

InRule® and Microsoft Dynamics® CRM

Nine Considerations for Managing Business Rules

Overview

Knowing when to use the out-of-the-box business rule functionality in Microsoft Dynamics® CRM versus the InRule® Business Rule Management System may feel like a daunting task if you are unaware of the features and capabilities at your disposal or which of those features and capabilities solves your particular business problem. This document sheds light on nine key characteristics of business rule management that may factor into your decision.

1. RULE COMPLEXITY

The Portable Business Logic (PBL) feature in Microsoft Dynamics® CRM provides the proper level of functionality for creating business rules against a single entity. You can create rules to set values on fields, perform calculations and validate data, as well as more user interface-specific logic like the ability to enable/disable fields, show/hide fields and set field requirement levels.

However, when the complexity of the business rules increases, there is a stronger argument for creating and maintaining that logic in InRule’s irAuthor® application. Unlike PBL, irAuthor allows you to build rules with access to data points across multiple entities. Since the entity data models in Microsoft Dynamics® CRM support one-to-one or one-to-many relationships, your rules may also require mixing and matching evaluations, validations, and calculations amongst those same related entities. Additionally, conditional statements may require complex grouping expressions (mixing of “and” and “or”) within a single rule, which is supported in irAuthor’s Business Language and Decision Table formats.

	Supports Rule Complexity	Multi-Entity Assessments	Complex Expression Grouping	Embed If...Then Statements
Microsoft Dynamics® CRM PBL				
InRule	•	•	•	•

2. IMPLEMENTATION

Authoring business rules, regardless of whether they are simple or complex, should be intuitive. Creating complex business rules for Microsoft Dynamics® CRM requires access to the entities, fields and relationships, as well as an interface that allows the rule author to quickly and easily create, change and deploy rule changes with minimal effort. The management of Portable Business Logic in Microsoft Dynamics® CRM consists of HTML-based editors used to construct the condition and action statements that comprise a business rule.

InRule's irAuthor application provides multiple options for building business logic depending on the use case, level of complexity and functionality:

Type	Description
Business Language	An intuitive dropdown and context-driven rule authoring control for building various rule constructs (If/Then, Select Case, Do While) and/or execution of actions in a natural language syntax using hundreds of built-in templates.
Decision Table	The table format is driven by one or more conditions resulting in the potential execution of one or more actions. A single decision table can house thousands of use cases in a single interface for better visibility and easier maintenance.
Calculated Fields	Field values can be dynamically generated using any combination of more than 225 built-in functions.
Vocabulary	Customize Business Language rules by creating your own templates using familiar business domain technologies.

