

WHITE PAPER

A Guide to Harvesting Business Rules:
From Discovery to Documentation to Leveraging Decision Logic

By William Leannah,
IT Consultant and Adjunct Instructor at Marquette University

Published by InRule Technology®, Inc.

The process of harvesting business rules can provide significant benefits, from expressing and organizing business rules to identifying and reducing inaccurate and inconsistent decision logic implementation, to modernizing and revitalizing legacy applications and jump-starting new applications.

**Before the reward
there must be
labor. You plant
before you harvest.**

*Ralph Ransom,
Author*



Dissemination or reproduction of this publication or any part of it for any purpose or in any form whatever is not permitted without the prior express written consent of InRule Technology, Inc. Information contained in this publication may be subject to revision without advance notice.

© 2013 InRule Technology, Inc. All rights reserved. InRule and InRule Technology are registered trademarks of InRule Technology, Inc. All other trademarks referenced herein belong to their respective companies.

Some of the product and company names used in this white paper are trademarks and/or registered trademarks. They are used explicitly for reference purposes and are, independent of marking, property of their respective owners.

Contents

Abstract	4
Initial Considerations	5
Using Context to Organize Rules	5
Deciding on the Problem(s) to Solve	6
Developing a Planned Approach to Capturing Business Rules	8
Reviewing Existing and Available Documentation.....	9
Examining Legacy System Code or Observing Current Behavior.....	9
Data Mining	10
Conducting SME Interviews or Workshops	11
Things to Watch Out For.....	13
Defining Business Rules and Other Common Implementation Patterns.....	14
Establishing a Common Vocabulary and Reducing Translations	15
Case Study Example: Max Tuxedo Rental.....	16
Modeling the Facts	17
Crafting the Business Rules	18
Conclusion and Summary	20
Acronyms.....	20
Glossary	21
References.....	22

Abstract

The purpose of this guide is to provide the reader with the guidance and direction they will need for identifying and documenting their business rules. It is the intent of this guide to describe the rule harvesting process as a practical hands-on approach that can be easily followed to maximize implementation success.

Rule harvesting is the process of identifying and documenting an organization’s business rules before attempting to implement them within InRule®, or any other system. Performing this analysis ahead of time will help an organization plan and implement business rules that provide the greatest amount of effectiveness and operational accuracy. This is a repeatable and scientific approach. Once mastered, it will become a meaningful technique that will add business capability and drive value.

The rule harvesting process is also known as the Business Rules Capture-Model-Craft Process which is depicted here in a “swim lane” diagram.

Performing rule harvesting analysis will help an organization plan and implement business rules for the greatest effectiveness and accuracy.

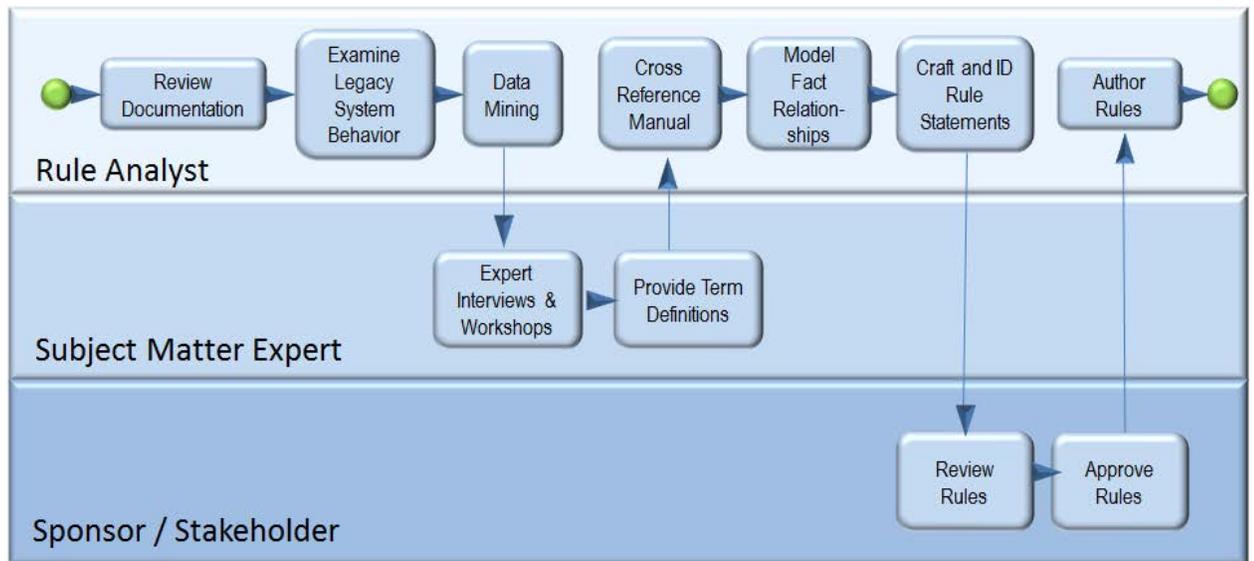


Figure 1: Capture-Model-Craft Process

Initial Considerations

Attempting to find all of an organizations business rules can be a daunting and overwhelming task. However, as Socrates famously stated: "Confusion is the beginning of wisdom." There are many things to consider, but first and foremost start by deciding to keep things as simple as possible and doing things one step at a time. The most important aspect of this work is to always be explicit, direct, and precise when defining business rules.

Business rules must be unambiguous and to the point. They should never be based on assumptions about how they might work. They must **NOT** be left open to **translation or interpretation**, regardless of an individual's experience level within an organization (Goodman, 2008). Those things will only lead to confusion and problems will eventually ensue within the project and in the final implementation.

