

1. Rules

The Merriam-Webster dictionary defines anarchy¹ as “a situation of confusion and wild behavior in which the people in a country, group, organization, etc., are not controlled by rules or laws.” Rules prevent information anarchy².

Rules, or business rules, guide, control, suggest, or influence behavior. Rules cause things to happen, prevent things from happening, or suggest that it might be a good idea if something did or did not happen. Rules help shape judgment, help make decisions, help evaluate, help shape behavior, and help reach conclusions.

Don't make the mistake of thinking that business rules are completely inflexible and that you cannot break rules. Sure, maybe there are some rules that can never be broken. Maybe there are some rules that you can break. It helps to think of breaking rules as penalties in a football game. The point is that the guidance, control, suggestions, and influence offered by business rules is a choice of business professionals. The meaning of a business rule is separate from the level of enforcement someone might apply to the rule.

A rule states a fact about the world (declarative rule). A rule can provide instructions (production rule).

1.1. *Human-readable and machine-readable business rules*

Business professionals interact with facts and rules every day and may not even realize it. Most business rules are in human readable form. But business rules can be represented in both human-readable form and machine-readable form. With the move to digital, more and more business rules are being represented in both human readable form and more importantly machine-readable form. Machine-readable business rules help automate processes which have been manual in the past.

Rules and facts must be written in some formal language in order for computer software applications to reason using the rules and draw conclusions about facts. Such a computer software application is often called a rule engine.

1.2. *Business rules drive intelligent software agents and expert systems automating work*

The global consultancy firm Gartner classifies XBRL as a transformational technology³. Gartner defines transformational as something that “enables new ways of doing business across industries that will result in major shifts in industry dynamics”. Major shifts means lots of change and some winners and some losers.

An example of one major shift is provided by what professional accountants call the “disclosure checklist”. Accountants creating financial reports often use accounting and reporting checklists or “disclosure checklists” as memory joggers to help them

¹ Anarchy definition, Merriam-Webster, <http://www.merriam-webster.com/dictionary/anarchy>

² *Understanding that Business Rules Prevent Anarchy*, <http://xbrl.squarespace.com/journal/2016/7/15/understanding-that-business-rules-prevent-anarchy.html>

³ Charles Hoffman and Liv Watson, *XBRL for Dummies*, page 145

get the reports right⁴. These memory joggers were created to be read by humans and can be a couple hundred pages. What if a financial report was structured, such as an XBRL-based public company financial report that must be submitted to the U.S. Securities and Exchange Commission? What if these human-readable memory joggers could be made machine-readable? And what if an intelligent software agent could be created to automate the manual task of checking a financial report to make sure that report was mechanically correct.

Note the statement “mechanically correct”. This is a very important distinction. No computer program will ever have the judgement of a professional accountant. But computer programs can perform work if the financial report is structured and the necessary business rules are made machine-readable. How much of a disclosure checklist can be automated? That percentage is as-of-yet to be determined. Perhaps 20% can be automated or even 80% will be automated. Maybe even a higher percentage. The probability 0% of a disclosure checklist can be automated is extremely low.

Humans augmented by machine capabilities, much like an electronic calculator enabling a human to do math quicker, will empower knowledge workers who know how to leverage the use of those machines.

1.3. Business rules are metadata and follow the rules of formal logic

Business rules provide a thick metadata layer that enables computer systems to perform useful work. The more business rules that exist, the more work a computer system can perform.

A simple example of a business rule the accounting equation⁵ which is universally accepted: *Assets = Liabilities and Equity*. The accounting equation is an axiom. *Axioms* describe self-evident logical principles that no one would argue with. The accounting equation is the foundation upon which double-entry accounting is built. *Theorems* are deductions which can be proven by constructing a chain of reasoning by applying axioms in the form of IF...THEN statements. These axioms and theorems are the basis for formal logic⁶. Formal logic is a discipline of philosophy which has been around since the days of Aristotle. Computers work based on formal logic. But you have to be careful. Computers are dumb beasts (see the section *Major obstacles to harnessing the power of computers*). Computers cannot follow all the rules of logic, only a limited set. To effectively get the tool, the computer, to perform work correctly, you need to follow certain guidelines and not push the tool beyond its capabilities.

