



- (51) 국제특허분류: 미분류
- (21) 국제출원번호: PCT/KR2015/001778
- (22) 국제출원일: 2015년 2월 24일 (24.02.2015)
- (25) 출원언어: 한국어
- (26) 공개언어: 한국어
- (30) 우선권정보: 10-2014-0028140 2014년 3월 11일 (11.03.2014) KR
- (71) 출원인: (주)이노룰스 (INNORULES) [KR/KR]; 135-879 서울시 강남구 삼성로 100길 83층, Seoul (KR).
- (72) 발명자: 조도연 (CHO, Doyeon); 130-756 서울시 동대문구 전농로 10길 20, 101동 503호, Seoul (KR). 현창환 (HYUN, Changhwan); 138-850 서울시 송파구 가락로 19길 3, 201호, Seoul (KR). 김성민 (KIM, Sungmin); 138-110 서울시 송파구 거마로 9길 19 101동 1706호, Seoul (KR).
- (74) 대리인: 남정훈 (NAM, Jung Hoon); 135-717 서울시 강남구 논현로 507, 1301, Seoul (KR).
- (81) 지정국 (별도의 표시가 없는 한, 가능한 모든 종류의 국내 권리의 보호를 위하여): AE, AG, AL, AM, AO,

AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

- (84) 지정국 (별도의 표시가 없는 한, 가능한 모든 종류의 역내 권리의 보호를 위하여): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 유라시아 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 유럽 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

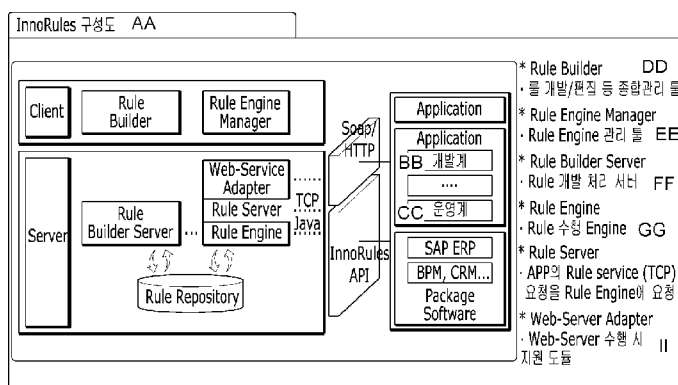
공개:

- 국제조사보고서 없이 공개하며 보고서 접수 후 이를 별도 공개함 (규칙 48.2(g))

(54) Title: BUSINESS RULE MANAGEMENT SYSTEM HAVING HIERARCHICAL RULE STRUCTURE AND REPRESENTATION METHOD THEREOF

(54) 발명의 명칭: 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템 및 그 표현 방법

[도1]



AA ... Innorules configuration diagram
BB ... Development system
CC ... Operation system
DD ... Comprehensive management tool for rule development/editing, etc.
EE ... Rule engine management tool
FF ... Rule development processing server
GG ... Rule performing engine
HH ... Request rule service (TCP) request of APP from rule engine
II ... Assistance module during performing web-server

(57) Abstract: The purpose of the present invention is to provide a rule representation template for IT nonprofessionals to easily and conveniently represent company task rules in a non-programming manner through a user interface similar to Excel. To this end, provided is a business rule management system (BRMS) having a hierarchical rule structure, the business rule management system (BRMS) generating and managing rules for performing a determination method regarding tasks, and solving the tasks using the rules. The business rule management system (BRMS) having a hierarchical rule structure comprises: a rule processing requesting unit for receiving a rule processing request for a task; a rule generation and management unit for developing and managing rules and generating the generated rules as a specific rule template as needed; a rule storage space for storing the generated rules; a rule cache unit for storing a result value processed using the rules; and a rule engine for solving a task using a rule capable of processing a task corresponding to the rule processing request.

(57) 요약서:

[다음 쪽 계속]



본 발명은 IT 비전문가들이 기업의 업무 규칙을 엑셀과 유사한 사용자 인터페이스를 통해 Non-Programming 방식으로 쉽고 편리하게 표현하는 규칙 표현 템플릿을 제공하는 것이 목적이다. 이를 위해서, 업무에 대한 결정 방법을 수행하는 룰(rule)을 생성 및 관리하며, 이를 통해 업무 해결을 수행하는 비즈니스 룰 관리 시스템(BRMS)에 있어서, 업무에 대한 룰 처리 요청이 입력되는 룰 처리 요청부;와 상기 룰을 개발하고 관리하며 생성한 룰을 필요에 따라 특정의 룰 템플릿으로 생성하는 룰 생성 및 관리부;와 생성된 상기 룰이 저장되어 있는 룰 저장소;와 상기 룰을 이용하여 처리된 결과값이 저장되어 있는 룰 캐시부; 및 상기 룰 처리 요청에 해당하는 업무를 처리할 수 있는 룰을 이용하여 업무 해결을 수행하는 룰 엔진;을 포함하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템이 제공된다.

명세서

발명의 명칭: 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템 및 그 표현 방법

기술분야

- [1] 본 발명은 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템 및 그 방법에 관한 것으로, 보다 상세하게는 서로 계층적으로 연관되어 있는 업무 요건(비즈니스 룰)에 대한 결정 방법을 수행하는 룰을 통해, 연관된 룰의 결과를 함수의 입력값으로 이용하여 최적의 업무 결과를 수행하도록 하는 시스템 및 그 방법에 대한 것이다.

배경기술

- [2] 종래의 업무 시스템 개발의 수단으로서, 업무 요건(비즈니스 룰)을 시스템 사양에 의존적인 특정 프로그램 언어를 통해 하드 코딩을 함으로써 시스템을 구축했다. 이러한 경우, 업무 요건 또는 시스템의 사양 변경이 발생했을 때, 프로그램의 소스 코드를 조정/수정하거나, 프로그램 언어를 바꾸어 재 작성하는 작업 공정이 발생한다.
- [3] 이러한 작업 공정이 빈번히 일어나면, 개발 비용이 증가되고, 생산성이나 품질의 저하, 납기의 연장으로 이어진다. 이러한 업무를 해결하기 위해서, 최근 기업들에서는 BRMS(BUSINESS RULE MANAGEMENT SYSTEM)의 수단을 도입하는 곳이 증가하고 있다.
- [4] BRMS는 정의하고자 하는 비즈니스 룰을 프로그램과 분리하여 관리하고, 룰 엔진에 의한 해석/실행 서비스를 통해 프로그램에 결과를 제공하는 것이다.
- [5] 이와 같이 BRMS는 비즈니스 상의 룰을 별도로 일괄 관리함으로써, 어플리케이션으로부터 독립시켜 업무 로직을 개발/관리/실행 가능하게 한다.
- [6] 대한민국공개특허 제10-2013-0055683호인 "비즈니스 룰의 편집 및 컴파일"에는 그래프 기반의 연산을 이용하여 비즈니스 룰의 변환 및 편집을 하는 기술이 언급되어 있다.
- [7] 그러나, 상기의 특허에도 기업의 다양한 업무규칙을 프로그램에서 분리해 체계적인 객체화를 통해 비즈니스 룰로 관리하고, 규칙의 History 관리 및 규칙 검증 등 다양한 관리 기능과 API를 통한 서비스를 제공하여 어플리케이션에 유연성을 제공함으로써 신속하고 유연한 비즈니스 운영이 가능하게 하는, 시스템 개발 및 유지보수 노력, 비용을 절감할 수 있는 특징은 기술되어 있지 않다.

발명의 상세한 설명

기술적 과제

- [8] 본 발명이 이루고자 하는 기술적 과제는, IT 비전문가 들이 기업의 업무 규칙을 엑셀과 유사한 사용자 인터페이스를 통해 Non-Programming 방식으로 쉽고

편리하게 표현할 수 있도록, 다양한 업무 요건의 규칙을 수용하는 테이블 형식, 디시전 테이블 형식, 분기 형식, 플로우 형식 등의 규칙(룰) 표현 템플릿을 제공하는 데 있다.