Consider establishing a small rule capture team that involves more than one person. That way work streams can be segmented, allowing team members to share ideas and collaborate on solving problems. At some point this effort will involve Subject Matter Experts (SME's), likely from outside of the IT organization. The activities performed by team members are explained further in this paper; this will help determine the right people for the rule capture team.

Using Context to Organize Rules

Business Rules Engines (BRE), a component of a Business Rules Management Systems (BRMS), are context-driven machines. They cannot deal with general purpose problems or situations that have not been explicitly described (Giarratano, 2005). This means that a BRE needs a **prescribed context** in which to operate. In general, the context is provided by the rules being enforced for a given solution. The context is the target landscape for what a particular rule set "is about" or which problems they are attempting to solve. For example: a rule set that determines the discount group rate when renting tuxedos. It is a common business practice to provide special pricing when consumers buy in bulk. The business rules surrounding this practice forms the context of the special pricing model. It provides a purpose and a meaning for how the business rules are described and how they are executed at runtime.

The establishment of that context begins with deciding which problems are going to be solved with the rules-based implementation (Leannah, 2006). Realize that solving all problems simultaneously within a single implementation is often not feasible. The reason for this is context. A rule for problem "A" will be different than a rule for problem "B". For example, Mortgage loan eligibility rules are very different from rules that support a firm's internal payroll systems. They will reside in different buckets or solution spaces. Therefore, they should be kept separate from each other. It provides the logical groupings for the defined rule sets which helps to keep things maintainable as the individual situations evolve and mature over time. As things do evolve, individual rule sets can be updated and maintained without impacting each other.

Deciding on the Problem(s) to Solve

Start out by creating a candidate list of the areas within the implementation that could benefit from a rules-based approach or an automated decision (Taylor, 2007). This list is important; it will allow the team to focus their rule capture efforts on the specific rule sets that will participate within those decisions and avoid those that don't. In a Service Oriented Architecture (SOA) environment, implemented candidate solutions are typically built as decision services. These services represent the operational decisions that drive the business (Erl, 2008).

Depending on the industry and specific business problem being solved the list could look something like the following:

Candidate Item	Description
Determine Loan Eligibility	A decision service that determines if a loan applicant is eligible for bank product "X".
Determine Territory Assignment	A decision service that determines the territory assignment for a sales agent who participates in distribution channel "X".
Determine Shipping Fee	A decision service that calculates the shipping fees associated with a customer order of product "X" shipped to destination "Y".
Determine Patient Medications	A decision service that determines which medications can be taken simultaneously without harming patient "X".
Determine Investment Risk	A decision service that determines the financial risk of investing in a stock "X" or mutual fund "Y".

Each item on the list has a different context for the rules. Once the candidate list is created, determine the following:

Implementation priority

Set the priority for each item on the list with the business sponsor, and/or any other significant project stakeholders. Establishing the priority will allow the significance or criticality to be understood for the item within the current phase of the project. This technique is closely aligned with an agile or scrum approach to software application development when delivering a business capability.

Extremely low priority numbers could be an indicator of solutions that should be avoided all together.

Estimated rule count

Attempt to estimate the rule count, even if it's a rough estimate that needs to be refined. *High rule counts (300+) could be an indicator of legacy code bloat that might require additional decomposition work to sort out. Rule count numbers that are too low (< 30) may be better suited for alternate solutions.*

Estimated problem complexity

Try to gauge problems that will be difficult to solve versus ones that are just considered easy-to-do or low-hanging-fruit.

Sometimes the greatest value can be driven by problems with medium to high complexity. Problems with greater difficulty will typically produce a higher reward. Do not be afraid to select one medium to high problem for the initial release and avoiding all others until successfully implemented. Completing those early on will certainly help to drive future organizational adoption.

Rule change frequency

Lastly, determine how often the rules within the candidate solution(s) will need to change. The ability to change things quickly, responding to market demands is a key reason for investing in business rule technology.

Many organizations start by looking at their change request logs: processes, calculations, and logic that require frequent updates are often good candidates for implementation as business rules.

Static rule sets that don't change very often may be an indicator of poor implementation candidates. Judge the variability or frequency of the solutions. Low change rule sets can still benefit from this approach, but be advised that this could be perceived to be IT gadgetry-driven instead of business value-add driven.

Defining and estimating these metrics upfront will allow the targeted scope to be framed out (or boxed in) for each iteration of the rule harvesting effort.

Fostering transparency and visibility between the business and the implemented rule sets will produce a high level of accuracy and precision. This is driven by the elevated understanding between the individuals who run the business and the stewards of the technology.

Regardless of the outcome of this exercise, know that not every problem requires an automated solution. Avoid solving too many problems at once, especially in the initial release. Doing so could prove to bog down efforts, risking long-term organizational perceptions of the rules-based approach. The idea here is to address the highest business value-add items first, before continuing on with things that might just be a "nice-to-have".

An organization's change request logs can help identify candidates for implementing calculations and logic as business rules.

This process is depicted in the swim lane diagram below.

