



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2015년04월20일

(11) 등록번호 10-1512875

(24) 등록일자 2015년04월10일

- (51) 국제특허분류(Int. Cl.)
G06Q 10/06 (2012.01) G06Q 50/30 (2012.01)
- (21) 출원번호 10-2011-0138339
- (22) 출원일자 2011년12월20일
심사청구일자 2013년08월05일
- (65) 공개번호 10-2013-0082832
- (43) 공개일자 2013년07월22일
- (56) 선행기술조사문헌
KR1020040031429 A*
*는 심사관에 의하여 인용된 문헌

- (73) 특허권자
주식회사 케이티
경기도 성남시 분당구 불정로 90(정자동)
- (72) 발명자
박순향
경기 안양시 동안구 학의로 390, 105동 1503호 (평촌동, 푸른마을대우아파트)
김동완
서울 중구 다산로 32, 10동 604호 (신당동, 남산타운아파트)
이민구
경기 성남시 분당구 판교로 430, 106동 801호 (이매동, 아름마을아파트)
- (74) 대리인
특허법인신성

전체 청구항 수 : 총 5 항

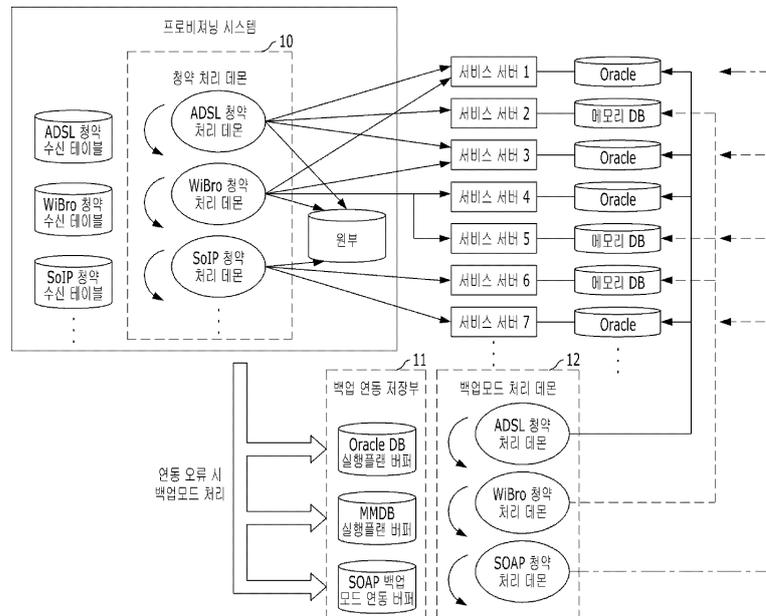
심사관 : 김정훈

(54) 발명의 명칭 **청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 시스템**

(57) 요약

본 발명은 청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 시스템에 관한 것으로, 프로비저닝 시스템이 특정 서비스를 위한 서비스 프로비저닝 처리 시 몇몇 서비스 서버들과의 사이에 네트워크 장애 등의 예외상황이 발생하는 경우에도 고객의 상품 가입에 대한 청약 처리는 성공적으로 완료하여 통신사의 비정상적인 매(뒷면에 계속)

대표도



출 누락이 발생하지 않게 하고, 일부 연동불가인 서비스 서버 구간에 대해서는 연동 장애 상황 해소 시 준실시간 (near real time)적인 백업모드 연동 방식으로 프로비저닝을 처리하여 청약 처리 과정에서의 시스템간 일시적인 연동 장애 상황에 능동적으로 대처하여 고객 VoC 최소화는 물론 청약 처리의 성공율을 향상시켜 서비스 활성화와 대고객 서비스 만족도를 향상시킬 수 있는, 청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 시스템을 제공하고자 한다.

이를 위하여, 본 발명은, 백업모드 프로비저닝 시스템에 있어서, 청약 수신 테이블을 페취(fetch)하여 원부를 구축하고, 서비스 서버의 요청 형태로 데이터를 가공하여 프로비저닝(Provisioning)하는 과정에서 상기 서비스 서버와의 연동이 불가능한 경우 상기 서비스 서버에 연동하려던 처리명령이나 호출 로직을 백업시키며, 청약 오더의 계약 ID(Identification)가 백업 연동 저장부에 존재하는 경우 상기 계약 ID에 대해 순차적인 처리를 보장하기 위한 청약 처리 수단; 상기 청약 처리 수단으로부터 백업되는 처리명령이나 호출 로직을 저장하기 위한 상기 백업 연동 저장부; 및 연동 오류가 발생하여 상기 백업 연동 저장부에 저장되어 있는 처리명령이나 호출 로직을 상기 서비스 서버와 연동 처리하기 위한 백업모드 처리 수단을 포함하되, 상기 청약 처리 수단은, 각 상품별로 전용의 청약 처리 데몬을 구비하고, 상기 백업 연동 저장부는, DB별 실행플랜 버퍼나 프로토콜별 백업모드 연동 버퍼를 포함하며, 상기 백업모드 처리 수단은, 상기 DB별 실행플랜 버퍼나 상기 프로토콜별 백업모드 연동 버퍼에 대한 전용의 백업모드 처리 데몬을 구비한다.

명세서

청구범위

청구항 1

삭제

청구항 2

백업모드 프로비저닝 시스템에 있어서,

청약 수신 테이블을 페취(fetch)하여 원부를 구축하고, 서비스 서버의 요청 형태로 데이터를 가공하여 프로비저닝(Provisioning)하는 과정에서 상기 서비스 서버와의 연동이 불가능한 경우 상기 서비스 서버에 연동하려던 처리명령이나 호출 로직을 백업시키며, 청약 오더의 계약 ID(Identification)가 백업 연동 저장부에 존재하는 경우 상기 계약 ID에 대해 순차적인 처리를 보장하기 위한 청약 처리 수단;

상기 청약 처리 수단으로부터 백업되는 처리명령이나 호출 로직을 저장하기 위한 상기 백업 연동 저장부; 및

연동 오류가 발생하여 상기 백업 연동 저장부에 저장되어 있는 처리명령이나 호출 로직을 상기 서비스 서버와 연동 처리하기 위한 백업모드 처리 수단을 포함하되,

상기 청약 처리 수단은, 각 상품별로 전용의 청약 처리 데몬을 구비하고,

상기 백업 연동 저장부는, DB별 실행플랜 버퍼나 프로토콜별 백업모드 연동 버퍼를 포함하며,

상기 백업모드 처리 수단은, 상기 DB별 실행플랜 버퍼나 상기 프로토콜별 백업모드 연동 버퍼에 대한 전용의 백업모드 처리 데몬을 구비하는, 백업모드 프로비저닝 시스템.

