inaccessible by any other means in any other product without extending multiple java classes directly. Actions such as adding related records (i.e. adding WPMATERIAL lines to a WORKORDER) can now be programmed directly through the JavaScript environment without having to extend even one class using Java.

In addition to abstracting the Maximo capabilities of these objects, *RulesManager adds a number of highly useful convenience methods to these objects, such as "xml" and "csv."* These two methods in particular provide instant access to highly useful data representations for a wide variety of use.

Rules can be set to fire for any combination of users, groups, sites, orgs or applications.

Rules can be set to fire for any combination of users, groups, sites, orgs or applications. This means that a rule can be granularized down to a single user, group of users, collection of sites, or any imaginable combination.

Whenever the rule writer desires, they can simply click the "publish" button in the editor. The rules files are sent back to the Maximo server and the new rules are instantly in place. *No restart of the web server, no redeployment or modification to the EAR files and no system reboots are required.* A user can keep a Maximo page in a browser, modify a rule in the editor, publish and simply refresh in the browser to test the rule. In this way, rules can be tweaked and modified over and over to meet the desired result. The time savings in this step alone are huge.

A JavaScript rule can be attached to any Maximo WORKFLOW event, so a user can add any imaginable capability at the point at which Maximo workflow events occur.

RulesManager SE requires no Java programming.

RulesManager SE does not modify any of Maximo's meta-data or presentation configurations to achieve what it does. Previous customization performed by other third-party products or by direct java extensions by the customer or their agents all continue to work seamlessly with the addition of RulesManager SE. No previous capability is ever lost.

RulesManager SE can be turned off 100% with a single switch and reboot of the server. If there is ever a support issue involving the product, it is a trivial matter to temporarily switch the product off completely and test the system without it. When the product is turned off, Maximo reverts to a 100% IBM/MRO version of itself.

Rules written today are guaranteed to work from today forward. Any rule written in any version of RulesManager (from Maximo version 5.1) will function on any version of Maximo going forward. Customers who have written rules as far back as Maximo 5.1 are today deploying the exact rules files in Maximo 6.2. This speaks to the long-term maintenance costs. Since the customer has no Java Classes to validate or support, they simply receive a RulesManager SE update (if necessary) for their patch of Maximo and reboot the server. All rules are guaranteed to work going forward, so the cost to the customer is drastically reduced and the risk involved is nearly eliminated.

Remote Connection Capabilities

RulesManager SE doesn't just abstract the Maximo Objects in the JavaScript space. A large number of free utility objects are delivered in the JavaScript space, which add much capability to the rules that can be written:

- JDBC – A JDBC Object is available in JavaScript, which allows the user to write rules that reach out to remote systems to query or update entities.
- FTP – A full ftp client for sending/receiving files
- XML – Native support for XML reading/writing. Handling of XML importing and exporting is built directly into the engine. All Maximo-structured objects have an XML property attached for direct access to an xml representation of the entity.
- Web Services – Any XML web service can be accessed live during any event in Maximo.
- HTTP – An abstracted http object can reach out from your rule to post data to a web server or retrieve URLs from remote web servers.
- File/Dir Objects – A complete set of objects for accessing directories and files
- Email – A robust email client is available in your JavaScript rules to send any imaginable type of email, including templated email, attachments, mailing lists, etc.
- Maximo – A Maximo object, which is capable of reaching out to any version of Maximo 5 or above, is available as well. Legacy Maximo instances or those belonging to contractors can be connected to and data can be read or exchanged seamlessly. RulesManager SE is capable of connecting simultaneously to any number of Maximo systems of different versions at the same time. Combined with JDBC, this makes a powerful data migration story.

All of the capabilities of the product can be exploited in any standalone java environment, such as Linux, SunOS or Windows, by simply executing the standalone shell on a saved “.js” file. These self-standing scripts are executed on a variety of platforms without the presence of the Maximo product on that platform.

About Total Resource Management

Total Resource Management (TRM) is focused on improving the asset and operational performance of organizations through the effective use of information technologies. TRM is an IBM Premier Business Partner with over fifteen years; experience delivering asset and service management solutions based upon IBM Maximo. TRM supports clients across a wide range of industries, including government, defense, cities, facilities, energy, utilities, transportation and life sciences. TRM is based in Alexandria, Virginia and has business centers across the U.S. For more information, visit www.trmnet.com or call 703-548-4285.

RulesManager has a Stand-Alone Capability

The entire discussion so far has focused on RulesManager SE in the context of a running Maximo business engine. There is one more very important aspect of RulesManager SE which bears prominent mention: RulesManager SE will allow stand-alone scripts to be executed by a “shell,” which comes with the product.