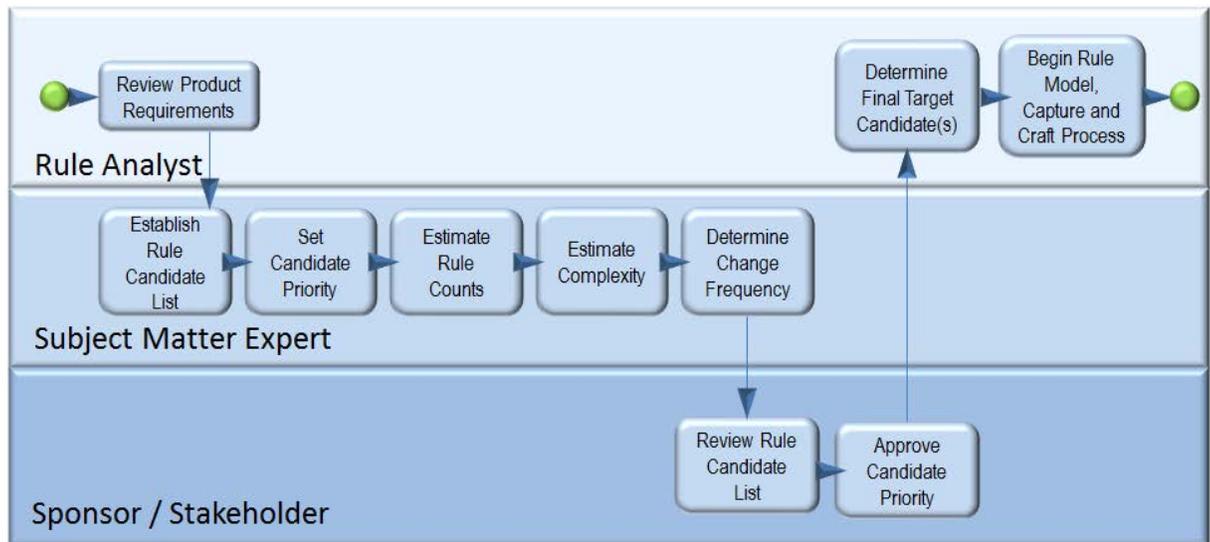


Figure 2: Determining Initial Scope Process

Developing a Planned Approach to Capturing Business Rules

Managing anything successfully doesn't happen by accident and harvesting business rules within an organization is no exception. The business rules must first be found before they can be implemented within InRule. Business rules are rarely going to be explicitly stated anywhere. It should be expected that no one single person, document, or automated system will be the sole source of every business rule that is needed. That is the problem now being solved and a primary reason for purchasing a Business Rules Management System (BRMS) like InRule in the first place.

Business rules can be and often are sprinkled throughout the organization. They are kept in various systems, contained within random documents, hidden in unidentified places, and sometimes are only known within people's heads. Extracting them from this source material must be organized and well planned to ensure completeness by avoiding gaps within the analysis (Morgan, 2002). Identify source material by targeting the artifacts, people, data, and legacy systems that will aid in solving the selected candidate solution(s) from the previous section. The following list provides an example of source material that should be available to the rule capture team:

1. Business or Government Documentation (MS Excel Spreadsheets)
2. Legacy System Source Code or Observing its Runtime Behavior
3. Current System Transactional or Data Warehouse Databases (Data Mining)
4. Subject Matter Expert (SME's) Knowledge

Once the source material is identified, develop a rule capture plan that includes processing the material as if mining for gold. Develop the process the rule capture team members will follow

to review the source material and identify any potential rules. It is also highly likely that new or previously hidden rules will be discovered when executing this process. The key word here is “identify”: during this phase the goal should only be to *identify* a possible rule, not prepare it for implementation (authored in InRule or coded in some other way). Modeling, refining, and crafting rules will happen during the later phases of this effort. In this early stage, it’s important to stay focused on the rule capture efforts by processing and cross referencing the source material. Producing a simple list of captured and unrefined business rules will suffice for now. Things do not have to be perfect during this phase in the cycle; just finding the rules is often hard enough.

Reviewing Existing and Available Documentation

Before conducting any sessions with business unit partners (SME’s), review all existing material first. This will cut down on revisiting already agreed to or known covered ground and helps avoid wasting time. Request the latest versions of the source documents and make sure that they are pertinent to the selected candidate solution(s).

To conduct a good analysis of the documents attempt to get electronic copies that can be marked up with plenty of notes and comments (Morgan, 2002). Highlight any statement within the document that could be a business rule or rule related: for example, anything that involves performing a calculation or making a decision, such as: “If the home is in a flood plain, then additional insurance is required” or “Orders over 300 lbs. must be shipped by ground freight.” Do not worry about making mistakes; often what is highlighted is only a clue that will lead to a final rule.

During the review process expect more documents to emerge as gaps are identified. Reviewing and analyzing documents is of course a manual process that requires the complete focus and attention of the person doing the work. Fully utilize the rule capture team by spreading the work out over several people. The calendar time demands of most modern IT projects will typically require more than one person to complete this work (Morgan, 2002).

Examining Legacy System Code or Observing Current Behavior

One of the most challenging tasks associated with the rule harvesting process is the interrogation of existing systems and source code. Unfortunately, reviewing existing source code is often a necessary and non-trivial task (Morgan, 2002). Depending on the age, language, and orientation of the code the analysis may require someone with a specific skillset to complete this work. Often the review of code is left to individuals who are considered to be programmers or software developers within an organization. These individuals are usually a scarce and highly constrained resource. They may not have the time or funding necessary to assist the team. Consider the options carefully and inquire about potentially outsourcing this work.

If there is no one available who can read the code, the rules analyst must observe the runtime behavior of the current system. This is done by entering records into the test environment of the system and recording the outcomes. The best way to conduct an exhaustive search for rules using this method is to intentionally make entry mistakes and document the error

messages. Those error messages are usually not rules, but can be considered additional clues because they represent the current set of automated decisions (Morgan, 2002).