⁴ Automating Accounting and Reporting Checklists, <http://xbri.squarespace.com/journal/2016/5/5/automating-accounting-and-reporting-checklists.html>

⁵ Wikipedia, *Accounting Equation*, retrieved August 3, 2016, https://en.wikipedia.org/wiki/Accounting_equation

⁶ Wikipedia, *Logic*, retrieved August 3, 2016, <https://en.wikipedia.org/wiki/Logic>

1.4. Basic example of business rules, fundamental accounting concept relations

A basic example of business rules is the fundamental accounting concept relations⁷. The fundamental accounting concept relations build upon the accounting equation. For example, assets can be broken down into current and noncurrent portions. And so you can create the rule: Assets = Current assets + Noncurrent assets. Another rule is: Liabilities = Current liabilities + Noncurrent liabilities.

But accountants understand that not every economic entity breaks assets and liabilities down into their current and noncurrent portions. For example, banks report using an unclassified balance sheet, reporting only total assets and total liabilities.

To address this, another rule is created. Economic entities are partitioned into groups based on how they report, their reporting style. Those that provide classified balance sheets and therefore report current and noncurrent assets and liabilities are put into one group. Those that report using an unclassified balance sheet, reporting only totals for assets and liabilities, are put into a different group. And so the result is rules that are universally applicable to all economic entities. Not every economic entity uses every fundamental accounting concept relations rule, only rules that apply to the group that the economic entity is in are applicable. This is only a brief description of the fundamental accounting concept relations. See the actual fundamental accounting concept relations metadata⁸ for a more comprehensive explanation.

An example of a more comprehensive set of business rules is accounting and reporting checklists⁹ that are used by professional accountants to create external financial reports. Today, these business rules are organized in the form of a human-readable accounting and disclosure checklist which are used as a memory jogger by professional accountants creating a financial report.

1.5. Business Rules and Business Professionals

Business professionals interact with business rules every day and may not even realize it. Most business rules are in human-readable form. But business rules can be represented in both human-readable form and machine-readable form. With the move to digital, more and more business rules are being represented in both human readable form and more importantly machine-readable form. Machine-readable business rules help automate processes which have been manual in the past.

1.6. Business Professionals Create and Maintain Business Rules

Business professionals create and maintain business rules. Imagine an information technology professional being responsible for maintaining the fundamental

⁷ *Public Company Quality Continues to Improve, 84% are Consistent*, <http://xbrl.squarespace.com/journal/2016/7/1/public-company-quality-continues-to-improve-84-are-consistent.html>

⁸ *Fundamental Accounting Concept Relations*, <http://xbrl.squarespace.com/fundamental-accounting-concept/>

⁹ *Automating Accounting and Reporting Checklists*, <http://xbrl.squarespace.com/journal/2016/5/5/automating-accounting-and-reporting-checklists.html>

accounting concept relations rules. That simply would never work. To maintain the fundamental accounting concept relations rules one needs to have an intimate understanding of financial accounting and financial reporting.

As Article 9 of the *Business Rules Manifesto*¹⁰ states, business rules are of, by, and for business people; not IT people.

- 9.1. Rules should arise from knowledgeable business people.
- 9.2. Business people should have tools available to help them formulate, validate, and manage rules.
- 9.3. Business people should have tools available to help them verify business rules against each other for consistency.

Rather than creating tools that only IT professionals can use because they are so complicated; business professionals need to demand software tools that properly expose functionality that exposes business rules to business users such that business users are working with business domain knowledge, not technical details that should have been buried deeply within the software applications.

Business professionals need to understand the *Law of Conservation of Complexity*¹¹ which states that complexity can never be removed from a system, but complexity can be moved. The *Law of Conservation of Complexity* states: "Every application has an inherent amount of irreducible complexity. The only question is: Who will have to deal with it - the user, the application developer, or the platform developer?"

