



Business Rule Management mit SAP BRF+

Anpassungen an regelbasierten Geschäftsprozessen in SAP erfordern meist einen Entwicklungsauftrag mit entsprechendem Overhead. Mit dem Framework BRF+ können Key-User fachliche Regeln aber selbst pflegen – und das nicht nur für SAP-Applikationen.

Von Ralf Wenzel

■ Viele Entscheidungen in Geschäftsprozessen verlaufen regelbasiert und werden programmiert – in SAP per ABAP. Das hat den Nachteil, dass jede Regeländerung einen Entwicklungsauftrag nach sich zieht, verbunden mit entsprechendem Overhead – was unflexibel, langsam und teuer ist. Zudem ist die Regelände-

rung Bestandteil einer Applikation, die üblicherweise eine deutlich kürzere Lebensdauer hat als das Regelwerk (nicht die konkrete Regel) an sich. Das Regelwerk wird also „gekapselt“, wie Entwickler sagen würden. Man kann die Applikation unabhängig vom Regelwerk ändern und umgekehrt.

Damit Key-User solche fachlichen Regeln selbst pflegen können, bedarf es einer Anwendung, die eine Regelpflege ohne Programmierkenntnisse ermöglicht und die man außerdem aus verschiedensten Umgebungen heraus aufrufen kann. In SAP heißt diese Komponente Business Rule Framework plus (BRF+). Sie ist keineswegs auf SAP-Applikationen beschränkt, wie man im Folgenden noch sehen wird.

Erstaunlicherweise nutzen recht wenige SAP-Systeme BRF+ – obwohl quer durch fast alle Geschäftsprozesse regelbasierte Entscheidungen anfallen. Dieser Artikel möchte die Geheimnisse um BRF+ lüften und anhand eines konkreten Beispiels den gesamten Prozess darstellen.

Das Beispiel ist bewusst einfach gehalten, damit der Fokus nicht auf dem Verstehen des Beispiels liegt. Zunächst wird beispielhaft ein einfaches Regelwerk in BRF+ aufgebaut. Sodann wird eine Klasse entwickelt, um die ABAP-Anbindung vorzuführen, und schließlich werden mit Unit-Tests verschiedene Fälle durchgespielt.

Hierzu bietet sich das SAP-Datenmodell SFLIGHT an, weil es erstens aus SAP-Schulungen hinreichend bekannt und zweitens auf jedem SAP-System vorhanden ist. Denn mit den herunterladbaren Quellen zu diesem Artikel kann jeder die Beispiele nachbauen, debuggen und zur Kenntnisvertiefung erweitern (siehe ix.de/zvub).

Die enthaltene Steuerungsklasse und die dazugehörigen Unit-Tests kann man ebenso einfach übernehmen. Allerdings ist auf hinreichend erläuternde Kommentare zu achten, wie auch im Downloadmaterial nachzuvollziehen ist.

Beim hier präsentierten ABAP-Coding fehlt die ungarische Notation, da ich ein entschiedener Gegner dieser Methode bin (siehe ix.de/zvub).

Das Regelwerk liegt als XML-Export vor, es lässt sich über die XML-Importfunktion der BRF+ Workbench in andere Systeme übernehmen. Theoretisch könnte man mit diesen Import- und Exportfunktionen am SAP-Transportwesen vorbearbeiten, aber davor sei an dieser Stelle ausdrücklich gewarnt. Wie konventionelle Programmierung auch gehört ein Regelwerk durch Simulation und Test geprüft und über das SAP-Transportwesen übertragen. Alles andere widerspricht den grundlegendsten Regeln von Softwarequalitätsmanagement.