Be advised that this type of brute force search through an existing system will take time to complete. This is especially true for expert users as they will naturally avoid data entry errors when navigating systems which they are comfortable using. Additional problems with the observation method include finding all of the functionality nuances or niche situations within the legacy system that don't happen very often. These areas are not always obvious and will sometimes require accessing parts of the application that are not regularly visited by users.

When mining legacy system behavior try to keep in mind:

- Is it possible to reasonably estimate how many business rules might be within the current systems?
- Of the existing set of business rules, can one estimate how many of those should exist going forward? In other words, before attempting to reverse engineer any legacy application, can it be determined if it is worth the effort to begin with?

As legacy systems are used over time, changes are frequently not implemented. This is usually the result of an intentional trade off related of the IT costs associated with the work versus the overall priority of the change. This common practice leaves many business rules being manually enforced by the users of the given application, which means the business rules are really stuck within peoples' heads. They in effect become the tribal knowledge of an organization and are passed down from one user to another (Ross, 2003). If this is the case, then it is safe to assume that only a percentage of the total business rules within the current system have enough value to be carried forward into InRule and others must be harvested from organizational knowledge. These gaps should be documented, verified, and tracked as the rules harvesting effort progresses.

Data Mining

Companies that have been in business for a while will typically have large amounts of business transactions and customer data at their disposal. This data represents business that an organization has already accepted as good and valid. Therefore, examining it can provide the rules analyst with examples of what good transactions look like so that they can be compared to rejected instances (Kargupta, 2004).

To explain all that would be required to conduct a full data mining effort is beyond the scope of this document. However, some helpful considerations insights into the data mining process are provided and interested readers may pursue further material on the topic.

Most data mining initiatives begin with the detection and identification of critical features or attributes within the targeted datasets. A feature is a "Region of Interest" (RoI) within the data. The output of any data mining activity is to compile a list of many features (RoI's) from the sourced datasets. Once those relevant features have been identified they are then characterized or categorized by their associations with other found features (RoI's). The goal of these exercises is to expose correlations within the data which lead to the discovery of any

potential business rules. Be mindful of business events which could have occurred and might be reflected within the existing state (Kargupta, 2004).

Accurately discovering patterns within datasets is a complex task. This effort usually involves the use of data analysis tools designed to facilitate and accelerate the work required. Most mid-sized or larger organizations will typically have a data warehousing and/or business intelligence group within their IT departments. Associating these efforts with that group would be beneficial if existing reports could be used to help in the rule harvesting process.

The use of existing reports would reduce the IT cost associated, but sometimes new reports can be produced with little or no effort. Regardless, it is worth investigating the possibility of aligning the rule capture team with the business intelligence group.

Conducting SME Interviews or Workshops

Up until now the rule capture effort could have been completed with little or no involvement with subject matter experts (SME's) or stakeholders from the business unit. At some point within the process it is a very good idea to meet with them and the sooner this can happen, the better.

Business rules are not always implemented within a system or documented within a report. They sometimes only exist in peoples' heads (Morgan, 2002). Maybe the business rules were too expensive to implement within the existing systems so automating them was avoided. Perhaps the rules that are there are some form or version that have now become stale or out-of-date. Regardless of the reasons, this is another significant problem for the organization to solve. What InRule provides is the capability for a business user (SME) to create and maintain their own business logic, resulting in their ability to share business knowledge across the entire organization.

Conducting structured interviews or workshops with the SME's has the following goals:

1. Request Feedback
2. Obtain Clarity
3. Identify Additional Unknowns

To achieve a precise business rules implementation it is important to continuously drive for clear, explicit definitions of things (Ross, 2003). These sessions are intended to capture the inputs from individuals who have the specific business knowledge that is needed to be successful. There is always a tendency to invite as many people as possible to these meetings. However, limiting the invitations to smaller, more focused groups by targeting individuals with specific knowledge in a particular area will result in more effective sessions.

A good size for a focus team is six people or fewer (Morgan, 2002). Larger groups will be more difficult to keep organized and managed. Involve only those individuals who can add value to the process and avoid selecting random participants. Ask for individuals with specific knowledge that is **directly related** to the selected candidate solution(s). Those who have enough history with the current solution to understand not just **how** its processes work, but

The sooner stakeholders and subject matter experts can be involved, the better.

why they must work in a certain way to achieve desired results (and why they may have been changed in the past), can provide valuable insight.

For larger organizations, conducting a small SME selection process prior to executing the rule harvesting effort might be necessary. Once these individuals have been identified it would be helpful to keep the same group engaged for the entire process. This will maximize the team's ability to produce consistent results.

The scope of this work will require multiple meetings, so before doing anything decide on an organized and planned approach. Break things down into manageable chunks and don't attempt to accomplish too much in any given session. During the sessions walk through the material and artifacts that have been collected so far and take plenty of notes. Post-meeting, cross reference the meeting notes with other material. Struggling through cross referencing tasks during the meetings will only slow things down and introduce potential distractions. It is better to perform the cross reference work between meetings as preparation for the next one (Morgan, 2002).

There may be things that come out of the sessions that will require further investigation and follow-up. Discovering new documents, systems, or previously unknown research areas is a good thing and is part of what should be accomplished with this effort. This process should form an iterative and playback cycle designed to drive understanding with everyone involved with this phase of the project. Begin each meeting with a brief summary from the previous session.