청구항 3

제 2 항에 있어서,

상기 백업모드 처리 데몬은 구동 시 디비전(DIVISION : 연동 시스템 구분)을 명시하고, 상기 DB별 실행플랜 버퍼나 상기 프로토콜별 백업모드 연동 버퍼로부터 상기 디비전에 해당하는 실행플랜이나 백업모드 연동필요 내역을 페취하여 해당 서비스 서버로 백업모드로 프로비저닝하되,

디비전 연동용 정보 구조 테이블을 이용하여 상기 해당 서비스 서버로 연동을 수행하는, 백업모드 프로비저닝 시스템.

청구항 4

제 3 항에 있어서,

상기 디비전 연동용 정보 구조 테이블은,

연동 시스템명, 서버 IP, 서버 연동 포트(Port), 서버 연동 URL, 서버 이름, 서버 연동 방식, 및 서버 연동 여부를 나타내는 필드를 포함하는, 백업모드 프로비저닝 시스템.

청구항 5

제 2 항 내지 제 4 항 중 어느 한 항에 있어서,

상기 DB별 실행플랜 버퍼는,

연동 시스템 구분(DIVISION) 필드, 입력 시간(STARTDATE) 필드, 처리 시간(ENDDATE) 필드, 서비스 계약

ID(SA_ID) 필드, 작업 지시 번호(WO_NO) 필드, 전송 일시(INS_DATE) 필드, 처리 유형 코드(INTF_CODE) 필드, 고객 ID(CUST_ID) 필드, SQL 구문 실행플랜(COMMAND) 필드, 바인드 데이터 버퍼(BIND_DATA) 필드, 바인드 변수 사용유무(BIND_METHOD) 필드, 에러 코드(ERRCODE) 필드, 에러 메시지(ERRMSG) 필드, 로그인 ID(LOGNAME) 필드, 및 에러 발생 일자(ERRDATE) 필드를 포함하는, 백업모드 프로비저닝 시스템.

청구항 6

제 2 항 내지 제 4 항 중 어느 한 항에 있어서,
상기 DB별 실행플랜 버퍼는,

연동 시스템 구분(DIVISION) 필드, 서비스 계약 ID(SA_ID) 필드, 작업 지시 번호(WO_NO) 필드, 전송 일시(INS_DATE) 필드, 로그인 ID(LOGNAME) 필드, 작업 내용(COMMAND) 필드, 에러 코드(ERRCODE) 필드, 에러 메시지(ERRMSG) 필드, 에러 발생 시간(ERRDATE) 필드, 수신 시간(RECV_DATE) 필드, 및 처리 유형 코드(INTF_CODE) 필드를 포함하는, 백업모드 프로비저닝 시스템.

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

발명의 설명

기술 분야

[0001] 본 발명은 백업모드 청약 처리 시스템에 관한 것으로, 더욱 상세하게는 서비스 상품 청약 처리 시스템에서 서비스 서버들에게 서비스에 필요한 정보를 프로비저닝(Provisioning)하는 과정에서 서비스 서버들이 네트워크 장애 등으로 인해 연동이 어려운 상황에서 해당 서비스 서버로의 프로비저닝 실패로 인해 전체 청약 처리 프로세스가 롤백(roll-back)되는 상황을 막기 위해 일부 서비스 서버들의 네트워크 장애 등의 예외상황 발생 시 전체 프로세스를 즉시 오류로 처리하지 않고 연동불가 상태인 서버들만 백업모드 버퍼에 저장하고 별도 백업모드 처리 데몬을 이용하여 비실시간으로 연동 처리하여 청약 처리 실패율을 최대한 감소시킴으로써 매출 향상에 기여할 수 있도록 하는, 청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 시스템에 관한 것이다.

[0002] 그리고 본 발명의 일실시예에서는 "오라클 DB"를 예로 들어 설명하나, 본 발명이 이에 한정되는 것이 아님을 미리 밝혀둔다.

배경 기술

[0003] 고객이 통신사 상품을 이용하기 위해 통신사에 서비스 이용을 청약하는 정보는 통신사의 영업 전산 시스템에 입력된다. 이후, 필요한 설비 구축의 과정을 거쳐 서비스에 필요한 고객의 상품 가입 정보들이 다양한 서비스 서버들에게 전달되어 서비스를 제공할 준비가 이루어진다.

[0004] 프로비저닝 시스템은 이러한 서비스에 필요한 고객, 상품, 계정, 단말, 및 부가서비스 등의 정보를 서비스 원부(청약 원부)로 구축 및 관리하고, 다양한 서비스 서버들이 요구하는 형태로 가공하여 해당 서비스 서버들에게 전달함으로써, 서비스 서버들이 서비스를 제공할 준비를 할 수 있도록 해 주는 시스템이다.

[0005] 프로비저닝 시스템이 서비스 원부를 구축하고 다수의 서비스 서버들에게 필요한 정보를 연동하는 과정에서 몇몇 서비스 서버들과의 일시적인 네트워크 장애로 인해 연동이 불가능한 경우가 발생할 수 있는데, 이때, 청약 처리

하던 전체 프로세스를 중단하고 롤백하는 경우 청약 처리 실패율이 높아지고 곧바로 고객의 VoC(Voice of Customer)로 이어질 수 있으며, 그에 따라 통신사의 대고객 이미지나 매출에 부정적인 영향을 미칠 수 있어 크리티컬(critical)한 이슈가 될 수 있다.

[0006] 따라서 프로비저닝 시스템은 프로비저닝할 대상 서비스 서버들과의 연동 및 정보 전달에 있어서 네트워크 장애 등의 예외상황에 대비하여 청약 처리 실패율 및 고객 VoC 최소화를 위한 적절한 체계를 고려하여 설계 및 구축되어야 한다.

