



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2011년12월26일
(11) 등록번호 10-1098356
(24) 등록일자 2011년12월19일

(51) Int. Cl.

G06Q 40/00 (2006.01)

(21) 출원번호 10-2004-7012668

(22) 출원일자(국제출원일자) 2003년02월14일

심사청구일자 2008년02월14일

(85) 번역문제출일자 2004년08월14일

(65) 공개번호 10-2004-0097128

(43) 공개일자 2004년11월17일

(86) 국제출원번호 PCT/US2003/004424

(87) 국제공개번호 WO 2003/069444

국제공개일자 2003년08월21일

(30) 우선권주장

60/356,148 2002년02월14일 미국(US)

60/439,797 2003년01월14일 미국(US)

(56) 선행기술조사문헌

KR1020000036545 A*

US04914587 A1*

US5475585 A

US4774664 A

*는 심사관에 의하여 인용된 문헌

(73) 특허권자

폐신 자차리

미국 뉴욕 10023 뉴욕 웨스트 60th 스트리트 200
아파트먼트 23췌

(72) 발명자

폐신 자차리

미국 뉴욕 10023 뉴욕 웨스트 60th 스트리트 200
아파트먼트 23췌

(74) 대리인

리앤목특허법인

전체 청구항 수 : 총 20 항

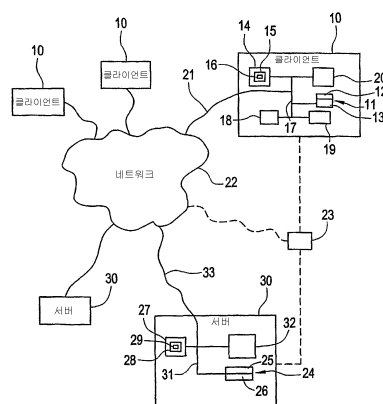
심사관 : 김정진

(54) 분산형 자본 시스템의 장치 및 방법

(57) 요약

본 발명에 따른 방법들, 시스템들 및 제조 물품들은 네트워크(22)를 통해 금융 거래들을 수행하기 위해 제공된다. 사용자(10)는 당사자들 각각이 상기 네트워크(22) 상의 데이터 처리 시스템에 대응하는 다수의 당사자들 중 적어도 하나의 당사자와 적어도 한 번의 금융 거래를 수행할 것을 요구한다. 상기 금융 거래에 관한 실시간 금융 정보는 획득되고, 상기 사용자는 적어도 하나의 당사자와의 금융 거래가 금융 거래의 수행 이전에 이루어질 수 있다는 것을 테스트 및 확인할 수 있다.

대표도 - 도3



특허청구의 범위

청구항 1

삭제

청구항 2

삭제

청구항 3

삭제

청구항 4

삭제

청구항 5

삭제

청구항 6

삭제

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

청구항 12

삭제

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

삭제

청구항 22

삭제

청구항 23

삭제

청구항 24

삭제

청구항 25

삭제

청구항 26

삭제

청구항 27

삭제

청구항 28

삭제

청구항 29

삭제

청구항 30

삭제

청구항 31

삭제

청구항 32

삭제

청구항 33

삭제

청구항 34

삭제

청구항 35

삭제

청구항 36

삭제

청구항 37

삭제

청구항 38

삭제

청구항 39

삭제

청구항 40

삭제

청구항 41

삭제

청구항 42

삭제

청구항 43

삭제

청구항 44

삭제

청구항 45

삭제

청구항 46

삭제

청구항 47

삭제

청구항 48

삭제

청구항 49

삭제

청구항 50

삭제

청구항 51

삭제

청구항 52

삭제

청구항 53

삭제

청구항 54

삭제

청구항 55

삭제

청구항 56

삭제

청구항 57

삭제

청구항 58

삭제

청구항 59

삭제

청구항 60

삭제

청구항 61

삭제

청구항 62

삭제

청구항 63

삭제

청구항 64

삭제

청구항 65

삭제

청구항 66

삭제

청구항 67

삭제

청구항 68

삭제

청구항 69

삭제

청구항 70

삭제

청구항 71

삭제

청구항 72

삭제

청구항 73

삭제

청구항 74

삭제

청구항 75

삭제

청구항 76

삭제

청구항 77

삭제

청구항 78

삭제

청구항 79

삭제

청구항 80

삭제

청구항 81

삭제

청구항 82

금융 서비스 시스템에 있어서,

제1 금융 거래에 대한 요구를 수신하는 수단;

상기 제1 금융 거래에서 적어도 하나의 잠재적인 참가자에 대한 정보를 수신하는 수단;

상기 적어도 하나의 잠재적인 참가자에게 상기 제1 금융 거래를 통지하는 수단; 및

제2 금융 거래에 대한 제2 요구를 수신하는 수단;

을 포함하며,

상기 제2 금융 거래의 결과는 상기 제1 금융 거래를 개시하도록 지시되는 것을 특징으로 하는, 금융 서비스 시스템.

청구항 83

삭제

청구항 84

금융 서비스 시스템에 있어서,

사용자 인터페이스를 작동시키고 온-스크린 금융 거래의 구성, 테스트, 및 실행을 가능하게 하는 모듈;

을 포함하며,

상기 온-스크린 금융 거래의 구성은,

적어도 하나의 금융 거래에 수반되는 복수의 실체(entity)들에 관련된 복수의 아이콘들을 온-스크린 거래 작업 패드 상에 집합시키는 것;과,

상기 금융 거래가 이루어짐을 나타내는 자본 자산 벡터(capital asset vector)를 사용하여 상기 아이콘들을 연결시키는 것;

을 포함하며,

상기 온-스크린 금융 거래의 테스트는,

상기 금융 거래의 실행가능성을 결정하고 모든 거래량 및 상기 실체들 모두의 계정들에서의 결과적인 변경들의 시뮬레이팅된 계산(accounting)을 처리하도록 상기 실체들의 계정들에 관한 실-시간 정보를 획득함으로써, 상기 온-스크린 금융 거래에서 집합되는 상기 실체들 모두에 관한 체크(check)를 포함하고,

상기 온-스크린 금융 거래의 실행은,

진행 막대(progress bar)를 사용하여 온-스크린에서 보이는 실행 아이템들의 리스트를 수행하는 것;

을 포함함을 특징으로 하는, 금융 서비스 시스템.

청구항 85

삭제

청구항 86

삭제

청구항 87

삭제

청구항 88

삭제

청구항 89

삭제

청구항 90

삭제

청구항 91

삭제

청구항 92

네트워크를 통해 금융 서비스를 수행하도록 구성된 컴퓨터 시스템에 있어서,

상기 컴퓨터 시스템은, 적어도 하나의 메모리;를 포함하며,

상기 적어도 하나의 메모리에는,

상기 컴퓨터 시스템으로 하여금,

금융 거래에 대한 요구를 수신하는 단계;

인터넷 또는 보안 금융 네트워크를 통해, 상기 금융 거래에 수반되는 적어도 하나의 실체(entity)에 질의(query)를 송신하는 단계;

인터넷 또는 보안 금융 네트워크를 통해, 상기 금융 거래에 수반되는 적어도 하나의 실체로부터 실-시간 정보를 획득하는 단계;

상기 금융 거래가 실행 및 수행될 수 있다는 것을 상기 적어도 하나의 실체로 확인하는 단계; 및

상기 금융 거래가 실행될 수 있을 때 상기 금융 거래를 검증 및 실행하는 단계;

를 수행할 수 있게 하도록 구성된 명령어들을 지니는 프로그램이 저장되어 있으며,

상기 실행하는 단계는, 상기 금융 거래 중 적어도 일부분을 정합시키기 위해 금융 거래에서 직접적으로 관련되어 있는 당사자들 사이에서만 이기보다는, 동일한 네팅 합계에서 전혀 관련되어 있지 않은 당사자들을 함께 포함한 금융 거래를 수행하는 모든 당사자들 간의 제로섬 원리(zero-sum principles)를 모든 크레딧(credits) 및 데빗(debits)의 수취계정(receivables) 및 채무(liabilities)가 충실히 따르게 하는 스프레드-벡터 네팅(spread-vector netting) 프로세스를 이용하고,

상기 컴퓨터 시스템은, 상기 프로그램을 실행하는 적어도 하나의 프로세서;를 포함하는 것을 특징으로 하는, 컴퓨터 시스템.

청구항 93

삭제

청구항 94

삭제

청구항 95

컴퓨터 시스템으로 하여금 네트워크를 통해 금융 거래를 수행하게 하는 콘텐츠가 수록된 컴퓨터 판독가능 매체에 있어서,

상기 컴퓨터 시스템은,

인터넷 또는 보안 금융 서버를 통해, 대출해 주는 적어도 하나의 당사자로부터 금융 시스템으로 자금을 제공하는 단계;

상기 대출해 주는 적어도 하나의 당사자에 대한 계정에서 상기 자금을 출금하는 단계;

상기 인터넷 또는 상기 보안 금융 서버를 통해, 상기 대출해 주는 적어도 하나의 당사자로부터의 상기 자금을 상기 자금을 구하는 적어도 하나의 차용인과 직접 정합시키는 단계; 및

상기 대출해 주는 적어도 하나의 당사자가 차용된 자금액을 적어도 부분적으로 상기 적어도 하나의 차용인에게 제공하도록 차용된 자금액을 상기 적어도 하나의 차용인의 각각의 계정에 입금하는 단계;

를 수행하는 프로그램을 지니며,

상기 적어도 하나의 차용인은, 상기 대출해 주는 적어도 하나의 당사자에게 차용된 자금액에 대한 개별적인 상환 스케줄을 주는 것을 특징으로 하는, 컴퓨터 판독가능 매체.

청구항 96

삭제

청구항 97

네트워크를 통해 금융 서비스를 수행하도록 구성된 컴퓨터 시스템에 있어서,

상기 컴퓨터 시스템은, 적어도 하나의 메모리;를 포함하며,

상기 적어도 하나의 메모리에는,

상기 컴퓨터 시스템으로 하여금,

인터넷 또는 보안 금융 서버를 통해, 대출해 주는 적어도 하나의 당사자로부터 금융 시스템을 이용하는 적어도 다른 당사자에게 자금을 제공하는 단계;

상기 대출해 주는 적어도 하나의 당사자에 대한 계정에서 상기 자금을 출금하는 단계;

상기 인터넷 또는 상기 보안 금융 서버를 통해, 상기 대출해 주는 적어도 하나의 당사자로부터의 상기 자금을 상기 자금을 구하는 적어도 하나의 차용인과 직접 정합시키는 단계; 및

상기 대출해 주는 적어도 하나의 당사자가 차용된 자금액을 적어도 부분적으로 상기 적어도 하나의 차용인에게 제공하도록 상기 대출해 주는 적어도 하나의 당사자로부터 차용된 자금액을 상기 적어도 하나의 차용인의 각각의 계정에 입금하는 단계;

를 수행하게 하도록 구성된 명령어들을 지니는 프로그램이 저장되어 있으며,

상기 적어도 하나의 차용인은, 상기 대출해 주는 적어도 하나의 당사자에게 차용된 자금액에 대한 개별적인 상환 스케줄을 주고,

상기 컴퓨터 시스템은, 상기 프로그램을 실행하는 적어도 하나의 프로세서;를 포함하는 것을 특징으로 하는, 컴퓨터 시스템.

청구항 98

삭제

청구항 99

삭제

청구항 100

삭제

청구항 101

컴퓨터 네트워크를 통해 금융 거래를 수행하는 방법에 있어서,

당사자들 간의 대체가능한 금융 거래를 실시하는 단계;

적어도 하나의 당사자로부터 거래될 제1 항목을 적어도 하나의 대응 당사자로부터 거래될 적어도 제2 항목과 정합시킴으로써 상기 대체가능한 금융 거래를 구현하는 단계; 및

상기 당사자들 및 대응 당사자들 간의 네팅(netting) 합계를 수행하는 단계;

를 포함하는 것을 특징으로 하는, 방법.

청구항 102

컴퓨터 네트워크를 통해 금융 거래를 수행하는 방법에 있어서,

참가자 시스템 네트워크를 통한 자금 이동이 최적화되게끔 인식 전파(awareness propagation)를 수행하는 단계;

제1 사용자의 계정에 대한 적어도 하나의 네트 자금 공제를 수행하는 단계;

적어도 하나의 제2 사용자의 계정에 대한 적어도 하나의 네트 자금 부가를 수행하는 단계; 및

현재 네트 자금 불균형의 절대값을 충족하는 것과 현재 네트 자금 불균형의 절대값을 초과하는 것 중 어느 하나인 금액에 대하여 연속적인 사용자들의 계정에 대한 적어도 하나의 네트 자금 공제 및 상기 적어도 하나의 네트 자금 부가가 실행되게끔 상기 인식 전파를 반복 수행하는 단계;

를 포함하는 것을 특징으로 하는, 방법.

청구항 103

제102항에 있어서,

사용자 계정들로부터의 적어도 하나의 네트 자금 공제 및 적어도 하나의 네트 자금 부가로 반복 네트 자금 불균형이 최적화되게끔 상기 인식 전파를 반복 수행하는 단계;

를 더 포함하는 것을 특징으로 하는, 방법.

청구항 104

컴퓨터 네트워크를 통해 금융 거래를 수행하는 방법에 있어서,

참가자 시스템 네트워크를 통한 자금 이동이 최적화되게끔 인식 전파를 수행하는 단계;

제1 금융 기관의 사용자로부터 제1 자금 이동을 수행하도록 하는 명령을 수신하는 단계;

상기 제1 금융 기관으로의 제2 자금 이동이 제2 금융 기관에서 스케줄되는 것을 인식 전파로 결정하는 단계;

상기 제1 자금 이동 및 상기 제2 자금 이동에 대한 정보 및 그에 대한 대체 매개변수들을 거래하는 단계;

상기 제1 금융 기관 및 상기 제2 금융 기관에서 의도된 네트 매개변수 배치들을 결정하는 단계;

상기 제1 자금 이동에 대한 최적의 대체 매개변수들을 반복 계산하는 단계; 및

상기 제1 자금 이동 및 상기 제2 자금 이동에 대한 분산 대체를 수행하는 단계;

를 포함하는 것을 특징으로 하는, 방법.

청구항 105

금융 서비스 시스템에 있어서,

인터넷 또는 보안 금융 서버를 통해, 대출해 주는 적어도 하나의 당사자로부터 금융 시스템을 이용하는 적어도 다른 사용자에게 자금을 제공하는 수단;

상기 대출해 주는 적어도 하나의 당사자의 계정에서 상기 자금을 출금하는 수단;

상기 인터넷 또는 상기 보안 금융 서버를 통해, 상기 대출해 주는 적어도 하나의 당사자로부터의 상기 자금을 상기 자금을 구하는 적어도 하나의 차용인과 직접 정합시키는 수단; 및

상기 대출해 주는 적어도 하나의 당사자가 차용된 자금액을 적어도 부분적으로 상기 적어도 하나의 차용인에게 제공하도록 상기 대출해 주는 적어도 하나의 당사자로부터 차용된 자금액을 상기 적어도 하나의 차용인의 각각의 계정에 입금하는 수단;

을 포함하며,

상기 적어도 하나의 차용인은, 상기 대출해 주는 적어도 하나의 당사자에게 차용된 자금액에 대한 개별적인 상환 스케줄을 주는 것을 특징으로 하는, 금융 서비스 시스템.

청구항 106

삭제

청구항 107

삭제

청구항 108

삭제

청구항 109

삭제

청구항 110

삭제

청구항 111

삭제

청구항 112

삭제

청구항 113

삭제

청구항 114

삭제

청구항 115

컴퓨터 시스템으로 하여금 네트워크를 통해 금융 거래를 수행하게 하는 콘텐츠가 수록된 컴퓨터 판독가능 매체에 있어서,

상기 컴퓨터 시스템은,

금융 거래에 대한 요구를 수신하는 단계;

인터넷 또는 보안 금융 네트워크를 통해, 상기 금융 거래에 수반되는 적어도 하나의 실체(entity)에 질의(query)를 송신하는 단계;

인터넷 또는 보안 금융 네트워크를 통해, 상기 금융 거래에 수반되는 적어도 하나의 실체로부터 실-시간 정보를 획득하는 단계;

상기 금융 거래가 실행 및 수행될 수 있다는 것을 상기 적어도 하나의 실체로 확인하는 단계; 및

상기 금융 거래가 실행될 수 있을 때 상기 금융 거래를 검증 및 실행하는 단계;

를 수행하는 프로그램을 지니며,

상기 실행하는 단계는, 상기 금융 거래 중 적어도 일부분을 정합시키기 위해 금융 거래에서 직접적으로 관련되어 있는 당사자들 사이에서만 이기보다는, 동일한 네팅 합계에서 전혀 관련되어 있지 않은 당사자들을 함께 포함한 금융 거래를 수행하는 모든 당사자들 간의 제로섬 원리(zero-sum principles)를 모든 크레딧(credits) 및 데빗(debits)의 수취계정(receivables) 및 채무(liabilities)가 충실히 따르게 하는 스프레드-벡터 네팅(spread-vector netting) 프로세스를 이용하는, 컴퓨터 판독가능 매체.

청구항 116

삭제

청구항 117

컴퓨터 시스템으로 하여금 네트워크를 통해 금융 거래를 수행하게 하는 콘텐츠가 수록된 컴퓨터 판독가능 매체에 있어서,

상기 컴퓨터 시스템은,

당사자들 간의 대체가능한 금융 거래를 실시하는 단계;

적어도 하나의 당사자로부터 거래될 제1 항목을 적어도 하나의 대응 당사자로부터 거래될 적어도 제2 항목과 정합시킴으로써 상기 대체가능한 금융 거래를 구현하는 단계; 및

상기 당사자들 및 대응 당사자들 간의 네팅(netting) 합계를 수행하는 단계;

를 수행하는 프로그램을 지니는, 컴퓨터 판독가능 매체.

청구항 118

컴퓨터 시스템으로 하여금 네트워크를 통해 금융 거래를 수행하게 하는 콘텐츠가 수록된 컴퓨터 판독가능 매체에 있어서,

상기 컴퓨터 시스템은,

참가자 시스템 네트워크를 통한 자금 이동이 최적화되게끔 인식 전파(awareness propagation)를 수행하는 단계;

제1 사용자의 계정에 대한 적어도 하나의 네트 자금 공제를 수행하는 단계;

적어도 하나의 제2 사용자의 계정에 대한 적어도 하나의 네트 자금 부가를 수행하는 단계; 및

현재 네트 자금 불균형의 절대값을 충족하는 것과 현재 네트 자금 불균형의 절대값을 초과하는 것 중 어느 하나인 금액에 대하여 연속적인 사용자들의 계정에 대한 적어도 하나의 네트 자금 공제 및 상기 적어도 하나의 네트 자금 부가가 실행되게끔 상기 인식 전파를 반복 수행하는 단계;

를 수행하는 프로그램을 지니는, 컴퓨터 판독가능 매체.

청구항 119

컴퓨터 시스템으로 하여금 네트워크를 통해 금융 거래를 수행하게 하는 콘텐츠가 수록된 컴퓨터 판독가능 매체에 있어서,

상기 컴퓨터 시스템은,

참가자 시스템 네트워크를 통한 자금 이동이 최적화되게끔 인식 전파를 수행하는 단계;

제1 금융 기관의 사용자로부터 제1 자금 이동을 수행하도록 하는 명령을 수신하는 단계;

상기 제1 금융 기관으로의 제2 자금 이동이 제2 금융 기관에서 스케줄되는 것을 인식 전파로 결정하는 단계;

상기 제1 자금 이동 및 상기 제2 자금 이동에 대한 정보 및 그에 대한 대체 매개변수들을 거래하는 단계;

상기 제1 금융 기관 및 상기 제2 금융 기관에서 의도된 네트 매개변수 배치들을 결정하는 단계;

상기 제1 자금 이동에 대한 최적의 대체 매개변수들을 반복 계산하는 단계; 및
상기 제1 자금 이동 및 상기 제2 자금 이동에 대한 분산 대체를 수행하는 단계;
를 수행하는 프로그램을 지니는, 컴퓨터 판독가능 매체.

청구항 120

삭제

청구항 121

네트워크를 통해 금융 거래를 수행하도록 구성된 컴퓨터 시스템에 있어서,
상기 컴퓨터 시스템은, 적어도 하나의 메모리;를 포함하며,
상기 적어도 하나의 메모리에는,
상기 컴퓨터 시스템으로 하여금,
당사자들 간의 대체가능한 금융 거래를 실시하는 단계;
적어도 하나의 당사자로부터 거래될 제1 항목을 적어도 하나의 대응 당사자로부터 거래될 적어도 제2 항목과 정
합시킴으로써 상기 대체가능한 금융 거래를 구현하는 단계; 및
상기 당사자들 및 대응 당사자들 간의 네팅(netting) 합계를 수행하는 단계;
를 수행하게 하도록 구성된 명령어들을 지니는 프로그램이 저장되어 있으며,
상기 컴퓨터 시스템은, 상기 프로그램을 실행하는 적어도 하나의 프로세서;를 포함하는 것을 특징으로 하는, 컴
퓨터 시스템.

청구항 122

네트워크를 통해 금융 거래를 수행하도록 구성된 컴퓨터 시스템에 있어서,
상기 컴퓨터 시스템은, 적어도 하나의 메모리;를 포함하며,
상기 적어도 하나의 메모리에는,
상기 컴퓨터 시스템으로 하여금,
참가자 시스템 네트워크를 통한 자금 이동이 최적화되게끔 인식 전파(awareness propagation)를 수행하는 단계;
제1 사용자의 계정에 대한 적어도 하나의 네트 자금 공제를 수행하는 단계;
적어도 하나의 제2 사용자의 계정에 대한 적어도 하나의 네트 자금 부가를 수행하는 단계; 및
현재 네트 자금 불균형의 절대값을 충족하는 것과 현재 네트 자금 불균형의 절대값을 초과하는 것 중 어느 하나
인 금액에 대하여 연속적인 사용자들의 계정에 대한 적어도 하나의 네트 자금 공제 및 상기 적어도 하나의 네트
자금 부가가 실행되게끔 상기 인식 전파를 반복 수행하는 단계;
를 수행하게 하도록 구성된 명령어들을 지니는 프로그램이 저장되어 있으며,
상기 컴퓨터 시스템은, 상기 프로그램을 실행하는 적어도 하나의 프로세서;를 포함하는 것을 특징으로 하는, 컴
퓨터 시스템.

청구항 123

네트워크를 통해 금융 거래를 수행하도록 구성된 컴퓨터 시스템에 있어서,
상기 컴퓨터 시스템은, 적어도 하나의 메모리;를 포함하며,
상기 적어도 하나의 메모리에는,
상기 컴퓨터 시스템으로 하여금,

참가자 시스템 네트워크를 통한 자금 이동이 최적화되게끔 인식 전파를 수행하는 단계;

제1 금융 기관의 사용자로부터 제1 자금 이동을 수행하도록 하는 명령을 수신하는 단계;

상기 제1 금융 기관으로의 제2 자금 이동이 제2 금융 기관에서 스케줄되는 것을 인식 전파로 결정하는 단계;

상기 제1 자금 이동 및 상기 제2 자금 이동에 대한 정보 및 그에 대한 대체 매개변수들을 거래하는 단계;

상기 제1 금융 기관 및 상기 제2 금융 기관에서 의도된 네트 매개변수 배치들을 결정하는 단계;

상기 제1 자금 이동에 대한 최적의 대체 매개변수들을 반복 계산하는 단계; 및

상기 제1 자금 이동 및 상기 제2 자금 이동에 대한 분산 대체를 수행하는 단계;

를 수행하게 하도록 구성된 명령어들을 지니는 프로그램이 저장되어 있으며,

상기 컴퓨터 시스템은, 상기 프로그램을 실행하는 적어도 하나의 프로세서;를 포함하는 것을 특징으로 하는, 컴퓨터 시스템.

청구항 124

제95항에 있어서, 추가 당사자들과의 연속적인 정합은, 상기 금융 거래가 완료될 때까지 수행되는 것을 특징으로 하는, 컴퓨터 판독가능 매체.

청구항 125

제97항에 있어서, 추가 당사자들과의 연속적인 정합은, 상기 금융 거래가 완료될 때까지 수행되는 것을 특징으로 하는, 컴퓨터 시스템.

청구항 126

제105항에 있어서, 추가 당사자들과의 연속적인 정합은, 상기 금융 거래가 완료될 때까지 수행되는 것을 특징으로 하는, 금융 서비스 시스템.

명세서

기술분야

[0001] 본 발명은 일반적으로 글로벌 컴퓨터 네트워크를 통해 복수의 금융 거래를 수행할 수 있는 자동화된 분산형 금융 또는 자본 시스템 방법 및 장치에 관한 것이다.

배경기술

[0002] 금융 서비스에 있어서, 특히 지리적인 차이(시간차)가 통화, 법률 및 정치적인 영토의 차이에 의해 악화되는 경우, 지리적인 위치는 만만치 않은 장벽이다. 세계 경제의 전망 및 완전한 시스템화, 직접적인 처리 및 합리적인 운용의 요구에도 불구하고, 통합된 글로벌 금융 서비스를 처리하도록 명확하게 설계된 구조를 구성한 조직이 아직 없다.

발명의 상세한 설명

[0003] 본 발명은 글로벌 컴퓨터 네트워크를 통해 복수의 금융 거래를 수행할 수 있는 자동화된 분산형 자본 시스템 방법 및 장치에 관한 것이다.

[0004] 특히, 본 발명의 분산형 자본 시스템(DCS; distributed capital system)은 크게 2개의 카테고리에 해당하는 거래를 처리할 수 있다. 전통적(Traditional)(개인), 및 협조적(Collaborative)(다수 당사자). 상기 2개의 카테고리는 거래 범위를 포함하고, 본 명세서에 기술되는 본 발명의 DCS는 고유한 거래 이벤트(계산서 지불, 외화 환전 등과 같은)의 조합, 테스트, 수행 및 관리를 포함한다.

[0005] 특히, 본 발명의 DCS는 다른 거래들 중에서 다음을 처리한다. 계산서 지불 구성(structuring), 관리(management), 및 수행(execution); 송장 작성(invoicing) 구성, 관리 및 수행; 자금 대체 개시 및 관리; 국제적으로 대체하지 않고 자금을 본국으로 보내는 것을 포함하는 분산 자금 대체(DFT; distributed fund transfer) 개시 및 관리; (중개인이거나 제3자 금융 기관 없이) 전 세계의 당사자들간의 특히 통화가 대칭적인(즉 거래의 각

당사자가 상대측 당사자와 전부 교환되는 균등한 가치를 보유) 또는 비대칭적인(거래의 각 당사자가 상이한 가치의 양을 보유), 대체물의 직접 교환; 자동화된 외환 시장; 프로그래밍된 신용카드 지불 스케줄링; 전 세계의 ATM 요금 지불 방지; 전통적인 통화-교환 거래를 요구하지 않는 상이한 통화의 온라인 구입 거래; 소비자 신용 및 직접 대여 시스템; 분산 신용-평가 시스템; 무제한 참가자로부터의 소량 내지 대량을 처리하는 신디케이션(syndication) 시스템; 및 개별적인 또는 타인과 협력하는 프로그래밍된 투자.

[0006] 특히, 본 발명의 태양은 인터넷과 같은 컴퓨터 네트워크를 통해 복수의 금융 거래를 수행할 수 있는 자동화된 분산형 자본 시스템에 관한 것이다. 본 발명의 분산형 자본 시스템(DCS)은 어떤 수의 당사자들간의 어떤 유형의 금융 거래를 관리한다. 따라서, 예를 들어, 한 당사자가 어떤 통화 또는 환전 가치에 대한 금융 거래를 수행하기를 원하고 제1 대응 당사자가 완전한 통화 또는 환전 가치에 대한 거래를 수행할 수 없는 경우, 본 발명의 DCS는 거래가 완전히 수행될 수 있을 때까지 추가 대응 당사자들을 자동으로 획득한다. 따라서, 네트워크상의 다수의 당사자들 중에 분산된 충분한 가치가 있는 경우, 본 발명의 DCS는 어떤 가치에 대해 금융 거래를 수행할 수 있다.

[0007] 금융 거래를 개시하는 당사자는 대응 당사자들을 선택할 수 있거나 본 발명의 DCS는 상기 대응 당사자들을 자동으로 획득할 수 있다. 또한, 대응 당사자들은 네트워크의 어느 곳에도 위치할 수 있다. 따라서, 네트워크가 예를 들어 인터넷인 경우, 본 발명을 구성하는 방법 및 시스템은 지리적인 경계에 제한되지 않는다. 본 발명의 DCS는 다수의 국가들에 있는 당사자들간의 거래를 수행할 수 있고 각 거래 통화를 자동으로 교환할 수 있다.

[0008] 따라서, 분산형 자본 시스템은 어떤 수의 당사자가 어떤 수의 대응 당사자와 어떤 양의 가치를 협조하여 거래하도록 허용하고, 독립적인 시장 중심(market-driven) 가격 및 조건으로 구성되고 가치 거래를 처리한다. 이것은 예를 들어 브로커(broker)와 같은 중개인 없이 수행된다.

[0009] 어떤 수의 당사자 및 대응 당사자가 있을 수 있기 때문에, DCS는 예를 들어 단일 당사자 및 단일 대응 당사자(전통적이고 간단한(traditional simple)), 동시에 몇몇 전통적인 간단한 거래를 수행하는 단일 당사자(전통적이고 복잡한(traditional compound)), 단일 당사자 및 다수의 대응 당사자(협조적이고 간단한(collaborative simple)), 및 다수의 당사자 및 다수의 대응 당사자(협조적이고 복잡한(collaborative compound))를 포함하는 금융 거래를 수행할 수 있다.

[0010] 본 발명의 방법에 따라, 프로그램을 지니는 데이터 처리 시스템에서 이루어지는 방법이 제공된다. 상기 데이터 처리 시스템은 네트워크를 통해 복수의 원격용 데이터 처리 시스템들 중 적어도 하나에 접속된다. 상기 방법은 상기 프로그램에 의해 수행되는 단계들을 포함하며, 상기 단계들은, 복수의 당사자들 각각이 상기 데이터 처리 시스템들 중 하나에 해당하는 복수의 당사자들 중 적어도 하나와 적어도 한번 금융 거래를 수행하기 위한 요구를 사용자로부터 받는 단계; 상기 금융 거래에 관한 실시간 금융 정보를 획득하는 단계; 및 상기 금융 거래가 적어도 하나의 당사자와 수행될 수 있다는 것을 확인하는 단계를 포함한다.

[0011] 본 발명의 방법에 따라, 프로그램을 지니는 데이터 처리 시스템에서 이루어지는 방법이 제공된다. 상기 데이터 처리 시스템은 네트워크를 통해 복수의 원격용 데이터 처리 시스템들 중 적어도 하나에 접속된다. 상기 방법은 상기 프로그램에 의해 수행되는 단계들을 포함하며, 상기 단계들은, 복수의 당사자들 각각이 상기 데이터 처리 시스템들 중 하나에 해당하는 복수의 당사자들 중 적어도 하나와 적어도 한번 금융 거래를 수행하기 위한 요구를 사용자로부터 받는 단계; 상기 금융 거래에 관한 실시간 금융 정보를 획득하는 단계; 상기 금융 거래의 통화 또는 환전 가치를 결정하는 단계; 상기 적어도 하나의 당사자가 상기 통화 또는 환전 가치에 대해 상기 금융 거래를 수행할지의 여부를 결정하는 단계; 및 확인된 적어도 하나의 당사자가 상기 통화 또는 환전 가치에 대해 상기 금융 거래를 수행하지 않는 것에 응답하여 상기 당사자들이 상기 금융 거래를 수행할 총 통화 또는 환전 가치가 상기 금융 거래의 결정된 통화 또는 환전 가치와 동일할 때까지 상기 금융 거래가 상기 적어도 하나의 당사자에 대한 추가의 당사자들과 수행될 수 있는지를 확인하는 단계를 포함한다.