Do not get into any technical discussions during these meetings and focus entirely on business rule discovery. The purpose here is to drive out an accurate understanding of the business and depict how they see themselves (current state) or how they would like to see themselves (future state). Avoid conversations that discuss screen flows or system interactions (Ross, 2009). Those ideas and details will be flushed out during the technical design discussions. Separate the concerns of the business from the concerns of the technology in peoples' minds (Goodman, 2008). The rule harvesting effort will be more successful if people truly believe that they can understand their business rules outside the context of any system. Knowing how the pieces of knowledge build up to logic that forms the decision is a key technique.

Don't worry about obtaining agreement and consensus too early. It is likely that some SME's will disagree with each other and possibly even with the management group. These occurrences are normal and are typically resolved as a result of additional research and input from other stakeholders. To keep things moving forward, when disagreements occur redirect the group by assuring them a business rule approval and review process will be put into place before they are deployed to the production environment.

Things to Watch Out For

Albert Einstein said, "We can't solve problems by using the same kind of thinking we used when we created them." As with anything new, people might automatically reject these ideas and will resist the rule capture effort. This phenomenon is called FUD, or Fear, Uncertainty, and Doubt. In other words, be sensitive to any political issues that may exist. Although the likelihood of FUD being present within the project is high, know that having open and direct communication with those involved is the best way to handle things.

Patience is sometimes a formidable adversary and people need to understand that the rules-based approach really does work. Positive organizational changes will happen, just not overnight. Stay diligent and consistent with the message that this will be a positive change for everyone. They should know that it is only a matter of time before most companies are embracing this approach to developing rules-based systems and software. Organizations that avoid liberating their rules from source code will find themselves behind-the-times and will struggle to just catch up to everyone else. It may be helpful to point out that implementing decision logic and calculations as rules helps both IT (reduced maintenance burden, faster implementation) and SME's (more control over the logic that underlies their daily operations.)

Redundantly implemented business rules across legacy systems can form inconsistencies or paradoxes within the current environment. This may have been causing inaccurate business transactions to occur within the existing systems for years, which could have serious consequences for the organization. Retroactively fixing these problems goes beyond simply repairing the source code or implementing InRule. When these problems are discovered escalate them to high level business managers or executives. Be sure to prepare and gather all of the information regarding the issue(s) so that what is communicated is a complete and accurate picture. A documented impact analysis is a good way to ensure the exact representation of the problem.

Determining the historical costs associated with changes to hard coded logic in the legacy systems can also help to gain support for the new, business rule-based solution. It is not uncommon for organizations to experience great pain with changes to legacy systems. Some companies have lengthy IT processes in place which may take months or years to implement even the smallest change. This is typically based on the priority of the requests compared to all of the other IT work along with amount of files or lines of code that are required to change.

Sometimes even the subject matter experts may disagree with the implemented rules. Having inconsistencies requires cross referencing rules between the different sources mentioned above. This is sometimes a methodical and time consuming task that will need to be accounted for in the plan. That way the most valid (vetted) version of the rule can be carried forward within the new implementation. Know that this is completely normal and should be expected. This process can provide increased visibility to the management staff and give workers with the ability to sort out any inconsistencies on their own before the IT project begins. Remember, good analysis work always pays huge dividends in the end and attempting to avoid it as a cost cutting measure is never recommended.

The ability to detect and target rules is a vital skill for an expert analyst.

Defining Business Rules and Other Common Implementation Patterns

Before attempting to capture and document a rule it would be good to know what a rule is versus what it is not. The ability to detect and target rules is a vital skill for an expert analyst to master. True business rules are about ensuring a consistent business practice (Ross, 2003). They represent the knowledge of a specific business operation. Business rules are meant to be declarative; they apply in all circumstances in which their subjects are present.

The basic pattern behind every rule is to define it in a deterministic way, which creates a predictable outcome. The purpose of this design pattern is to position every rule as a controller of behavior within a prescribed context. This is commonly referred to as “subject must constraint” (Morgan, 2002) – for example, “a driver must be 25 to rent a car.” By their very nature business rules constrain a relationship between two or more business entities (Ross, 2009). (Not to be confused with source code objects or database tables, which are programmable abstractions of business entities.) Even rules that calculate a number are based on this fundamental principle. This is how business rules ensure a consistent business practice within an organization.

The presence of business rules within an organization predates modern technology. Therefore, one should be able to describe and use business rules without ever turning on a computer (Leannah, 2006). Business rules belong to the company and not its application servers. At their core, they simply do not care that they are being now being enforced by computer software. The rule is the rule and they fundamentally exist without an electronic dependency.

System Requirements and Behavior versus Business Rules

In some cases, SME’s may have a tendency to focus on data validation or standard behavior logic that is specific to a particular system that they are familiar with. Be aware that this approach represents rule implementation pattern which focuses on the system. Standard behavior logic like system data validation tasks which drive “requiredness” or “in-the-right format” are rules, but typically not of the business.

System based requirements are important in the delivery of good software solutions. System requirements or standard behavior logic like data validation tasks can and often are implemented within InRule as needed. These patterns include statements like “Customer first name is required” or “Date of birth in the wrong format”. Managing statements like these generally do not need to be represented as business rules and can be reserved for a later stage in the systems requirements process. Here is an example of a business rule pattern: “A customer must pay a \$25 refundable security deposit to rent a tuxedo”. Notice how this statement is separating the concerns of the business from the concerns of the system by focusing on a consistent business practice rather than a system requirement. It is important to know and understand this difference. Doing so allows things to be sorted out better prior to the rules implementation.