1.7. Reusing and sharing business rules

The *Business Rules Manifesto*, Article 4¹², points out that business rules should be declarative rather than procedural. The declarative approach has important advantages including that your business rules become reusable across both processes and software platforms. As such, the rules become both highly re-engineerable and highly re-deployable.

Declarative involves stating *that* something is the case. Procedural involves stating *how* to do something. The following is a simple example of procedural rules and declarative: Suppose you desire a cup of coffee.

Procedural:

1. Go to kitchen.
2. Get water, coffee, sugar, cream.
3. Heat the water on the stove until the water boils.
4. Put the coffee, sugar, and cream into the water.
5. Bring the result to me.

Declarative:

¹⁰ *Business Rules Manifesto*, <http://www.businessrulesgroup.org/brmanifesto.htm>

¹¹ Understanding the Law of Conservation of Complexity, <http://xbrl.squarespace.com/journal/2015/5/24/understanding-the-law-of-conservation-of-complexity.html>

¹² *Business Rules Manifesto*, Article 4. Declarative, Not Procedural, <http://www.businessrulesgroup.org/brmanifesto.htm>

1. Get me a cup of coffee.

Taking a procedural approach you define the entire process and provide each step necessary to obtain the desired result. Taking a declarative approach you state the desired result, and let the system determine the best way to get that result; all you care about is the result without worrying how the result will be achieved.

A procedure is used in only one way, but a declarative specification can be used in many different ways¹³. Again, either a procedural or a declarative approach to representing business rules will work. It is important to understand the pros and cons of each approach.

Next, business rules should not be mixed within software application code. Why? There are three reasons. First, if business rules are within application code then it takes a programmer to change the code. Second, if the business rules are embedded within one software application that it is challenging to reuse those same rules within another application. Third, sharing business rules becomes easy.

Think of what it would mean if you could create a spreadsheet and test the spreadsheet against a shared set of rules provided somewhere on the Internet or on a private intranet.

This does not mean that all business rules become publically available. Using security rules could be available only to a department, or a company, or even across an entire supply chain.

Basically, just as applications and the databases they use were separated from one another (they were combined in the early days of software); business rules and applications are now being separated.

An example of business rule reuse can be seen via the fundamental accounting concept relations rules^{14,15}. All the metadata for these rules are declarative in nature and represented using the XBRL global standard. One software vendor, XBRL Cloud, uses the fundamental accounting concept relations rules to validate XBRL-based financial filings which are submitted to the SEC¹⁶. Another software vendor uses the exact same metadata and completely different software to query XBRL-based financial filings^{17,18}.

1.8. Types of Rules

The best way to understand what business rules are is to have some examples of business rules. Business rules can be put into general groups by type of rule.

¹³ John F. Sowa, *Fads and Fallacies about Logic*, page 3, <http://www.jfsowa.com/pubs/fflogic.pdf>

¹⁴ *Fundamental Accounting Concept Relations*,
<http://www.xbrlsite.com/2016/fac/v3/Documentation/#home>

¹⁵ *Crash Course in the Fundamental Accounting Concepts*,
<http://www.xbrlsite.com/2016/Prototype/FundamentalAccountingConceptsPureXBRL/Documentation/FundamentalAccountingConcepts.html>

¹⁶ *XBRL Cloud's Validation Report*,
<http://xbrlsite.azurewebsites.net/2016/ReferenceImplementation/2016-04-15/evidence-package/index.html#USFACRenderingSummary.html>

¹⁷ *Reporting Style Examples*, <http://www.xbrlsite.com/2016/fac/v3/Examples/Index.html>

¹⁸ Comparison Formats,
http://www.xbrlsite.com/2016/fac/v3/Examples/Index_Compare.html

Different people group business rules into different groups such as "structural rules" or "behavioral rules". Others break them down into "quality logic" and "business logic".

