Building flexible, easy to change and rock-solid applications with BRFplus decision services

Carsten Ziegler, James Taylor
Learning Points

- Learn how the empowerment of business experts is built into SAP applications

- See how to achieve continuous decision and process improvement

- Get step-by-step instructions on how to apply this approach in your systems
Presenters

- Carsten Ziegler – SAP
  - SAP-wide Business Rules Architect
  - Chief Product Owner BRFplus
- James Taylor – Decision Management Solutions
  - CEO, Decision Management Solutions
  - Leading expert in Decision Management and the application of Business Rules
Agenda

- Presenters
- **Challenges in Enterprise Systems**
- The Decision Service Management Approach
- Decision Service Management with BRFplus
- ROI
- Best Practices and Learnings
“Once you pour electronic concrete, it’s hard to get out.”

*Shell*

“There is no such thing as a temporary solution. Don’t put in things you don’t want to keep.”

*Unilever*
Enterprise Applications

STOP

Wait rather than act

Report but don’t learn

Local exceptions
Global standards

Escalate rather than empower

Built to last, not to change
To improve they must make more decisions.

Before

Manual

Managed

Not Made

After

Manual

New

Managed

Not Made

Larger boxes represent more decisions, by volume.

Reproduced from Smart (Enough) Systems, Prentice Hall 2007
Decisions

Making choices or selections and taking action
Candidate decisions

- **Determine** if a customer is eligible for a benefit
- **Validate** the completeness of an invoice
- **Calculate** the discount for an order
- **Assess** the risk of a transaction
- **Select** the terms for a loan
- **Choose** which claims to Fast Track

- These are **decision** words
  - The system must **answer a question** each time
Problem 1: Waterfall does not work for decisions

- Standard methodology for implementation and change projects
- Assumes a phased/waterfall approach:
  - Blueprint > Realization > Testing > Productive use

Problems
- Assumes you know everything when you start, though you don’t
- No structure for ongoing optimization after go-live
- What you learn in realization cannot be fed back into your blueprint (particularly bad for decision-making!)
Decisions are high change components

- **Regulations** change  
  Change to keep eligibility decisions compliant
- **Policies** change  
  Change validation to track new data requirements
- **Competitors** change  
  Change the discount to remain competitive
- **Markets** change  
  Change the assessment to manage risk
- **Consumer behavior** changes  
  Change to keep selecting the right deal terms
- **Fraudsters** adapt  
  Change the routing to focus on new fraud
Automation vs. Agility or Business vs. IT

**Automation: Full**
- Hard-coded decisions

**Semi**
- Decisions in documents

**No**
- Decisions as tacit knowledge

**Agility:**
- **Low**
- **Medium**
- **High**

**Questions**
- What are my costs for changes?
- How to gain transparency for business users?
- How to enforce policies in business operations?
- How do operations scale?
- How to ensure process quality?
- How to achieve legal compliance?
Problem 2: Decisions, and rules, are not processes

- **Age < 21**: Accept low-risk applicant
- **21 ≤ Age < 50**: Process medium-risk applicant
- **Age > 50**: Decline high-risk applicant
Problem 2: Decisions, and rules, are not processes

- **Age < 21**: Accept low-risk applicant
- **21 ≤ Age < 50**: Previous Heart Attack, Existing Cancer
- **Good Medical Record**: Process medium-risk applicant
- **Age > 50**: Decline high-risk applicant
Problem 2: Decisions, and rules, are not processes

Real Experience. Real Advantage.
Problem 2: Decisions, and rules, are not processes
Embedding decisions cause process complexity
Separate decision, simpler process

<table>
<thead>
<tr>
<th>Age</th>
<th>Class I sports</th>
<th>Class II sports</th>
<th>Heart Attack?</th>
<th>Cancer?</th>
<th>New Customer?</th>
<th># Claims</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;21</td>
<td>N</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Low Risk</td>
</tr>
<tr>
<td>&lt;21</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>High Risk</td>
</tr>
<tr>
<td>&lt;21</td>
<td>Y</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Medium Risk</td>
</tr>
<tr>
<td>21 - 50</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>High Risk</td>
</tr>
<tr>
<td>21 - 50</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>High Risk</td>
</tr>
<tr>
<td>21 - 50</td>
<td></td>
<td>N</td>
<td></td>
<td></td>
<td>N</td>
<td></td>
<td>Medium Risk</td>
</tr>
<tr>
<td>&gt;50</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td>&lt;2</td>
<td>High Risk</td>
</tr>
<tr>
<td>&gt;50</td>
<td></td>
<td></td>
<td>N</td>
<td>&lt;1</td>
<td></td>
<td>&gt;1</td>
<td>High Risk</td>
</tr>
</tbody>
</table>
Process rules are not decision rules