[0012] 본 발명에 따른 제조물에 따라, 데이터 처리 시스템이 한가지 방법을 수행하게 하는 명령어들을 포함하는 컴퓨터 판독가능 매체가 제공된다. 상기 데이터 처리 시스템은 네트워크를 통해 복수의 원격용 데이터 처리 시스템들 중 적어도 하나에 접속된다. 상기 방법은 프로그램에 의해 수행되는 단계들을 포함하고, 상기 단계들은, 복수의 당사자들 각각이 상기 데이터 처리 시스템들 중 하나에 해당하는 복수의 당사자들 중 적어도 하나와 적어도 한번 금융 거래를 수행하기 위한 요구를 사용자로부터 받는 단계; 상기 금융 거래에 관한 실시간 금융 정보를 획득하는 단계; 및 상기 금융 거래가 적어도 하나의 당사자와 수행될 수 있다는 것을 확인하는 단계를 포함한다.

- [0013] 본 발명에 따른 제조물에 따라, 데이터 처리 시스템이 한가지 방법을 수행하게 하는 명령어들을 포함하는 컴퓨터 판독가능 매체가 제공된다. 상기 데이터 처리 시스템은 네트워크를 통해 복수의 원격용 데이터 처리 시스템들 중 적어도 하나에 접속된다. 상기 방법은 프로그램에 의해 수행되는 단계들을 포함하며, 상기 단계들은, 복수의 당사자들 각각이 상기 데이터 처리 시스템들 중 하나에 해당하는 복수의 당사자들 중 적어도 하나와 적어도 한번 금융 거래를 수행하기 위한 요구를 사용자로부터 받는 단계; 상기 금융 거래에 관한 실시간 금융 거래를 획득하는 단계; 상기 금융 거래의 통화 또는 환전 가치를 결정하는 단계; 상기 적어도 하나의 당사자가 상기 통화 또는 환전 가치에 대해 상기 금융 거래를 수행할 지의 여부를 결정하는 단계; 및 상기 확인된 적어도 하나의 당사자가 상기 통화 또는 환전 가치에 대해 상기 금융 거래를 수행하지 않는 것에 응답하여 상기 당사자들이 상기 금융 거래를 수행할 총 통화 또는 환전 가치가 상기 금융 거래의 결정된 통화 또는 환전 가치와 동일할 때까지 상기 금융 거래가 상기 적어도 하나의 당사자에 대한 추가의 당사자들과 수행될 수 있다는 것을 확인하는 단계를 포함한다.
- [0014] 본 발명에 따른 시스템에 따라, 네트워크를 통해 복수의 원격용 데이터 처리 시스템들 중 적어도 하나에 접속된 데이터 처리 시스템이 제공된다. 상기 데이터 처리 시스템은 복수의 당사자들 각각이 상기 데이터 처리 시스템들 중 하나에 해당하는 복수의 당사자들 중 적어도 하나와 적어도 한번 금융 거래를 수행하기 위한 요구를 사용자로부터 받는 수단; 상기 금융 거래에 관한 실시간 금융 정보를 획득하는 수단; 및 상기 금융 거래가 적어도 하나의 당사자와 수행될 수 있다는 것을 확인하는 수단을 포함한다.
- [0015] 본 발명에 따른 시스템에 따라, 네트워크를 통해 복수의 원격용 데이터 처리 시스템 중 적어도 하나에 접속된 데이터 처리 시스템이 제공된다. 상기 데이터 처리 시스템은 복수의 당사자들 각각이 상기 데이터 처리 시스템들 중 하나에 해당하는 복수의 당사자들 중 적어도 하나와 적어도 한번 금융 거래를 수행하기 위한 요구를 사용자로부터 받는 수단; 상기 금융 거래에 관한 실시간 금융 정보를 획득하는 수단; 상기 금융 거래의 통화 또는 환전 가치를 결정하는 수단; 상기 적어도 하나의 당사자가 상기 통화 또는 환전 가치에 대해 상기 금융 거래를 수행할 지의 여부를 결정하는 수단; 및 상기 확인된 적어도 하나의 당사자가 상기 통화 또는 환전 가치에 대해 상기 금융 거래를 수행하지 않는 것에 응답하여 상기 당사자들이 상기 금융 거래를 할 총 통화 또는 환전 가치가 상기 금융 거래의 결정된 통화 또는 환전 가치와 동일할 때까지 상기 금융 거래가 상기 적어도 하나의 당사자에 대한 추가의 당사자들과 수행될 수 있다는 것을 확인하는 수단을 포함한다.
- [0016] 본 발명에 따른 시스템에 따라, 네트워크를 통해 복수의 원격용 데이터 처리 시스템들 중 적어도 하나에 접속된 데이터 처리 시스템이 제공된다. 상기 데이터 처리 시스템은 복수의 당사자들 각각이 상기 데이터 처리 시스템들 중 하나에 해당하는 복수의 당사자들 중 적어도 하나와 적어도 한번 금융 거래를 수행하기 위한 요구를 사용자로부터 받고, 상기 금융 거래에 관한 실시간 금융 정보를 획득하며, 상기 금융 거래가 적어도 하나의 당사자와 수행될 수 있다는 것을 확인하는 프로그램을 포함하는 메모리; 및 상기 프로그램을 수행하는 처리 유닛을 포함한다.
- [0017] 본 발명에 따른 시스템에 따라, 네트워크를 통해 복수의 원격용 데이터 처리 시스템들 중 적어도 하나에 접속된 데이터 처리 시스템이 제공된다. 상기 데이터 처리 시스템은 복수의 당사자들 각각이 상기 데이터 처리 시스템들 중 하나에 해당하는 복수의 당사자들 중 적어도 하나와 적어도 한번 금융 거래를 수행하기 위한 요구를 사용자로부터 받고, 상기 금융 거래에 관한 실시간 금융 정보를 획득하며, 상기 금융 거래의 통화 또는 환전 가치를 결정하고, 상기 적어도 하나의 당사자가 상기 통화 또는 환전 가치에 대해 상기 금융 거래를 수행할 지의 여부를 결정하며, 상기 확인된 적어도 하나의 당사자가 상기 통화 또는 환전 가치에 대해 상기 금융 거래를 수행하지 않는 것에 응답하여 상기 당사자들이 상기 금융 거래를 할 총 통화 또는 환전 가치가 상기 금융 거래의 결정된 통화 또는 환전 가치와 동일할 때까지 상기 금융 거래가 상기 적어도 하나의 당사자에 대한 추가의 당사자들과 수행될 수 있다는 것을 확인하는 프로그램을 포함하는 메모리; 및 상기 프로그램을 수행하는 처리 유닛을 포함한다.
- [0018] 본 발명에 따른 제조물에 따라 데이터 구조를 지니는 프로그램으로 부호화된 컴퓨터 판독가능 메모리 장치가 제공된다. 상기 프로그램은 네트워크를 통해 복수의 원격용 데이터 처리 시스템들 중 적어도 하나에 접속된 데이터 처리 시스템 내의 프로세서에 의해 수행된다. 상기 데이터 구조는 복수의 엔트리들을 지니고, 각각의 엔트리는, 금융 거래의 통화 또는 환전 가치를 저장하는 제1 저장 영역; 및 각각의 제2 저장 영역이 상기 금융 거래에 대한 당사자의 식별 정보 및 상기 당사자가 상기 금융 거래를 할 금액을 저장하는 복수의 제2 저장 영역을 포함하며, 상기 프로그램은 추가의 당사자들 및 상기 당사자들이 상기 금융 거래를 수행할 총 금액이 상기 통화 또는 환전 가치와 동일할 때까지 상기 추가의 적격한 당사자들이 상기 금융 거래를 수행할 금액을 확인한다.

- [0019] 이하 본 발명의 상세한 설명이 더 잘 이해될 수 있도록 하기 위하여, 그리고 해당 기술에 본 명세서가 더 잘 이해될 수 있도록 하기 위하여, 본 발명에 따른 몇몇 특징들을 기술하였다. 물론, 후술되고 첨부된 청구범위의 대상을 형성하는 본 발명에 따른 추가 특징들이 있다.
- [0020] 이러한 점에 있어서, 본 발명에 따른 적어도 하나의 실시예를 상세하게 설명하기 전에, 본 발명은 도면에서 도시되고 이하 설명에서 기술되는 구성요소들의 배열 및 구성의 세부 사항에 제한되지 않는다는 것이 이해되어야 한다. 본 발명에 따른 방법 및 장치는 다양한 방식으로 다르게 실시되고 실행되며 수행될 수 있다. 또한, 본 명세서에서 사용되는 표현 및 용어 뿐 아니라 이하 포함된 개요는 설명을 위한 것이고 제한으로서 여겨지지 않아야 하는 것으로 이해되어야 한다.
- [0021] 당업자는 본 개시가 기초하는 개념은 본 발명의 몇몇 목적들을 수행하기 위한 다른 구조, 방법 및 시스템의 설계의 기초로서 용이하게 이용될 수 있다는 것을 이해할 것이다. 따라서 청구범위는 본 발명에 따른 방법 및 장치의 범위 및 정신을 벗어나지 않는 한 균등한 구성을 포함하는 것으로 여겨지는 것이 중요하다.

실시예

- [0046] 본 발명에 따른 방법 및 시스템은 글로벌 컴퓨터 네트워크를 통해 이용 가능한 자동화된 분산형 금융 서비스를 가능하게 한다. 특히, 본 발명의 시스템은 개인이 어떤 사람 또는 회사에 공동으로 (어떤 작은 금액을) 빌려주게 하고 독립적으로 시장 중심 가격 및 조건으로 정렬하며, 상환 분배를 처리하도록 허용한다. 분산형 자본 시스템은 어떤 원하는 금액으로 중개인 없이 통화를 환산하고 순수 가격, 보증금 또는 다른 대체 가능한 수단 거래에 대해 종래의 방법보다 더 신속하고 저렴하게 대응 당사자들을 일치시킨다. 본 발명에 따른 분산형 자본 시스템은 크기 조절 가능, 네트워크 효과, 및 개방형 공동체의 개념을 포함한다.
- [0047] 본 발명에 따른 방법 및 시스템은 네트워크 협조적인(collaborative) 거래 서비스를 지원하도록 설계된 확장 시스템을 포함한다. 이러한 협조적인 형식이 보다 복잡하고 종래의 시스템보다 더 많은 시스템 구조를 필요로 하기 때문에, 현재의 거래 구조(일 대 일 상거래 관계)는 본 발명의 협조적인 실시예의 기능의 부분집합으로서 이용될 수 있다.
- [0048] 시스템 구조
- [0049] 본 발명의 일 실시예가 이제 본 발명에 따른 방법 및 시스템을 수행하는데 적합한 금융 시스템을 도시하는 도면을 참조하여 기술된다.
- [0050] 본 발명은 클라이언트-서버 환경(도 1 참조)에서 수행되거나 클라이언트 컴퓨터만이 이용되는 분산형 환경에서 수행될 수 있다. 따라서, 본 발명에 있어서, 특정 동작 또는 서비스는 네트워크의 가장자리에서 또는 중심에서 또는 둘 다에서, 클라이언트 또는 서버에서 수행될 수 있다. 따라서, 클라이언트 또는 서버에서, 또는 둘 다에서 원하는 동작/서비스를 위한 대응하는 프로그램이 이용될 수 있다. 본 발명에 있어서, 클라이언트 및 서버 각각에서의 한 쌍의 대응하는 프로그램이 사용자의 관점에서 동일한 동작을 수행한다 하더라도, 상기 프로그램은 도 2에 도시된 바와 같이, 분산형 금융 네트워크에서 완벽하게 동작이 수행되게 하는 것을 포함하는, 시스템 내부에서 상이한 동작을 수행할 수 있다.
- [0051] 보다 상세하게는, 본 발명의 분산형 자본 네트워크에 있어서, 네트워크의 모든 노드는 한 세트의 가장 가까운 이웃을 가지며, 각 노드는 이웃과 인식-유니버스(awareness-universe)를 공유한다. 따라서, 요구하는 당사자에게 질문이 자발적으로 관련될 수 있다. 본 발명은 이것을 "인식 전파(awareness propagation)"로 지칭한다. 따라서, 본 발명의 동작은 도 2에 도시된 바와 같이 클라이언트 컴퓨터만으로 제공될 수 있다.
- [0052] 하지만, 바람직하기로는 적어도 하나의 클라이언트 컴퓨터, 및 아마도 적어도 하나의 서버 컴퓨터가 본 발명에 따른 방법 및 시스템을 수행하는데 사용될 수 있다.
- [0053] 클라이언트-서버 환경에서, 적어도 하나의 클라이언트 및 적어도 하나의 서버는 각각 통신 링크를 통해, 근거리 통신망(LAN), 광역 통신망(WAN), 및/또는 인터넷과 같은 네트워크(도 1 참조)에 접속된다. 본 발명에 따른 방법에서의 단계들은 클라이언트에서 또는 서버에서, 또는 둘 다에서 수행되고, 서버가 사용되는 경우 브라우저 애플리케이션 또는 그런 종류의 다른 것을 이용하여 예를 들어 인터넷을 통해 클라이언트에 의해 액세스되는 서버에서 수행된다.
- [0054] 특히, 도 1 내지 도 3에 도시된 클라이언트는 상기 클라이언트로부터 떨어져 있는 금융 서비스를 액세스하는 경우 (즉 서버에서) 사용자에 의해 동작되는 개인용 컴퓨터, 이동 컴퓨터 장치와 같은 이동 단말기, 이동 전화 또

는 이동 데이터 조직 수단(organizer)일 수 있다. 도 1에는 단지 2개의 클라이언트가 도시되지만, 당업자는 다른 클라이언트들 또는 서버들에 접속되는 복수의 유사한 클라이언트들이 있을 수 있다는 것을 알 것이다.

[0055] 도 3에 도시된 바와 같은 클라이언트 컴퓨터(10)는 전형적으로 클라이언트 데이터 처리 수단 또는 메커니즘으로서 프로세서(11)를 포함한다. 상기 프로세서(11)는 중앙 처리 장치(CPU)(12) 및 입력/출력(I/O) 인터페이스(13)를 포함한다. 상기 클라이언트 컴퓨터(10)는 또한 데이터 구조(16)를 구비하는 프로그램(15)을 갖는 메모리(14), 모든 것이 접속되는 버스(17), 및 입력 장치 또는 수단(18), 디스플레이(19)를 포함하고, 또한 제2 저장 장치(20)를 포함할 수 있다. 상기 버스(17)는 클라이언트(10) 내부에 있을 수 있고 키보드 또는 입력 장치(18)에 어댑터를 포함할 수 있거나, 외부 접속을 포함할 수 있다.

[0056] 상기 데이터 구조(16)는 복수의 엔트리들을 포함할 수 있고, 각 엔트리는 적어도 하나의 제1 저장 영역 및 복수의 제2 저장 영역을 포함한다. 상기 제1 저장 영역은 금융 거래의 통화 또는 환전 가치를 저장한다. 상기 제2 저장 영역 각각은 상기 금융 거래에 대한 당사자의 식별 정보 및 상기 당사자가 상기 금융 거래를 할 금액을 저장한다. 프로그램은 추가의 당사자들 및 상기 당사자들이 상기 금융 거래를 수행할 총 금액이 상기 통화 또는 환전 가치와 동일할 때까지 상기 추가의 적격한 당사자들이 상기 금융 거래를 수행할 금액을 확인한다. 상기 데이터 구조는 또한 당업자가 이하 설명으로부터 이해하는 바와 같이 다른 저장된 정보 또는 정합 코드와 관련된 것을 포함하는 대안적인 실시예들을 구비할 수 있다.

[0057] 본 발명에 따른 방법 및 시스템에 있어서, 클라이언트(10)는 어드레스 또는 포트에 의해 명시되는 통신 중단 포트를 이용하여, 클라이언트 통신 수단 또는 메커니즘과 같은 통신 링크(21)를 통해 다른 클라이언트들(10) 또는 서버들(30)에 접속된다. 상기 통신 링크는 이동 통신 링크, 교환 회선 통신 링크를 포함하거나, LAN, WAN, 인터넷 또는 그들의 조합과 같은 데이터 처리 장치의 네트워크를 포함할 수 있다. 상기 통신 링크(21)는 서버(30)와 통신을 설정하고 유지하기 위하여 다양한 통신 프로토콜을 수행할 수 있는 어댑터 유닛일 수 있다. 상기 통신 링크(21)는 하드웨어의 특수한 부분으로 구성되거나 대응하는 프로그램 명령어를 수행하는 일반 CPU에 의해 구현될 수 있다. 상기 통신 링크(21)는 대응하는 프로그램 명령어를 수행하는 프로세서(11)에 적어도 부분적으로 포함될 수 있다.

[0058] 클라이언트(10)의 프로세서(11)는 내장되거나 외장될 수 있고, 소정의 동작에 적합한 프로그램(15)을 수행한다. 상기 프로세서(11)는 메모리(14)에 액세스할 수 있고, 상기 메모리에는 소정의 동작을 수행하기 위한 데이터 구조(16) 및 프로그램(15)을 포함하는 적어도 하나의 일련의 코드 명령어가 저장될 수 있다. 상기 메모리(14) 및 프로그램(15)은 클라이언트 내에 위치하거나 외부에 위치할 수 있다.

[0059] 상기 프로그램(15)은 원하는 동작 또는 서비스를 수행하기 위한 별개의 프로그램 코드를 포함하거나, 서비스를 제공하는 더 큰 프로그램의 모듈의 부분일 수 있다. 상기 프로그램(15)은 또한 추가로 후술되는 바와 같이, 서비스의 부분 동작을 수행하는 복수의 모듈을 포함할 수 있다.

[0060] 상기 프로세서(11)는 복수의 서비스/동작에 대응하는 복수의 프로그램(15)을 액세스 및/또는 수행하도록 적합화될 수 있는 것으로 이해된다.

[0061] 프로그램(15)에 의해 제공되는 동작 또는 서비스는 예를 들어 사용자 인터페이스 지원, 전자 우편 애플리케이션 수행, 금융 거래 설정 등일 수 있다.

[0062] 클라이언트(10)의 입력 수단(18)은 키보드, 마우스, 또는 음성 처리 수단과 같은 표준 입력 장치를 포함할 수 있다.

[0063] 저장 장치(20)는 사용자를 위한 특별한 서비스를 제공하는 경우 텍스트 파일, 데이터 파일, 이미지, 오디오, 비디오 파일 등과 같은 적어도 하나의 데이터 파일을 저장한다. 저장 수단(20)과 같은 데이터 저장 장치는 예를 들어 네트워크(22)를 통해 접속되는 분산형 데이터베이스를 포함하는 데이터베이스일 수 있다. 상기 저장 장치(20)는 LAN 또는 WAN과 같은 통신 네트워크를 통해 또는 직접 서버(30) 및/또는 클라이언트(10)에 접속될 수 있다. 내부 저장 장치(20) 또는 외부 저장 장치(23)는 옵션이고 데이터는 또한 예를 들어 네트워크를 통해 수신되어 직접 처리될 수 있다.

[0064] 서버(30)가 비분산형(non-distributed) 환경에서 사용되는 경우, 서버(30)는 서버 데이터 처리 수단 또는 메커니즘인 CPU(25) 및 I/O 인터페이스(26)를 구비하는 프로세서(24)를 포함하지만, 또한 하나 또는 복수의 기계에 복수의 개별적인 프로세서들(24)을 포함하는 분산형 CPU(25)로 구성될 수 있다. 서버(30)의 프로세서(24)는 일반 데이터 처리 유닛이지만, 바람직하기로는 많은 자원, 즉 높은 처리 능력 및 많은 양의 데이터를 저장하는 큰

메모리를 갖는 데이터 처리 유닛일 수 있다.

- [0065] 서버(30)는 또한 버스(31)에 의해 모두 접속되는, 데이터 구조(29)를 구비하는 프로그램(28)을 갖는 메모리를 포함한다. 상기 버스(31) 또는 유사한 접속 라인은 또한 서버(30)가 분산형 시스템으로 구성되는 경우 외부 접속으로 구성될 수 있다. 상기 서버 프로세서(24)는 사용자에게 다양한 서비스를 제공하기 위하여, 즉 클라이언트(10)를 동작시키는 사용자가 원하는 다양한 금융 거래를 수행하기 위하여 바람직하기로는 다수의 프로그램을 저장하기 위한 저장 장치(즉, 저장 장치(32))에 액세스할 수 있다.
- [0066] 데이터 구조(29)는 복수의 엔트리들을 포함할 수 있고, 각 엔트리는 적어도 하나의 제1 저장 영역 및 복수의 제2 저장 영역을 포함한다. 상기 제1 저장 영역은 금융 거래의 통화 또는 환전 가치를 저장한다. 상기 제2 저장 영역 각각은 상기 금융 거래에 대한 당사자의 식별 정보 및 상기 당사자가 상기 금융 거래를 할 금액을 저장한다. 프로그램은 추가의 당사자들 및 상기 당사자들이 상기 금융 거래를 수행할 총 금액이 상기 통화 또는 환전 가치와 동일할 때까지 추가의 적격한 당사자들이 상기 금융 거래를 수행할 금액을 확인한다. 상기 데이터 구조는 또한 당업자가 이하 설명으로부터 이해하는 바와 같이 다른 저장된 정보 또는 정합 코드와 관련된 것을 포함하는 대안적인 실시예들을 구비할 수 있다.
- [0067] 서버(30)는 단일 유닛이거나 복수의 서버들(30) 또는 데이터 처리 유닛들의 분산형 시스템일 수 있고, 서로 직접 또는 간접 접속된 다수의 사용자에게 의해 공유될 수 있다. 서버(30)는 클라이언트(10)로부터의 요구를 서비스하는데 필요한 원하는 동작을 위해 적어도 하나의 서버 프로그램(28)을 수행한다.
- [0068] 서버(30)로부터의 통신 링크(33)는 바람직하기로는 복수의 클라이언트(10)와 통신하도록 적합화된다.
- [0069] 서버 프로그램(28)은 사용자가 금융 거래를 조합하고, 제안된 금융 거래를 테스트하며, 각 거래 요구가 금융 시스템을 통해 안전하게 전송되고 고유 거래 번호를 가지고 코드 스탬프(codestamp)되게 보장하도록 허용하는 것과 같은, 금융 서비스를 사용자에게 제공하는 것에 관련된 다수의 동작들을 제공하는 것에 관련될 수 있다.
- [0070] 본 발명의 시스템이 어떤 기능을 수행하는 것으로 기술되는 것을 유념하라. 하지만, 당업자는 프로그램이 시스템 실제보다는 기능을 수행하는 것으로 알 것이다.
- [0071] 본 발명의 하나의 구성의 태양이 메모리에 저장되는 것으로 기술되었지만, 당업자는 본 발명에 따른 시스템 및 방법의 모두 또는 일부가 현재 알려져 있거나 나중에 개발되는 다른 형태의 ROM 또는 RAM, 또는 인터넷과 같은 네트워크로부터 수신된 반송파(carrier wave), 하드디스크, 플로피 디스크, 및 CD-ROM과 같은 보조 저장 장치와 같은, 다른 컴퓨터 판독 가능 매체에 저장되거나 독출될 수 있는 것으로 이해할 것이다. 추가로 시스템의 특정 구성요소들이 기술되었지만, 당업자는 본 발명에 따른 시스템 및 방법을 가지고 이용하는데 적합한 시스템이 추가 또는 상이한 구성요소를 포함할 수 있다는 것을 이해할 것이다.
- [0072] 상술된 특징 및 처리 동작들은 전용 하드웨어에 의해 구현되거나 데이터 처리 유닛에서 수행되는 코드 명령어를 포함하는 프로그램으로서 구현될 수 있다는 것을 유념한다. 상기 일련의 동작들 중 일부가 하드웨어에서 수행되고, 상기 처리 동작들 중 다른 것은 소프트웨어를 이용하여 수행되는 것이 또한 가능하다.
- [0073] 또한, 본 발명에 따른 방법 및 시스템은 클라이언트에서 제공되는 인터페이스의 롤 다운(roll-down) 윈도우, 또는 대화 상자, 메뉴에서 선택 버튼을 포함하는 사용자 선택 수단을 제공하여 수행되고, 사용자는 키보드 또는 그런 종류의 다른 것을 통해 명령을 입력할 수 있다. 선택 수단은 전용 하드웨어 부분으로 구성되거나 그 기능은 예를 들어 선택을 입력하기 위한 키보드 및 선택 윈도우를 디스플레이하는 디스플레이 유닛을 포함하는, 클라이언트 프로세서에서 수행되는 코드 명령어에 의해 수행될 수 있다.
- [0074] 프로그램 모듈들
- [0075] 본 발명의 프로그램(적어도 15 또는 28 또는 둘 다)의 일 실시예는 4개의 주요한 프로그램 모듈을 포함한다: 벡터-흐름 형태(VfT; Vector-flow Topography), 코드 분할 다중 거래(CDMX; Code-Division Multiple Transaction), 스프레드 벡터 해상도(SVR; Spread Vector Resolution), 및 매트릭스 쿼터마스터(MaQs; Matrix Quartermaster). 각 프로그램 모듈은 독립적으로 다양한 신규 금융 서비스를 지원할 수 있고, 다른 애플리케이션들과 개별적으로 사용될 수 있다. 하지만, 추가 후술되는 다른 프로그램 모듈과 함께 접속되어 작용하는 경우 본 발명에 따른 분산형 자본 시스템 플랫폼을 생성한다.
- [0076] 하지만, 당업자는 본 발명의 특징을 달성하는데 사용될 수 있는 추가 모듈 또는 프로그램이 있거나, 프로그램 모듈들이 동일한 목적으로 단일 프로그램에 조합될 수 있다는 것을 알 것이다. 또한, 상기 5가지 주요 프로그램 모듈들의 특징 및 처리 동작들은 전용 하드웨어에 의해 구현되거나 데이터 처리 유닛에서 수행되는 코드 명령어

를 포함하는 프로그램으로서 구현될 수 있다. 상기 일련의 동작들 중 일부가 하드웨어에서 수행되고, 상기 처리 동작들 중 다른 것은 소프트웨어를 이용하여 수행되는 것이 또한 가능하다.

[0077] 상술된 바와 같이, 본 발명에 따른 이하 프로그램 모듈은 분산형 자본 시스템 구조의 부모(parent) 프로그램에 의해 수행될 수 있고, 클라이언트-서버 환경의 일부이거나 클라이언트 컴퓨터의 분산형 플랫폼의 일부일 수 있다.

[0078] 1) 벡터-흐름 형태 프로그램 모듈

[0079] 벡터-흐름 형태(VfT; Vector-flow Topography) 프로그램 모듈은 사용자 인터페이스를 수행하고 모든 유형의 거래의 유전형 구성, 테스트, 및 관리를 가능하게 한다. 상기 VfT 모듈은 사용자가 온-스크린 거래 작업 패드에 정렬되고 연결된 영상 객체들을 가지고 금융 거래를 신속하게 구성하고, 수행하기 전에 자금, 지불, 계정 잔고 등의 제안된 흐름을 테스트할 수 있게 한다. VfT 프로그램 모듈은 또한 거래 이후에 또는 시스템에 거래 명세의 초기 입력 이후의 어떤 다른 시간에 다양한 데이터, 확인, 및 다른 표시를 수신하여 사용자에게 디스플레이한다.

[0080] 따라서, 사용자들은 거래 이벤트를 구성하고 저장할 수 있고, 원하는 만큼 자주 다시 사용할 수 있다. 상기 VfT 프로그램 모듈은 사용자가 정보가 CDMX 프로그램 모듈(이하 참조)에 의해 추적되는, 구성된 거래의 테스트 및 모든 수행의 이력을 용이하게 액세스하도록 허용한다. 이런 식으로, 사용자는 항상 자신의 거래 이벤트 중 어느 것에 대한 모든 세부 항목을 검토할 수 있다.

[0081] 계산서-지불 등과 같은 간단한 거래를 조합하고 수행할 수 있지만, VfT 프로그램 모듈은 또한 소위 "복잡한(compound)" 거래를 처리한다. 여기서 "복잡한"은 원하는 상업적인 배치 결과를 달성하기 위하여 다수의 또는 합성의, 중간 단계들 또는 거래들을 구성하는 것으로 정의된다. VfT 프로그램 모듈의 "끌어 놓기(drag and drop)" 기능은 복잡한 거래가 용이하게 조합되고 테스트되며 수행되도록 허용한다.

[0082] 2) 코드 분할 다중 거래(CDMX) 프로그램 모듈

[0083] CDMX 프로그램 모듈은 본 발명의 분산형 자본 시스템을 가능하게 하는 추적 및 어카운팅(accounting)을 수행한다. CDMX 프로그램 모듈은 2가지 유형의 데이터를 동시에 (금융 및 통신), 안전하게 은밀히 처리하고, 2가지 범위 사이의 관계를 보존하고 이용하지만, 또한 전송의 완전한 익명, 분리, 기밀성을 유지한다. 따라서 CDMX 프로그램 모듈은 통신 프로토콜 및 거래 프로토콜 모두를 이용하고, 정확한 데이터 요건을 감시하고 일치시켜, 임의-당사자 접속이 달성되게 하고(SVR 프로그램 모듈의 알고리즘에 의해 처리되는, 이하 참조), 임의 당사자들은 완전히 대규모의 대응 당사자의 일치감에 따른다. CDMX 프로그램은 임의 당사자 정합을 완성하기 위하여 다중 평균 정합 엔진(SVR 모듈 - 이하 참조)과 인터페이스하고, 고안된 추적 코드를 본 발명의 DCS의 금액에 적합하게 추가한다. 비록 금융 벡터가 상이한 더 작은 구성요소로 분리되는 경우 또는 나중에 재결합되는 경우에도, CDMX 프로그램 모듈은 제로섬(zero-sum) 계산 환경에서 시스템을 통과하는 모든 구성요소들을 추적한다.

[0084] CDMX 프로그램 모듈은 사용자 프로그램 거래의 4가지 유형의 카테고리를 추적할 것이다. 실시간, 수동, 고정 및 분할. 실시간(Real-time)은 거래 요구의 수행시에 또는 설정 시간이 만료된 경우에 즉시 처리를 의미한다. 수동(Passive)은 사용자가 처리의 타이밍 또는 양을 결정하지 않고, 시스템이 사용자가 표시한 시간 제한이 만료되기 전에 최적 시간 및/또는 임의로 요구를 수행하는 것을 의미한다. 고정(Fixed)은 사용자가 수행을 위한 한 세트의 요건을 표시하고, 거래는 상기 요건이 충족됨으로써 트리거되지 않으면 발생하지 않는다는 것을 의미한다. 분할(installment)은 다른 3가지 유형의 조합으로 구성되고(예를 들어 소정 시간에 발생하는 동일한 거래일 수 있는), 시스템에 입력되는 경우 더 작은 구성요소 거래들로 분해되고, 각각은 모든 분할을 전체 부모 거래에 연결하는 특별한 부모 코드를 가지고 태그(tag)된다.

[0085] CDMX 모듈을 위한 프로토콜은 적합한 데이터베이스의 CDMX 모듈 장부에 의해 보유되고 기록되는 모든 거래 세부 내역을 가지고, 협조하는 복잡한 이벤트(이하 더 상세하게 기술)인지 또는 전통적인 간단한 이벤트인지에 상관없이, 프로그램이 어떤 거래를 발신 계정 쪽으로 추적할 수 있도록 설계된다. 분산형 자금 이전(DFT; Distributed Funds Transfer) 거래가 풍부하고 예정된 타겟 계정이 즉시 보이지 않는 분산형 환경에서조차, CDMX 프로그램 모듈은 프로그램의 CDMX 장부, 또는 분산형 클라이언트/서버/서버트(servent)의 매우 작은 부분 집합이 사후(post facto) 분산을 재구성하도록 허용하고, 자금 대체 네트워크의 되감기 검토를 달성하며, 어떤 분산형 자본 시스템 거래의 세부 사항(즉, 무엇, 어떻게, 얼마나, 누가 등)을 결정할 수 있다.

[0086] CDMX 프로그램 모듈이 기록한 정보는 분산형 클라이언트/서버/서버트에서조차 암호화된 형태로 유지될 것이고, 소환장 등과 같은 공적이고 법적 행위로 인한 것과 같이, 시스템 모니터에 의해 신중한 행동에 의한 것을 제외

하고 어떤 사람에게 이용가능하지 않다. CDMX 프로그램 모듈은 또한 분산형 장치에 있는 내장된 기능을 포함할 수 있다. 이러한 기능은 조화를 위해 달성된 CDMX 데이터를 중앙 수집/감시 서버 등에 자동으로 전송한다.