1.8.1. General types of business rules

At their essence, business rules articulate information about something or about the relationship between one thing and some other thing. Some examples that can help you better understand exactly what they are:

- **Assertions:** For example asserting that the balance sheet balances or "Assets = Liabilities + Equity".
- **Computations:** For example, calculating things, such as "Total Property, Plant and Equipment = Land + Buildings + Fixtures + IT Equipment + Other Property, Plant, and Equipment".
- **Constraints:** For example, specific behavioral constraints that control when it is appropriate to create, update, or remove information.
- **Continuity cross-checks:** For example, if a fact is used that fact does not conflict with or contradict other facts.
- **Process-oriented rules:** For example, the disclosure checklist commonly used to create a financial statement which might have a rule, "If Property, Plant, and Equipment exists, then a Property, Plant and Equipment policies and disclosures must exist."
- **Regulations:** Another type of rule is a regulation which must be complied with, such as "The following is the set of ten things that must be reported if you have Property, Plant and Equipment on your balance sheet: depreciation method by class, useful life by class, amount under capital leases by class ..." and so on. Many people refer to these as reportability rules.
- **Instructions or documentation:** Rules can document relations or provide instructions, such as "Cash flow types must be either operating, financing, or investing."
- **Relations:** How things can be related, such as whole-part relations. For example, how the business segments of an economic entity are related.

1.8.2. Categories of business rules from the XBRL perspective

Business rules can be categorized. The Business Rules Group provides a good summary of business rule categories. But sometimes how terms are used differ between groups using such terms. Below I have provided a set of business rule categories inspired by the Business Rules Group categories which is then reconciled to XBRL terminology.

- **Definition of business terms:** The very definition of business terms is a category of business rule. Each term is a rule. In XBRL, the report elements defined in an XBRL taxonomy schema is how business terms are defined. Terms are essentially identifiers. In XBRL, terms are grouped into one of the following categories of terms: Network, Hypercube (a.k.a. Table), Dimension (a.k.a. Axis), Member, Primary Items (a.k.a. Line Items), Primary Item (a.k.a. a concrete Concept), and Abstract (a.k.a. an abstract Concept or Primary Item). Business professionals have to go through a process of naming things

that exist in reality and giving them names as contrast to providing additional preferred labels for names that already exist. These terms describes how business professionals think and talk about real world notions, ideas, and other such phenomenon. The definitions of terms in the past have been documented in the form of human-readable glossaries. We now make these terms human-readable and machine-readable by defining them in XBRL taxonomy schemas. Information technology professionals sometimes define terms in the form of an entity/relationship model.

- **Structural assertions:** This term appears to describe two types of structures in XBRL:
 - **XBRL Fact:** A fact in XBRL is something that is reported within an XBRL instance. A fact is a structure comprised of other structures generally defined in the form of terms in an XBRL taxonomy schema but there are a few things defined in the XBRL instance itself (entity identifier, period, XBRL footnotes). So, a fact is a hard-coded structural assertion defined by the XBRL technical specification. A fact has an aspect model. This is the same as what I call the multidimensional model of XBRL.
 - **Other XBRL relations:** This category of structural assertions includes all other relations and resources definable using XBRL including presentation, calculation, definition, labels, references, and XBRL formulas.
- **Action assertions:** Action assertions constrain or influence behavior in some way. Action assertions cause things to happen or prevent things from happening. They can also be used to make suggestions. XBRL Formula provides for existence assertions, consistency assertions, and value assertions.
- **Derivations:** A derivation is a mathematical algorithm or a logical inference (deduction) that is used to derive, or what I have called impute, other structural relations (i.e. XBRL facts or other relations). Derivations create new knowledge based on existing knowledge. XBRL Formula has a mechanism for creating new facts.

A bit of clarification is helpful to make sure all of the above is clear. The notion of derivations might not be familiar to some people or you might be familiar with it in different terms. Here is some clarifying information that distinguishes between explicitly provided facts and derived facts:

- **Base Fact:** a base fact is a fact that has been explicitly reported in a financial report. For example, if you report the fact "us-gaap:Assets" for a specific economic entity for a specific period.
- **Derived Fact:** a derived fact is a fact whose value is created by an inference or mathematical computation. For example, if the base facts "us-gaap:Assets" and "us-gaap:AssetsCurrent" are reported then the fact "us-gaap:AssetsNoncurrent" can be derived because of two pieces of information: (1) the values of us-gaap:Assets and us-gaap:AssetsCurrent are known and (2) the business rule "Assets = Current assets + Noncurrent assets" is known; so deductive reasoning can be used to obtain the derived fact "us-gaap:AssetsNoncurrent".

