(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2021년11월05일
(11) 등록번호 10-2321860
(24) 등록일자 2021년10월29일

(51) 국제특허분류(Int. Cl.)
*G06F 9/54* (2018.01)
(52) CPC특허분류
*G06F 9/546* (2013.01)
(21) 출원번호 10-2019-7037423
(22) 출원일자(국제) 2018년06월08일
심사청구일자 2021년04월08일
(85) 번역문제출일자 2019년12월18일
(65) 공개번호 10-2020-0012901
(43) 공개일자 2020년02월05일
(86) 국제출원번호 PCT/EP2018/065192
(87) 국제공개번호 WO 2018/224659
국제공개일자 2018년12월13일
(30) 우선권주장
1755105 2017년06월08일 프랑스(FR)
15/617,180 2017년06월08일 미국(US)
(56) 선행기술조사문헌
US20060146848 A1
*(뒷면에 계속)*
전체 청구항 수 : 총 23 항

(73) 특허권자
**아마데우스 에스.에이.에스.**
프랑스 에프-06410 비요 소피아 앙띠폴리 루뜨 드 뺑 몽따르 485
(72) 발명자
**장 이젠**
프랑스 06902 소피아 앙티폴리스 루트 뒤 펭 몽타르 485
**캠포라 시몬**
영국 더블유52에스엔 런던 억스브리지 로드 32 아크 타워 132
*(뒷면에 계속)*
(74) 대리인
**김태홍, 김진희**

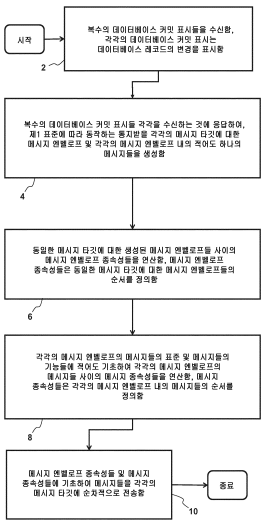
심사관 : 유진태

(54) 발명의 명칭 **다중-표준 메시지 프로세싱**

## (57) 요 약

메시지들이 메시지 프로세싱 엔진에 의해 분산 메시징 시스템에서 프로세싱된다. 메시지 타깃들은 제1 표준 또는 제2 레거시 표준에 따라 동작한다. 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 제1 표준에 따라 동작하는 통지받을 각각의 메시지 타깃에 대한 메시지 엔벨로프 및 각각의 메시지 엔벨로프 내의 적어도 하나의 메시지들이 생성된다. 동일한 메시지 타깃에 대한 생성된 메시지 엔벨로프들 사이의 메시지 엔벨로프 종속성들이 연산된다. 각각의 메시지 엔벨로프의 메시지들의 표준 및 메시지들의 기능들에 적어도 기초하여 각각의 메시지 엔벨로프 내의 메시지들 사이의 메시지 종속성들이 연산된다. 메시지 엔벨로프 종속성들 및 메시지 종속성들에 기초하여 메시지들이 순차적으로 각각의 메시지 타깃에 전송된다.

*대 표 도* - 도1



(72) 발명자

**마찬드 이반 자비에**

미국 매사추세츠주 01886 웨스트포드 파워스 로드 78

**폰톨리엣 로익**

미국 매사추세츠주 02118 보스턴 유닛 5 콜롬버스 애비뉴 555

(56) 선행기술조사문헌

US20100293235 A1

US20080098405 A1

US20140201762 A1

KR1020150037980 A

# 명 세 서

## 청구범위

### 청구항 1

복수의 메시지 타깃들을 갖는 분산 메시징 시스템에서 메시지를 프로세싱하기 위한 방법에 있어서,

상기 복수의 메시지 타깃들 중 적어도 하나의 메시지 타깃은 제1 표준에 따라 동작하고, 상기 복수의 메시지 타깃들 중 적어도 하나의 메시지 타깃은 제2 레거시 표준(second legacy standard)에 따라 동작하고,

상기 방법은,

메시지 프로세싱 엔진에 의해, 복수의 데이터베이스 커밋 표시(database commit indication)들을 수신하는 단계 - 각각의 데이터베이스 커밋 표시는 데이터베이스 레코드의 변경을 표시함 -;

상기 복수의 데이터베이스 커밋 표시들 중 한 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 상기 메시지 프로세싱 엔진에 의해, 상기 제1 표준에 따라 동작하는 통지받을 각각의 메시지 타깃에 대한 메시지 엔벨로프(message envelope) 및 각각의 메시지 엔벨로프 내의 적어도 하나의 메시지를 생성하는 단계;

상기 메시지 프로세싱 엔진에 의해, 동일한 메시지 타깃에 대한 생성된 메시지 엔벨로프들 사이의 메시지 엔벨로프 종속성(message envelope dependency)을 연산(computing)하는 단계 - 상기 메시지 엔벨로프 종속성은 상기 동일한 메시지 타깃에 대한 메시지 엔벨로프들의 순서를 정의함 -;

상기 메시지 프로세싱 엔진에 의해, 적어도 각각의 메시지 엔벨로프의 메시지들의 표준 및 상기 메시지들의 기능들에 기초하여 각각의 메시지 엔벨로프 내의 메시지들 사이의 메시지 종속성을 연산하는 단계 - 상기 메시지 종속성은 각각의 메시지 엔벨로프 내의 상기 메시지들의 순서를 정의함 -; 및

상기 메시지 엔벨로프 종속성 및 상기 메시지 종속성에 기초하여 순차적으로 상기 메시지 프로세싱 엔진으로부터 메시지를 각각의 메시지 타깃에 전송하는 단계

를 포함하는, 메시지 프로세싱 방법.

### 청구항 2

제1항에 있어서,

상기 제1 표준에 따라 동작하는 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제1 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 메시지 엔벨로프를 생성하는 단계를 더 포함하는, 메시지 프로세싱 방법.

### 청구항 3

제2항에 있어서,

각각의 메시지 엔벨로프 내의 적어도 하나의 메시지를 생성하는 단계를 더 포함하는, 메시지 프로세싱 방법.

### 청구항 4

제3항에 있어서,

상기 각각의 메시지 엔벨로프 내의 적어도 하나의 메시지를 생성하는 단계는,

상기 제1 표준에 따라 동작하는 제1 메시지 타깃에 릴레이될 복수의 상호종속적 제1 메시지(interdependent first message)들을 갖는 제1 메시지 엔벨로프를 생성하는 단계 - 상기 복수의 상호종속적 제1 메시지들은 상기 제1 표준에 따라 특정됨 -; 및

상기 제2 레거시 표준에 따라 동작하는 제2 메시지 타깃에 릴레이될 하나의 제2 메시지를 생성하는 단계 - 상기 제2 메시지는 상기 제2 레거시 표준에 따라 특정됨 -

를 더 포함하는, 메시지 프로세싱 방법.

**청구항 5**

제4항에 있어서,

상기 제1 메시지 엔벨로프와 상기 제1 메시지 타깃에 대해 이미 계류 중인(pending) 다른 메시지 엔벨로프들 사이의 상기 메시지 엔벨로프 종속성을 연산하는 단계 - 상기 엔벨로프 종속성은 상기 제1 메시지 타깃에 대한 메시지 엔벨로프들의 순서를 정의함 -;

적어도 상기 제1 메시지 타깃 및 상기 제1 메시지들의 기능들에 기초하여 상기 제1 메시지 엔벨로프의 상기 제1 메시지들 사이의 메시지 종속성을 연산하는 단계 - 상기 메시지 종속성은 각각의 메시지 엔벨로프 내의 메시지들의 순서를 정의함 -; 및

상기 메시지 엔벨로프 종속성 및 상기 메시지 종속성에 기초하여 순차적으로 상기 제1 메시지들을 상기 제1 메시지 타깃에 전송하는 단계

를 더 포함하는, 메시지 프로세싱 방법.

**청구항 6**

제4항에 있어서,

상기 제2 레거시 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지를 생성하는 단계 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 -; 및

다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이 아니라는 결정에 응답하여 상기 제3 메시지에 대한 메시지 종속성을 연산하는 일 없이 상기 제3 메시지를 전송하는 단계

를 더 포함하는, 메시지 프로세싱 방법.

**청구항 7**

제4항에 있어서,

상기 제2 레거시 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지를 생성하는 단계 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 -; 및

적어도 하나의 다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이라는 결정에 응답하여, 적어도 상기 메시지 타깃 및 상기 제3 메시지의 기능에 기초하여 상기 제3 메시지의 메시지 종속성 - 상기 메시지 종속성은 상기 제3 메시지의 메시지 타깃에 대한 메시지들의 순서를 정의함 - 을 연산하고, 상기 메시지 엔벨로프 종속성 및 상기 메시지 종속성에 기초하여 순차적으로 상기 적어도 하나의 다른 메시지 및 상기 제3 메시지를 상기 제3 메시지의 메시지 타깃에 전송하는 단계

를 더 포함하는, 메시지 프로세싱 방법.

**청구항 8**

제1항에 있어서,

각각의 메시지 엔벨로프에 대해, 적어도 메시지의 표준 및 메시지 타입에 따라 상기 복수의 상호종속적 메시지들 각각에 주어진 우선순위 값을 할당하는 단계; 및

상기 할당된 우선순위 값에 기초하여 각각의 메시지 엔벨로프에 의한 메시지들 사이의 메시지 종속성을 연산하는 단계

를 포함하는, 메시지 프로세싱 방법.

**청구항 9**

제1항에 있어서,

메시지가 상기 메시지의 메시지 엔벨로프 내의 임의의 다른 메시지와는 독립적이고 상기 메시지의 메시지 엔벨로프가 임의의 다른 메시지 엔벨로프와는 독립적이라는 결정에 응답하여 상기 메시지를 전송하는 단계를 포함하는, 메시지 프로세싱 방법.

**청구항 10**

제9항에 있어서,

상기 전송된 메시지의 메시지 종속성 및 상기 메시지의 메시지 엔벨로프 내의 메시지 순서에서의 다음 메시지 - 존재하는 경우 - 의 메시지 종속성을 업데이트하는 단계를 더 포함하는, 메시지 프로세싱 방법.

**청구항 11**

제9항에 있어서,

상기 전송된 메시지가 상기 전송된 메시지의 메시지 엔벨로프 내의 메시지 순서에서 마지막 메시지였다는 결정에 응답하여 상기 전송된 메시지의 메시지 엔벨로프의 메시지 엔벨로프 종속성을 업데이트하고 메시지 엔벨로프들의 순서에서의 다음 메시지 엔벨로프 - 존재하는 경우 - 의 엔벨로프 종속성을 업데이트하는 단계를 더 포함하는, 메시지 프로세싱 방법.

**청구항 12**

복수의 메시지 타깃들을 갖는 분산 메시징 시스템에서 메시지를 프로세싱하기 위한 시스템에 있어서,

상기 복수의 메시지 타깃들 중 적어도 하나의 메시지 타깃은 제1 표준에 따라 동작하고, 상기 복수의 메시지 타깃들 중 적어도 하나의 메시지 타깃은 제2 레거시 표준에 따라 동작하고,

상기 메시지 프로세싱 시스템은,

하나 이상의 프로세서; 및

상기 하나 이상의 프로세서에 결합되는 메모리를 포함하고,

상기 메모리는 상기 하나 이상의 프로세서에 의해 실행될 때, 상기 메시지 프로세싱 시스템으로 하여금,

복수의 데이터베이스 커밋 표시들 - 각각의 데이터베이스 커밋 표시는 데이터베이스 레코드의 변경을 표시함 - 을 수신하고;

상기 복수의 데이터베이스 커밋 표시들 중 한 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 상기 제1 표준에 따라 동작하는 통지받을 각각의 메시지 타깃에 대한 메시지 엔벨로프 및 각각의 메시지 엔벨로프 내의 적어도 하나의 메시지를 생성하고;

동일한 메시지 타깃에 대한 생성된 메시지 엔벨로프들 사이의 메시지 엔벨로프 종속성 - 상기 메시지 엔벨로프 종속성은 상기 동일한 메시지 타깃에 대한 메시지 엔벨로프들의 순서를 정의함 - 을 연산하고;

적어도 각각의 메시지 엔벨로프의 메시지들의 표준 및 상기 메시지들의 기능들에 기초하여 각각의 메시지 엔벨로프 내의 메시지들 사이의 메시지 종속성 - 상기 메시지 종속성은 각각의 메시지 엔벨로프 내의 상기 메시지들의 순서를 정의함 - 을 연산하고;

상기 메시지 엔벨로프 종속성 및 상기 메시지 종속성에 기초하여 순차적으로 상기 메시지들을 각각의 메시지 타깃에 전송하게 하는

프로그램 코드를 저장한, 메시지 프로세싱 시스템.

**청구항 13**

제12항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

상기 제1 표준에 따라 동작하는 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제1 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 메시지 엔벨로프를 생성하게 하는, 메시지 프로세싱 시스템.

**청구항 14**

제13항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

각각의 메시지 엔벨로프 내의 적어도 하나의 메시지를 생성하게 하는, 메시지 프로세싱 시스템.

**청구항 15**

제14항에 있어서, 상기 메시지 프로세싱 시스템으로 하여금,

상기 제1 표준에 따라 동작하는 제1 메시지 타깃에 릴레이될 복수의 상호종속적 제1 메시지들 - 상기 복수의 상호종속적 제1 메시지들은 상기 제1 표준에 따라 특정됨 - 을 갖는 제1 메시지 엔벨로프를 생성하고;

상기 제2 레거시 표준에 따라 동작하는 제2 메시지 타깃에 릴레이될 하나의 제2 메시지 - 상기 제2 메시지는 상기 제2 레거시 표준에 따라 특정됨 - 를 생성하게 함으로써,

상기 프로그램 코드는 실행시 상기 메시지 프로세싱 시스템으로 하여금 각각의 메시지 엔벨로프 내의 적어도 하나의 메시지를 생성하게 하는, 메시지 프로세싱 시스템.

**청구항 16**

제15항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

상기 제1 메시지 엔벨로프와 상기 제1 메시지 타깃에 대해 이미 계류 중인 다른 메시지 엔벨로프들 사이의 메시지 엔벨로프 종속성 - 상기 엔벨로프 종속성은 상기 제1 메시지 타깃에 대한 메시지 엔벨로프들의 순서를 정의함 - 을 연산하고;

적어도 상기 제1 메시지 타깃 및 상기 제1 메시지들의 기능들에 기초하여 상기 제1 메시지 엔벨로프의 상기 제1 메시지들 사이의 메시지 종속성 - 상기 메시지 종속성은 각각의 메시지 엔벨로프 내의 메시지들의 순서를 정의함 - 을 연산하고;

상기 메시지 엔벨로프 종속성 및 상기 메시지 종속성에 기초하여 순차적으로 상기 제1 메시지들을 상기 제1 메시지 타깃에 전송하게 하는, 메시지 프로세싱 시스템.

**청구항 17**

제15항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

상기 제2 레거시 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 - 를 생성하고;

다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이 아니라는 결정에 응답하여 상기 제3 메시지에 대한 메시지 종속성을 연산하는 일 없이 상기 제3 메시지를 전송하게 하는, 메시지 프로세싱 시스템.

**청구항 18**

제15항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

상기 제2 레거시 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 - 를 생성하고;

적어도 하나의 다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이라는 결정에 응답하여, 적어도 상기 메시지 타깃 및 상기 제3 메시지의 기능에 기초하여 상기 제3 메시지의 메시지 종속성 - 상기 메시지 종속성은 상기 제3 메시지의 메시지 타깃에 대한 메시지들의 순서를 정의함 - 을 연산하고, 상기 메시지 엔벨로프 종속성 및 상기 메시지 종속성에 기초하여 순차적으로 상기 적어도 하나의 다른 메시지 및 상기 제3 메시지를 상기 제3 메시지의 메시지 타깃에 전송하게 하는, 메시지 프로세싱 시스템.

**청구항 19**

제12항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

각각의 메시지 엔벨로프에 대해, 적어도 메시지의 표준 및 메시지 타입에 따라 상기 복수의 상호종속적 메시지들 각각에 주어진 우선순위 값을 할당하고;

상기 할당된 우선순위 값에 기초하여 각각의 메시지 엔벨로프에 의한 메시지들 사이의 메시지 종속성을 연산하게 하는, 메시지 프로세싱 시스템.

**청구항 20**

제12항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

메시지가 상기 메시지의 메시지 엔벨로프 내의 임의의 다른 메시지와는 독립적이고 상기 메시지의 메시지 엔벨로프가 임의의 다른 메시지 엔벨로프와는 독립적이라는 결정에 응답하여 상기 메시지를 전송하게 하는, 메시지 프로세싱 시스템.

**청구항 21**

제20항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

상기 전송된 메시지의 메시지 종속성 및 상기 메시지의 메시지 엔벨로프 내의 메시지 순서에서의 다음 메시지 – 존재하는 경우 – 의 메시지 종속성을 업데이트하게 하는, 메시지 프로세싱 시스템.

**청구항 22**

제20항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

상기 전송된 메시지가 상기 전송된 메시지의 메시지 엔벨로프 내의 메시지 순서에서 마지막 메시지였다는 결정에 응답하여 상기 전송된 메시지의 메시지 엔벨로프의 메시지 엔벨로프 종속성을 업데이트하고 메시지 엔벨로프들의 순서에서의 다음 메시지 엔벨로프 – 존재하는 경우 – 의 엔벨로프 종속성을 업데이트하게 하는, 메시지 프로세싱 시스템.

**청구항 23**

복수의 메시지 타깃들을 갖는 분산 메시징 시스템에서 메시지들을 프로세싱하기 위한, 비일시적 컴퓨터 판독 가능 저장 매체에 저장된 컴퓨터 프로그램에 있어서,

상기 복수의 메시지 타깃들 중 적어도 하나의 메시지 타깃은 제1 표준에 따라 동작하고, 상기 복수의 메시지 타깃들 중 적어도 하나의 메시지 타깃은 제2 레거시 표준에 따라 동작하고,

상기 컴퓨터 프로그램은,

하나 이상의 프로세서에 의해 실행될 때, 상기 하나 이상의 프로세서로 하여금,

복수의 데이터베이스 커밋 표시들 – 각각의 데이터베이스 커밋 표시는 데이터베이스 레코드의 변경을 표시함 – 을 수신하고;

상기 복수의 데이터베이스 커밋 표시들 중 한 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 상기 제1 표준에 따라 동작하는 통지받을 각각의 메시지 타깃에 대한 메시지 엔벨로프 및 각각의 메시지 엔벨로프 내의 적어도 하나의 메시지를 생성하고;

동일한 메시지 타깃에 대한 생성된 메시지 엔벨로프들 사이의 메시지 엔벨로프 종속성 – 상기 메시지 엔벨로프 종속성은 상기 동일한 메시지 타깃에 대한 메시지 엔벨로프들의 순서를 정의함 – 을 연산하고;

적어도 각각의 메시지 엔벨로프의 메시지들의 표준 및 상기 메시지들의 기능들에 기초하여 각각의 메시지 엔벨로프 내의 메시지들 사이의 메시지 종속성 – 상기 메시지 종속성은 각각의 메시지 엔벨로프 내의 상기 메시지들의 순서를 정의함 – 을 연산하고;

상기 메시지 엔벨로프 종속성 및 상기 메시지 종속성에 기초하여 순차적으로 상기 메시지들을 각각의 메시지 타깃에 전송하게 하는

상기 비일시적 컴퓨터 판독 가능 저장 매체에 저장된 프로그램 코드를 포함하는, 비일시적 컴퓨터 판독 가능 저

장 매체에 저장된 컴퓨터 프로그램.

## 발명의 설명

### 기 술 분 야

[0001] 본 발명은 일반적으로 메시지 프로세싱에 관한 것이다. 더 구체적으로는, 본 발명은 메시지들을 프로세싱하여 메시지들 사이의 종속성(dependency)들 및 메시지 순서를 관찰하는 것에 관한 것이다.

### 배 경 기 술

[0002] 다양한 메시지 프로세싱 시스템들이 종래 기술에 알려져 있다.

[0003] US 7,240,097 B2호는 병렬 프로세싱을 위해 메시지들의 순서를 보존하는 것이 가능한 시스템을 설명한다. 메시지 시스템은 일단 수신된 즉시 입력 메시지들을 프로세싱하고, 그 메시지들을 목적지들로 전송할 때 입력 메시지들의 순서를 보존한다.

[0004] US 8,200,765 B2호는, 메시지들의 중요도 값에 기초하여 메시지들이 출력 큐 내에 놓여지는, 즉, 덜 중요한 메시지들보다 더 중요한 메시지들이 선호되는 중앙집중식 시스템을 설명한다. 입력 및 출력 큐들 양측 모두가 채용된다. 제안된 시스템은 중앙집중식 데이터베이스 커넥션을 통해 분산/병렬 시스템 방식으로 배치되고, 여기서 출력 큐들은 각각의 팜 노드 서버(farm node server)에서 채용된다. 메시지 순서화는 메시지들이 출력 큐 내에 놓여지기 전에 연산(compute)된다.

[0005] US 2007/0124398 A1호는 할당된 시퀀스 식별자에 따라 메시지들을 수신자들에게 전송하는 것이 가능하다. 제안된 방법은 하나의 엔벨로프(envelope)에 대해 생성되는 메시지들을 우선순위화할 수 있고, 엔벨로프들은 시퀀스 식별자들을 사용함으로써 서로 연결된다.

[0006] US 7,680,793 B2호는 커밋 타임스탬프(commit timestamp)에 기초하여 메시지를 순서대로 전송한다. 메시지들은 비즈니스 값 및 커밋 타임스탬프 양측 모두에 기초하여 순서대로 전송된다.

[0007] US 8,875,155 B2호는 메시지들의 그룹들에 대한 메시지 순서들을 관리한다.

[0008] EP 2 811 437호는 들어오는 메시지들, 특히 HTTP(Hypertext Transfer Protocol), SOAP(Simple Object Access Protocol) 또는 XML(Extensible Markup Language) 기반 메시지들에 대해 시퀀싱하기 위한 SOA(Service Oriented Architecture) 시스템을 제안한다.

[0009] 게다가, 텔레타이프(teletype) 또는, 간단히, TTY 메시지들과 같은 특정 메시지 표준들이 알려져 있다. 예를 들어, EP 2254046 B1호는 TTY 메시지들의 전송 순서를 결정하기 위해 TTY 메시지들의 메시지 시퀀싱을 설명한다.

[0010] 더욱이, 이하 NDC라고 간단히 지칭되는 "신규 분배 능력(New Distribution Capability)" 표준이 2015년에 국제 항공 운송 협회(International Air Transport Association)(IATA)에 의해 배포되었다. NDC는 항공사들과 전역 분배 시스템들(global distribution systems)(GDS) 사이의 XML 기반 데이터 송신 표준의 개발 및 시장 채택을 위한 항공사 분배 포맷이다. 예를 들어, NDC는 항공사들이 여행사들로부터의 쇼핑 요청들에 직접 응답하여, 주문 프로세싱을 관리하는 것을 가능하게 하고 해당하는 비교 쇼핑을 가능하게 한다. NDC는 시스템 또는 소프트웨어 또는 데이터베이스가 아니라, 항공사들과 여행사들 사이에서 정보를 교환하기 위한 표준이다.

## 발명의 내용

[0011] 레거시 포맷(legacy format)에 따른 메시지들 및 레거시 포맷의 후속자(successor), 즉, 신규 포맷에 따른 메시지 양측 모두를 프로세싱하는 동안, 양측 모두의 포맷들에 따른 메시지들이 올바른 메시지 순서로 수신자들에게 전송된다는 것을 보장하는 것이 가능한 메시지 프로세싱 시스템을 제공하는 것이 목적이다. 구체적으로는, TTY와 같은 레거시 포맷 및 NDC 표준에 따른 XML 기반 메시지를 단일화된 방식으로 프로세싱할 수 있는 메시지 프로세싱 시스템을 제공하는 것이 목적이다.

[0012] NDC는 하나의 데이터베이스 변경 커밋이 다수의 메시지들을 하나의 메시지 타깃으로 트리거하는 것을 가능하게 한다. 이들 메시지들은 상이한 타입들일 수도 있고, 상이한 전송 방법들로 전송될 수도 있고 특정 종속성 및 인과관계를 가질 수도 있다. 따라서, NDC 메시지 흐름들을 지원하고, 동시에, 보다 단순한 기능성들을 갖는 레거시 메시지 시스템들을 지원함에 있어서 하위 호환가능(backward compatible)하여, 메시지 흐름들의 타입들 양측 모두가, 2개의 독립적인 메시지 시스템들과는 대조적으로, 통합된 그리고 단일화된 방식으로 지원되는 메시지

시스템을 제공하는 것이 목적이다.