- [9] 또한, 계층구조로 세분화된 단위 업무 규칙 별 상호 구조적인 관계를 정립하여, 공통 업무 또는 단위 업무별로 필요한 업무 로직을 내포한 비즈니스 룰을 등록하고, 룰 작성 과정에서 발생하는 룰의 실행 흐름과 룰 간 호출관계를 그래픽하게 표현하여 제공함으로써 가독성을 높이고, 유지보수가 용이한 룰 저작 도구를 제공하는데 있다.

- [10] 본 발명의 다른 목적들은 이하의 실시 예에 대한 설명을 통해 쉽게 이해될 수 있을 것이다.

과제 해결 수단

- [11] 본 발명의 일측에 따르면, 업무에 대한 결정 방법을 수행하는 룰(rule)을 생성 및 관리하며, 이를 통해 업무 해결을 수행하는 비즈니스 룰 관리 시스템(BRMS)에 있어서, 업무에 대한 룰 처리 요청이 입력되는 룰 처리 요청부;와 상기 룰을 개발하고 관리하며 생성한 룰을 필요에 따라 특정의 룰 템플릿으로 생성하는 룰 생성 및 관리부;와 생성된 상기 룰이 저장되어 있는 룰 저장소;와 상기 룰의 내용 및 룰을 이용하여 처리된 결과값이 저장되는 룰 캐시부; 및 상기 룰 처리 요청에 해당하는 업무를 처리할 수 있는 룰을 실행하여 업무 해결을 수행하는 룰 엔진;을 포함하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템이 제공된다.
- [12] 여기서, 상기 각 룰 템플릿은 실행 영역과 조건 영역으로 구성되고 계층적인 연관 구조로 이루어져 있어, 상기 계층적인 구조에서 N 번째의 depth에 해당하는 룰 템플릿은 실행 영역 및 조건 영역에서 N+1번째 depth에 대한 룰 호출식을 통한 결과값과, 기타 다양한 산술/논리/문자/날짜 함수 및 연산자 등을 사용하는 표현식(룰 표현식)을 통해 조건 판단 및 반환 할 실행 처리 요청을 수행하는 것을 특징으로 할 수 있다.
- [13] 여기서, 상기 룰은 모듈화되어 있는 함수이며 각각의 룰은 연관된 업무 흐름에 따라 서로 계층적으로 연관되어 있으며, 하위 룰을 통한 결과값은 상위 룰의 조건 판단이나 결과값을 생성하기 위해 입력되며, 개별 룰은 다른 룰을 호출하거나 피호출될 수 있는 것을 특징으로 할 수 있다.
- [14] 여기서, 상기 룰 생성 및 관리부는 상기 업무를 계층적으로 표현하여, 상기 업무 수행에 필요한 룰을 해당 속성과 버전을 리스트를 통해 표현하여 사용자에게 의해 생성된 함수가 포함된 새로운 룰을 생성하거나 이미 생성된 룰을 편집하는 편집툴을 제공하며, 생성된 룰을 특정의 템플릿 형식으로 생성하는 것을 특징으로 할 수 있다.
- [15] 여기서, 상기 룰 엔진은, 상기 룰 처리 요청이 입력되는 경우에, 상기 룰 저장소에서 해당하는 룰을 검색하고, 상기 캐시부의 상기 결과값 중 필요한

결과값이 있는 경우에 이를 검색된 룰에 적용하여 상기 룰 처리 요청에 대한 결과값을 생성하는 것을 특징으로 할 수 있다.

- [16] 여기서, 상기 룰 템플릿은 엑셀 형식의 디시전(decision) 테이블 템플릿으로 구성되어 있어 필요에 따라서 상기 룰 템플릿 간에 내용을 카피하여 수정 편집이 가능하고, 엑셀 프로그램에서 작성한 내용을 상기 룰 템플릿 생성시에 이용할 수 있는 것을 특징으로 할 수 있다.
- [17] 여기서, 상기 디시전(decision) 테이블 템플릿은 조건 영역과 리턴 영역을 포함하고, 상기 조건 영역은 비교 팩터인 하나의 조건식(룰 표현식)과 상기 조건식의 결과와 비교할 일련의 비교값으로 조건 영역의 각 조건을 열로 구성하고, 상기 리턴 영역의 행에는 상기 조건식의 결과와 상기 비교값의 비교를 통해 각 조건이 모두 만족하는 값을 표현하는 것을 특징으로 할 수 있다.
- [18] 또한, 업무에 대한 결정 방법을 수행하는 룰(rule)을 생성 및 관리하며, 이를 통해 업무 해결을 수행하는 비즈니스 룰 관리 시스템(BRMS)을 이용한 방법에 있어서, a) 업무에 대한 룰 처리 요청이 입력되는 단계;와 b) 상기 룰 중에서 상기 룰 처리 요청을 수행하는 업무 처리 룰을 선택하는 단계;와 c) 룰 저장소에 저장되어 있는 상기 룰들간의 연관도연관 관계를 분석하여 상기 업무 처리 룰과 연관된 룰을 탐색하는 단계; 및 d) 룰 캐시부에 저장된 상기 업무 처리 룰과 연관된 룰의 결과값을 이용하여 상기 업무 처리 룰을 통해 상기 룰 처리 요청에 대한 결과를 생성하는 단계;를 포함하되, 상기 룰은 필요에 따라 특정의 형식을 가지는 룰 템플릿으로 구성되어 있는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 방법이 제공된다.
- [19] 여기서, 상기 업무 처리 룰 및 상기 업무 처리 룰과 연관된 룰은 각각 함수로 구성되며, 업무 흐름에 따라 서로 계층적인 연관 구조를 가지고 있으며, 개별 룰은 다른 룰을 호출하거나 피호출될 수 있으며, 각각 하위 계층의 룰의 결과값을 함수의 옵션값으로 이용하는 것을 특징으로 할 수 있다.
- [20] 여기서, 상기 룰 템플릿은 실행 영역과 조건 영역으로 구성되어 있는 계층적인 연관 구조로 이루어져 있어, 상기 계층적인 구조에서 N 번째에 depth에 해당하는 룰 템플릿은 N+1번째의 depth에 해당하는 룰 템플릿의 결과값을 상기 실행 영역의 함수의 입력값 또는 상기 조건 영역의 조건값으로 사용하여 상기 룰 처리 요청을 수행하는 것을 특징으로 할 수 있다.
- [21] 여기서, 상기 룰 템플릿 중의 일부는 엑셀 형식의 디시전(decision) 테이블 템플릿으로 구성되어 있어 필요에 따라서 상기 룰 템플릿 간에 내용을 복사하여 수정 편집이 가능하고, 엑셀 프로그램에서 작성한 내용을 상기 룰 템플릿 생성시에 이용할 수 있는 것을 특징으로 할 수 있다.
- [22] 여기서, 상기 디시전(decision) 테이블 템플릿은 조건 영역과 리턴 영역을 포함하고, 상기 조건 영역은 비교 팩터인 하나의 조건식(룰 표현식)과 상기 조건식의 결과와 비교할 일련의 비교값으로 조건 영역의 각 조건을 열로 구성하고, 상기 리턴 영역의 행에는 상기 조건식의 결과와 상기 비교값의 비교를

통해 각 조건이 모두 만족하는 값을 표현하는 것을 특징으로 할 수 있다.

발명의 효과

- [23] 본 발명은 기업의 다양한 업무규칙을 프로그램에서 분리해 체계적인 객체화를 통해 비즈니스 룰로 관리하고, 규칙의 History 관리 및 규칙 검증 등 다양한 관리 기능과 API를 통한 서비스를 제공하여 어플리케이션에 유연성을 제공함으로써 신속하고 유연한 비즈니스 운영이 가능하게 하고, 시스템 개발 및 유지보수 노력, 비용을 절감할 수 있는 효과가 있다.
- [24] 또한, 본 발명은 빠른 처리성능을 제공하기 위하여 룰 저장소에 저장된 룰 내용과 중간 단계의 결과값을 메모리에 Caching하여 수행 시 DB를 접근하지 않고 메모리에서 바로 처리할 수 있도록 하여 업무 결정 속도를 높일 수 있는 효과가 있다.
- [25] 또한, 본 발명은 유사 규칙 작성 시 간단한 조작만으로 새로운 규칙으로 반영할 수 있으며, 다양한 검색 및 일괄 수정 기능 등을 제공하여 방대한 분량의 규칙을 편리하게 관리할 수 있는 효과가 있다.