발명의 내용

해결하려는 과제

[0007] 따라서 상기와 같은 종래 기술은, 프로비저닝 시스템이 특정 서비스를 위한 프로비저닝 처리 시 다양한 서비스 서버들로 연동하는 과정에서 몇몇 서비스 서버들과는 네트워크 장애로 청약 정보의 프로비저닝이 어려운 경우가 발생할 가능성이 있는데, 이때, 전체 처리중인 프로세스를 즉시 롤백하여 청약 처리 내역 전체를 오류 상태로 종료하는 경우 통신사의 청약 처리 실패율 증가로 매출에 직접적인 영향을 미칠 수 있으며, 또한 고객 민원 (VoC)이 증가하여 기업이미지 저하로 이어질 수 있는 문제점이 있으며, 이러한 문제점을 해결하고 상기 요구에 부응하고자 하는 것이 본 발명의 과제이다.

[0008] 따라서 본 발명은 프로비저닝 시스템이 특정 서비스를 위한 서비스 프로비저닝 처리 시 몇몇 서비스 서버들과의 사이에 네트워크 장애 등의 예외상황이 발생하는 경우에도 고객의 상품 가입에 대한 청약 처리는 성공적으로 완료하여 통신사의 비정상적인 매출 누락이 발생하지 않게 하고, 일부 연동불가인 서비스 서버 구간에 대해서는 연동 장애 상황 해소 시 준실시간(near real time)적인 백업모드 연동 방식으로 프로비저닝을 처리하여 청약 처리 과정에서의 시스템간 일시적인 연동 장애 상황에 능동적으로 대처하여 고객 VoC 최소화는 물론 청약 처리의 성공율을 향상시켜 서비스 활성화와 대고객 서비스 만족도를 향상시킬 수 있는, 청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 시스템을 제공하는 데 그 목적이 있다.

[0009] 본 발명의 목적들은 이상에서 언급한 목적으로 제한되지 않으며, 언급되지 않은 본 발명의 다른 목적 및 장점들은 하기의 설명에 의해서 이해될 수 있으며, 본 발명의 실시 예에 의해 보다 분명하게 알게 될 것이다. 또한, 본 발명의 목적 및 장점들은 특허 청구 범위에 나타낸 수단 및 그 조합에 의해 실현될 수 있음을 쉽게 알 수 있을 것이다.

과제의 해결 수단

[0010] 상기 목적을 달성하기 위한 본 발명의 시스템은, 백업모드 프로비저닝 시스템에 있어서, 청약 수신 테이블을 페취(fetch)하여 원부를 구축하고, 서비스 서버의 요청 형태로 데이터를 가공하여 프로비저닝(Provisioning)하는 과정에서 상기 서비스 서버와의 연동이 불가능한 경우 상기 서비스 서버에 연동하려던 처리명령이나 호출 로직을 백업시키며, 청약 오더의 계약 ID(Identification)가 백업 연동 저장부에 존재하는 경우 상기 계약 ID에 대해 순차적인 처리를 보장하기 위한 청약 처리 수단; 상기 청약 처리 수단으로부터 백업되는 처리명령이나 호출 로직을 저장하기 위한 상기 백업 연동 저장부; 및 연동 오류가 발생하여 상기 백업 연동 저장부에 저장되어 있는 처리명령이나 호출 로직을 상기 서비스 서버와 연동 처리하기 위한 백업모드 처리 수단을 포함하되, 상기 청약 처리 수단은, 각 상품별로 전용의 청약 처리 데몬을 구비하고, 상기 백업 연동 저장부는, DB별 실행플랜 버퍼나 프로토콜별 백업모드 연동 버퍼를 포함하며, 상기 백업모드 처리 수단은, 상기 DB별 실행플랜 버퍼나 상기 프로토콜별 백업모드 연동 버퍼에 대한 전용의 백업모드 처리 데몬을 구비한다.

[0011] 삭제

발명의 효과

[0012] 상기와 같은 본 발명은, 프로비저닝 시스템이 특정 서비스를 위한 서비스 프로비저닝 처리 시 몇몇 서비스 서버

들과의 사이에 네트워크 장애 등의 예외상황이 발생하는 경우에도 고객의 상품 가입에 대한 청약 처리를 성공적으로 완료하여 통신사의 비정상적인 매출 누락이 발생하지 않게 하고, 일부 연동불가인 서비스 서버 구간에 대해서는 연동 장애 상황 해소 시 준실시간(near real time)적인 백업모드 연동 방식으로 프로비저닝을 처리하여 청약 처리 과정에서의 시스템간 일시적인 연동 장애 상황에 능동적으로 대처하여 고객 VoC 최소화는 물론 청약 처리의 성공율을 향상시켜 서비스 활성화와 대고객 서비스 만족도를 향상시킬 수 있는 효과가 있다.

도면의 간단한 설명

- [0013] 도 1은 본 발명에 따른 청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 시스템의 일실시에 구성도,
- 도 2는 본 발명에 따른 오라클 DB 연동용 실행플랜 버퍼의 구조를 나타내는 도면,
- 도 3은 본 발명에 따른 메모리 DB 연동용 실행플랜 버퍼의 구조를 나타내는 도면,
- 도 4는 본 발명에 따른 오라클 DB 연동용 실행플랜 버퍼의 일예시도,
- 도 5는 본 발명에 따른 메모리 DB 연동용 실행플랜 버퍼의 일예시도,
- 도 6은 본 발명에 따른 디비전(DIVISION) 연동용 정보 구조 테이블의 일예시도,
- 도 7은 본 발명에 따른 청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 방법 중 오류 발생에 대해 백업 연동 저장부에 기록하는 과정에 대한 일실시에 흐름도,
- 도 8은 본 발명에 따른 청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 방법 중 백업 연동 저장부와 백업모드 처리 데몬을 이용하여 오류가 발생한 연동 처리를 백업모드로 실행하는 과정에 대한 일실시에 흐름도이다.