[0013] 게다가, 개방적이고 확장가능하고 따라서 NDC 표준보다 훨씬 더 광범위한 기능성들을 지원할 수도 있는 장래의 메시지 표준들을 지원할 수도 있는 메시지 시스템을 제공하는 것이 목적이다.

[0014] 이들 목적들은 독립 청구항들에 의해 다루어지고 해결된다.

[0015] 제1 양태에 따르면, 메시지 프로세싱 엔진에 의해 분산 메시징 시스템에서 메시지들을 프로세싱하기 위한 방법이 제공된다. 복수의 메시지 타깃들 중 적어도 하나의 메시지 타깃은 제1 표준에 따라 동작하고, 복수의 메시지 타깃들 중 적어도 하나의 메시지 타깃은 제2 레거시 표준에 따라 동작한다. 이 방법은 메시지 프로세싱 엔진에 의해 수행된다.

[0016] 복수의 데이터베이스 커밋 표시들이 수신되고, 여기서 각각의 데이터베이스 커밋 표시는 데이터베이스 레코드의 변경을 표시한다. 복수의 데이터베이스 커밋 표시들 중 한 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 제1 표준에 따라 동작하는 통지받을 각각의 메시지 타깃에 대한 메시지 엔벨로프가 생성되고 각각의 메시지 엔벨로프 내의 적어도 하나의 메시지가 생성된다.

[0017] 동일한 메시지 타깃에 대한 생성된 메시지 엔벨로프들 사이의 메시지 엔벨로프 종속성들이 연산된다. 메시지 엔벨로프 종속성들은 동일한 메시지 타깃에 대한 메시지 엔벨로프들의 순서를 정의한다.

[0018] 각각의 메시지 엔벨로프의 메시지들의 표준 및 메시지들의 기능들에 적어도 기초하여 각각의 메시지 엔벨로프 내의 메시지들 사이의 메시지 종속성들이 연산된다. 메시지 종속성들은 각각의 메시지 엔벨로프 내의 메시지들의 순서를 정의한다.

[0019] 메시지 엔벨로프 종속성들 및 메시지 종속성들에 기초하여 메시지들이 순차적으로 각각의 메시지 타깃에 전송된다.

[0020] 제2 양태에 따르면, 상기의 방법을 구현하도록 배열되는 메시지 시스템이 제공된다.

[0021] 제3 양태에 따르면, 컴퓨터 상에서 실행될 때 상기의 방법을 구현하는 컴퓨터 프로그램 제품이 제공된다.

[0022] 추가 양태들이 종속 청구항들에 의해 제시된다.

## *도면의 간단한 설명*

[0023] 본 발명은 첨부 도면들을 참조하여 설명될 것이다. 유사한 참조 번호들은 동일한 또는 기능적으로 유사한 요소들을 일반적으로 표시한다.

도 1, 도 2, 도 3 및 도 4는 본 명세서에서 설명되는 기능-레벨 메시지 프로세싱 방법에 대한 흐름도들이다.

도 5는 본 명세서에서 설명되는 메시지 프로세싱 시스템에 의해 정의된 바와 같은 메시지 엔벨로프 레벨에서의 그리고 메시지 레벨에서의 종속성들을 시각화한다.

도 6은 메시지 프로세싱 시스템의 예시적인 프로세스 지향 뷰(process-oriented view)이다.

도 7은 메시지 프로세싱 시스템에 대한 예시적인 하드웨어 아키텍처를 도시한다.

도 8 및 도 9는 메시지 및 종속성 프로세싱의 구현예에 대한 더 상세한 흐름도들이다.

도 10은 메시지 전송 프로세스의 구현예에 대한 상세한 흐름도이다.

도 11, 도 12 및 도 13은 메시지 엔벨로프들과 메시지들 그리고 메시지 엔벨로프들과 메시지들 사이의 종속성들을 유지하는 테이블의 상이한 상태들을 도시한다.

도 14는 요청 클라이언트 및/또는 데이터베이스 테이블의 내부 아키텍처의 예시적인 개략도이다.

## *발명을 실시하기 위한 구체적인 내용*

[0024] 도 6 내지 도 14와 관련하여 더 구체적인 구현예를 설명하기 전에, 도 1 내지 도 5를 참조하여 몇 가지 더 일반적인 양태들이 먼저 논의될 것이다.

[0025] 일반적으로, 메시지 타깃들에게 전송될 메시지들의 순서는 메시지 프로세싱 시스템에서 다루어질 일 양태이다. 예를 들어, GDS에서의 정보 흐름들에 대해 예약 메시지들의 올바른 순서가 유지되어야 한다. 예를 들어, 예약

시스템의 데이터베이스에서의 부킹(booking)(예약)은 하나 또는 수 개의 변경 커밋들에서 수정된다. 수신자는 시스템 전체의 데이터 일관성을 유지하기 위해 메시지 교환의 형태로, 즉, 이들 변경 커밋들의 형태로 순서대로 부킹 수정의 대응하는 통지들을 수신하기로 되어 있다. 변경 커밋들의 순서가 변경되는 경우, GDS 인스턴스들 및/또는 항공사 시스템들 간의 레코드들 사이의 불일치 또는 비일관성이 발생할 것이다. 예를 들어, 여행사는, 2개의 후속 변경 커밋들에서, 비행 취소를 행한 후에 상이한 날짜에 대해 동일한 비행을 재부킹한다. 고객, 즉, 항공사는 올바른 동기화를 갖기 위해, 취소 메시지를 먼저 수신한 후에만 비행 재부킹을 위한 메시지를 수신할 필요가 있다. 그에 따라, 가능한 레코드 불일치를 회피하기 위해, 메시지 타깃에 대한 올바른 메시지 시퀀스는 부킹 소스와 부킹 타깃 사이에서 예약이 일관되게 유지되도록 보증한다.

[0026] GDS 예약 동기화 흐름들과 같은 오늘날의 메시징 흐름들에서는, 통상적으로 - 하나의 메시지 타깃에 대해 - 부킹 레코드의 변경 커밋당 최대 하나의 TTY 메시지(PNR이라고도 알려진 승객 이름 레코드(passenger name record)에 대한 소위 트랜잭션 종료 엔벨로프(envelope of end-of-transaction))가 생성되어 GDS 노드들 사이의 수정들을 교환한다. 전통적인 TTY 프레임워크에 대한 전용된 방법이 EP 2254046 B1호에 의해 제안되었다. 그러나, NDC 예약들의 맥락에서, 시나리오가 변경되고 있는데, 이는, NDC 표준들의 구현들이, NDC 사양에 따라 동작하고 따라서 NDC 피처(feature)들을 구현하는 메시지 타깃들에 대한 하나의 예약 레코드 변경 커밋에 대해 다수의 메시지들을 통상적으로 트리거할 것으로 예상되기 때문이다. 이들 다수의 메시지들은 상이한 메시지 타입들(TTY, XML, 또는 심지어 장래의 다른 타입들), 상이한 메시지 기능들(예컨대, 부가 또는 삭제 메시지들), 전송 방법들(동기식: 메시지가 전송되고 확인 응답이 동일한 세션에서 수신된 후에, 메시지 전달이 성공적인 것으로 알려진 후에 세션이 닫힌다; 또는 비동기식: 세션이 메시지가 전송된 전송자 측에서 닫히고, 확인 응답이 다른 세션을 통해 전송자에게 다시 전송된다)을 가질 수도 있고, 이들은 다양한 송신 및 네트워크 프로토콜들(예컨대, HTTP/SOAP) 그리고 또한 메시지들 사이의 관계들을 정의하는 전송 우선순위들을 채용할 수도 있다. 또한, 레거시 TTY 메시지 흐름들은 여전히 메시지 프로세싱 시스템들에 의해 유지되어야 한다, 즉, 메시지 프로세싱 시스템들은 하위 호환가능해야 하는데, 이는 예약 레코드들을 동기화시키기 위해 TTY 흐름들 및 NDC 흐름들이 공존할 것이기 때문이다. 강건성(robustness), 하위 호환성을 달성하고 안정성을 유지하기 위해, 신규 메시지 프로세싱 메커니즘이 따라서 요구된다.

[0027] 요약하면, 이 신규 메시지 프로세싱 메커니즘은, 서로에 대한 종속 관계를 가지며 따라서 잘 정의된 순서로 메시지 타깃에 전송될 필요가 있는 다수의 메시지들이 하나의 변경 커밋에 대해 트리거되는 NDC 예약들의 맥락에서 메시지 시퀀스들의 문제를 다루어야 한다. 신규 메시지 프로세싱은 또한, TDC, XML 및 다른 메시지 타입들이 사용되는 하이브리드 모드에서 부킹 레코드를 동기화시키는 하나의 일반 메시지 프로세싱 프레임워크로 NDC 메시지 흐름들 및 레거시 TTY 흐름들을 통합시키는 문제를 다루어야 한다.

[0028] 이들 문제들을 다루기 위해, 레거시 메시지 흐름들 및 NDC 메시지 흐름들을 2개의 별개의 그리고 독립적인 메시지 프로세싱 시스템들에 의해 핸들링하는 것과는 대조적으로, 통합된 방식으로 메시지들을 프로세싱하고 메시지 순서들을 관리하는 것이 제안된다. 도 1을 참조하면, 메시지들은 메시지 프로세싱 엔진에 의해 분산 메시징 시스템에서 프로세싱된다.

[0029] 분산 메시징 시스템은 다수의 타입들의 메시지 타깃들을 포함한다. 복수의 메시지 타깃들 중 적어도 하나의 메시지 타깃은 제1 표준에 따라 동작하고, 복수의 메시지 타깃들 중 적어도 하나의 메시지 타깃은 제2 레거시 표준에 따라 동작한다. 본 명세서에서 사용되는 바와 같은 메시지 타깃이라는 용어는, 대형 시스템들 예컨대 전역 분배 시스템(GDS), 출국 관리 시스템(Departure Control System)(DCS), 개별 수신자 스테이션들뿐만 아니라, 항공사와 같은 특정 조직의 로컬 영역 네트워크(LAN) 또는 광역 네트워크(WAN)를 포함하는, 다양한 레벨들에서의 메시지 목적지들을 포함한다.

[0030] 일부 실시예들에서, 제1 표준에 따라 동작하는 메시지 타깃들은 완전히 하위 호환가능하고, 따라서 또한 제2 표준에 따라 동작하는데, 즉, 제2 표준에 따라 메시지들을 또한 프로세싱하도록 배열되는 한편, 제2 표준에 따라 동작하는 메시지 타깃들은, 제1 표준을 구현하지 않고 따라서 제2 표준에 따라 메시지들을 프로세싱하는 것이 불가능한 레거시 메시지 타깃들이다. 따라서, 일반적으로, 본 명세서에서 언급되는 제1 표준에 대한 언급은 실제로 제1 표준 및 제2 표준 그리고 제1 표준 및 제2 표준에 따른 각각의 메시지들을 포괄하는 한편, 본 명세서에서 언급되는 제2 표준에 대한 언급은 단지 제2 표준에 따른 메시지들 및 메시지 타깃들만을 포괄한다.

[0031] 다시 도 1을 참조하면, 2에서, 복수의 데이터베이스 커밋 표시들이 메시지 프로세싱 엔진에 의해 수신된다. 각각의 데이터베이스 커밋 표시는 데이터베이스 레코드의 변경을 표시한다. 데이터베이스 커밋 표시는, 메시지 프로세싱 엔진에 대해 상류에 위치되는 데이터베이스에서의 데이터베이스 레코드에 대한 변경들을 마무리하는 데

이터베이스 커밋과 리트리빙 쿼리 사이에서 승객 이름 레코드(PNR)와 같은 데이터베이스 레코드에 대해 행해진 모든 변경들의 컨테이너이다. 따라서, 상류 데이터베이스에서 변경이 커밋될 때마다, 데이터베이스 변경 커밋 표시가 생성되고 메시지 프로세싱 엔진에 게시된다. 따라서, 데이터베이스 커밋 표시들은 메시지 프로세싱 엔진에 대한 입력이다.

[0032] 4에서, 복수의 데이터베이스 커밋 표시들 중 한 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 제1 표준에 따라 동작하는 통지받을 각각의 메시지 타깃에 대한 메시지 엔벨로프가 생성되고 각각의 메시지 엔벨로프 내의 적어도 하나의 메시지가 생성된다. 본 명세서에 사용되는 바와 같은 메시지 엔벨로프는, 전송되지 않지만 메시지 엔벨로프의 메시지들을 그룹화하는 레퍼런스를 구성하는 메시지의 특수 인스턴스로서 간주된다. 이것은 디렉토리 또는 폴더가 디렉토리에 위치되는 모든 파일을 그룹화하는 파일의 특수 인스턴스인 파일 시스템들과 유사하다. 메시지 엔벨로프 및 그의 메시지들의 세트는 이하 메시지 리스트라고도 또한 지칭된다.

[0033] 일반적으로, 메시지 프로세싱 엔진은 수신된 데이터베이스 커밋 표시에 의해 영향받은 메시지 타깃들을 결정하고, 제1 표준에 따라 동작하는 통지받아야 하는 각각의 영향받은 메시지 타깃에 대한 메시지 엔벨로프를 생성한다. 메시지 엔벨로프 내의 모든 메시지들은 이 메시지 타깃에 어드레싱되고 잘 정의된 순서로 메시지 타깃에 전송될 것이다. 전송의 순서를 결정 및 보장하기 위해, 2개의 상이한 레벨들에서의 종속성들, 즉, 메시지 엔벨로프들의 레벨에서의 메시지 엔벨로프 종속성 및 메시지 레벨에서의 메시지 레벨 종속성이 여기에 도입된다. 메시지 엔벨로프 종속성들은 특정 메시지 타깃에 대해 생성되는 모든 메시지 엔벨로프들 간의 순서를 정의한다. 메시지 종속성들은 특정 메시지 엔벨로프 내의 메시지의 순서를 정의한다.

[0034] 따라서, 메시지 엔벨로프들 및 그 메시지 엔벨로프들 내의 메시지들의 생성은 또한, 6에서의 동일한 메시지 타깃에 대한 생성된 메시지 엔벨로프들 사이의 메시지 엔벨로프 종속성들의 연산, 그리고 8에서의 각각의 메시지 엔벨로프 내의 메시지들 사이의 메시지 종속성들의 연산을 수반한다.

[0035] 일반적으로, 데이터베이스 커밋 표시에 대해 생성된 메시지 엔벨로프가 6에서의 동일한 메시지 타깃에 대해 이미 존재하는 다른 메시지 엔벨로프들, 즉, 보다 앞선 데이터베이스 커밋 메시지의 수신에 응답하여 이전에 그 메시지 타깃에 대해 이미 생성된 메시지 엔벨로프들에 종속되도록 연산된다. 따라서, 특정 메시지 타깃에 대한 메시지 엔벨로프들은 특정 메시지 타깃에 영향을 미치는 데이터베이스 커밋 메시지들의 수신 순서에 기초하여 순서화된다.

[0036] 다른 한편으로, 8에서 각각의 메시지 엔벨로프의 메시지들의 표준 및 메시지들의 기능들에 적어도 기초하여 메시지 종속성들이 연산된다. 특히, 제1 표준의 메시지들은, 레코드를 부가하는 부가 메시지들, 레코드를 삭제하는 삭제 메시지들, 레코드를 분할하는 분할 메시지들 등과 같은 상이한 기능들(아래에서는 "메시지 이름"이라고도 또한 지칭됨)로 분류될 수 있다.

[0037] 특정 메시지 엔벨로프들과 메시지들 사이에 확립되고 특정 시점에서 유효한 관계들이 메시지 종속성 맵(message dependency map)이라고도 또한 지칭된다. 메시지 종속성 맵들은 하나 이상의 데이터베이스 테이블들에서 메시지 프로세싱 엔진에 의해 유지된다. 10에서 메시지 엔벨로프 종속성들 및 메시지 종속성들에 기초하여 메시지들이 메시지 프로세싱 엔진에 의해 각각의 메시지 타깃에 순차적으로 전송된다.

[0038] 상이한 타입들의 메시지 타깃들에 부가적으로, 데이터베이스 커밋 표시들을 생성하는 상이한 타입들의 데이터베이스 시스템들이 분산 메시징 시스템에 존재할 수도 있다. 하나의 타입의 데이터베이스 시스템들은 제1(신규) 표준에 따라 동작하고, 이미 제1 표준을 준수하는 데이터베이스 커밋 표시들을 (그리고 하위 호환성의 이유로 제2 표준을 준수하는 데이터베이스 커밋 표시들을 또한) 출력할 수도 있다. 다른 타입의 데이터베이스 시스템들은 단지 제2 레거시 표준에 따라서만 여전히 동작하고 따라서 단지 레거시 데이터베이스 커밋 표시들만을 출력한다.

[0039] 데이터베이스 커밋 메시지의 소스(제1 표준에 따라 동작하는 데이터베이스 시스템 또는 제2 레거시 표준에 따라 동작하는 데이터베이스 시스템)에 따라 그리고 통지받을 메시지 타깃의 타입(제1 표준에 따라 동작하는 메시지 타깃 또는 제2 레거시 표준에 따라 동작하는 메시지 타깃)에 따라, 상이한 경우들의 메시지 프로세싱이 차별화될 수 있다.

[0040] 도 2에 의해 시각화되는 제1 경우는, 11에서 수신된 데이터베이스 커밋 표시가, 제1 표준에 따라 동작하는 데이터베이스 시스템으로부터의 복수의 데이터베이스 커밋 표시들 중 제1 데이터베이스 커밋 표시인 것으로 가정한다. 일부 실시예들에서, 이 데이터베이스 커밋 표시는 제1 표준에 따른 데이터베이스 커밋 표시로서, 예를 들어, 아래에 추가로 설명되는 바와 같은 플래그에 의해, 마킹된다. 제1 표준에 따른 데이터베이스 커밋 표시를

수신하는 것에 응답하여, 메시지 프로세싱 엔진은 통지받을 제1 표준에 따라 동작하는 각각의 메시지 타깃에 대한 메시지 엔벨로프를 생성하고 각각의 메시지 엔벨로프 내의 적어도 하나의 메시지를 생성한다.

[0041] 여기서, 2개의 상이한 하위 경우들이 구별될 수 있다. 간단히 제1 메시지 타깃들이라고 지칭되는, 제1 표준을 구현하는 통지받을 각각의 메시지 타깃에 대해, 12에서 각각의 제1 메시지 엔벨로프가 생성되는데, 그 각각은 통상적으로, 제1 표준에 따라 동작하는 각각의 제1 메시지 타깃에 릴레이될 복수의 상호종속적 제1 메시지(interdependent first message)들을 포함한다. 복수의 상호종속적 제1 메시지들은 (일반적으로 또한, 상기에 이미 설명된 바와 같이 하위 호환성의 이유로 제2 표준에 따른 제2 메시지들에 따른 메시지들을 포괄하는) 제1 표준에 따라 특정된다. 간단히 제2 메시지 타깃들이라고 지칭되는, 제2 레거시 표준을 구현하는 통지받을 각각의 메시지 타깃에 대해, 13에서 제2 표준에 따라 동작하는 제2 메시지 타깃에 릴레이될 (메시지 타깃당 하나의) 제2 메시지가 생성된다. 제2 메시지는 제2 레거시 표준에 따라 특정된다.

[0042] 그 후에, 메시지 프로세싱 엔진은 14에서 제1 메시지 엔벨로프와 제1 메시지 타깃에 대해 이미 계류 중인(pending) 다른 메시지 엔벨로프들 사이의 메시지 엔벨로프 종속성들을 연산한다. 엔벨로프 종속성들은 제1 메시지 타깃에 대한 메시지 엔벨로프들의 순서를 정의한다.

[0043] 메시지 프로세싱 엔진은, 15에서 제1 메시지 타깃 및 제1 메시지들의 기능들에 적어도 기초하여 제1 메시지 엔벨로프의 제1 메시지들 사이의 메시지 종속성들을 추가로 연산한다. 메시지 종속성들은 각각의 메시지 엔벨로프 내의 메시지들의 순서를 정의한다.

[0044] 그 후에, 16에서 제1 메시지들은 메시지 엔벨로프 종속성들 및 메시지 종속성들에 기초하여 순차적으로 제1 메시지 타깃에 전송된다.

[0045] 제2 레거시 표준을 구현하는 메시지 타깃들에 대해 생성되는 제2 메시지들과 관련하여, 관련된 메시지 타깃에 대해 다른 메시지가 계류 중이 아닌 경우 이들 메시지들은 직접 전송되거나, 또는 관련된 메시지 타깃에 대한 다른 계류 중인 메시지들이 계류 중인 경우 메시지 레벨 종속성이 연산된다. 후자의 경우에, 메시지 레벨 종속성 연산에 따라 제2 메시지들이 순차적으로 전송된다.

[0046] 도 3에 의해 시각화되는 제2 경우는, 17에서 수신된 데이터베이스 커밋 표시가, 제2 레거시 표준에 따라 동작하는 데이터베이스 시스템으로부터의 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시인 것으로 가정한다. 일부 실시예들에서, 이 데이터베이스 커밋 표시는 제1 표준을 표시하는 마킹을 포함하지 않고, 따라서 메시지 프로세싱 시스템은 데이터베이스 커밋 표시를 제2 레거시 표준에 따른 표시인 것으로 인식한다.

[0047] 제2 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 메시지 프로세싱 엔진은, 18에서, 통지받을 각각의 메시지 타깃에 대해 하나의 메시지를 생성하고, 메시지 타깃에 대한 적어도 하나의 메시지 엔벨로프가 과거에 생성되었고 여전히 계류 중이라면, 각각의 메시지 타깃에 대한 메시지를 각각의 타깃에 대한 기존 메시지 엔벨로프와 잠재적으로 연관시킨다. 일반적으로, 제2 데이터베이스 커밋 메시지는 제2 레거시 표준에 따르기 때문에, 단지 제2 레거시 표준들에 따른 각각의 메시지들만이 메시지 프로세싱 엔진에 의해 생성된다. 다시 말해, 제2 레거시 표준에 따른 데이터베이스 커밋 표시는 제1 표준에 따른 메시지들의 생성 또는 메시지 엔벨로프들의 생성을 야기하지 않는다.

[0048] 제1 표준에 따라 동작하는 메시지 타깃들 및 제2 레거시 표준에 따라 동작하는 메시지 타깃들 양측 모두에 대해, 메시지 프로세싱 엔진은, 이하 제3 메시지라고 지칭되는, 제2 표준에 따라 포맷팅되는 하나의 메시지를 생성한다. 19에서 결정된 바와 같은, 어떠한 계류 중인 메시지 엔벨로프들 및/또는 메시지들도 없는 각각의 메시지 타깃에 대해, 20에서 제2 표준에 따라 생성된 제3 메시지가 어떠한 메시지 종속성 계산도 없이 즉시 전송된다.

[0049] 19에서 결정된 바와 같은, 계류 중인 메시지 엔벨로프 및/또는 적어도 하나의 계류 중인 메시지를 갖는 메시지 타깃들에 대해, 제2 레거시 표준에 따라 생성된 메시지는 일반적으로, 계류 중인 메시지 엔벨로프와 연관되거나 또는, 메시지 엔벨로프가 계류 중이 아닌 경우에는, 계류 중인 메시지들의 시퀀스에 포함된다. 이 경우에, 21에서, 메시지 프로세싱 엔진은 제1 메시지 타깃 및 제3 메시지가 제2 레거시 표준에 따른 메시지라는 사실에 적어도 기초하여 제3 메시지의 메시지 종속성을 연산한다. 연산된 메시지 종속성은 고려된 메시지 타깃에 대한 계류 중인 메시지 엔벨로프의 다른 메시지들의 시퀀스에서 제3 메시지의 위치를 정의한다.

[0050] 예를 들어, 고려된 메시지 타깃이 제1 표준에 따라 동작하는 경우, 즉, 그것이 제1 메시지 타깃인 경우, 제1 메시지들을 가진 제1 엔벨로프는 이미 계류 중일 수도 있다. 이 경우에, 제3 메시지는 일반적으로 메시지 엔벨로

프에 포함되고, 제1 메시지들을 제3 메시지와 관련시켜 메시지 종속성이 연산된다. 22에서, 제1 메시지들 및 제3 메시지는 메시지 엔벨로프 종속성들 및 메시지 종속성들에 기초하여 순차적으로 제1 메시지 타깃에 전송된다.