Einführung in die BRF+ Workbench

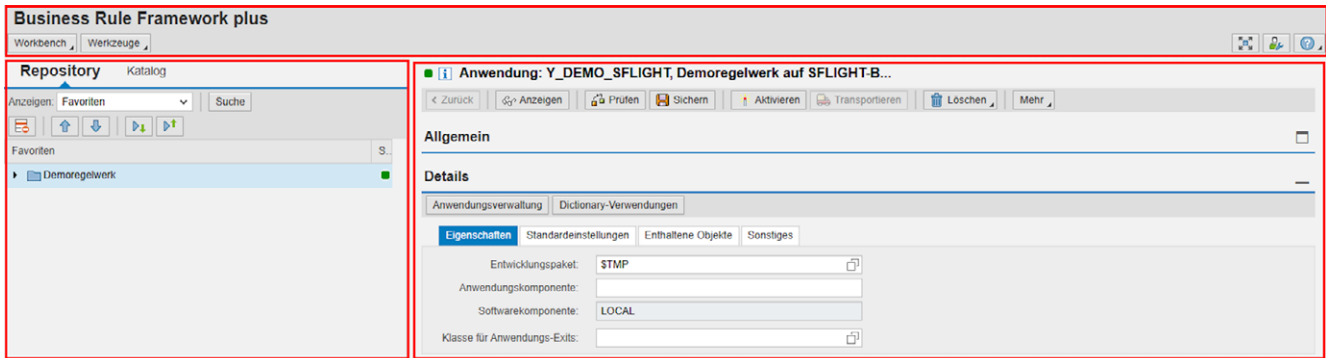
Das ehemalige Formula and Derivation Tool (FDT, das Kürzel taucht im Folgenden beispielsweise in den Namen der SAP-Klassen wieder auf), später BRF, ist jetzt als BRF+ eine Webdynpro-Anwendung. Man startet sie im Browser mit dem Transaktionscode FDT_WORKBENCH oder BRFLPLUS (siehe Abbildung 1). Dabei fällt auf, dass die Übersetzung ins Deutsche in vielen Fällen verbesserungswürdig ist.

Elemente eines Regelwerks

Grundsätzlich können mehrere Regelwerkselemente denselben Namen tragen; die Eindeutigkeit in der Datenbank stellen UUIDs her. Man könnte also alle

X-TRACT

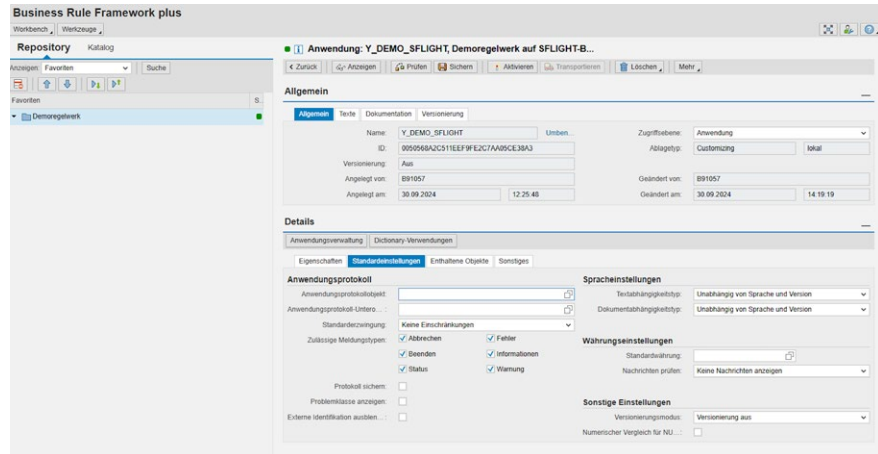
- ▶ Bei regelbasierten Geschäftsprozessen sind sowohl das Erstellen als auch Anpassungen mit Programmieraufwand verbunden.
- ▶ In SAP bietet das Business Rule Framework plus (BRF+) eine Regelwerks-Engine, mit der kompetente User ihre fachlichen Regeln selbst warten können.
- ▶ Mit BRF+ können Anwender den Weg ins SAP Decision Management beschreiten.