도면의 간단한 설명

- [26] 도1은 InnoRules BRMS(Business Rule Management System)의 구성도이다.
- [27] 도2는 룰 빌더의 주요 모듈을 나타낸 도면이다.
- [28] 도3은 InnoRules Server 시스템 구성의 한 예를 나타낸 도면이다.
- [29] 도4는 Table 룰을 나타낸 도면이다.
- [30] 도5는 Flow 룰을 나타낸 도면이다.
- [31] 도6은 가변 룰을 나타낸 도면이다.
- [32] 도7은 루프 룰을 나타낸 도면이다.
- [33] 도8은 Decision Table 룰을 나타낸 도면이다.
- [34] 도9는 Data 룰을 나타낸 도면이다.
- [35] 도10은 CallBack 룰을 나타낸 도면이다.
- [36] 도11은 조건의 우선순위를 설명하기 위한 도면이다.
- [37] 도12는 처리단계의 공통 Component를 화를 설명하기 위한 도면이다.
- [38] 도13는 본 발명의 일실시예로 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템을 통한 룰 구현 및 서비스 과정을 도시한 도면이다.
- [39] 도14는 본 발명의 일실시예로 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템을 통한 룰 기반 업무 운영 과정을 도시한 도면이다.
- [40] 도15는 본 발명의 일실시예로 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템의 구성을 나타낸 도면이다.
- [41] 도16은 본 발명의 일실시예로 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템의 동작을 나타낸 순서도이다.
- [42] 도17은 본 발명의 일 실시예로 룰 개발 및 편집환경의 화면 구성을 도시한 도면이다.

[43] 도18은 본 발명의 일 실시예로 계층적인 룰 연관관계를 이용하여 업무에 대한 룰 처리 요청을 수행하는 것을 나타낸 도면이다.

발명의 실시를 위한 최선의 형태

[44] 본 발명은 다양한 변환을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세한 설명에 상세하게 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변환, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.

[45] 본 발명에서 사용한 용어는 단지 특정 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다.

[46] 이하, 본 발명의 실시예를 첨부한 도면들을 참조하여 상세히 설명하기로 한다.

[47]

[48] 도1은 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템을 설명하기 위한 개념도이다.

[49] 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템은 다음과 같은 구성으로 비즈니스 룰 관리를 수행한다.

[50] 우선 통합관리 도구로서 룰을 개발하고 편집 등 종합관리를 수행하는 Rule Builder와 룰을 수행하는 Rule Engine과 개발 및 편집된 룰을 저장하는 Rule Repository(룰 저장소)로 구성되어 있다.

[51] Rule Builder 는 client 에서 룰을 개발하기 위한 구성이고 server 의 Rule Builder server 를 통해 개발된 룰 및 편집된 룰을 Rule Repository에 저장한다.

[52] Rule Builder Server 는 Rule 개발 처리를 담당하며 Rule Builder의 요청을 받아 룰 정보 및 내용을 룰 DB에 조회/변경하는 모듈이다.

[53] Rule Engine은 application의 요청을 받아 룰 서비스를 수행하는 모듈이다.

[54] Rule Repository는 룰 정보 및 내용, 사용자, 항목 등 룰과 관련된 모든 데이터를 관리하는 룰 저장소이다. Rule Repository는 서버에 포함되어 있을 수 있다.

[55] Rule Server 는 TCP 를 통한 Rule service 요청을 Rule Engine 에 전달한다.

[56] Web-server Adapter는 Web Service 요청을 Rule Engine을 전달하는 모듈이다.

[57] Rule Engine Manager 는 Rule Engine과 Rule Builder Server의 상태를 모니터링하고 환경옵션을 관리하는 모듈이다.

[58]

[59] 도2는 룰 빌더의 주요 모듈을 나타낸 도면이다.

[60] Permissions Module

[61] - User Permission: 사용자는 관리자와 일반 사용자로 구성된다.

[62] -- 관리자: 사용자 관리와 최상위 업무 관리, 일반 사용자 권한 부여는 관리자 권한으로만 수행할 수 있다.

- [63] -- 일반 사용자: 부여된 권한에 따라 항목 관리와, 룰 작성 및 관리를 수행할 수 있다.
- [64] - Business Permission: 각 업무 단위 별로 일반 사용자에게 권한을 부여한다.
- [65]
- [66] External Interface Module
- [67] - DataBase: DB 조회룰에서 사용할 기간계 DB에 대한 연결 정보를 구성한다.
- [68] - SAP R/3: RFC(Remote Function Call) 룰에서 사용할 SAP 서버에 대한 연결 정보를 구성한다.
- [69]
- [70] Rule Editor Module
- [71] - 테이블 룰: 조건 영역의 각 행의 룰 표현식을 수행하고 연산하여 결과가 참(True)인 조건 영역 행에 대한 실행 영역을 수행하고 결과를 반환하는 룰이다.
- [72] - 루프룰: 반복 수행 로직을 수행하는 룰이다.
- [73] - Decision Table 룰: 조건 영역 각 열 단위에 등록된 룰 표현식을 수행한 결과와 해당 열에 등록된 비교값과의 비교를 통해 만족하는 각 행에 등록되어 있는 결과값을 반환하는 룰이다.
- [74] - DB 조회룰: 기간계 Legacy DB를 조회하기 위하여 Dynamic SQL를 사용하는 룰이다.
- [75] - 가변룰: 파라미터에 따라 해당 이름을 갖는 룰로 분기(실행 후 결과값 참조)하는 룰이다.
- [76] - Call Back 룰: 인자 전달을 통해 일반 프로그램에 있는 모듈을 호출하여 실행하고 그 결과를 돌려받는 룰이다.
- [77] - RFC 룰: SAP 서버와 연동하는 룰이다.
- [78] - Flow 룰: Flow Chart 형식으로 업무 흐름을 표현하는 룰이다.
- [79] - 데이터 룰: Legacy DB Table의 특정 칼럼을 조건 영역의 비교값으로 사용하고, 또 다른 특정 칼럼을 결과값으로 이용하는 룰이다. 기타 동작 메커니즘은 Decision Table 룰과 같다.
- [80] - Syntax Checker: 룰 저장 시 문법 오류를 검사한다.
- [81]
- [82] Rule Management Module
- [83] - Business Copy: 업무 단위의 복사 기능으로 기존 룰 그룹을 일괄 복사하여 신속한 룰 개발을 가능하게 한다.
- [84] - Relational Map: 선택된 룰을 호출하는 상위 룰과 선택된 룰에서 호출하는 하위 룰들간의 관계를 표시하여 룰들의 호출 관계를 손쉽게 파악할 수 있다.
- [85] - Rule Versioning: 일자 별로 업무가 변경될 시, 룰의 버전을 관리하여 룰 호출 시 일자에 맞는 버전이 자동 실행된다.
- [86] - Rule Reporter: 다양한 단계의 룰 리포트를 자동 생성한다.
- [87]

[88] Rule Tester Module

[89] - Rule Tester: 룰 생성 후 정확성 및 타당성을 검증하는 것으로 룰에서 사용되는 항목 값들에 여러 경우의 다양한 데이터를 사용자가 입력하여 룰의 수행 결과를 테스트 및 Simulation이 가능하다.

[90] - Rule Tracer: 하위 룰을 포함하여 룰의 실행 결과를 단계별로 보여주어 룰의 실행경로를 파악할 수 있다.

[91]

[92] Rule Deployment Module

[93] - Rule Deploy: 개발 시스템에서 운용 시스템으로 룰을 이관하는 경우 동시 이관 대상 룰, 룰의 에러 여부, 담당자 승인 여부 등을 검증한다.