발명을 실시하기 위한 구체적인 내용

- [0014] 상술한 목적, 특징 및 장점은 첨부된 도면을 참조하여 상세하게 후술되어 있는 상세한 설명을 통하여 보다 명확해 질 것이며, 그에 따라 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자가 본 발명의 기술적 사상을 용이하게 실시할 수 있을 것이다. 또한, 본 발명을 설명함에 있어서 본 발명과 관련된 공지 기술에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에 그 상세한 설명을 생략하기로 한다. 이하, 첨부된 도면을 참조하여 본 발명에 따른 바람직한 실시 예를 상세히 설명하기로 한다.
- [0015] 그리고 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때 이는 "직접적으로 연결"되어 있는 경우뿐만 아니라 그 중간에 다른 소자를 사이에 두고 "전기적으로 연결"되어 있는 경우도 포함한다. 또한, 어떤 부분이 어떤 구성요소를 "포함" 또는 "구비"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함하거나 구비할 수 있는 것을 의미한다.
- [0016] 도 1은 본 발명에 따른 청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 시스템의 일실시에 구성도이다.
- [0017] 도 1에 도시된 바와 같이, 본 발명에 따른 청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 시스템은, 청약 수신 테이블(청약 정보 수신 테이블)을 페취(fetch)하여 청약 원부를 구축하고, 서비스 서버의 요청 형태로 데이터를 가공하여 프로비저닝하는 과정에서 프로비저닝한 서비스 서버들 중 일부 서비스 서버가 일시적인 네트워크 장애 등의 원인으로 연동이 불가능한 상태인 경우, 전체 청약 처리 프로세스를 즉시 오류로 처리하지 않고, 오류가 발생하는 서비스 서버에 연동하려던 처리명령이나 호출 로직을 백업시키고, 해당 청약 오더의 계약 ID(Identification)가 백업 연동 저장부(11)에 존재하는 경우에 연동오류가 발생한 계약 ID에 대해 순차적인 처리를 보장하기 위한 청약 처리 데몬(10), 상기 청약 처리 데몬(10)으로부터 백업되는 "오류가 발생하는 서비스 서버에 연동하려던 처리명령이나 호출 로직"을 저장하기 위한 백업 연동 저장부(11), 및 연동 오류가 발생하여 상기 백업 연동 저장부(11)에 저장되어 있는 처리명령이나 호출 로직을 비실시간(준실시간)으로 해당 서비스 서버와 연동 처리하기 위한 백업모드 처리 데몬(12)을 포함한다.
- [0018] 다시 말하면, 각 상품(예 : ADSL, WiBro, SoIP 등)의 청약 처리 전용 데몬은 해당 청약 수신 테이블을 페취

(fetch)하여 청약 원부를 구축하고, 서비스 제공에 필요한 서비스 서버들이 요청한 형태대로 원부 데이터를 가공하여 해당 서비스 서버로 프로비저닝한다. 여기서, 프로비저닝 과정은 직접 DB 연동으로 하거나 SOAP(Simple Object Access Protocol) 등의 웹서비스 연동으로 프로비저닝을 한다. 이때, 프로비저닝할 서비스 서버들 중 일부 서비스 서버들이 일시적인 네트워크 장애 등의 원인으로 연동이 불가능한 상태인 경우, 청약 처리 데몬이 전체 프로세스를 즉시 중단하지는 않고, 오류가 발생하는 해당 서비스 서버에 연동하려던 처리 명령이나 호출 로직을 로컬의 DB별 실행플랜 버퍼(예 : 오라클 DB 실행플랜 버퍼, MMDB 실행플랜 버퍼)나 백업모드 연동 버퍼(예 : SOAP 백업모드 연동 버퍼)에 저장해 두고, 각각의 백업모드 처리 데몬이 별도로 구동되어 연동 오류가 발생한 청약 오더들에 대한 처리를 재시도하도록 한다.

[0019] 이를 좀 더 상세히 살펴보면 다음과 같다.

[0020] 본 발명의 일실시예에 따르면, 백업모드 프로비저닝 시스템이 특정 서비스에 필요한 정보들을 해당 서비스를 제공하는 데에 필요한 서비스 서버들에게 프로비저닝하는 과정에서 일부 연동대상 서비스 서버들과 연동불가 상태에 직면했을 때 서비스 서버들로 프로비저닝할 정보를 백업모드 프로비저닝 시스템 내부에 프로토콜별 "백업모드 연동버퍼" 또는 데이터베이스 유형별 "오라클(Oracle) DB 실행플랜(Plan) 버퍼"나 "MMDB 실행플랜 버퍼"에 저장하고, 이러한 "백업모드 연동버퍼"와 "실행플랜 버퍼"에 저장된 연동정보를 주기적으로 백업모드 처리 데몬(12)이 돌면서 연동오류가 발생한 서비스 서버들과 연동 재시도를 수행하고, 이 과정에서 일시적인 연결장애인 경우는 대부분 정상으로 데이터 프로비저닝이 완료되며, 이 과정에서도 해결이 안되는 서비스 서버들에 대해서는 부가적으로 운용요원이 개입하여 오류 원인 파악 및 조치를 하도록 할 수도 있다.

[0021] 그리고 백업모드 처리 데몬(12)을 구동 시 디비전(DIVISION : 연동 시스템 구분)을 지정하여 띄우며, 구동된 데몬은 "오라클 DB 실행플랜 버퍼", "MMDB 실행플랜 버퍼" 또는 "백업모드 연동버퍼"로부터 지정된 디비전(DIVISION)에 해당하는 실행플랜이나 백업모드 연동필요 내역을 페취(fetch)하여 해당 서비스 서버로 백업모드로 프로비저닝한다. 이때, 디비전(DIVISION)은 연동을 재시도할 서비스 서버를 나타내며, 디비전(DIVISION) 연동용 정보 구조 테이블에서 서버 IP(Internet Protocol), 서버 연동 포트(Port), 서버 연동 URL(Uniform Resource Locator), 서버 연동 방식 등을 참고하여 해당 서비스 서버로 연동을 수행하게 된다. 백업모드 처리 데몬(12)은 항상 동작하도록 자동 모드로 구동하거나 부가적으로 운용자가 처리 시점을 판단하여 수동으로 구동하도록 할 수도 있다.