Die BRF+ Workbench zeigt links den Navigationsbaum und rechts den Arbeitsbereich; im Kopfbereich darüber finden sich die Personalisierung und globale Funktionen (Abb. 1).

Elemente gleich benennen, was natürlich nicht sinnvoll ist. Bei der Namensvergabe sollte man allerdings „gegenhierarchisch“ vorgehen: Üblicherweise vergibt man Namen von Objekten so, dass sie von links nach rechts spezieller werden, etwa SFLIGHT_ABFL_ZEIT_OK. Da in der Webdynpro-Oberfläche Namen aus Platzmangel oftmals abgeschnitten werden, sollte man den wesentlichen, unterscheidenden Teil so weit links wie möglich im Namen manifestieren.

Das oberste Element (also der oberste Knoten eines Anwendungsbaums) ist die Anwendung, die die einzelnen Teile eines Regelwerks bündelt. Darunter finden sich insbesondere Funktionen, die von außen ansprechbar sind. Sie bilden die Schnittstelle zwischen ERP-System und Regelwerks-Engine. Bei den Funktionen ist zwischen funktionsbasierten und ereignisbasierten Funktionen zu unterscheiden. Während erstere nur für einfache Funktionen oder zur Kapselung komplexer Ausdrücke dienen, eröffnen die ereignisbasierten Funktionen eine



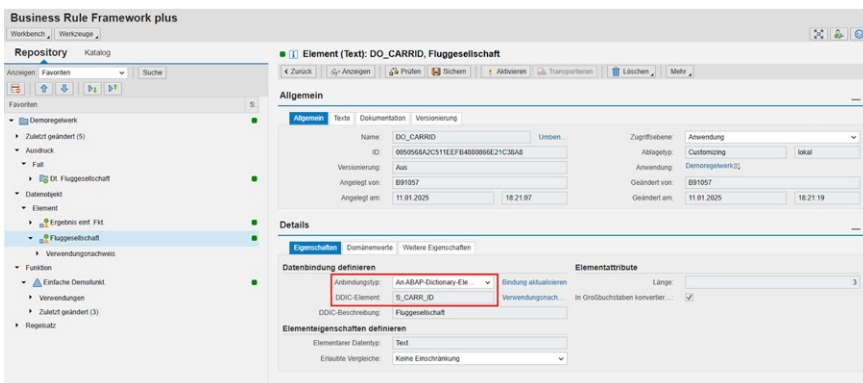
Die Anwendung ist die Basis jeder BRF+-Applikation und damit der Hauptknoten des Anwendungsbaumes (Abb. 2).

Fülle von Möglichkeiten, die BRF+ ausmachen.

Da kein Programm ohne Daten funktioniert, benötigt BRF+ Variablen, in denen die Daten gehalten werden. In BRF+ heißen sie Datenobjekte. Zumindest für die Import- und Exportparameter eines Regelwerkes sind Datenobjekte mit DDIC-Anbindung sinnvoll (Data Dictionary, das Datenwörterbuch in SAP). Innerhalb eines Regelwerkes kann man zweifelsohne auch auf Variablen ohne Anbindung zurückgreifen, vor allem auf intern, also innerhalb von BRF+ deklarierte. Zur Verfügung stehen Einzelvari-

ablen, aus mehreren Einzelvariablen zusammengefasste Strukturen und ganze Tabellen.

Operative Elemente in BRF+ sind beispielsweise Aktionen und Ausdrücke. Mit einer Aktion kann man etwa einen Workflow starten, eine Mail versenden oder eine Nachricht für das Anwendungsprotokoll erzeugen. Insbesondere können Aktionen Nachfolgeaktionen haben, sodass eine Regel eine ganze Kette von Aktionen auslösen kann. Ausdrücke sind besonders sinnvoll zur Modularisierung von Unterentscheidungen. Ein Ausdruck kann eine Fallunterscheidung, eine Entscheidungstabelle, eine Schleife, eine Tabellenoperation und vieles mehr enthalten. Sogar komplette Funktionen lassen sich innerhalb von Ausdrücken verarbeiten.