[94] - Consistency Checker: 개발 시스템에서 운용 시스템으로 이관되는 룰과, 운용계의 기존 룰 들간의 상호 모순성을 검증한다.

[95] - Rule History: 개발 시스템에서 운용 시스템으로 룰 이관 시, 이관 히스토리를 관리한다.

[96]

[97] 도 3은 InnoRules Server 시스템 구성의 한 예를 나타낸 도면이다.

[98]

[99] 도 4는 Table 룰을 나타낸 도면이다.

[100]

[101] 도 5는 Flow 룰을 나타낸 도면이다.

[102]

[103] 도 6은 가변 룰을 나타낸 도면이다.

[104]

[105] 도 7은 루프 룰을 나타낸 도면이다.

[106]

[107] 도 8은 Decision Table 룰을 나타낸 도면이다.

[108] Decision Table 룰은 DATA 성격이 강한 업무에서 주로 사용하는 룰 타입으로 한정된 동일 조건으로 구성된 룰로써 하위 룰을 포함한 룰인 경우 사용하며 정의된 룰의 내용은 메모리에 존재하므로 업무에 자주 이용되는 룰인 경우에 사용한다.

[109] Decision Table 룰은 조건영역과 리턴영역으로 구성되어 조건영역의 Rule(여기서 룰은 {}로 표현된다)의 수행 결과에 따라 리턴 영역의 값(또는 룰 표현식 실행 결과)가 결과로서 상위 룰에 반환된다.

[110] Decision Table 룰은 비교 팩터인 하나의 조건식(룰 표현식)과 이 조건식의 결과와 비교할 일련의 비교값으로 조건 영역의 각 조건(열)을 구성하고, 각 조건의 조건식의 결과와 비교값의 비교를 통해 각 조건이 모두 만족하는 행의 리턴이 반환되는 룰 템플릿이다.

[111] 각 조건은 실행 판단식을 통해 해당 조건의 조건식의 실행 여부를 지정할 수

있다. 실행 판단식이 기술되지 않았거나 실행 결과가 “참(T)”일 경우 해당 조건은 리턴 영역의 룰 표현식 실행 판단에 사용이 되고, 그렇지 않을 경우 제외된다.

[112] 각 조건(열) 간의 관계는 “AND”이다.

[113] 조건을 모두 만족하는 행이 없을 경우, 룰 리턴은 디폴트 리턴 영역에 지정된 디폴트 값(또는 룰 표현식 실행 결과)를 갖게 되고, 결과가 있을 경우에는 무시된다.

[114] Decision Table 룰의 지정 가능한 리턴 형식은 다음과 같다

[115] 1. 단일값 형식

[116] 단일값 형식으로 지정한 경우, 만족하는 첫 번째 조건 행에 대응하는 리턴 영역의 동일 행에 있는 룰 표현식이 실행되고, 룰 리턴은 단일값 형식을 갖는다. 항상 만족하는 1개의 조건이 있어야 하며, 룰 리턴 또한 항상 1개의 결과 값(행)을 가져야 한다.

[117] 2. 다중값 형식

[118] 다중값 형식으로 지정하고 모든 만족 조건을 실행을 체크하는 경우, 만족하는 모든 조건 행이 대응하는 리턴 영역의 동일 행에 있는 룰 표현식이 실행되고, 룰 리턴은 다중값 형식을 갖는다. 다중값 형식으로 지정하고 모든 만족 조건 실행을 체크하지 않은 경우, 만족하는 첫 번째 조건 행에 대응하는 리턴 영역의 동일 행에 있는 룰 표현식이 실행되고 룰 리턴은 다중값 형식을 갖는다. 다중값 형식은 만족하는 조건이 없어 룰 리턴이 값을 가지지 않을 수 있고, 다수의 만족하는 조건이 있어 복수 개의 결과 값(행)을 가질 수 있다.

[119] 여기서, 리턴 항목의 데이터 타입은 문자형과 숫자형, 논리형이 사용 가능하다.

[120]

[121] 룰 표현식의 사용위치는 다음과 같다.

[122] 1. 조건 영역의 실행 판단식

[123] 룰 표현식이 갖는 규칙을 준수하는 모든 룰 표현식 사용이 가능하다.

[124] 룰 표현식 결과는 기본형 단일값 형식으로서, 논리형 값인 "T" 또는 "F" 중 하나의 값을 가져야 한다.

[125] 2. 조건 영역의 조건식

[126] 룰 표현식이 갖는 규칙을 준수하는 모든 룰 표현식 사용이 가능하다.

[127] 룰 표현식 결과는 기본형 단일값 형식이어야 한다.

[128] 3. 조건 영역의 비교값

[129] 숫자형 또는 문자형 또는 논리형인 하나의 상수 값만 사용이 가능하다.

[130] 4. 리턴 영역

[131] 룰 표현식이 갖는 규칙을 준수하는 모든 룰 표현식 사용이 가능하다.

[132] 5. 디폴트 리턴 영역

[133] 조건을 모두 만족하는 행이 없을 경우, 룰 리턴은 디폴트 리턴 영역에 지정된 디폴트 값이나 디폴트 리턴 영역에 기술한 룰 표현식의 실행 결과를 갖게 된다.

[134] 디폴트 룰 표현식에는 리턴 영역의 룰 표현식이 갖는 규칙을 준수하는 모든 룰

표현식 사용이 가능하다.

[135]

[136] 도 9는 Data 룰을 나타낸 도면이다.

[137]

[138] 도 10은 CallBack 룰을 나타낸 도면이다.

[139]

[140] 도 11은 Table 룰로서, 조건의 우선순위를 설명하기 위한 도면이다.

[141] Decision 및 Table 룰 처리시, 조건 열은 왼쪽에서 오른쪽, 위에서 아래로 evaluation을 수행하고, 각 열 간의 AND 연산 결과가 “T”인 조건에 대응하는 행에 대해 실행 영역의 값(룰 호출식과 함수, 연산자 등으로 구성된 룰 표현식일 경우 해당 룰 표현식을 수행한 결과, 이러한 조건 및 실행 영역 셀 내 룰 표현식 사용은 N-Depth까지 가능하여 무제한)을 반환한다. 조건 영역의 각 조건 열이나 실행 영역의 각 열은 개수 제한이 없고, Rule Builder를 통해 가감이 가능하다.

[142]

[143] 도 12는 처리단계의 공통 Component를 화를 설명하기 위한 도면이다.

[144] 처리 흐름을 중요시하는 업무인 경우, 일반적으로 프로그램 코딩으로 구현한 경우, 업무 흐름을 중심으로 코딩하게 된다. 이런 업무의 유지보수는 주로 프로세스가 중간에 삽입되거나, 업무 단위 별로 흐름이 달라서 중복 코딩작업을 하는 경우가 많다. 이런 업무를 룰로 구성할 경우에는 처리 흐름 단계별로 종속성을 제거한 공통 컴포넌트 룰(공통룰)을 구성하고, 컴포넌트 룰로 업무 흐름 룰을 구성한다.

[145]

[146] 도13은 본 발명의 일실시예로 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템을 통한 룰 구현 및 서비스 과정을 도시한 도면이다.

[147] 본 발명에서의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)은 다양한 업무 규칙을 중복 배제 및 체계적인 객체화를 통해 룰로 표현하고, BRMS의 규칙 검증 및 History 관리, API를 통한 서비스 등 다양한 기능을 통해 어플리케이션의 유연성을 실현할 수 있다.

[148] 룰(비즈니스 룰)은 업무지침, 업무 경험, 기존 프로그램, 기존 DB를 참조하여 생성하고, 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)의 룰 통합관리 도구를 이용하여 룰을 개발 및 테스트하여 룰을 생성하고 이를 관리 및 운영한다.

[149] 생성된 룰은 룰 저장소(즉, 룰 저장부(120))에 저장된다. 이때, 룰은 업무 연관 관계에 따라 계층적인 구조의 연관 관계(객체화, 공통화, 집중화)를 가지며 저장된다.

[150]

[151] 이후 해당 룰 처리 요청이 입력되면 룰 엔진이 이를 실행하여 결과를 반환한다.