[0087] 본 발명의 CDMX 프로그램 모듈은 다음 기능을 수행한다. a) 코드 스탬핑(Codestamping), b) 코드 사이클링(Codecycling), 및 c) 코드정합 수집(Codematch Aggregation).

[0088] 코드 스탬핑 및 코드정합 수집 서버 프로그램은 매트릭스 쿼터 마스터(Matrix Quartermaster) 모듈(추가 후술되는)과 절차적인 관계를 가지며, 코드 사이클링 서버 프로그램은 스프레드 벡터 해상도 프로그램 모듈(추가 후술되는)과의 반복적인 교환을 포함하는 반복 처리를 수행한다는 것을 주목한다.

[0089] 코드 스탬핑에 있어서, 코드 스탬핑은 어떤 신규 활동이 개시될 때마다 발생한다. 시스템에 입력한 모든 거래 이벤트 요구는 코드 스탬핑 서버 프로그램을 사용하여 코드 스탬핑되고, 프로그램은 상기 요구를 입력한 사용자에게 요구된 거래에 관한 부모 코드가 되는 고유 거래 번호를 할당한다. 상기 고유 거래 번호는 거래를 개시한 계정 또는 사람의 어떤 고유 식별자 뿐 아니라 거래에 고유한 부호화(coding)를 포함한다.

[0090] 협조하는 거래 이벤트(후술되는)에 있어서, 프로그램은 거래 번호를 복사할 수 있지만(거래에 하나보다 많은 참가자(즉 대응 당사자들)가 있기 때문에), 어떤 거래 번호가 번호에 부가된 고유한 사용자-계정 식별자를 가지고 있기 때문에 부모 코드 스탬프는 각각 고유하다는 것을 유념한다.

[0091] SVR 프로그램 모듈(이하 참조)이 대응 당사자 정합을 완료(결정)할 때마다, 프로그램은 또한 협조적인 거래에서의 정합된 쌍을 태그하는 코드 사이클링 기능을 수행한다. 코드 사이클링이 요구된 거래의 유형에 필요한 경우, 코드 사이클링 프로그램 모듈은 몇몇 추가 자식 코드를 생성하여 부모 거래를 추적할 수 있는 스프레드 벡터 해상도(SVR) 프로그램 모듈에 따라 지시되고 할당되는 구성 금액 각각을 추적한다. 기록은 프로그램에 의해 암호화된 형태로 유지될 수 있다.

[0092] 프로그램은 또한 코드정합 수집 기능을 수행한다. 상기 코드정합 수집 기능에 의해, CDMX 프로그램 모듈은 수행을 위해 매트릭스 쿼터마스터(MaQs) 프로그램 모듈에 데이터를 전달하기 전에 금액을 얻거나 활동을 최적화하기 위하여, 단일 사용자 또는 계정에 대해 모든 이용가능한 거래 금액을 모은다.

[0093] 코드 사이클링은 전통적인 간단한 또는 전통적인 복잡한(추가 후술되는) 거래 유형을 처리하는데 필요하지 않다는 것을 유념한다.

[0094] 3) 스프레드 벡터 해상도(SVR) 모듈

[0095] SVR 프로그램 모듈은 다차원(즉, 벡터) 객체의 입력을 취하고, 각 차원 양을 자동으로 분리하며, 수행되기 전에 독립적으로 처리한다. 처리하기 전에 차원 양을 분리하는 장점은 각 차원에서의 효율이 함께 처리되는 다차원의 교차에서 액세스되거나 이용가능한 효율을 얻는 것보다 더 충분히 얻어질 수 있다는 것이다. 다시 말하면, SVR 모듈은 무리한(즉 다차원) 객체의 복잡함에 구속되지 않는 효율을 얻을 수 있다.

[0096] 따라서, 종래의 2명의 연결되지 않은 사용자들로부터의 2개의 지불은 함께 처리될 수 없는데, 왜냐하면 단일 차원의 지불 금액이 서로 관련되지 않은 다차원 객체로서 2개의 지불을 수행하는, 지불인 및 피지불인의 식별 정보와 같은, 다른 차원의 지불 변수에 의해 복잡하기 때문이다. 하지만, SVR 프로그램 모듈은 각 차원 값을 분리하여 별개로 처리할 수 있다. 실제로, 이것은 완전히 관련되지 않은 당사자들을 함께 동일한 네팅(netting) 합계에 포함하는 것을 의미한다. SVR 프로그램 모듈은 대응 당사자 정합을 할당하는데 효율적인 다양한 알고리즘을 포함하는 정합 유틸리티이다. SVR 프로그램 모듈은 금융 벡터의 디지털 표현을 처리하고(즉, 구성 벡터의 모든 수행을 결정하는 예를 들어 집합되거나 결합된 벡터를 수학적으로 산출하도록 결합되고 조작될 수 있는), 상기 디지털 표현을 받아들이며, 이 정보를 정확하게 모든 것을 고려할 수 있는 시스템(후술되는)에 출력하고, 결정된 벡터를 수행한다.

[0097] SVR 프로그램 모듈은 특히 협조하는 복잡한 거래(이하 참조)의 대응 당사자들은 동일하게 잔고가 맞지 않는 경우, 즉 거래 환경의 양측간의 동일하지 않은 잔고가 있는 경우, 동작하도록 설계된다. 이러한 비대칭 상황에 있어서, SVR 프로그램 모듈은 지연 없이 거래를 처리하고 벡터-흐름 입자(particulation)를 통해 비대칭을 해결한다. 이하 더 상세하게 후술되는 바와 같이, SVR 정합은 이론적으로 많은 수의 참가자들을 갖는 크게 유동적인 시장의 제로-섬 "시장-메커니즘(market-mechanism)"이 제한된 참가자 거래 환경에 이르도록 허용한다. 비록 전체 거래에서 비 제로 섬 계정이 순간적으로 나타날 수 있지만, 시스템 거래 영역의 매크로 벡터 흐름 처리는 제로 섬 원리에 충실하다. 연속하는 흐름이 처리되기 때문에, 모든 이전 거래가 제로섬이 되고, 따라서 프로그램에 의해 어느 쪽에서도 남아있는 금액이 없는 정확한 정합이 있어야 한다.

- [0098] CDMX 프로그램 모듈은 주어진 기간에 정합되는 시스템에서의 남아있는 가치가 없을 때 완료되는 매크로 장부를 유지한다.
- [0099] 이런 식으로, SVR 프로그램 모듈은 네팅 기능, 자금 대체 알고리즘, 통화 환전 알고리즘, 통합 계정 알고리즘, 대부 신디케이션, 및 ATM-공유 등(추가 후술되는)을 수행한다.
- [0100] SVR 프로그램 모듈에서 사용되는 알고리즘들
- [0101] 상세하게는, 협조하는 복잡한 거래(상세한 설명을 위해 이하 참조)를 충족하기 위하여 SVR 프로그램 모듈에 의해 사용되는 하나의 가능한 알고리즘이 도 4에 도시된다. 도 4에는 엔을 달러로, 달러를 엔으로 환전을 요구하는 대응 당사자들이 있다.
- [0102] 도 4에 있어서, 니어사이드(Nearside)는 가능한 알고리즘의 교환 처리를 설명하는데 사용되는 참조부호를 나타낸다. 니어사이드는 선처리 상태이다. 모든 환전이 통화 쌍(currency pair)으로 구성되기 때문에, 오프사이드(Oppside)는 환전 거래를 수행하는데 사용되는 타겟(대응 당사자) 통화에서의 요구에 대한 참조부호이다. 따라서, 니어사이드 요구를 충족하는데 필요한 오프사이드 요구의 전체 수가 최소가 되는 경우 최적 효율이 발생한다. 프로그램은 이러한 것을 기초로 하여 환전에 우선 순위를 매긴다. 능동-탐색(active-seeking) 클래스 금액이 바람직하기로는 단일 오프사이드 금액으로 채워진다는 것을 유념한다.
- [0103] 따라서, 시스템으로의 입력이 있는 경우, SVR 프로그램 모듈은 금액이 수학적인 평균 위에 있는 경우, 수동-채움(passive-fill)으로서, 또는 금액이 수학적인 평균 아래에 있는 경우 능동-탐색으로서 환전되는 금액을 자동으로 분류할 것이다. 시스템을 통해 흐르는 벡터/자금의 양에 의존하여 프로그램이 수학적인 평균을 다시 계산하기 때문에 수학적인 평균은 계속해서 변하는 것을 유념한다.
- [0104] 모든 요구는 SVR 프로그램에 의해 모든 요구가 실제로 처리되는 때에 실시간 요구인 것을 특징으로 하는 것을 유념한다. 단일 포인트 고정 시간(즉, 어떤 날짜에) 또는 이벤트 트리거되는 수행 특성(즉, 예를 들어 다우 존스 지수가 10,000에 도달하는 경우)을 가지지 않고, 시간 기간 제한 특성(즉 어떤 날짜에 의해)을 갖는 사용자 프로그램 거래로부터 나오는 수동 요구의 인구가 있기 때문에, 프로그램에 의해 요구되는 능동 재분류는 몇몇 또는 모두를 실시간으로 변환하기 위하여 수동 거래의 저장소에 들어가서 시스템에 유동성을 가져오도록 처리하며, 완화된 휘발성(volatility)의 고른 진행(즉 일관된 유동성)을 시도한다.
- [0105] 또한, 몇몇 경우들에 있어서, 프로그램은 다수의 수단을 구비하고 2개의 수단을 구별하기 위하여 체적(volume)-평균 세퍼레이터(separator)를 제공할 수 있다(도 5 참조). 예를 들어, 거래의 체적이 미리 설정된 트리거를 활성화하기에 충분히 증가하는 경우, 프로그램은 상기 체적을 처리하기 위하여 신규 평행한 채널을 생성하여 증가된 벡터 흐름을 수용할 것이다. 트리거는 상기 체적을 섹션으로 나눈다. 각 섹션은 이전의 단일-분할기(divider) 프로그램 기능과 같이 동작한다(도 4 참조). 도시된 바와 같이 상반부에 있는 더 큰 벡터들이 분해되고 정합이 일어나는 경우, 나머지는 하반부로 떨어질 것이다. 체적이 감소함에 따라, 미리 설정된 트리거가 개시되고 프로그램은 세퍼레이터를 이동하여 프로그램 기능을 원래의 단일-분할기 상태로 되돌릴 것이다.
- [0106] 능동-탐색 금액은 각 타겟 통화의 수동-채움 금액을 목표로 한다. 2개의 가능한 정합가능한 요구는 서로 능동적으로 찾을 수 없다. 금액이 능동-탐색으로서 프로그램에 의해 원래 분류되는 경우, 프로그램은 단일의 대응 당사자 정합 금액을 가지고 요구를 충족하려 할 것이다(도 4 참조). 금액이 수동-탐색으로서 프로그램에 의해 원래 분류되는 경우, 상기 금액은 평균 아래로 떨어질 때까지 능동-탐색 금액에 의해 줄어들 것이고 자신이 능동-탐색 금액이 된다. 그 다음 금액이 채워지고 거래가 완료된다.
- [0107] 따라서, 통화를 환전하기 위한 요구의 정합은 몇 가지 알고리즘을 이용하여 정합될 수 있다. 오프사이드에 있는 능동-탐색 거래가 니어사이드에 있는 수동-채움 거래로부터 정합되거나 그 반대, 또는 엇갈린 선형 정합에 의해, 또는 비스듬한 니어사이드 모두가 반대측 정합을 찾는다(도 7a 참조).
- [0108] 시장과 거래의 보통 활동에서, 때때로 벡터 흐름에 불균형이 발생한다. 매일 판매자가 어떤 가격(E)으로 어떤 것을 제공하고 있는 경우가 있지만 다수의 구매자는 훨씬 낮은 가격(D)을 지불하려고 하는 경우와 같이, 직접 참가자로부터의 시스템에 보이는 유동성의 부족은 한 쌍의 삼각형으로 표현될 수 있고, 한 쪽은 다른 쪽보다 훨씬 더 크다. 이것은 이용가능한 큰 체적의 벡터(즉, "판매자")와 다른 쪽에 있는 너무 작은 대응 당사자(즉, "구매자")를 나타낸다.
- [0109] 이러한 불균형은 참고로 지향성 대응 당사자 정합으로서 구성되는 국내 이벤트 또는 트랜스-통화 이벤트(trans-currency event) 또는 거래 유형의 일 측이 다른 측보다 더 풍부한 어떤 수의 다른 상황일 수 있다. SVR 프로그

램 모듈은 자기 선택 벡터의 유입을 취할 것이다. 자기 선택은 모든 벡터들이 적어도 하나의 동일한 변수를 갖는 것을 보장한다. 예를 들어, 국내 분산형 자금 대체에 있어서, 통화는 모든 벡터들에 대해 동일하고, 정합 쌍(match-pair)되는 방향 변수이다. 통화 환전에 있어서, 방향은 중요하지 않고 따라서 정합 쌍 원리는 통화-쌍(currency-pair)이다.

[0110] 이러한 종류의 불균형의 경우에 있어서, 시스템은 가능한 많이 처리할 것이고, 이것은 한 쪽에 있는 이용가능한 모든 벡터가 처리될 때까지를 의미하고, 반대쪽에 있는 이용가능한 벡터가 제로가 되도록 실험적인 상태로 불균형이 된다(도 6 및 도 7a 참조).

[0111] 이것이 발생하는 경우, 대규모-세계적인 유동성 위기로 인한 것 또는 더 유사한 것일 수 있고, 적은 사용자 인구로 인한 것일 수 있다(적어도 시스템이 여전히 도입되고 있고 단순히 유동성이 문제가 되지 않을 정도까지 충분한 사람이 없는 경우). 이 경우에 있어서, 시스템의 프로그램은 유동성이 예를 들어 중개인에 의해 제공될 수 있고, 벡터의 소유자들이 자신의 거래가 충족되게 할 수 있는 (즉, 전통적인 채널로부터의 자본 유입 - 도 7a 참조) 더 전통적인 채널 쪽으로 나머지 일 측 벡터 흐름을 향하도록 미리 설정될 수 있다. 이런 식으로, 사용자는 시스템 참가자들로부터 제3 당사자 대형 은행 중개인/상인으로의 유동성 풀(pool)의 스위치를 인식할 수 없다.

[0112] 어카운팅 네팅(Accounting Netting)

[0113] 회사가 정해진 시간 기간 동안 다수의 거래를 하는 경우, 예를 들어 실체들이 많은 대응 당사자들과 거래 및 금융 관계를 갖는 법인 회사에 있어서, 부채(부정적) 및 수취 어음(긍정적)이 있을 것이고, 종종 동일한 대응 당사자와 긍정적 및 부정적인 거래를 할 것이다.

[0114] 하나의 관계에서 단일 결과를 얻는 동일 당사자 거래보다는, SVR 프로그램은 특정 네팅(netting) 처리를 결정하고, 대신에 모든 당사자의 기간-부정적(period-negative) 및 기간-긍정적(period-positive) 가치를 합한다(회사가 거래해야 하는 모든 당사자를 통해). 따라서, 직접적인 동일-당사자 네팅은 바람직하기로는 거래 영역에 있는 모든 당사자들 중에서 매크로 스프레드 네팅을 위한 단계를 설정하기 위하여 바이패스된다.

[0115] 거래 스프레드-벡터 네팅(SVN; Spread-Vector Netting)

[0116] 스프레드-벡터 네팅 또는 분산형 직접 환전(Distributed Direct Exchange)은 다수 당사자 컴팩션(compaction)이고, 어디에 있는 어떤 당사자가 거래하고 있는 경우 가능하다. SVR 프로그램 모듈은 SVN 기능을 수행하는 경우 어떤 및 모든 거래 사이즈, 개인 또는 단체를 처리할 수 있다. 종래의 네팅에서 보다 훨씬 더 많은 수의 참가자들이 가능하다.

[0117] SVN에서, 다수 당사자들은 항상 스프레드-네팅을 차지하고 있고, 상호 책임 및 연계된 정리가 필요하지 않다. 종래의 네팅은 고정수의 참가 당사자들 뿐 아니라 네팅하는 고정 시간 기간을 요구한다. 스프레드-네팅은 이들 중 어느 것도 요구하지 않는다. 복잡함은 당사자의 수에 상관되지 않을 것이고 리스크가 당사자의 수에 역으로 상관할 것이다.

[0118] SVN에서 (채무를 충족하기 위한 지불 송금이 서클에 있는 각 채무자가 지불받기를 대기하기 때문에 중단될 수 있도록 원형 배열로 다수의 채무자로서 정의되는) 채무 서클을 스프레드-네팅하기 위하여, 프로그램은 CPN 아래에 기술된 것과 같이 네팅하지만, 벡터 관계를 유지하는 것 대신에, 프로그램은 각 당사자에 대한 네팅 결과가 네팅 이전과 동일한 절대 값이 되도록 어떤 필요한 벡터로 방향을 바꾼다(도 8a 및 도 8b 참조). 당사자(D)는 당사자(A)에게 빚이 있고, 당사자(C)는 당사자(D)에게 빚이 있는 경우 거래 방향을 추적하는 경우, 스프레드-네팅 이후에, 프로그램은 당사자(D)가 당사자(C)와 당사자(B)를 보상하도록 허용하고, 이전과 거래 방향이 완전히 반대이며, 후자에 대해서는 완전히 새로운 관계가 된다. 스프레드-네팅은 종래 네팅에 비해 위험 감소에 있어서 50% 이상의 개선을 달성한 것으로 나타난다. 따라서, 전체 거래가 34이고, 네팅된 물리적인 청산(settlement)이 8이며, 위험에 대해 감소된 양은 $26/34 = 76\%$ 이다.

[0119] 4) 매트릭스 쿼터 마스터(MaQs)(aka 분산형 은행 융자 신용(DBLC; Distributed Banking Lending Credit)) 프로그램 모듈

[0120] 매트릭스 쿼터 마스터(MaQs; Matrix Quartermaster) 프로그램 모듈은 최종 총 거래 정보를 입력으로서 취하고, 상기 정보를 수행 명령으로 변환하며, 기반 구조가 필요한 것을 통해 거래의 수행을 개시하고, 이미 존재하는 프로토콜을 사용하며, 모든 당사자의 거래가 달성되도록 보장하기 위하여 다수의 계정 및 시스템을 처리하는 협상 관리자이다. MaQs 프로그램 모듈은 CDMX 프로그램 모듈 또는 유사한 추적 모듈과 작용하여 자본 이동의 수행

을 활성화한다.

- [0121] MaQs 프로그램 모듈은 기존의 기반 구조에 대해 명령들을 실행한다. MaQs 프로그램 모듈은 금융 및 은행 세계의 나머지와 상호 작용이 가능하도록 하는 다양한 API들을 유지하거나 통합한다. 이것은 예를 들면, 자동 어음 교환소(Automated Clearing House; ACH), 연방 준비 페드와이어 제도(Federal Reserve Fedwire System), ECN의 아키텔라고(Archipelago), 인스티넷(Instinet), 및 아일랜드(Island), 신용카드사들, 은행들, 유틸리티 회사들의 계정들, 웹 상인들 등을 포함하는 다양한 범위의 실체들을 포함한다.
- [0122] MaQs 프로그램 모듈은 CMDX 프로그램 모듈에 의해 제공된 사용자 계정 정보 및 Vft 프로그램 모듈에 의해 집합된 실행 명령들을 거래를 실행하기 위해 사용되고 있는 기반 구조에 따라 적절하고 의례적인 실행 명령어들로 패키징화한다. 다수의 기반구조 네트워크들이 사용될 수 있기 때문에, MaQs 프로그램 모듈은 거래명령들을 다양한 포맷된 실행 명령어들로 패키징화할 수 있다.
- [0123] 따라서, MaQs 프로그램 모듈은 본 발명의 분산 금융 시스템(distributed financial system)에서 생성되고, 프로그램되고, 해결되거나 또는 그 밖의 거래들을 실행한다. MaQs 프로그램 모듈은 시장들, 제품 및 서비스 제공자들의 더 큰 네트워크로의 연결일뿐만 아니라, 물론 은행들 및 (유틸리티 회사들, 보증 회사들 등과 같은) 모든 다른 적절한 실체들로의 연결이다. 시스템이 (이하에서 참조될 ATM 공유 또는 투자 엔진과 같은 어떤 기능들 또는 서비스들을 구현하는데 필요할 수 있는) 작동들에 포함된 자신의 계정들을 가지는 경우에, 상기 MaQs 프로그램 모듈은 시스템으로부터 자본 벡터들(capital vectors)을 취할 것이고, 자본 벡터들을 시스템에 공급할 것이다.
- [0124] 상기 프로그램이 계정들 및 시장들로부터 검색된 데이터를 디스플레이 및 조작을 위해 시스템으로 전달할 수 있도록, 상기 프로그램은 본 시스템과 (예를 들면, 주식 시장 펀드들, 뉴욕 주식 거래소 등과 같이) 시장에 있는 다른 시스템간의 브리지들(bridges)을 포함한다.
- [0125] 은행 다중화기 API(Bank Multiplexer API)
- [0126] 다음은 MaQs 프로그램 모듈에 대한 성분일 수 있는 한 종류의 API의 예시를 기술한다. 시스템으로 입력되는 시장 데이터 공급(market data-feed) 및 명령어들 그리고 사용자 은행 계정들로의 분배를 위해 처리되고 실행되는 명령어들은 은행 다중화기 프로그램(Bank Multiplexer program)에 의해 처리된다. 여기서, "다중화기(multiplexer)"는 동일한 API가 하나 이상의 계정 또는 기관(institution)에 동시에 접속하기 위해 사실상 이중으로 사용될 수 있다는 것을 나타내기 위해 사용되었다. 본 발명의 프로그램은 배타적으로 연속적인 순서로 거래들을 처리하지 않으며, 오히려 동시에 다수의 거래들을 처리할 것이다.
- [0127] 은행 다중화기 프로그램은, 예를 들면 각 처리된 거래량에 대해 식별자 코드들을 독출하고, 이러한 코드들을 다양한 금융 네트워크 명령어들, 예를 들면 시스템에 등록된 은행 계정들로 전송될 직접 예금(direct deposit) 및 직접 출금(direct-debit) 명령어들로 번역한다. 수동 요구들(passive requests)이 시스템에 입력될 때, 부가 코드는 프로그램으로 하여금 다른 시간대에 이러한 프로그램된 거래량들을 까먹도록 하고, 상기 거래량들을 (이하에서 참고될) 프로그램된 거래를 실행하기 위해 필요로 할 때까지 투자 엔진(investment engine)에서의 사용을 위해 시스템 소유의 계정에 예금한다. 투자 엔진을 구동시키는 상기 프로그램은 필요한 총 자금(aggregated funds)을 사용할 것이며, 이후에 상기 프로그램은 원래의 환전 정합량(exchange-matched amounts)을 적절한 최종 사용자의 대응 당사자들의 계정들(counterparty accounts)로 송금함으로써, 향후의 사용을 위해 시스템에서의 이익을 보유한다.
- [0128] 5) 코드 분할 다중 거래(CDMT: Code Division Multiple Transaction) 프로그램 모듈
- [0129] 코드 분할 다중 액세스는, 동일한 통신 기반구조를 통해 대역폭을 증가시키려는 특수한 목적을 위해, 많은 다른 것들과 얽힌 단일 소스-싱크 쌍들(single source-sink pairs)간의 인터링크된 연결(즉, 음성/데이터 전송)을 유지하도록 특별히 설계된 통신 기술이다. 스프레드 벡터 해상도(Spread Vector Resolution)와 관련하여 작용하는 코드 분할 다중 거래(CDMT: Code Division Multiple Transaction)는 원격통신 대역폭 제공을 고려하지 않으며, 오히려 대체 가능한-거래 상대방들(fungible-trade counterparties)의 정합을 고려하며, 게다가 (접속된 세션, 예컨대 전화중의 전체 주기동안에 세트 출력으로 전달된, 세트 입력만을 취할 수 있고, 전화중에는 미지의 당사자들간에는 임의적으로 스위칭할 수 없는 CDMA와는 달리) 코드 분할 다중 거래(CDMT: Code Division Multiple Transaction)는 거래의 전체 기간동안에 대해서 거래의 각 마지막에서 대응 상대방들을 고정시키지 않는다는 점에 의해 구별된다. 사실상, 함께 작동하는 2개의 기술들인 CDMA 및 SVR은 링크할 수 있고, 링크들을 교환할 수 있으며, 대응 당사자 측면중 하나의 신규 링크들을, 그렇지 않으면 관련이 없고 원하는 경우에는 익명으로 남을

수 있는, 다수의 임의의 대응 당사자들에 추가할 수 있다.

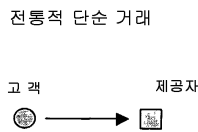
- [0130] 그러나, SVR 프로그램 모듈과 관련하여 작용하는 본 발명의 코드-분할 다중 거래(CDMT) 프로그램 모듈은 원격 통신 대역폭 제공을 고려하지 않으며, 오히려 대체-거래 상대들의 정합(matching)을 실행하고, 더욱이 거래의 전 주기동안 거래의 각 중단에서 대응 상대들을 고정시키지 않는다는 점에서 식별된다(도 10 참조).
- [0131] 사실상, 함께 작동하는 CDMX 및 SVR 프로그램 모듈들을 사용하는 프로그램은 링크할 수 있고, 링크들을 교환할 수 있으며, 대응 당사자들의 신규 링크들을 그렇지 않으면 관련이 없으며 원한다면 익명으로 남을 수 있는 다수의 임의의 대응 당사자들에 추가할 수 있다.
- [0132] 금융 거래는 제로-섬 구조(zero-sum framework)에서 실행되기 때문에, 모든 대응 당사자쌍(counterparty pair)은 가치의 균형으로 이루어져야 하며, 그렇지 않으면 거래는 유효하다고 생각되어 지지 않으며, 실제로 발생하지 않을 것이다. CDMX 프로그램 모듈을 통한 프로그램은 예를 들면, 익명의 구매자들 및 판매자들이 가격에 기초하여 단순히 자동적으로 정합되도록 할 수 있다(예를 들면, 동일한 가격에서 다수의 구매자들 및 판매자들이 있을 때, 제로-섬이 불균형적인 수의 상대방들로 확산될 수 있도록, 상기 시스템은 입력되는 거래 요구들을 자동적으로 코드분할할 것이다.)(도 10 참조). 따라서, (일본 엔의 구매자들이기도 한) US 달러의 5명의 판매자들은 단지 3명의 대응 당사자들에 의해 서빙된다. (도 9-10 참조).
- [0133] 참가(Sign-up)
- [0134] 우선, 참가자는 시스템 회원으로 참가해야 한다. 일반적으로, 본 발명의 시스템을 가진 계정을 생성하는데 참가하기 위해, 참가자는 표준적인 opt-in 절차를 통과하기 위해 단순히 전자 메일 계정을 가져야 하며, 예를 들면 웹사이트에 사용자 이름 및 패스워드를 개시하고, 그러한 웹사이트는 기본적인 사용자 정보를 가지고 계정 파일을 생성하며, opt-in 확인 요구를 제공된 전자 메일로 전송한다.
- [0135] 검증이 되면, 사용자 계정은 활성화될 것이고, 사용자는 분산된 클라이언트 애플리케이션을 다운로드하고, 다음에 분산된 자본 계정내로부터 사용자가 조작할 수 있기를 원하는 어떠한 제3 당사자 계정들로 새로운 분산 자본 계정 액세스 배제(new distributed capital account access clearance)를 주기 위하여 다양한 계정 및 금융 데이터의 입력에 착수하거나 또는 사용자는 같은 목적을 위해 중앙 서버(centralized server)로 동일한 종류의 정보를 공급할 수 있다.
- [0136] 그러나, 사용자는 참가(sign-up)함으로써 상기 시스템에 대한 완전한 액세스를 제공받을 수는 없으며, 오히려 초기 참가시에 사용자에게 이용가능한 (전통적인 단순한 거래 및 제한된 전통적 복잡한 거래를 실행하기 위한 능력과 같은) 최소한의 기능을 가지고, 스테이징(staging)을 통해서 진행해야 하지만, 실행된 거래의 가치 및 수, 그리고 자세한 투자자 상태의 증거에 따라 온라인 세미나 및 테스트를 통해 금융 법규들, 리스크 프로파일들, 투자 구조들, 세금 규정들, 등을 인식하고, 친숙하며, 고수하는 증명을 통해 이용 가능한 추가적으로 더욱 강력한 툴들(예를 들면, 협조적 거래)이 이용가능하다.
- [0137] 기업 애플리케이션들을 위해, 본 시스템은 다수의 액세스 패스워드들 및 사용자 배제 수준들(user-clearance levels)을 지원할 수 있는데, 예를 들면, CFO는 기업 시스템 계정에 대해 중요한 특권들(Master Priviledges) 및 완전한 기능(functionality)을 가지도록 하고, 따라서 기업의 자산들을 조작하며 관리하기 위한 최대한의 자유를 가지는 반면, 하위 계층의 금융 요원은 가시성 이유들(visibility reasons)에 대해 동일한 계정들에 대한 액세스를 할 수 있지만 조작하거나 관리할 중요한 권한들(Master privileges)을 가지지 않도록 할 수 있다.
- [0138] 이러한 접근 수단은 비디오 게임들에서 너무나 가시적인 챌린지 램프 설계의 매우 진보된 기술을 금융 서비스 제공으로 가져온다. 전통적인 사용자 통계 장애물들(user statistic hurdles)외에도 교육적인 또는 문제 해결 상호작용을 통합하는 챌린지-램프(challenge-ramp)를 통합함으로써, 금융 서비스들 주변의 대화형 환경(interactive environment)을 생성하는 것이 가능하다.
- [0139] 이하에서 상세히 기술될 신용 평가 시스템(credit rating system)은 그들의 여행중에 어떤 인공물들(artifacts) 또는 기술들을 축적하는 사용자들에게 (더욱 강력한 검이나 또는 마법 주문들과 같은) 강력한 파워 또는 힘이 생기는 롤-플레이 게임들(role-playing games)에서와 같은 상당히 동일한 방식으로 분류 시스템(classification system)을 구동할 수 있다. 따라서, 본 시스템은 신용 또는 무결성(Integrity) 외에도 금융 기술(skill), 자본(또는 자산)력 등에 따라 사용자들을 분류할 수 있다. 사용자들의 인구가 증가함에 따라, 참조를 위해 인구(population)에 이용 가능한 카테고리별 순위들이 존재할 수 있으며, 따라서 넓은 사회가 성숙함에 따라, 사용자들의 평판은 신용 점수 또는 Moody'sTM 평가 등의 중요성을 가지게 될 것이다, 그러나, 신용 평

가 관리력(the power of credit rating management)은 공평하게 사용자들 및 사회의 수준에 있으며, EquifaxTM 등과 같은 비밀주의의 제3자 조직의 수준에 있지는 않다.