[0022] 그리고 청약 처리 데몬(10)은 각각의 청약 오더 처리 시에 키가 되는 정보(예를 들면, 계약 ID)가 백업 연동 저장부(11)에 존재하는지 체크하여 백업모드 테이블에 존재하는 경우는 해당 서버와의 연동이 정상이라도 백업모드 테이블에 순서대로 저장을 해야 하며, 서버와의 연동이 정상화되어 백업모드 테이블에 더 이상 해당 키에 대한 오더가 쌓여있지 않은 경우 즉, 해당 키에 대해 백업모드 테이블에 쌓여있던 정보가 이미 백업모드 처리 데몬(12)에 의해 정상 처리되어 해당 서비스 서버로 연동 처리된 경우에는 백업모드 테이블에 더 이상 저장하지 않고 바로 정상연동되는 서비스 서버로 연동(프로비저닝) 처리를 한다. 이는 백업 연동 저장부(11)의 백업모드 테이블에 특정 키에 대해 처리할 오더가 있는 경우 시간순으로 차례대로 백업모드 테이블의 오더를 먼저 순차적으로 처리한 후 해당 키(예 : 계약 ID)에 대하여 백업모드 테이블에 쌓인 오더가 완전히 처리되어 클리어(clear)되었는지 확인한 후에 정상연동을 그 다음 순서로 처리하여 오더 처리 시 순차적인 처리가 되게 하여 해당 키에 대한 오더 처리 일관성을 보장하기 위한 절차이다.

[0023] 일례로, 서비스 서버들 중에 연동 방식이 데이터베이스 링크(database link)로 오라클 DB에 접속하여 직접 DB 쿼리(query)를 수행하는 서비스 서버와의 연동에 대해 일시적인 네트워크 장애 등이 발생 시 백업모드 프로비저닝 시스템의 로컬에 위치하는 "오라클 DB 실행플랜 버퍼"에 연동 시스템 정보, 데이터베이스 실행명령문, 바인딩 변수 및 바인딩 값들을 저장하고, 이후 오라클 백업모드 처리 데몬이 "오라클 DB 실행플랜 버퍼"로부터 해당 디비전(DIVISION)에 해당하는 실행플랜을 페취(fetch)해서 차례대로 데이터베이스 링크(database link)로 접속하여 데이터베이스 명령문을 실행함으로써, 해당 디비전(DIVISION)의 서비스 서버로 서비스에 필요한 정보를 프로비저닝한다.

[0024] 다른 예로, 서비스 서버들 중에 연동 방식이 메모리 데이터베이스(Memory DB)에 접속하여 직접 DB 쿼리(query)를 수행하는 서비스 서버와의 연동에 대해 일시적인 네트워크 장애 등이 발생 시 백업모드 프로비저닝 시스템의 로컬에 위치하는 "MMDB 실행플랜 버퍼"에 연동 시스템 정보, 데이터베이스 실행명령문을 저장하고, 이후 MMDB 백업모드 처리 데몬이 "MMDB 실행플랜 버퍼"로부터 해당 디비전(DIVISION)에 해당하는 실행플랜을 페취(fetch)해서 차례대로 MMDB에 접속하여 데이터베이스 명령문을 실행함으로써, 해당 디비전(DIVISION)의 서비스 서버로 서비스에 필요한 정보를 프로비저닝한다.