Flow control, routing
IT ownership
Governed as part of the process
Lifecycle dependent on process lifecycle

Business Process

Decision Rules

Business decision making
Business-side ownership
Governed by the business
Independent lifecycle

After Gladys Lam, BRSolutions
Separating processes and decisions adds agility

**Application**
- Completeness check
- Data validation
- No-go criteria

**Credit Score**
- Calculation of credit score
- Enable/disable contract terms

**Approval**
- Automatic/manual approval
- Compliance rules
- Risk management

<table>
<thead>
<tr>
<th>Business Application</th>
<th>Banking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Process</td>
<td>Loan approval</td>
</tr>
<tr>
<td>Business Rule</td>
<td>Calculation of credit score</td>
</tr>
</tbody>
</table>

Business Application Release 1 ➔ Business Application Release 2


Problem 3: Change and Transport Management (CTS)

- CTS is a set of tools and a methodology to implement changes in DEV, transport changes to QAS, and upon success transport to PROD.
- CTS supports transport of customizing and repository objects:
  - Code, database/dictionary definitions.
- CTS locks customizing and repository from changes in PROD.

**Problem - one size does not fit all**

- Downtime required.
- CTS and working in the development system requires technical support.
- Master data needs to be replicated into DEV.
- Mandatory artificial classification of all artifacts into Customizing, repository, and application data (master data and transactional data) with strict change management; no changes of customizing in PROD.
Missed Opportunities

Traditional Approach

- Waterfall process with release cycles: spec, design, code
- Same change management for all objects (DB, UI, decision logic)
- High costs also for small changes

Questions

- How long will it take to get a requirement implemented when the cut-off date passed by?
- Will there be another project for my requirements in the future?
- What will be the costs of the project and who will have to pay for it?
- What will happen if the implemented decisions are outdated?
Agenda

- Presenters
- Challenges in Enterprise Systems
- The Decision Service Management Approach
- Decision Service Management with BRFplus
- ROI
- Best Practices and Learnings
Suitable Decisions

**Banking**: Relationship based Pricing, Scorecards, Credit Decisioning

**Education**: Fee Calculations, Course Selections

**Healthcare**: Claims, Patient Monitoring, Fraud Detection

**Logistics and Shipping**: Parts Management, Duties Calculations, Pricing Calculations

**Public Sector**: Tax Calculations, Customs Duties, Land Regulations, License Fee Calculations

**Insurance**: New Products, Claims Settlement, Agent Commissions
These decisions involve many kinds of rules...
public class Application {
    private Customer customers[];
    private Customer goldCustomers[];
    ...
    public void checkOrder() {
        for (int i = 0; i < numCustomers; i++) {
            Customer aCustomer = customers[i];
            if (aCustomer.checkIfGold()) {
                numGoldCustomers++;
                goldCustomers[numGoldCustomers] = aCustomer;
            }
        }
    }
}
Alternatives to business rules

**Explicitly coded within application or database**

<table>
<thead>
<tr>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Familiar to IT</td>
<td>• IT department must be involved in any change</td>
</tr>
<tr>
<td>• Simplifies development and testing processes</td>
<td>• Little re-use of logic across multiple decisions</td>
</tr>
</tbody>
</table>

**Parameterized in tables or configuration files**

<table>
<thead>
<tr>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Decreases cost of change</td>
<td>• Use of parameters can decrease readability</td>
</tr>
<tr>
<td>• No business/IT collaboration</td>
<td>• Assumes kinds of changes can be exhaustively specified</td>
</tr>
</tbody>
</table>
A BRMS like BRFplus is compelling because

- Design Transparency
- Execution Transparency
- Agile Compliance
- Collaboration
- Platform for analytics
From Missed to Realized Opportunities

Decision Service Management Approach

- Clear separation of decisions from processes (agile but robust systems)
- Domain experts own and change implemented decisions
- Permanent and incremental decision optimizations
Agenda