[0051] 일부 특정 경우들에서, 예를 들어, 제3 메시지가 제1 메시지 엔벨로프에서의 모든 다른 메시지들에 비해 가장 높은 우선순위를 갖는다고 간주되는 경우, 제3 메시지가 그러할 수 있는 한 메시지 종속성이 연산되지 않고 직접 전송된다.

[0052] 도 4를 참조하면, 선택적으로, 24에서 적어도 메시지의 표준 및 메시지 타입에 따라 복수의 상호종속적 메시지들 각각에 주어진 우선순위 값이 할당된다. 25에서 할당된 우선순위 값들에 기초하여 각각의 메시지 엔벨로프에 의한 메시지들 사이의 메시지 종속성이 연산된다. 우선순위 값들을 정의하는 테이블의 예가 아래에 추가로 주어질 것이다.

[0053] 선택적으로, 26에서, 메시지가 메시지의 메시지 엔벨로프 내의 임의의 다른 메시지와는 독립적이고 메시지의 메시지 엔벨로프가 임의의 다른 메시지 엔벨로프와는 독립적이라는 결정에 응답하여 메시지가 전송된다. 추가로, 선택적으로, 27에서, 전송된 메시지의 메시지 종속성 및 메시지의 메시지 엔벨로프 내의 메시지의 순서 내에서의 다음 메시지의 메시지 종속성이 업데이트된다. 아래에 더 상세히 설명되는 바와 같이, 이 업데이트는, 메시지 엔벨로프에서의 다음 메시지 - 존재하는 경우 - 가 독립적으로 되도록 그리고 따라서 전송될 준비가 되도록 렌더링할 수도 있다.

[0054] 전송된 마지막 메시지가 현재 메시지 엔벨로프의 마지막 메시지인 경우, 전송된 마지막 메시지의 메시지 엔벨로프의 메시지 엔벨로프 종속성이 업데이트되고 또한 메시지 엔벨로프들의 순서로 다음 메시지 엔벨로프의 메시지 엔벨로프 종속성이 업데이트된다. 이 업데이트는 전송된 마지막 메시지의 메시지 엔벨로프가 완전히 프로세싱되도록 그리고 다음 메시지 엔벨로프가 독립적으로 되도록 렌더링하고, 그 결과, 다음 메시지 엔벨로프의 제1 메시지도 독립적으로 되도록 그래서 전송될 준비가 되도록 렌더링한다.

[0055] 선택적으로, 제1 표준은 IATA의 신규 분배 능력(NDC)을 준수한다.

[0056] 일부 실시예들에서, 본 명세서에서 설명되는 방법론들은, 예를 들어, GDS 예약 동기화 시스템을 레거시 TTY 메시지들로부터 한 세트의 하이브리드 메시지들: TTY 메시지들, XML 메시지들 그리고 잠재적으로 다른 메시지들로 확장시킨다.

[0057] 높은 레벨 뷰의 메시지 프로세싱이 도 5에 의해 도시되어 있다. 예로서, 비행 세그먼트 "취소-재부킹" 시나리오가 예약 레코드 B의 변경 커밋 표시들(34)에서 발생하였다. 변경 커밋 표시들 B2 및 B3이 GDS 예약 프로세스(32)로부터 메시지 생성 및 시퀀싱 프로세스(42)에 의해 수신된다. 변경 커밋 표시 B2는 예약 레코드 B에 대한 비행 세그먼트 XX123의 취소를 표시하고, 변경 커밋 표시 B3은 예약 레코드 B에 대한 비행 세그먼트 XX 123의 부가를 표시한다. 메시지 생성 및 시퀀싱 프로세스(42)는, 다른 예약 레코드 A와 관련된 변경 커밋들 A0, A1, A2, 및 A3과 같은, GDS 예약 프로세스(32)에 의해 유지되는 다른 예약 레코드들에 대한 변경 커밋들(34)을 또한 수신한다.

[0058] 이상적으로, 변경 커밋 표시들(34)은 순서화된 방식으로 GDS 예약 프로세스(32)에 의해 출력된다. 그러나, 이것은, A0, A3, A2, A1의 순서로 GDS 예약 프로세스(32)에 의해 전송되는, 예약 레코드 A와 관련된 변경 커밋들에 대해 예시적으로 도시된 바와 같이, 항상 그렇지는 않을 수도 있다. 예를 들어, GDS 예약 프로세스(32)는, 상기에 설명된 바와 같이, 2개의 상이한 데이터베이스 시스템들, 즉, 제1 표준에 따라 동작하는(예를 들어, 예약 레코드 B를 관리하는) 데이터베이스 시스템 및 제2 레거시 표준에 따라 동작하는(예를 들어, 예약 레코드 A를 관리하는) 다른 데이터베이스 시스템으로부터 실제로 유래할 수도 있다.

[0059] 예약 레코드 B에 관한 예시적인 "취소-재부킹" 흐름의 변경 커밋 표시들(34)을 (다른 예약 레코드들과 관련된 다른 변경 커밋들(34)과 함께) 수신하는 것에 응답하여, 메시지 생성 및 시퀀싱 프로세스(42)는, 예약 레코드 B에 관한 예시적인 "취소-재부킹" 흐름의 이들 변경 커밋 표시들(34)이 NDC 표준과 같은 최신 표준(즉, 제2 레거시 표준과는 대조적인 제1 표준)에 따른 변경 커밋들임을 인식한다. 예를 들어, 예약 레코드 B에 관한 예시적인 "취소-재부킹" 흐름의 변경 커밋 표시들(34)은, 이들을 NDC 변경 커밋 표시들로서 식별하는 특수 표시자를 포함한다.

[0060] 변경 커밋 표시(34)를 NDC 변경 커밋인 것으로 인식하는 것에 응답하여, 메시지 생성 및 시퀀싱 프로세스(42)는, 통지받을 각각의 메시지 타깃, 여기서는 NDC와 같은 제1 표준에 따라 동작하는 메시지 타깃(49a)에 대한 메시지 엔벨로프(36), 및 메시지 엔벨로프(36)에 속하는 하나 또는 다수의 메시지들(38)을 생성한다. 메시지 엔벨로프는, 인식된 NDC 변경 커밋(34)과 관련되고 메시지 타깃(49a)에 전송될 하나 또는 다수의 메

시지들을 그룹화한다. 게다가, 메시지 생성 및 시퀀싱 프로세스(42)는 2개의 레벨들에서의, 즉, 메시지 엔벨로프 레벨에서뿐만 아니라 메시지 레벨에서의 종속성들을 확립한다. 양측 모두의 레벨들에서의 종속성들은 이하 종속성 맵이라고도 또한 지칭된다.

[0061] 메시지 엔벨로프(36)를 생성하는 목적은, 이 메시지 엔벨로프의 메시지들이 전송될 수 있는지 여부와 그 시기, 즉, 메시지 엔벨로프(36)의 메시지들이 다른 변경 커밋들(34)에 대해 생성되는 다른 메시지들(38)에 대한 종속성들을 갖고 있지 않은지 여부를 관리 및 결정하기 위한 것이다. 그 의미에서, 기능적 관점으로부터, 메시지 엔벨로프(36)는 그 메시지 엔벨로프(36)에 속하는 모든 메시지들(38)에 대한 기준 노드를 구성한다(그에 따라 메시지 엔벨로프들은 이하 "R"로 또한 약칭된다).

[0062] 관련된 예약 레코드에 대한 하나 이상의 다른 메시지 엔벨로프들(36)이 관련된 메시지 타깃(49a)에 대해 이미 존재하는 경우, 신규로 생성된 메시지 엔벨로프(36)가, 잠재적으로 이미 존재하는 메시지 엔벨로프 레벨 종속성 맵에 포함된다. 도 3의 예에서, 비행 세그먼트 XX123의 취소를 표시하는 변경 커밋 표시 B2가 메시지 생성 및 시퀀싱 프로세스(42)에 의해 수신된다. 메시지 생성 및 시퀀싱 프로세스(42)는 메시지 타깃(49a)이 통지받아야 한다고 결정하고 메시지 엔벨로프 R2 및 그 메시지 엔벨로프 R2에 속하는 3개의 메시지들 2.1 XML, 2.2 XML 및 2.3 TTY를 생성한다.

[0063] 다음으로, 변경 커밋 표시 B3이 수신된다. 메시지 생성 및 시퀀싱 프로세스(42)는 메시지 타깃(49a)이 통지받아야 한다고 다시 결정하고 한 세트의 메시지들 3.1 XML, 3.2 XML 및 3.3 XML을 갖는 추가 메시지 엔벨로프 R3을 생성한다. 점선 화살표 37로 표시된 바와 같이, 메시지 엔벨로프 R3이 메시지 엔벨로프 R2에 종속된다는 점에서 메시지 엔벨로프들 R2 및 R3 사이의 종속성이 확립된다.

[0064] 부가적으로, 메시지 엔벨로프(36) 내에, 메시지들(38)이 순서대로 또한 놓여진다, 즉, 메시지 레벨 종속성 맵이 확립된다. 도 5의 예에서, 화살표 39로 표시된 바와 같이, 메시지 엔벨로프 R2 내의 메시지 순서는 2.1 XML, 2.2 XML 및 2.3 TTY이고, 메시지 엔벨로프 R3 내의 메시지 순서는 3.1 XML, 3.2 XML 및 3.3 XML이다. 일부 실시예들에서, 보다 높은 우선순위 값들을 갖는 메시지들이 보다 낮은 우선순위 값들을 갖는 메시지들보다 메시지 종속성 맵에서 상류에 위치되도록, 메시지 레벨 종속성 맵들은 미리 정의된 메시지 우선순위 값들에 기초하여 확립된다.

[0065] 다음으로, 메시지 생성 및 시퀀싱 프로세스(42)는 예약 레코드 B와 관련된 변경 커밋 표시 B4를 다시 수신한다. 변경 커밋 표시 B4는 레거시 예약 시스템으로부터 유래하고, 메시지 생성 및 시퀀싱 프로세스(42)는 다시 제1 표준에 따라 동작하는 메시지 타깃(49a)이 통지받아야 한다고 결정한다. 따라서, 메시지 생성 및 시퀀싱 프로세스(42)는 TTY 메시지, 메시지 4 TTY를 생성하고, 그것을, EP 2254046 B1호에 의해 설명된 TTY 프레임워크 메커니즘들에 따라 기존 종속성 맵 내에 정렬시킨다. 도 5의 예에서, 메시지 4 TTY가 메시지 2.3 TTY에 종속되도록 메시지 4 TTY에 대해 메시지 레벨 종속성이 연산된다(제시의 이유로, 여기서 메시지 2.3 TTY가 아직 전송되지 않았을 수 있다고 가정되는데, 이는, 예를 들어, 도 5에 도시되지 않은 예약 레코드 B와 관련된 다른 이전 TTY 메시지들이 여전히 메시지 타깃(49a)에 대해 계류 중이기 때문이다).

[0066] 메시지 엔벨로프 레벨에서의 그리고 메시지 레벨에서의 종속성 맵들은 메시지들을 메시지 타깃들(49)에 전송하는 시퀀스를 제어한다. 일반적으로, 특정 메시지 엔벨로프(36)가 다른 메시지 엔벨로프들(36)에 종속되지 않는 경우, 특정 메시지 엔벨로프(36)의 메시지들이 단지 메시지 타깃들(49)에 전송된다. 따라서, 도 5의 예에서, 메시지 엔벨로프 R3의 메시지들 3.1 XML, 3.2 XML 및 3.3 XML은 메시지 엔벨로프 R2가 완전히 프로세싱된 후에만, 즉, 메시지 엔벨로프 R2의 모든 메시지들이 메시지 타깃(49a)에 성공적으로 전송된 후에만 전송된다. 메시지 엔벨로프 내에서, 메시지들의 전송 순서는 메시지 레벨 종속성에 의해 결정된다. 도 5의 예에서, 메시지 엔벨로프 R2 내의 메시지 전송 순서는 먼저 메시지 2.1 XML이고, 그 후에 메시지 2.2 XML이고 마지막으로 메시지 2.3 TTY이다. 메시지 엔벨로프 R3의 경우, 메시지 전송 순서는 먼저 메시지 3.1 XML이고, 그 후에 메시지 3.2 XML이고 마지막으로 메시지 3.3 XML이다. 메시지 4 TTY는 메시지 2.3 TTY가 성공적으로 송신된 후에 전송될 수 있다, 즉, 메시지 4 TTY는 메시지 3.1 XML과 병렬적으로 전송된다.

[0067] 레거시 데이터베이스 시스템으로부터 수신되는 변경 커밋 표시들(34)은 일반적으로 단지 제2 레거시 표준에 따른 메시지들로, 도 5의 예에서는 TTY 메시지들로 변환된다. 관련된 주어진 메시지 타깃에 대해 이전 메시지가 계류 중이 아닌 경우에, 이들 메시지들은 메시지 엔벨로프 레벨 종속성 및 메시지 레벨 종속성을 연산하는 일 없이 즉시 전송될 수 있다. 도 5의 예에서, 예약 레코드 A는 레거시 데이터베이스 시스템에 의해 유지되고, 따라서, 변경 커밋 표시들 A0, A1, A2 및 A3은 TTY 메시지들로만 단지 변환된다. 예를 들어, 변경 커밋 표시 A0을 수신하는 것에 응답하여, 메시지 생성 및 시퀀싱 프로세스(42)는 (제2 레거시 표준에 따라 동작하는 메시지 타

깃인) 메시지 타깃(49b)이 통지받아야 한다고 결정하고, 메시지 타깃(49b)에 대해 다른 메시지가 계류 중이 아니기 때문에 그 후에 즉시 전송되는 메시지 0 TTY를 생성한다. 마찬가지로, 메시지 0 TTY가 이미 메시지 타깃(49b)에 성공적으로 전달되었다고 가정하면, 변경 커밋 표시 A1은 메시지 레벨 종속성을 연산하는 일 없이 즉시 다시 전송되는 메시지 1 TTY의 생성을 트리거한다.

[0068] 일반적으로, 메시지가 다른 메시지의 성공적인 전달에 종속되지 않는다는 결정에 응답하여, 메시지가 전송되기에 적격이고 메시지 전송 큐로 전달되는 것으로 결정되고, 여기서 이 종속성은 메시지 엔벨로프 레벨에서의 그리고 메시지 레벨에서의 종속성 맵들에 의해 정의된다. 예를 들어, 메시지 타깃(49)으로부터 수신 확인 응답을 수신함으로써, 특정 메시지의 성공적인 전달이 인식되는 경우, 이 특정 메시지의 종속성 스테이터스(dependency status)는 메시지 프로세싱 시스템에 의해 업데이트된다. 이 종속성 스테이터스 업데이트에 응답하여, 특정 메시지에 종속되는 다른 메시지들의 종속성 스테이터스는 그 후에, 이들 다른 메시지들의 전송이 다음에 개시되는 효과와 함께 "종속적"인 것으로부터 "독립적"인 것으로 업데이트된다. 특정 메시지 엔벨로프(36)의 메시지 레벨 종속성 맵에서의 마지막 메시지가 메시지 타깃(49)에 성공적으로 전달되었다는, 즉, 특정 메시지 엔벨로프(36)의 모든 메시지들이 전송되었다는 결정에 응답하여, 특정 메시지 엔벨로프(36)가 완전히 프로세싱된 것으로 간주되고, 특정 메시지 엔벨로프(36)에 종속되는 다음 메시지 엔벨로프의 메시지 엔벨로프 레벨에서의 종속성 스테이터스가 "종속적"인 것으로부터 "독립적"인 것으로 업데이트된다. 이것은 다음 메시지 엔벨로프의 메시지 레벨 종속성 맵의 제1 메시지가 전송될 준비가 되도록 렌더링하고, 다음 메시지 엔벨로프의 이 제1 메시지의 전송이 개시된다. 이 전체 프로세스는 메시지 엔벨로프의 모든 메시지가 이들의 메시지 타깃들에 전송될 때까지 동일한 방식으로 계속된다.

[0069] 앞에서 사용된 TTY 및 XML은 상이한 메시지 표준들에 따른 상이한 타입들의 메시지들에 대한 예들이다. TTY는 하나 이상의 레거시 메시지 서브타입들을 채용하는 레거시 표준에 대한 예이다. XML은, 예를 들어, 상이한 데이터베이스 조작 기능들에 대한 복수의 메시지 서브타입들을 채용하는 차세대 표준에 대한 예이다. 통상적으로, 상이한 메시지 타입들은 상이한 타입들의 메시지 타깃들, 즉, 상이한 메시지 표준들을 구현하는 메시지 타깃들에 대해 생성된다. 도 5의 예는 2개의 특정 타입들의 메시지 타깃들, 즉, 레거시 표준만을 단지 구현하는 TTY 메시지 타깃들(49b)(즉, 신규 세대 표준이 아님) 및 NDC 표준과 같은 신규 세대 표준을 구현하는 XML 메시지 타깃들(49a)(상기에 설명된 바와 같이, XML 메시지 타깃(49a)은 잠재적으로 하위 호환가능하고 따라서 레거시 메시지들을 또한 프로세싱하는 것이 가능할 수도 있다)을 나타낸다.

[0070] 본 메시지 프로세싱 시스템은 2개의 메시지 표준들의 메시지들을 프로세싱하는 것으로 제한되지 않지만, 그것은 그러한 신규 표준을 구현하는 메시지 타깃들(49c)에 대한 하나 이상의 추가 표준들 및 타입들에 따른 메시지들을 프로세싱하기 위해 확장가능하고 일반적으로 적합하다. 메시지 엔벨로프 레벨 및 메시지 레벨에서의 종속성들을 확립한다는 개념은, 예를 들어, 기존 표준들/타입들의 메시지들에 대한 신규 표준/타입의 메시지들의 관계를 정의하는 우선순위 값들을 신규 표준/타입의 메시지들에 할당함으로써, 임의의 다른 신규 메시지 표준들 및 타입들을 핸들링하기 위해 이용될 수 있다.

[0071] 이제 설명의 더 구체적인 부분으로 돌아가서, 상기에 설명된 일반 메시지 프로세싱 메커니즘들의 특정 구현예가 도 6 내지 도 14를 참조하여 다음에 설명될 것이다.

[0072] 메시지 프로세싱 엔진(50)을 구현하기 위한 예시적인 프로세스 아키텍처가 도 6에 예시되어 있다. 시스템 아키텍처는, 전용된 또/또는 공유된 하드웨어 리소스들을 사용하여 소프트웨어에 의해 구현되는 다음의 컴포넌트들을 포함한다:

[0073] - 레거시 GDS 변경 커밋 표시들을 생성하는 레거시 전역 분배 시스템(GDS) 예약 엔진(41);

[0074] - NDC PNR 변경 커밋 표시들을 생성하고, 이들 변경 커밋 표시들이 NDC 변경 커밋 표시들(즉, 제1 표준에 따름)이고, TTY 메시지 타깃들과 같은, 제2 레거시 표준에 따라 여전히 동작하는 메시지 타깃들이 통지받지 않는 한, NDC 표준에 따라 프로세싱되어야 함을 표시하는 플래그로 이들 변경 커밋들을 마킹하는 NDC 예약 엔진(40);

[0075] - TTY 메시지들, XML 메시지들 또는 다른 타입들의 메시지와 같은 모든 종류들의 예약 메시지들을 생성하는 NDC 메시지 생성 및 시퀀싱 프로세스(42). NDC 메시지 생성 및 시퀀싱 프로세스(42)는 메시지 종속성 맵들을 연산한다.

[0076] - NDC 메시지 생성 및 시퀀싱 프로세스(42)의 일부를 형성하는 TTY 메시지 시퀀싱 엔진(43). TTY 메시지 시퀀싱 엔진(43)은, 예를 들어, EP 2254046 B1호에 의해 설명된 TTY 메시지들의 전송 순서를 결정하는 TTY

메시지들의 종속성 특성들을 연산한다.

[0077] - 메시지 우선순위 값들을 저장하는 메시지 우선순위 테이블(44). 메시지 우선순위 값들은 분석 결과들에 기초하여 미리 연산되고, 요청되거나 또는 적절할 때마다 업데이트될 수 있다.

[0078] - 적어도 2개의 레벨들에서의 종속성 맵들, 즉, 메시지들에 대한 메시지 레벨에서의 종속성 체인들 및 메시지 엔벨로프에 대한 메시지 엔벨로프 레벨에서의 종속성 맵들을 저장하는 종속성 맵 및 메시지 테이블(45).

[0079] - 메시지 전송을 제어하기 위한 그리고 메시지 레벨에서의 종속성 체인들 및 엔벨로프 레벨에서의 종속성 맵을 업데이트하기 위한 메인 프로세스인 메시지 전송 상태 머신(46). 메시지 전송 상태 머신(46)은 메시지 큐(46a) 및 2개의 서브 프로세스들, 즉, 다음의 것을 포함한다.

[0080] - 메시지들을 지속적으로 데큐잉(dequeue)하고 메시지들을 타깃들(49)에 전송하는 메시지 전송자 서브프로세스(47), 그리고

[0081] - 메시지 응답들에 대한 청취자인 메시지 응답 수신자 서브프로세스(48). 타깃 응답 수신자 서브프로세스(48)는 메시지 타깃들(49)로부터 수신되는 수신 확인 응답들에 응답하여 종속성 맵 및 메시지 테이블(45)의 변경들을 추적하기 위해 전송자 데큐 동작들을 커밋하고 메시지 레벨에서의 종속성 체인들 및 메시지 엔벨로프 레벨에서의 종속성 맵들을 업데이트한다.

[0082] 이들 시스템 아키텍처 컴포넌트들은 도 6의 블록들을 연결하는 화살표들로 예시된 바와 같이 다음의 방식으로 서로 상호작용한다.

[0083] NDC 예약 엔진(40)은, 예를 들어, 신규 여행 예약에 응답하여, 신규 데이터베이스 레코드를 생성하거나, 또는 기존 여행 예약의 변경에 응답하여 기존 데이터 레코드를 업데이트한다. 신규 데이터베이스 레코드의 생성 및 기존 데이터베이스 레코드의 업데이트는, 각각, NDC 예약 엔진(40)이 NDC 메시지 생성 및 시퀀싱 프로세스(42)에 게시하는 변경 커밋 표시들을 생성한다(화살표 N01). NDC 예약 엔진(42)으로부터 유래하는 모든 게시물들은 NDC 예약 변경 커밋들로서, 즉, NDC 예약 엔진(40)으로부터 유래하는 데이터베이스 변경 커밋 표시들로서 마킹(예를 들어, 플래깅)된다.

[0084] 유사하게, 레거시 GDS 예약 엔진(42)은, 예를 들어, 신규 여행 예약에 응답하여, 신규 데이터베이스 레코드를 생성할 수도 있거나, 또는 기존 여행 예약의 변경에 응답하여 기존 데이터 레코드를 업데이트한다. 신규 데이터베이스 레코드의 생성 및 기존 데이터베이스 레코드의 업데이트는, 각각, 레거시 GDS 예약 엔진(42)이 NDC 메시지 생성 및 시퀀싱 프로세스(42)에 게시하는 변경 커밋 표시를 생성한다(화살표 N02). 그러나, 레거시 GDS 예약 엔진(40)에 의해 게시되는 변경 커밋 표시들은 마킹 또는 플래깅되지 않는다.

[0085] 레거시 GDS 예약 엔진(42)으로부터 변경 커밋 표시를 수신하는 것에 응답하여, NDC 메시지 생성 및 시퀀싱 프로세스(42)는 통지받을 메시지 타깃들에 대한 TTY 메시지들만을 생성한다.

[0086] NDC 예약 엔진(40)으로부터 데이터베이스 변경 커밋 표시들을 수신하는 것에 응답하여, NDC 메시지 생성 및 시퀀싱 프로세스(42)는 통지받을 NDC 메시지 타깃들에 대한 메시지 엔벨로프들 및 각각의 메시지 엔벨로프들과 연관된 메시지들을 생성한다. 이를 위해, NDC 메시지 생성 및 시퀀싱 프로세스(42)는 메시지 우선순위 테이블(44)로부터 메시지 우선순위 값들을 리트리빙(리트리빙 요청들을 표현하는 화살표 N05 및 리트리빙 응답들을 표현하는 화살표 N06)할 뿐만 아니라, 종속성 맵 및 메시지 테이블(45)로부터 현재 메시지 엔벨로프들 및 연관된 메시지들에 관한 정보를 리트리빙하고, 신규로 생성된 메시지 엔벨로프들 및 연관된 메시지들을 종속성 맵 및 메시지 테이블(45) 내에 삽입(리트리빙 및 삽입 요청들을 표현하는 화살표 N03 및 리트리빙 및 삽입 응답들을 표현하는 화살표 N04)한다. NDC 메시지 생성 및 시퀀싱 프로세스(42)와 메시지 우선순위 테이블(44) 및 종속성 맵 및 메시지 테이블(45)의 상호작용들이 특정 예에 기초하여 아래에 추가로 설명될 것이다.