3. TESTING

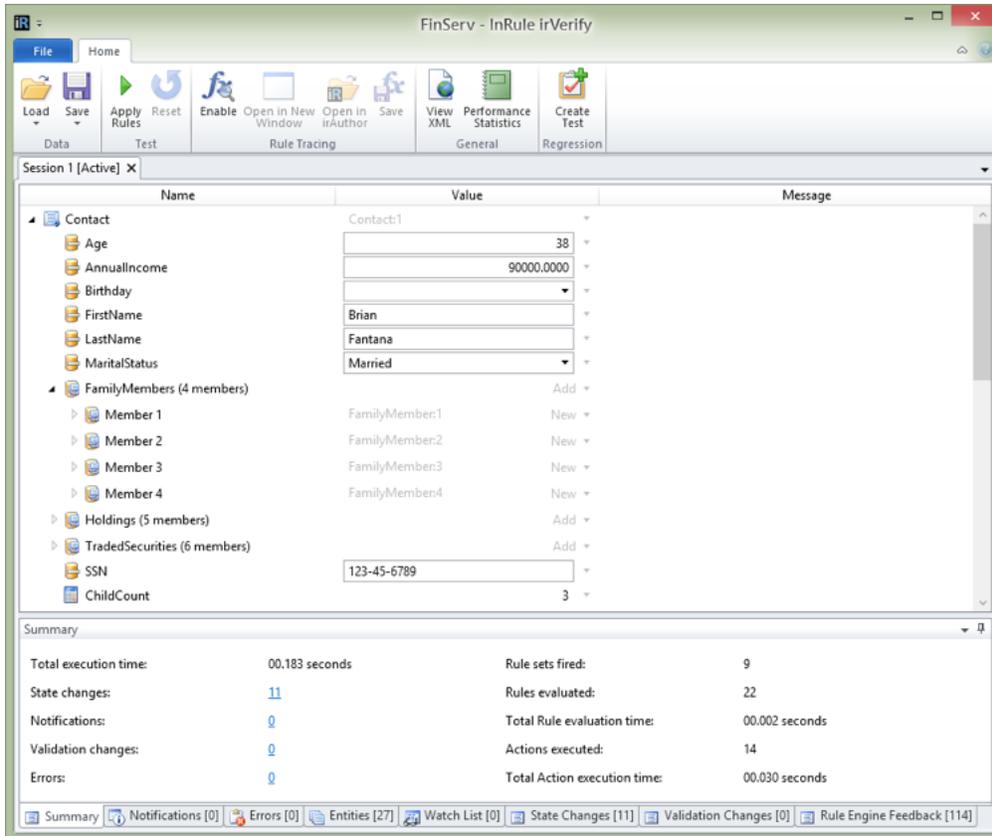
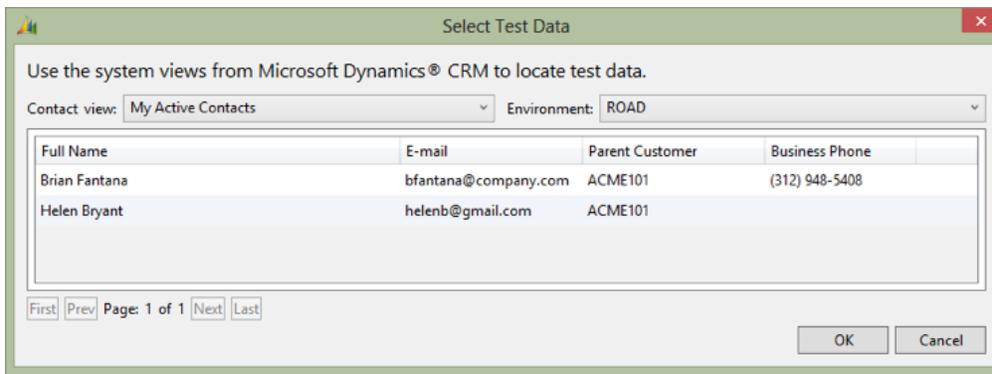
When a business rule changes or new rules are added, they must be tested and verified before they are deployed.

Testing PBL directly in Microsoft Dynamics® CRM is not possible without having a completely separate CRM test environment. Even then, changes to the rule must be made, published to the system and then the user needs to re-create the scenario in CRM to initiate the call to the business rules and then verify the results. Any errors or faulty logic encountered in the testing process would require repeating the same steps until you achieve the desired results.

InRule's irAuthor has direct integration into your configured CRM environments. By leveraging the CRM Organization Service, irAuthor can access all Views associated with the entities that have been imported into irAuthor.

When the rule author selects the appropriate entity instance, a copy of the data is then loaded into InRule's testing application, irVerify®.

From irVerify, the author can apply the rules to the data, verify outputs and results and view all logging information captured during the rule execution process, including execution times, state changes and end-to-end rule tracing.



4. DEPLOYMENT

Deployment of Portable Business Logic changes consists of publishing those changes back into Microsoft Dynamics® CRM. While this may be seen as a benefit to some, others may perceive the inability to stage PBL changes for additional oversight or testing as a gap in the deployment process.

The rule applications created in irAuthor can be saved to InRule’s rule repository, irCatalog®. Like Team Foundation Server (TFS) or other source control repositories, irCatalog provides the ability to check-out and check-in rule changes. This results in an audit trail of who made changes, what was changed, and when the modifications were made. Multiple catalogs can be installed and configured (for example, Test, Dev, Prod) and with InRule’s Catalog Manager application, rule applications can be promoted or rolled back among those catalogs.