[152]

- [153] 도14는 본 발명의 일실시예로 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템을 통한 룰 기반 업무 운영 과정을 도시한 도면이다.
- [154] 업무 분석을 통해 룰을 정의하고 테스트 및 오류 추적을 통해 룰을 개발한다.
- [155] 이후 개발한 룰을 등록하거나 기존 룰에 복사 또는 조립하여 새로운 룰을 룰 저장소에 등록 및 업데이트 할 수 있다.
- [156] 이때 룰은 다양한 룰 템플릿 형식으로 등록될 수 있다.
- [157] 룰 저장소에 등록된 룰은 룰 검색 및 룰 이력 관리가 가능하며, 룰 연관 관계 분석에 의해 업무 요건 변화와 연관된 룰 범위 파악이 가능하여 빠르게 업무 변화에 대응이 가능하다. 본 발명에서의 룰은 엑셀 프로그램과 호환 가능하여 필요에 따라서 카피하여 수정 편집을 할 수 있다. 또한, 엑셀 프로그램에서 작성한 내용을 룰 생성시에 이용할 수도 있다.
- [158] 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)은 룰 개발, 운영, 처리과정을 다양한 룰 리포트로 생성하여 화면상에 표시할 수 있다.
- [159]
- [160] 도15는 본 발명의 일실시예로 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템의 구성을 나타낸 도면이다.
- [161] 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)은 룰 생성 및 관리부(110), 룰 저장부(120), 룰 처리 요청부(130), 룰 엔진(140), 룰 캐시부(150)를 포함한다.
- [162] 룰 생성 및 관리부(110)는 업무 처리에 필요한 룰을 개발하고 편집, 관리하며 생성한 룰을 필요에 따라 특정의 룰 템플릿으로 생성한다.
- [163] 이를 위해서 룰 생성 및 관리부(110)는 디스플레이 장치에 해당 룰 개발 및 관리 편집 과정을 GUI 형식으로 표현할 수 있다.
- [164] 또한, 룰 생성 및 관리부(110)는 업무 수행에 필요한 룰의 해당 속성과 버전을 리스트를 통해 표현할 수 있고, 사용자에게 의해 생성된 함수가 포함된 새로운 룰을 생성하거나 이미 생성된 룰을 수정 및 편집할 수 있다.
- [165] 룰 저장부(120)는 룰 생성 및 관리부(110)를 통해 생성된 룰이 저장되어 있다.
- [166] 룰 처리 요청부(130)는 업무에 대한 룰 처리 요청을 받아 룰 엔진에 전달하고, 룰 엔진의 실행 결과를 반환하는 다양한 프로그램 언어에 대응하는 API(Application Programming Interface) 이다. 예를 들어, 자바, C/C++, COBOL 등이 있다.
- [167] 룰 엔진(140)은 룰 처리 요청에 해당하는 업무를 처리할 수 있는 룰을 이용하여 업무 해결을 수행한다.
- [168] 룰 캐시부(150)는 룰을 이용하여 처리된 결과값이 저장되어 있다.
- [169] 룰 엔진(140)은 룰 처리 요청에 해당하는 업무를 처리하는 경우에 룰 캐시부(150)에 저장되어 있는 관련 룰의 결과값을 이용하여 해당 룰 처리 요청에 대한 결과를 생성할 수 있다.

발명의 실시를 위한 형태

- [170] 도16은 본 발명의 일실시예로 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템의 동작을 나타낸 순서도이다.
- [171] S100 단계는 룰 처리 요청이 입력되는 단계이다.
- [172] 본 발명의 사용자는 해결하고자 하는 업무가 발생하면, 룰 처리 요청을 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)에 요청한다.
- [173] 여기서, 룰 처리 요청은 사용자가 디스플레이 화면에서 업무 프로그램을 통해 해당 업무를 처리하는 시작 단계를 포함한다.
- [174] S110 단계는 룰 중에서 상기 룰 처리 요청을 수행하는 업무 처리 룰을 선택하는 단계이다.
- [175] 사용자에게 의해서 업무에 대한 룰 처리 요청이 입력되면 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)은 해당 업무를 수행할 수 있는 룰을 선택하고, 연관되어 실행할 룰을 다음 S120 단계를 통해 탐색한다.
- [176] 즉, 해당 업무를 수행할 수 있는 룰을 사용자가 원하는 형식의 룰 템플릿으로 화면에 표시한다.
- [177] S120 단계는 룰 저장소에 저장되어 있는 상기 룰들간의 연관 관계를 분석하여 상기 업무 처리 룰과 연관된 룰을 탐색하는 단계이다.
- [178] 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)은 사용자가 원하는 업무를 수행할 수 있는 룰 템플릿의 내용을 분석하여 해당 룰 템플릿의 함수에 포함되어 있는 연관된 룰을 탐색하여 알아 내고 S130 단계를 거쳐 캐시를 이용하거나, 추가 실행을 하여 사용자가 요구한 룰의 결과를 생성하고 반환한다.
- [179] S130 단계는 룰 캐시부에 저장된 상기 업무 처리 룰과 연관된 룰의 결과값을 이용하여 상기 업무 처리 룰을 통해 상기 룰 처리 요청에 대한 결과를 생성하는 단계이다.
- [180] 즉, 사용자가 원하는 업무를 수행할 수 있는 룰 템플릿의 함수를 구성하는 연관된 룰의 결과를 룰 템플릿의 함수에 대치하여 사용자가 원하는 업무의 처리 결과를 생성한다.
- [181] 여기서, 업무 처리 룰 및 상기 업무 처리 룰과 연관된 룰은 각각 함수로 구성되며, 업무 흐름에 따라 서로 계층적인 연관 구조를 가지고 있으며, 개별 룰은 다른 룰을 호출하거나 피호출될 수 있으며(즉, 룰은 다른 룰을 함수에 포함할 수 있으며, 다른 룰의 함수에 포함될 수 있다), 각각 하위 계층의 룰의 결과값을 함수의 옵션값으로 이용할 수 있다.
- [182] 본 발명에서의 룰 템플릿은 실행 영역과 조건 영역으로 구성되어 있는 계층적인 연관 구조로 이루어져 있다.
- [183] 실행영역은 해당 룰 템플릿이 수행하는 업무에 대한 함수를 의미하고, 조건 영역은 해당 함수를 구성하는 연관된 룰의 결과값을 의미한다.

- [184] 계층적인 구조를 이루는 룰 템플릿에서,은 N 번째의 depth 에 해당하는 룰 템플릿은 N+1번째의 depth 에 해당하는 룰 템플릿의 결과값을 상기 실행 영역의 함수의 입력값 또는 상기 조건 영역의 조건값으로 사용하여 룰 처리 요청을 수행한다.
- [185] 이에 대한 상세한 설명은 도18에서 설명하기로 한다.
- [186]
- [187] 도17은 본 발명의 일 실시예로 룰 개발 및 편집환경의 화면 구성을 도시한 도면이다.
- [188] 1번 화면은 계층적인 업무 구조를 표현한 화면 구성이다.
- [189] 1번 화면에 도시되어 있는 바와 같이 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)은 업무를 계층적, 체계적으로 표시하여 업무간에 상하 종속적인 계층 구조를 쉽게 파악하고 정의할 수 있도록 표시한다.
- [190] 2번 화면은 룰 리스트를 표시하는 화면이다.
- [191] 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)은 룰 컴포넌트를 구성하는 최소 단위의 룰을 속성과 버전을 표시하여 리스트로 표시할 수 있다.
- [192] 3번 화면은 룰 생성 및 편집을 위한 화면이다.
- [193] 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)은 엑셀 형태와 같은 직관적이고 친숙한 사용환경을 제공하여 사용자가 보다 편리하고 간편하게 룰을 생성하고 편집할 수 있도록 한다.
- [194] 4번 화면은 룰 템플릿을 선택하는 화면이다.
- [195] 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)은 생성 또는 편집한 룰을 다양한 형태의 룰 템플릿으로 변환하여 사용자가 직관적으로 해당 룰을 파악할 수 있도록 할 수 있다.
- [196] 여기서, 룰 템플릿은 테이블 룰, 가변 룰, DB 조회룰, Call Back 룰, 루프룰, RFC 룰, Flow 룰 등이 될 수 있다.
- [197]
- [198] 도18은 본 발명의 일 실시예로 계층적인 룰 연관관계를 이용하여 업무에 대한 룰 처리 요청을 수행하는 것을 나타낸 도면이다.
- [199] 사용자가 할인율 계산에 대한 업무를 요청하는 경우에 본 발명의 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템(100)은 할인율을 처리하는 룰을 다수의 룰 템플릿으로 작성하고, 각 룰 템플릿의 조건 영역이나 실행 영역 셀에 룰 표현식을 기술함으로써 구현한다. 도해에서 사용된 중괄호({ }) 표기가 다른 룰을 호출하는 표현식이다.
- [200] 여기서, 룰 템플릿 10은 할인율에 대한 계산을 처리하는 룰이 Decision Table 룰 형태로 표시되어 있는 룰 템플릿이고, 룰 템플릿 20은 룰 템플릿 10의 1번째 depth 위치에 있는 룰 템플릿이다.
- [201] 즉, 룰 템플릿 20은 룰 템플릿 10의 함수를 구성하는 {요일추가할인율}에 대한