[0087] 게다가, NDC 메시지 생성 및 시퀀싱 프로세스(42)는 특정 메시지가 임의의 다른 메시지에 종속되지 않는다고 결정할 수도 있는데, 이는 이 메시지가 메시지 타깃(49)에 전송될 준비가 됨을 의미한다. 이 상황에서, NDC 메시지 생성 및 시퀀싱 프로세스(42)는 동작들 N03(리트리빙 요청) 및 N04(리트리빙 응답)를 통해 종속성 맵 및 메시지 테이블(45)로부터 특정 메시지를 리트리빙하고, 특정 메시지를 표현하는 메시지 큐 아이템을 생성하며, 동작 N07을 통해 메시지 전송 상태 머신(46)의 메시지 큐(46a)에 메시지 큐 아이템을 놓는다.

[0088] 이에 후속하여, 메시지 큐 아이템은 동작 N08에 의해 메시지 전송 상태 머신(46)에 의해 데큐잉되고 메시지 전송자 서브프로세스(47)에 의해 올바른 메시지 타깃(49)에 전송된다(화살표 N09). 메시지 전송 상태 머신(46)의 메시지 응답 수신자 서브프로세스(48)에 의해 (수신의 확인응답과 같은) 메시지 응답이 메시지 타깃(49)으로부

터 수신될 수도 있다(화살표 N10).

[0089] 메시지 타깃(49)으로부터의 응답이 수신된 후에, 즉, 특정 메시지가 메시지 타깃(49)에 성공적으로 전달된 후에, 메시지 전송 상태 머신(46)은 종속성 맵 및 메시지 테이블(45)에서의 특정 메시지의 메시지 종속성 스테이터스를 업데이트한다(화살표 N11). 이 업데이트는 다음 메시지 또는 메시지들을 임의의 다른 메시지와는 독립적으로 되도록 렌더링할 수도 있다, 즉, 이 다음 메시지 또는 이들 다음 메시지들의 전송이 다음에 개시될 수 있다. 이를 위해, 특정 메시지의 성공적인 전달에 의해 독립적으로 된 메시지는 메시지 전송 상태 머신(46)에 의해 리트리빙되고(화살표 N12) 메시지 큐들(46a) 내에 큐잉된다. 그 후에, 동작들 N08, N09 및 N10이 상기에 설명된 것과 동일한 방식으로 실행된다.

[0090] 메시지 우선순위 테이블(44)의 예는 다음과 같다:

[0091]

| 메시지 타깃 | 메시지 타입 | 메시지 이름 | 우선순위 값 |
|---|---|---|---|
| YY | XML | DEL | 90 |
| YY | TTY | TTY | 50 |
| YY | XML | ADD | 10 |

[0092]

| YY | NEW | GEN | NULL |
|---|---|---|---|
| ZZ | XML | DEL | 95 |
| ZZ | TTY | TTY | 90 |
| ZZ | XML | ADD | 70 |
| ZZ | XML | SPLIT | 0 |

[0093] 컬럼(column) "메시지 타깃"은 메시지 목적지들을 정의한다, 즉, 알려진 메시지 타깃들(49) 모두를 리스팅한다. 컬럼 "메시지 타입"은 메시지 표준, 즉, 메시지 타입 TTY와 같은 레거시 표준에 따른 또는 XML 메시지들을 사용하는 NDC와 같은 현재 표준에 따른 메시지 표준을 표시한다. 메시지 표준/타입에 따라, 컬럼 "메시지 이름"에 표시되는 상이한 메시지 서브타입들의 메시지들이 있을 수도 있다. 상기의 예에 도시된 바와 같이, 레거시 표준은 모든 TTY 메시지들이 메시지 이름 TTY라고 지칭되도록 상이한 메시지 타입들을 제공하지 않을 수도 있다. 그러나, XML 타입 메시지들을 사용하는 신규 표준은 삭제 메시지들("DEL") 및 부가 메시지들("ADD")과 같은 상이한 메시지들을 정의할 수도 있다. 여전히 제3 표준("NEW")은 일반 메시지들("GEN")을 정의할 수도 있다.

[0094] 컬럼 "우선순위 값"은 특정 메시지 타깃에 어드레싱될 주어진 메시지 타입 및 메시지 이름의 모든 메시지들에 대한 메시지 우선순위 값을 정의한다. 따라서, 우선순위 값들은 메시지 타깃, 표준 및 메시지 서브타입/이름의 레벨들 상의 메시지들 사이를 차별화시키는 것을 가능하게 한다. 상기의 예에서, "우선순위 값" 컬럼에서의 값들은 다음의 의미들을 갖는다: 0의 우선순위 값은 메시지가 전혀 전송되지 않을 것이라는 것을 의미한다. 이것은, 예를 들어, 특정 기능을 갖는 메시지들이 제1 표준에 의해 이미 특정되었지만, 실제로 배치된 메시지 타깃들에 의해 아직 지원되지 않는 경우들에 합리적이다. 상기의 테이블(44)의 예에서, 이것은, 예를 들어, 메시지 프로세싱 엔진(50)에 의해 내부적으로 이미 생성되었지만 메시지 타깃들에서의 구현 결여로 인해 (아직) 전송되지 않은 SPLIT 메시지에 대한 경우이다. 우선순위 값이 설정되지 않은 경우("NULL"), 메시지 엔벨로프들의 레벨에서의 종속성이 없을 때 메시지가 전송될 수 있다. 엔벨로프의 메시지들은 보다 높은 값에서부터 보다 낮은 값까지의 우선순위 값들의 이들의 우선순위 값들에 따라 상호종속적이다.

[0095] 메시지 우선순위 테이블(44)은 일반적으로 정적이지만, 언제라도 구성가능하다. 신규 타깃이 부가될 수 있고, 신규 메시지 타입들이 다양한 타깃들에 대해 부가될 수 있고 또한 신규 메시지 이름들이 주어진 메시지 타입에 대해 부가될 수 있다. 더욱이, 예를 들어, 특정 타깃이 제거되어야 할 때, 기존 로우(row)들이 업데이트 또는 삭제될 수 있다. 우선순위 값들이 또한 재구성될 수 있다.

[0096] 일부 실시예들에서, 메시지 우선순위 테이블(44)은 데이터베이스 시스템에 의해 호스팅되지 않지만, 예를 들어, 배치 서버(batch server)의 메모리 캐시 파일의 형태로 저장된다. 캐시 파일은 요구에 따라 또는 적절하다면 임의의 서버와 동기화될 수 있다. 예를 들어, 메시지 우선순위 테이블(44)에 대한 변경들은 배치 서버의 콘텐츠 관리 시스템에 의해 이루어진 후에 모든 배치된 서버들로 브로드캐스팅될 수도 있다.

[0097] 다른 한편으로, 종속성 맵 및 메시지 테이블(45)은 고도로 동적인데, 이는 종속성 맵 및 메시지 테이블(45)이 임의의 주어진 시점에서의 메시지 프로세싱 시스템의 현재 스테이터스를 표현하기 때문이다. 따라서, 일부 실시예들에서, 종속성 맵 및 메시지 테이블(45)은 데이터베이스 시스템에 의해 호스팅된다. 본 명세서에서 설명되는 메시지 프로세싱 시스템은 종속성 맵 및 메시지 테이블(45)을 실시간으로 판독, 삽입 및 업데이트하기 위한 다수의 라이브 세션들을 실행할 수도 있다.

[0098] 종속성 맵 및 메시지 테이블(45)의 컬럼들의 예시적인 정의는 다음과 같다:

| 컬럼 이름 | 정의 |
|---|---|
| ID | 메시지 식별자 번호 |
| P | NDC 예약 엔진(40) 또는 레거시 GDS 예약 엔진(41)의 연관된 예약 레코드를 식별하는 참조 키. 문자들 및 번호들의 스트링 및/또는 P-키-순서 |
| I | 변경 커밋 번호: P(i)는 예약 레코드 *P*에 대한 i번째 데이터베이스 변경 커밋을 표현한다 |
| J | 메시지 타깃 |
| RD_STAT | 메시지 엔벨로프 레벨에서의 종속성 스테이터스 |
| D_STAT | 메시지 레벨에서의 종속성 스테이터스 |
| D_ID | 메시지 엔벨로프 또는 메시지가 종속되는 메시지 엔벨로프 또는 메시지를 표시하는 종속성 식별자 |
| N | 메시지 이름 |
| 타입 | 메시지 타입 |
| 메시지 | 생성된 메시지 자체. $M_j^{P(i),n}(x)$는, 예약 레코드 P의 변경 커밋 i에 대해 생성된, 타깃 j에 전송될 x번째 메시지를 표현하고, 메시지 이름은 n이다. |

[0099]

[0100] 2개의 종속성 상태들 RD_STAT 및 D_STAT는, 메시지가 전송된 경우에는 값들 "전송됨(sent)"(S), 또는 타깃에 전송될 메시지가 없는 경우에는 "보이드(void)"(V)(예를 들어, 테이블 엔트리는 메시지와는 대조적으로 메시지 엔벨로프를 표현한다), 또는 메시지가 메시지 레벨에서 다른 메시지에 대한 종속성을 갖는 경우 또는 메시지의 메시지 엔벨로프가 다른 메시지 엔벨로프에 대한 종속성을 갖는 경우에는 "종속적(dependent)"(D), 또는 종속성이 존재하지 않는(= 전송될 준비가 된) 경우에는 "N" 또는 종속성이 아직 식별되지 않았을 때, 예를 들어, 대응하는 메시지가 바로 그때에 누락될 때, 즉, NDC 메시지 생성 및 시퀀싱 프로세스(42)에서 여전히 미해결될 때에는 "미지수(unknown)"(U)를 취할 수도 있다. RD_STAT 셀이 채워지고 D_STAT 셀이 비어 있는 경우, 테이블 엔트리들은 메시지 엔벨로프들을 표현한다. 예를 들어, $\{M_j^{P(i),n}(x)\}$의 메시지 리스트들에 대한 메시지 엔벨로프는 $R_j^{P(i)}$로 표기된다.

[0101] 일부 실시예들에서, 메시지 프로세싱 시스템(50)은, 도 7에 도시된 바와 같이, 각각의 잠재적으로 다수의 프로세스 인스턴스들(54)을 갖는 서버들(52)의 어레이로서 배치된다. 로드 밸런스 피처들은 서버(52)의 프로세스 인스턴스들(54)에 걸쳐 그리고/또는 서버 어레이의 서버들(52)에 걸쳐 메시지 프로세싱 로드들을 분산시킬 수도 있다. 그에 따라, 메시지 프로세싱 시스템(50)은 잠재적인 트래픽 증가에 대처하기 위해 확장가능하다.

[0102] 메시지 프로세싱 시스템(50)은 종속성 맵 및 메시지 테이블(45)을 유지하는 적어도 하나의 데이터베이스 서버(56)(리던던시(redundancy) 및/또는 성능의 이유로 부가적인 데이터베이스가 존재할 수도 있다)를 특징으로 한다. 모든 프로세스 인스턴스들(54)은 데이터베이스 서버(56)에 연결된다. 전역적으로 보면, 메시지 프로세싱 시스템(50)은 도 6에 도시된 모든 컴포넌트들 및 여행사 클라이언트들과 같은 추가 컴포넌트들을 포함하는 보다 큰 분산 시스템의 정보 프로세싱 체인에서 하나의 스테이지를 구성한다. 다른 한편으로, 메시지 프로세싱 시스

템(50)은 병렬 프로세싱을 수행하는데, 이는 각각의 개별 프로세스 인스턴스(54)가 한 번에 하나의 레코드 엔벨로프를 프로세싱하는 것이 가능하기 때문이다.

[0103] 상기에 설명된 바와 같이, NDC 메시지 생성 및 시퀀싱 프로세스(42)는 메시지 프로세싱 시스템(50)의 메인 프로세스이다. NDC 메시지 생성 및 시퀀싱 프로세스(42)는 레거시 TTY 메시지 시퀀싱 엔진(43)을 포함하고 메시지 우선순위 테이블(44) 및 종속성 맵 및 메시지 테이블(45)과 상호작용한다. 도 6에 의해 도시된 동작들 N01 내지 N07은 다음에 설명되는 바와 같이 도 8의 예시적인 프로세스 흐름도에 의해 구현될 수도 있다.

[0104] 602에서 NDC 예약 엔진(40) 또는 레거시 GDS 예약 엔진(41)으로부터 데이터베이스 변경 커밋 표시가 수신될 때마다, NDC 메시지 생성 및 시퀀싱 프로세스(42)는 먼저, 604에서, 데이터베이스 변경 커밋에 의해 영향받은 메시지 타깃들의 리스트(49)를 구축하고, 따라서 NDC 예약 엔진(40) 또는 레거시 GDS 예약 엔진(41)에서 발생한 예약 레코드 변경에 대해 타깃에 알리는 하나 이상의 메시지들을 수신할 것이다.

[0105] 일반적으로, 상이한 타깃들(49)에 대한 메시지들은 종속 관계로 되어 있지 않다, 즉, 제2 타깃으로 지향되는 제2 메시지의 스테이터스에 관계없이 제1 타깃에 대한 제1 메시지가 전송될 수 있다. 따라서, 상기에 이미 설명된 바와 같이, 메시지 엔벨로프 레벨 종속성들은 동일한 메시지 타깃에 대한 메시지 엔벨로프들 사이에서만 확립되고, 자연스럽게, 메시지 레벨 종속성들은 메시지 엔벨로프의 메시지들 사이에서만 그리고 따라서 하나의 동일한 메시지 타깃에 대한 메시지 사이에서만 확립된다. 그 결과, 도 8에 의해 도시된 바와 같이, NDC 메시지 생성 및 시퀀싱 프로세스(42)는 리스트에서의 메시지 타깃들(49)을 차례로 프로세싱한다. 주어진 데이터베이스 변경 커밋 표시에 대해, 각각의 메시지 타깃이 도 8에 의해 도시된 다음의 활동들에 의해 통지받기 위해 레거시 메시지(도 8의 좌측편) 또는 하나 이상의 메시지들을 갖는 신규 메시지 엔벨로프(도 8의 우측편)를 생성함으로써 메시지 종속성 맵이 구축 또는 확장된다.

[0106] 606에서, NDC 메시지 생성 및 시퀀싱 프로세스(42)는 604에서 확립된 메시지 타깃 리스트의 끝에 도달되었는지 여부를 체크한다. 이것이 그러한 경우, 수신된 데이터베이스 변경 커밋 표시가 완전히 프로세싱되었기 때문에 도 8의 프로세스가 종료된다. 이것이 그러하지 않은 경우, 프로세스는 메시지 타깃 리스트 상의 다음 메시지 타깃(49)으로 진행하고, 608에서 이 다음 타깃을 판독하며, 다음 타깃이 제1 표준(즉, 이 예에서는 NDC)에 따라 동작하는지 또는 제2 레거시 표준에 따라 동작하는지(즉, 이 예에서는 TTY 메시지들만을 프로세싱한다) 여부를 결정한다. 그 후에, 프로세스는 610에서 수신된 데이터베이스 변경 커밋 표시가 NDC 커밋으로서 마킹되는지 여부, 즉, 그것이 NDC 예약 엔진(40) 또는 레거시 GDS 예약 엔진(41)의 커밋 표시인지 여부를 체크한다.

[0107] 박스들 608 및 610의 2개의 결정들은 현재 수신자가 도 8의 좌측편을 통해 또는 우측편을 통해 프로세싱되는지 여부를 결정한다. 데이터베이스 변경 커밋 표시가 NDC 예약 엔진(40)으로부터 유래하고 따라서 제1 표준에 부합하고 현재 메시지 타깃이 제1 표준에 따라 동작한다는 결정에 응답하여, 박스 610의 "예 분기(yes branch)"가 선정된다. 모든 다른 경우들에서, 즉, 데이터베이스 변경 커밋 표시가 610에서 레거시 GDS 예약 엔진(41)으로부터 유래하거나 그리고/또는 현재 메시지 타깃이 608에서 제2 레거시 표준에 따라 동작한다는 결정에 응답하여, 박스 610의 "아니오 분기(no branch)"가 선정된다. 도 8에 의해 도시된 전체 프로세스의 양측 모두의 분기들이 종속성 맵 및 메시지 테이블(45)을 사용한다. 박스 610의 좌측편 "아니오 분기"는, 고려된 메시지 타깃에 대한 다른 이전에 계류 중인 메시지들이 계류 중이 아니기 때문에 TTY 메시지들이 직접 전송되지 않는 한, 메시지 레벨 종속성들뿐만 아니라 메시지들 자체를 종속성 맵 및 메시지 테이블(45)에 저장한다. 박스 610의 우측편 "예 분기"는 메시지 엔벨로프 레벨에서의 그리고 메시지 레벨에서의 종속성들뿐만 아니라 메시지들 자체를 종속성 맵 및 메시지 테이블(45)에 저장한다.

[0108] 레거시/비-NDC 메시지 타깃들 및/또는 레거시 변경 커밋 표시들에 대해, NDC 메시지 생성 및 시퀀싱 프로세스(42), 더 구체적으로는 TTY 시퀀싱 엔진(43)은 활동들 612 내지 624에 의해 TTY 메시지를 생성한다. 612에서 TTY 메시지가 포맷팅되고, 614에서 TTY 종속성 아이템이 초기화된다. 여기서, TTY 메시지들에 대한 메시지 우선순위 값들은 메시지 우선순위 테이블(44)로부터 리트리빙된다. 616이 현재 TTY 메시지가 606에서 리스트들로부터 선택된 타깃에 대한 제1 메시지라고 결정하는 경우, 즉, 이 타깃에 대해 계류 중인 이전 TTY 메시지들이 없는 경우, TTY 메시지는 메시지 전송 상태 머신(46)에 전달되고 큐(46a) 내에 직접 놓여진다. 그 후에, 종속성 맵 및 메시지 테이블(45)은 620에서 커밋되고 전체 프로세스가 626에서 타깃 리스트 상의 다음 타깃으로 이동한다. 606에서 검증된 바와 같이, 타깃 리스트의 끝에 이제 도달되는 경우, 프로세스는 종료된다.

[0109] 616에서, 하나 이상의 다른 XML 및/또는 TTY 메시지들이 606에서 이전에 선택된 현재 메시지 타깃에 대해 이미 계류 중이라는 결정에 응답하여, NDC 메시지 생성 및 시퀀싱 프로세스(42)는 622에서 TTY 메시지의 메시지 레벨 종속성 스테이터스를 연산한다, 즉, 파라미터 D_STAT가 TTY 메시지에 대해 설정된다. D_STAT가 "N"인 것으로 연

산되는 경우, 즉, TTY 메시지에 대한 종속성이 존재하지 않는 경우, TTY 메시지는 메시지 타깃(49)에 직접 전송될 수 있다. 따라서, 상기에 설명된 것과 유사하게, TTY 메시지는 메시지 전송 상태 머신(46)에 전달되고 618에서 큐(46a) 내에 직접 놓여지고 프로세스는 상기에 설명된 바와 같이 진행된다. 동작 624가 "N" 이외의 메시지의 임의의 다른 종속성 상태를 결정하는 경우, 종속성 맵 및 메시지 테이블(45)은 그 후에 620에서 커밋되고 전체 프로세스가 626에서 타깃 리스트 상의 다음 타깃으로 이동한다.

[0110] TTY 레거시 메시지들을 프로세싱하는 것의 추가 세부사항들이 EP 2254046 B1호에, 특히 EP 2254046 B1호의 도 5, 도 6 및 도 7 및 단락들 [0030] 내지 [0034]에서의 첨부된 설명에 설명되어 있는데, 이들은 본 명세서에 참조로 포함된다.

[0111] 608에서 결정된 바와 같이, NDC 표준을 구현하는 메시지 타깃들(49)의 경우, 그리고 현재 프로세싱된 변경 커밋 표시가, 610에서 결정된 바와 같이, NDC 플래깅된다면, NDC 메시지 생성 및 시퀀싱 프로세스(42)는 도 8의 활동들 612 내지 624에 의해 하나 이상의 NDC 메시지들을 생성한다.

[0112] 628에서, 하나 이상의 메시지들이 제1 표준(여기서는: XML에 기초하는 NDC)에 따라 그리고/또는 제2 표준(여기서는: TTY에 기초하는 레거시 메시지 표준)에 따라 생성된다. 동작들 630 및 632는 동작들 614 및 616과 기능적으로 유사하지만, - 614 및 616과는 상이함 - 메시지 엔벨로프 레벨에서의 종속성 체인에 관련된다. 630에서, 하나 이상의 NDC 메시지들에 대한 메시지 엔벨로프가 생성되고 메시지 엔벨로프의 종속성 스테이터스 RD_STAT가 초기화된다. 632에서의 체크는, 생성된 메시지 엔벨로프가 종속성 맵 및 메시지 테이블(45)에 저장된 메시지 엔벨로프 종속성 맵에서 현재 루프에서 고려된 현재 타깃(49)에 대한 제1 메시지 엔벨로프인지 여부를 결정한다. 632에서 현재 타깃(49)에 대한 이전 메시지 엔벨로프가 없다고 결정하는 것에 응답하여, 신규로 생성된 메시지 엔벨로프는 종속성을 갖고 있지 않고, 이에 따라 634에서 파라미터 RD_STAT가 그 후에 N으로 설정된다. 다른 한편으로, 632에서 현재 타깃(49)에 대한 하나 이상의 메시지 엔벨로프들이 종속성 맵 및 메시지 테이블(45)에 이미 존재한다고 결정하는 것에 응답하여, 636에서 파라미터 RD_STAT가 그 후에 D로 설정된다. RD_STAT가 634 또는 636에서 연산된 후에, 프로세스(638)는 종속성 맵 및 메시지 테이블(45)에 존재하는 메시지 엔벨로프를 저장하기 위해 호출된다.

[0113] 638에서 종속성 맵 및 메시지 테이블(45)에 대한 성공적인 커밋 후에, 프로세스(640)는 $\{M_j^{P(i),n}(x)\}$ 의 메시지 종속성 체인, 즉, 현재 루프의 타깃(49)으로 지향되는 메시지 엔벨로프 내의 메시지들의 종속성들을 구축하기 위해 호출된다. 동작 640은 메시지 우선순위 테이블(44)에서의 미리 정의된 메시지 우선순위 값들을 사용한다.

[0114] 동작 640의 상세한 흐름도가 도 9에 예시되어 있다. 이제 도 9를 참조하면, 도 9의 702에 나타낸 바와 같이, 도 8의 동작 640의 입력들은 다음과 같다:

[0115] i) 메시지 엔벨로프 $R_j^{P(i)}$,

[0116] ii) 메시지 리스트, $\{M_j^{P(i),n}(x)\}$ 및

[0117] iii) 메시지 리스트에서의 메시지들의 메시지 이름들, 타입들 및 타깃들.

[0118] 출력들은, 입력들에 대해 구축되는 종속성 아이템들의 리스트이다. 제1 프로세스들 704 및 706은 메시지 우선순위 테이블(44)을 판독하고 메시지 리스트에서의 메시지들 $M_j^{P(i),n}(x)$ 의 우선순위 값들을 발견하는 것을 목표로 한다. 그 후에, 입력 메시지 리스트가 보다 높은 우선순위 값들로부터 보다 낮은 우선순위 값들로 정렬된다. 정렬된 메시지 리스트는 $\{M_j^{P(i),n}(x)'\}$ 로서 표기된다.

[0119] 일단 메시지들이 순서대로 정렬된 후에, 체크 708을 통해 메시지 리스트에서의 메시지들 각각을 루핑하면, 체크 710은 먼저 현재 고려된 메시지가 메시지 우선순위 테이블(44)에 정의된 유효한 우선순위 값을 갖는지를 결정한다. 710에서 현재 고려된 메시지가 유효한 우선순위 값을 갖는다는 결정에 응답하여, 메시지가 메시지 리스트에서의 제1 메시지 $M_j^{P(i),n}(1)$ 인지 여부를 결정하기 위해 추가 체크 720이 호출된다. 예(yes)인 경우,

722에서, 메시지 $M_j^{P(i),n}(1)'$의 D_ID가 $R_j^{P(i)}$의 ID에 할당된다. 그 후에, $R_j^{P(i)}$의 RD_STAT 값이 N 또는 V인지, 즉, 메시지 엔벨로프가 다른 메시지 엔벨로프들과는 독립적인지가 추가로 체크된다(724). 이것이 참(true)인 경우, 726에서 현재 메시지는 독립적인 것으로, 즉, $\mathrm{e}M_j^{P(i),n}(1)'$에 대한 D_STAT 값 = N으로 설정된다. 아니오(not)인 경우, 728에서 메시지는 종속적인 것으로, 즉, $\mathrm{e}M_j^{P(i),n}(1)'$에 대한 D_STAT 값 = D로 설정된다. 그 후에, 도 9의 프로세스는 738에서 메시지 리스트에서의 다음 메시지로 이동한다.

