

Business Rule Solutions Ron Ross

Engineering “Why”

Have you ever been confused about why you were not allowed to do what you tried to do? Been judged or evaluated in a way you didn't expect? Stumped by the result or decision a business system produced?

If so you are a 'why victim.' In today's business world all of us are why-victims more and more often. The remedy is business rules. Not technical rules masquerading as business rules, but real business rules expressed of, by, and for the business represented purely in business terms.

We call the larger solution *Why Engineering*. This Column introduces Why Engineering and its basic principles, along with the Why Button. Find out what it takes to be an effective at Why Engineering in today's business environment.

The Why Button

Business rules are all about answering the question *why?*. Why things are disallowed. Why specific judgments or evaluations are made. Why certain decisions are reached. Imagine you had a *Why Button* handy whenever you encountered some disconnect in day-to-day business operations. Hit the Why Button and presto – answers appear in the form of relevant business rules.

Not technical rules but rather business rules you can read and understand no matter whether you're a business manager, business product developer, operational worker, or IT professional. A single representation accessible to all audiences. Representation that is:

Precise enough to remove all ambiguity.

Detailed enough to produce the same results no matter whether applied by workers 'manually' or automated by machines.

What would that do for your business? For one thing it would keep know-how from walking out the door – i.e., make your business logic explicit, not tacit, so you can retain it.

For another it would eliminate semantic silos – people using the same words, but not really communicating. Overall it would mean stepping up to do business intelligently in the knowledge economy.

The Who of Why Engineering

To achieve this vision you need to become a *Why Engineer*[™].

A Why Engineer is not a knowledge engineer in the sense of expert systems and not a technical wizard in ontologies. Rather, a Why Engineer is someone who uses rigorous discipline to capture, represent and communicate business rules based on carefully engineered business vocabulary. A Why Engineer is someone who: Takes great care – and pride – in how things are expressed and defined.

Can probe deeply into the ‘why’ of business logic never once using a term or structure whose origin lies in IT or system design.

What can a Why Engineer offer your requirements process? Business rules provide the ‘why’ – the basic rationale – for business requirements and elements of system design. They provide a solid basis for motivating each part of the solution you envision.

Today’s system-driven approaches result in a lot of arm-waving about the motivation for business requirements. A great many pages of generally formless documentation are often produced that no one really reads.

All that is far from harmless noise. It detracts from delineating:

The core business policies needed to actualize the business strategy.

The elemental know-how that differentiates your product/service and provides the basis for achieving excellence in its delivery.

How have requirement methodologies strayed so far from the very know-how that keeps you in business?! The Why Engineer puts things back on track.

The What of Why Engineering

Why Engineering is based on three fundamental principles:

Principle 1. The same complete, intelligible, unambiguous, deployable meanings (business rules and definitions) should be presented to all key audiences in a business – managers, business product developers, operations, and IT professionals.

Principle 2. A Why Button should be part of every architecture and business solution.

Principle 3. The same form of “why” answers (business rules) created originally should be re-used and provided to each audience in identical form whenever the Why Button is hit.

Obviously, answers should be available only to those duly authorized – but authorizations are simply more business rules.

Why Engineering is engineering in the fullest sense of the word. All engineering strives to produce something useful for people or their organizations. In Why Engineering that product is business rules, explicit business logic.

In general, engineering requires two things: *source material* and *structural principles*. For Why Engineering:

The source material is literally *words* – or more accurately the concepts and meanings behind the words.

The structural principles indicate how business logic can be represented in an unambiguous, anomaly-free form that is free of any IT or system-design artifacts or bias.

Why Engineering is engineering at its best.

As in all good engineering the product of Why Engineering is *highly* re-usable. What you develop as business logic is directly re-usable as the answers produced by the Why Button in operation. Nothing more is required. It is *exactly* the same stuff – unified and re-used with pinpoint accuracy.

Good engineering is also always concerned with sustainability of the product. Point-in-time (“bandaid”) solutions are avoided. In Why Engineering, sustainability can be achieved by business-level rulebook management.¹

Learning and Applying Why Engineering

Why Engineering really has nothing to do with IT directly. It can be (and has been) used even where no automated system is being built. It’s a very pure form of business analysis.

In a sense Why Engineering is simply about highly precise business communication. Is that a skill every practitioner possesses naturally? Unfortunately no – not even close. It must be learned.

Fortunately, effective techniques for Why Engineering are available and have been proven in practice. They consist of structured natural language tools such as BRS ConceptSpeak™, RuleSpeak® and TableSpeak™.² These notations – really *thinking tools* – are based on a rich standard, SBVR (*Semantics of Business Vocabulary and Business Rules*)³, developed over many years by world-class experts in formal logic, linguistics, and software engineering. SBVR itself is based on ISO terminology standards.

Is Why Engineering hard? Yes and no. The organizing principles and thinking tools can be readily learned. As for any engineering discipline, however, there’s a definite learning curve. It takes diligence and practice to become really good at it.

Actually, the hardest part of Why Engineering is getting at the buried assumptions and know-how in people’s heads – or lost in the jumble of legacy systems. The thinking tools of Why Engineering simply offer professionals the essential means for discovery, representation and validation.

¹ “What Rulebooks, Rulebook Management and GRBS Are About”, <http://goo.gl/iBwsrE>

² http://www.brsolutions.com/b_ipspeakprimers.php

³ Refer to the SBVR *Insider* section on www.BRCommunity.com.

The ultimate prize – common understanding in explicit form – is something the Why Engineer must still work hard to achieve. If it were easy, everyone would already be doing it!

<http://www.brsolutions.com/>