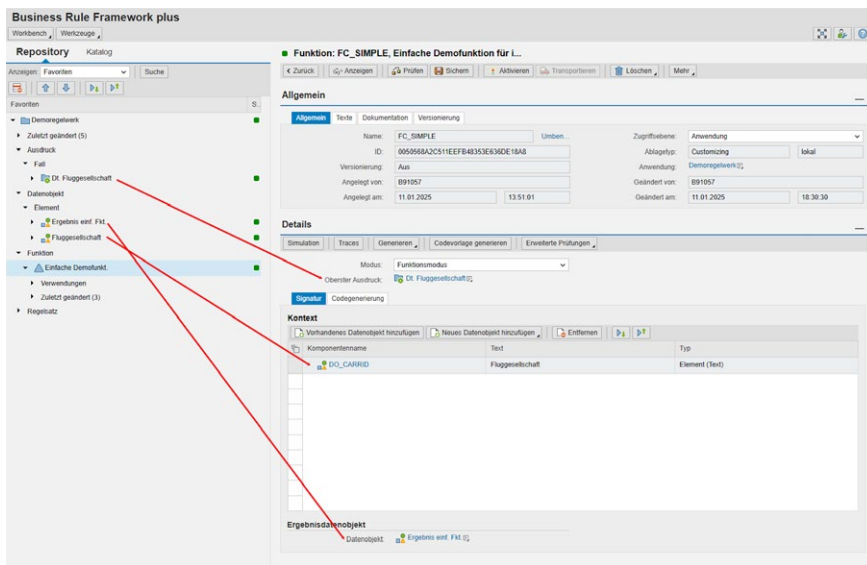
Das Datenobjekt DO_CARRID ist der Importparameter der Regel und steuert die Entscheidungen in BRF+ (Abb. 3).

Anbindung an SAP- und andere Systeme

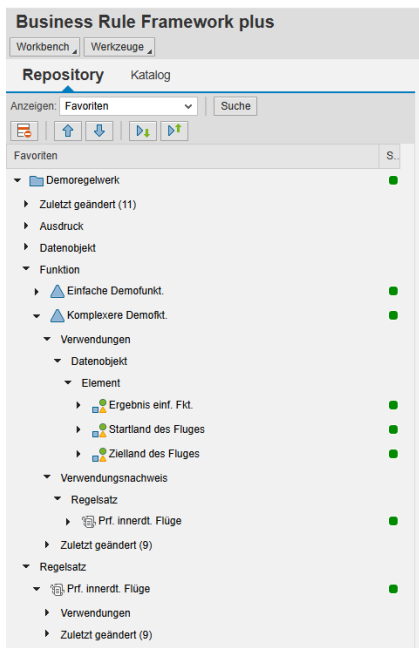
Beispiele für regelbasierte Entscheidungsprozesse im SAP-System selbst bietet unter anderem das SAP-Transportwesen. So kann man Eigenschaften des Transportauftrages oder seiner Objekte an ein Regelwerk übergeben und es entscheiden lassen, ob der Transportauftrag freigegeben werden darf oder nicht. Auch



Das Ergebnisdatenobjekt EX_CARRIER_IS_GERMAN ist die Variable, über die ein Aufrufer das Ergebnis der Regelbearbeitung erhält (Abb. 4).



Die Funktion FC_SIMPLE hat eine Signatur, die sich aus Kontext (Eingabeparameter) und Ergebnisdatenobjekt (Ausgabeparameter) zusammensetzt (Abb. 5).



Die komplexere Beispielfunktion verwendet einen Regelsatz, der Regeln zur Entscheidungsfindung definiert (Abb. 6).