- [0140] 초기 참가(signup)에 대한 다른 대안은 공식 기관이 사용자를 검증하는 것으로, 이것은 사회 보안 번호(social security number)와 같이 다른 개인 정보 및 공식적인 식별 정보(identify)를 필요로 할 것이다.
- [0141] 시스템 등록(System Registry)
- [0142] 일단 사용자가 본 발명의 금융 시스템의 일원이 되면, 사용자는 그가 만들고자하는, 테스트하고자 하는 및/또는 실행하고자 하는 거래들의 포함을 위해 이용 가능하기를 원하는 은행 계정들, 유틸리티 계정들, 신용 카드 계정들 등과 같은 그의 계정들을 등록할 수 있을 것이다.
- [0143] 금융 계정 데이터 및 액세스를 본 시스템으로 로딩하는 다양한 다른 방법들이 가능하다. 예를 들면, 개인 계정 총계 서비스(personal account aggregating service)는 개인용 총계 서비스 계정에서 사용자에게 대해 이용 가능하도록 하기 위해 각 금융 계정 웹사이트에 대해 사용자가 가지고 여행하는 로그인 및 패스워드 정보를 캡처할 수 있는 보조 수단(assistant)을 사용한다.
- [0144] 지금까지 모든 기존의 시스템들은 사용자에게 의해 관리되고 사용자에게 의해 액세스되는 계정들을 포함한다. 알려진 경우에서 이러한 시스템들의 사용자들은 다른 사용자들에게 액세스를 부여하거나 그러한 액세스 권한을 전송하지 못한다. 그러나, 본 발명은 의도적으로 거래 사건들에 대해 이러한 다수 액세스(multiparty access) 및 액세스의 부여를 가능하게 한다.
- [0145] 예를 들면, 협조적인 거래들(collaborative transactions)에서, 계정 액세스 또는 심지어 하나의 거래 액세스는 그러한 참여 권한(participation privilege)이 부여된 사용자들에 의해 수신되며, 그러한 사용자들에게 요구되는 특별한 행동 필요없는 사용자들에게 이용 가능하게 된다. 물론, 사용자는 부여된 권한을 거부할 권한을 가지지만, 액세스 권한을 얻는 것은 다른 시스템 사용자들의 선동으로 발생된 것이었다. 다양한 금융 서비스 제공자들은 기능을 증가시키고 사용자들의 기존의 계정들내의 그러한 기능에 대한 액세스를 부여할 것으로 기대되지만, 어디에서도 협조적인 참여자들(collaborating participants)로서 서로간에 대화하며, 서로간의 계정들 및 거래들에 허가들을 부여하는 시스템 사용자들에게 용이하도록 설계된 금융 서비스 플랫폼은 존재하지 않는다.
- [0146] 금융 거래 유형들 - 4개의 중요한 실시예들
- [0147] 본 발명에 따른 방법들 및 시스템들은 4개의 주된 유형들의 금융 거래들을 포함하는데, 전통적 단순, 전통적 복잡, 협조적 단순, 및 협조적 복잡으로, 이것들은 더욱 상세하게 기술되기 전에 이하에서 일반적으로 기술될 것이다.
- [0148] 전통적인 거래들(Traditional Transactions)
- [0149] 전통적 단순(Traditional Simple)
- [0150] 현재, 금융 서비스들은 금융 서비스 제공자 및 고객 또는 클라이언트간에 1대 1 관계들의 환경에서 지배적으로 추구된다. 1대 1 상업적 관계들, 및 그 환경들에서 발생하는 거래 사건들은 상당히 직접적이다. 즉, 대부분 하나의 판매자/제공자, 하나의 구매자/사용자 및 한가지 형태의 구입 통화 또는 신용거래로, 구매자가 판매자로부터 제품들을 구입하거나 또는 사용자는 신용카드 회사에 지불하는 것과 같이 모든 것들은 단순하고 선형적인 방식으로 배열된다.
- [0151] 이러한 것들을 지원하는 이러한 금융 대화들(interactions) 및 서비스들은 본 발명의 시스템이 전통적인 단순 거래 사건들로서 지칭하는 것들이다. 전통적 단순 거래들의 예시들은 (이하의 그림 1 참조) 중계자로부터 주식들의 구입 또는 웹사이트로 로그인하고 구입하거나 또는 신용 카드 요금을 지불하는 것이다. 비록 거래의 온라인 측면은 이론적으로 초기에 그리고 보다 빠르게 가능하게 했지만, 고객 및 제공자간의 필수적인 관계는 변화되지 않았다. 실제로, 한번에 하나의 회사의 금융 서비스들에 접속하고 있는 하나의 사용자들은 인터넷과 같은 전세계적으로 전산화된 네트워크의 실제 네트워크-장점을 사용하지 못하고 있다.
- [0152] 비록 본 발명은 다수의 접속된 시스템들, 사용자들, 제공자들, 고객들, 구매자들, 판매자들 등의 실제 네트워크-장점에 영향을 주는 네트워크 협조 거래 서비스들을 지원하도록 설계되었지만, 현재의 방법들 및 시스템들은 신용 카드 웹사이트 또는 은행 웹사이트 등과 같이 거래들의 참여자들중 단지 하나의 관점의 노드에 사용자를 위치시키는 반면에, 본 발명을 이용하는 사용자는 그/그녀의 거래 분야에서 모든 계정들의 포함을 보고 조작할

수 있는 전지의 관점(omniscient perspective)이 제공되기 때문에, 전통적 단순 거래들은 이러한 분포된 금융 시스템을 사용해서 손쉽게 실행될 수 있다.

[0153] 그림 1



[0154]

[0155] 전통적 복잡(Traditional Compound)

[0156] 전통적 복잡 거래에서(이하의 그림 2 참조), 사용자는 다수의 전통적 단순 거래들을 동시에 관리하고, 통합하고, 그리고 프로그램할 수 있다. 예를 들면, 사용자는 사용자의 가게 대출(Home Loan), 유틸리티 요금(Utility bills), 주식들(stocks), 증서들(bonds), 온라인 구입들, 및 사용자가 하고자 하는 거래를 포함하는 다른 것들과 함께 사용자의 신용 카드 요금들을 관리하고 통합할 수 있다. 전통적 복잡 거래는 새로운 홈-순가 대출(home equity loan)을 얻는 것을 포함하며, 상기 대출을 이용해서 신용 카드 및 오토-론(auto-loan)의 요금을 지불하거나, 또는 아이들의 대학 수업료를 지불한다. 현재, 이러한 종류의 결정들을 고찰하기 위한 단지 한 가지 방법이 있는데, 결정들은 한번에 하나씩 실행되어야 하며, 사용자는 다양한 소스들로부터 데이터를 수집해야 하며, 다음에는 계산하는 방법을 이해하기 위한 기술 및 배경을 가져야 하며, 다수의 1 대 1 거래들의 정렬된 조합을 통해 다양한 최적의 금융 상황들을 얻어야 하며, 이러한 복잡성은 대부분의 소비자들을 실망시킨다. 명료하게 하기 위해, 다양한 금융 서비스들의 사용자가 다양한 제3자들에 의한 많은 부채들을 통합하는 것과 같이, (또는 그/그녀가 있어야 하는지를 검사하기 위해) 조합된 또는 통합된 방식으로 다수의 이용 가능한 서비스들을 이용하고자 하는 경우에, 사용자가 사용자 자신의 특유한 상황에 대한 명확하고 직접적인 금융 실행 전략을 수집하는 것을 용이하게 할 수 있는 기존의 자동화된 지원 서비스가 없다. 금융 서비스들의 사용자들은 그들의 많은 약속들(commitments) 및 책임들(responsibilities)을 처리하기 위해 실행할 필요가 있는 전통적 복잡 거래들에 대한 지원을 얻을 수가 없기 때문에, 일련의 공동 작용이 가능한 전통적 단순 거래들(coordinated Traditional Simple transactions)을 정리하고 타이밍을 맞추므로써, 사용자들은 복잡한 거래들을 인위적으로 결합하도록 방치된다. 어떤 기존의 서비스들은 한 장소 및 인터페이스에서 다수의 계정들의 정보를 수집할 수 있지만, 어떠한 기존의 시스템도 사용자로 다수의 계정들의 상호작용을 관리하고, 통합하고, 프로그램하도록 하지는 않는다. 그러나, 본 발명은 이것을 가능하게 한다.

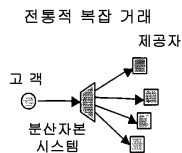
[0157] 영업 사용자들에 대해서, 사용자는 다양한 금융 활동들과 회사의 책임들간에 상호작용(interplay)을 평가할 수 있으며, 예를 들면 월급 지불, 벤더 지불을 하거나, 자금을 보충하거나(refinance), 대출을 얻을 수 있으며, 더 나은 의사 결정을 가능하게 하는 (법인의 재정 상태의 가시성(visibility)에 이르는) 신중하고 전례가 없는 데이터들 캡처할 수 있다. 따라서, 벤더 지불(vendor billing)은 자금 관리(treasury management) (예를 들면, 통화 관리, 신용 연계 매매(credit hedging)) 또는 수입 흐름(revenue stream) 등의 다음에 실행될 수 있으며, 모든 것들은 같은 인터페이스내에 있다. 결과들이 하나의 계정 혹은 관계가 아닌 전체에 영향을 미칠 때, 전통적 복잡 거래들은 사용자들로 하여금 그 결과들을 평가하기 위해 거래 테스트(test transaction) 외에도 그들의 다양한 금융 준비들(financial arrangements)간의 상호작용(interplay), 수취계정(receivables) 및 채무(liabilities)(자금-흐름 관리 등) 및 책임(obligations), 스케줄을 세우도록 한다.

[0158] 예를 들면, 더욱 커다란 기업들은 분리된 그룹들을 가지려는 경향이 있으며, 각각의 분리된 그룹은 인증하고, 수신하며 및 법인발행채권 구입들(corporate purchases)에 대해 지불한다. 이러한 분리(separation)는 승인에 이르기 위해 필요한 총체적 정보를 모으기 위해 이종의 그룹들(disparate groups)이 반복된 전달(communication)에 의존해야 할 때, 지불외에도 구입 모두에 있어서, 승인을 처리하는데 있어서 지연들에 이바지한다. 법인 금융 부서에 의해 처리되고 발행되는 송금 명령(remittance order) 및 최종 승인을 향한 절차를 통한 진행(progression)을 트리거 또는 진척시키기 위해 전자 메일을 사용 또는 로그인하는 각 부서 또는 그룹을 가지고 전통적 복잡 거래들은 이러한 송장 절차(invoicing process)를 합리화하기 위해 결합될 수 있다. 이제 단순 시스템들이 사용자에게 전송될 전자 메일 또는 다른 통지와 함께 생기는 몇몇 변수들에 대해 설정되는 경계들을 허가하는 동일한 방식으로, 이러한 경우에 송장(invoice)에 대한 책임을 지는 상급 당사자는 송장 처리(invoice handling)의 과정을 따라 다양한 트리거들(triggers)을 지휘할 수 있으며, 다양한 부서들(예컨대 법인 메일룸(mailroom))에서 이종 사용자들을 허가할 수 있으며, 반면에, 회계부서가 지불을 송금하기로 발표하

기 전에 다른 그룹 또는 부서(재산(goods)을 사용하는 특정 팀)는 정확성(accuracy)(예를 들면, 프로젝트에 대해 요구되는 올바른 부분)을 승인한다. 이러한 제어는 법인의 구조를 통해 구현될 수 있으며, 이러한 법인의 구조는 기업으로 하여금 회사의 재정 상태에 대해 더 많고 신속한 데이터를 가지도록 한다.

[0159] 더욱이, 사용자는 통합된 거래들의 매개변수들을 설정할 수 있다(예를 들면, 어떤 값에 이를 때 팔아야 하는 주식들, 어떤 시간 또는 날짜 등에서 팔아야 하는 주식들). 따라서, 거래들은 자동으로 그리고 원격으로 설정될 수 있으며, 사용자는 요소 거래 결과들을 기술하는 요약 정보 외에도 전반적인 복잡 거래 결과들을 액세스할 수 있을 뿐 아니라 거래가 발생했을 때 통보를 받을 수 있다.

[0160] 그림 2



[0161]

[0162] 협조적 거래(Collaborative Transactions)

[0163] 본 발명의 일 실시예는 금융 활동 분야에서 네트워크링된 또는 협조적인 옵션들의 존재로써 대강 정의될 수 있다. 이것은 기존의 금융, 은행업무 및 상업 활동과는 특이하게 다르다. (일반적으로 상업을 포함하는) 이러한 활동들에 대한 기존의 패러다임은 특이한 선형 고객-제공자 배치(singular, linear Customer-Provider arrangement)에서의 관계들을 지향하는 거의 배타적 구조이다. 그러나, 협조적 활동들은 상호간에 지향되어지고, 하나의 관계 이상을 포함하며, 여기서 모든 당사자들(parties)은 동시에 고객들 및 제공자들이 될 수 있다.

[0164] 본 발명에 따른 시스템들 및 방법들에 의해 지원되도록 설계된 협조적 거래들은 다수의 당사자들 및 다양한 형태의 통화 및/또는 신용을 포함한다. 협조적 거래는 과도적인 (또는 확산 벡터(spread-vector)) 실행들을 포함하며, 이것은 거래 방정식들(transaction equations)이 크레딧들 및 데비트들의 넷-제로(net-zeroing of credits and debits)와 같이 복잡할 수 있음을 의미하며, 수취계정(receivables) 및 채무(liabilities) 등은 전통적 거래에서처럼 직접 관련된 당사자들간에서만 이라기보다는 네트워크 분야의 모든 참여자들 가운데에서 글로벌 레벨에서 처리된다.

[0165] 이것은 다음과 같은 점을 의미하는데, 본 발명의 분산 자본 시스템은 어떠한 하나의 참가자보다 거래 분야를 더 잘 인식하고 있기 때문에, 수신자 및 채무자 당사자들이 송금 또는 지불들을 수용하기 위해 사용되고 있는 기관들(institutions) 및 계정들에 따라 정합될 수 있을 때, 분야에서의 활동 자금 움직임의 전체량이 분야에서의 지불된 동일한 채무량에 대해 감소될 수 있도록, 본 발명의 분산 자본 시스템은 능동적으로 그리고 지능적으로 거래의 세부계획들(logistics)을 최적화할 수 있는데, 예를 들면, 수천의 참여자들에 걸친 채무의 과도적 혼합(transitive shuffle of liability)을 이론적으로 연산할 수 있고, 따라서, 그 은행에 존재하고 ACH 또는 CHIPS 등과 같은 은행간 전달 네트워크로 나가는 넷 자금들, 또는 감소된 넷 자금들이 없이, 예를 들면 자금 이동이 (비록 이러한 개개의 참여자들은 전혀 상호간에 접촉이 없지만, 동일한 기관에서 적어도 부분적으로 정합되고 식별될 수 있는) 내부 은행 실행으로서 처리될 수 있도록 한다.

[0166] 협조적 거래들을 시도하고 가시화하기 위한 접근 가능한 방법은 주식 시장들의 특유한 실체(specific reality)를 검사하는 것일 것이다. 구매자들 및 판매자들은 거의 서로를 모르지만, 거래되고 있는 증서들(instruments)은 대체 가능하기 때문에, 일방 당사자가 판매한 증서는 인접성 또는 인식에 관계없이 어떤 다른 당사자의 구입한 증거일 수 있다. 그 소유의 증서가 당사자 A로부터 당사자 B로 변경되는 당사자들(participants)는 상대방이 누구인지에 개의치 않으며 고려할 필요가 없으며, 그들은 단지 그들의 거래가 국지적으로 넷-제로(net-zero)로 되는지를 고려할 필요가 있으며, 시장 구조 자체는 모든 거래들에 걸친 그리고 모든 참가자들간에 글로벌 레벨상의 넷 제로(net-zeroing)를 고려한다. 본 발명은 이러한 보증된 글로벌-제로 구조(global-zeroing mechanism)를 시장들로 기능하기 위해 (예를 들면, 유동성을 얻는) 임계량(critical mass) 또는 구조(예를 들면, 보안성(securitization) 등)를 가지지 않는 다양한 거래 상황들로 가져간다.

[0167] 현재, 글로벌-레벨의 제로(global-level zeroing)는 보안화되거나 또는 중요한 항목들(securitized or commoditized items)을 거래하는 이론적으로 많은 수의 참여자들로 능동적인 유동 시장들(liquid markets)에서만 명백하다. 본 발명은 위에서 기술된 SVR 프로그램 모듈로 불리는 프로그램을 가지고 얻어진 과도적인 실행들

을 사용해서 이러한 글로벌 레벨 제로 구조를 제한된 참여자 거래 환경들로 가져가며, 2개의 참여하는 실체들만큼 적은 것을 포함하는 어떤 수의 거래들은 이제 모든 참여하는 실체들에 걸쳐 넷 제로 책임을 지울(net-zero responsibilities) 시스템을 액세스할 수 있다. 2개의 실체 거래, 또는 2개의 실체들간의 다수의 거래들은 매우 쉽게 순익이 날 수 있으며, 따라서 당사자들간에 제로(zeroing)를 해결하기 위해 매우 복잡한 네트워크 기술이 필요하지 않을 수 있다; 그러나, 그럼에도 불구하고 본 발명은 이러한 2개의 실체 거래 환경을 관리할 수 있다. 그러나, 3 또는 이상의 당사자들이 존재할 때, 글로벌 넷-제로(net-zero)를 해결하기는 직관적으로 쉽지 않으며, 이것은 본 발명이 가장 유용하며 가장 가치있는 경우로, 위험시의 전 자본(total capital at risk), 사용된 전 경제 자본(total economic capital deployed), 거래의 민첩성(alacrity of transaction) 등의 견지에서 효율성을 가져온다.

[0168] 이러한 공개 시장에 참여함에 있어서, 모든 참여자들은 동시에 고객들(Customers)이며 제공자들(Providers)이다. 모든 참여자들이 고객이자 제공자 특징들을 입증하는 경우에, 거래 유형은 가장 복잡하며, 확장적인 거래 유형(예를 들면, 주식 시장은 협조적 복잡 거래의 단지 하나의 예시이다)인 협조적 복잡(Collaborative Compound)으로서 자격을 가진다.

[0169] 대체성(fungibility)는 과도적인 해결(transitive resolution)을 용이하게 한다. 예를 들면, 이것이 의미하는 것은 만약 당사자 A가 당사자 B에 어떤 양 X를 빚지고 있고, 당사자 B는 당사자 C에서 어느 양 X를 빚지고 있다면, 모든 3명의 당사자들의 거래의 역할들 및 책임들은 당사자 A가 당사자 C에게 송금하는 한번의 지불 X로 만족될 수 있다. 분산 자본 시스템은 당사자 A, B, 및 C가 서로 알고 있거나 또는 심지어 서로를 인식하고 있는지에 관계없이 이러한 과도적인 상황을 발견하고 실행할 것이며, 동일한 효과로 공공 실체들(public entities)의 구매자들 및 판매자들은 그들의 거래에 대한 상대방이 누구인지를 알지 못하며 고려하지 않는다. 협조적 거래에 있어서는 누가 현재의 구매자 또는 판매자인지는 중요하지 않으며, 오히려 쟁점은 채무(liabilities) 및 어음(receivables)이 이행되었는지, 그리고 올바른 당사자들이 그들이 특히 책임을 지고 있는 채무를 이행 완료함에 대해 면목을 세웠는지이며, 반면에 올바른 수신 당사자가 그들이 빚지고 있는 것을 실제로 받았는지를 확실히 하는 것이다.

[0170] 거래 분야에서는 임의의 참여자들은 위의 예시에서 가정한 바와 같이 정확히 같은 X의 채무에 도달하지는 않기 때문에, 분산 자본 시스템(distributed capital system)은 일치하지 않는 (예를 들면, 불균형의) 양을 다루기 위한 방법 및 시스템을 가지고 있다. 위에서 언급된 바와 같이, 스프레드-벡터 해상도(Spread-Vector Resolution) 프로그램 모듈은, 흐름 기반의 모델(flow-based model)로서 거래 분야를 다룸에 있어서, 끝없는 방식으로 그러한 당사자들의 부분적인 또는 전 채무(full liability) 또는 지불되어야 할 양들의 추종하는 대응 상대들 정합들(follow-on counterparty matches)을 연결함으로써 부정합되는 양들(non-matching amounts)의 "해결(resolution)"을 얻는다.

[0171] 협조적 단순(Collaborative Simple)

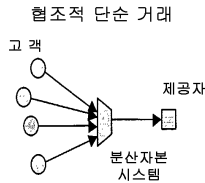
[0172] 협조적 단순 거래(그림 3 참조)에서는, 2 또는 이상의 당사자들은 같은 기회에 참여하기 위해 독립적으로 선택한다. 간단한 예시에서, 이것은 투자자들의 그룹의 선택에 이용 가능한 개인-투자 기회를 만드는 작은 기업가를 포함할 것이다. 현재 이와 같은 상황은 기업가가 (법인에서 투자된 자금들의 공동출자-자산 집합(pooled-asset aggregation)에도 불구하고) 다른 것들에 독립적인 각각의 것에 대한 당사자가 되는 1 대 1 관계들의 집합으로서 추구될 것이다.

[0173] 뮤추얼 펀드들(mutual funds)은 분산 자금 시스템이 어떻게 투자자들에 대한 고객-서비스 제공을 개선시킬 수 있는지의 좋은 예시이다. 그러나, 현재의 뮤추얼 펀드들은 전통적인 1 대 1 관계 패러다임 및 거래의 정적인 처리 모두에 따른다. 그러나, 예를 들면, 분산 자금 시스템은 예를 들면, 자금 상태에 대해 일정한 가시성(constant visibility)을 제공함으로써 고객, 및 그에 관련된 심지어 다른 투자자들에게 반환될 더욱 많은 서비스 및 가치를 허가하도록 할 것이다.

[0174] 예를 들면, 제3자에게 돈을 대여하고 있는 2명의 친구들이 이자와 함께 갚기로 제안한 제3의 친구에게 제공하는 경우로, 여기서 모든 대여하는 당사자들은 "투자 기회(investment opportunity)" 의 "고객들(customers)"이다. 다른 예시는 기부중인 회중(congregation)으로부터 돈을 거두는 교회로, 돈을 대출의 형태로 은행(들)으로부터 대출한 것이며, 여기서 대출, 기부, 이자와 함께 지불, 등은 본 발명의 분산 금융 시스템에 의해 관리될 수 있으며, 만약 교회가 다수의 은행들뿐만 아니라 회중의 각 회원들간에 모든 거래를 한다면, 좀더싼 가격으로 처리될 수 있다.

[0175] 많은 수의 투자자들이 예금 라인들(credit lines)의 신디케이트 조직(syndication) 등과 같이 예를 들면, 시니어 인스트루먼트(senior instruments) (debt)에 참여하도록 허가하기를 원하는 피고용인 소유의 법인들 및 공공 법인들은 협조적 단순 거래의 사용자들일 수 있다.

[0176] 그림 3



[0177]

[0178] 협조적 복잡(Collaborative Compound)

[0179] 본 발명의 방법들 및 시스템에 따른 다른 실시에는 협조적 복잡 거래(그림 4 참조)이다. 협조적 복잡 거래에서, 거래의 양 측면들은 그들 자신들을 고객들이라고 (그리고 그렇게 하는 것은 그들을 또한 제공자들로 만드는 상황에서) 생각하고 있다. 고객들은 상호간에 어떤 직접적인 인식을 가지고 있지 않으며, 본 발명의 분산 자본 시스템은 거래를 더욱 효율적이거나 또는 비용을 덜 들도록 또는 양자 모두가 되도록 하기 위해 단순히 협조적 복잡 실시예를 사용하고 있다.

[0180] 협조적 복잡 거래의 예시에서, 사용자의 회사는 어떤 활동들에 대해 영국 파운드들을 요구할 수 있다. 전통적인 방식으로, 사용자는 물리적으로 또는 온라인상에서 은행으로 가서, US\$를 GB£로 변환될 어떤 양의 자본을 준비한다.

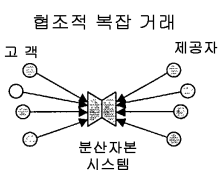
[0181] 제1 사용자와 관련이 없는 세상 어딘가에 있는 제2 사용자는 어떤 회사 활동들을 위해 U.S. 달러들을 필요로 하고, 전통적으로 상기 사용자는 유사한 방식으로, 물리적으로 또는 온라인 등을 통해 은행 또는 다른 금융 기관 등으로 가서, GB£를 US\$로 변환될 어떤 양의 자본을 준비한다.

[0182] 사용자들 모두는 고객들이고, 금융 기관들은 그들 모두에게 전통적인 방식으로 서비스들을 제공할 수 있다. 그러나, 사용자들 모두는 상호간에 거래하도록 준비할 수 있거나, 또는 자동으로 그리고 익명으로 본 발명의 분산 자본 시스템에서의 서로간의 요구들을 상호 만족시키기 위해 따라서 동시에 고객들이 될 수 있다.

[0183] 특히, 사용자들은 환전(exchange of currency)을 요구하기 위해 본 발명의 분산 자본 시스템을 사용할 수 있다. 본 발명에 따른 분산 자본 시스템의 프로그램은 사용자들/당사자들 모두에 의해 요구된 자금들의 정확한 정합의 문제를 해결하기 위해 벡터 흐름 알고리즘(vector-flow algorithm)을 사용함으로써, 당사자들이 가능한 전체 양으로 거래하도록 하고, (예를 들면, 만약 한 명의 사용자가 다른 당사자가 제공할 수 있는 것보다 더 많은 US\$를 요구했다면) 다음에는 어떤 남은 양을 취하도록 하고, 다른 "고객(customer)"이 그러한 부분적인 양을 이행하도록하고, 그러한 고객의 나머지가 그러한 거래의 반대편에 있는 다른 다음 고객에 채워지도록 한다.

[0184] 이것은 (예를 들면, 은행이 원하는 현금 변환의 정확한 양을 이행하는 예시에서와 같이) 모든 거래들이 하나의 대응하는 상대방(counterparty)에 의해 완전히 만족되어야 한다는 가정을 강요하는 두드러진 정적인 처리라기보다는 거래의 흐름 기반의 처리이다. 이러한 유형의 협조적 복잡 거래는 - 작건 또는 많건- 관계된 양에도 불구하고 발생할 수 있으며, 제3 당사자 중개인들(3rd party intermediaries)의 제거에 기인해서 수수료가 감소됨에 따라 유동성(liquidity)은 이론적으로 증가하고, 경제 영역들간에 더 적은 알력(friction)을 초래하기 때문에, 따라서 지구 주위의 지역적 및 국가적 경제들에 걸쳐 변동의 완화(moderation of volatility)에 기여한다.

[0185] 그림 4



[0186]

[0187] 거래 타이밍(Timing of Transactions)

[0188] 이러한 거래의 4가지 실시예들의 유형을 논의함에 있어서, 사용자는 그들이 (요금 지불, 또는 투자-등급 요구에 대해 전통적으로 원하는) 즉시 처리(instantaneous processing)를 원하는지 또는 (예를 들면, 추후에 분납 요금 지불을 위해) 거래가 일어날 시간을 설정하고자 하는지, 또는 (어떤 주식-가격, 또는 사용자들이 사용하고자 하는 데이터에 대한 임의의 변수들, 예를 들면 라스 베가스의 대기 온도, 또는 그린 베이 파커 게임의 점수 등과 같은) 발생할 사건-트리거 거래(event-triggered transaction)를 원하는지를 나타낼 수 있다. 제1 거래는 시스템에 의해 실시간 또는 즉시 유형의 요구로 지정되며, 다음은 수동 요구로, 그리고 최종은 고정된 요구로 지정된다. 후자의 시스템에 있어서, 거래의 자금들은 본 시스템의 투자 엔진(investment engine)에 대해 이용 가능할 수 있다(이하의 투자 엔진 참조).

[0189] 전통적 단순 거래(Traditional Simple Transaction)

[0190] 본 발명의 방법들 및 시스템들에 따른 일 실시예는 전통적 단순 금융 거래로, 이제 더욱 상세하게 기술될 것이다. 비록 이하에서는 단지 전통적 단순 거래만이 기술될 것이지만, 다음의 다른 유형들의 거래들은 현재 시스템의 사용자 인터페이스를 사용할 수 있다는 점에 주의해야 한다. 따라서, 전통적 단순 거래와 관련하여 기술된 사용자 인터페이스는 다른 거래들과 관련하여서 상세하게 기술되지는 않을 것이다.

[0191] 위에서 언급된 바와 같이, 전통적 단순 거래는 대부분의 온라인 사용자들에게 익숙하며, 고객들 및 제공자들, 그리고 요금 지불과 같은 단순한 금융 거래만을 포함한다.

[0192] 따라서, 일단 사용자가 시스템에 로그인하고, 단계 S100에서의 로그인 절차(도 11a 참조), 및 단계 S101에서 인증 절차를 거쳤다면, 단계 S102에서 프로그램은 디스플레이 스크린상에 표시된 거래 옵션들과 함께 워드 프로세싱 프로그램이 열릴 때 열리는 스크린과 유사하게 공란의 디스플레이 스크린을 연다.

[0193] 다음에, 사용자는 그의 마우스를 사용해서 커서는 "파일(File)"에 위치시키고, 마우스를 클릭하며, 단계 S103에서 프로그램은 사용자에게 이용 가능한 거래들을 열거하는 풀다운 메뉴를 제공하거나, 또는 다이얼로그 윈도우, 옵션 팔레트, 또는 톨바 또는 사용자가 그/그녀의 계정에 설정했던 고객 설계 거래들의 어떤 유사한 표현(예를 들면, 기존 거래 1, 기존 거래 2, 새로운 거래 설정 등)을 제공한다.

[0194] 비록 활동들은 적절한 선택들 위에서 커서를 사용하는 마우스의 실행을 가지고 수행되는 것으로 기술될 것이지만, 당해 분야의 숙련자중 하나라면 이러한 실행들은 키보드상의 또는 음성 액추에이션 또는 다른 유사한 방법들에 의한 명령 기능에 의해 실행될 수 있으며, 인터페이스 설계 영역의 "통상의 지식을 가진 자"의 개선은 동일한 그리고 증가된 정보량들을 제공하는 더 나은 방법들을 구현할 것이라는 점에 주목해야 한다.

[0195] 더욱이, 비록 다양한 동작들 및 거래들이 이산 단계들 및 특유의 전문 용어들을 사용해서 기술될 것이지만, 당해 분야의 숙련자라면 거래는 다양한 방식들로 그리고 다른 전문 용어 또는 단계들을 사용해서 그리고 이하에서 기술될 방법들의 변형들을 사용해서 얻어질 수 있다는 점을 이해할 수 있을 것이다.

[0196] 이것이 신규의 또는 처음 거래라고 가정하면, 일단 "파일(File)" 메뉴에서, 사용자는 "신규 거래(New Transaction)"를 선택할 수 있고, 만약 거래가 그러한 옵션상에서 왼쪽을 클릭함으로써 단계 S104에서 신규한 것으로 결정된다면, 단계 S105에서 그러한 옵션상에서 왼쪽 클릭을 함으로써, 상기 프로그램은 디스플레이 스크린상에 공백 거래 워크패드(workpad)를 열 것이다(도 12 참조).

[0197] 일단 사용자가 기본 공백의 거래 워크패드 디스플레이에 위치하면, 사용자는 그의 커서를 제목 "보기(View)"로 이동시켜 그러한 옵션에서 왼쪽 클릭을 할 수 있으며, 여기서 상기 프로그램은 단계 S106에서 "툴바(Toolbars)"를 포함하는 풀다운 메뉴를 제공할 것이다.

[0198] 다음에 사용자는 "툴바(Toolbars)"에서 왼쪽 클릭을 하거나 또는 "툴바(Toolbars)"의 오른쪽으로 가서 하위메뉴가 펼쳐지도록 할 수 있으며, 단계 S107에서 상기 프로그램은 사용자가 선택할 수 있는 거래 옵션들 목록을 제공할 것이다.

[0199] "툴바(Toolbars)" 밑에는, 예를 들면 "내 은행(My Banks)", "내 신용 카드(My Credit Cards)", "내 통신(My Communications)", "내 유틸리티(My Utilities)", "내 보험(My Insurance)", "내 대출(My Loan)" 등외에도 "거래 객체(Transaction Objects)"라고 명명된 선택을 포함하는 옵션들의 목록이 있을 수 있다(도 13 참조).

[0200] 당해 분야의 숙련자라면 설계 인자들은 "툴바(Toolbars)"가 "거래 실체들(Transaction Entities)"과 같은 더욱 적절한 명칭 또는 사용자들과 가장 효율적으로 소통되는 어떤 다른 명칭으로 대체될 수 있다는 점을 이해할

것이다.