- **Derivation:** a derivation is an algorithm used to infer or compute a Derived Fact. (i.e. a business rule). In the derived fact example above, the derivation is "Assets = Current assets + Noncurrent assets". There are two types of derivations:
 - **Logical inference:** a logical inference is a Derivation that produces a Derived Fact using logical induction (from particulars) or deduction (from general principles).
 - **Mathematical inference:** a Derivation that produces a Derived Fact according to a specified mathematical algorithm.

1.8.3. Logical layers of business rules

Business rules can be organized into convenient logical layers that interact with one another. These layers¹⁹ contribute to helping business professionals manage, use, and otherwise organize and interact with business rules.

- **Flow logic:** (sequence, process or flow)
 - **Procedural logic** – model sequence, loop, or iterative procedures.
 - **Flow logic** – fully automated sequence of operations, actions, tasks, decisions, rules.
 - **Workflow logic** – type of flow logic, semi-automated or manual processes that need an action to be taken from outside the system by another system or human.
- **Information assertions:** (compliance, quality, consistency, completeness, accuracy)
 - **Business terminology logic** – definition of business concepts
 - **Validation logic** – validate action assertions.
 - Action assertions
 - Structural assertions
 - **Decision logic** – type of validation logic, handles execution que and conflict resolution
 - Execution logic
 - Conflict resolution logic
 - **Derivation logic** – deviations which derives new facts using existing facts, rules, and logical or mathematical reasoning
 - Logical inference
 - Mathematical inference

1.9. Scaling Business Rules

Managing business rules becomes more complex as the number of rules increase. Scaling business rules is important. Using a decision model based approach can help

¹⁹ This is one of the better explanations of these different layers that I have run across, <http://wiki.flexrule.com/index.php?title=Logic>

manage large sets of business rules. The article, *How DMN Allows Business Rules to Scale*²⁰ points out four primary problems that you run into:

- **The ‘Rush to Detail’:** business rule development encourages policy makers to focus on rule implementation prematurely, before they have considered the broader goals and structure of their business decisions and to what extent they will be automated. This approach is like starting to build a house by laying bricks, rather than drawing plans and establishing foundations.
- **Poor Dependency Management:** a growing and poorly understood set of inter-dependencies between rules causing changes to have unintended consequences—making the rule set brittle and reducing its agility.
- **Insufficient Transparency:** the bewildering size of a rule set, use of technical (rather than business) terms and style for expressing rules and a poor connection between rules and their business context (their rationale and place in the business process)—making the meaning and motivation of rules more obscure.
- **Lack of Growth Management:** poor discipline about the scope, quality and placement of rules that are added to the rule set—making it hard to find rules and leading to ‘stale’ rules and duplicates.

1.10. Constrained Natural Language Business Rules Syntax

There are all sorts of syntaxes for business rules. Natural language is one syntax or format²¹. By constrained natural language we mean: (a) Only words from a specified, limited set of words is allowed; (b) only specified forms of phrase and clause is used; and (c) words, phrases, and clauses are only used in specified combinations.

The benefit of constrained natural language business rules is that they are easy for business professionals to read. A properly created constrained natural language business rules set can be converted to any other syntax using machine-based processes.

²⁰ Jan Purchase, *How DMN Allows Business Rules to Scale*, November 19, 2015, <http://blog.luxmagi.com/2015/11/how-dmn-allows-business-rules-to-scale/>

²¹ Graham Witt, *Writing Natural Language Rule Statements — a Systematic Approach: Part 1 — Basic Principles*, <https://www.brcommunity.com/articles.php?id=b660>