Regelwerk am Beispiel SFLIGHT

Unser Beispiel soll die Arbeitsweise mit BRF+ verdeutlichen und nicht das Ausschöpfen aller Möglichkeiten vorstellen – was auch den Rahmen deutlich sprengen würde. Daher ist das hier aufzubauende Beispiel recht einfach gehalten. Ein weiteres, etwas komplexeres Beispiel steht im XML-Format als Download zur Verfügung (siehe ix.de/zvub).

Nach dem Start der BRF+ Workbench (mit Transaktionscode BRFPPLUS oder FDT_WORKBENCH) kann man direkt mit dem ersten Objekt des Beispielprojektes anfangen: Links oben im Navigationsbereich findet sich der Button „Anwendung anlegen“. Zunächst erstellt man darüber die Anwendung Y_DEMO_SFLIGHT mit beschreibendem Kurz- und Langtext. Im als XML herunterladbaren Beispiel ist noch eine kleine Dokumentation hinterlegt.

Der Einfachheit halber verwendet das Beispiel ausschließlich lokale Objekte. Im produktiven Entwicklungsbetrieb würde man stattdessen per Details/Eigenschaften einen Transportauftrag hinterlegen. Alle Standardeinstellungen können übernommen werden, genauer ansehen kann man sie sich über den gleichnamigen Reiter. Nach der abschließenden Aktivierung wechselt die Farbe des Statusbuttons von Weiß auf Grün, zu sehen rechts neben dem jeweiligen Element im Anwendungsbaum (siehe Abbildung 2).

Alle weiteren Objekte dieser Anwendung legt man am einfachsten über Rechtsklick auf den Namen der Anwendung im Navigationsbaum an. Hierzu gibt es im Kontextmenü den Menüpunkt Anlegen mit der Auswahl der Objektart. Das ist der einfachste Weg und funktioniert einheitlich für alle Objekte, sodass sich weitere Erläuterungen erübrigen.

Zusätzlich benötigt die zu erstellende Beispielfunktion einige Datenobjekte. Da ist zunächst ein Importparameter für die zu verwendende Funktion. Man generiert ihn über das Kontextmenü der Anwendung mit Anlegen/Datenobjekt/Element, benennt es DO_CARRID und bindet es an das SAP-Datenelement S_CARR_ID (siehe Abbildung 3). Für das Datenobjekt DO_RESULT_SIMPLE läuft das Anlegen analog, lediglich ohne Bindung an ein SAP-Datenelement. Hier reicht die Default-Einstellung Text völlig aus.

Die Beispielfunktion hat eine einfache Aufgabe: Sie soll bestimmen, ob die eingegebene Airline eine deutsche oder ausländische Fluggesellschaft ist. Das erfordert eine Fallunterscheidung, die man über das Kontextmenü der Anwendung unter Anlegen/Ausdruck/Fall... anlegt.

im Mahnwesen fallen viele Entscheidungen an, die regelbasiert sind und daher durch BRF+ durchgeführt werden können (und sollten!). Wie man sieht, fehlt es weniger an Anwendungsmöglichkeiten als an der Kreativität, sie zu finden und mit Leben zu füllen.

Natürlich kann man BRF+ auch für eigene Anwendungen nutzen, wie sich im Verlauf des Artikels zeigen wird. Konkrete Beispiele aus der Praxis wären etwa die Auswertung von Bluttests bei einem Blutspendedienst oder die Auswahl eines konkreten Versicherungsscheinformulars bei einer Rentenversicherung.

Wer Angst hat, dass man sich per BRF+ eine Insel aufbaut, auf die man von außerhalb von SAP keinen Zugriff hat, sei getröstet: BRF+ ist kein SAP-internes Tool. So lässt es sich über das Generieren eines Remote-Function-Call-Funktionsbausteins (RFC) auch an Nicht-SAP-Systeme anschließen. Somit können auch Anwendungen außerhalb des SAP-Ökosystems von den in BRF+ hinterlegten Regeln profitieren.