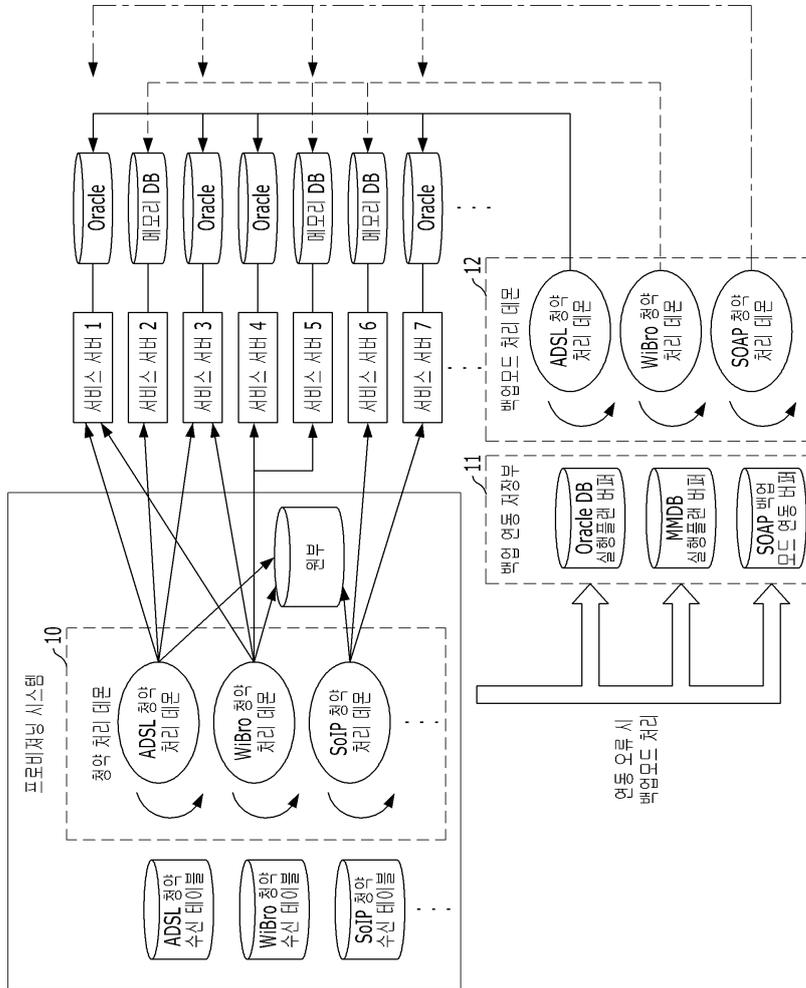
- [0025] 도 2는 본 발명에 따른 오라클 DB 연동용 실행플랜 버퍼(TB_ORA_PLAN_ORDER)의 구조를 나타내는 도면이다.
- [0026] 도 2에 도시된 바와 같이, 본 발명에 따른 오라클 DB 연동용 실행플랜 버퍼는, 연동 시스템 구분(DIVISION) 필드, 입력 시간(STARTDATE) 필드, 처리 시간(ENDDATE) 필드, 서비스 계약 ID(SA_ID) 필드, 작업 지시 번호(WO_NO) 필드, 전송 일시(INS_DATE) 필드, 처리 유형 코드(INTF_CODE) 필드, 고객 ID(CUST_ID) 필드, SQL 구문 실행플랜(COMMAND) 필드, 바인드 데이터 버퍼(BIND_DATA) 필드, 바인드 변수 사용유무(BIND_METHOD) 필드, 에러 코드(ERRCODE) 필드, 에러 메시지(ERRMSG) 필드, 로그인 ID(LOGNAME) 필드, 및 에러 발생 일자(ERRDATE) 필드를 포함한다.
- [0027] 여기서, 연동 시스템 구분(DIVISION) 필드는 오라클 데이터베이스로 연동할 시스템 중 현재 연동이 불가능한 시스템들을 구분하는 구분자를 넣는 필드로서, 예를 들면, 망 접속을 위한 AAA라는 인증 시스템인 경우 "AAAA", IPTV 서비스 인증 시스템인 경우 "HDS"와 같이 시스템별로 각 시스템을 표현하기 위해 할당된 구분자 키워드를 디비전(DIVISION) 필드에 넣는다.
- [0028] 그리고 서비스 계약 ID(SA_ID) 필드는 해당 청약에 대해 영업 전산 시스템에서 오더를 받은 서비스 서버로 프로비저닝해야 할 계약의 계약 ID를 의미하며, 처리 유형 코드(INTF_CODE) 필드는 오더 처리할 유형이 신규, 변경, 해지, 정지, 부활 등과 같이 어떤 유형인지를 표시하는 필드이다.
- [0029] 그리고 SQL 구문 실행플랜(COMMAND) 필드에는 실제 오라클 데이터베이스에 실행시켜야 할 명령어(쿼리문)를 넣고, 바인드 데이터 버퍼(BIND_DATA) 필드에는 명령어의 각 변수에 해당하는 상수값을 넣는다.
- [0030] 그리고 로그인 ID(LOGNAME) 필드에는 해당 프로비저닝할 청약의 로그인 ID 값을 넣으며, 에러 코드 필드, 에러 메시지 필드, 에러 발생 일자 필드에는 각각 해당 값을 넣어 백업모드 처리 데몬(12)이 프로비저닝 재시도 시 참고하거나 부가적으로 운용자가 수작업으로 오류 데이터를 처리 시 참고할 수 있도록 한다.
- [0031] 도 3은 본 발명에 따른 메모리 DB 연동용 실행플랜 버퍼(TB_MMDB_PLAN_ORDER)의 구조를 나타내는 도면이다.
- [0032] 도 3에 도시된 바와 같이, 본 발명에 따른 메모리 DB 연동용 실행플랜 버퍼는, 연동 시스템 구분(DIVISION) 필드, 서비스 계약 ID(SA_ID) 필드, 작업 지시 번호(WO_NO) 필드, 전송 일시(INS_DATE) 필드, 로그인 ID(LOGNAME), 응용 서비스 ID 필드, 작업 내용(COMMAND) 필드, 에러 코드(ERRCODE) 필드, 에러 메시지(ERRMSG) 필드, 에러 발생 시간(ERRDATE) 필드, 수신 시간(RECV_DATE) 필드, 및 처리 유형 코드(INTF_CODE) 필드를 포함한다.
- [0033] 여기서, 연동 시스템 구분(DIVISION) 필드는 메모리 데이터베이스로 연동할 시스템 중 현재 연동이 불가능한 시스템들을 구분하는 구분자를 넣는 필드로서, 예를 들면, 망 접속을 위한 AAA의 메모리 DB인 경우 "AAA_MMDB"로, 인터넷 전화 제공 서비스 서버 중 하나인 번호 번역 서버인 경우 "NRS"와 같이 시스템별로 각 시스템을 표현하기 위해 할당된 구분자 키워드를 디비전(DIVISION) 필드에 넣는다. MMDB 실행플랜을 버퍼에 저장하기 위해서는 작업 내용(COMMAND) 필드에 데이터베이스 수행 명령문(쿼리)을 그대로 넣는다.
- [0034] 도 4는 본 발명에 따른 오라클 DB 연동용 실행플랜 버퍼의 일예시도로서, 본 발명의 일실시예에 따른 오라클 DB 연동용 실행플랜 버퍼의 범례를 보여주고 있다.
- [0035] 도 5는 본 발명에 따른 메모리 DB 연동용 실행플랜 버퍼의 일예시도로서, 본 발명의 일실시예에 따른 MMDB 연동용 실행플랜 버퍼의 범례를 보여주고 있다.
- [0036] 도 6은 본 발명에 따른 디비전(DIVISION) 연동용 정보 구조 테이블의 일예시도로서, 디비전(DIVISION) 연동용 타겟 서버 정보를 표현하는 정보 구조를 보여주고 있다.
- [0037] 즉, 도 6의 디비전(DIVISION) 연동용 정보 구조 테이블은, 백업모드 처리 데몬(12)이 오라클 데이터베이스 연동용 실행플랜 버퍼와 메모리 데이터베이스 연동용 실행플랜 버퍼의 데이터를 페취(fetch)하여 디비전(DIVISION)

에 해당하는 서비스 서버로 프로비저닝을 보내기 위해 참조하는 테이블이다.