- [0201] 전통적 단순 거래의 예시에서, 사용자는 가스 요금을 지불하고자 할 수 있으며, 이것은 2개의 당사자들(3개의 실체들)을 포함하는 단순 거래로 - 사용자 및 가스 회사가 당사자들이 된다(사용자, 가스 회사, 및 3개의 실체들인 그들간의 거래 실행).
- [0202] 따라서, 사용자는 "거래 객체(Transaction Objects)"를 선택하고, 상기 옵션에서 왼쪽을 클릭할 수 있으며, 단계 S107에서 상기 프로그램은 상기 명령을 수신하고 "거래 객체(Transaction Objects)"로 명명된 거래 워크패드에 인접한 작업 영역을 디스플레이할 것이며(도 14 참조), 이 안에는 다양한 금융 거래 실행들을 나타내는 다수의 아이콘들이 배치된다.
- [0203] 예를 들면, 아이콘들은 자본 자산 벡터(Capital Asset Vector)(지불, 더욱 상세하게는 일반적인 자본의 이동 = "자본 이동(Move Funds)"), 환전(Convert Currency), 구매, 판매, 대여(lend), 차용(Borrow), 및 통화 헤지(Currency Hedge), 등과 같은 금융 거래 실행들(financial transaction actions)을 나타낼 수 있다. 이러한 거래 객체 툴바 또는 팔레트에 있는 동안, 상기 객체들은 활발하지 못할 것이다(inert). 그들중 어느 것이 거래 워크패드상으로 드래드될 때, 그들은 활성화 객체, 사용자가 결합할 거래의 빌딩 블록 실체들(building block entities)이 될 것이다.
- [0204] 일단 활성화되면(예를 들면, 거래 패드위에 놓여지면), 마우스오버(mouse-over), 마우스 버튼 클릭(우측 또는 좌측) 또는 다른 인터페이스-설계 지시의 사용자 친화적 액세스중 어느 하나를 통해, 사용자는 그러한 실체에 적합한 "속성들(Properties)" 정보를 볼 수 있다. "객체들(objects)"은 실체들(entities)의 서브세트이라는 점에 주목하라. 따라서, "거래 객체(transaction object)"는 거래 실행들에 특유한 용어이며, 여기서 "실체(entity)"는 그들간의 실행 외에도 거래에 대한 당사자들 모두를 식별하는데 사용된다.
- [0205] "신규 거래(New Transaction)"의 경우에, "객체 실체들(object entities)"은 일반적인 기능-가능성(functional-feasibility), 시스템 상태 유형 정보 외에는 "속성들" 정보를 디스플레이하지는 않을 것인데, 왜냐하면 거래가 신규이고 이러한 특유 거래내의 실행이 아직 사용되지 않았으며, 디스플레이하기 위한 기록을 생성할 사용 히스토리를 가지고 있지 않기 때문이다. 그러나, 이러한 "실체들"은 결합되고 있는 현재의 신규 거래 위의 다양한 거래들에 관련되어 있기 때문에 "신규 거래"를 결합하기 위한 "거래 워크패드"상에 놓여진 활성화된 "당사자 실체(party entities)"는 풍부한 "속성들" 정보를 정확히 나타낼 것이다. 사용자는 각 사용자에게 가장 중요한 것에 따라 이러한 속성들 정보의 표현을 구성할 수 있다.
- [0206] "거래 객체(Transaction Objects)" 툴바내에는, "신규 객체 검색(Search for New Objects)" 버튼이 있을 것이며, 이것은 사용자가 "거래 객체(Transaction Objects)" 툴바로의 최근의 추가사항들에 대해 시스템에 주기적으로 조회할 수 있도록 할 것이며, 따라서 사용자-생성 및 사용자-관리 거래들에 기능을 추가할 것이다. 그러나, 상기 프로그램은 상기 시스템에 의해 제공된 주기적인 갱신들과 관련되거나 또는 (온라인 투자자 교육 세미나 또는 트레이닝 프로그램 또는 제품 및 서비스의 설정 주기 혹은 자격을 얻은 또는 자격이 주어진 용법과 같은) 얻어진 전체 인자들과 관련하여 "거래 객체(Transaction Objects)"에서 자동적으로 나타날 신규의 "거래 객체(Transaction Objects)"를 제공할 것이다. 이러한 신규의 "거래 객체(Transaction Objects)"중 일부는 시스템으로부터 수신하기 위해서는 승인 또는 (세밀한 투자자 상태 등과 같은) 특별한 액세스를 필요로 할 수 있다.
- [0207] 다음에, 사용자는 "툴바(Toolbars)" 메뉴로부터 "내 은행들(My Banks)"을 선택할 수 있고, 상기 프로그램은 명령을 수신하고, 사용자에 의해 미리 등록된 다른 은행들 계정들을 지칭하는 아이콘들을 도시하는 거래 워크패드에 인접한 다른 라벨링된 섹션을 디스플레이할 것이다.
- [0208] 비록 본 발명에서, 계정들 및 거래 실행들은 바람직하게는 디스플레이상에서 아이콘들로 도시되지만, 상기 계정들 및 거래 실행들은 풀다운 메뉴, 테이블, 그리드 또는 유사한 것에서 이용 가능할 수 있다는 점에 주목해야 한다. 거래들은 거래 객체들 아이콘들이 움직임 또는 (텍스트 이름, 또는 유사한 것과 같은) 마우스를 이용한 플래이스홀더 표현(placeholder representations)에 의해 결합될 수 있거나 또는 실체들을 처리하기에 특유한 음성 액추에이션(voice actuation)에 의해 결합될 수 있다. 비록 다음의 기술은 마우스를 이용하여 조작되는 아이콘들의 사용에 관련될 것이지만, 다해 분야의 통상의 지식을 가진 자라면 키보드, 또는 디스플레이, 또는 음성 액추에이션 등의 명령 기능들의 사용도 명령들을 실행하거나 또는 거래들을 개시하도록 실시될 수 있다는 점을 인식할 것이다.
- [0209] 마지막으로, 사용자는 "툴바(Toolbars)" 메뉴로부터 "내 유틸리티들(My Utilities)"을 선택할 수 있고, 이에 의해 상기 프로그램은 "내 은행들(My Banks)" 및 "거래 객체들(transactions Objects)" 영역들에 인접한 라벨링

된 섹션을 디스플레이할 것이며, 이에 의해 "가스 회사", "식수 회사", "전자 회사" 등을 묘사하는 다수의 아이콘들이 "내 유틸리티들(My Utilities)" 섹션에 디스플레이될 것이다.

- [0210] 금융 거래를 실행하기 위해서, 거래 당사자들 및 그들간에 실행하도록 선택된 거래 실행은 거래로서 이행되도록 원하는 관계에서 배열된 거래 워크패드 영역내에 위치되어야 한다.
- [0211] 따라서, 사용자는 커서를 (사용자에 의해 등록된 특정 은행에서 특정 은행 계정에 특유할 수 있는) "내 은행 계정(My Bank Account)" 아이콘 위에 위치시킬 수 있고, 마우스를 사용해서 왼쪽을 클릭하고, 홀드시키고, 이동시키거나 또는 아이콘 "내 은행 계정(My Bank Account)"을 공백의 거래 워크패드 섹션상에 끌어서 아이콘을 그곳에 둘 수 있다. 원래의 위치로부터 끄는 것은 원래의 위치로부터 그것을 제거하는 것은 아니며, 커서가 원래의 위치로부터 떨어진 클릭된 객체를 이동시킴으로써 오히려 복사본을 생성하는 것이다. 따라서, 단계 S108에서 사용자가 거래를 결합함에 따라, 상기 프로그램은 거래 워크패드 섹션내에 거래 실체를 받아들인다.
- [0212] 사용자가 마우스오버(mouseover) 또는 거래 워크패드 섹션내에서 "내 은행 계정(My Bank Account)" 아이콘에 관련된 "속성들" 정보를 요구하는 행동을 실행할 때, 단계 S109에서, 상기 프로그램은 VfT 프로그램 모듈을 사용해서 사용자의 은행 계정의 상태(예를 들면, 계정 번호, 수표, 예금에서의 잔고 등)에 대한 실시간 정보를 얻기 위한 명령으로서 이러한 행동을 받아들인다. VfT 프로그램 모듈은 이러한 명령을 실행하고, 어느 사용자가 이러한 요구를 전송하고 있는지 그리고 어느 실체를 위한 요구인지를 CDMX 프로그램 모듈에 알리고, 상기 CDMX 프로그램 모듈은 모든 사용자-등록 계정들의 기록들을 보유하는 "계정 등록(Account Registry)"을 액세스하고, 이러한 계정들에 대해 액세스-인증 데이터(access-authorizing data) 외에도 적절하고 공식적인 식별(proper and formal identifying)을 수집한다.
- [0213] 대안적으로, 상기 시스템은 필요시에 사용자 패스워드들을 요구하고, 어떠한 사용자 패스워드들도 저장하지 않도록 구성될 수 있다. 저장되고 있는 것 외에도, 상기 시스템 외부의 실제 세상의 실체로부터의 정보를 검색하기 위해 요구를 실행하는 MaQs 프로그램 모듈에 대한 이러한 검색된 정보를 처리하기 전에 이러한 액세스를 진행하는 것은 타임스탬핑(timestamped)되며, CDMX 프로그램 모듈에 의해 거래 코드가 할당된다.
- [0214] 따라서, 단계 S109에서, 상기 프로그램은 정보를 얻기 위해 필요한 어떠한 보안 금융 네트워크를 통해 그리고 인터넷을 통해 사용자의 은행에 조회한다. 이러한 요구는 클라이언트로부터 다른 클라이언트 또는 서버로 전송되고, 상기 요구가 전송되는 상기 클라이언트 또는 서버는 상기 은행에 대한 조회를 실행한다는 점에 주목하자. (그러나, 상기 요구가 쉽게 처리되도록, 상기 조회되는 은행은 본 시스템의 일부일 수 있다.
- [0215] 또한 하나의 복잡한 거래 옵션에서, 사용자는 옵션으로 "갱신을 위해 모든 계정들을 검색(Scan All Accounts for Updates)"을 선택할 수 있으며, 상기 프로그램은 사용자의 모든 미리 등록된 계정들을 검색하고 각 계정과 관련된 실시간 데이터를 사용자에게 제공하며, 사용자에게 신규의 요금들, 뮤추얼 펀드에 대한 잔고의 변화들 또는 은행 계정들 등에 대해 조언한다.).
- [0216] 단계 S110에서, MaQs 프로그램 모듈은 서버에 있는 은행으로부터 정보를 수신하고, 이 지점에서 상기 프로그램은 또한 상기 네트워크를 통해 요구를 발송한 결과로서 제공될 수 있는 네트워크 정보 또는 일반적인 시스템 상태를 포함하는 검색된 정보를 다시 CDMX 프로그램 모듈로 전달한다. CDMX 프로그램 모듈을 통한 상기 프로그램은 모든 수신된 정보를 타임스탬핑하고, 코딩하고, 그리고 저장한다. 다음에, 상기 CDMX 프로그램 모듈은 단지 요구된 정보만을 다시 VfT 프로그램 모듈로 전달하기 위해 수신한 정보를 필터링하고, 따라서 사용자 은행 계정의 실시간 상태는 단계 S111에서 사용자에게 대해 그것을 디스플레이하는 클라이언트 컴퓨터, 풀다운 메뉴, 테이블, 다른 윈도우 또는 유사한 것 하에서 이용 가능하거나 또는 아이콘 옆의 스크린상에 제공된다.
- [0217] 사용자는 그의 은행 계정을 시스템에 미리 등록시켰기 때문에, 이러한 조회는 안전하게 실행될 수 있으며, 사용자가 예를 들면 ATM 기계로부터 현금을 인출하고자 할 때 사용자가 인출하는 것과 같이 결과들은 거의 동시에 검색된다는 점에 주목해야 한다. 더욱이, 기존의 보안 네트워크들이 사용되기 때문에, 이러한 조회 및 정보를 수신하기 위해 신규의 물리적 기반 구조가 필요하지 않다.
- [0218] 이제 사용자는 "가스 회사(Gas Company)" 아이콘에 관하여 동일한 절차들을 실행하여, "가스 회사(Gas Company)" 아이콘을 잡고 거래 워크패드로 끌어놓을 수 있으며, "가스 회사(Gas Company)" 아이콘에서 마우스오버 또는 유사한 동작을 실행함으로써, 상기 프로그램은 "가스 회사(Gas Company)"에 관한 실시간 및 다른 정보를 얻을 수 있다(예를 들면, 계정 이름 및 번호, 고객 서비스 전화, 계정 기한, 과거의 지불된 양 등).
- [0219] 또한, 사용자는 "거래 객체들(Transaction Objects)" 섹션으로부터 거래 객체를 선택하기 위해 동일한 절차들을 실행할 수 있다. 이러한 경우에, "자본 자산 벡터(Capital Asset Vector)"는 "내 은행 계정(My Bank Account)"

및 "가스 회사(Gas Company)"간에 위치시키고 연결하기 위한 적절한 금융 거래 실행일 수 있다. 따라서, "자본 자산 벡터(Capital Asset Vector)"가 "거래 객체들(Transaction Objects)" 섹션으로부터 검색 완료되고, 거래 워크패드 섹션에서 상기 프로그램에 의해 수신 완료되면, 단계 S112에서(도면들 11b 및 14 참조), 사용자는 이제 게시해야 하며, 사용자 자신은 그것이 실제로 실행되기를 원한다면 금융 거래의 게시를 확인할 수 있다.

[0220] 따라서, 사용자는 거래를 상기 프로그램이 단계 S113(도 15 참조)에서 수신할 수 있는 완전히 결합된 거래(예를 들면, 사용자의 은행 계정으로부터 가스 요금의 지불)로 만들기 위해 2개의 실체들(예를 들면, "내 은행 계정(My Bank Account)" 및 "가스 회사(Gas Company)"간에 "자본 자산 벡터(Capital Asset Vector)"를 첨부할 수 있다.

[0221] "자본 자산 벡터(Capital Asset Vector)" 거래 객체가 "내 은행 계정(My Bank Account)" 실체 및 "가스 회사(Gas Company)" 실체 아이콘 모두에 첨부될 때까지는, 아무런 거래가 발생하지 않으며, 실행되지도 않고, 마우스오버(mouse over) 혹은 거래 워크패드상의 어떠한 실체에 대한 다른 "속성들(Properties)" 체크는 이러한 거래에 대한 확인 또는 결과들의 증거 또는 디스플레이를 생성하지 않을 것이라는 점에 주목해야 한다. 단계 S114에서, 사용자로 하여금 실행을 게시(도 16 참조)하도록 허가하기 전에, 상기 프로그램은 적어도 결합된 거래의 선택스(syntax)가 수행가능한 지를 확실히 하기 위해 모든 거래의 집합을 검사할 것이다(비록 상기 프로그램은 양들에서 그리고 선택된 실체들 등에서 모든 사용자 오류들을 검사할 수 없고, 피할 수 없을지라도). 유사하게, 상기 프로그램은 결합된 거래가 사용자 게시를 통해 이용 가능하게 되기 전에, (지불이 송금 요구(예를 들면 사용자로부터 가스 회사로의 반환 요구 또는 요금 제시) 보다는 가스 회사로 송금되도록) 자본 벡터의 양 외에도 "자본 자산 벡터(Capital Asset Vector)"의 방향 모두를 설정하고 확인하도록 사용자에게 요구할 것이다.

[0222] 거래를 결합하기 위해, 다음에 사용자는 "자본 자산 벡터(Capital Asset Vector)" 아이콘의 끝부분에서 왼쪽을 클릭할 수 있고, 상기 아이콘을 잡고 "내 은행 계정(My Bank Account)" 아이콘에 끌어 놓을 수 있으며, 여기서 그것은 "스냅(snap)"과 함께 부착될 것이다. 부가하여, 상기 프로그램은 상기 부착이 성공적이라는 것을 도시하기 위해 색을 변경하거나 또는 두껍게 하기 위하여 "내 은행 계정(My Bank Account)" 아이콘의 경계를 변경할 수 있다. 상기 "스냅(snap)" 특징은 "그리기(Draw)" 애플리케이션을 사용할 수 있고, 화살표들을 다양한 객체들에 부착시킬 수 있는 사용자에게는 익숙할 것이다.

[0223] 다음에, 사용자는 "자본 자산 벡터(Capital Asset Vector)"의 헤드에서 왼쪽 클릭할 수 있고, 그것을 잡고 "가스 회사(Gas Company)" 아이콘으로 끌어다 놓을 수 있으며, 여기서 그것은 또한 "스냅(snap)"과 함께 부착될 것이며, 성공적인 연결을 전달하기 위해 어떠한 사용자 중심의 설계가 선택되든 두꺼운 경계 또는 유사한 것으로 되는 프로그램을 초래한다.

[0224] 비록 위의 단계들은 다른 실체에 지불하는 하나의 실체로부터의 자금들의 이동 위치의 순서에 따라 기술되었지만, 당해 분야의 숙련자라면 이러한 단계들은 어떠한 순서로도 실행될 수 있으며, 상기 실체들은 각 위치 이후에 즉시 또는 모든 실체들이 거래 워크패드 등에 위치한 후에 거래 워크패드상에 어떠한 순서로 위치될 수 있다는 점을 인식할 것이라는 점에 주목해야 한다.

[0225] 일단 3개의 실체들(즉, "내 은행 계정(My Bank Account)", "자본 자산 벡터(Capital Asset Vector)", 및 "가스 회사(Gas Company)")간에 접속이 생성되면, 거래는 완전히 결합되고, 능동적인 거래가 되고, 시스템의 거래 선택스 실행 가능성 검사(syntax feasibility check)를 통과하고, 사용자 설정 및 거래 객체 임계(transaction object Criticals)의 확인을 기다린다.

[0226] 일단 완전히 결합되면, 사용자에게 의한 자본 자산 벡터(Capital Asset Vector)의, 사용자 중심의 설계가 알려줄 수 있을 때 이용 가능한, 다른 유사한 "속성들(properties)" 요구 행동 또는 마우스 오버는 그러한 자본 자산 벡터의 특징들을 상세하게 하는 정보를 디스플레이하기 위한 프로그램을 게시할 것이다. 만약 실행이 현재 시간에 진행된다면, 이것은 예를 들면, 벡터 소스 실체(Vector Source Entity)(예를 들면, 기관 명칭, 자금들이 인출될 계정 번호 등을 포함하는 사용자의 은행); 벡터 목적지 실체(Vector Destination Entity)(예를 들면, 기관 명칭, 타겟 계정 번호 등을 포함하는 사용자의 가스 회사); 실행 순간에 사용될 거래 시스템 또는 네트워크(ACH, PayPalTM, e-Checking 등), 네트워크 또는 시스템 상태 등을 포함한다.

[0227] 또한, 과거에 사용된 기존의 결합된 거래들의 경우에서 (예를 들면, 지난달의 가스 요금을 지불하는 것은, 변경시킬 타이밍 및 양과 함께, 이번 달의 요금 거래 및 벡터와 동일한 거래 실체들 및 집합 오리엔테이션(assembly orientation)과 같을 것이다.), 상기 프로그램은 그러한 특유 자본 자산 벡터를 사용해서 이전의 거래 활동의 표시를 제공할 것이며(예를 들면, 특정한 사용자 은행 계정으로부터 가스 회사로의 지불), 또한 사용자 중심 설

계 지시들(user-centered design dictates)이 거래 히스토리에 따라 프로그램에 의해 참조된 최적의 디스플레이 방법일 수 있을 때 상세한 점들이 이용될 수 있다.

- [0228] 만약 다른 시점들에서 다수의 다른 송금 실체들이 가스 회사에 지불하도록 사용된다면, 다수의 결합된 거래들이 존재할 것이고, 각각의 거래는 자신의 고유하게 정의된 자본 자산 벡터를 가지며, 각각의 거래는 사용자가 다시 사용하기 위해 언제라도 열어볼 수 있는 별개의 파일로서 저장된다는 점에 주목해야 한다(예를 들면, 만약 사용자가 다수의 은행 계정들 및 신용 카드 계정들 등을 가지고 있고, 한가지 경우에서 계정 1로부터 그리고 다른 경우에서 신용 카드 계정 등으로부터 송금하기를 원한다면, 사용자는 원하는 송금 방법에 따라 저장된 거래 파일을 열도록 선택할 것이다). 게다가, 만약 사용자가 한 위치에서 특정 가스 회사 계정을 포함하는 모든 활동의 상세들을 보고자 한다면, 어느 송금 계정이 지불을 전송하였는지에 불구하고, 상기 프로그램이 가스 회사 계정에 대한 활동에 따라 그것들 모두에서 가스 회사 기록을 갱신함에 따라, 그/그녀는 단지 "가스 회사로 지불(payment to Gas Co)" 거래 파일중 어느 하나의 "속성들(Properties)"에서 상기 가스 회사 계정 히스토리를 액세스할 수 있다.
- [0229] 만약 사용자가 이러한 "가스 회사로 지불(payment to Gas Co)" 거래 파일들중 하나에서 특정 자본 자산 벡터(Capital Asset Vector)"에 대한 "거래 히스토리(Transaction History)"를 보고자 한다면, 사용자는 이러한 옵션을 선택하고, 상기 프로그램은 거래량, 날짜 등을 포함하여 이러한 2개의 특정 실체들(송금 실체, 및 수신 실체)간에 이루어진 이전의 거래들을 도시하는 디스플레이를 제공하기 위해 데이터베이스를 액세스할 것이다.
- [0230] 거래 객체 임계(Transaction Object Criticals)(예를 들면, 상기 가스 회사 예시에서 벡터 임계)가 설정되고, 사용자에게 의해 확인될 때까지는 결합된 거래는 "실행을 위해 이용가능한" 상태로의 변화를 진행할 수 없다. 거래 객체 임계를 설정하고 확인한 후에 (자본 자산 벡터의 경우에 이것은 벡터 임계로 지칭될 것이다), 결합된 거래는 사용자의 재량에 따라 개시를 위해 이용 가능한 상태로 변화할 것이다. 상기 프로그램은 음성, 시각, 애니메이션, 또는 밝기, 컬러, 차원(dimensionality), 위치, 또는 유사한 것과 같은 상태에서의 변화들을 포함하는 디스플레이 또는 표현에서 변경들에 의해 이러한 변화를 도시할 수 있다.
- [0231] 사용자는 자본 자산 벡터상에서 더블 클릭, 또는 오른쪽 클릭을 할 수 있으며, 이에 의해 단계 S115에서, 상기 프로그램은 디스플레이 스크린에 다양한 옵션들을 포함하고, 그중 하나의 옵션은 "벡터 임계(Vector Criticals)"인 메뉴를 열 것이다.
- [0232] 자본 자산 거래 객체들에 대해서, 2개의 임계의 최소가 사용자에게 의해 설정되고 확인되어야 한다. 다른 임계는 디폴트 혹은 설정되지 않은 채로 될 수 있다. 자본 자산 벡터들에 대한 2개의 필수적인 임계는 : 1) 방향(Direction)으로, 이것은 벡터 소스 할당(=송금 실체의 결정), 및 벡터 결정 할당(=수신 실체의 결정)을 의미하며, 및 2) 양(Amount)으로, 이것은 송금되어야 하는 사용자가 원하는 양이다. 자본 자산 벡터가 거래 집합 단계에서의 실체들에 스냅-부착(snap-attached)될 때, 상기 프로그램은 벡터 소스 또는 벡터 목적지 선택중 어느 하나로서의 선택을 위해 이러한 실체들이 임계 인터페이스에서 이용가능하도록 할 것이다. 만약 사용자가 거래를 위한 벡터 타이밍을 설정하도록 선택하지 않는다면, 기술적 실행 가능성이 허용될 때, 상기 프로그램은 이러한 설정을 디폴트로 하여 "거래 실행(Execute this transaction)"을 누르자마자 거래의 실행을 즉시 개시하도록 할 것이다.
- [0233] 다른 임계는 (예를 들면, 타겟 실체에 의해 수용된 가능한 통화들 중에서) 통화 유형, 지불 날짜, 및 (예를 들면, 지불수표(paycheck)의 직접 예금, 통화-교환 타겟, 또는 유사한 것의 확인과 같은) 다른 외부 사건들에 의해 트리거되는 다른 날짜에서 다수의 벡터들 또는 지불들을 자동으로 송금하기 위한 옵션 등과 같은 것들을 포함할 수 있다. 사용자는 벡터 특유의 패스워드를 설정할 수 있으며, 또는 대안적으로는 이러한 벡터의 임계에 대한 액세스를 보호하기 위해 벡터를 설정해서 사용자의 시스템 로그인 패스워드를 사용하도록 할 수 있다; 또한, 객체 임계를 보거나 또는 변경하기 위해서 모든 거래 객체들이 패스워드를 요구하도록, 이러한 설정은 상기 프로그램에 의해 전체적으로 제어될 수 있다.
- [0234] 이러한 거래를 실행하는데 있어 더욱 정교한 옵션들(sophisticated options)을 위해서, 액세스될 각 거래 실체는 시스템의 프로그램이 파일상에 이러한 개인 정보를 보유하게 하기보다는, 시행 순간에 패스워드를 입력하도록 사용자에게 요구하도록 사용자는 이러한 벡터 또는 모든 거래 실체들 설정의 더 많은 안전한 옵션을 선택할 수 있다.
- [0235] 프로그램에 의해 사용자에게는 임계(Critiacs)하에서 거래의 타이밍 설정 옵션이 제공된다는 점에 주목해야 한다. 거래는 즉시, 실시간으로 일어날 수 있으며, 또는 만약 거래 타이밍이 지정되지 않거나 또는 추후의 날짜로

지정되었다면, 실행되자마자 상기 거래는 수동적으로 채워지도록 설정될 것이며, 이것은 자금들이 사용자의 은행 계정들로부터 인출될 수 있지만 채워지는 날짜 전에 시스템의 투자 엔진으로 전환될 수 있다는 것을 의미한다(이하의 투자 엔진 참조).

- [0236] 임계를 설정한 후에, 상기 프로그램은 단계 S117에서 임계 인터페이스를 닫고, 그 후에 거래의 집합은 실행되기에 충분하도록 완료된다. 사용자는 이 지점에서 거래를 저장하고자 원할 수 있다. 사용자는 거래가 완전히 결합되었거나 실행 가능성이 있는지를 불문하고, 어느 단계에서도 거래를 저장할 수 있으며, 또는 그렇지 않으면 한 동안의 단계에서 텍스트 문서를 저장 및 닫을 수 있으며, 편집 또는 완성을 위해 추후에 다시 복귀할 수 있다.
- [0237] 거래를 저장하기 위해, 사용자는 드롭-다운 메뉴에서 "파일(File)"을 클릭해야 하며, "저장(Save)"을 선택해야 하며, 또는 키보드 또는 유사한 것에서 이러한 기능을 실행할 수 있고, 이에 의해 상기 프로그램은 저장될 거래에 명칭 부여를 요구하는 다이얼로그 박스를 제공할 것이다. 다음에 사용자는 파일 이름을 입력하고(예를 들면, 정규 가스 회사 지불), "저장(Save)"을 누르고, 이에 의해 상기 프로그램은 단계 S118에서 상기 거래를 저장할 것이며, 만약 향후에 사용자에게 의해 요구된다면, 사용자에게 손쉽게 이용 가능한 이러한 파일들을 생성하기 위한 최적으로서 지시된 바와 같이, "최근(Recent)" 아래에서 "거래"의 다른 윈도우 또는 드롭-다운 메뉴에서 이용 가능한 거래를 생성할 것이다. 사용자는 언제라도 재사용 또는 검토, 편집, 조절 등을 위해 이용 가능한 어떠한 수의 거래들을 생성 및 저장할 수 있다.
- [0238] 거래가 신택스적으로(syntactically) 실행 가능할 때, 이하에서 언급될 "이러한 거래의 테스트(test this Transaction)" 버튼은 및 "실행 준비(Prepare for Execution)" 버튼은 사용자에게 상기 버튼 기능이 비활성 또는 이용 불가능한 상태로부터 활성 또는 이용 가능한 상태로 변화되었다는 것을 알리기 위해 희미한, 회색의, 차원적인 평면, 또는 유사한 것으로부터 밝은, 컬러의, 양각 및 음영을 가지는, 또는 유사한 것으로 변경될 수 있다는 점에 주목해야 한다.
- [0239] 사용자가 거래가 실행 가능하도록 거래의 결합 및 확인을 종료한 시점에서, 프로그램은 사용자로 하여금 실제의 것을 사실상 실행시키기 전에 거래의 실행을 시뮬레이팅하도록 하는, 사용자가 올바르게 이러한 버튼과 관련된 거래를 결합시키도록 그리고 시스템에서 어떤 다른 거래 또는 활성과는 결합시키지 않도록 하는, "거래 테스트(Test this transaction)" 버튼을 이용가능하게 할 것이다.
- [0240] 사용자가 "거래 테스트(Test this transaction)"를 클릭할 때, VfT 프로그램 모듈은 단계 S119에서, 사용자에게 대해 표현하기 위해, 모든 거래량들 및 모든 실체들에서의 결과적인 변경들의 시뮬레이팅된 계산(accounting)을 처리하는 것 외에도 (기술적인, 금융적인 면 등의) 실행 가능성을 결정하기 위하여 MaQs 프로그램 모듈을 통해 결합된 거래의 모든 실체들에 대한 검사를 개시하도록 하는 명령으로서 이러한 실행을 수신한다.
- [0241] 이러한 방식으로, 사용자가 시스템을 통해 처리하도록 선택하는 모든 사용자 금융 활동들의 범위에 걸쳐 사용자가 결정하고 처리하기 위해 시스템은 사용자의 재정의 예측적인 인식을 제공할 수 있다. 일 실시예에서, 시스템의 프로그램은 과거의 그리고 통계적인 자원들에 기초하여 설정 결정-분석 및 지원(customized decision analysis and support)을 사용자에게 제공할 수 있다.
- [0242] 따라서, VfT 프로그램 모듈은 테스트 명령을 실행하고, 사용자가 이러한 요구를 전송하고 있는 CDMX 프로그램 모듈에 알리고, 상기 실체들을 위해, 그리고 CDMX 프로그램 모듈은 모든 사용자-등록 계정들의 기록들을 보유하는 안전한 "계정 등록(Account Registry)"을 액세스하고, 이러한 계정들에 대한 데이터를 액세스-인증하는 것 외에 적절하고 공식적인 표시(identifying)를 수집한다. 저장되는 것 외에 부가하여, 시스템의 외부에 있는 (결합된 실체 아이콘들을 통해 거래시에 식별되는) 실세계 실체들로부터 정보를 검색하기 위해 요구를 실행하는 MaQs 프로그램 모듈에 대한 이러한 검색된 정보를 처리하기 전에, CDMX 프로그램 모듈은 테스트(시뮬레이션) 목적이라는 표시로 표시된 이러한 액세스 거래 코드를 타임스태핑 및 할당한다.
- [0243] 따라서, 테스트 단계의 일부로서, 요구된 정보를 획득하기 위해 필요하고 이용 가능한 어떤 안전한 금융 네트워크들을 통해서 또는 인터넷을 통해서 적절한 실세계 실체들에서의 사용자의 계정들을 질의한다. 예를 들면, 클라이언트-서버 환경에서, 상기 요구는 클라이언트로부터 서버로 전송되며, 서버는 외부 실체들에 대하여 질의들을 실행한다.
- [0244] 테스트 단계의 일부로서, 프로그램은 가스 회사 시스템이 분주히 작동하고 있다(송금된 거래량을 수신할 수 있는)는 것을 검증하기 위한 배경 검사(back-ground check)를 실행하기 위하여 질의(query)를 송신하며, 요구된 거래 통화, 현재의 거래 처리 시간(예를 들면, 실시간, 18분, 2시간, 4일, 또는 다른 타이머 주기 등), 또는 유사한 것을 결정한다. 더욱이, 은행 거래 시스템이 분주히 작동하고 있는지(거래량을 전송할 수 있는지) 등의

(자금들이 이용 가능한지를 확인하는) 잔고를 검증하기 위해, 동일한 단계에서 유사한 검사가 실행된다.

