RULES & DECISION ENGINE EVALUATION TEMPLATE

Feature or characteristic	IDIOM v3	Market Alternative	Reason it is important
INTEGRATION FEATURES			
Access Methods			
Web Services/BPEL	**Yes**		Increasingly popular web access protocol
Native Calls	**Yes**		For all platforms; easiest to code, fastest to execute.
DLL	**Yes**		Standard PC deployable callable envelope
IBM Business Rule Beans	**Yes**		Allows management of rules within IBMs proprietary rules architecture
Messaging			
JMA/JMS	**Yes**		Java standard asynchronous communications
MSMQ	**Yes**		Microsoft standard asynchronous communications
Business Object Models			
Document Object Model (DOM)	**Yes**		Industry standard object model
JDOM	**Yes**		Alternate Java version
Platforms			
Java	**Yes**		Major open source platform
.Net	**Yes**		Dominant desktop and major corporate platform
C++	**Yes**		Legacy high speed execution
Performance			
Compilable for maximum performance	**Yes**		Millisecond(s) for full sets of commercial decisions (100 decisions delivered in 2milliseconds)
Native classes	**Yes**		Needed where optimal performance is required
Stateless for unlimited runtime instances	**Yes**		Scales to the maximum that the environment can support
Support for very large business objects	**Yes**		Modern business objects are getting increasingly complex (millions of nodes e.g. MDDL)
Auditable			
Logging of all rules fired	**Yes**		Optional for performance reasons
Regression testing	**Yes**		Review and confirm the integrity of audited rules

RULES MANAGEMENT			
Ease of Use			
No scripting	**Yes**		No arcane knowledge needed to use the product; no language to learn
Wizard managed rules builder	**Yes**		Error free rules; Non technical interface
"Mind Map" style Decision Trees	**Yes**		Free form development of decision/rule context
Palette based rule construction	**Yes**		Lego style rules construction using pre-built resources
Color coded development	**Yes**		Entire UI is color coded for easy navigation and context awareness
Visible rule-firing sequence	**Yes**		Predictable runtime execution
Flexible Rule Selection			
Apply rules for past dates	**Yes**		Rerun "as at" prior dates
Preload rules for forward dates	**Yes**		Pre-build and pretest rules for future use
Temporary rules	**Yes**		Support for short term overrides
Rules by exception (context dependent Rules)	**Yes**		Allow runtime parameters to automatically select rule overrides
Meta Rules (rules to control rules)			
Change effective date	**Yes**		Run different parts of an objects rules at different dates
Rollback/Redo	**Yes**		Rule shown to be wrong? - then roll it back or repeat
Change context	**Yes**		Need to run different sets of rules on the same document? - then change context
Documentation			
Automatic English generation of rules	**Yes**		Present the business knowledge in business English
"Where Used" for all rule elements	**Yes**		Understand the impact of change
Authorization and Control			
Workgroup based authorization	**Yes**		Authority is devolved to expert domains for self-management of rules
User based change Log	**Yes**		Individuals still accountable for specific changes
Vocabulary Maintenance			
Facts and terms defined via XML schemas	**Yes**		Using the most pervasive standards for shared definition of vocabulary; creation of a vocabulary can be expensive in some rules products
Management of changes in terminology	**Yes**		Changes of terminology must be reconciled with existing rules
Support for very large Schemas	**Yes**		Public domain schemas now render millions of nodes (e.g. MDDL); private schemas typically render tens of thousands.
Automatic rendering of complex (type) Schemas	**Yes**		Terms must be defined in context, therefore complex types must be fully resolved prior to presentation.
Rule Integrity			
Single possible result per rule	**Yes**		No conflicts or collisions
Management of datatype	**Yes**		Reduces opportunities for error
Collections processed as a Unit of Work	**Yes**		Reduces rules development effort, manages complexity

Verifiable			
Deductive rules	**Yes**		Rules are statically defined for predictable and testable outcomes
Scenarios can be tested	**Yes**		Testing at the same level that the rules will operate at
Rule firing sequence and results tracking	**Yes**		Traceability is at the level that the user defines the rules
Scale			
Rules for multiple domains	**Yes**		Allows assembly of complex processes across multiple parties
Multi-user rules repository	**Yes**		Allows sharing of rules across domains
COMMERCIAL			
Vendor is niche focused (no other products)	**Yes**		Focused product avoids platform and tool clash; small footprint, no architectural "baggage" or extraneous functionality
ISV Support			
Restricted rules management for ISV Customers	**Yes**		To support customer self-managed customization
"Black-box" deployable ISV Rules	**Yes**		Easily deployed upgrades to the application's business rules
Many integration, platform, and deployment options	**Yes**		One source of rules for all customers regardless of platform
Unlimited Deployment			
No deployment fee's	**Yes**		No additional runtime costs; maximizes deployment options
All platforms supported from same rules	**Yes**		Deploy to multiple platforms in parallel
Rules can be swapped between Business Partners	**Yes**		Business Partners can provide each other with proprietary rules
Support for Third Party rules suppliers	**Yes**		Industry standard rules can be supplied as a service to multiple consumers
Price	**Yes**		Compare our price!