룰이다.

- [202] 여기서, 할인율을 구하기 위해서는 조건 영역의 {VIP여부}에 대한 룰과 {전월실적체크}에 대한 룰이 필요하다.
- [203] {VIP여부}에 대한 룰은 {고객정보체크}에 대한 룰이 필요하고, {전월실적체크}에 대한 룰은 {실적집계}에 대한 룰이 필요하다.
- [204] 여기서, {고객정보체크}에 대한 룰과 {실적집계}에 대한 룰은 룰 템플릿 10의 2번째 depth를 가진 연관 룰이 된다.
- [205] 룰 템플릿 30은 {서비스적용여부}에 대한 룰이 Table 룰 형태로 표시되어 있는 룰 템플릿이고, 룰 템플릿 40은 {전월실적등급}에 대한 룰이 Flow 룰 형태로 표현되어 있는 Flow 룰이다.
- [206] 상기에서는 본 발명의 실시예를 참조하여 설명하였지만, 해당 기술 분야에서 통상의 지식을 가진 자라면 하기의 특허 청구의 범위에 기재된 본 발명의 사상 및 영역으로부터 벗어나지 않는 범위 내에서 본 발명을 다양하게 수정 및 변경시킬 수 있음을 이해할 수 있을 것이다.
- [207] 본 발명은 또한 컴퓨터로 읽을 수 있는 기록 매체에 컴퓨터가 읽을 수 있는 코드로서 구현하는 것이 가능하다. 컴퓨터가 읽을 수 있는 기록 매체는 컴퓨터 장치에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록 장치를 포함한다. 컴퓨터가 읽을 수 있는 기록 매체의 예로는 롬(ROM), 램(RAM), CD-ROM, 자기 테이프, 플로피 디스크, 광 데이터 저장장치 등이 있으며, 또한 캐리어 웨이브(인터넷을 통한 전송)의 형태로 구현되는 것도 포함한다. 또한, 컴퓨터가 읽을 수 있는 기록 매체는 유무선 통신망으로 연결된 컴퓨터 장치에 분산되어 분산 방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다.

산업상 이용가능성

- [208] 본 발명은 IT 비전문가들이 기업의 업무 규칙을 엑셀과 유사한 사용자 인터페이스를 통해 Non-Programming 방식으로 쉽고 편리하게 표현할 수 있는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템을 제공할 수 있는 산업상 이용 가능성이 있다.

청구범위

- [청구항 1] 업무에 대한 결정 방법을 수행하는 룰(rule)을 생성 및 관리하며, 이를 통해 업무 해결을 수행하는 비즈니스 룰 관리 시스템(BRMS)에 있어서,
 업무에 대한 룰 처리 요청이 입력되는 룰 처리 요청부;
 상기 룰을 개발하고 관리하며 생성한 룰을 필요에 따라 특정의 룰 템플릿으로 생성하는 룰 생성 및 관리부;
 생성된 상기 룰이 저장되어 있는 룰 저장소;
 상기 룰을 이용하여 처리된 결과값이 저장되어 있는 룰 캐시부; 및
 상기 룰 처리 요청에 해당하는 업무를 처리할 수 있는 룰을 이용하여 업무 해결을 수행하는 룰 엔진;
 을 포함하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템.
- [청구항 2] 업무에 대한 결정 방법을 수행하는 룰(rule)을 생성 및 관리하며, 이를 통해 업무 해결을 수행하는 비즈니스 룰 관리 시스템(BRMS)에 있어서,
 업무에 대한 룰 처리 요청이 입력되는 룰 처리 요청부;
 상기 룰을 개발하고 관리하며 생성한 룰을 필요에 따라 특정의 룰 템플릿으로 생성하는 룰 생성 및 관리부;
 생성된 상기 룰이 저장되어 있는 룰 저장소;
 상기 룰을 이용하여 처리된 결과값이 저장되어 있는 룰 캐시부; 및
 상기 룰 처리 요청에 해당하는 업무를 처리할 수 있는 룰을 이용하여 업무 해결을 수행하는 룰 엔진;
 을 포함하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템.
- [청구항 3] 제1항에 있어서,
 상기 룰은 모듈화되어 있는 함수이며 각각의 룰은 연관된 업무 흐름에 따라 서로 계층적으로 연관되어 있으며, 하위 룰을 통한 결과값은 상위 룰에 결과값을 생성하기 위해 입력되며, 개별 룰은 다른 룰을 호출하거나 피호출될 수 있는 것을 특징으로 하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템.
- [청구항 4] 제1항에 있어서,
 상기 룰 생성 및 관리부는
 상기 업무를 계층적으로 표현하여,
 상기 업무 수행에 필요한 룰의 해당 속성과 버전을 리스트를 통해 표현하여 사용자에게 의해 생성된 함수가 포함된 새로운 룰을 생성하거나 이미 생성된 룰을 편집하는 편집툴을 제공하며,

[청구항 5]

생성된 룰을 특정의 템플릿 형식으로 생성하는 것을 특징으로 하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템.
제1항에 있어서,

상기 룰 엔진은,

상기 룰 처리 요청이 입력되는 경우에, 상기 룰 저장소에서 해당하는 룰을 검색하고, 상기 캐시부의 상기 결과값 중 필요한 결과값이 있는 경우에 이를 검색된 룰에 적용하여 상기 룰 처리 요청에 대한 결과값을 생성하는 것을 특징으로 하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템.

[청구항 6]

제1항 내지 제5항 중 어느 하나의 항에 있어서,

상기 룰 템플릿 중의 일부는 엑셀 형식의 디시전(decision) 테이블 템플릿으로 구성되어 있어 필요에 따라서 상기 룰 템플릿 간에 내용을 카피하여 수정 편집이 가능하고, 엑셀 프로그램에서 작성한 내용을 상기 룰 템플릿 생성시에 이용할 수 있는 것을 특징으로 하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템.

[청구항 7]

제6항에 있어서,

상기 디시전(decision) 테이블 템플릿은 조건 영역과 리턴 영역을 포함하고, 상기 조건 영역은 비교 팩터인 하나의 조건식(룰 표현식)과 상기 조건식의 결과와 비교할 일련의 비교값으로 조건 영역의 각 조건을 열로 구성하고, 상기 리턴 영역의 행에는 상기 조건식의 결과와 상기 비교값의 비교를 통해 각 조건이 모두 만족하는 값을 표현하는 것을 특징으로 하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 시스템.