[0120] 720에서 결정된 바와 같이, 현재 루프에서 프로세싱된 메시지가 더 이상 제1 메시지가 아닌 경우, 730에서, 메시지 $M_j^{P(i),n}(x)'$의 D_ID가 메시지 $M_j^{P(i),n}(x-1)'$의 ID에 할당된다. 그 후에, 현재 루프의 메시지의 우선순위 값이 0과 동일한지를 결정하기 위해 추가 체크 732가 수행된다. 상기에 언급된 바와 같이, 0의 우선순위 값은 메시지가 전혀 전송되지 않을 것이라는 것을 의미한다. 따라서, 우선순위 값 = 0이 인식되는 경우, 그러면 $M_j^{P(i),n}(x)'$의 D_STAT가 V로 설정된다. 아니오인 경우, 동작들 734 및 736 각각에서 $M_j^{P(i),n}(x)'$의 D_STAT가 D로 설정된다. 그 후에, 도 9의 프로세스는 738에서 메시지 리스트에서의 다음 메시지로 이동한다.

[0121] 현재 메시지의 우선순위 값이 710에서 정의되지 않은 것으로 결정되는 경우, 이것은 메시지가 메시지 엔벨로프 $R_j^{P(i)}$에만 단지 종속된다는 것을 의미한다. 따라서, (동작 722와 동일한) 712에서 D_ID는 항상 $R_j^{P(i)}$의 ID로 설정되고, 714에서 메시지 엔벨로프의 RD_STAT가 N 또는 V인 것으로 결정되는지 여부에 따라, 메시지의 D_STAT가 716에서는 D로 또는 718에서는 N으로 설정된다. 그 후에, 도 9의 프로세스는 738에서 메시지 리스트에서의 다음 메시지로 이동한다.

[0122] 이제 도 8을 다시 참조하면, 동작 640이 종결된 후에, 메시지 엔벨로프의 메시지들이 전송을 위해 메시지 큐(46a)에 놓여질 수 있는지를 결정하기 위해 메시지 엔벨로프 종속성 RD_STAT에 대한 체크 642가 호출된다. RD_STAT = N인 경우, 동작 644는 먼저 전송하기 위해 메시지 엔벨로프의 제1 메시지, 즉, $M_j^{P(i),n}(1)'$를 큐(46a) 내에 놓도록 실행된다. 그 후에, 644에서 D_STAT가 N과 동일한 메시지 엔벨로프의 다른 메시지들 $M_j^{P(i),n}(x)'$이 또한 큐(46a) 내에 놓여진다.

[0123] 642에서 메시지 엔벨로프의 RD_STAT가 N이 아닌 것으로 결정되는 경우, 현재 루프의 메시지 타깃(49)에 대한 메시지 종속성 맵에서 TTY 메시지가 $M_j^{P(i),n}(1)'$인지를 결정하기 위해 추가 체크 646이 수행된다. 이것이 긍정적인 경우, 648에서 메시지 종속성 맵이 업데이트된다. 더 구체적으로는, 648에서, $M_j^{P(i),n}(1)'$의 D_ID를 $P(i-1)$의 TTY 메시지에 또는, TTY가 존재하지 않는 경우에는, 메시지 엔벨로프 $R_j^{P(i-1)}$에 링크하기 위해 $\{M_j^{P(i),n}(x)\}$이 업데이트된다.

[0124] 이 스테이지에서, 동작들 644, 646 및 648이 각각 종결된 후에, 메시지 리스트 $\{M_j^{P(i),n}(x)\}$의 연산이 완료되고 따라서 $(P(i),j)$에 대한 종속성 체인이 정의된다. 부가적으로, D_STAT = N인 모든 메시지들은 이들 메시지들을 메시지 타깃들(49)에 전송하기 위해 메시지 큐들(46a) 내에 모두 놓여진다. 그 후에, 커밋 동작 620을 통해 결과들이 데이터베이스에 저장된다. 그 후에, 도 8의 프로세스는 626에서 타깃 리스트에서의 다음 메시지 타깃으로 이동한다.

[0125] NDC 메시지 생성 및 시퀀싱 프로세스(42)는, 예를 들어, 아래에 제시되는 예시적인 테이블 I 및 테이블 II에서의, 메시지 리스트의 예시적인 업데이트 이력을 참조하여 추가로 설명될 수 있는데, 이들 양측 모두는 특정 시점에서의 종속성 맵 및 메시지 테이블(45)의 예들이다.

[0126] 이 업데이트 이력 예에서, 2개의 예약 레코드들 AAAAAA 및 BBBBBB에 대한 커밋 표시들이 NDC 메시지 생성 및 시퀀싱 프로세스(42)에 의해 수신된다. YY 및 ZZ라고 지칭되는 2개의 메시지 타깃들(49)은 이들 2개의 예약 레코드들과 관련되고, 따라서 메시지 프로세싱 시스템(50)에 의해 생성되는 메시지들의 수신자들이다. 메시지 엔벨로프 엔트리들의 경우, 처음 6개의 컬럼들만이 단지 채워진다. 메시지 엔트리들은 제5 컬럼 RD_STAT에 대한 값들을 포함하지 않는데, 이는 이 파라미터가 메시지 엔벨로프 레벨에 대한 종속성을 지칭하기 때문이다.

[0127] 테이블 I. 프로세싱한 메시지 엔벨로프들 0 및 1의 종속성 맵들

| ID | P | I | J | RD_STAT | D_ID | D_STAT | N | 타입 | 메시지 |
|---|---|---|---|---|---|---|---|---|---|
| 001 | AAAAAA | 0 | YY | S | NULL | NULL | NULL | NULL | NULL |
| 002 | BBBBBB | 0 | ZZ | S | NULL | NULL | NULL | NULL | NULL |
| 003 | AAAAAA | 0 | YY | NULL | 001 | S | TTY | T | $M_{YY}^{A(0),TTY}(1)'$ |
| 004 | AAAAAA | 1 | ZZ | N | 001 | NULL | NULL | NULL | NULL |
| 005 | AAAAAA | 1 | ZZ | NULL | 004 | S | DEL | X | $M_{ZZ}^{A(1),DEL}(1)'$ |
| 006 | BBBBBB | 0 | ZZ | NULL | 002 | V | TTY | T | $M_{ZZ}^{B(0),TTY}(1)'$ |
| *007* | *AAAAAA* | *1* | *ZZ* | *NULL* | *005* | *N* | *TTY* | *T* | $M_{ZZ}^{A(1),TTY}(2)'$ |
| 008 | BBBBBB | 0 | ZZ | NULL | 006 | S | ADD | X | $M_{ZZ}^{B(0),ADD}(2)'$ |
| 009 | BBBBBB | 1 | YY | N | 002 | NULL | NULL | NULL | NULL |
| *010* | *BBBBBB* | *1* | *YY* | *NULL* | *009* | *N* | *TTY* | *Y* | $M_{ZZ}^{B(1),TTY}(1)'$ |
| 011 | AAAAAA | 1 | YY | N | 001 | NULL | NULL | NULL | NULL |
| *012* | *AAAAAA* | *1* | *YY* | *NULL* | *011* | *N* | *ADD* | *X* | $M_{YY}^{A(1),XML}(1)'$ |

[0128]

[0129] 메시지 엔벨로프들 AAAAAA(0) 및 BBBBBB(0)은 예약 레코드들 AAAAAA 및 BBBBBB의 첫 번째의 각각의 제1 메시지 엔벨로프들이다. 따라서, ID 001 및 002를 갖는 테이블 엔트리들은 각각 도 8의 체크 632에서 "시퀀스에 대해 첫 번째"인 것으로서 검출된다. AAAAAA(1)은 제1 메시지 엔벨로프가 아니라, 적어도 하나의 메시지가 메시지 타깃 ZZ에 대해 생성되는 예약 레코드 AAAAAA와 관련된 제1 메시지 엔벨로프이다. 그 결과, ID 004를 갖는 메시지 엔벨로프 테이블 엔트리는 또한 도 8의 체크 632에서 "시퀀스에 대해 첫 번째"인 것으로서 검출된다. 테이블 I에 나타낸 바와 같이, 이들 메시지 엔벨로프 테이블 엔트리들은 파라미터 D_ID가 설정되어 있지 않고, 이들의 각각의 RD_STAT가 "V"로 설정되는데, 이는 이들 메시지 엔벨로프들이 각각의 메시지 타깃들(49)의 다른 메시지 엔벨로프들에 종속되지 않기 때문이다. 그에 따라, 메시지들 $M_{YY}^{A(0),TTY}(1)$ 및 $M_{ZZ}^{A(1),DEL}(1)$ 은 이들이 NDC 메시지 생성 및 시퀀싱 프로세스(42)에 의해 생성되면 즉시 전송될 수 있고, 이들의 D_STAT가 S로 업데이트된다. $M_{ZZ}^{B(0),TTY}(1)'$ 은 메시지 리스트 $\{M_{ZZ}^{B(0),n}(x)'\}$에서의 제1 메시지이지만, 그것은 전송될 필요가 없는 것으로서 연산되고(예를 들어, 이는 이 TTY 메시지가 메시지 타깃에 기능적 영향들을 갖지 않거나, 또는 메시지 타깃이 TTY 메시지에 미등록된 것으로 알려져 있거나 하는 것 등으로 되기 때문이다, 추가 배경을 위해 본 명세서에 참조로 포함되는 EP 2254046 B1호, 단락 [0032] 및 도 7 참조), 따라서 그의 D_STAT는 최종적으로 V로서 업데이트된다.

[0130] 그 후에, 메시지 $M_{ZZ}^{B(0),ADD}(2)'$ 가 전송되고 그의 D_STAT가 S로 업데이트된다. 메시지들 $M_{ZZ}^{A(1),TTY}(2)', M_{YY}^{B(1),TTY}(1)'$ 및 $M_{YY}^{A(1),XML}(1)'$ 가 연산되고, 이들은 전송을 위해 메시지 큐(46a) 내에 놓여질 준비가 되어 있다.

[0131] 테이블 I에 따른 메시지 엔벨로프들 및 메시지의 프로세싱이 NDC 메시지 생성 및 시퀀싱 프로세스(42)에 의해 완료되고 종속성 맵 및 메시지 테이블(45)이 방금 커밋되었을 때의 순간에, 데이터베이스 레코드들 AAAAAA 및

BBBBBB에 대한 추가 변경들에 의해 트리거된, 데이터베이스 레코드 AAAAAA의 데이터베이스 변경 커밋 표시들 2 및 10 그리고 BBBBBB의 데이터베이스 변경 커밋 표시 2가 NDC 메시지 생성 및 시퀀싱 프로세스(42)에 의해 수신된다. 메시지 큐들(46a)로 전달된 메시지들이 아직 타깃들(49)에 전송되지 않는다.

[0132] 데이터베이스 레코드 AAAAAA의 변경 커밋들 2 및 10 그리고 데이터베이스 레코드 BBBBBB의 변경 커밋 2의 표시들을 수신하는 것에 응답하여, 변경 커밋 표시 AAAAAA(2)에 대해 2개의 추가 메시지 리스트들, 즉, $\{M_{YY}^{A(2),TTY}(1),\ M_{YY}^{A(2),DEL}(2),\ M_{YY}^{A(2),NDC}(3)\}$ 및 $\{M_{ZZ}^{A(2),TTY}(1),\ M_{ZZ}^{A(2),SPLIT}(2)\}$가 NDC 메시지 생성 및 시퀀싱 프로세스(42)에 의해 생성된다. 게다가, 변경 커밋 표시 AAAAAA(10)에 대해 하나의 메시지 리스트, 즉, $\{M_{YY}^{A(10),TTY}(1),\ M_{YY}^{A(10),ADD}(2)\}$가 생성되고, 변경 커밋 표시 BBBBBB(2)에 대해 하나의 메시지 리스트 $\{M_{YY}^{B(2),TTY}(1),\ M_{YY}^{B(2),ADD}(2),\ M_{YY}^{B(2),DEL}(3)\}$이 생성된다. 테이블 II는 이들 메시지 리스트들 모두가 NDC 메시지 생성 및 시퀀싱 프로세스(42)에 의해 프로세싱된 후의 종속성 맵 및 메시지 테이블(45)을 나타낸다.

[0133] 테이블 II. 프로세싱한 메시지 엔벨로프들 2 및 10의 종속성 맵들

| ID | P | I | J | RD_STAT | D_ID | D_STAT | N | 타입 | 메시지 |
|---|---|---|---|---|---|---|---|---|---|
| 001 | AAAAAA | 0 | YY | S | NULL | NULL | NULL | NULL | NULL |
| 002 | BBBBBB | 0 | ZZ | S | NULL | NULL | NULL | NULL | NULL |
| 003 | AAAAAA | 0 | YY | NULL | 001 | S | TTY | T | $M_{YY}^{A(0),TTY}(1)'$ |
| 004 | AAAAAA | 1 | ZZ | N | 001 | NULL | NULL | NULL | NULL |
| 005 | AAAAAA | 1 | ZZ | NULL | 004 | S | DEL | X | $M_{ZZ}^{A(1),DEL}(1)'$ |
| 006 | BBBBBB | 0 | ZZ | NULL | 002 | V | TTY | T | $M_{ZZ}^{B(0),TTY}(1)'$ |
| *007* | *AAAAAA* | *1* | *ZZ* | *NULL* | *005* | *N* | *TTY* | *T* | $M_{ZZ}^{A(1),TTY}(2)'$ |
| 008 | BBBBBB | 0 | ZZ | NULL | 006 | S | ADD | X | $M_{ZZ}^{B(0),ADD}(2)'$ |

[0134]

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 009 | BBBBBB | 1 | YY | N | NULL | NULL | NULL | NULL | NULL |
| *010* | *BBBBBB* | *1* | *YY* | *NULL* | *009* | *N* | *TTY* | *Y* | $M_{ZZ}^{B(1),TTY}(1)'$ |
| 011 | AAAAAA | 1 | YY | N | 001 | NULL | NULL | NULL | NULL |
| *012* | *AAAAAA* | *1* | *YY* | *NULL* | *011* | *N* | *ADD* | *X* | $M_{YY}^{A(1),XML}(1)'$ |
| 013 | BBBBBB | 2 | YY | D | NULL | NULL | NULL | NULL | NULL |
| 014 | BBBBBB | 2 | YY | NULL | 013 | D | DEL | X | $M_{YY}^{B(2),DEL}(1)'$ |
| 015 | BBBBBB | 2 | YY | NULL | 014 | D | TTY | T | $M_{YY}^{B(2),TTY}(2)'$ |
| 016 | AAAAAA | 2 | ZZ | D | 004 | NULL | NULL | NULL | NULL |
| 017 | AAAAAA | 2 | YY | D | 011 | NULL | NULL | NULL | NULL |
| 018 | AAAAAA | 2 | ZZ | NULL | 007 | D | TTY | T | $M_{ZZ}^{A(2),TTY}(1)'$ |
| 019 | AAAAAA | 2 | YY | NULL | 017 | D | DEL | X | $M_{YY}^{A(2),DEL}(1)'$ |
| 020 | BBBBBB | 2 | YY | NULL | 015 | D | ADD | X | $M_{YY}^{B(2),ADD}(3)'$ |
| 021 | AAAAAA | 2 | ZZ | NULL | 018 | V | SPLIT | X | $M_{ZZ}^{A(2),SPLIT}(2)'$ |
| 022 | AAAAAA | 2 | YY | NULL | 019 | D | TTY | T | $M_{YY}^{A(2),TTY}(2)'$ |
| 023 | AAAAAA | 2 | YY | NULL | 017 | D | NDC | N | $M_{YY}^{A(2),NDC}(3)'$ |
| 024 | AAAAAA | 10 | YY | U | NULL | NULL | NULL | NULL | NULL |
| 025 | AAAAAA | 10 | YY | NULL | 024 | D | TTY | T | $M_{YY}^{A(10),TTY}(1)'$ |
| 026 | AAAAAA | 10 | YY | NULL | 025 | D | ADD | X | $M_{YY}^{A(10),ADD}(2)'$ |

[0135]

[0136] 정렬된 리스트 $\{M_{YY}^{A(2),n}(x)\}$는, ID들 019, 022 및 023을 갖는 테이블 엔트리들로부터 도출가능한, 메시지 우선순위 테이블(44)의 우선순위 값 정의에 기초하여 이루어진다. 이들의 메시지 엔벨로프 $R_{YY}^{A(2)}$는 ID 017을 갖는 테이블 엔트리로서 저장된다. 2개의 메시지들 $M_{YY}^{A(2),DEL}(1)'$ 및 $M_{YY}^{A(2),NDC}(3)'$는, 710 및 720에서의 체크들뿐만 아니라 동작들 712 및 722에 의해 야기된 바와 같이(도 9), 양측 모두의 메시지들을 $R_{YY}^{A(2)}$에 링크시키는 동일한 종속성 파라미터 D_ID를 갖는다.

[0137] 메시지 엔벨로프 $R_{ZZ}^{A(2)}$ 및 정렬된 리스트 $\{M_{ZZ}^{A(2),n}(x)\}$는 ID들 016, 018 및 021을 갖는 테이블 엔트리들로서 저장된다. $M_{ZZ}^{A(2),TTY}(1)'$는 도 8의 체크들 642 및 646뿐만 아니라 동작 648에 의해 출력된 $R_{ZZ}^{A(2)}$ 대신에 $M_{ZZ}^{A(1),TTY}(2)'$에 링크되는 그의 종속성 스테이터스 D_ID를 갖는다. $M_{ZZ}^{A(2),SPLIT}(2)'$의 종속성 스테이터스 D_ID는 도 9의 동작 730에서 018로서 연산되지만, 메시지 스테이터스 D_STAT는 체크 732 및 동작 736으로 인해 V로서 연산된다(도 9).

[0138] 메시지 엔벨로프 $R_{YY}^{B(2)}$ 및 메시지 리스트 $\{M_{YY}^{B(2),n}(x)\}$는 ID들 013, 014, 015 및 020을 갖는 테이블 엔트리들로서 저장된다. 메시지 엔벨로프 $R_{YY}^{A(10)}$ 및 메시지 리스트 $\{M_{YY}^{A(10),n}(x)\}$는 ID들 024 내지 026을 갖는 테이블 엔트리들로서 저장된다. 메시지 엔벨로프 $R_{YY}^{A(10)}$의 RD_STAT는 도 8의 630에서 U(미지수)로서 연산

되고(예를 들어, 이는, 어떤 이유로, 변경 커밋 표시(10)가 변경 커밋 표시(9) 전에 메시지 생성 및 시퀀싱 프로세스(42)에 도달하였고, 변경 커밋 표시(10) 내의 게시 콘텍스트가 이전 변경 커밋 표시가 존재한다는 것을 나타내고, 그 후에 메시지 엔벨로프 종속성 스테이터스가 변경 커밋 표시(9)의 도달을 대기하기 위해 U로 설정되기 때문이다. 미지수 스테이터스의 연산에 대한 세부사항들은, 본 명세서에 참조로 포함되는 EP 2254046 B1호에서 단락 [0031] 및 [0032] 그리고 도 7에 또한 설명되어 있다), 모든 메시지들은 D의 종속성 스테이터스 D_STAT를 갖는데, 이는 이들이 이들의 메시지 엔벨로프의 메시지 엔벨로프 종속성 RD_STAT가 N으로 변화될 때에 전송을 위해 단지 큐잉될 것이라는 것을 의미한다.

[0139] 이제 도 10을 참조하면, 메시지 전송 상태 머신(46)은 메시지 큐들(46a) 내에 큐잉되는 메시지들을 전송하고, 이들이 타깃(49)에 전송된 후에, 즉, 메시지 타깃들로부터 수신된 수신 확인들에 응답하여, 이들 메시지들의 종속성 스테이터스 D_STAT를 N으로부터 S로 업데이트할 뿐만 아니라 이들의 메시지 엔벨로프들 및 종속성 맵 및 메시지 테이블(45)에 유지된 메시지 엔벨로프들에서의 남아 있는 메시지들의 스테이터스를 업데이트하고, 전송하기에 적격인 추가 메시지들을 메시지 큐들(46a)에 최종적으로 놓는 것을 담당한다.

[0140] 메시지 전송 상태 머신(46)은 메시지가 메시지 큐(46a)에 존재할 때마다 활성화된다. 메시지 전송 상태 머신(46)은 메시지를 데큐잉하고, 전송자 프로세스(47)는 802에서 메시지 타입에 대해 정의된 확립된 통신 채널을 통해 올바른 메시지 타깃(49)에 메시지를 전송한다. 레거시 TTY 메시지들 및 XML 메시지가 상이한 통신 채널들을 통해 전송될 수도 있다(예를 들어, 전용된 채널을 사용하는 비동기식 방식의 TTY 메시지들 및 SOAP/HTTP를 사용하는 동기식 방식의 XML 메시지들). 일단 메시지 전송 방법(동기식 또는 비동기식)에 기초하여 메시지가 데큐잉 및 전송되면, 수신자 프로세스(48)는 804에서 타깃 시스템(동기식 모드의 경우)으로부터의 또는 예를 들어, EP 2693337A1호에 설명된 바와 같은, 내부 메시지 델리게이터 시스템(internal message delegator system)(비동기식 모드의 경우)으로부터의 확인 응답이 존재하는지 여부를 체크한다. 단지 메시지 타깃(49)으로부터 유효한 확인 응답이 수신되고 확인 응답 내에 에러 코드/정보가 제시되지 않을 때에만, 812에서 메시지가 "전송됨"(S)으로서 마킹된다. 동작들 806, 808 및 810에 의해 구성된 재시도 동작은, 비동기식 모드에서, 성공적으로 전달되지 않은 메시지들(수신된 에러 코드를 갖는 확인 응답 또는 메시지 송신 후 주어진 시간 간격 내에 수신된 유효한 확인 응답이 없음)에 대한 전송 동작들을 반복하기 위한 보안 방법이다. 선택적으로, 주어진 최대 재시도 횟수를 초과하는 메시지들은 이들이 최종적으로 전송되지 않은 경우에 복구를 위해 마킹되고 큐에 저장된다(도 10에 의해 도시되지 않음).

[0141] 804에서, 메시지 $M_j^{P(i),n}(y)'$에 대해 유효하고 올바른 응답이 수신된다고 결정될 때, 812에서 메시지의 종속성 스테이터스 D_STAT가 스테이터스 S(전송됨)로 업데이트된다.

[0142] 다음으로, $M_j^{P(i),n}(y)'$가 메시지 리스트 $\{M_j^{P(i),n}(x)'\}$에서의 마지막 메시지인지를 체크하기 위해 동작 814가 그 후에 호출된다. 예인 경우, 동작 816은 $R_j^{P(i)}$의 메시지 엔벨로프 레벨 종속성 스테이터스 RD_STAT를 스테이터스 S(전송됨)로 업데이트하고, 또한 동작 812의 D_STAT 업데이트를 저장한 종속성 맵 및 메시지 테이블(45)에 저장하였다. 그 후에, 818에서, 메시지 엔벨로프 레벨 종속성 체인 $P(i+1)$의 다음 메시지 엔벨로프 - 존재하는 경우 - 의 종속성 스테이터스가 RD_STAT = N(종속성 없음)으로 설정되는데, 이는 현재 메시지 엔벨로프의 모든 메시지들이 전송되었고, 따라서, 전체 메시지 엔벨로프가 완전히 프로세싱되었기 때문이다. 부가적으로, 여전히 818에서, 종속성 맵 및 메시지 테이블(45)에 따라, D_ID가 $R_j^{P(i+1)}$의 ID로 설정되는 모든 메시지들이 이들의 종속성 스테이터스 D_STAT를 N으로 설정하도록 업데이트된다. 818에서, 이들 메시지들은 또한 전송을 위해 메시지 큐(46a) 내에 놓여진다. 최종적으로, 종속성 맵 및 메시지 테이블(45)은 메시지 엔벨로프 레벨에서의 종속성 체인 내의 다음 메시지 엔벨로프 $R_j^{P(i+1)}$ 및 다음 메시지 리스트 $\{M_j^{P(i+1),n}(x)'\}$, 즉, 다음 메시지 엔벨로프의 메시지들에 대한 가장 최근의 변경사항들을 저장하기 위해 822에서 한 번 더 커밋된다.