IX-Workshop „SAP BTP: Von Datensilos zu integrierten Geschäftsprozessen“

Die SAP Business Technology Platform (BTP) ermöglicht es Unternehmen, Daten aus verschiedenen Quellen zentral zu verwalten und für Analysen und Entscheidungen verfügbar zu machen – ein Muss in der heutigen datengetriebenen Geschäftswelt.

Dieser Workshop bietet eine umfassende Einführung in die Integration und das Datenmanagement mit der SAP Business Technology Platform (BTP) und zeigt, wie Geschäftsanwendungen mit SAP BTP verbunden werden können. Die Teilnehmer lernen, Daten aus verschiedenen Systemen zusammenzuführen, deren Qualität zu verbessern und eine einheitliche Datenbasis für Geschäftsprozesse zu schaffen. Darüber hinaus werden sie in die Lage versetzt, den reibungslosen Datenaustausch und die Konsistenz über alle Geschäftsanwendungen hinweg sicherzustellen.

Der Workshop findet online statt, weitere Informationen unter heise.de/s/lw1xn.

An dieser Stelle zeigt sich, dass die deutsche Übersetzung in BRF+ durchaus Schwächen besitzt – das englische „Case“ wird zu „Fall“ statt „Fallunterscheidung“. Den Ausdruck der Fallunterscheidung benennt man mit `EX_CARRIER_IS_GERMAN` und versieht ihn mit einer sinnvollen Beschreibung. Als Ergebnisdatenobjekt erhält er als Rückgabewert das soeben angelegte `DO_RESULT_SIMPLE`. Dessen Bedingung besagt, dass bei der Fluggesellschaft LH ausgegeben werden soll, dass es eine deutsche Fluggesellschaft ist, andernfalls, dass es keine deutsche Fluggesellschaft ist (siehe Abbildung 4).

Funktionen als Schnittstelle zum BRF+

Der besondere Status einer Funktion als Schnittstelle zum aufrufenden System wurde eingangs schon beschrieben. Die Beispielfunktion `FC_SIMPLE` verknüpft jetzt den Eingabeparameter mit dem Ergebnisdatenobjekt (Ausgabeparameter) über den Ausdruck. Angelegt wird die Funktion ebenfalls über das Kontextmenü der Anwendung per Anlegen/Funktion unter Vergabe des Funktionsnamens `FC_SIMPLE` und eines beschreibenden Textes (siehe Abbildung 5).

Verwendet wird der sogenannte Funktionsmodus mit `EX_CARRIER_IS_GERMAN` als „Oberster Ausdruck“, `DO_CARRID` als Eingabeparameter und dem Ergebnisdatenobjekt `DO_RESULT_SIMPLE`.

In BRF+ sind die Arbeiten nun abgeschlossen. Über den Button Simulation kann man die Funktion testen, wie man das von Funktionsbausteinen und Klassen in SAP kennt.

In ABAP lässt sich die BRF+-Funktion schon jetzt über die Methode `CL_FDT_FUNCTION_PROCESS->PROCESS()` finden und aufrufen. Sie ist syntaktisch korrekt, besitzt ID und Signatur und ist aktiviert. An dieser Stelle können ABAP-Entwick-

ler anfangen, die Einbindung in den Prozess zu programmieren.

Die Steuerungsklasse dient dazu, die Anfrage an BRF+ schnittstellengerecht aufzubereiten und zu übergeben sowie die Entgegennahme des Ergebnisses sicherzustellen. Man könnte sie auch Rahmenprogramm nennen. Im Rahmen dieser Steuerungsklasse werden auch die wichtigsten BRF+-Aufrufe gezeigt, die das ABAP-Repository anbietet. Das Coding soll hier im Einzelnen nicht erklärt werden, das ist Aufgabe der dort eingefügten Kommentare (siehe ix.de/zvub). Zum Lernen ist es gut, sich in der Funktionsdefinition der BRF+ Workbench über den Button Codegenerierung den Code zur Funktion erzeugen zu lassen.