[청구항 8]

업무에 대한 결정 방법을 수행하는 룰(rule)을 생성 및 관리하며, 이를 통해 업무 해결을 수행하는 비즈니스 룰 관리 시스템(BRMS)을 이용한 방법에 있어서,

a) 업무에 대한 룰 처리 요청이 입력되는 단계;

b) 상기 룰 중에서 상기 룰 처리 요청을 수행하는 업무 처리 룰을 선택하는 단계;

c) 룰 저장소에 저장되어 있는 상기 룰들간의 연관 관계를 분석하여 상기 업무 처리 룰과 연관된 룰을 검색하는 단계; 및

d) 룰 캐시부에 저장된 상기 업무 처리 룰과 연관된 룰의 결과값을 이용하여 상기 업무 처리 룰을 통해 상기 룰 처리 요청에 대한 결과를 생성하는 단계;

를 포함하되,

상기 룰은 필요에 따라 특정의 형식을 가지는 룰 템플릿으로 구성되어 있는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리

방법.

[청구항 9]

제8항에 있어서,

상기 업무 처리 룰 및 상기 업무 처리 룰과 연관된 룰은 각각 함수로 구성되며, 업무 흐름에 따라 서로 계층적인 연관 구조를 가지고 있으며, 개별 룰은 다른 룰을 호출하거나 피호출될 수 있으며, 각각 하위 계층의 룰의 결과값을 함수의 옵션값으로 이용하는 것을 특징으로 하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 방법.

[청구항 10]

제8항에 있어서,

상기 룰 템플릿은 실행 영역과 조건 영역으로 구성되어 있는 계층적인 연관 구조로 이루어져 있어, 상기 계층적인 구조에서 N 번째의 depth 에 해당하는 룰 템플릿은 N+1 번째의 depth 에 해당하는 룰 템플릿의 결과값을 상기 실행 영역의 함수의 입력값 또는 상기 조건 영역의 조건값으로 사용하여 상기 룰 처리 요청을 수행하는 것을 특징으로 하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 방법.

[청구항 11]

제8항 내지 제10항 중 어느 하나의 항에 있어서,

상기 룰 템플릿 중의 일부는 엑셀 형식의 디시전(decision) 테이블 템플릿으로 구성되어 있어 필요에 따라서 상기 룰 템플릿 간에 내용을 카피하여 수정 편집이 가능하고, 엑셀 프로그램에서 작성한 내용을 상기 룰 템플릿 생성시에 이용할 수 있는 것을 특징으로 하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 방법.

[청구항 12]

제11항에 있어서,

상기 디시전(decision) 테이블 템플릿은 조건 영역과 리턴 영역을 포함하고, 상기 조건 영역은 비교 팩터인 하나의 조건식(룰 표현식)과 상기 조건식의 결과와 비교할 일련의 비교값으로 조건 영역의 각 조건을 열로 구성하고, 상기 리턴 영역의 행에는 상기 조건식의 결과와 상기 비교값의 비교를 통해 각 조건이 모두 만족하는 값을 표현하는 것을 특징으로 하는 계층적인 룰 구조를 가지고 있는 비즈니스 룰 관리 방법.

[도7]

항목		데이터
1	[납입기간]	[납입기간].ARRAV
2	[보험기간]	[보험기간].ARRAV
3	[보증코드]	[보증코드].ARRAV
4	[납입주기코드]	[납입주기코드].ARRAV
5	[가입보험금액]	[가입보험금액].ARRAV
6	[보증별평균나이]	[보증별평균나이].ARRAV

루프를 리턴

	^A B 결과코드	^A B 결과메세지	¹ ₂ 보험료	¹ ₂ 월초보험료	¹ ₂ 할인보험료	^A B 보증코드
리턴	{보증별보험료:[\$1_기준일]=[기준 일지],[기준일]=[기준일지]}					[보증코드]

[도8]

[illegible]

[도9]

물데이터 모델 변경 (보상)JB2B수사_전문인평가계수금액		데이터 모델 이름 (보상)JB2B수사_전문인평가계수금액		데이터 모델 ID CLPS0010R		연결 DB 정보		상품 서비스	
타이틀 이름		DATAMODE5		Data Model ID					
트칼럼명(선택) 칼럼 이름									
칼럼 형식 칼럼 이름									
문자형 CODE1 문자형 CODE2 문자형 CODE3 문자형 CODE4 문자형 CODE5									
키 Key									
1 순차형 RET1 시간당보수									
2 순차형 COED1 전문인평가계									
3 순차형 VF_DT 시작일자									
[DEVPART] (보상)전문인평가계수금액:2009-02-02									
☐ 문기를 ☒ 단행 ☐ 다중									
☐ 마지막 버전 열기 ☐ 허용값 이름 조회(힌트)									
데이터 조회 > 검색 개수 20 Memory Cache									
H₂ 시간당보수금액 H₃									
Physical RET1 VF_DT VF_DT									
Logical 시간당보수 전문인평가계 시작일자 종료일자									
수행편의성 [적용일자]									
조건식 [보상]전문인평가계구분코드									
1 100,000.01 2009020199991231									
2 80,000.02 2009020199991231									
3 60,000.03 2009020199991231									
리턴 H ₂ 시간당보수금액									
기본값									
인증 1999991231									

[도 10]

서비스 이름		sbsnDayTcFind	
프로그램에 전달할 파라미터 데이터			
 			
	파라미터	데이터	
1	시작일자	[이자계산시작일자]	
2	영업일수	[영업일수L]	
3	영업일여부	"V"	
프로그램에 전달받을 리턴 데이터			
 			
	A _B 영업일		
리턴			
데이터			

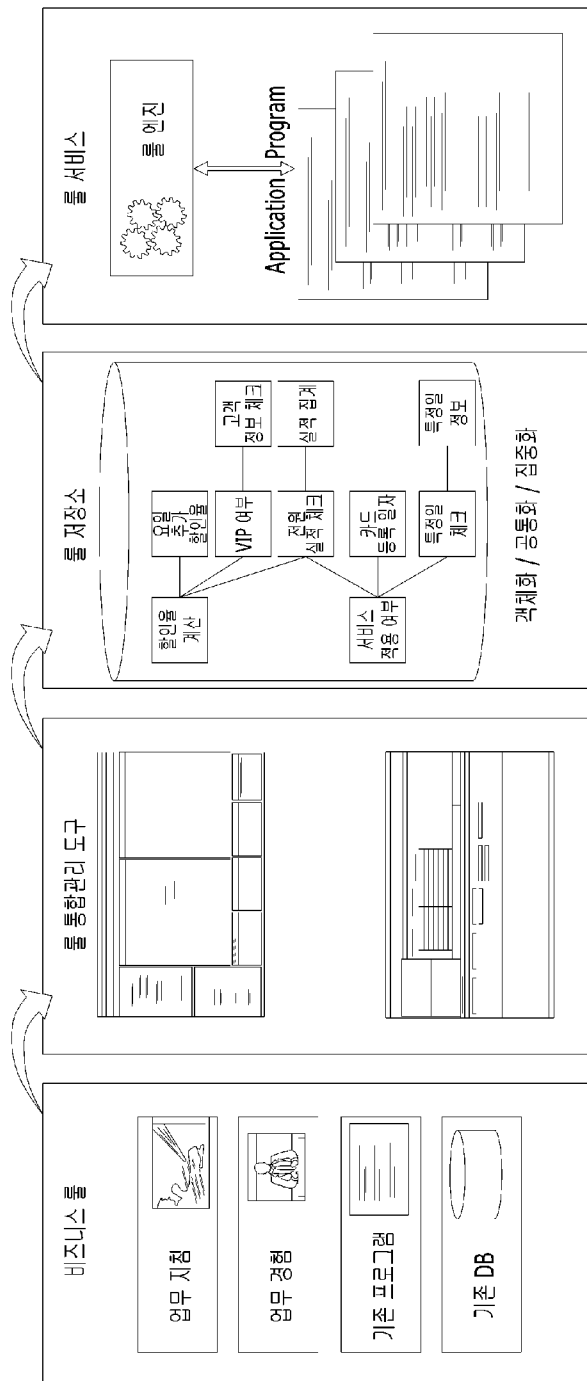
[표 11]