[0143] 814에서 현재 메시지가 메시지 리스트의 마지막 메시지가 아니라고 결정되는 경우, 동작 820은 전송하기에 적격인 메시지들을 선택하도록 호출된다: D_ID가 $M_j^{P(i),n}(y)'$의 ID로 설정되는 모든 메시지들(즉, 타깃에 성공적으로 전달된 현재 메시지에 종속되는 모든 메시지들)이 종속성 맵 및 메시지 테이블(45)로부터 선택되고, 이들

의 종속성 스테이터스 D_STAT가 N으로 설정되고 이들 메시지들이 전송을 위해 메시지 큐(46a) 내에 놓여진다. 그 후에, 동작 822는 메시지 리스트들 $\{M_j^{P(i),n}(x)'\}$ 및 $\{M_j^{P(i+1),n}(x)'\}$에 대한 가장 최근의 변경사항들을 저장하기 위해 종속성 맵 및 메시지 테이블(45)을 커밋하도록 인보크(invoke)된다.

[0144] 메시지 전송 상태 머신(46)은 상기에 이미 논의된 테이블 II를 참조하여 추가로 예시될 수 있다. 먼저, 테이블 III를 갖는 도 11에 도시된 바와 같이, $M_{ZZ}^{A(1),TTY}(2)', M_{ZZ}^{B(1),TTY}(1)'$ 및 $M_{YY}^{A(1),XML}(1)'$의 성공적인 전송 후에 종속성 맵들이 업데이트되어, 그 포인트에서의 종속성 맵 및 메시지 테이블(45)의 상태를 나타낸다.

[0145] 메시지 리스트의, 즉, 메시지 엔벨로프 $R_j^{P(i)}$ 내의, 마지막 메시지가 메시지 타깃에 성공적으로 전송될 때, 종속성 맵 및 메시지 테이블(45)에서의 메시지 엔벨로프 테이블 엔트리가 업데이트되어 현재 메시지 엔벨로프 $R_j^{P(i)}$에 대한 메시지 전송 프로세스를 완료한다. 예를 들어, 다음의 테이블 엔트리들은, 도 11의 화살표들로 시각화된 바와 같이, 도 10의 동작 816에서 업데이트된다:

[0146] - ID = 007을 갖는 메시지의 성공적인 송신은 D_STAT = S로 설정되는데, 이는 ID = 004를 갖는 메시지 엔벨로프의 모든 메시지들이 전송되었다는 것을 의미한다. 이에 따라, 메시지 엔벨로프 레벨 종속성은 "전송됨"으로 또한 설정될 수 있다, 즉, 메시지 엔벨로프 ID = 004의 RD_STAT는 S로 설정된다.

[0147] - ID = 010을 갖는 메시지는 메시지 엔벨로프 ID = 009의 유일한 메시지이다. 따라서, 메시지 ID = 010이 성공적으로 송신될 때, 그의 D_STAT는 S로 설정되고 메시지 엔벨로프 레벨 종속성은 "전송됨"으로 또한 설정될 수 있다, 즉, 메시지 엔벨로프 ID = 009의 RD_STAT는 S로 설정된다.

[0148] - 마찬가지로, ID = 012를 갖는 메시지는 메시지 엔벨로프 ID = 011의 유일한 메시지이다. 따라서, 메시지 ID = 012가 성공적으로 송신될 때, 그의 D_STAT는 S로 설정되고 메시지 엔벨로프 레벨 종속성은 "전송됨"으로 또한 설정될 수 있다, 즉, 메시지 엔벨로프 ID = 011의 RD_STAT는 S로 설정된다.

[0149] 그 후에, 메시지 엔벨로프 레벨 종속성들의 업데이트들은 도 10의 동작 818에서 RD_STAT = S로 설정된 메시지 엔벨로프들 $R_j^{P(i)}$에 종속되는 다음 엔벨로프들 $R_j^{P(i+1)}$로 계속된다. 일반적으로, 메시지 엔벨로프 $R_j^{P(i+1)}$이 종속되는 메시지 엔벨로프 $R_j^{P(i)}$가 S로 설정될 때마다 다음 메시지 엔벨로프 $R_j^{P(i+1)}$의 RD_STAT는 N으로 업데이트된다. 도 11의 테이블 III의 예에서, 이들은 (다시 도 11의 각각의 화살표들로 표시되는) 다음의 천이(transition)들이다:

[0150] - 메시지 엔벨로프 ID = 004의 RD_STAT를 S로 설정하는 것에 응답하여, 메시지 엔벨로프 ID = 004(즉, D_ID = 004를 가짐)에 종속되는 ID = 016을 갖는 메시지 엔벨로프의 메시지 엔벨로프 레벨 종속성 스테이터스가 RD_STAT = N으로 설정된다.

[0151] - 메시지 엔벨로프 ID = 009의 RD_STAT를 S로 설정하는 것에 응답하여, 메시지 엔벨로프 ID = 009(즉, D_ID = 009를 가짐)에 종속되는 ID = 013을 갖는 메시지 엔벨로프의 메시지 엔벨로프 레벨 종속성 스테이터스가 RD_STAT = N으로 설정된다.

[0152] - 메시지 엔벨로프 ID = 011의 RD_STAT를 S로 설정하는 것에 응답하여, 메시지 엔벨로프 ID = 011(즉, D_ID = 011을 가짐)에 종속되는 ID = 017을 갖는 메시지 엔벨로프의 메시지 엔벨로프 레벨 종속성 스테이터스가 RD_STAT = N으로 설정된다.

[0153] 메시지 엔벨로프 레벨 종속성 스테이터스 RD_STAT의 이들 업데이트들에 응답하여, 업데이트된 메시지 엔벨로프들의 제1 메시지들, 즉, $R_j^{P(i+1)}$에 대한 종속성을 갖는 메시지들은 도 10의 동작 818에서 전송을 위해 업데이트 및 큐잉된다. 예를 들어, 다시 도 11을 참조하면, 다음의 메시지들이 더 이상 종속되지 않는 것으로 마킹된다:

[0154] - 메시지 엔벨로프 ID = 013의 RD_STAT를 N으로 설정하는 것에 응답하여, 메시지 엔벨로프 ID = 013(즉, D_ID = 013을 가짐)에서의 제1 메시지인 ID = 014를 갖는 메시지의 메시지 레벨 종속성 스테이터스는

D_STAT = N으로 설정되고 메시지 큐(46a)로 전달된다.

[0155] - 메시지 엔벨로프 ID = 017의 RD_STAT를 N으로 설정하는 것에 응답하여, 메시지 엔벨로프 ID = 017(즉, D_ID = 017을 가짐, 즉, 2개의 메시지들 사이에 종속성이 없다)에서 양측 모두가 제1 메시지들인 ID = 019 및 ID = 023을 갖는 이들 2개의 메시지들의 메시지 레벨 종속성 스테이터스는 D_STAT = N으로 설정되고 양측 모두의 메시지들은 메시지 큐(46a)로 전달된다.

[0156] 도 10의 동작 818은 또한, 성공적으로 전송된 메시지들 및 N(독립적)으로 설정된 메시지들로 인해 업데이트되어야 하는 레거시 메시지 레벨 종속성들이 존재하는지를 체크한다. 예를 들어, 변경 커밋 표시 2(예를 들어, I = 2)와 관련된 레거시 표준 메시지 ID = 018은, 메시지 엔벨로프 ID = 004에 속하고 성공적으로 전송된 메시지 ID = 007(예를 들어, D_ID = 007)에 대한 메시지 레벨 종속성을 갖는다. 따라서, 메시지 ID = 018의 종속성 스테이터스는 N으로 설정되고 메시지 ID = 018은 또한 전송을 위해 메시지 큐(46a)로 전달된다.

[0157] 완전성을 위해, 레거시 메시지 ID = 018은 특수 경우인데, 이는 그것이 동일한 데이터베이스 레코드에 대한 각각의 메시지 타깃에 대한 레거시 TTY 메시지들 간의 직접적인 메시지 레벨 종속성 체인을 구축하기 위한 메시지 프로세싱 엔진(50)의 하나의 임의적 기능이기 때문이라는 것에 주목해야 한다. 메시지 엔벨로프 ID = 004에서의 레거시 메시지 ID = 007은, 메시지 엔벨로프 ID = 004 내의 메시지 ID = 005에 대한 메시지 레벨 종속성을 갖는 통상적인 메시지이다. 다른 한편으로, 레거시 메시지 ID = 018은 실제로 메시지 엔벨로프 ID = 016에 속하고 그에 따라 그의 D_ID는 016이어야 한다. 그러나, 그것은 레거시 메시지이기 때문에, 특정 레거시 종속성 연산 프로세스는 D_ID = 016 대신에 그의 D_ID = 007을 설정하는데, 이는 레거시 메시지 = 007 바로 다음에 레거시 메시지 ID = 018을 전송하는 것이 메시지 엔벨로프에서의 임의의 잠재적인 다른 XML 메시지들(여기서는: ID = 21을 갖는 메시지)의 시퀀스 및 종속성에 영향을 미치지 않기 때문이다, 즉, 메시지 레벨 종속성 체인이 NDC 플래깅된 변경 커밋 표시들에 대해 유지되기 때문이다.

[0158] 상기에 언급된 제1 메시지들(ID들 014, 019, 023 및 018을 갖는 메시지들)이 성공적으로 전송된 후에 종속성 맵 및 메시지 테이블(45)에 의해 정의된 종속성 맵들이 도 12의 테이블 IV에 나타내어진다. ID들 014, 019, 023 및 018을 갖는 메시지들의 메시지 레벨 종속성 스테이터스는 수신자 프로세스(48)가 각각의 수신 확인 응답들을 수신하는 것에 응답하여 D_STAT = S로 설정된다.

[0159] 다음으로, 도 10의 동작 820은 종속성 맵 및 메시지 테이블(45)에 의해 정의되는 메시지 레벨 종속성 체인에 기초하여 각각의 메시지 엔벨로프들에서의 다음 메시지들의 메시지 레벨 종속성 스테이터스를 업데이트한다. 도 12의 예에서, 다음의 종속성 업데이트들이 메시지 레벨에서 행해진다:

[0160] - 메시지 ID = 014의 성공적인 송신에 응답하여, 메시지 ID = 014의 메시지 레벨 종속성 스테이터스는 "전송됨", 즉, D_STAT = S로 설정된다. 그 후에, 메시지 ID = 0014에 종속되는 다음 메시지, 즉, ID = 015를 갖는 메시지(D_ID_014를 가짐)의 메시지 레벨 종속성 스테이터스는 독립적인 것, 즉, D_STAT = N으로 설정되고, 메시지 ID_015가 전송을 위해 메시지 큐(46a)로 전달된다.

[0161] - 마찬가지로, 메시지 ID = 019의 성공적인 전달에 응답하여, 메시지 ID = 019의 메시지 레벨 종속성 스테이터스는 "전송됨", 즉, D_STAT = S로 설정된다. 그 후에, 메시지 ID = 0019에 종속되는 다음 메시지, 즉, ID = 022를 갖는 메시지(D_ID_019를 가짐)의 메시지 레벨 종속성 스테이터스는 독립적인 것, 즉, D_STAT = N으로 설정되고, 메시지 ID_022가 전송을 위해 메시지 큐(46a)로 전달된다.

[0162] 추가로, 메시지 엔벨로프 레벨에서, ID = 016을 갖는 메시지 엔벨로프$(R_{ZZ}^{A(2)})$는 S로 업데이트되는데, 이는 ID = 018을 갖는 메시지$(M_{ZZ}^{A(2),TTY}(1)')$가 종속성 체인에서 메시지 타깃 ZZ에 전송될 유일한 메시지이고(메시지 ID = 021이 또한 메시지 엔벨로프 ID = 016에 속하지만, 보이드 스테이터스(D_STAT = V)를 가지며 그에 따라 전송되지 않을 것이다) 메시지 ID = 018이 전송되었기 때문이다.

[0163] 상기에 언급된 모든 메시지들이 전송된 후에 종속성 맵 및 메시지 테이블(45)에 의해 정의된 종속성 맵들이 테이블 V를 갖는 도 13에 나타내어진다. 도 13은 메시지 엔벨로프 레벨 종속성들 및 메시지 레벨 종속성들의 다음의 업데이트들을 또한 표시한다:

[0164] - ID_022를 갖는 메시지가 성공적으로 전송된 후에, 메시지 레벨 종속성 스테이터스 D_STAT가 S로 설정된다. 이것은 ID = 017을 갖는 메시지 엔벨로프의 모든 메시지들이 성공적으로 전달되었다는 것을 또한 의미한

다. 이에 따라, 메시지 엔벨로프 레벨 종속성은 "전송됨"으로 또한 설정될 수 있다, 즉, 메시지 엔벨로프 ID = 017의 RD_STAT는 N으로부터 S로 설정된다.

[0165] - ID_015를 갖는 메시지가 성공적으로 전송된 후에, 메시지 레벨 종속성 스테이터스 D_STAT가 S로 설정된다. 메시지 ID_015에 종속되는 다음 메시지는, ID = 020을 갖는 메시지(D_ID = 015를 가짐)이다. 따라서, 메시지 ID = 015의 메시지 레벨 종속성이 N으로 설정되고 메시지는 전송을 위해 메시지 큐(46a)로 전달된다.

[0166] - 그 후에, 도 13은 메시지 타깃 YY로부터 대응하는 확인 응답을 수신하는 것에 응답하여 메시지 ID = 020의 메시지 레벨 종속성 스테이터스가 추후에 "전송됨"으로 설정될 것임을 "(S)"로 이미 표시한다.

[0167] - 그 후에, 도 13은, 메시지 ID_020의 성공적인 전달을 결정하는 것에 응답하여, 메시지 엔벨로프 ID = 013의 메시지 엔벨로프 종속성 스테이터스가 "전송됨"(S)으로 설정될 것임을 또한 표시하는데, 이는 메시지 ID = 020이 이 메시지 엔벨로프의 마지막 메시지였기 때문이다.

[0168] 최종적으로, 도 14는 상기에 설명된 메시지 프로세싱 시스템(50)의 기능성을 제공하는 컴퓨터 시스템(100)의 개략적 표현이다. 컴퓨터 시스템(100)은, 컴퓨터 시스템(100)으로 하여금, 상기에 상술된 바와 같이 메시지 프로세싱 시스템(50)의 임의의 방법들 및 활동들 중 임의의 것을 수행하게 하는 명령어들의 세트를 포함할 수도 있다. 컴퓨터 시스템(100)은, 버스(104)를 통해 서로 통신하는 하나 이상의 프로세서들(101), 메인 메모리(102) 및 네트워크 인터페이스 디바이스(103)를 포함한다. 적어도 하나의 프로세서(101)는, 본 명세서에서 설명되는 메시지 프로세싱 방법론들을 수행하기 위한 프로그램 명령어들을 유지하는 적어도 하나의 CPU 캐시를 포함한다. 선택적으로, 컴퓨터 시스템(100)은 정적 메모리(105) 및 디스크 드라이브 유닛(106)을 더 포함할 수도 있다. 비디오 디스플레이(107), 영숫자 입력 디바이스(108) 및 커서 제어 디바이스(109)는 컴퓨터 시스템(100)을 동작시키기 위해 인간-머신 인터페이스를 구성한다. 네트워크 인터페이스 디바이스(103)는, 메시지 프로세싱 시스템(50)을, NDC 예약 엔진(40), 레거시 GDS 예약 엔진(41) 및 메시지 타깃들(49)을 포함하는 임의의 수 및 종류들의 다른 시스템들에 연결하는 유선 및/또는 무선 인터페이스이다. 메시지 타깃들(49)은 인터넷 및/또는 임의의 다른 네트워크 상에 상주할 수도 있다. 네트워크 인터페이스 디바이스(103)는 표준 통신 프로토콜들, 예컨대, HTTP/TCP/IP 프로토콜 스택, IEEE 802.11 및/또는 독점적 통신 프로토콜들을 이용한다.

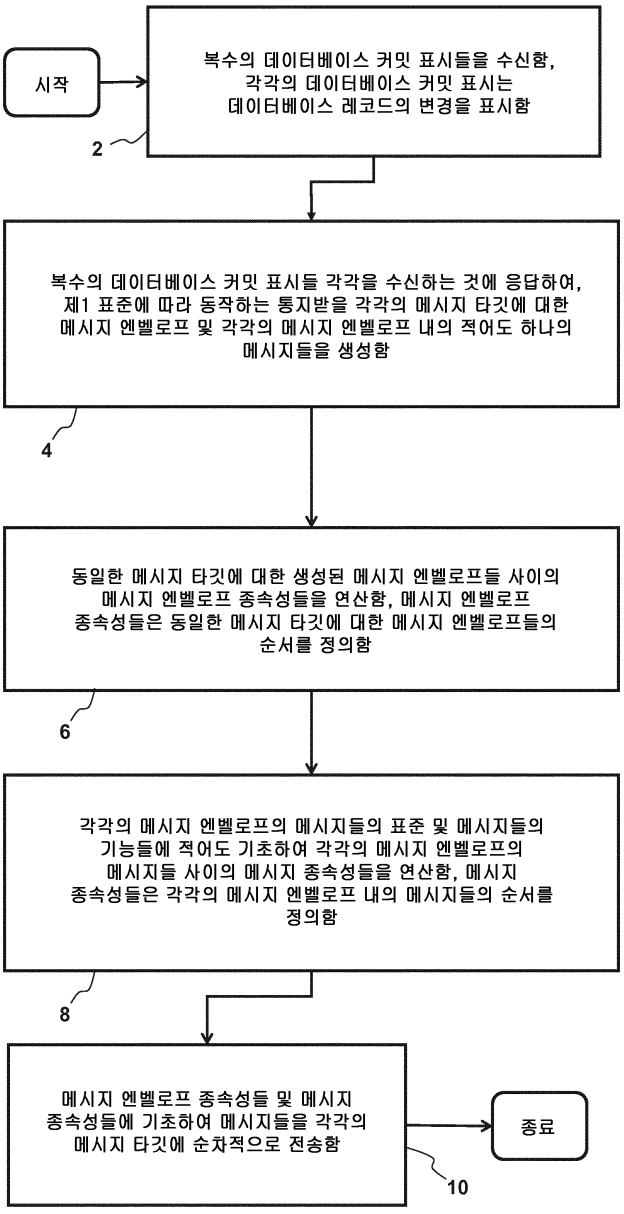
[0169] 상기에 설명된 방법론들 중 어느 하나 또는 전부를 구체화하는 명령어들의 세트(즉, 소프트웨어)(110)는, 머신 판독가능 매체, 예를 들어, 적어도 하나의 CPU 캐시를 갖는 적어도 하나의 프로세서(101) 및/또는 메인 메모리(102) 내에 또는 그 상에 완전히 또는 적어도 부분적으로 상주한다. 그 중에서도, 명령어들은 메시지 프로세싱 시스템의 기능들을 구현하여, 들어오는 데이터베이스 변경 커밋들을 프로세싱하고, 메시지 엔벨로프들 및 메시지들을 생성할 뿐만 아니라, 상기에 상세히 설명된 바와 같이 메시지들을 큐잉 및 전송할 수도 있다. 소프트웨어(110)가 상주하는 머신 판독가능 매체는 또한, 디스크 드라이브 유닛(106)의 일부인 비휘발성 데이터 캐리어(111)(예를 들어, 비분리형 자기 하드 디스크 또는 광학 또는 자기 제거가능 디스크)일 수도 있다. 소프트웨어(110)는 네트워크 인터페이스 디바이스(103)에 의해 인터넷을 통해 전파된 신호(112)로서 추가로 송신 또는 수신될 수도 있다.

[0170] 요약하면, 본 명세서에서 설명되는 방법론들은 메시지 엔벨로프 레벨 및 메시지 레벨 양측 모두에서 종속성 체인의 생성을 용이하게 한다. 메시지 엔벨로프들과 메시지들 사이의 종속성 관계들은 주어진 메시지 우선순위 값들을 (재)정의함으로써 구성가능하다. 전반적으로, 예를 들어, TTY 메시지들에 기초하는 레거시 메시지 흐름들뿐만 아니라 NDC와 같은 최신 표준에 따른 더 정교한 하이브리드 메시지 흐름들 양측 모두를 지원하는 일반 프레임워크가 제시된다. 이 프레임워크는 장래의 훨씬 더 정교한 표준들 및 메시지 흐름들이 끊김없이 부가될 수 있기 때문에 확장가능하다.

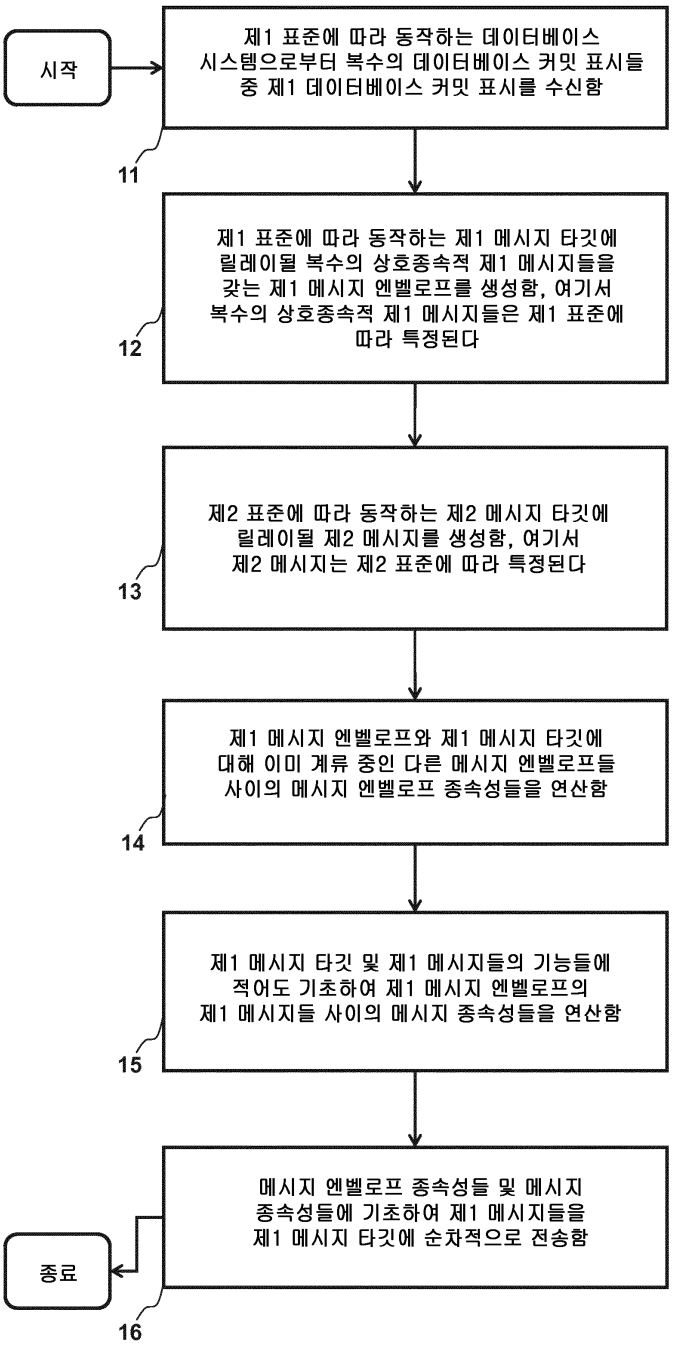
[0171] 본 발명의 교시에 따라 구성된 특정 제품들 및 방법들이 본 명세서에 설명되었지만, 본 특허의 커버리지의 범주는 이에 제한되지 않는다. 반대로, 본 특허는 문자 그대로 또는 등가물들의 원칙(doctrine) 하에서 첨부된 청구범위의 범주 내에 완전히 속하는 본 발명의 교시의 모든 실시예들을 커버한다.