Business Processes, Workflow, and Business Rules

Business processes, activities, and workflows are often confused with business rules. Business rules are not about describing a process (Ross, 2003). They might participate within a given business process by controlling a specific step or decision point. However, the business rules themselves do not describe a process for completing tasks. They do not depict a series of steps for completing a transaction; that is what a business process or workflow does. Business rules are about influencing or executing decisions across all processes and systems regardless of their automation status. A well described business process can be used to identify where the rules might exist or play a role in some business activity.

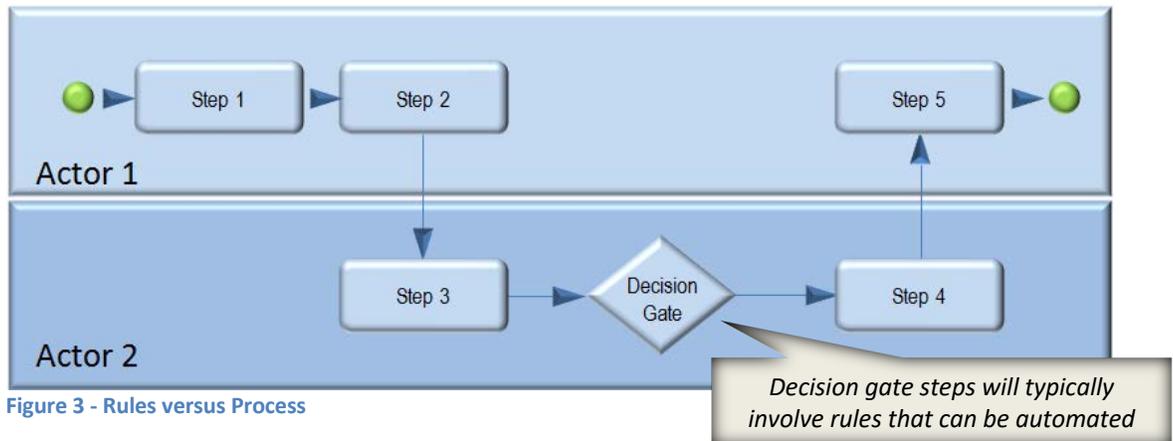


Figure 3 - Rules versus Process

Business rules are not events or transactions that have occurred. A business event is something that happens, like a customer renting a tuxedo. As with business processes, business rules are related to events but they are different. Business rules should constrain business events without directly describing them. Business events are commonly described as part of a Use Case or Fact Model. They can depict how a single rule could be applied to multiple events. (Ross, 2009)

(Please Note: Fact Models are described in a later section of this document.)

Establishing a Common Vocabulary and Reducing Translations

Before any serious work can be completed within the implementation it is absolutely critical that everyone is speaking the same language (Goodman, 2008). Unfortunately, this is harder than it sounds. What do you mean speak the same language; don't we do that already? Often times when an individual works for a company or within an industry for an extended period of time they become very knowledgeable within their respective business discipline. These Subject Matter Experts, or SME's, can exist in either an individual business unit or within the Information Technology (IT) department. Over the years, they adopt certain terminology or tribal knowledge which best describes their business operation (Ross, 2009). They assume that everyone within the organization has obtained a similar experience level and uses the same terminology. Of course often that is not the case, especially considering the typical divides that can exist between IT and the business unit. The entire organization needs to understand what their exact terminology is and how it is defined.

To accomplish this task, create a centralized glossary or dictionary for the organization that contains a list of common business terms and their corresponding definitions. (Ron Ross calls this the Concepts Catalog, Ross, 2003.)

A glossary is helpful for aligning organization-specific vocabulary and terms from multiple groups, for example a “program” to underwriters is often called a “product” by the sales team. Every term that appears in the Fact Model (explained below) or a business rule should be represented in the glossary. Coordinating the establishment and maintenance of the company’s glossary or Concepts Catalog is another step in business rule management. In general the goal of this step is to ensure greater accuracy and establish agreement on terms and vocabulary.)

Regularly reviewing the list with the SME’s during playback sessions will ensure the greatest accuracy of terms definition. This can be a simple list, documented and managed by any of the Microsoft Office products. The terms in the list should form the business concepts or business grammar surrounding the selected problems to be solved by the implementation.

When complete, this catalog of concepts will eliminate the need for human translation of the business terms and the chance that they become misinterpreted (Ross 2003). The glossary can be incrementally developed over time as more rules-based projects and initiatives are funded. Not every term has to be immediately identified and defined, only the terms required to support the targeted rule sets or selected candidate solution(s).

Once started and developed, the glossary should be made freely available to all of the company’s employees. The business terms must be explicitly defined and agreed too (Goodman, 2008; Ross, 2009). Understand that business rules and business terminology are inherently linked. Changes to one will impact the other. Changing the meaning of a term could completely invalidate a business rule. Changes or updates to the terms could be an indicator for future rule modifications.

Business rules are built on top of facts which are based on terms (Ross, 2003). This means that the rules themselves (and the rules engine) need a common vocabulary or context as much as the people do. A term is used by the business to share and communicate ideas about their organization. Terms should be declared in a natural language syntax and describe a single unambiguous concept. A term is a noun that represents something within the business. They represent the most basic things about a particular business problem or process. Terms must be atomic; representing a single object or idea that is indivisible. Terms need to be unique, similar

Case Study Example: Max Tuxedo Rental

This paper will occasionally refer to this example to explain concepts and terms.

Max Wilson owns a tuxedo business. The business is doing well and Mr. Wilson has just hired two new employees to help with the workload. Spring represents the stores’ primary renting season because demand is high and availability of good tuxedos is low.