- [0038] 도 6에 도시된 바와 같이, 본 발명에 따른 디비전(DIVISION) 연동용 정보 구조 테이블은, 연동 시스템명, 서버 IP, 서버 연동 포트(Port), 서버 연동 URL, 서버 이름, 서버 연동 방식, 및 서버 연동 여부를 나타내는 필드를 포함한다.
- [0039] 도 7은 본 발명에 따른 청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 방법 중 오류 발생에 대해 백업 연동 저장부에 기록하는 과정에 대한 일실시에 흐름도이다.
- [0040] 먼저, 상품별 청약 처리 데몬(10)은 상품별 청약 수신 테이블을 읽어(701) 제약조건 만족여부를 체크하고 필요한 각종 로직을 처리한 후(702) 로컬에 청약 원부를 구축한 후(703) 관련 서비스 서버들로 프로비저닝(Provisioning)을 시작한다(704).
- [0041] 이후, 서비스 서버I(I=0,...,N)의 연동상태가 정상인지 확인하여(705) 연동상태가 정상이 아니면 "713" 과정으로 진행하고, 연동상태가 정상이면 현재 오더의 키(Key)가 되는 계약 ID(SA_ID)가 서비스 서버I에 해당하는 백업 연동 저장부(11)(서비스 서버I가 오라클 DB이면 오라클 DB 실행플랜 버퍼, 메모리 DB이면 MMDB 실행플랜 버퍼, SOAP 연동 유형이면 SOAP 백업모드 연동 버퍼)에 존재하는지 확인하여(706) 만약 백업 연동 저장부(11)에 존재하면 현재 수행하던 프로비저닝 정보도 해당 서비스 서버I용 백업 연동 저장부(11)에 저장하도록 지시한 후에(707) "713" 과정으로 진행한다. 이렇게 함으로써 해당 계약 ID(SA_ID)에 대해 이전에 백업 연동 저장부(11)에 쌓인 오더가 모두 처리된 이후에 현재 건의 처리가 진행되도록 할 수 있다.
- [0042] 한편, 상기 확인 결과(706), 현재 오더의 키(계약 ID)가 서비스 서버I의 백업 연동 저장부(11)에 존재하지 않는 경우, I번째 서비스 서버로 프로비저닝 연동 처리를 수행한다(708). 프로비저닝 시 I번째 서비스 서버로의 연동이 성공인지 확인하여(709) 연동 성공이면 연동 성공 결과를 로그에 기록한 후(710) 프로비저닝할 서비스 서버가 더 존재하는지 확인하여(711) 존재하지 않으면 트랜잭션을 종료하고, 서비스 서버가 더 존재하면 I값을 1증가시킨 후(712) 상기 "704" 과정으로 진행하여 이후의 과정을 반복 수행한다.
- [0043] 만약, 프로비저닝 시 I번째 서비스 서버로의 연동이 실패한 경우(709) 연동대상의 프로토콜 유형을 확인하여(713) 오라클 데이터베이스 연동인 경우 오라클 실행플랜 버퍼에 기록하고(714), 메모리 데이터베이스 연동인 경우 MMDB 실행플랜 버퍼에 기록하며(715), SOAP 연동인 경우 SOAP 백업모드 연동 버퍼에 기록한다(716). 이후, 상기 "711" 과정으로 진행하여 프로비저닝할 서비스 서버가 더 존재하는 동안 계속 상기 과정을 반복 수행한 후 트랜잭션을 종료한다.
- [0044] 도 8은 본 발명에 따른 청약 처리 시 연동유형별 오류 최소화를 위한 백업모드 프로비저닝 방법 중 백업 연동 저장부와 백업모드 처리 데몬을 이용하여 오류가 발생한 연동 처리를 백업모드로 실행하는 과정에 대한 일실시에 흐름도이다.
- [0045] 먼저, 백업모드 처리 데몬(12)은 구동 시 처리할 서비스 서버에 해당하는 디비전(DIVISION : 연동 시스템 구분)을 명시하여 구동을 시작한다.
- [0046] 그리고 백업모드 처리 데몬(12)은 구동되면 "백업모드 실행플랜 버퍼(즉, 백업 연동 저장부)"에서 해당 디비전(DIVISION)에 대한 실행플랜을 페취(fetch)하여(801) 실행플랜이 존재하는지 확인한다(802).
- [0047] 상기 확인 결과(802), 실행플랜이 존재하지 않으면 백업모드 처리 데몬(12)을 종료하고, 실행플랜이 존재하면 디비전(DIVISION)에 해당하는 서비스 서버로 해당 실행플랜을 연동할 준비를 하는데, 이때 필요 시 변수에 값을 바인드(bind)하는 등의 작업을 수행한다(803).
- [0048] 이후, 디비전(DIVISION)의 타겟 서버(서비스 서버) 정보를 읽고(804) 프로토콜로 접속할 준비를 하여(805) 정의된 프로토콜로 타겟 서버 접속 및 실행플랜을 호출한다(806). 오류가 발생하였는지를 확인하여(807) 오류가 발생하지 않았으면, 즉 성공이면 "백업모드 실행플랜 버퍼"에 처리 시간(ENDDATE)을 기록하고(808) 상기 "801" 과정으로 진행하며, 오류가 발생하였으면 "백업모드 실행플랜 버퍼"에 에러 코드, 에러 메시지, 및 에러 발생 일자를 기록한 후에(809), "801" 과정으로 진행하여 해당 "백업모드 실행플랜 버퍼"에 해당 디비전(DIVISION)으로 프로비저닝할 오더가 존재하는 동안 계속 상기 과정을 반복 수행한다.
- [0049] 한편, 부가적으로 백업모드 처리 데몬 수행 후에도 일정기간 계속 백업모드 실행플랜 버퍼에 존재하는 오더 건

도면

도면1



도면2

필드명	설명
DIVISION	연동 시스템 구분 (AAA, BAS, CISP1, DB_IAMS, GURO, HDS, IAMS, KIBOT, MNA2, MNA6, SGW, SMS, SOIPBIZAS, TIMECODI, UMS, VOIPAS, WPOS)
STARTDATE	입력 시간
ENDDATE	처리 시간
SA_ID	서비스 계약 ID
WO_NO	작업 지시 번호
INS_DATE	전송 일시
INTF_CODE	처리 유형 코드
CUST_ID	고객 ID
COMMAND	SQL구문(Bind) 실행플랜
BIND_DATA	Bind 데이터 버퍼
BIND_METHOD	Bind 변수 사용유무
ERRCODE	에러 코드
ERRMSG	에러 메시지
LOGNAME	로그인 ID(응용서비스ID)
ERRDATE	에러 발생 일자