- [0245] 상기 프로그램은 다양한 실체 계정들로부터 정보를 수신하고, 그 시점에 상기 MaQs 프로그램 모듈은 상기 네트워크들을 통해 요구를 행한 결과로서 올 수 있는 어떤 일반적인 시스템-상태 또는 네트워크 정보 갱신들 뿐만 아니라 검색된 정보를 상기 CDMX 프로그램 모듈로 다시 전달한다. 그다음 상기 CDMX 프로그램 모듈은 모든 수신된 정보를 타임스탬핑하고 부호화하며 저장하고, 다시 테스트로서 구별될 수 있는 방식으로 상기 코드에 다시 마킹한다. 그다음 상기 CDMX 프로그램 모듈은 요구된 정보만을 다시 상기 VtT 프로그램 모듈로 전달하기 위하여 그것이 수신하는 정보를 필터링한다. 따라서, 사용자의 거래에 참가한 실시간 시뮬레이션된 실행 데이터는 단계 S120에서 사용자를 위해 그것을 상기 실체 아이콘들 옆의 스크린상에 표시하거나 그것을 풀-다운 메뉴 등 하에서 이용가능하게 하는 프로그램에 의해 상기 클라이언트에 제공된다(도 17 참조).
- [0246] 공통 메시징 및 경고 대화들, 창들, 경보들 등에 부가하여, 테스트 진행 및 실제 실행 진행하에서 거래의 디스플레이 및 전체 스크린 인터페이스의 그래픽 특성들은 진행이 실제로 진행중이고 관리되고 있는 사용자에게 쉽사리 전달하기 위하여, 밝기, 컬러, 치수, 크기, 배치 등과 같은 것의 차를 사용하여 구별될 수 있도록 변경될 수 있다.
- [0247] 사용자가 지불 기능을 테스트할 수 있는 방법의 예는 다음과 같다: 예를 들어, "테스트" 기능을 사용함으로써, 상기 프로그램은 자금이 사용자의 은행에서 이용가능한지를 자동으로 체크할 것이고 또한 가스 회사의 계좌가 상기 거래를 수락하도록 이용가능한지를 체크할 것이다. 예를 들어, 사용자의 은행 계좌가 미결제 자금(uncollected fund)을 갖고 있거나 상기 가스 회사의 계좌가 오프라인인 경우, 상기 거래는 발생할 수 없을 것이고, 상기 프로그램은 팝업되거나 그렇지 않으면 상기 디스플레이상에 나타날 대화 상자에 메시지로 사용자에게 상기 거래가 발생할 수 없다는 것을 통지할 것이다. 사용자는 상기 거래를 즉시 다시 시도하는 옵션을 가지거나 상기 거래가 나중에 발생하도록 프로그램할 수 있거나 상기 거래를 취소할 수 있다.
- [0248] 특히, 상기 "테스트" 옵션을 사용할 때, 사용자는 상기 VtT 프로그램 모듈을 사용하는 상기 프로그램에 의해 시뮬레이션된 거래의 시각적 표시를 볼 것이다. 예를 들어, 상기 프로그램은 크기가 커지는 "나의 은행 계좌" 아이콘의 상부에 있는 돈-가방 아이콘을 표시할 수 있다; 사용자에게 의해 설정된 정확한 금액이 획득될 때까지 금액이 증가하는, 상기 돈-가방 아이콘에 부착된, 금액이 표시된 텍스트 문자들을 갖는다.
- [0249] 동시에, 상기 프로그램은 예를 들어 "나의 은행 계좌" 아이콘에 부착된 작은 회계(accounting) 박스가 표시되도록 할 수 있는데, 상기 회계 박스는 사용자의 은행에서의 현재의 잔액을 보여줄 것이고, 둘째 줄에 포지티브 돈-가방 증가와 동등한 부채와 동일한, 점점 더 네거티브(빨강)가 되는 회전 금액을 보여줄 것이다.
- [0250] 일단 사용자에게 의해 설정된 바와 같은 백터 금액이 도달되면, 상기 프로그램은 상기 백터 금액이 상기 가방내에서 두번 명멸하게 하거나 통신 완료에 대한 어떤 유사한 표시를 야기할 것이다.
- [0251] 다음, 상기 프로그램은 예를 들어 현재의 계좌 잔액의 부착된 텍스트 표시자와 함께, 상기 가스 회사 아이콘 위에 원기둥 아이콘을 표시할 것이다. 계좌에 빚이 있는 경우, 이 텍스트는 네거티브 및 빨간색이 될 것이며, 상기 원기둥 아이콘은 비게 될 것이며 그것의 경계는 빨간색이 될 것이다. 현재 빚이 있는 것으로 표시된 아무런 금액도 없는 경우, 검은색의 달러 금액이 표시될 것이고(즉 \$0.00) 상기 원기둥 아이콘은 비게 될 것이며, 상기 경계는 검은색이 될 것이다. 상기 가스 회사 계좌가 현재 송장 금액보다 더 큰 신용 금액을 유지하고 있는 경우, 상기 신용 금액은 녹색 텍스트로 표시될 것이고, 상기 원기둥 아이콘 경계는 녹색이 될 것이며, 상기 원기둥 아이콘은 작은 양의 녹색 체적을 보여줄 것이다.
- [0252] 테스트 동안, 상기 프로그램은 한 실체로부터 다른 실체로 자금을 이동시킬 것이어서, 상기 돈-가방 아이콘은 줄어들기 시작하고, 동시에 자산 백터는 예를 들어 그것을 통해 이동하는 크기의 증가를 보여줄 것이다(즉, 어떤 큰 것이 파이프를 통해 이동중이었던 것처럼, 그것이 상기 파이프의 길이를 따라 이동될 때 그것의 직경을 임시로 스트레칭하는 것).
- [0253] 그다음 상기 프로그램은 예를 들어 상기 거래가 진행됨에 따라 천천히 채워지는, 상기 가스 회사 아이콘 위에 상기 원기둥 아이콘을 표시할 것이고(즉, 물이 그것의 내부를 채우는 것과 같이), 상기 원기둥 아이콘에 부착된 텍스트는 상기 원기둥 아이콘상에 표시된 자산 유입을 수치적으로 나타낸다.
- [0254] 거래된 금액이 한 실체로부터 다른 실체로 이동완료되었을 때, 상기 프로그램은 예를 들어 둘째 줄에 최종 금액이 유지되는 상기 원기둥 아이콘에 부착된 텍스트를 표시하고 셋째 줄에 상기 거래의 회계를 보여주는 텍스트를 표시할 것이다(즉, 최종 결과, 지불된 잔액을 보여주고 아무런 자금도 빚지지 않은 거래 완료를 보여주는 검은

색의 달러 금액(즉, \$0.00), 어떤 남아있는 미결제 잔액(네거티브 빨강) 또는 녹색 텍스트의 신용(초과 지불)).

[0255] 또한 상기 프로그램은 예를 들어 상기 거래의 회계를 보여주는, 상기 "나의 은행 계좌"에 부착된 텍스트를 표시할 것이고, 자금들은 상기 은행 계좌의 잔고에서 빼내진다.

[0256] 상기 "테스트" 특징이 특정 유형의 아이콘들을 사용하여 설명되었을지라도, 어떤 유형의 아이콘도 사용될 수 있다는 것과, 사용자의 은행 계좌로부터 상기 가스 회사의 자금의 이동을 보여주는 어떤 유형의 표시자도 사용될 수 있다는 것을 주목하라. 따라서, 상기 거래에서 자금의 이동을 표시하는 사용자 인터페이스에 관한 기술은 상기 프로그램이 은행 계좌에서 자금의 이용가능성, 자금을 이체하기 위한 은행의 능력 및 상기 거래를 수락하는 가스 회사의 능력 등을 테스트하는 동안 사용자의 이익을 위한 것이다.

[0257] 상기 테스트가 완료된 후, 상기 프로그램에 의해 단계 S121에서, 사용자는 "테스트 다시 실행", "거래 수정" 또는 "거래 저장"에 대한 대화 상자를 통해 거래 옵션들이 프롬프트될 것이다.

[0258] 이 예에서, 사용자는 "거래 저장"을 선택할 수 있고, 상기 프로그램은 후속 실행 또는 장래의 검색, 리뷰, 수정 및 실행 등을 위해 상기 거래를 저장할 것이다.

[0259] 후일 또는 다중 자동-시간 지불의 경우, 거래 활성화의 경고 통지들이 상기 프로그램에 의해 사용자가 지정한 이메일 계정으로 송신될 수 있는데, 상기 통지는 보안 시스템 웹페이지에 대한 URL을 포함할 수 있고, 인터페이스는 처음 시스템 로그인과 패스워드를 요구한 후, 성공적인 사용자-로그인시, 상기 프로그램은 거래 명세의 표시를 제공하고, 요구된 패스워드들의 입력을 취하는 창들을 형성하며, "실행 준비" 및 "거래 실행" 버튼들 뿐만 아니라 확인 검사 상자들 등을 제공할 것이다. 상기 프로그램은 상기 시스템에 다음 로그인시 사용자에게 이용가능하고, 다음번에 사용자가 상기 (PC) 클라이언트로부터 시스템으로 로그인할 때 자동으로 로컬-클라이언트(즉, 사용자의 PC)에 갱신가능한, 어떤 다른 곳에서 설명된 바와 같이 저장된 거래 파일들에 거래 명세 및 결과들을 기록할 것이다.

[0260] 사용자가 방금 테스트된 거래의 상세를 리뷰하고자 할 때, 사용자는 "거래의 텍스트 요약"이라고 불리우는, 거래 워크패드의 하부에 있는 버튼을 클릭할 수 있고, 그것에 의해 상기 프로그램은 사용자의 은행 계좌의 관련된 모든 상세 및 상기 CDMX 프로그램 모듈이 테스트 거래에 할당된 유일한 코드를 포함하는 테스트 거래 요약을 표시할 것이고, 상기 코드는 상기 코드가 테스트된 후 실제로 실행되었다는 것을 보여주기 위하여 실행시에 부분적으로 변경될 수 있다(예를 들어, 상기 코드는 숫자들 및 문자들로 구성될 수 있는데, 한 문자 "T"는 "테스트"를 나타내고, 실행시 "E"가 상기 코드에 부가될 수 있다). 이러한 방식으로, 상기 CDMX 프로그램 모듈은 용이하게 구별할 수 있고, 사용자는 리뷰되고 있는 거래가 실행 또는 테스트되었거나 양자가 행해졌는지, 그리고 상기 가스 회사의 청구서, 지불 등에 대한 상세 뿐만 아니라, 거래 금액, 상기 벡터 임제에 의해 선택된 타이밍 등을 용이하게 이해할 수 있다.

[0261] 사용자는 요망되는 경우 이것을 프린트할 수 있다. 사용자는 또한 테스트된 거래 및/또는 실행된 거래 중 하나 또는 양자에 대한 텍스트 요약을 프린트할 수 있다. 실행된 거래의 요약은 거래의 영수증으로서 취하기 위하여 모든 필요한 정보 및 추적 코드들을 포함할 것이고, 상기 정보는 지불이 개시되었다는 것을 증명하기에 충분할 것이다.

[0262] 상기 CDMX 프로그램 모듈은 모든 거래들 및 그들에 포함된 실체들을 추적하기 때문에, 거래 코드들을 포함하는, 상기 프로그램에 의해 제공되는 사용자-프린트 텍스트 요약은 실체들을 수신하는 실제 세상에서 고객 서비스 전문가에 대해 정확한 거래 상세들을 식별할 수 있을 것이다. 거래들은 (ACH, PayPalTM 등과 같은) 전통적이고 기존의 안전 지불 네트워크를 통해 실행될 것이기 때문에, 상기 수신하는 실체들은 이미 시스템-개시 거래들을 수신할 수 있다. 그래서 상기 수신하는 실체는 어떤 제3자(즉 본 발명의 시스템)가 지불을 개시했는지를 상관하지 않을 수 있고 동일한 기반 구조를 통해 수신된 모든 다른 것들과 함께 단순히 상기 지불을 기록할 것이어서, 고객이 상기 시스템-개시 거래에서 벡터 목적지 실체들인 수신하는 실체들의 고객 서비스 부문에 어드레싱하기를 원하고 지불 금액 및 타이밍 등과 같은 특정 거래 상세들을 논의하기를 원하는 경우 리뷰를 위해 그것이 용이하게 검색될 수 있게 한다.

[0263] 이제, 상기 테스트가 성공적이었고 사용자가 상기 거래를 실행하고 싶도록 시뮬레이션된 결과들이 만족스러운 경우, 사용자는 거래 워크패드의 하부에 있는 "실행 준비" 버튼을 클릭할 수 있다. 단계 S122에서 상기 프로그램은 거래 워크패드를 줄어들게 할 수 있을 것인데, 예를 들어 상기 디스플레이 스크린의 상부 좌측으로 움츠리게 하여 훨씬 더 작은 형태로 상기 거래 실체들 및 그들을 연결하는 벡터만을 보여준다.

- [0264] 상기 창 아래에, 상기 프로그램은 사용자에게 의해 저장되는 바와 같은 거래명을 표시할 것이고, 상기 거래명 아래에, 상기 특정한 수집된 거래 및 벡터를 사용하여 형성된 모든 이력 거래들(어떤 것이 존재하는 경우)을 보여주는 예를 들어 현재의 연대순으로 아래로 회계 원장 또는 거래 이력을 표시할 것이다.
- [0265] 상기 창 아래에, 상기 프로그램은 "거래되는 것에 대한"으로서 현재의 거래의 금액 및 날짜를 보여주는 최종 회계 엔트리를 보여줄 것이다.
- [0266] 상기 프로그램은 상기 디스플레이 스크린의 하부에 "이 거래를 잠금" 버튼을 표시할 것이고, 그다음 사용자는 상기 버튼을 클릭할 수 있다. 상기 "이 거래를 잠금" 버튼을 누르자마자, 상기 프로그램은 예를 들어 모든 필요한 양상들에서 검사되고 검증될 필요가 있는 아이템들(즉, 사용자의 은행 계정내의 충분한 자금, 상기 가스 회사 시스템 온라인의 이용가능성 등)의 리스트를 포함하여, 상기 거래의 회미한 구성 요소들을 보여주는, 수직 직사각형 창을 표시할 것이다.
- [0267] 그다음 상기 프로그램은 검사되고 검증될 상기 아이템들의 체크리스트를 통해 진행할 것이고, 각 아이템은 상기 프로그램에 의한 리뷰시 불이 들어온다(즉, 상기 아이템은 문제가 있는 경우, 빨간색으로 변할 것이거나, 거래가 진행될 수 있는 경우 녹색으로 변할 것이다).
- [0268] 일단 상기 프로그램이 상기 거래의 모든 양상들을 긍정적인 것으로 검사하고 검증한 경우, 상기 프로그램은 단계 S123에서 확인 아이콘 뿐만 아니라, "거래가 잠겨지고 행할 준비가 됨" 메시지 등과 같은 것을 표시할 것이고, 그다음 상기 프로그램은 상기 거래가 있는 거래 워크패드와 관련된 표시된 "이 거래를 실행함" 버튼 등과 같은 것을 활성화할 것이다. 상기 디스플레이상의 버튼의 배치는 사용자가 상기 버튼을 관련된 거래와 정확하게 연관시키도록 그리고 상기 시스템에서 어떤 다른 거래 또는 활동과 그것을 혼동하지 않도록 되어 있어야 한다. (상기 버튼은 사용자의 재량으로 사용에 대한 활성 이용가능성의 상태로 전이될 것이다. 상기 전이는 오디오, 비디오, 애니메이션과 같은 것들을 포함하는 변경들 또는 밝기, 컬러, 형상, 치수, 배치 등과 같은 상태 변경들로 인식될 수 있다).
- [0269] 그다음 사용자는 상기 "이 거래를 실행함" 버튼을 클릭할 수 있고, 단계 S124에서 상기 프로그램은 상기 수직 직사각형 창의 내용들을 지우고 수직 진행-아래쪽 하이라이트된 막대 뿐만 아니라 상기 거래에 대한 새로운 실행 아이템 리스트를 표시할 것이다. 각 아이템("가스 회사 시스템으로 거래의 사전-통지를 송신하라", "사용자 은행 계정으로부터 설정 금액을 분할하라", "분할된 금액을 해제하라", "가스 회사에 의한 수신을 확인하라" 등과 같은)이 상기 프로그램에 의해 수행되고 검증될 때, 완료된 아이템은 예를 들어 녹색으로 변할 것이고, 예를 들어 그것의 좌측에 체크마크를 획득하며, 진행 막대는 상기 목록상에서 아래쪽으로 계속될 것이다. 선택된 각 거래 실체에 대하여, 상기 프로그램은 상기 거래 유형, 목적-실체(즉 송금, 요구, 구입, 판매 등) 및 포함되는 당사자-실체(즉, 은행, 보험 회사, 공익 사업체 등)에 따라, 검증될 아이템들의 미리 결정된 집합의 명령어들에 액세스할 것이다.
- [0270] 실행중, 줄어든 거래 및 벡터 어셈블리는 상기 실행 프로세스에서의 각 실제 진행 단계와 함께, 상기 테스트에서 설명된 그래픽 디스플레이를 반복할 것이다. 따라서, 예를 들어 상기 실행 프로세스가 "분할된 금액을 해제하라"를 실행하는 경우, 상기 테스트 절차동안 설명된 바와 같은 상기 돈-가방은 상기 벡터 등을 통해 유출되고 이동되기 시작할 것이다.
- [0271] 상기 프로그램은 또한 번호가 상기 거래에 대한 모든 당사자들(즉, 은행, 사용자, 가스 회사)에 의해 추적될 수 있는 상기 거래에 대한 유일한 거래 번호를 제공한다.
- [0272] 상기 거래의 완료 및 상기 거래 번호의 할당시, 상기 프로그램은 단계 S125에서, 상기 거래 번호 뿐만 아니라, 상기 벡터 거래의 다이어그램, 모든 계정 상세들, 기관 접촉 정보 및 특정 공식적인 거래-시스템 타임 스탬프 데이터를 포함하는, 포괄적인 거래 기록을 표시할 것이다.
- [0273] 그다음 사용자는 사용자의 기록들을 위해 상기 포괄적인 거래 기록을 프린트할 수 있거나, 단지 국부적으로 또는 상기 서버상에 디지털 복사본들을 저장할 수 있다(네트워크 기록들은 बैंकिंग 규정에 따라 자동으로 유지될 것이다.). 또한 사용자는 (Quicken[™]과 같은) 어떤 주요 금융 회계 소프트웨어로 거래 상세를 보낼 수 있다. 거꾸로, 주요 금융 회계 소프트웨어는 이러한 애플리케이션들내에서부터 본 발명에 의해 가능하게 되는 거래들을 실행하는 능력을 가질 것이고, 모든 요약 데이터는 상기 금융 소프트웨어의 로그들 등으로 직접 반환되는 것이 고려된다.
- [0274] 상기 저장된 거래는 장래의 거래들을 위한 템플릿이 되는데, 그것에 의해 사용자는 상기 저장된 거래를 재개할

수 있고 미래의 날짜에 다른 백터된 거래를 수행할 수 있는데, 이것은 임계를 확인하고 실행할 필요만 있다. 마찬가지로, 원본과 유사한 신규 거래들로 이들을 신속하게 수정하기 위하여, 기존의 저장된 거래들의 복사본들이 형성될 수 있다.

[0275] 이것의 예는 사용자가 원래의 은행 계정 대신에 다른 송금 실체를 가지고, 제2의 "가스 회사에 대한 지불" 거래 어셈블리를 생성하고 싶은 경우일 수 있다. 사용자는 우선 기존의 저장된 거래 파일 아이콘을 한번 클릭할 것이고, 그다음 "파일" 폴다운 메뉴로 가서, "거래 파일 복사"를 선택하는데, 상기 프로그램은 기존의 저장된 거래의 카본 카피를 생성할 것이고, 그것의 이름은 원본이 아니라는 것을 반영하도록 자동으로 조정되지만 카본 카피이다. 그다음 사용자는 상기 카본 카피를 더블클릭할 수 있고, 상기 프로그램은 사용자가 상기 실체들 중 어떤 실체 또는 모든 실체들을 편집하고 교체하며 조정할 수 있도록 할 것이다. 사용자가 이것을 행하는데 있어서의 이익은 상기 카본 카피에서의 실체들에 대한 모든 설정들이 그들이 원본에 있는 것과 동일하게 유지될 것이라는 것이어서(그들이 사용자의 재량으로 변경될 수 있을지라도), 예를 들어 백터 방향은 상기 가스 회사로 송금을 행하도록 요구되는 바와 같이 이미 적절히 설정될 것이다.

[0276] 상기 거래가 실행 이전에 테스트되는 것으로 설명될지라도, 사용자는 거래를 설정할 수 있고 그다음 직접 실행을 진행할 수 있어서 테스트 기능을 스킵할 수 있다.

[0277] 전통적인 단순한 거래가 본 발명과 일치하는 분산 금융 시스템의 전 능력 중 작은 부분만을 사용할지라도, 상기 기능은 현재에 사용자들에 가장 쉽사리 이용가능한 것 중 하나일 것 같다. 본 발명과 일치하는 상기 분산 금융 시스템은 제3자 플랫폼이기 때문에, 사용자는 단일 계정을 유지할 수 있고, 동일한 계정 및 로그인 시스템을 통해, 은행 계정들이든 아니든 어떤 수의 다른 계정들, 신용 카드 계정들, 공공사업체 청구서들, 다양한 대부금, 투자 포트폴리오 등을 감시할 수 있고, 여기에서 부분적으로 설명된 바와 같이 경험적인 거래 당사자-실체들 및 거래 목적-실체들을 사용하여 요망될 때 그들중 어떤 것 또는 모두로부터 거래들을 형성하고 테스트하며 실행할 수 있다.

[0278] 전통적인 복합 거래

[0279] 본 발명과 일치하는 일 실시예에서, 전통적인 단순 거래와 유사한, 전통적인 복합 금융 거래가 형성되고, 테스트되며 실행될 수 있다. 본질적으로, 상술된 바와 같이, 상기 전통적인 단순 거래는 전통적인 금융 시스템들에서 보통 접속되지 않는 두개의 실체들을 포함한다. 상기 전통적인 복합 거래는 많은 다양한 형태를 취할 수 있기 때문에, 3개의 예들이 설명될 것이다. 더욱이, 사용자 인터페이스가 상기 전통적인 단순 거래에 관해 상세히 설명되었기 때문에, 상기 사용자 인터페이스는 이후에 상세히 설명되지 않을 것이다.

[0280] 예 1

[0281] 일단 사용자가 로그인하고 인증되며, "신규 거래"를 가져오고, 상기 전통적인 단순 거래에서 상술된 바와 같이 공백 거래 워크페드에 있는 경우, 사용자는 상기 전통적인 단순 거래에서 설명된 것과 동일한 프로세스를 계속할 수 있다. 예를 들어, 사용자가 "나의 주식"을 선택하는 것을 원할 수 있고 상기 "나의 주식" 아이콘을 마우스오버함으로써, 상기 프로그램은 사용자의 계정에 액세스할 수 있고 상기 계정과 관련된 어떤 실시간 데이터도 획득하고 표시할 수 있다.

[0282] 본 발명의 방법들 및 시스템들과 일치하는 한 예에서, 사용자는 "나의 주식" 및 "나의 자동 대출"을 선택할 수 있다 - 현재의 당해 자동 대출은 예를 들어, 18개월 지불이 행하기 위해 남아있다는 것을 갖는다. 사용자는 이제 두개의 아이콘들 및 사용자가 원하는 것은 무슨 주식이든지 판매하기 위한 거래 목적 아이콘 - "판매" - 를 사용하여 거래를 할 수 있고, 그다음 단일 프로그램된 거래로서 상기 자동 대출금을 전부 갚기 위하여 유동 자산을 관리하는 자산 백터를 사용할 수 있고, 본 발명의 상기 분산 금융 프로그램에 의해 모든 것이 기록되고 관리된다.

[0283] 상기 전통적인 단순 거래에서와 같이, 사용자는 거래를 형성하고, 그것을 테스트하며, 그것을 변경하고, 그것을 하나의 거래로서 실행할 수 있다. 따라서, 상기 금융 거래에 수반되는 인터브로커 또는 틈새 당사자들을 가질 필요가 없다.

[0284] 상기 MaQs 프로그램 모듈은 상기 VfT 프로그램 모듈을 사용하여 구성되며 상기 CDMX 프로그램 모듈을 사용하여 부호화된 명령들을 실행하는 것과 시장과 통신하는 것을 관리한다는 것을 주목하라. 상기 VfT 프로그램 모듈은 상기 MaQs 프로그램 모듈 및 상기 CDMX 프로그램 모듈로부터 공급된 정보에 의해 갱신된, 모든 거래들의 가시적인 기록을 유지할 것이다. 또한, 상기 VfT 프로그램 모듈은 상기 거래들을 저장할 것이고 금액, 상대 당사자 계정 번호들(counterpart account numbers), 시간 등을 분석하기 위하여 그들을 리뷰 및 추적 목적들을 위해 이용

가능하게 할 것이다.

[0285] 예 2

[0286] 본 발명의 방법들 및 시스템들과 일치하는 다른 예에 있어서, "신규 거래" 메뉴 아래에서, 사용자는 "조회"와 같은 옵션을 선택할 수 있고, "대출", "보험" 등과 같은 다수의 옵션들에서 선택할 수 있다. 이 경우, 사용자는 "대출"을 선택할 수 있는데, 이 시점에서 상기 프로그램은 사용자에게 다양한 대출 매개 변수들(즉, "대출 매개 변수 설정" 옵션)을 입력하도록 프롬프트할 것이다. 일단 사용자가 상기 매개 변수들을 입력하면, 상기 프로그램은 사용자의 명세에 관한 대출을 위해, 시장(즉, 은행들, 신용 카드 회사들 등) 또는 (대출을 신디케이트하는데 참가하는 많은 대금업자들을 초대하는) 상기 분산 자본 시스템 집단에 조회할 것이고, 이자율, 신용 데이터 등에 대한 실시간 정보를 반환할 것이다. 사용자가 대출을 신청하기로 결정한 경우, 사용자는 그렇게 할 수 있고, 전통적인 방식으로 (전통적인 단순 거래를 구성할) 은행과 같은 단일 상대 당사자 또는 (공동 단순 거래를 구성할) 본 발명의 현재 프로그램에 의해 자동으로 조정되는, 신디케이트 방식으로 다수의 대금업자들과 온라인 상으로 그렇게 할 수 있다. 어떤 접근이 대출을 안전하게 하는데 사용되는지에 상관없이, 이하, 이 예에서 초기 활동은 연속적인 거래 활동에 링크되며, 전통적인 복합 거래의 구성 요소가 된다. 이러한 방식으로, 상기 거래 유형들은 단지 독립적인 거래 유형들로서 사용되는 것에 한정되지 않고, 사용자들이 따라 잡을 수 있는 어떤 능률적인 결과를 달성하도록 결합될 수 있다.

[0287] 예를 들어 사용자에게 온라인 대출이 허가되고 은행 또는 신용 연합등에서 이용가능한 자금을 갖고 있다고 가정하면, 사용자는 예를 들어 사용자의 신용 카드들 또는 아이의 대학 대출금 등을 지불하기 위하여 대출 자금을 전개할 수 있다. 상기 거래는 사용자가 "나의 대출" 아이콘, "나의 신용 카드들" 아이콘 및 "나의 대학 대출" 아이콘을 가져오고, 그들을 "자산 벡터"와 같은 "거래 목적"과 연결시키며, 대출 자금을 사용하여 신용 카드 및 대학 대출 청구서를 지불한다는 점에서 상기 전통적인 단순 거래들과 유사하다(도 18 참조).

[0288] 자금은 하나 이상의 "자산 벡터"를 사용하여 대출금으로부터 분할될 수 있다는 것을 주목하라. 상기 프로그램은 동시에 또는 연속적으로 나오는 자산 벡터들을 실행하도록 지도될 수 있다. 테스트 및 실행 요구들은 한번에 모든 계류중인 거래들을 위해 실시될 수 있다.

[0289] 더욱이, 당업자는 이러한 유형의 거래가 다수의 상이한 타겟들(즉, 피고용인들에게 봉급을 지불하는 고용인 등)과 상호작용하는 어떤 실체간에 수행될 수 있다는 것을 인지할 것이다.

[0290] 예 3

[0291] 본 발명의 방법들 및 시스템들과 일치하는 또 다른 예에 있어서, 예 1에서와 같이, 사용자가 단지 특정 가격에 주식을 "판매"하기를 원하는 경우, 사용자는 투기적인 활동을 프로그램할 수 있다. 상기 프로그램은 사용자에게 판매의 매개 변수들을 설정하는 옵션을 제공할 것이다(대출 매개 변수들이 설정되는 예 2에서와 같이). 주식 가격은 상기 프로그램에 의해 실시간으로 획득될 수 있고 사용자를 위해 표시될 수 있기 때문에, 사용자는 주식 가격이 사용자가 팔고 싶은 가격보다 더 낮은 지를 알 수 있을 것이다.

[0292] 따라서, 상기 프로그램은 상기 프로그램이 사용자에게 의해 프로그램되는 바와 같이, 실시간으로 어떤 간격으로 시장에서 거래를 테스트하게 하여, 사용자에게 사용자가 원하는 가격이 달성될 때 주식을 판매하는 옵션들을 제공할 수 있거나, 그 시간에 주식 가격이 무엇이든지간에 특정 시간 또는 날짜에 주식을 파는 옵션을 제공할 수 있다(즉 "판매 매개 변수 설정" 기능은 "대출 매개 변수 설정"의 것과 유사한, 메뉴의 일부로서 제공된다).

[0293] 또한 상기 프로그램은 사용자가 어떤 트리거된 이벤트에서 어떤 금융 증서에 투자하도록 상기 시스템을 미리 설정하고 상기 시스템이 자동으로 상기 투자에서 빠져 나와서, 퇴장-트리거가 시동되는 경우, 이에 따라 사용자의 은행 계정으로 다시 이익을 전달할 수 있도록 트레일링-스톱-로스(trailing-stop-loss)와 같은 기능들을 통합한다.

[0294] 따라서, 사용자는 사용자에게 의해 설정된 매개 변수들에 따라, 장래에 발생할 수 있는 어떤 유형의 투기 (또는 비-투기) 활동을 프로그램할 수 있다.

[0295] 게다가, 상기 프로그램은 상기 매개 변수들이 "테스트"시 충족될 때 상기 거래들의 자동 실행시, 사용자가 로그인할 때 사용자에게 메시지를 제공하거나 사용자의 컴퓨터를 통해 사용자에게 또는 사용자의 이동 전화, 페이지 또는 이동 오거나이저 등에 이메일을 제공함으로써, 프로그램된 거래가 발생했다는 것을 사용자에게 자동으로 (또는 사용자에게 의한 요구시) 통지할 수 있다.