The store has a computer system to keep track of the rentals and inventory along with performing some rudimentary calculations. However, most of the business decisions need to be made manually by either Max or his staff. To ensure a consistent business practice Max would like to deploy a Business Rules Management System. The focus of the new system is to ensure that customer service decisions are consistently made.

terms should not be repeated or overlapped with other terms. Term definition redundancy will only serve to breed confusion. Attempt to keep those cases to the smallest number of instances possible.

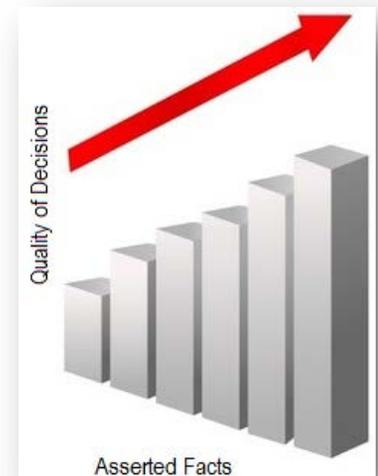
The following is an example glossary or Concepts Catalog for the business described in the example case study above.

Term	Definition or Description
Customer	The person who rents a tuxedo from the store.
Store	The place of business where tuxedos are rented.
Booking	Reserving a tuxedo for a rental period.
Rental Agreement	Defines the terms and conditions for how the tuxedo can be used, what fees are associated with its use, and when it must be returned.
Return Date	The date that a tuxedo must be returned without incurring a late fee.
Fees	The fees assessed to use a tuxedo for the rental period.

Modeling the Facts

The glossary is the collection of nouns with their own individual meanings. Somehow these nouns have to be linked together before they can lead to the creation of business rules. The Fact Model is the collection of verbs; business rules are a way of sharing knowledge. Facts are not business rules but they are something else that can be known. Facts do not constrain the relationship; they are the relationship between terms. To execute the business rules and make a decision InRule relies on the assertion (or retraction) of facts from within the problem space or landscape.

A fact represents knowledge from the real-world that is believed to be true and lacking bias. Facts can be thought of as metadata or data about data (Giarratano, 2005). They are the precursor to a business rule. A fact is a piece of consideration used to make a business decision. Facts shape the context for problem solving by providing the necessary information for which the system can reason. They ensure that a final conclusion can be drawn or reached for each request. The quality of the decisions that the system makes is in direct proportion to the amount or type of facts that are asserted to it at runtime (Leannah, 2006). This is typical for forward-chaining business rules engines. They move from the data to the decisions and InRule is such a device.



In order for the facts to be well organized and meaningful they must first be modeled (Ross, 2003). Facts represent what is possible to know about a business process or event, without constraining it. The intent of the Fact Model is to depict the relationships between the defined terms (facts). In the end, it is the relationships between facts that are ultimately constrained by the business rules.

The Fact Model is not a system, UML, Entity Relationship Diagram (ERD), or any other type of diagram commonly associated with the traditional software development approach (Ross, 2003). The Fact Model is its own independent visual representation of the truth. It forms the business case for the entire scope of selected candidate solution(s). It is a reflection of how the business views itself (current state) or how it would like to view itself (future state). It is not intended to represent a system or the interactions between systems (Ross, 2003). The following is an example Fact Model for the business case described in the example case study.

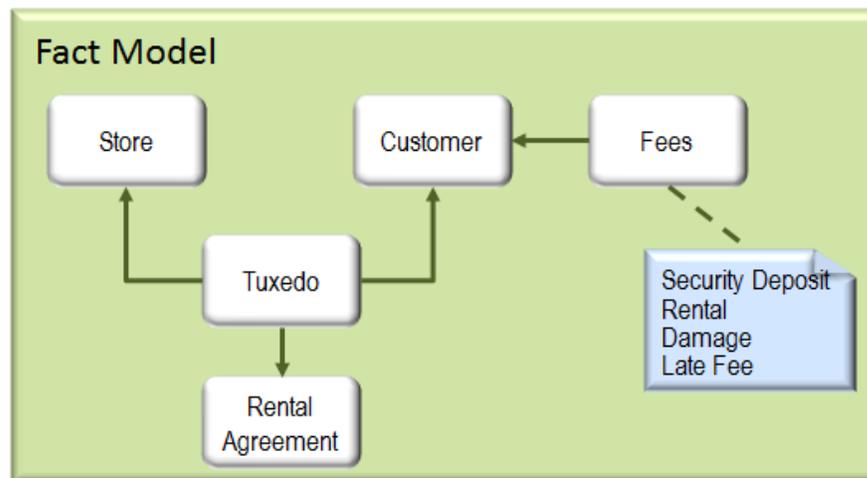


Figure 4: Example Fact Model

Crafting the Business Rules

Before a captured rule is implemented or authored, it is best to first refine the rule. The refinement process begins by comparing the rule against a model such as the Fact Model to determine if it applies to the current business case or selected candidate solution(s).

As a captured rule is refined, it is transformed into a crafted rule. One can think of a captured rule as the raw material that requires processing or manufacturing in order to become useful. Captured rules are likely to not even be rules, but other material that needs to be analyzed, decomposed, and broken apart (Morgan, 2002).

Before a captured rule can become a crafted rule it must have two things, a global unique identifier (E.g. R101) and a natural language rule statement. This provides more benefits than simply keeping things organized. It allows business rules to obtain organizational identities. It depicts the business rules as standalone items that naturally become first class citizens within the company (Ross, 2009). To achieve this status, the business rules must adhere to a particular format used during the rule crafting process.