도면3

필드명	설명
DIVISION	연동 시스템 구분 (AAA_CNTR2, AAA_GURO2, AAA_MMDB, AIMS, CISPMMDB, CPS, IAMS, IAPD, NRS, SGW, VOIPIAMS)
SA_ID	서비스 계약 ID
WO_NO	작업 지시 번호
INS_DATE	전송 일시
LOGNAME	로그인 ID(응용서비스 ID)
COMMAND	작업 내용(SQL)
ERRCODE	에러 코드
ERRMSG	에러 메시지
ERRDATE	에러 발생 시간
RECV_DATE	수신 시간
INTF_CODE	처리 유형 코드

도면4

필드명	Data1	Data2
DIVISION	AAA	SMS
STARTDATE	201006141446380672	200911271822360421
ENDDATE		
SA_ID	wb005091200	ads80801030
WO_NO	wono0000060	wono0000001
INS_DATE	09/25/2008 00:02:00	04/01/2007 00:01:00
INTF_CODE	4202	4201
CUST_ID	cust1005091	0837cust030
COMMAND	UPDATE terminals@DB_AAA SET terminalseq = :n1, terminalmacid = upper(:n2), terminalesn = :n3 WHERE hd_sa_id = :n4	INSERT INTO TB_NESPOT_ORDER@DB_SMS (p_sa_id, sa_id, wo_no, ins_date, ord_flag,chg_code, svc_id, passwd, mac_id,mac_flag,p_sa_cd, ...) VALUES (:n1,:n2,:n3,to_date(:n4,'YYYYMMDDHH2 4MISS'),:n5, :n6, :n7,:n8,:n9,:n10,:n11,:n12,:n13, nvl(:n14,')',...)
BIND_DATA	700820 hs-w0-10-06-04- a B700820 wb005091201	ads80801030 ads80801030 wono00000 01 20070401000100 01 99 ads80801ca. h...
BIND_METHOD		
ERRCODE	1	1
ERRMSG	ORA-00001: unique constraint (WBAUTH.IDX_TERMINALS_T RMINALMACID) viol	ORA-00001: 무결성 제약 조건(IASM.TB_NESPOT_ORDER_PK) 에 위배됩니다
LOGNAME	w10050910a	ads80801ca.h
ERRDATE	20110217	20091201

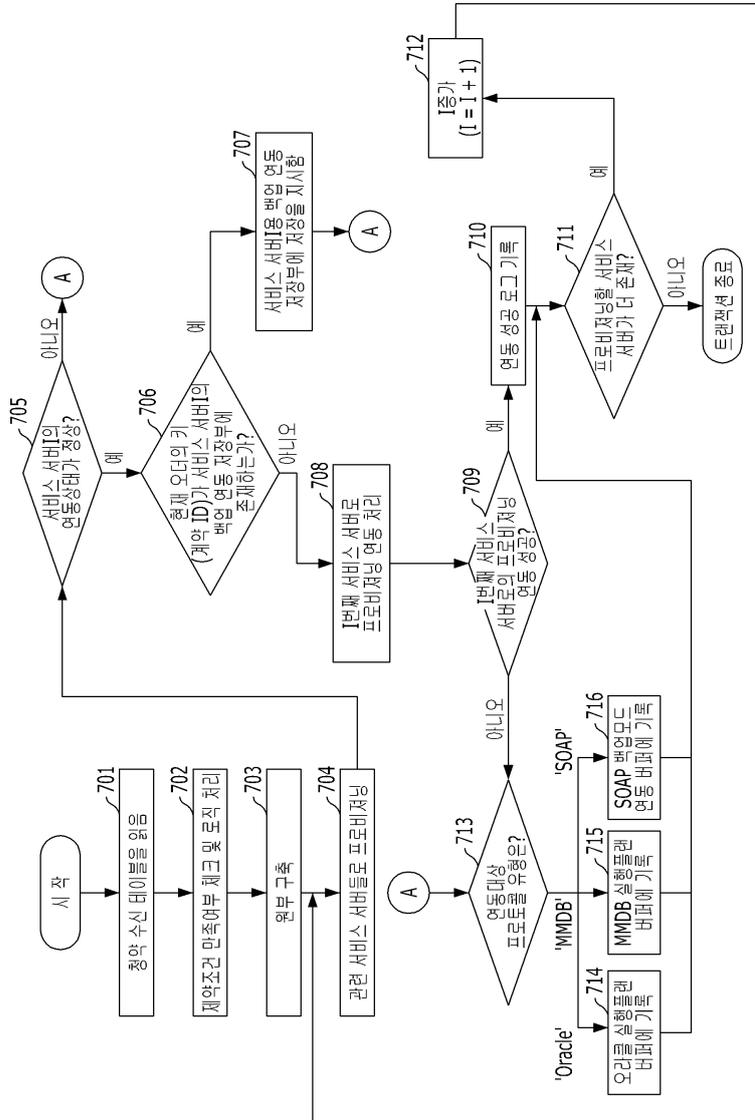
도면5

필드명	Data1	Data2
DIVISION	AIMS	IAMS
SA_ID	NS101031010	voip4132020
WO_NO	o0000000001	wono0000001
INS_DATE	11/01/2007 00:01:00	11/01/2007 00:01:00
LOGNAME	01010103101	0277413202
COMMAND	INSERT INTO TB_AIMS_ORDER (sa_id, insert_date, puid, old_puid, svc_no, svc_cd, ..) VALUES ('NS101031010' , TO_DATE('20110511095629 019300', 'YYYYMMDDHH24MISSSSSS S'), '07010103101', ...)	INSERT INTO iams.USERS (loginid, password, realid ,...) VALUES ('SM!pri4132211', '827CCB0EEA8A706C4C34A16891 F84E7B', 'SM!pri4132211',)
ERRCODE	-200790	
ERRMSG	Unable to insert(or update) NULL into NOT NULL column.	
ERRDATE	09/06/2011 00:00:00	
RECV_DATE	201105110956290195	201108101810290543
INTF_CODE	4201	4201

도면6

필드명	설명
DIVISION	연동 시스템명
SERVER_IP	서버 IP
SERVER_PORT	서버 연동 PORT
SERVER_URL	서버 연동 URL
SERVER_NAME	서버 이름
SERVER_TYPE	서버 연동 방식
SERVER_USED	서버 연동 여부

도면7



도면8