- [0296] 따라서, 상기 거래는 본 발명의 분산 금융 시스템을 통해 원격으로, 안전하게 그리고 자동으로 수행될 수 있다.
- [0297] 공동 단순 거래
- [0298] 상기 전통적인 복합 거래와 상기 공동 단순 거래간의 차이점은 전자는 단일 고객이 한 장소에서 함께 다수의 단일-사용자 계정들 및 거래들을 조작하는 것을 가능하게 하는 반면에, 후자는 다수의 고객들이 동일한 계정 및/또는 거래에 참가할 가능성이 있다는 것이다. 상기 공동 단순 거래의 한 실시예와 일치하는 방법들 및 시스템들이 아래에 설명된다. 전통적인 복합 거래의 예 2에서 상술된 바와 같이, 상기 거래 유형들은 사용자들의 요구를 달성하는 확장된 거래들을 생성하는데 적합하다고 사용자들이 알 때 결합될 수 있다는 것을 주목하라.
- [0299] 예 1
- [0300] 공동 단순 거래의 예에 있어서, 사용자는 다른 시스템 사용자들로 확장되는, 투자 기회 또는 다른 자금-모금 활동을 설정할 수 있다. 상기 "투자 기회"는 이러한 유형의 거래가 소정 수의 사용자들에 제공되는 것을 제외하곤, 상기 전통적인 단순 거래 예에서 설명된 것과 유사한, 거래 워크패드상에서 사용자에게 의해 설정되거나, 그렇지 않으면 요망되는 투자 자금들의 상한 금액과 같이 정의되거나 제약된다(예를 들어, 자선 단체에 기부하기 위한 돈을 모금하기 위하여 또는 법인 유가 증권에 투자하도록 다른 사람들을 초대하기 위하여).
- [0301] 그다음 사용자는 예를 들어 사용자가 상기 "기회"를 송신하고 싶은 사용자들("피초대자들")의 목록을 생성하기 위하여 또는 다른 예에서, 미리 설정된 제약을 충족하는 어떤 사용자에게 상기 기회에 대한 액세스를 제공하고 가변 제약들을 선택하기 위하여 메뉴 옵션들을 사용할 수 있다. 이러한 유형의 거래에 참가하는 사용자의 유형의 구별은 정방향 객체로 표현된(도 19) 아마도 이름이 있는 특별히 초대된 당사자들 및 원들로 표현된(도 19) 이름이 없지만 미리 설정된 제약들에 따라 자격이 주어진 당사자들을 가지고, 사용자 인터페이스상에 가시적으로 표시될 수 있다. 상기 프로그램의 상기 CDMX 모듈은 제공(offering)을 위해 유일한 모 거래 번호를 제공할 것이고, 그다음 각 피초대자 사용자 이름을 위해 차일드 코드(child code)를 제공할 것이다; 또한 CDMX는 열거된 각 피초대자에게 분배하기 위하여 유일한 제공 거래 피초대자 패스워드들을 생성할 것이다.
- [0302] 더욱이, 사용자는 상기 메뉴 및 풀다운 옵션들에 액세스할 수 있고, 사용자는 상기 거래에 대한 다양한 제약들, 예를 들어 상기 거래에서 수락 및/또는 참가에 대한 시간 제한(즉 30일)을 조정할 수 있다.
- [0303] 일단 피초대자들이 사용자에게 의해 확인되면, 상기 프로그램은 자동으로 사용자의 시스템 계정들을 통지할 것이어서, 사용자들이 그들의 계정들에 로그인하는 다음 번에, 그들은 아마도 명멸하는 아이콘 등의 형태로, 상기 제공 기회의 통지를 알 것이다. 예를 들어 마우스-클릭을 가지고, 상기 기회에 참가하기 위한 초대에 대한 통지를 확인응답할 때, 상기 시스템은 특정 제공 거래 패스워드의 입력을 요구할 것이다. 사용자가 상기 패스워드를 가지고 있지 않은 경우, 상기 제공은 액세스가능하지 않을 것이다. 상기 피초대자들은 이메일을 통해 (또는 전화, 우편 등에 의해) 그들의 개별 제공-특정 패스워드들을 수신할 것이다. 이러한 방식으로, 상기 거래의 사용자 소유자는 틀린 사람이 그 또는 그녀를 위해 의도되지 않았던 거래에 대한 액세스를 우발적으로 획득하기 위하여 실수로 피초대자를 두번 식별해야 할 것이다(상기 시스템에 대해 상기 피초대자를 식별하는데 있어서 한번, 그리고 아마도 상기 특정 제공 거래 패스워드를 갖는 이메일을 동일한 틀린 사람에게 송신함으로써, 상기 피초대자와 통신하는데 있어서 다시 한번).
- [0304] 정당한 피초대자가 상기 시스템에 의해 성공적으로 상기 제공에 관해 통지되지만, 틀린 사람이 상기 제공 특정 거래 패스워드를 수신하는 경우, 그 사람이 시스템 계정을 가지고 있어서 로그인 할지라도, 상기 제공 초대에 관해 아무런 통지도 없을 것이고, 그래서 상기 패스워드는 쓸모없게 될 것이다. 거꾸로, 상기 시스템-식별 피초대자가 틀리고 상기 틀린 사람이 상기 제공 거래의 초대 통지를 수신하는 경우, 사용자는 상기 기회에 액세스하기 위하여 필요한 패스워드를 가지지 않을 것이다. 이러한 방식으로 상기 시스템은 안전하도록 설계되고, 설계 부주의하고, 의도되지 않으며 심지어 요망되지 않은 금융 상호작용에 의해 제한되도록 설계된다.
- [0305] 액세스가 입력되는 다음의 정확한 특정 피초대자 제공 패스워드를 가지고 통지되는 정확한 시스템 계정을 통해 달성되는 경우, 예를 들어 마우스오버동안 또는 메뉴를 풀업하기 위하여 마우스를 정확하게 클릭함으로써 또는 대화 상자 등을 통해, 사용자에게 상기 시스템에 의해 상기 "투자 기회"에 대한 액세스가 제공될 것이다. 상기 거래 명세는 상기 프로그램에 의해 실시간으로 제공될 것이고, 상기 프로그램은 모든 정보를 사용자에게 제공할 것이다(즉, 요구된 금액, 요구된 날짜, 다른 투자자들, 다른 투자자들이 제공한 금액 등).
- [0306] 상기 "투자" 거래 상세의 리뷰 이후에, 상기 피초대자는 상기 "투자 기회"의 "제공"을 "수락" 또는 "거절"할 수 있다. 상기 피초대자가 상기 제공을 거절하는 경우, 상기 프로그램은 상기 거래 워크패드 스크린에서 상기 아이

콘을 삭제할 것이고, 일 예에서, 상기 제공 당사자에 코멘트를 송신하는 수단을 상기 피초대자에게 제공할 것이다. 사용자는 또한 사용자가 후일에 다시 상기 거래를 리뷰하기를 원하는 경우 수락 또는 거절없이 상기 거래에서 빠져 나올 수 있다.

[0307] 사용자가 상기 거래를 수락하기를 원하는 경우, 사용자는 예를 들어 "수락"을 클릭할 수 있고, 그다음 상기 전통적인 단순 거래의 것과 유사한 거래를 설정하도록 진행할 수 있는데, 예를 들어 사용자는 실체 - 이 경우 상기 "투자 기회"에 지불하기 위한 돈을 빼내기 위하여 은행 계정에 액세스한다.

[0308] 상기 전통적인 단순 거래에서와 같이, 사용자는 상기 거래를 테스트하고 그것을 저장할 수 있다. 더욱이, 사용자가 정기적이거나 임의의 간격에 송금되는 설정되거나 가변적인 금액을 포함할 수 있는, 그의 은행 계정 또는 뮤추얼 자금 계정 등으로부터의 자금을 사용하여 상점으로부터의 청구서의 지불을 설정하는 방식과 유사하게, 사용자는 상기 "투자 기회"에 지불하는 지불 방식을 설정할 수 있다. 일단 전체 금액이 상기 "투자 기회"로 지불되면, 상기 프로그램은 아이콘을 닫거나 이용가능하지 않게 할 것이고, 사용자는 청구서가 지불될 때 획득된 것과 유사하게 상기 "투자 기회"로 지불된 금액의 거래 기록을 가질 것이다. 상기 투자 기회 거래 이벤트는 후일에 다시 투자한다는 점에서 다시 이용가능하지 않을 수 있지만, 상기 거래 파일은 상기 피초대자 클라이언트에서 유지될 수 있고 상기 투자 자산에 대한 실시간 정보로 갱신될 수 있다. 이러한 방식으로, 심지어 일회 거래 이벤트가 실시간 정보로 갱신될 수 있다.

[0309] 예 2

[0310] 본 발명과 일치하는 다른 예에서, 두 시스템 사용자들은 온라인 구입에서 협력하는 것을 원할 수 있다. 상기 구입은 아이템을 판매하는 제3자로 송금하기 위하여, 두 사용자들의 은행 계정들, 신용 카드들 등의 각 하나 또는 다수로부터 다수의 상이한 금액의 자금을 요구할 것이다.

[0311] 하지만, 전자 통화(e-currency)와 달리, 판매자는 공동 거래를 수신할 특정 금액을 필요로 하지 않는데, 이것은 상기 프로그램이 상기 MaQs 프로그램 모듈을 통해 기존의 기반 구조를 통한 기존의 계정들에 지불을 유도할 수 있기 때문이다.

[0312] 상기 거래는 예 1에서와 같이 형성되는데, "구입 기회"는 공동으로 상기 구입 금액이 도달될 때까지 상기 "구입 기회"로 지불하기 위하여 제1 사용자가 제2 사용자를 초대할 수 있는 경우 형성될 수 있다. 그 시점에서, 제1 사용자는 판매하는 판매자를 "초대"할 수 있다. 이것을 달성하기 위하여 제1 사용자는 상기 "구입 기회"가 상기 시스템의 프로그램을 통하여 상기 판매자에게 자동으로 송신되게 할 것이고, 그다음 이메일을 통해 구두로 특정 기회 패스워드를 상기 판매자에게 개별적으로 통지할 것이다. 이러한 유형의 거래에 참가하기 위하여 사용자들은 시스템 계정을 가지고 있어야 한다는 것을 주목하라(그렇지 않으면 참가자들을 인증하기 위한 이중-안전 접근은 달성하기 어렵다). 상기 프로그램의 상기 CDMX 모듈은 그들이 안전하게 수행된다는 것을 보장하도록 각 거래를 따를 것이다.

[0313] 예 3

[0314] 본 발명과 일치하는 다른 실시예에서, 예 1은 외국 통화 환전 구성 요소를 가지고, 한 단계 더 사용될 수 있다. 예를 들어, 도 19에 도시된 바와 같이, 상기 거래 워크패드상에서 아이콘(1)으로서 식별되는, 상기 "투자 기회"는 아이콘들(2-5)로 식별되는, 4명의 잠재적인 참가자들로 송신될 수 있다. 사용자는 아이콘(1)과 사용자(좌측의 큰 은행 계정 아이콘)간에 (수직 막대로서 식별되는) 외환 구성 요소를 설정할 수 있고, 거래에 대한 상기 실체들 중 어느 실체도 한 통화를 자동으로 다른 통화로 변환되도록 하기 위하여 한 통화로 자금을 공급할 수 있다. 여기에 단일 대형 거래를 수행하도록 결합된 상이한 거래 유형들의 다른 예가 있다. 통화 환전 능력에 대해서, 하기의 공동 복합 예를 참조하라.

[0315] 공동 복합

[0316] 이 거래는 4개의 실시예들 중 가장 복잡한 것이고, 상기 시스템이 주로 설계되었던 것이다. 공동 복합 거래에서, 수백만 사용자들은 예를 들어 그들의 금융 목표의 추구를 위해 연합될 수 있다. 이전에 설명된 바와 같이, 공동 거래에 있어서, 거래 양측은 그들 자신들을 고객들로서 간주하고 있고, 그렇게 하는데 있어서, 각각의 다른 사람의 제공자 역할을 수행해야 한다. 종종 상기 고객들은 서로를 직접 인지하지 못하고, 본 발명은 단지 상기 거래를 더 효율적이거나 덜 비용이 들거나 둘다가 되도록 공동 복합 실시예를 사용하고 있다.

[0317] 오늘날 현재 이 모델을 따르는 거래들은 존재하지 않으므로, 다음 예는 공동 복합 거래를 설명하기 위한 것이다.

- [0318] 본 발명과 일치하는 상기 공동 복합 거래는 하기에 더 상세히 설명된다. 공동 복합 거래에 있어서, 상기 프로그램은 VfT, SVR, CDMX 및 MaQs 프로그램 모듈들과 같은, 모든 중요한 프로그램 모듈들을 이용한다.
- [0319] 더욱이, 하기에 제공된 제1 예가 통화의 환전을 포함할지라도, 이것은 상이한 거래 유형들 중 어느 것에서도 - 전통적인 단순 거래부터 본 공동 복합까지 달성될 수 있다는 것을 주목하라. 예를 들어, 청구서의 지불을 포함하는 전통적인 단순 거래는 아이템에 대해 프랑스 프랑(또는 유로)로 프랑스내의 실체에 지불하고 미국 달러를 프랑(또는 유로)으로 환전함으로써 수행될 수 있다. 따라서, 통화 환전은 상기 공동 복합 예의 단독 영역이 아니다.
- [0320] 예 1
- [0321] 따라서, 본 발명과 일치하는 한 예에서, 그리고 공동 복합 거래에 대해 간략하게 이전에 설명된 바와 같이, 사용자의 회사는 어떤 활동을 위해 영국 파운드를 요구할 수 있다. 사용자는 시스템 웹사이트에 액세스할 것이고 예를 들어 전통적인 단순 거래하에서 설명된 바와 같이, 상기 VfT 프로그램 모듈을 사용하는 상기 거래를 형성하는 단계들을 사용하여, 거래 워크패드상에서 도 20의 단계 S200에서 "통화 환전" 거래를 형성할 것이다. 일단 사용자가 상기 거래를 형성하면, 사용자는 이전에 설명된 바와 같이, 요망되는 경우 상기 거래를 테스트하고 실행할 수 있다.
- [0322] 테스트될 때, 사용자가 그러한 정보를 알기를 원한다고 나타내는 경우 단계 S201에서 실시간 시장 환율을 획득하기 위하여, 상기 CDMX 프로그램 모듈 및 상기 MaQs 프로그램 모듈을 사용하여 국가 또는 상대 당사자 데이터를 조회할 것이고, 사용자가 상기 거래를 수행하고 싶은지에 대한 결정을 할 수 있도록 사용자에게 상기 정보를 제공하는 것이 이용가능하다.
- [0323] 통화 환율이 사용자에게 의해 요망되는 것이 아닌 경우(즉, 영국 파운드의 가격이 너무 높은 경우), 상기 통화가 어떤 환전값에 도달될 때까지 상기 프로그램이 계획된 간격으로 시장을 계속 조회할 수 있도록 사용자는 상기 거래를 프로그램할 수 있을 것이거나, 사용자는 다른 단순 거래에 대해 이전에 설명된 바와 같이, 거래가 어떤 날 등에 일어나도록 프로그램할 수 있다. 상기 계획된 거래는, 사용자에게 의해 즉시(즉 실시간으로) 수행되도록 표시되지 않는 한 또는 고정된 유형인 이벤트 트리거에 의해 디폴트로 상기 프로그램에 의해 수동-필(passive-fill) 또는 수동 유형 거래로 지정된다.
- [0324] 사용자가 단계 S202에서 실시간으로 상기 거래를 수행하기로 결정하는 경우, 상기 프로그램은 상기 시스템을 통해 거래를 추적하기 위하여 단계 S203에서 유일한 거래 번호(즉 모 코드)를 코드스탬프하거나 할당하도록 상기 CDMX 프로그램 모듈을 시동시킬 것이다. 상기 프로그램은 상기 CDMX 프로그램 모듈을 사용하여 암호화된 형태로 상기 거래에 관한 기록된 정보를 보존할 것이고, 확인 데이터는 사용자 인터페이스를 통해 사용자에게 상기 정보를 제공하는, 단계 S204에서 상기 프로그램에 의해 상기 VfT 모듈로 다시 전달된다. 실시간 지정 거래에 있어서, 요구의 실행시 하지만 요구를 충족한 이후에(즉 하기에 설명되는 바와 같이 정합 코드들의 할당 또는 상대 당사자들의 정합), 단계 S205에서 사용자의 계정들로부터 자금이 빼내지지 않을 것이어서, 관련된 계정(들)에 대한 수행 명령들은 즉시 송신될 수 있다(즉, 명령들은 정합-쌍이 결정될 때까지 가능하지 않다).
- [0325] 일단 상기 프로그램이 요구를 코드스탬프하면, 상기 프로그램은 상기 거래를 수행하기 위하여 상기 MaQs 프로그램 모듈 및 상기 SVR 프로그램 모듈을 시동시킨다. 일단 상기 프로그램의 상기 MaQs 및 상기 SVR 모듈들이 시동되면, 연산 수단 또는 알고리즘적으로 계산된 분리자와 같은, 중요한 알고리즘들 중 하나를 사용하여, 상기 SVR 프로그램 모듈은 단계 S206에서 원하는 거래를 달성하기 위하여 상대 당사자들을 정합시킬 것이다.
- [0326] 특히, 거래 벡터들은 방향 및 금액 유형(예를 들어 통화 쌍)에 따라 분리되고 값에 의해 랭크되어 있다. 그다음 상기 SVR 프로그램 모듈은 이전에 설명된 가변 알고리즘들에 따라 교차-정합으로 진행한다.
- [0327] 이전에 설명된 바와 같이, 금액이 능동-탐색으로서 상기 프로그램에 의해 원래 분류되는 경우, 상기 프로그램은 단일 상대 당사자 정합 금액으로 상기 요구를 충족하려고 할 것이다. 더욱이, 금액이 수동-탐색으로서 상기 프로그램에 의해 처음에 분류되는 경우, 그것은 평균 아래로 떨어질때까지 그것은 능동-탐색 금액 만큼 감소될 것이고, 능동-탐색 금액 자체가 되도록 재분류되며, 그 후 그것은 채워지고 상기 거래는 완료된다.
- [0328] 따라서, 달러를 영국 파운드로 환전하기 위한 요구와 영국 파운드를 달러로 환전하기 위한 다른 사용자에게 의한 요구의 정합은 몇몇 알고리즘들 중 적어도 하나를 사용하여 정합될 수 있다. 환전을 위해 상기 프로그램이 어떤 알고리즘을 사용하는지 (그리고 상기 알고리즘이 프로그램 가능한지) 상관없이, 상기 SVR 프로그램 모듈은 항상 상대 당사자 정합-쌍들을 생성할 것이다.

- [0329] 상기 프로그램은 사용자에게 의한 영국 파운드에 대한 요구를 영국 파운드를 보유하고 있는 다른 사용자에게 의한 미국 달러에 대한 정합 요구와 정합시킬 것이다. 다른 쪽보다 한쪽에서 더 큰 금액이 존재하고, 정합이 행해진 후 어느 한쪽에 남아있는 영이 아닌 나머지가 존재하는 경우, 상기 프로그램은 상기 시스템을 통해 다른 패스(pass)로 상기 나머지를 정합시킨다. 예를 들어, 더 큰 거래가 상기 시스템을 통해 전달될 때, 상기 SVR 프로그램 모듈은 상기 거래 금액을 다수의 단편들로 쪼개고, 상기 시스템을 통한 연속적인 거래 패스들에서 상이한 상대 당사자들에 대해 각각의 것을 정합시킬 것이다.
- [0330] 상기 CDMX 프로그램 모듈은 단계 S207에서 상기 상대 당사자 정합 결정(백터-스프레딩) 프로세스를 통해 개별 당사자들을 코드짜이클 및 추적하도록(상기 정합 쌍들을 태그하도록) 그리고 상기 정합 프로세스를 통해 상기 거래의 상이하거나 연속적인 단편들을 추적하기 위하여 몇몇 부가적인 차일드(child)-코드들을 생성하도록 시동될 것이고, 모든 나머지 금액들이 정합되거나 결정될 때까지, 상기 단편들을 모 거래(parent transaction)에 대해 추적가능하게 만들 것이다. 상기 코드들 각각은 상기 CDMX 프로그램 모듈에 의해 회계 추적 원장에 기록되고, 단계 S208에서 (하기에 더 설명되는) 코드정합 수집자에서 축적되도록 송신된다. 상기 정합들이 형성된 후, 상기 CDMX 프로그램 모듈은 지불 실행전에 상기 거래 단편들을 다시 모을 것이어서, 최종적으로 원하는 거래가 달성된다.
- [0331] 상기 시스템은 엄격하고 연속적인 방식으로 각각의 거래를 완료하는 것이 아니라, 비록 현재의 거래가 아직 수행 중에 있는 동안이라도, 연속 거래 요구들을 혼합 상태에 있게 하거나, 또는 상기 거래들을 서로 뒤엎히게 한다는 점에 유념하기 바란다.
- [0332] 상기 SVR 프로그램 모듈이 다수의 거래들을 동시에 처리하고 있기 때문에, 정합된 쌍들은 다수의 대응 당사자들을 지닌다.
- [0333] 따라서, 상기 프로그램이 사용자의 계정으로부터, 총액이 기존의 금융 시스템 기반 구조를 통해 수행된 완전한 본래 의도된 거래와 동일한, 적합한 사용자-수신 계정으로 자금 대체 명령들을 수행하도록 MaQs 프로그램 모듈을 호출하기 전에, 상기 프로그램은 단계(S209)에서 수집(동일 사용자에게 의해 거래된 금액의 집합), 획득(금액의 획득)을 위해 CDMX 프로그램 모듈의 코드 정합 수집자(Codematch Aggregator) 기능을 호출한다. 따라서, 코드 정합 수집자 프로그램이 프로그램된 설정값에 따라, 은행 다중화기(Bank Multiplexer) 프로그램의 활성화를 대기시킬 수 있기 때문에, 단계(S210)에서 사용자 계정에 대한 자금의 대체가 일괄 처리될 수 있다.
- [0334] SDEX (실시간 즉시 대체) 거래들에서, 대응 당사자 정합들이 이미 수행된 이후에 명령들이 사용자 계정들에 대해 수행된다. CDEX (고정된 수동 채움) 거래들에서, 사용자 계정들은, 정확한 사용자 표시 거래가 이루어질 때까지, 대응 당사자 정합들이 상기 SVR 프로그램 모듈에 의해 결정되는 시간 동안과 그 이후에 유지를 위해 초기에 시스템 소유 계정(이하 투자 엔진(Investment Engine) 참조)에 대해 조회하게 될 수 있다. 상기 거래들이 통화 교환을 포함하고, 이같은 거래들의 소유자들에 의해 위험한 것으로서 명시되지 않은 경우에, 사용자의 사전 스케줄 거래가 만료되어 자금을 요구할 때까지 상기 시스템은 자기 자신의 프로그램된 제량으로 기본 통화에서의 유지를 위해 자금을 즉시 인출하거나, 또는 타겟 통화에서 교환 및 유지할 수 있다.
- [0335] 전형적인 단순 거래(Traditional Simple transaction)에 대해 언급된 바와 같이, 상기 프로그램은 사용자에게 상기 거래의 완료, 자신의 계정에 대한 상기 타겟 통화의 예금, 그리고 자기 자신의 계정으로부터의 임의의 수수료를 더한 자신의 통화에서의 해당하는 금액의 지불을, 자신의 로그인 계정 내의 메시지들, 전자 우편, 또는 전화, 셀 폰, 팩시밀리, 또는 기타의 원하는 매체를 통해, 통지한다. 상기 거래에 대해 지불되는 수수료는, 이자로서, 상기 프로그램에 의해 상기 시스템 소유 계정에 전달된다.
- [0336] 예 2
- [0337] 도 21a에서 본 발명에 따른 제2 예로 도시된 바와 같이, 기업 A의 -44 금액은 활성 탐색자(active seeker)이며, 이러한 프로그램이 도 21b의 기업 A의 +120과 교차 정합하여 44 제로-섬 쌍을 형성하고, 기업 A의 +76(도 21a 및 도 21c 참조)의 시스템 나머지를 남긴다.
- [0338] 상기 프로그램은 기관 B의 +32를 활성 탐색하여 32 제로-섬 쌍(도 21a 참조)을 형성하여, 기관 B의 -88을 상기 시스템에 남기고, 이러한 프로그램이 활성 탐색자(Bob +12)와 즉시 정합하여, 12 제로-섬 쌍을 형성하고 상기 시스템에 기관 -76을 남긴다(도 21c 참조). 이러한 프로그램은, 기업 A의 +76 및 기관 B의 -76에 대해, 상기 기간 동안 최종 제로-섬 쌍을 형성하여, (전 영역에 대해 제로-섬을 확인하도록) 상기 시스템을 비게 한다(도 21c 참조).

- [0339] 따라서, 매크로 장부(Macro Ledger) 제로-섬 쌍들은 목록으로 도 21a 및 도 21c에 기재되어 있으며, 제로-섬 쌍이 양측 상에 동일한 대응 당사자를 지닐 경우, 이러한 제로-섬 쌍은 상기 프로그램에 의해 제거되어 도 21c의 경험적으로 감소된 매크로-장부를 만들어 낼 수 있다. 도 21c의 경험적 매크로 장부는 가장 효율적인 자본 벡터들(금액들 및 방향들)을 표시한다. 획득과는 무관하게 모든 금액이 대체된 경우, 164개의 자본 단위가 이동된다. 단일의 당사자 획득이 사용될 경우, 100 개의 자본 단위(기관 B에서 기업 A까지의 88개와 기업 A에서 Bob까지의 12개)가 이동된다. 스프레드 벡터(Spread-Vector) 획득이 사용될 경우, (하기의 경험적 매크로 장부에서 표시된 바와 같이 기관 B는 Bob으로 12개를 이동시키고 기업 A로 76개를 이동시킴으로써) 단지 88개의 단위가 이동된다. 이러한 최소의 자본 이동은 여전히 (기업 A, 기관 B, 및 Bob에 대한 개별 장부들 각각의 값들을 합산함으로써) 전 영역에서 각각의 당사자에 대한 순수 장부 계정을 만족시킨다.
- [0340] 예 3
- [0341] 본 발명에 따른 다른 한 예에 있어서, 대규모로, 단일의 사용자가 영국 파운드로 자금을 환전하는 대신에, 미국과 같은 나라에서는, 예를 들면, 제네바 은행 계정에 예금되도록 유로로 교환되는 미국 US\$4,000,000가 채납 수수료로 지불될 수 있다. 본 발명의 시스템은 이같은 복합 거래를 수행하여 수천 또는 수백만의 알려져 있지 않은 대응 당사자들 간의 완전한 협조적인 정합을 결정하기 위해 US를 단위로 한 유로에 대한 환전 요구에 따라 유럽에서 (유로를 가지고 달러를 찾고 있는) 개인들 및 기업에 적합한다. 본 발명의 시스템의 효능은 엄격하게 말하면 통화화 국한되는 것이 아니라, 대체 무역 도구로서 취급될 수 있는 그 어떤 것에 국한된다.
- [0342] 상기 시스템이 통화 환전, 또는 임의의 거래의 요구들을 만족시킬 정도로 충분한 거주자 유동성을 지니지 않을 경우, 상기 프로그램은 시장을 자동 조사하고 전형적인 은행들에 환전 요구들을 전달하여 상기 요구들을 충족시킬 수 있게 한다. 이러한 자본 유입이, 결과적으로 생기는 임의의 종래 시장 수수료의 빠져 나감을 제외하고는, 사용자에게 투명하다. 유동성 위기의 경우에, 환전 요구들이 만족되지 않으며, 마찬가지로 일반 시장 또한 어느 누구도 타겟 통화를 공급하려고 하지 않는다고 생각할 수 있다.
- [0343] 다시 말하면, 본 명세서에서 언급된 시스템은 유동성 위기를 회피할 수 없지만, 상기 프로그램은 매일 내용 감시 등이 제공될 경우 그같은 일들의 조기 경고를 줄 수 있고, 상기 거래들이 프로그램 및 스케줄(수동 채움)될 수 있다는 사실에 기인하여, 상기 시스템이 완화 유동성 기간 동안 사전에 스케줄된 수동 채움들을 행하도록 프로그램될 수 있기 때문에 어느 정도의 완화가 초래될 수 있다.
- [0344] 더욱이, (하기에 언급되는) 분산형 자금 대체(Distributed Funds Transfer; DFT)가 이러한 시스템에 의해 허용된다는 사실에 기인하여, 상기 시스템이 자본의 자유로운 흐름에 관한 항력(drag)을 생성하는) 자본 비행의 정책으로부터 저항하는 규제들을 우회할 수 있기 때문에, 그리고 개인용 시스템이, 한 나라로부터 자본을 수출하려고 시도하는 사람으로부터 생기고, 규제적 제약 등의 이유가 아니라, 미래의 자본 수출자의 기본 통화를 요구하는 당사자들이 없다는 이유로 상기 시스템의 DFT 기능을 사용할 수 없는 초기의 국부화된 유동성 위기를 보다 명확하게 통지할 수 있기 때문에, 대중 시장의 세계적인 유동성 위기의 조기 경고가 초래될 수 있다.
- [0345] 예 4
- [0346] 본 발명에 따른 협조적인 복합 거래의 다른 한 예에 있어서, 자금의 환전은 서로 보완적인 자본 관리 요구들을 지니는, 서로 자기 선택하거나 또는 무작위로 정합되는 2개의 회사들 간에 이루어질 수 있다. 예를 들면, 미국 회사가 중국에서 번 RMB 이윤을 (US 달러로서) US에 다시 보내기를 원할 경우, 중국으로부터 자금을 대체하는 것에 관한 제약이 있을 가능성이 있다. 그리고, 미국 회사가 중국에서의 신규 투자를 위해 RMB로 환전되는 US 달러의 대리자를 지니기를 원할 경우, 신규 자본을 받아들이는 것에 관한 제약이 있을 가능성이 있다.
- [0347] 분산형 자금 대체 공정을 수행하는 SVR 프로그램 모듈은, 타겟 은행 계정의 적합한 설정값들이 제공될 경우에, 하나가 미국 및 중국 각각에서 이루어지는, 국내 회사의 계정으로부터 상대측 회사의 계정들 각각으로의 2가지 국내 대체로 구성된 벡터 스케줄을 간단히 결정할 수 있다. 액면가로, 이는 2개의 독립적인 비관련 국내 자금 이동인 것처럼 보이지만, 매크로 이벤트로서 분산형 시스템을 통해 경화를 교환하는 2개의 회사가 된다. 이러한 방식으로, 상기 2개의 회사는 각각 자신들의 목적들을 달성하여 국경선을 넘지만, 국제적인 자본 이동에 관한 어려움들, 통지, 비용 및 가능한 제약 없이 자금을 이동시킬 수 있다.
- [0348] 즉시 요구 일자가 만족될 수 없거나 또는 자금 액수가 어떤 이유로 채워져 있는 경우, 상기 프로그램은 시스템 제안 일자까지 기다리거나 또는 차후의 거래를 삭제할 것을 사용자에게 통지한다. 일단 상기 거래가 수행될 수 있는 경우에, 상기 거래 자금은 양 당사자들의 은행 계정으로부터 인출되고, 상기 프로그램은 상기 요구들을 충족시켜 상기 타겟 통화의 금액을 사용자의 은행 계정에 송금한다.

- [0349] 이자 및 자선에 대한 투자 엔진
- [0350] 본 발명에 따른 한 실시예에 있어서, 사용자의 방향에서, 즉, 수동 채움 요구 형태들에 대해, CDMX 및 MaQs 프로그램 모듈들을 통한 프로그램은 위험하지 않거나 또는 위험 정도가 낮은 투자를 일시적으로 유지하는 데 사용되는 자금으로, 회전문 유지 계정을 통해 분산된 자본 흐름들을 모은다.
- [0351] 위에서 언급된 바와 같이, 수동 거래 형태는 디폴트 형태이고 사용자가 특정 일자까지 상기 거래가 이루어져야 한다는 것을 나타내는 것을 특징으로 한다. 이것이 의미하는 것은 그 일자까지의 시간 완료가 상기 거래를 개시 및 수행하고 자본을 사용하도록 허용될 수 있는 일자로서 해석될 수 있다는 것을 의미한다. 따라서, 상기 시스템은 수동 채움 거래의 자금을 끌어 모우고, 이러한 자금을 단기간 투자를 위한 계정에 유지할 수 있다. 더욱이, 통화 환전과 같은 특정의 즉시 거래들을 위해, 상기 시스템은 할증금 없이 청구할 수 있다.
- [0352] 특히, 협조적인 복합 거래로 위에서 언급된 바와 같이, 예를 들면, 사용자가 수동 요구 통화 환전 거래를 개시할 경우, 상기 프로그램(즉, MaQs 프로그램 모듈)은 사용자의 자금을 사용자 계정으로부터 즉시 빼내고 상기 시스템 내의 보유 계정에 상기 자금을 이동시킬 수 있으며, 상기 자금은, 다른 일시 보유 계정들과 함께, 비록 소유자들이 자신들의 투자 승인을 제공했다 하더라도, (상기 자금이 소유자들에게 반환되거나 소유자들이 완료하기 시작한 거래들을 끝내는 방식에 따라 송금되기 때문에) 위험이 낮거나 없는, 전형적인 금융 기관 등에서의 금융 시장 거래 계정, 또는 보통 예금 계정에 투자될 수 있다.
- [0353] 상기 프로그램은 자동적으로 금융 시장 데이터를 끌어 모으고 어떤 유형의 금융 수단이 필요한 위험 매개변수 내에서 이용될 수 있는 지를 검사하여, 각각의 선택에 대한 자본의 기회 비용(즉, 진입, 탈출, 및 자본 요구들의 비용)을 결정한다. 그후, 상기 프로그램은 상기 시스템 내로/로부터의 자본 벡터 흐름 및 탈출 비율과 아울러, 코드 정합 시스템 내로의 자본 벡터의 비율을 검사하고, 필요한 시스템 유동성 레벨에 대한 가장 양호한 반환을 제공하는 "최적"의 자본 연습 옵션을 결정한다(즉, 자본 비율은 은행 다중화 기능에 의해 축소되고, 상기 시스템은 코드 정합 수집자 기능에 의해 요구되는 바와 같은 비율을 원한다).
- [0354] 제로 또는 거의 제로인 위험 옵션들의 유형들 중에는 개인적인 규모로 매우 작은 반환들이 만들어지지만, 예를 들면, 일년에 걸쳐 수백만 그리고 수백만번 곱해질 경우 발생한 수입의 상당 금액을 모으면, 하룻밤사이의 이자 수입; AAA 등급의 담보부 여신 포트폴리오의 구매; U.S.의 구매로 상기 시스템에 의해 은행 계정 내에 모여진 자금들이 보유된다. 레버리지(leverage)를 지닌 보물들, 이들 팔아 현금화시키고, 상기 현금을 이자 수입을 위해 은행 계정에 넣어 두며, 레버리지를 지닌 미국 보물들을 구입하고, 이들을 팔아 현금화시키며, 상기 현금을 AAA 등급 회사에 빌려주고, 레버리지를 지닌 미국 보물들을 사서, 연계 매매 자금(Hedge Fund)을 하룻밤 사이나 또는 그 이상 빌려준다.
- [0355] 대응 당사자 정합이 이루어지도록 스케줄될 경우에나, 또는 기일들이 계속 만기될 예정이어서 실행 명령들이 곧 이같은 유용 자본을 요구할 경우, 이러한 프로그램은 정합들이 이루어지기를 기다린 다음에, 상기 프로그램은, 본래의 사용자의 계정이 아니라 상기 시스템에 의해 소유된 소스 계정을 형성함에도 불구하고, 자금을, 이 자금이 갈 필요가 있는 곳에 송금한다. 개인 계정들은 이러한 투자 지령을 생성하는 데 사용되거나 또는 상기 시스템을 통해 흐르는 자본의 수집의 1퍼센트가 사용될 수 있다. 사용자는 자금이 투자를 위해 사용된다는 것을 알 수도 있고 모를 수도 있다.
- [0356] 상기 시스템이, 이전에 유용된 자금에 의해 이루어진 투자를 적소에 푸는 대신에, 사용자 거래를 완료하기 위해 자본 반환을 필요로 할 경우, 상기 프로그램은 상기 사용자 거래를 충족시키기 위해 보다 신규한 수입 자본을 유용한다. 상기 투자가 상기 시스템에 반환될 경우, 상기 투자는 또한 거래들을 수행하기 위한 신규 자본으로서 이용될 수 있다.
- [0357] 상기 투자에 의해 생기게 되는 이자는, 사용자가 지시하는 경우, 상기 프로그램에 의해 상기 사용자에게 반환될 수도 있고(즉, 사용자는 차후에 이루어질 거래를 설정하여, 상기 자금이 즉시 빼내어지고 상기 거래들이 이루어질 때까지 금융 시장 거래 계정에 넣어지게 하는 권한을 부여할 수 있으며, 상기 이자는 개시 거래 금액 이동이 앞으로 수행되는 동안 사용자에게 반환되고), 상기 이자는 사용자에게 의해 아마도 모여져서 자선을 위해 사용되도록 배당될 수도 있다. 시스템 이자는 또한 자동적으로, 상기 시스템의 갱신에 대한 자본 구입을 위한 자금, 또는 차후의 자선을 위한 자금을 모으도록 재투자될 수 있다.
- [0358] 상기 프로그램이 모든 것을 추적 및 수행하기 때문에, 투자는 무한정으로 유지될 수 있으며, 신규 수입 분산형 소스 자금이 이전의 소스 구입을 위해 지불하도록 상기 프로그램에 의해 편향되고, 이러한 자금의 편향(이동)은 시스템 위험을 증가시키지 않는 데, 그 이유는 1) 대부분의 구입이 신용 카드로 온라인을 통해 이루어지며, 2)

본 발명의 시스템이 일방향 계정 내로의 유용한 자금의 사전 할당을 용이하게 필요로 할 수 있기 때문이다.