## 도면

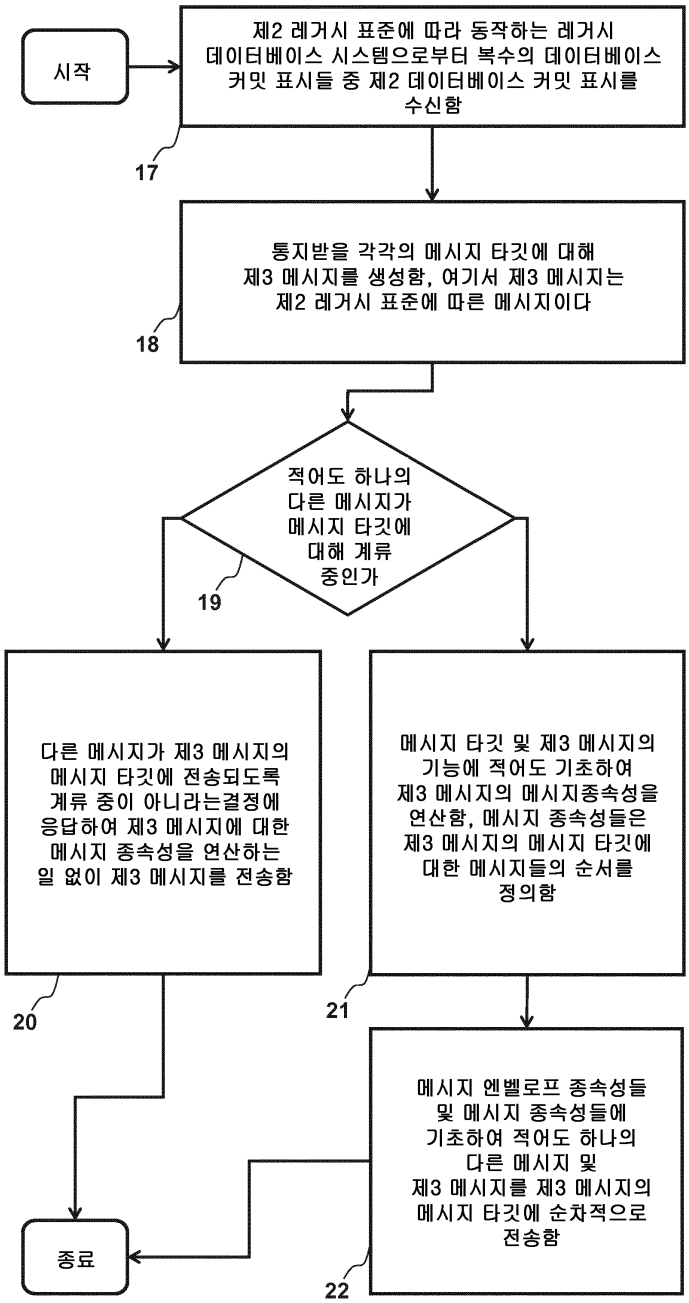
### 도면1


시작
복수의 데이터베이스 커밋 표시들을 수신함, 각각의 데이터베이스 커밋 표시는 데이터베이스 레코드의 변경을 표시함
2
복수의 데이터베이스 커밋 표시들 각각을 수신하는 것에 응답하여, 제1 표준에 따라 동작하는 통지받을 각각의 메시지 타깃에 대한 메시지 엔벨로프 및 각각의 메시지 엔벨로프 내의 적어도 하나의 메시지들을 생성함
4
동일한 메시지 타깃에 대한 생성된 메시지 엔벨로프들 사이의 메시지 엔벨로프 종속성들을 연산함, 메시지 엔벨로프 종속성들은 동일한 메시지 타깃에 대한 메시지 엔벨로프들의 순서를 정의함
6
각각의 메시지 엔벨로프의 메시지들의 표준 및 메시지들의 기능들에 적어도 기초하여 각각의 메시지 엔벨로프의 메시지들 사이의 메시지 종속성들을 연산함, 메시지 종속성들은 각각의 메시지 엔벨로프 내의 메시지들의 순서를 정의함
8
메시지 엔벨로프 종속성들 및 메시지 종속성들에 기초하여 메시지들을 각각의 메시지 타깃에 순차적으로 전송함
종료
10

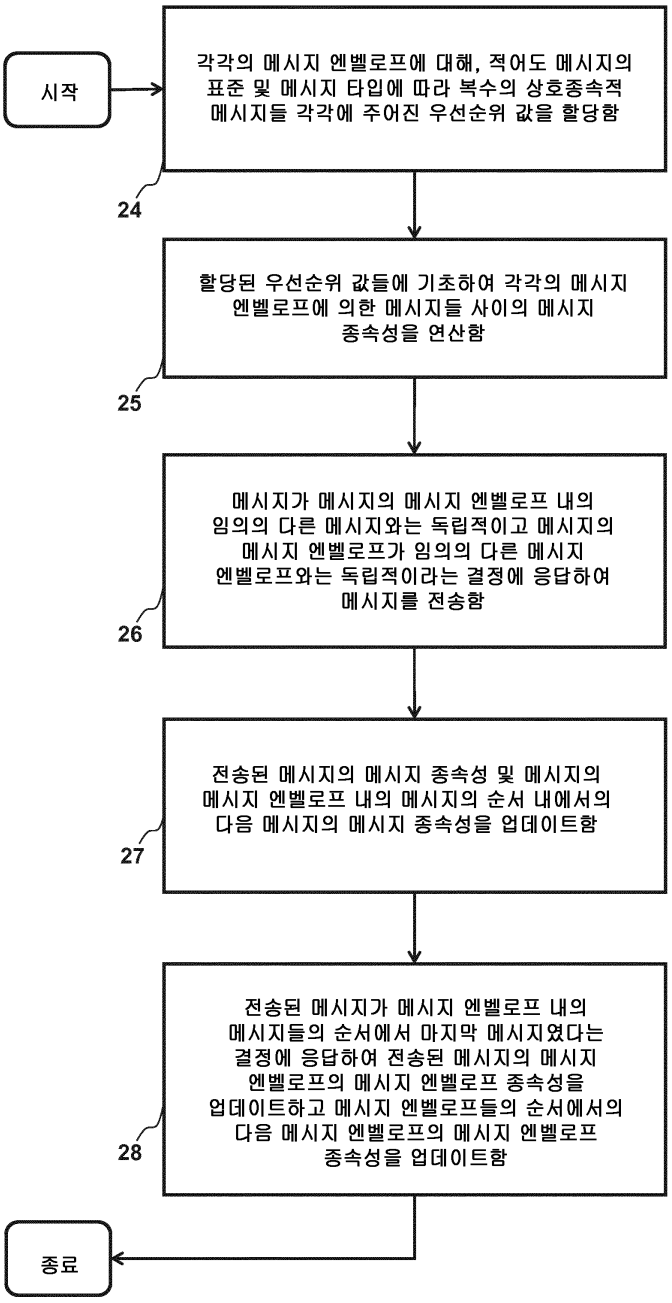
도면2


시작
제1 표준에 따라 동작하는 데이터베이스 시스템으로부터 복수의 데이터베이스 커밋 표시들 중 제1 데이터베이스 커밋 표시를 수신함
11
제1 표준에 따라 동작하는 제1 메시지 타깃에 릴레이될 복수의 상호종속적 제1 메시지들을 갖는 제1 메시지 엔벨로프를 생성함, 여기서 복수의 상호종속적 제1 메시지들은 제1 표준에 따라 특정된다
12
제2 표준에 따라 동작하는 제2 메시지 타깃에 릴레이될 제2 메시지를 생성함, 여기서 제2 메시지는 제2 표준에 따라 특정된다
13
제1 메시지 엔벨로프와 제1 메시지 타깃에 대해 이미 계류 중인 다른 메시지 엔벨로프들 사이의 메시지 엔벨로프 종속성들을 연산함
14
제1 메시지 타깃 및 제1 메시지들의 기능들에 적어도 기초하여 제1 메시지 엔벨로프의 제1 메시지들 사이의 메시지 종속성들을 연산함
15
메시지 엔벨로프 종속성들 및 메시지 종속성들에 기초하여 제1 메시지들을 제1 메시지 타깃에 순차적으로 전송함
종료
16

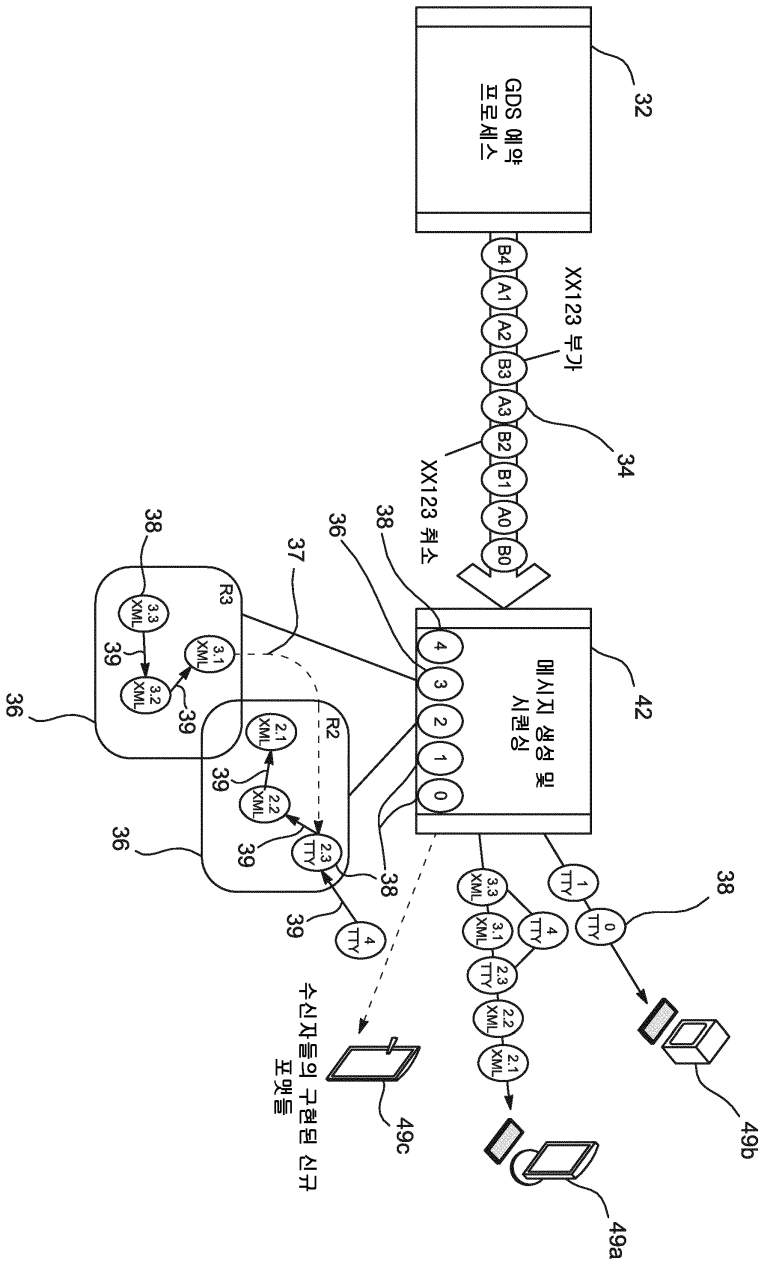
도면3


시작
제2 레거시 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신함
17
통지받을 각각의 메시지 타깃에 대해 제3 메시지를 생성함, 여기서 제3 메시지는 제2 레거시 표준에 따른 메시지이다
18
적어도 하나의 다른 메시지가 메시지 타깃에 대해 계류 중인가
19
다른 메시지가 제3 메시지의 메시지 타깃에 전송되도록 계류 중이 아니라는결정에 응답하여 제3 메시지에 대한 메시지 종속성을 연산하는 일 없이 제3 메시지를 전송함
20
메시지 타깃 및 제3 메시지의 기능에 적어도 기초하여 제3 메시지의 메시지종속성을 연산함, 메시지 종속성들은 제3 메시지의 메시지 타깃에 대한 메시지들의 순서를 정의함
21
메시지 엔벨로프 종속성들 및 메시지 종속성들에 기초하여 적어도 하나의 다른 메시지 및 제3 메시지를 제3 메시지의 메시지 타깃에 순차적으로 전송함
22
종료

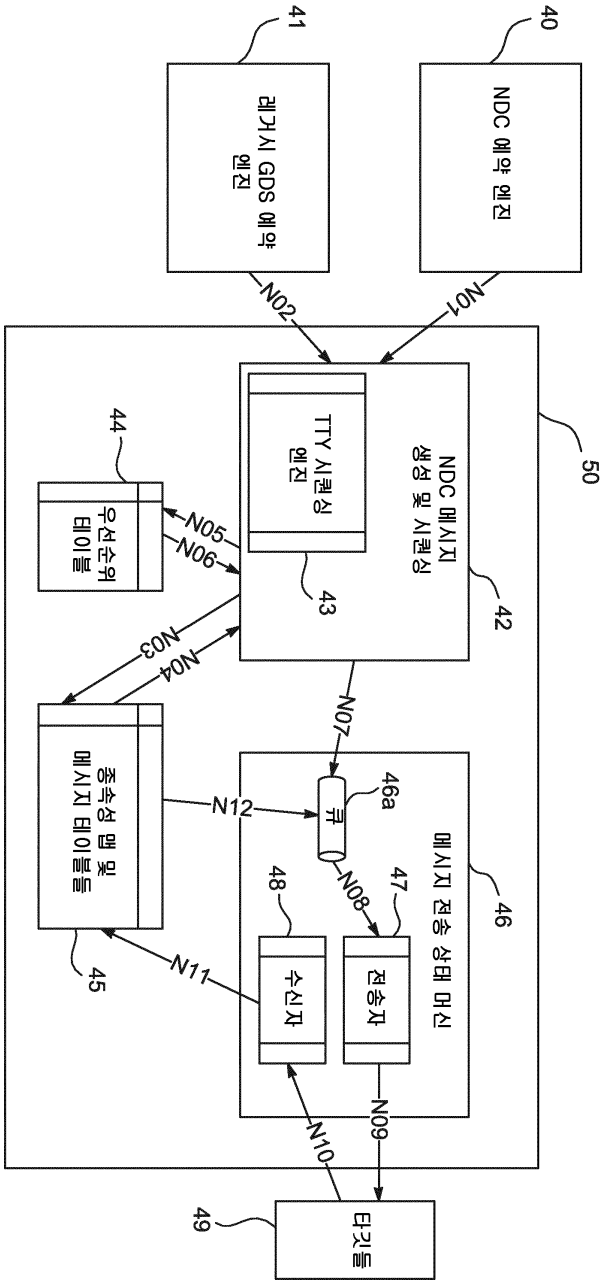
도면4


시작
각각의 메시지 엔벨로프에 대해, 적어도 메시지의 표준 및 메시지 타입에 따라 복수의 상호종속적 메시지들 각각에 주어진 우선순위 값을 할당함
24
할당된 우선순위 값들에 기초하여 각각의 메시지 엔벨로프에 의한 메시지들 사이의 메시지 종속성을 연산함
25
메시지가 메시지의 메시지 엔벨로프 내의 임의의 다른 메시지와는 독립적이고 메시지의 메시지 엔벨로프가 임의의 다른 메시지 엔벨로프와는 독립적이라는 결정에 응답하여 메시지를 전송함
26
전송된 메시지의 메시지 종속성 및 메시지의 메시지 엔벨로프 내의 메시지의 순서 내에서의 다음 메시지의 메시지 종속성을 업데이트함
27
전송된 메시지가 메시지 엔벨로프 내의 메시지들의 순서에서 마지막 메시지였다는 결정에 응답하여 전송된 메시지의 메시지 엔벨로프의 메시지 엔벨로프 종속성을 업데이트하고 메시지 엔벨로프들의 순서에서의 다음 메시지 엔벨로프의 메시지 엔벨로프 종속성을 업데이트함
28
종료

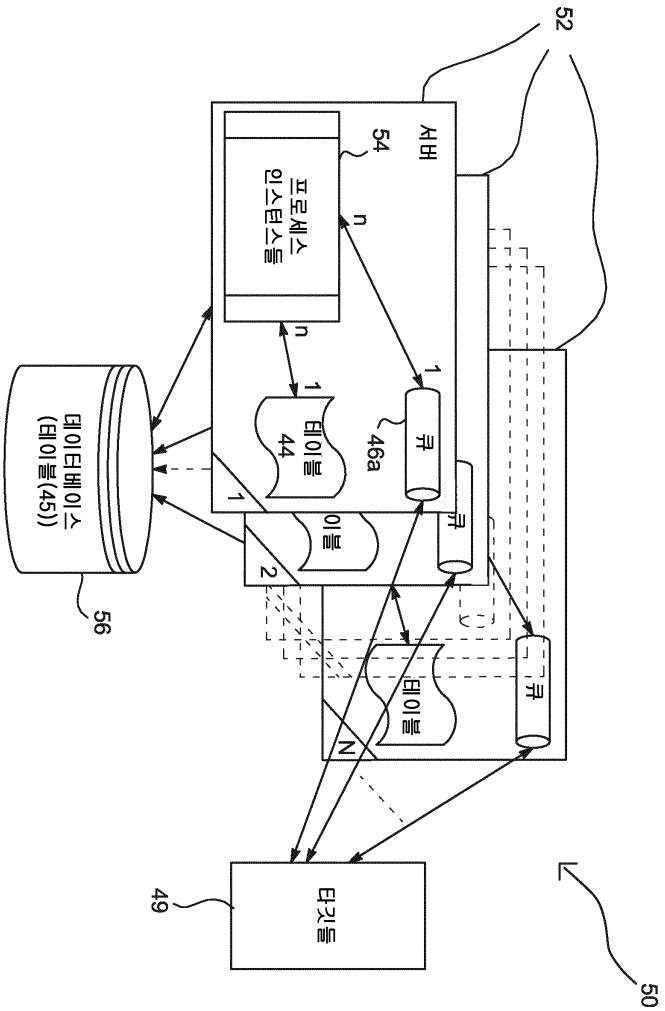
도면5


GDS 예약 프로세스
32
XX123 부가
XX123 취소
34
메시지 생성 및 시퀀싱
42
R3
R2
36
37
38
39
수신자들의 구성된 신규 포맷들
49a
49b
49c

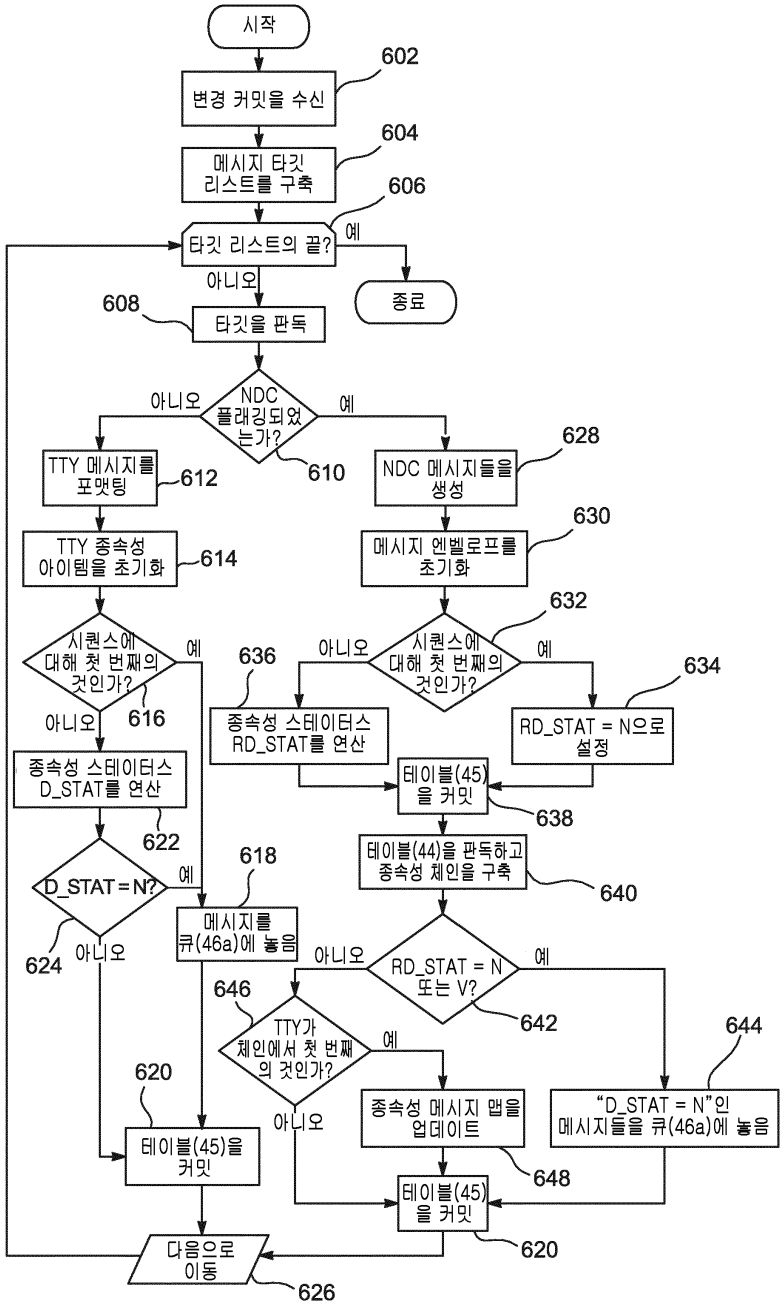
도면6


41
레거시 GDS 예약 엔진
40
NDC 예약 엔진
N02
N01
50
TTY 시퀀싱 엔진
NDC 메시지 생성 및 시퀀싱
42
43
44
우선순위 테이블
N05
N06
N03
N04
N07
종속성 맵 및 메시지 테이블들
45
N12
46a
큐
N08
48
47
수신자
전송자
메시지 전송 상태 머신
46
N11
N10
N09
49
타깃들

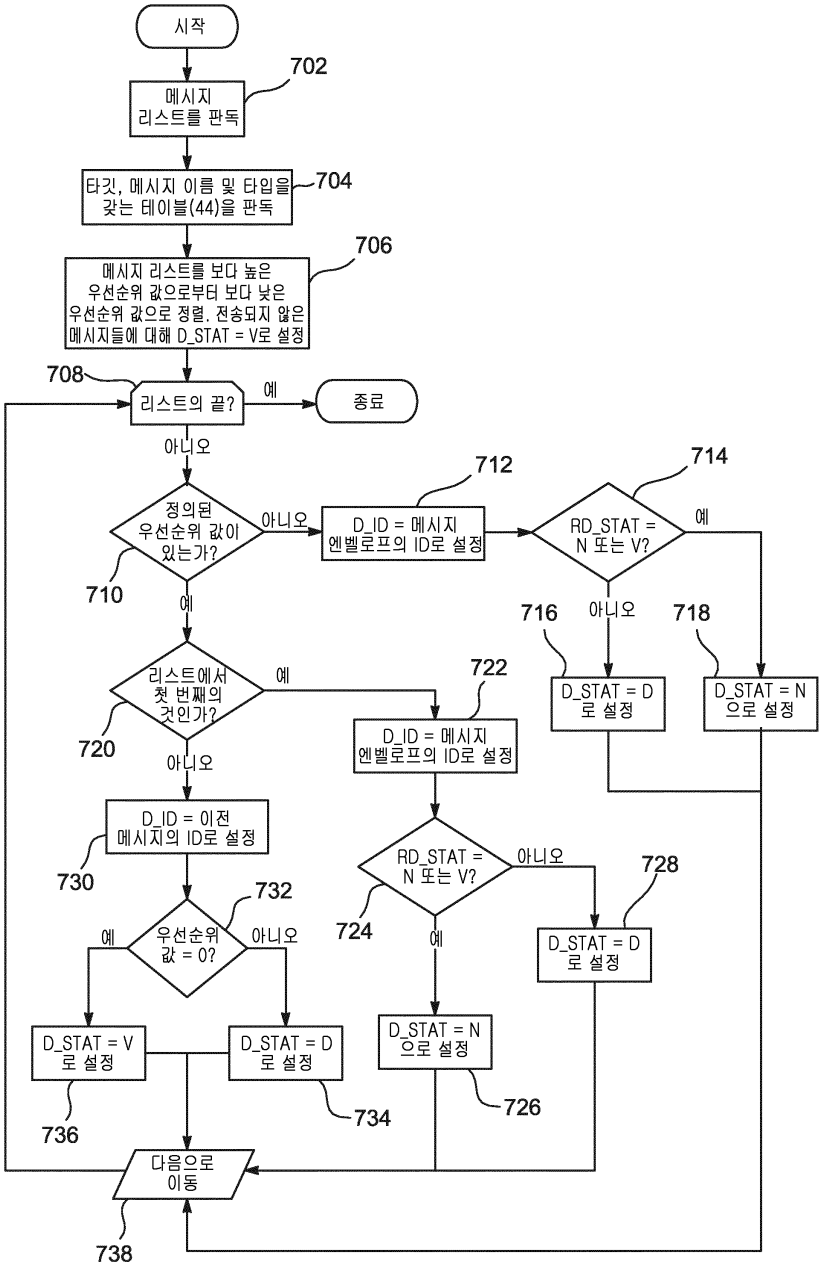
도면7


52
서버
54
프로세스
인스턴스들
n
n
1
1
테이블
44
46a
큐
큐
이블
1
2
데이터베이스
(테이블(45))
56
테이블
N
큐
타깃들
49
50