I ₂	보통로	조건 1	조건 2	조건 3	조건 4
206.100		[차량용도] = "1"	[치중구분] = "20"		
248.600			[치중구분] = "44"		
248.600			[치중구분] = "45"		
286.100			[치중구분] = "47"		
286.100			[치중구분] = "48"		
285.100			[치중구분] = "49"		
295.400			[치중구분] = "50"		
328.200			[치중구분] = "24"		
299.400			[치중구분] = "26"		
316.300			[치중구분] = "27"		
206.100		[차량용도] = "2"	[치중구분] = "20"		
248.600			[치중구분] = "44"		
248.600			[치중구분] = "45"		
286.100			[치중구분] = "47"		
286.100			[치중구분] = "48"		
285.100			[치중구분] = "49"		
295.400			[치중구분] = "50"		
328.200			[치중구분] = "24"		
299.400			[치중구분] = "26"		
316.300			[치중구분] = "27"		
Error("대인배상1 산출오류")					

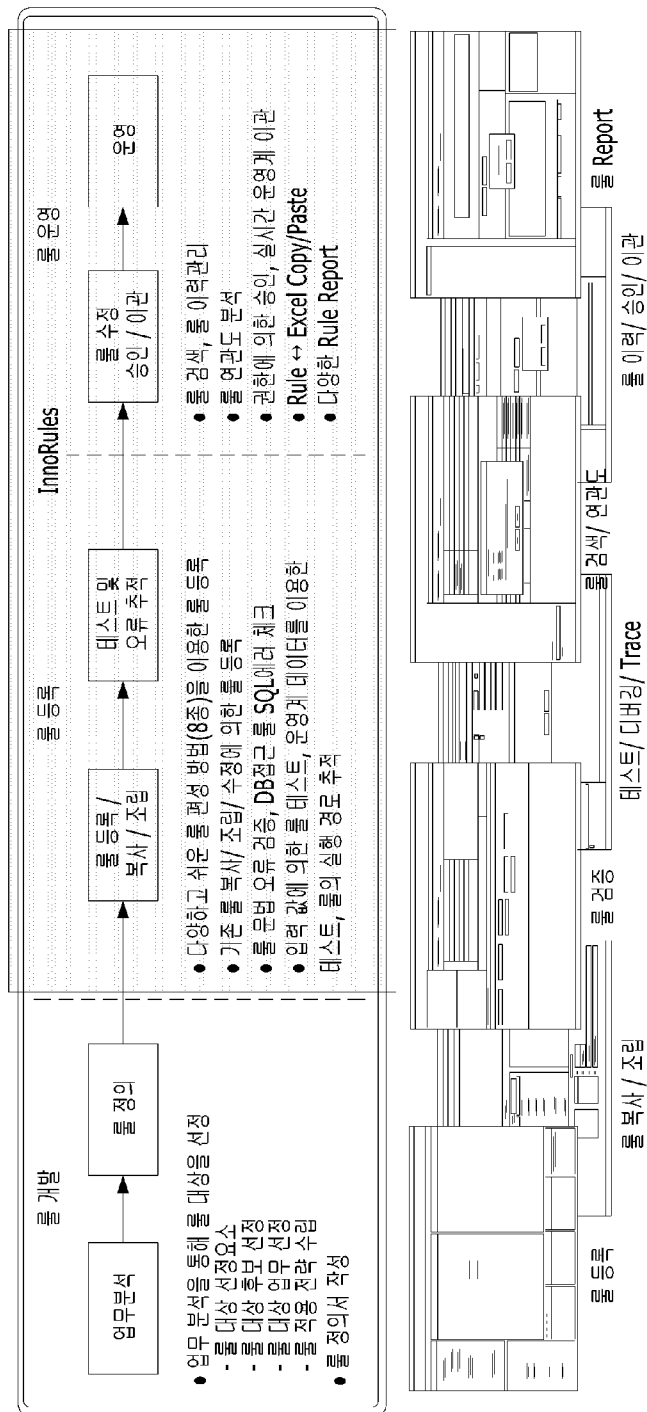
[도 12]

구분	조건 1	조건 2	조건 3
	DayAfter([입력일], -3) <= [변경기준일]	[변경기준일] = [입력일]	
	DayAfter([입력일], -10) <= [변경기준일]	[변경기준일] <= DayAfter([입력일], -3)	
	AddMonths([입력일], -3) <= [변경기준일]	[변경기준일] <= DayAfter([입력일], -10)	
	AddMonths([입력일], -3) > [변경기준일]	[순시관련(미제처리중) 계약여부] = "Y"	[입력지소속] = "5"
	AddMonths([입력일], -3) > [변경기준일]	[순시관련(미제처리중) 계약여부] = "Y"	[입력지소속] <> "5"
	AddMonths([입력일], -3) > [변경기준일]		

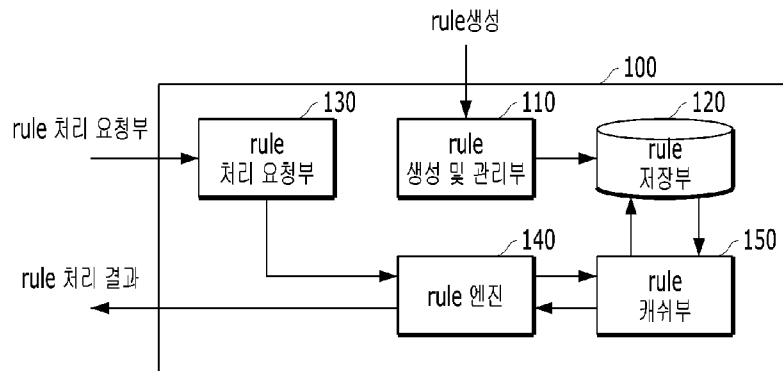
[도13]



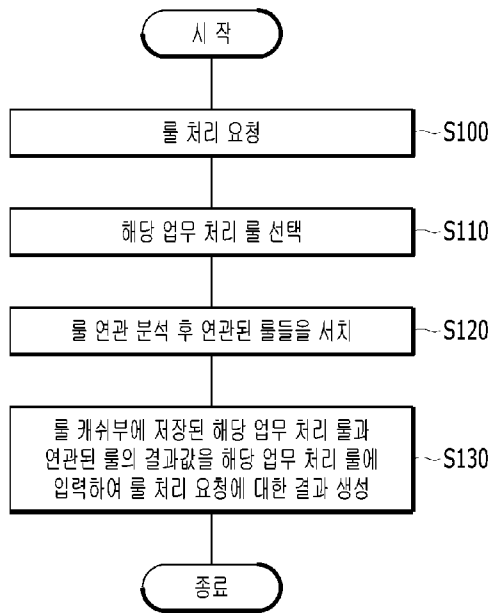
[도 14]



[도 15]



[도16]



[도17]

Rule Builder

파일 편집 보기 관리 도구 윈도우 현재 룰 도움말

규칙 편집기: [DEVPART] 개인소유_인수지침_점검_자동차

조건 1

번호	조건	연관된 룰	결과값
1	{개인소유_인수지침_점검_자동차}	{개인소유_인수지침_점검_자동차}	
2	{개인소유_인수지침_점검_자동차}	{개인소유_인수지침_점검_자동차}	
3			
4			
5			

조건 2

번호	조건	연관된 룰	결과값
1	{개인소유_인수지침_점검_자동차}	{개인소유_인수지침_점검_자동차}	
2	{개인소유_인수지침_점검_자동차}	{개인소유_인수지침_점검_자동차}	
3			
4			
5			
6			
7			

룰 정보

룰 형식: ☒ 서브(S) ☐ 메인(M)

룰 종류: ☒ 테이블 룰(T) ☐ 기변룰(V) ☐ DB 조회룰(D) ☐ Call Back 룰(C) ☐ 루프룰(L) ☐ RFC 룰(R) ☐ Flow 룰(F)

리턴 형식: ☒ 단일(I) ☐ 다중(U)

속성: ☒ 결과누적(A) ☐ 분기룰(B) ☐ 선행적자(P) ☐ 관계도에서 숨기기

날짜: 시작일: 2005-07-20 편집일: 2005-07-20

[도 18]

