

# Five Reasons to Build Agile Systems Using Business Rules Management Functionality



## Applies to:

SAP NetWeaver Composition Environment 7.1.1

## Summary

This article aims to provide five reasons why business rules management can significantly increase abilities to build applications that are easy to manage, yet at very competitive maintenance costs.

**Author:** Rajagopalan Narayanan

**Company:** SAP Labs India Pvt. Ltd.

**Created on:** 19 February 2008

## Author Bio



Rajagopalan Narayanan is a Product Manager in the SAP NetWeaver Product Management Business Process Management team. His focus area is Business Rules Management. He has many years of experience with Business Rules Management.

## Table of Contents

Introduction .....	3
About this article .....	3
1 – Modularity .....	3
2 – Consistency .....	4
3 – Simplicity and conciseness.....	4
4 - Self-descriptiveness and understandability.....	5
5 - Independently testable .....	5
Summary.....	6
Copyright.....	7

## Introduction

The biggest value of implementing business rules management functionality is realized in the maintenance phase of the application.

In today's competitive world, organizations realize they need to build business applications that are able to evolve as quickly as their business requirements change. It is also known that business rules driving operational decisions change really fast and independently of application software requirements.

Business rules management functionality components are designed to help organizations to build agile applications that can evolve at the pace of fast changing business requirements, and at the same time distribute responsibilities accordingly. Business users and analysts take direct control on business rules, while IT units control software requirements.

[Back to top](#)

## About this article

This article aims to provide five reasons why business rules management can significantly increase abilities to build applications that are easy to manage, yet at very competitive maintenance costs.

The key attributes of an agile system are:

1. Modularity
2. Consistency
3. Simplicity and conciseness
4. Self-descriptiveness and understandability
5. Independent testability

This article expands on each of these attributes and provides additional reasoning on how business rules management delivers its added value.

[Back to top](#)

## 1 – Modularity

By leveraging business rules management approach organizations are enabled to build modular applications. Organizations would model operational business decisions via business rules management functionality and automate these decisions with a rule engine. All rules that drive decisions would be captured and modeled into a business rules management component.

By following this approach organizations will realize the following benefits:

1. Any 'decision service' is an independent unit that can be reused by different applications. An example of a 'decision service' is a "loan pricing service".
2. Business rules are externalized and managed separately therefore same rules can be used across different decision services.
3. Following this example, application architects take ownership of the decision service and integration functions, while Business Analysts/Users and Rule Development teams are responsible about correct definitions of all business rules in place.

4. Rules, hence the whole business logic are clearly separated, development and integration responsibilities are very clearly defined, which leads to modular designs.

## 2 – Consistency

One of the major sources of error when business rules are concerned is that it is very difficult to ensure their correct and consistent application across different applications/systems.

Consistent approach towards all business processes and consistent application of business rules is crucial for every organization. Via introducing business rules management functionality organizations enable all their different units to work with a single version of the truth, thus realize a standardized deployment of rules and process optimization enterprise-wide.

Here is a summary of reasons how consistency is delivered via business rules management functionality:

1. All logically related rules will be organized and maintained by business functions rather than by a software module. This lessens duplication chances and improves consistency.
2. Same rules called from any number of places behave in exactly the same way.
3. Because rule driven decisions are independently testable, enforcing and achieving consistency across the enterprise is a lot less expensive.

[Back to top](#)

## 3 – Simplicity and conciseness

Business rules existing at one single organization may be numerous and very complex. In the majority of cases IT units are responsible for business rules implementations, which may lead to more difficulties, due to the fact that conventional programming models do not offer sufficiently powerful mechanisms to model business rules in their complete variety.

Using conventional programming models potentially results in very complex and hard to maintain business rules implementations. Business rules management components are designed to help organizations to overcome these challenges by enabling rules modeling in simple, concise and business user readable forms.

Here is a quick reasoning of why business rules management functionality adds value for building simple and user friendly rule representations:

1. Business rules management enables organizations to capture and represent business rules in a form that is close to how Business Analysts/Users can understand them. Per example:
  - decision tables (tabular rules similar to MS Excel spreadsheets)
  - sets of independent 'If – Then' rule statements
2. Business rules management eliminates the need to define complex abstractions to model business rules in code by providing out of the box abstractions for complex business rules.

[Back to top](#)

## 4 - Self-descriptiveness and understandability

Many decisions are based on complex data, for example taking an 'underwriting decision' for an insurance product. Complexity is driven by the fact that underlying business rules are numerous and interconnected/dependant, also all these rules are business related, based on organizational procedures, performance requirements or strategies, and therefore necessarily need to involve Business Analysts/Users into definition and validation phases.

In cases when business rules are not available in a simple, self-descriptive and understandable form convenient for offering to business users to define, manage and modify, here are some details on how business rules management functionality can be employed to alleviate the pressure and help IT and Business units overcome time constraints:

1. A 'pricing ruleset' per example will describe in great detail precisely which are the steps to follow when a pricing decision needs to be made. This high level of descriptiveness is not really achievable via code.
2. Pricing tables locked up in database tables are not able to communicate the complete picture in detail as effectively as a well designed decision table.
3. Documenting complex system behavior as it applies to business decisions becomes very simple when business rules technologies are used.

Providing self-descriptive ways to define, manage and change business rules naturally leads to decreased maintenance and training costs, as well as alleviates time pressures on the users who need to understand and apply these business rules.

[Back to top](#)

## 5 - Independently testable

Business rules can change at a different pace compared to the rest of the application. For example, custom regulations may change several times per year.

It is important therefore, to make sure that changes to business rules are independently testable from the application. If rules are locked across application code, that makes the task very challenging and time consuming.

Via business rules management organizations are enabled to build applications where business rules, by being externalized by virtue are independently testable. To summarize:

1. One of the main drawbacks of conventional programming techniques when it comes to enabling decision automation is testing.
2. When business rules change, as they will, impact testing will lead to ever higher costs and turnaround times.
3. Via business rules management different teams can develop and test decision services in parallel and thus saving considerable time and resources.

[Back to top](#)

## Summary

Business rules management drives agile applications building and enables effective change management in every organization. Business rules management functionality allow delivery of modular applications where business rules modifications are processed independently from the rest of the system, and at the same time empower business users to take real ownership on defining, managing and modifying rules. IT professionals take ownership on software requirements solely and at the other hand business users become accountable for correct definitions of executable business rules, participating actively in and controlling change management efforts.

[Back to top](#)

## Copyright

© 2008 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, System i, System i5, System p, System p5, System x, System z, System z9, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, POWER5+, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

These materials are provided "as is" without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall not be liable for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials.

SAP does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third party web pages nor provide any warranty whatsoever relating to third party web pages.

Any software coding and/or code lines/strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.