5. INTEGRATION

When a Microsoft Dynamics® CRM business process requires user interaction, considerations should be made for triggering rule execution off of built-in form events (tab, field change), plugin events (Create or Update) or some combination of the two.

Form events require configuration of a CRM entity form using the built-in form designer in coordination with PBL. Rule execution is done client-side.

Plugin events are typically used for more complex rule scenarios that cannot be accommodated via PBL. However, plugins are built on code and therefore require developer resources to create, implement, deploy, and maintain. The rules are also “hidden” in code and cannot be viewed by non-technical personnel. Rule execution is done server-side.

With InRule, integration can be accomplished in several ways. While these options may have a more technical aspect, the key differentiator is that all business rules are maintained in irAuthor and are not implemented in the code, so integration is typically a one-time, up-front resource allocation.

Type of Integration	Point of Execution	Description
Type of Integration	Point of Execution	Description
Plugin	Server	InRule provides the plugin which then must be deployed. Code changes, if any, are minimal, and typically done only for environment-specific configurations prior to the initial plugin deployment.
Web Service	Server	InRule provides the service which the client must then deploy and integrate into Microsoft Dynamics® CRM via custom JavaScript.
Plugin + Web Service	Server	A hybrid approach that entails the plugin calling the Web service. This option is commonly used for CRM Online.
JavaScript Engine	Client	Rules created in irAuthor can then be deployed into CRM as a separate Web resource directly from irAuthor. Additional configuration is required within CRM to hook up the JavaScript rule application to specific CRM events.

6. SCALABILITY

The execution of complex business rules can be processor-intensive. Architecting a solution that allows the business rules authoring/processing and Microsoft Dynamics® CRM to scale at different rates provides for more manageable long term growth and maintenance. While Microsoft Dynamics® CRM supports scalable server topologies, with front-end and back-end servers working together but still performing specific tasks, InRule can execute those processor intensive business rules out-of-process from Microsoft Dynamics® CRM. When CRM usage increases, resulting in an increase of business rule executions, the rule execution service can be scaled out independently to accommodate the increase in activity.

7. REUSABILITY

Some scenarios may require reusing the same business rules across multiple integration points, whether it is a user interaction event, back-end process, or a combination of the two, in order to maintain consistent business rule usage and eliminate unnecessary duplication efforts. While Microsoft Dynamics® CRM can support these patterns using custom plugins or workflow activities, those options require custom code and developer resources to implement and maintain.

With InRule, the rule applications created in irAuthor are simply containers of reusable business logic. Whether integration is done via plugin or Web service, either could make a call to the same set of rules so there is no need to re-create the business rules or duplicate them for every integration point. Those same rules could even be used in other applications outside of Microsoft Dynamics® CRM.

8. VISIBILITY

The ability to view and understand the business rules is essential for maintenance and transparency.

Since Portable Business Logic is configured and stored in Microsoft Dynamics® CRM, only licensed and permitted users of Microsoft Dynamics® CRM will have access to the PBL. Microsoft Dynamics® CRM does not provide a reporting mechanism on the PBL.

For InRule, irAuthor is the first step to transparency. As noted in the Implementation section, irAuthor provides multiple rule authoring formats and customization options for creating rules in a natural language format. Anyone, regardless of their level of technical expertise, can review the business rules created in irAuthor. A detailed rule application report can also be generated from within irAuthor. This report is in HTML so it can then easily be distributed outside the rule authoring group. As a third option, InRule provides an add-in for Microsoft Word which provides the ability to generate documents by inserting rule application artifacts and definitions.

9. LONG RUNNING PROCESSES

With Microsoft Dynamics® CRM, some business processes have longer lifespans that require the initiation and delegation of a task where the corresponding response or action does not take place until hours, or sometimes days, later. Microsoft Dynamics® CRM comes with a powerful and scalable long-running workflow engine built into the platform. While InRule does not inherently support such processes, it is advisable to use InRule within such a CRM workflow where complex decision points need to be evaluated.