- Presenters
- Challenges in Enterprise Systems
- The Decision Service Management Approach
- **Decision Service Management with BRFplus**
- ROI
- Best Practices and Learnings
**Decision Service Approach: Customer Landscape**

**Decision Service Manager**
- Extremely low hardware requirements
- Requires NetWeaver 7.0 Enhancement Package 3 or NetWeaver 7.3 Enhancement Package 1

**Managed Systems**
- Existing servers with customer business applications
- No Upgrade required, works with NetWeaver 6.40 or higher (>6 years old)
- Any number of managed systems can be connected with the Decision Service Manager (RFC)
Decision Service Modeling with BRFplus

- Central creation, test and management of all rules and decisioning logic
- Service implemented in rulesets using decision tables, trees, text rules, formulas
- Transparency - no code but graphical modeling
- Access to all relevant data in the managed systems (e.g. master data used in value help of decision table rules)
Decision Service Approach: Deployment

- Deployment describes the transfer of the decision service into one or many managed systems.
- Hot deployment - immediate use with no downtime for deployment.
- Activity periods - deploy today with as-of usage timepoint.
- Overview about all decision services in all systems, deployment statistics.
Decision Service Approach: Use

Decision Service Use

- Local execution of generated code, mass-enabled
  - Best performance
  - No dependency on Decision Service Manager
- Execution traces (inputs, result, decision path for decisioning analytics or legal compliance)
Decision Service Approach: Continuous Improvement

Decision Service Optimization

- Super-fast change cycles in the hand of the domain experts: Analyze, Optimize, Implement
- Automated business decisions with full transparency for better decision quality
- Immediate ROI
**Decision Service Approach**

**Decision Service enabled customer system landscape**

- Deployment of one service into multiple systems
- Reuse of decision services and single rules, decision tables, etc.
- Non-SAP systems connected by web service
Agenda

- Presenters
- Challenges in Enterprise Systems
- The Decision Service Management Approach
- Decision Service Management with BRFplus
- ROI
- Best Practices and Learnings
The ROI of Decisions and Business Rules

1. Increased Business Value
2. Increased Development Efficiency
3. Increased Maintenance Efficiency

And in that order!
Business Value

- Reduced losses from more accurate fraud decisions
- Reduced fines from demonstrable compliance with regulations
- Quicker time to market with new products or services
- More competitive pricing scheme
- Productivity of business staff increases
- Redeployment of assets when staff no longer need to do repetitive decision-making
- Reduced time to value for new systems functionality with benefits of a system being realized sooner
- Reduced costs or losses while waiting for a change to a system such as rework caused by out of date system behavior.
Development Efficiency

- Increased specification productivity as business analysts are involved directly
- Improved developer productivity—adding 1 business analyst can eliminate the need for multiple programmers
- Reduced re-work as there is an improved quality of business rules
- Reuse of previously developed logic because it is easier to see opportunities for reuse and to take advantage of them when logic is in rules in a repository
Maintenance Efficiency

- Reduced effort to find where to make a required change
- Reduced effort to understand the existing code so the right change can be made
- Reduced effort to make the change
- Reduced effort to test the change to confirm it works
- Reduced effort to deploy the change
Agenda

- Presenters
- Challenges in Enterprise Systems
- The Decision Service Management Approach
- Decision Service Management with BRFplus
- ROI
- Best Practices and Learnings
Best Practice: Begin with the Decision in mind

Find the **decisions** that matter to your business and focus on them as well as your processes.
Decisions have owners.
So have decision services.

Decision services are the responsibility of business-side experts.
Best Practice: 3 Steps to Decision Management

- Discover
- Build
- Improve
Key Learnings

- Focusing on Decisions improves Enterprise Applications
- Decisions are different and not the same as Processes or Rules
- Managing Decision Services creates agility and business value
- BRFplus delivers Decision Services optimized for SAP/ABAP applications
- Use BRFplus Decision Services whenever one of the following is true:
  - Decisions are value drivers
  - Decisions are likely to change over time
  - Configuration with customizing tables will become too complex
Thank you for participating.

Please remember to complete and return your evaluation form following this session.

For ongoing education on this area of focus, visit the Year-Round Community page at www.asug.com/yrc

SESSION CODE: 2414