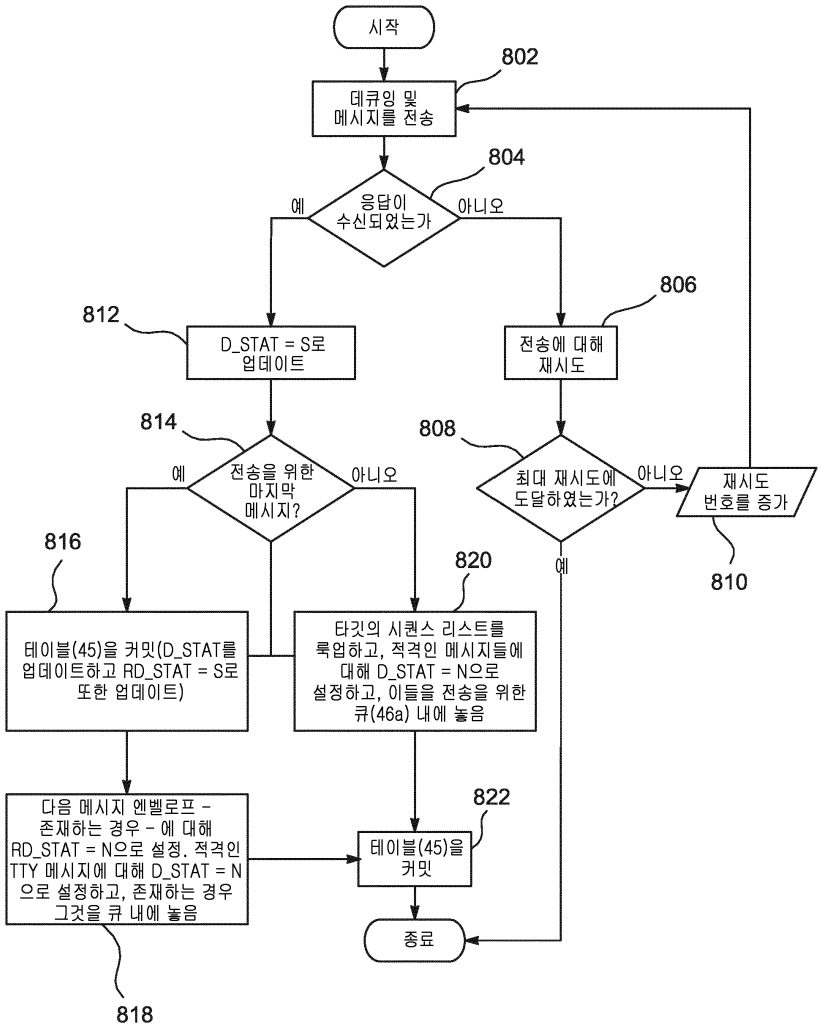
도면8


시작
602
변경 커밋을 수신
604
메시지 타깃 리스트를 구축
606
타깃 리스트의 끝?
예
아니오
종료
608
타깃을 판독
아니오
NDC 플래깅되었는가?
예
610
TTY 메시지를 포맷팅
612
NDC 메시지들을 생성
628
TTY 종속성 아이템을 초기화
614
메시지 엔벨로프를 초기화
630
632
시퀀스에 대해 첫 번째의 것인가?
예
아니오
636
종속성 스테이터스 RD_STAT를 연산
634
RD_STAT = N으로 설정
616
종속성 스테이터스 D_STAT를 연산
622
테이블(45)을 커밋
638
테이블(44)을 판독하고 종속성 체인을 구축
640
D_STAT = N?
예
618
메시지를 큐(46a)에 놓음
624
아니오
RD_STAT = N 또는 V?
642
646
TTY가 체인에서 첫 번째의 것인가?
644
620
종속성 메시지 맵을 업데이트
"D_STAT = N"인 메시지들을 큐(46a)에 놓음
648
테이블(45)을 커밋
620
다음으로 이동
626

도면9


시작
702
메시지 리스트를 판독
704
타깃, 메시지 이름 및 타입을 갖는 테이블(44)을 판독
706
메시지 리스트를 보다 높은 우선순위 값으로부터 보다 낮은 우선순위 값으로 정렬. 전송되지 않은 메시지들에 대해 D_STAT = V로 설정
708
리스트의 끝?
예
종료
아니오
712
정의된 우선순위 값이 있는가?
710
아니오
D_ID = 메시지 엔벨로프의 ID로 설정
714
RD_STAT = N 또는 V?
예
716
아니오
718
D_STAT = D 로 설정
D_STAT = N 으로 설정
예
722
리스트에서 첫 번째의 것인가?
720
예
D_ID = 메시지 엔벨로프의 ID로 설정
아니오
D_ID = 이전 메시지의 ID로 설정
730
RD_STAT = N 또는 V?
아니오
728
724
예
D_STAT = D 로 설정
732
우선순위 값 = 0?
예
아니오
D_STAT = V 로 설정
D_STAT = D 로 설정
D_STAT = N 으로 설정
736
734
726
다음으로 이동
738

도면10


시작
802
데큐잉 및 메시지를 전송
804
응답이 수신되었는가
예
아니오
806
812
D_STAT = S로 업데이트
전송에 대해 재시도
814
808
전송을 위한 마지막 메시지?
예
아니오
최대 재시도에 도달하였는가?
아니오
재시도 번호를 증가
810
816
820
예
테이블(45)을 커밋(D_STAT를 업데이트하고 RD_STAT = S로 또한 업데이트)
타깃의 시퀀스 리스트를 룩업하고, 적격인 메시지들에 대해 D_STAT = N으로 설정하고, 이들을 전송을 위한 큐(46a) 내에 놓음
822
다음 메시지 엔벨로프 - 존재하는 경우 - 에 대해 RD_STAT = N으로 설정. 적격인 TTY 메시지에 대해 D_STAT = N으로 설정하고, 존재하는 경우 그것을 큐 내에 놓음
테이블(45)을 커밋
종료
818

도면11

테이블 III. 메시지 엔벨로프 1의 메시지들을 전송한 후의 종속성 맵들.

| ID | P | I | J | RD_STAT | D_ID | D_STAT | N | 타입 | 메시지 |
|---|---|---|---|---|---|---|---|---|---|
| 001 | AAAAAA | 0 | YY | S | NULL | NULL | NULL | NULL | NULL |
| 002 | BBBBBB | 0 | ZZ | S | NULL | NULL | NULL | NULL | NULL |
| 003 | AAAAAA | 0 | YY | NULL | 001 | S | TTY | T | $M_{YY}^{A(0),TTY}(1)'$ |
| 004 | AAAAAA | 1 | ZZ | ***S*** | 001 | NULL | NULL | NULL | NULL |
| 005 | AAAAAA | 1 | ZZ | NULL | 004 | S | DEL | X | $M_{ZZ}^{A(1),DEL}(1)'$ |
| 006 | BBBBBB | 0 | ZZ | NULL | 002 | V | TTY | T | $M_{ZZ}^{B(0),TTY}(1)'$ |
| 007 | AAAAAA | 1 | ZZ | NULL | 005 | ***S*** | TTY | T | $M_{ZZ}^{A(1),TTY}(2)'$ |
| 008 | BBBBBB | 0 | ZZ | NULL | 006 | S | ADD | X | $M_{ZZ}^{B(0),ADD}(2)'$ |
| 009 | BBBBBB | 1 | YY | ***S*** | 002 | NULL | NULL | NULL | NULL |
| 010 | BBBBBB | 1 | YY | NULL | 009 | ***S*** | TTY | Y | $M_{ZZ}^{B(1),TTY}(1)'$ |
| 011 | AAAAAA | 1 | YY | ***S*** | 001 | NULL | NULL | NULL | NULL |
| 012 | AAAAAA | 1 | YY | NULL | 011 | ***S*** | ADD | X | $M_{YY}^{A(1),XML}(1)'$ |
| 013 | BBBBBB | 2 | YY | ***N*** | 009 | NULL | NULL | NULL | NULL |
| *014* | *BBBBBB* | *2* | *YY* | *NULL* | *013* | ***N*** | *DEL* | *X* | $M_{YY}^{B(2),DEL}(1)'$ |
| 015 | BBBBBB | 2 | YY | NULL | 014 | D | TTY | T | $M_{YY}^{B(2),TTY}(2)'$ |
| 016 | AAAAAA | 2 | ZZ | ***N*** | 004 | NULL | NULL | NULL | NULL |
| 017 | AAAAAA | 2 | YY | ***N*** | 011 | NULL | NULL | NULL | NULL |
| *018* | *AAAAAA* | *2* | *ZZ* | *NULL* | *007* | ***N*** | *TTY* | *T* | $M_{ZZ}^{A(2),TTY}(1)'$ |
| *019* | *AAAAAA* | *2* | *YY* | *NULL* | *017* | ***N*** | *DEL* | *X* | $M_{YY}^{A(2),DEL}(1)'$ |
| 020 | BBBBBB | 2 | YY | NULL | 015 | D | ADD | X | $M_{YY}^{B(2),ADD}(3)'$ |
| 021 | AAAAAA | 2 | ZZ | NULL | 018 | V | SPLIT | X | $M_{ZZ}^{A(2),SPLIT}(2)'$ |
| 022 | AAAAAA | 2 | YY | NULL | 019 | D | TTY | T | $M_{YY}^{A(2),TTY}(2)'$ |
| *023* | *AAAAAA* | *2* | *YY* | *NULL* | *017* | ***N*** | *NDC* | *N* | $M_{YY}^{A(2),NDC}(3)'$ |
| 024 | AAAAAA | 10 | YY | U | ? | NULL | NULL | NULL | NULL |
| 025 | AAAAAA | 10 | YY | NULL | 024 | D | TTY | T | $M_{YY}^{A(10),TTY}(1)'$ |
| 026 | AAAAAA | 10 | YY | NULL | 025 | D | SPLIT | X | $M_{YY}^{A(10),SPLIT}(2)'$ |

도면12

테이블 IV. 메시지 엔벨로프 2의 제1 메시지들을 전송한 후의 종속성 맵들.

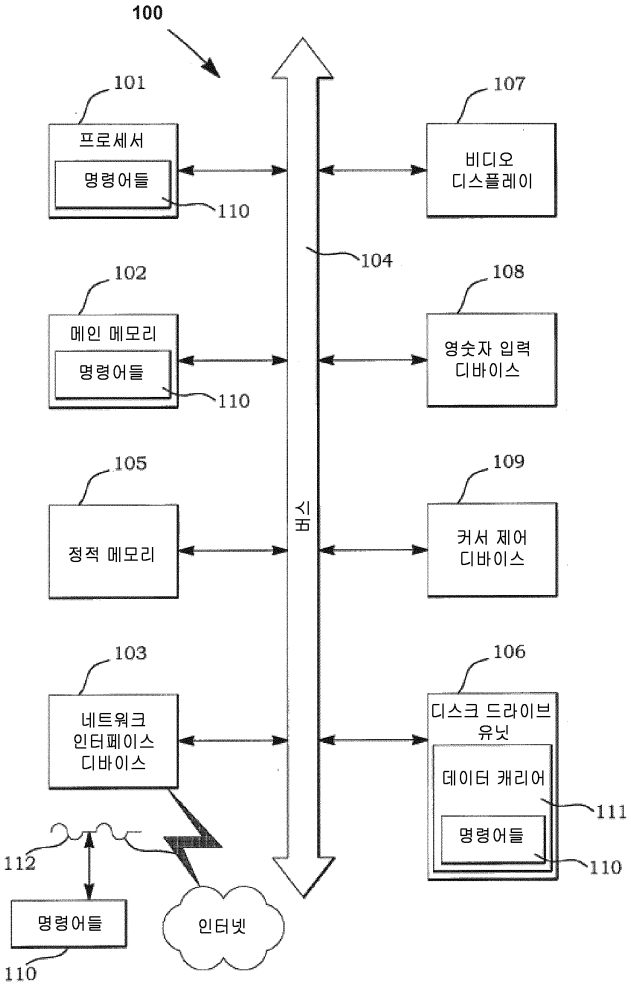
| ID | P | I | J | RD_STAT | D_ID | D_STAT | N | 타입 | 메시지 |
|---|---|---|---|---|---|---|---|---|---|
| 001 | AAAAAA | 0 | YY | S | NULL | NULL | NULL | NULL | NULL |
| 002 | BBBBBB | 0 | ZZ | S | NULL | NULL | NULL | NULL | NULL |
| 003 | AAAAAA | 0 | YY | NULL | 001 | S | TTY | T | $M_{YY}^{A(0),TTY}(1)'$ |
| 004 | AAAAAA | 1 | ZZ | S | 001 | NULL | NULL | NULL | NULL |
| 005 | AAAAAA | 1 | ZZ | NULL | 004 | S | DEL | X | $M_{ZZ}^{A(1),DEL}(1)'$ |
| 006 | BBBBBB | 0 | ZZ | NULL | 002 | V | TTY | T | $M_{ZZ}^{B(0),TTY}(1)'$ |
| 007 | AAAAAA | 1 | ZZ | NULL | 005 | S | TTY | T | $M_{ZZ}^{A(1),TTY}(2)'$ |
| 008 | BBBBBB | 0 | ZZ | NULL | 006 | S | ADD | X | $M_{ZZ}^{B(0),ADD}(2)'$ |
| 009 | BBBBBB | 1 | YY | S | 002 | NULL | NULL | NULL | NULL |
| 010 | BBBBBB | 1 | YY | NULL | 009 | S | TTY | Y | $M_{ZZ}^{B(1),TTY}(1)'$ |
| 011 | AAAAAA | 1 | YY | *S* | 001 | NULL | NULL | NULL | NULL |
| 012 | AAAAAA | 1 | YY | NULL | 011 | S | ADD | X | $M_{YY}^{A(1),XML}(1)'$ |
| 013 | BBBBBB | 2 | YY | N | 009 | NULL | NULL | NULL | NULL |
| 014 | BBBBBB | 2 | YY | NULL | 013 | ***S*** | DEL | X | $M_{YY}^{B(2),DEL}(1)'$ |
| *015* | *BBBBBB* | *2* | *YY* | *NULL* | *014* | ***N*** | *TTY* | *T* | $M_{YY}^{B(2),TTY}(2)'$ |
| 016 | AAAAAA | 2 | ZZ | ***S*** | 004 | NULL | NULL | NULL | NULL |
| 017 | AAAAAA | 2 | YY | N | 011 | NULL | NULL | NULL | NULL |
| 018 | AAAAAA | 2 | ZZ | NULL | 007 | ***S*** | TTY | T | $M_{ZZ}^{A(2),TTY}(1)'$ |
| 019 | AAAAAA | 2 | YY | NULL | 017 | ***S*** | DEL | X | $M_{YY}^{A(2),DEL}(1)'$ |
| 020 | BBBBBB | 2 | YY | NULL | 015 | D | ADD | X | $M_{YY}^{B(2),ADD}(3)'$ |
| 021 | AAAAAA | 2 | ZZ | NULL | 018 | V | SPLIT | X | $M_{ZZ}^{A(2),SPLIT}(2)'$ |
| *022* | *AAAAAA* | *2* | *YY* | *NULL* | *019* | ***N*** | *TTY* | *T* | $M_{YY}^{A(2),TTY}(2)'$ |
| 023 | AAAAAA | 2 | YY | NULL | 017 | *S* | NDC | N | $M_{YY}^{A(2),NDC}(3)'$ |
| 024 | AAAAAA | 10 | YY | U | ? | NULL | NULL | NULL | NULL |
| 025 | AAAAAA | 10 | YY | NULL | 024 | D | TTY | T | $M_{YY}^{A(10),TTY}(1)'$ |
| 26 | AAAAAA | 10 | YY | NULL | 025 | D | SPLIT | X | $M_{YY}^{A(10),SPLIT}(2)'$ |

도면13

테이블 V. 메시지 엔벨로프 2의 모든 메시지들을 전송한 후의 종속성 맵들.

| ID | P | I | J | RD_STAT | D_ID | D_STAT | N | 타입 | 메시지 |
|---|---|---|---|---|---|---|---|---|---|
| 001 | AAAAAA | 0 | YY | S | NULL | NULL | NULL | NULL | NULL |
| 002 | BBBBBB | 0 | ZZ | S | NULL | NULL | NULL | NULL | NULL |
| 003 | AAAAAA | 0 | YY | NULL | 001 | S | TTY | T | $M_{YY}^{A(0),TTY}(1)'$ |
| 004 | AAAAAA | 1 | ZZ | S | 001 | NULL | NULL | NULL | NULL |
| 005 | AAAAAA | 1 | ZZ | NULL | 004 | S | DEL | X | $M_{ZZ}^{A(1),DEL}(1)'$ |
| 006 | BBBBBB | 0 | ZZ | NULL | 002 | V | TTY | T | $M_{ZZ}^{B(0),TTY}(1)'$ |
| 007 | AAAAAA | 1 | ZZ | NULL | 005 | S | TTY | T | $M_{ZZ}^{A(1),TTY}(2)'$ |
| 008 | BBBBBB | 0 | ZZ | NULL | 006 | S | ADD | X | $M_{ZZ}^{B(0),ADD}(2)'$ |
| 009 | BBBBBB | 1 | YY | S | 002 | NULL | NULL | NULL | NULL |
| 010 | BBBBBB | 1 | YY | NULL | 009 | S | TTY | Y | $M_{ZZ}^{B(1),TTY}(1)'$ |
| 011 | AAAAAA | 1 | YY | S | 001 | NULL | NULL | NULL | NULL |
| 012 | AAAAAA | 1 | YY | NULL | 011 | S | ADD | X | $M_{YY}^{A(1),XML}(1)'$ |
| 013 | BBBBBB | 2 | YY | N (S) | 009 | NULL | NULL | NULL | NULL |
| 014 | BBBBBB | 2 | YY | NULL | 013 | S | DEL | X | $M_{YY}^{B(2),DEL}(1)'$ |
| 015 | BBBBBB | 2 | YY | NULL | 014 | S | TTY | T | $M_{YY}^{B(2),TTY}(2)'$ |
| 016 | AAAAAA | 2 | ZZ | S | 004 | NULL | NULL | NULL | NULL |
| 017 | AAAAAA | 2 | YY | S | 01 | NULL | NULL | NULL | NULL |
| 018 | AAAAAA | 2 | ZZ | NULL | 007 | S | TTY | T | $M_{ZZ}^{A(2),TTY}(1)'$ |
| 019 | AAAAAA | 2 | YY | NULL | 017 | S | DEL | X | $M_{YY}^{A(2),DEL}(1)'$ |
| *020* | *BBBBBB* | *2* | *YY* | *NULL* | *015* | *N (S)* | *ADD* | *X* | $M_{YY}^{B(2),ADD}(3)'$ |
| 021 | AAAAAA | 2 | ZZ | NULL | 018 | V | SPLIT | X | $M_{ZZ}^{A(2),SPLIT}(2)'$ |
| 022 | AAAAAA | 2 | YY | NULL | 019 | S | TTY | T | $M_{YY}^{A(2),TTY}(2)'$ |
| 023 | AAAAAA | 2 | YY | NULL | 017 | S | NDC | N | $M_{YY}^{A(2),NDC}(3)'$ |
| 024 | AAAAAA | 10 | YY | U | ? | NULL | NULL | NULL | NULL |
| 025 | AAAAAA | 10 | YY | NULL | 024 | D | TTY | T | $M_{YY}^{A(10),TTY}(1)'$ |
| 026 | AAAAAA | 10 | YY | NULL | 025 | D | SPLIT | X | $M_{YY}^{A(10),SPLIT}(2)'$ |

도면14




【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 6

【변경전】

제4항에 있어서,

상기 제2 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지를 생성하는 단계 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 -; 및

다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이 아니라는 결정에 응답하여 상기 제3 메시지에 대한 메시지 종속성을 연산하는 일 없이 상기 제3 메시지를 전송하는 단계

를 더 포함하는, 메시지 프로세싱 방법.

【변경후】

제4항에 있어서,

상기 제2 레거시 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지를 생성하는 단계 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 -; 및

다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이 아니라는 결정에 응답하여 상기 제3 메

시지에 대한 메시지 종속성을 연산하는 일 없이 상기 제3 메시지를 전송하는 단계

를 더 포함하는, 메시지 프로세싱 방법.

【직권보정 2】

【보정항목】 청구범위

【보정세부항목】 청구항 7

【변경전】

제4항에 있어서,

상기 제2 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지를 생성하는 단계 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 -; 및

적어도 하나의 다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이라는 결정에 응답하여, 적어도 상기 메시지 타깃 및 상기 제3 메시지의 기능에 기초하여 상기 제3 메시지의 메시지 종속성 - 상기 메시지 종속성은 상기 제3 메시지의 메시지 타깃에 대한 메시지들의 순서를 정의함 - 을 연산하고, 상기 메시지 엔벨로프 종속성 및 상기 메시지 종속성에 기초하여 순차적으로 상기 적어도 하나의 다른 메시지 및 상기 제3 메시지를 상기 제3 메시지의 메시지 타깃에 전송하는 단계

를 더 포함하는, 메시지 프로세싱 방법.

【변경후】

제4항에 있어서,

상기 제2 레거시 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지를 생성하는 단계 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 -; 및

적어도 하나의 다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이라는 결정에 응답하여, 적어도 상기 메시지 타깃 및 상기 제3 메시지의 기능에 기초하여 상기 제3 메시지의 메시지 종속성 - 상기 메시지 종속성은 상기 제3 메시지의 메시지 타깃에 대한 메시지들의 순서를 정의함 - 을 연산하고, 상기 메시지 엔벨로프 종속성 및 상기 메시지 종속성에 기초하여 순차적으로 상기 적어도 하나의 다른 메시지 및 상기 제3 메시지를 상기 제3 메시지의 메시지 타깃에 전송하는 단계

를 더 포함하는, 메시지 프로세싱 방법.

【직권보정 3】

【보정항목】 청구범위

【보정세부항목】 청구항 17

【변경전】

제15항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

상기 제2 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 - 를 생성하고;

다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이 아니라는 결정에 응답하여 상기 제3 메시지에 대한 메시지 종속성을 연산하는 일 없이 상기 제3 메시지를 전송하게 하는, 메시지 프로세싱 시스템.

【변경후】

제15항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

상기 제2 레거시 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 - 를 생성하고;

다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이 아니라는 결정에 응답하여 상기 제3 메시지에 대한 메시지 종속성을 연산하는 일 없이 상기 제3 메시지를 전송하게 하는, 메시지 프로세싱 시스템.

【직권보정 4】

【보정항목】 청구범위

【보정세부항목】 청구항 18

【변경전】

제15항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

상기 제2 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 - 를 생성하고;

적어도 하나의 다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이라는 결정에 응답하여, 적어도 상기 메시지 타깃 및 상기 제3 메시지의 기능에 기초하여 상기 제3 메시지의 메시지 종속성 - 상기 메시지 종속성은 상기 제3 메시지의 메시지 타깃에 대한 메시지들의 순서를 정의함 - 을 연산하고, 상기 메시지 엔벨로프 종속성 및 상기 메시지 종속성에 기초하여 순차적으로 상기 적어도 하나의 다른 메시지 및 상기 제3 메시지를 상기 제3 메시지의 메시지 타깃에 전송하게 하는, 메시지 프로세싱 시스템.

【변경후】

제15항에 있어서, 상기 프로그램 코드는 또한 상기 메시지 프로세싱 시스템으로 하여금,

상기 제2 레거시 표준에 따라 동작하는 레거시 데이터베이스 시스템으로부터 상기 복수의 데이터베이스 커밋 표시들 중 제2 데이터베이스 커밋 표시를 수신하는 것에 응답하여, 통지받을 각각의 메시지 타깃에 대한 제3 메시지 - 상기 제3 메시지는 상기 제2 레거시 표준에 따른 메시지임 - 를 생성하고;

적어도 하나의 다른 메시지가 상기 제3 메시지의 메시지 타깃에 전송되도록 계류 중이라는 결정에 응답하여, 적어도 상기 메시지 타깃 및 상기 제3 메시지의 기능에 기초하여 상기 제3 메시지의 메시지 종속성 - 상기 메시지 종속성은 상기 제3 메시지의 메시지 타깃에 대한 메시지들의 순서를 정의함 - 을 연산하고, 상기 메시지 엔벨로프 종속성 및 상기 메시지 종속성에 기초하여 순차적으로 상기 적어도 하나의 다른 메시지 및 상기 제3 메시지를 상기 제3 메시지의 메시지 타깃에 전송하게 하는, 메시지 프로세싱 시스템.