Die zweite Funktion, `FC_COMPLEX`, ist etwas komplexer und im XML enthalten, das zum Download bereitsteht. Eine Erläuterung würde den Rahmen des Artikels sprengen. `FC_COMPLEX` benötigt andere und insbesondere mehr Objekte, die das XML-Beispiel mitliefert. Hier wird zum Beispiel gezeigt, wie sich eine Regel in einem Regelsatz im Rahmen eines Entscheidungsprozesses verhält. Es soll dazu einladen, sich daran auszuprobieren und vor allem die Steuerungsklasse entsprechend zu erweitern.

Zum besseren Verständnis des komplexeren Beispiels seien hier noch die dort verwendeten Begrifflichkeiten Regelsatz und Regel erläutert (siehe Abbildung 6).

Eine Regel ist eine (optional zeitlich begrenzt gültige) Ablaufbedingung, für deren Realisation es viele Möglichkeiten gibt. Jede Regel lässt sich einzeln deaktivieren, sodass man durchaus mehrere Regeln quasi auf Vorrat definieren und nach Bedarf einfach ein- und ausschalten kann.

Eine Sammlung von sequenziell nacheinander abzuarbeitenden Regeln bildet einen Regelsatz (Ruleset). Eine ereignisbasierte Funktion kann man mit einem Regelwerk verbinden, das bei Aufruf der

Funktion abgearbeitet wird. In diesen Regelwerken kann man Funktionsbausteine aufrufen, Fallunterscheidungen machen, Entscheidungsbäume und Entscheidungstabellen hinterlegen und vieles mehr. Diese einzelnen Fähigkeiten lassen sich auch verschachteln, sodass man etwa einen Funktionsbaustein aufruft, der einen Wert ermittelt, den man in einer Entscheidungstabelle sucht. Anhand des dort gefundenen Wertes kann das Regelwerk eine Fallunterscheidung machen, ob dieser gültig ist oder nicht. Die potenzielle Komplexität solcher Entscheidungsprozesse ist schier endlos.

Damit Anwender sich das Verhalten des Programms vorab ansehen können und der Code für die Steuerungsklasse nicht abstrakt in der Luft hängt, wurden kommentierte Unit-Tests beigelegt. Diese zeigen an konkreten Beispielen, wie sich die Steuerungsklasse aufrufen lässt. Die Beispielfälle kann man etwa im Debugger durchlaufen lassen und so die Arbeitsweise der Steuerungsklasse Schritt für Schritt nachvollziehen.

Fazit

BRF+ bietet eine Regelwerks-Engine, deren zentraler Vorteil ist, Teil des gigantischen SAP-Baukastens zu sein. Wer die Einführung von BRF+ erwägt, stellt sich die Frage nach Produkten von Mitbewerbern. Die Anbindung an einen Nicht-SAP-Mitbewerber ist signifikant komplexer und Drittanbieterlösungen haben alle den Nachteil, dass sie von außen zugreifen (Restriktionen) und sie sich völlig anders bedienen lassen. Geübte SAP-Anwender (was Key-User zweifelsohne sind) dürften sich in der Webdynpro-Anwendung hingegen schnell zurechtfinden. Wer sich für BRF+ entscheidet, erhält in jedem Fall eine integrierte, im Handling bekannte und flexible Lösung, auf deren Basis man den Schritt ins SAP Decision Management gehen kann. (avr@ix.de)

Quellen

Links zur ungarischen Notation und Downloadquellen für den Code und das generierte XML der Beispielanwendung siehe ix.de/zvub.

RALF WENZEL

ist seit über 20 Jahren freiberuflicher SAP-Applikationsentwickler und betreibt die Unternehmensberatung Heuristika in Hamburg.