The goal of the crafting process is to describe every rule as a human readable natural language statement before it is implemented within any system. The primary output of all of this analysis work is to produce a list of crafted business rules that can be reviewed, agreed to, approved, understood, and successfully authored. Each and every business rule must be simple, atomic, unambiguous, and understood by all (Ross, 2009).

The science of describing rules this way is best portrayed by the work of Ron Ross and his company Business Rules Solutions (BRS). They have a framework for describing business rules in an explicit, predictable, and structured way called RuleSpeak®. The RuleSpeak® rule description framework provides the guidelines and rules for how the business rules themselves must be represented in order to even be a rule. Ron's guidelines position each and every rule into a declarative constraining device ("subject must constraint") which can then be successfully authored and maintained within InRule.

For more information on the Do's and Don'ts of RuleSpeak® by Ron Ross follow this link: [Basic RuleSpeak® Guidelines](#).

Crafting a business rule takes practice and usually requires several iterations to get right. The following is an example list of crafted rule statements for the business case described in the example case study.

Rule ID	Rule Statement
R101	A customer must be charged a \$20 late fee for each day past the return date if the tuxedo is returned late.
R102	A security deposit must not be returned for any tuxedo that was damaged during its rental.
R103	A customer must be charged a \$50 dollar fee for any damaged tuxedo that must be repaired.
R104	A damaged tuxedo must not be rented to a customer.
R105	A customer must pay a \$25 refundable security deposit to schedule a tuxedo rental.

Notice how the crafted rules above are agnostic of any system implementation. They do not even refer to the presence of the application that may or may not enforce them. Likewise, the statements above do not explain a business process or event; they intentionally do not depict how to get something done. They are simply declarations of the business truth or knowledge that apply regardless of their implementation.

Conclusion and Summary

Good analysis work and decomposing problems into viable automated solutions will be the key performance indicator (KPI) of a successful InRule implementation. This work should not and cannot be avoided and requires the human brain to complete.

The purpose of a business rule is to ensure a consistent business practice within an organization across all of its operations. To accomplish these goals, the rules themselves must first be discovered and then refined or crafted so that they can stand alone as known assets of the organization.

Business rules must be liberated from lost documentation, source code, and tribal knowledge so that they can become first class citizens within the company. They must be explicitly defined, agreed to, managed, and used without translation regardless of an individual's experience level. Once the rules become a managed organizational asset, the company will gain meaningful insight, understanding, and knowledge in ways never experienced before. This process will allow the InRule Business Rules Management System to be leveraged in the best possible most decisive way that produces measurable results.

Acronyms

- BRE = Business Rule Engine
- BRMS = Business Rules Management System
- BRS = Business Rules Solutions (Ron Ross)
- ERD = Entity Relationship Diagram
- FUD = Fear, Uncertainty, and Doubt
- KPI = Key Performance Indicator
- IT = Information Technology Department
- ROI = Region of Interest
- SDLC = Software Development Lifecycle or System Development Lifecycle
- SME = Subject Matter Expert
- SOA = Service Oriented Architecture
- UML = Unified Modeling Language

Glossary

- Agnostic = Separation of concerns between the business rules and the system. Not bound to a specific system or implementation.
- Business Entity = A logical abstraction which defines a reference metaphor to something that exists in the real world.
- Concepts Catalog = The business term dictionary for the organization.
- Fact Model = The visual relationship between the business terms.
- Forward-Chaining = Moving from the data to the decisions.
- RuleSpeak® = A standard framework for describing and identifying business rules created by the Ron Ross research group, BRS.
- Rule Statement = The natural language (English) version of the business rule.
- Separation of Concerns = The process of separating a software implementation into discrete functional areas with little overlap. A modularized implementation of the business rules into a self-contained unit.
- Service Oriented Architecture = A set a design principles which positions applications as a collection of interoperable services.
- Sponsor = A person or group that provides the necessary funding for the project.
- Stakeholder = A person or party of interest within an implementation.
- Subject Matter Expert = A person with specific business knowledge or expertise within a targeted domain.
- Temporal Redundancy = Rules that are implemented multiple times within the same code base to address a gap in the logic, but often lead to creating a paradox. A common problem in sequential machine implementations.

References

- Erl, Thomas, SOA: Principles of Service Design. Boston, MA: Prentice Hall, 2008
- Giarratano, Joseph C. and Gary D. Riley. Expert Systems: Principles and Programming, Fourth Edition: Principles and Programming. Canada: Course Technology, 2005
- Goodman, Loren. Getting to the Handshake: A ROAD Map for IT/Business Collaboration. Chicago, IL: InRule Technology Inc., 2008
- Kargupta, Hillol and Anupam Joshi and Krishnamoorthy Sivakumar and Yelena Yesha. Data Mining: Next Generation Challenges and Future Directions. Menlo Park, CA: The MIT Press, 2004
- Leannah, William J. Exploring Business Application Agility through Rules-Based Processing: Centralizing the Business Rules of an Enterprise. Milwaukee, WI: Marquette University, 2006
- Morgan, Tony. Business Rules and Information Systems: Aligning IT with Business Goals. Boston, MA: Addison-Wesley Information Technology, 2002
- Ross, Ronald G. Business Rule Concepts: Getting to the Point of Knowledge. Third Edition, Huston, TX: Business Rules Solutions, LLC, 2009
- Ross, Ronald G. Principles of the Business Rule Approach. New York, NY: Addison-Wesley Information Technology, 2003
- Taylor, James and Neil Radon. Smart (Enough) Systems: How to Deliver Competitive Advantage by Automating Hidden Decisions. Boston, MA: Prentice Hall, 2007.