[0359] 일단 분산형 소스 자금이 자선의 이유로 이자를 생성하는 데 사용된다면, 미립 감세(particulate tax-credit)는 자동적으로 상기 프로그램에 의해 각각의 소스에 다시 지급되며, 또한 상기 프로그램은 각각의 미립 금액과 같은 할당표들을 부여할 수 있다. 할당표들은, 직업 자선 수출에 의해 편성된 이용가능한 옵션들을 통해, 자선 기금이 어떠한 방식으로 사용되어야 하는 지에 있어서 각각의 참가자가 민주적으로 말할 수 있게 한다. 또한, 본 발명의 시스템은 사용자가 상기 시스템을 사용할 수 있게 한 자본에 대한 이점으로서 사용자에게 다시 그의 자선 활동으로부터, 레코드 로그들로부터의 가중된 평균을 기초로 하여 감세를 지급할 수 있다.

[0360] 따라서, 투자 활동 무대에서의 다른 금융 서비스는 사용자에게 제공될 수 있다. 또한, 사용자가 투자 기회를 이용함에 있어서 자기 자신의 자금을 사용하기를 원하지 않을 경우, 사용자는 아마도 투자 기회들에 참여하도록 금융 기관으로부터 자금을 차용할 수 있다.

[0361] 자본 유입

[0362] 본 발명에 따른 한 실시예에 있어서, 위에서 언급된 바와 같이, MaQs 프로그램 모듈은 상기 시스템의 적합한 기능을 보다 용이하게 하도록 충분한 유동성이 존재한다는 것을 확실하게 하도록 상기 시스템을 주기적으로 자동 검사한다. 만약 상기 시스템이 (산술 평균 따위나 또는 10퍼센트 이상의 변동, 예를 들면, 오프사이드(Oppside)에 따른 변화를 기초로 하여) 심하게 비대칭인 경우, 상기 프로그램은 신규 사업 소스에 관심이 있을 수 있는 JP 모르건과 같은 전형적인 은행 채널에 대해 자신을 열어둠으로써 신규 자본을 자동적으로 사용하거나, 또는 상기 시스템의 소유자가 자기 자신의 자본의 일부를 위탁하기를 원할 경우, 시스템 이자는 사용(상기 사업의 운영에 투자)될 수 있다.

[0363] 이러한 자본 유입은 사용자에게 투명한 것이며, 코드 정합은 계속 수행될 때까지 단순히 상기 시스템을 임의의 부정합 요구들이 통과함에 따라 멈추지 않게 계속 이루어진다.

[0364] 자본 운영

[0365] 본 발명에 따른 다른 한 실시예에 있어서, 위의 운영들에서 발생된 모여진 자금 또는 이자에 대한 자선의 동기 부여에 부가하여, 상기 시스템은 자체적으로 총체적인 시스템이 될 수 있는 보다 복잡한 기능들을 포함한다.

[0366] 예를 들면, 상기 프로그램은 현재 은행들이 제공하고 있는 것과 마찬가지로(이 경우, 상기 시스템이 사용자의 계정 내에 속박되어 있기 때문에, 상기 시스템은 사용자가 신용 카드 또는 신용 한도액 또는 대출을 요구할 경우 사용자에게 관한 신용 등급을 용이하게 수행할 수 있음), 시스템 사용자, 그룹, 소기업 또는 심지어는 대기업에 대한 신용을 연장하여 신용 및/또는 ATM 카드들을 발행하는 신용 서비스 운영을 제공할 수 있거나, 또는 신용 등급은 이하 설명되는 바와 같이 분산 형태를 기초로 하여 생성될 수 있다. 이러한 신용 등급은 에퀴팩스(Equifax) 등과 같은 제3자 신용에 의존하는 것보다는 오히려, 실제 지불의 CDMX 프로그램 추적 기능과 연관지어 생성될 수 있다.

[0367] 사용자들이 시스템에 의해 결정된 신용 등급에 따라 개시되게 하기 위해, 신용 등급의 진전은 EbayTM의 진전 다음에 이어질 수 있음으로써, 계정 사용이 사용자 등급을 전달함에 있어서 주된 요인이다. 그러나, 사용자 신용 등급에 대한 기초는 또한, 각각의 사용자에게 대한 비제로 개시 등급에 도달하도록 기존의 신용 점수 입력을 취하는 특정의 수학적식을 기초로 하는 다수의 다른 방식 이전에 이루어질 수 있다. 개시 신용 점수 입력은, 정식 신용국 점수, 또는 사용자가 그후 협조적인 환경에 참여하기 시작하는 데 사용할 수 있는 개시 신용 등급인 일정 범위 또는 특정 번호에 도달하도록 특정의 알고리즘을 통해 상기 프로그램에 의해 조작되는 사용자의 신용 카드 사용의 제출 레코드(들)를 기초로 한 적합한 것일 수 있다.

[0368] 융자 및 소액 융자

[0369] 본 발명에 따른 다른 한 실시예에 있어서, 상기 시스템은 (개인이든 법인이든) 시스템의 사용자들이 다른 시스템 사용자들로부터와 아울러, 상기 시스템이 그러한 목적으로 활성화된 계정들을 지녀야 할 경우 시스템 자체로부터 평가받아 융자/차용할 수 있게 하는 직접 융자 시스템을 제공할 수 있다.

[0370] 상기 직접 융자 시스템에 있어서, 사용자는 한정된 이자율 및 지불 스케줄로 (예를 들면, 아프리카에서 사용될 수 있는) 자금을 다른 원격 사용자들에게 제공할 수 있으며, 차용인들은 특정의 이자율 및 지불 스케줄로 자금을 차용할 수 있고, 차용인들/채권자들은 서로를 알 필요가 없는 데, 그 이유는 상기 프로그램이 스케줄된 시간에 자동적으로 상기 차용인들의 계정들에서 출금하고 채권자들의 계정들에 입금하고 심지어는 대출 상환을 본국

에 돌려 보내도록 통화 환전을 수행하기 때문이다.

[0371] 다른 한 예는 소액 용자이다. 그라민 은행 프로젝트(Grameen Bank Project)에 의해 구성된 소액 용자 모델은 오프라인임에도 불구하고 소규모 및 국소 규모로 동등 계층 네트워크 효과를 사용한다.

[0372] 본 명세서에서 언급된 본 발명의 협조적인 기능성에 있어서, 소액 용자는 대규모 국소 규모로 용이해져서, 개인들이 선택한 유성(planet)의 어느 한 모서리로 용이하고 신속하게 자기 자신들의 자본 흐름을 향하게 하는 것을 가능하게 한다. 동일한 네트워크 효과 동등 관계(즉, 자기 선택한 위탁된 개인들)를 지지하는 본 발명의 시스템의 협조적인 객체 모델에 있어서는, 재정 곤란, 위탁, 및 자금 흐름에 대한 전례없는 속도 및 유동성을 인도적인 노력들, 구조 노력들, 구급 노력들 등으로 야기시키는 잠재력이 자기 영역 내에 있다.

[0373] 따라서, 그라민 은행의 경제적 측면은 (비록 자금이 제공자 온라인으로부터 모여진다 하더라도, 기부금의 분배는 정부 조직에 의해 분배되는 오프라인에서 이루어짐)이 소액 오프라인을 처리하는 동안, 본 명세서에서 언급된 시스템은 소액이든 대중적이든, 임의의 금액 수준으로 협조적인 거래들 및 직불을 지원함으로써, 사용자들이 제공하는 차용 또는 투자 기회들을 참가자들(즉, 동료들)이 획득하기 위한 사람 관계, 평판 및 무결성(integrity)에 의존한 다음에, 직접적으로나 또는 한 그룹으로서 각각의 채권자/투자자와 대화할 수 있다.

[0374] 본 발명의 차용 및 용자에 대한 공동의 직접 해결 방안에 있어서, 상기 CDMX 프로그램 모듈은 모든 것을 추적하고, "성공 포인트(success point)"들은 성공적인 신용 거래와 자신들을 정렬하기 위한 모든 참가자들에게 제공됨으로써, 유인책이 될 수 있다. 상기 프로그램에 의해 사용자 인터페이스에서 표시되는 이같은 포인트들의 축적은 분산형 신용 등급 시스템 또는 더 광범위하게는, 분산형 무결성 등급이 됨으로써, 참가자들의 네트워크는 이전의 레코드를 기초로 하여, 제3자의 신용 가치에 관한 평가를 수행하기 시작할 수 있다.

[0375] 이는 사용자들, 회사들 등이 용화된 자금에 대한 보다 높은 이자율(즉, 성공적인 거래에 수반되는 것으로 알려져 있는 높은 신분의 채권자들 또는 투자자들이 할증금을 청구할 수 있는 데, 그 이유는 이들의 거래의 참가자 종종 다른 사람들에 의한 부가적인 참가, 궁극적으로는 거래에 대한 성공을 용이하게 하는 심리적인 요소를 제공하기 때문임)과 보다 낮은 차용율(신뢰성이 있는 차용인들이 최상의 지불 레코드를 지니고, 우수한 신용, 결과적으로는 보다 낮은 차용율을 향유함)을 청구하는 것을 가능하게 한다. 그 외에도, 집단(시스템) 내의 정직하고 신뢰성이 있는 참가자들에 의해 향유되는 무결성 할증금은 또한 보다 낮은 무결성 등급을 지닌 다른 사람들이 접근하지 못하는 투자 등에 대한 진보적이고 선택적인 접근 또는 그에 대한 능력의 제공을 포함할 수 있다.

[0376] 보편적인 온라인 구입

[0377] 본 발명에 따른 다른 한 실시예에 있어서, 전형적인 온라인 구입(즉, 전형적인 단순 거래들)은 사용자가 사용자 계정(앞서 언급된 거래들 중 어느 하나와 같이 설정된 은행, 주식 판매 등)으로부터의 직접 출금, (앞서 언급된 바와 같은) 시스템 대출, 또는 시스템 신용 카드로서 상기 시스템에 지불하는 옵션을 이용할 수 있는 경우에 이루어질 수 있다. 이들 거래들은 실시간으로 또는 프로그램된 시간에 수행될 수 있는 데, 이는 구입함에 있어서의 사용자 융통성을 허용한다.

[0378] 그러나, 몇몇 경우에서, 한 사용자가 예를 들면, 미국에 있으며 외국 통화의 목록에 있는 다른 한 나라의 물품을 사기를 원하는 경우에 통화는 한가지 문제이다. 본 발명을 사용할 경우, 사고자 할 때, 사용자는 상기 물품에 대한 비용을 지불하기 위해 상기 시스템을 이용함으로써, 상기 시스템이 타겟 통화로 상기 물품에 대한 비용을 지불하고 해당하는 미국 통화를 사용자의 계정에서 공제한다.

[0379] 특히, 상인의 웹사이트 상의 계산대 스크린에서 온라인을 중단한 다음에, 사용자는 지불 옵션으로서 "상기 시스템에의 지불(Pay with The system)"을 선택할 수 있다. 시스템 스크린은 선택되고, 상기 프로그램은 사용자에게 자신의 로그인 및 패스워드(단지 온라인 구입에 대한 특정의 패스워드가 또한 설정될 수 있음)를 요구한다. 사용자는 거래 워크패드 스크린(Transaction Workpad screen)에서 자신의 구입을 확인할 수 있으며, 시스템 상인 식별 코드, 및 (구입되는 물품, 또는 구입 순번에 대한 식별 코드와 같은) 기타의 식별 코드의 사용은 위에서 언급된 협조적인 복합 거래에서와 같이 통화를 환전할 수 있다. CDMX 및 MaQs 프로그램 모듈들은 구입을 추적하여 거래를 수행하고, 타겟 통화 자금을 상인 계정으로 대체하며, 사용자의 계정에서 자금을 출금한다. 상기 프로그램은 상기 자금이 제공되었고 상기 거래가 완료되었다는 것을 상기 상인에게 통지하고 또한 사용자에게 자신의 계정으로부터 상기 자금이 출금되었다는 것을 통지한다.

[0380] ATM 분단

[0381] 본 발명에 따른 다른 한 실시예에 있어서, 정량화된 접근(Quantified Access)이라 지칭되는 개념을 사용하면,

비-고유 ATM 카드는 임의의 은행 영역에서 임의의 ATM 기계를 사용하고 플러스(Plus) 또는 시러스(Cirrus) 또는 (비용을 초래하는) 다른 유사한 네트워크들을 이용할 수 없다.

[0382] 정량화된 접근이 의미하는 것을 자금을 인출하는 ATM 카드를 소유하지 않거나 상기 ATM 카드와 연관 또는 연계되어 있지 않고 또한 어떠한 방식으로든 상기 ATM 카드와 연관되지 않은 은행 계정에서 자금을 공제하기 위한 접근을 일시적으로 그리고 정량적으로 부여하는 것을 의미한다. 상기 ATM 카드에 부여된 승인은 임의의 은행 또는 임의의 은행의 ATM이 상기 ATM 카드를 고유의 ATM 카드로서 인증하는 그러한 것이다.

[0383] 이는 사용자 승인 없이 진행할 수 있다(즉, 즉시 인출되는 자금의 일시적인 삽입에 대한 사용자들의 계정들의 순간 사용, 결과적으로는 상기 계정을 본래 상태로 유지한다고 생각할 수 있다. 그러나, 규제적 문제들은 상기 시스템 사용자들이 이러한 목적으로 자신들의 계정들을 사용하기 위한 승인을 나타낼 것을 필요로 할 수 있다.

[0384] 그러한 공정은 다음과 같다. 사용자는 자신의 ATM 카드가 수수료 없이 사용할 수 있는 영역에 고유하지 않은 ATM 기계로 간다. 이는 또한 임의의 통화 영역에 있을 수 있다. 그러나, 이는 중요하지 않은 데, 상기 시스템이 필요에 따라 환전 거래를 수행할 수 있기 때문이며, 이러한 공정은 이하에서 상세하게 설명될 것이다.

[0385] 사용자는 카드를 삽입하여, 상기 카드가 상기 ATM 컴퓨터에 의해 유효한 카드로서 확인을 받는 데, 이 경우 상기 컴퓨터는 주어진 ATM 카드가 전세계의 실제 은행에서의 유효한 은행 계정과 연결되는 것의 확인을 위해 검사한다.

[0386] 일단 확인을 받을 경우, 고유 은행 사용자에게 대한 무료 거래가 아닌 경우 사용자는 대개 상기 거래에 대해 \$1.50 또는 특정의 유사 금액이 청구된다는 것을 나타내는 통지를 스크린을 통해 받는다.

[0387] 인튜이트(Intuit)의 퀵큰(QuickenTM) 및 최종 소비자들에 대한 다른 회계 소프트웨어와 같은 소프트웨어 프로그램이 네트워크를 통해 이미 은행 계정 데이터를 다운로드하는 것이 가능하기 때문에, 이러한 계정 감시가 실시간으로 수행될 수 있다고 생각할 수 있다. 따라서, 상기 시스템은 내부에 등록된 계정들 상의 활성도를 추적하기 위해 사용자들에 의해 설정될 수 있다. 사용자들이 이러한 방식으로 응답하도록 자신의 계정들을 설정한 경우, 사용자가 ATM 스크린 상의 ATM 사용에 대한 잠재적인 수수료를 승인하자마자, 시스템 프로그램은 ATM 인출에 대한 미결정의 은행 공제의 통지를 받고, 상기 은행들이 수행하는 것을 허용하지 않은 거래를 인계받는다.

[0388] 먼저, ATM에서의 사용자 세션을 삭제하지 않은 경우, 상기 프로그램은 은행에서의 수행을 중지하여, 가정용 은행(home bank)에서의 공제를 허용하지도 않고 ATM으로부터의 현금 분산을 허용하지도 않는다.

[0389] 그후, 상기 프로그램은 은행 다중화기 프로그램 모듈을 사용하는 사용자의 계정으로부터 상기 ATM 영역 내의 임의의 계정 X의 임의의 소유자의 의도가 주어진 경우에 이루어질 수 있는 것과 유사한 현금 거래용으로 사용되는 그러한 영역 내의 그러한 계정 X로 자금을 이동시켜서, 사용자가 인출하기 위한 정량화된 접근으로 자금을 융자한다. 따라서, CDMX 프로그램 모듈은 추적 로그들을 갱신한다. 그와 동시에, ATM을 소유하고 있는 은행에는 상기 ATM에서 판독된 비-고유 ATM 카드가 계정 X에 대한 고유 레벨의 정량화된 접근을 지닌다는 것을 나타내는 정보가 상기 프로그램에 의해 전송된다.

[0390] 자금은 즉시 계정 X로부터 분산되어 그의 균형을 본래의 상태로 유지하고(로그들은 동일 금액의 동시적 예금 및 인출을 보여줌), 어떠한 수수료도 청구되지 않는 데, 그 이유는 은행이 이를 고유의 ATM 카드 인출로서 승인하고 있기 때문이다.

[0391] 따라서, 정량화된 접근이 의미하는 것은 은행이 ATM을 고유 계정으로서 승인하도록 사용자가 대개 접근하고 있지 않은 계정으로부터 특정의 정의되고 제한된 자금 금액을 공제하도록 상기 프로그램에 의해 제공된다는 것을 의미한다. 다시 말하면, ATM 카드는 일시적이고 정량화된 금액만에 대해, 계정 X에 연계된 ATM 카드로서 승인된다.

[0392] 현재, ATM에 분배되는 현금이 사용자 계정들로부터 실제 물리적으로 얻어지지 않고 단지 전자적으로 공제되기 때문에, 사실상 상기 시스템으로 하여금 X 계정내로 자금을 이동하게 할 필요가 없을 수 있지만, 은행에서의 임의의 일반 계정은 산정되어야 하는 데, 이것이 의미하는 것은 상기 시스템이 임의의 그리고 모든 은행 영역 내의 은행 계정을 열어 유지할 경우, 단순히 자기 자신의 은행 계정을 사용하고, 그러한 방식에 대한 정량화된 접근을 부여하여, 모든 시스템 사용자에게 분담 계정을 본질적으로 제공함으로써, 전세계 어느곳이든, 어느 한 그리고 모든 영역에서 ATM 수수료를 회피할 수 있다.

[0393] 분산형 자금 대체

- [0394] 본 발명에 따른 다른 한 실시예에 있어서, 본 발명의 시스템은 임의의 전형적인 자금 교환을 천가지의 서로 다른 방향으로 소액 분산되고, 과도적인 결정을 통해 이동되고 본래의 금액을 합산하도록 집중하는 소액 대체들로서 수신지에 모여지는 분산형 자금 대체로 이루어질 수 있게 한다. SVR 프로그램 모듈은 이를 수행함으로써, 분산 패턴이 검출가능하지 않게 하는 데, 이것이 의미하는 것은 상기 대체들의 외부 감시가 분산 방법을 간파하지 못하기 때문에 상기 시스템에 본래 존재하는 특정 보안 또는 안전을 의미한다.
- [0395] 먼저, 정의를 내리면, 자금 대체, 또는 전자 자금 대체(electronic funds transfer; EFT)는 네트워크를 통해 A 점에서 B점으로 자금의 고의적인 타겟 이동이다.
- [0396] 분산형 자금 대체(distributed funds transfer; DFT)는 자금이 송신자로부터 수신자로 직접 이동하지 않고서도, 본래 의도된 대체 방향에 따라 송신자 및 수신자 각각에서의 획득이 수행되도록 자금 흐름 전역의 매크로 체로-섬 인식이 2개의 선택된 지점들(즉, 송신자 및 수신자)을 향해 그리고 이러한 지점들로부터 떨어진 자금 흐름의 벡터 편향을 허용한다는 취지로, 기존의 자금 대체 네트워크를 사용하는 방법이다.
- [0397] 전세계 상호 연결 네트워크들을 통한 통신 전송 속도들이 순간에 접근하기 때문에, 근접 클라이언트들, 서버들, 및 서벤트(servent)들에게 직면하고 있는 것과 원격 클라이언트들, 서버들, 및 서벤트들에게 가시화될 수 있는 것에 관한 정보를 제공하는 것이 인식 전파를 위해 가능하다. 이같은 속도들에서 동작하는 네트워크에서, "근접"이 "전세계"를 포함할 수 있다는 점에 유념하기 바란다. 본 발명이 자금 대체를 수행하도록 (기존의 규제를 받는) 기존의 기반 구조를 사용하기 때문에, 자금 대체는 외부적으로 예측 및 추적하기가 매우 곤란하다.
- [0398] 이것이 의미하는 것은 (물리적으로나 지리적으로) 원격으로 떨어져 있는 실체 R이 자신에 가까운 특정 지점(A)에서 물리적, 지리적 세계에서 자신으로부터 멀리 떨어져 있는 특정 지점(B)으로 자금을 이동하려는 소망 및 의도에 관한 특정의 정보가 네트워크에 알려지게 하는 순간에, 특정의 실체 S가 자신에 가까운 특정 지점으로부터 물리적 지리적 세계에서 자신으로부터 멀리 떨어져 있는 특정 지점 G로 이동하려는 소망 및 의도를 고려할 수 있다. 오늘날의 세계에서, 자금 대체는 자금 대체 네트워크를 통해 수행되는 다른 어떤 대체로부터 진공 포장된 상태로 각각 독립적으로 추구된다. 단지 (기간의 종료시에 송금될 수 있는 "순수" 금액을 달성하기 위해 고정 시간을 통해 동일 당사자에게 모든 내외 대체를 합산함으로써 두 당사자들 간의 총체적인 자본 이동을 저장하기 위한 계산적 해결 방안인) "획득(netting)"의 경우에서만, 다수의 대체들이 다른 대체들을 참조하여 고려된다.
- [0399] 본 발명의 시스템에 있어서, 시스템 네트워크 내의 각각의 참가인에 대한 위험 정의 가이드라인이 주어진 경우, 상기 프로그램은 인식 전파를 통해 가까워서 먼 그리고 멀리서 가까운 자금 대체 통지를 지켜볼 수 있으며, 모든 참가 클라이언트들, 서버들 및 서벤트들에서의 프리셋 알고리즘을 사용할 경우, 상기 프로그램은, 예를 들면, 모든 자금 이동의 총체적인 합산 거리가 최소화되도록 자금의 방향 변경을 달성한다. 이것이 의미하는 것은, 예를 들면, A→B의 긴 운반 거리, 그리고 또한 F→G의 긴 운반 거리로 자금을 이동시키는 대신에, 2개의 보다 작은 대체들 F→B, 및 A→G가 위험에서의 시간당 총 자본 금액을 감소시키고, 모든 당사자들의 의무를 수행함으로써 전체 시스템 위험을 낮추는 것을 의미할 수 있다.
- [0400] 본 예에 있어서, 총체적인 자본 이동을 감소시키기 위한 A→B, F→G 전환 대응 당사자들의 경우, 이는 단지 2가지의 의도된 대체들이 동일한 경우에만 수행가능하다. 이러한 당사자 (대칭) 상태는 임계 질량에 대한 한계값으로서 유지하는 데, 이는 그러한 시스템이 오늘날 존재하지 않는 이유를 설명한다. 본 발명은 전 영역의 모든 벡터들, 그리고 보다 중요한 점으로는 전 영역의 연속 벡터들을 유지함으로써 이러한 당사자 상태를 해결한다. 자금 대체 네트워크들이 끊임없이 부풀어 오르는 체적에 직면하기 때문에, 연속 벡터들이 올 것이라고 가정하는 것이 상당히 신뢰성이 있다. 본 발명은 이론상 영속하는 것으로 가정되는 벡터 필드, 또는 흐름의 일부인 벡터로서 거래들을 간주하는 대체가능 거래를 관리하는 유일한 시스템이다.
- [0401] 특히, 이하에서 언급되었지만, 상기 프로그램은 먼저 특정의 자금 대체 네트워크를 통해, 자금을 송금하기 위한 사용자의 명령을 받는다. 그후, 시스템 클라이언트/서버/서벤트의 프로그램은 인식 전파를 통해 자기 자신의 기관 또는 그 부근을 타겟으로 하는 의도된 자금 대체를 발견한다. 본 발명의 시스템의 클라이언트/서버/서벤트는 그러한 원격 클라이언트/서버/서벤트에 대한 직접 통신 경로를 식별하고, 통신을 연다. 의도된 대체 벡터들에 관한 2개의 클라이언트들/서버들/서벤트들의 거래 정보는 각각의 타겟에 의도된 순수 벡터 위치를 국부적으로 레코딩한다(한 벡터의 리콜(recall)이 이러한 경우에 금액 정보, 및 방향 정보를 포함하는 2차원 실체이다). (주: 현재 이것이 의미하는 것은 은행 정보가 완전히 과다하게 분산될 수 있으며 대부분 익명으로, 또는 적어도 암호화된 형태로, 금융 기록을 근절 및 파괴하는 것이 사실상 불가능한 정도까지, 자연 채양 또는 모든 사람을 평등하게 하는 이름으로 된 재산을 제거함에 목적을 둔 테러는 현대 사회에 극히 먼 위협이 된다는 것을 의미한다)

다).

- [0402] 두 클라이언트들/서버들/서벤트들은 일정한 정의된 시간에서 임의로 포함한 의도된 벡터들에 대한 최적의 대체 방향들을 반복적으로 계산한다. "반복적으로"이라 함은 현재 분산형 대체 최적화 공정에 포함될 수 있는 (즉, 인식 전파가 새로이 의도된 대체 벡터들에 관한 신규 정보를 모든 참가하는 클라이언트/서버/서벤트에게 가져다 주는) 의도된 대체 벡터들의 변화하는 네트워크 인구와의 끊임없는 대화를 가정한 것임에 유념하기 바란다. "최적"이라 함은 기간 인식 벡터 전 영역의 콘텍스트 및 범위에서 계산된 가장 효율적인 벡터 또는 벡터들로서 정의된다.
- [0403] 따라서, 주기 분산형(또는 분산형) 자금 대체에서, 상기 프로그램이 특정 계정 A에서 특정 의도된 타겟 계정 B로, A에서 B로 직접 자금을 대체하지 않고서도, 자금 전달을 달성할 수 있기 때문에, DFT는 해커들, 테러리스트들, 자연 재앙 등에 의해 대체 라인들이 손상받을 수 있는, 걱정있는 모든 고가의 대체 에이전트들에 대한 믿을 수 없는 도구가 된다.
- [0404] 주기성이 완결될 경우, 최적화된 분산형 대체가 상기 프로그램에 의해 수행된다. 몇몇의 순수 벡터 상태들이 충족되지 않은 상태에 있거나 또는 단지 부분적으로 충족된 상태에 있으며, 연속 주기는 이같은 불충족 및 부분 충족 벡터 상태들의 인식을 포함한다. 상기 프로그램의 여러 프리셋 규칙들은 구식의 불완전한 순수 벡터 상태가 먼저 충족됨에 따라 불완전하고 불충족된 거래들의 각각의 연속파에 대한 완료를 초래함을 유지한다.
- [0405] 따라서, 본 발명의 시스템은 TCP/IP가 인터넷을 통해 통신을 야기시키는 것과 같이 대체가능 명령 전송에 대하여 전달이 달성될 때까지의 동일한 가차없는 경로 변경을 야기시킨다.
- [0406] 보안 및 프라이버시 문제들
- [0407] 거래 벡터들은, 상기 시스템에서, 상기 시스템의 모든 사용자들에 대한 완전한 프라이버시를 제공하면서, 암호화된다. 그러나, 벡터량들, 방향들, 및 식별 정보들은 법적 권한 체제에서 개별적으로 해독될 수 있다. 본 발명의 프로그램은 잠재적인 세탁 형태 또는 탈세 행위들에 대해 시스템 관리자들에게 승인 및 경고하는 필터들을 제공할 수 있다.
- [0408] 여기서 강조되어야 할 점은 위에서 언급된 본 발명의 실시예들은 본 발명의 원리에 대한 명확한 이해를 위해 언급되는 단지 가능한 구현예들일 뿐이라는 것이다. 본 발명의 사상 및 원리들로부터 벗어나지 않고서도 위에서 언급된 본 발명의 실시예들에 대해 변형 및 수정예들이 이루어질 수 있다. 이같은 모든 변형 및 수정예들은 본 발명의 범위에 포함되고 이하의 청구항들에 의해 보호를 받도록 의도된 것이다.
- 도면의 간단한 설명**
- [0022] 도 1은 본 발명의 일 실시예에 따른 클라이언트-서버 환경을 나타내는 개략도이다.
- [0023] 도 2는 본 발명의 일 실시예에 따른 분산형 네트워크 환경을 나타내는 개략도이다.
- [0024] 도 3은 본 발명의 일 실시예에 따른 클라이언트-서버 환경에서 클라이언트 및 서버를 더 상세하게 나타내는 개략도이다.
- [0025] 도 4는 본 발명의 일 실시예에 따라, 금융 거래를 수행하는 스프레드-벡터 해상도 프로그램 모듈에 의한 단일 평균(mean)을 갖는 균형이 이루어진 요구를 나타내는 도면이다.
- [0026] 도 5는 본 발명의 일 실시예에 따라, 금융 거래를 수행하는 스프레드-벡터 해상도 프로그램 모듈에 의한 다중 평균을 갖는 균형이 이루어진 요구를 나타내는 도면이다.
- [0027] 도 6은 본 발명의 일 실시예에 따라, 금융 거래를 수행하는 스프레드-벡터 해상도 프로그램 모듈에 의해 처리되는 비대칭 정합을 나타내는 도면이다.
- [0028] 도 7a는 본 발명의 일 실시예에 따라, 금융 거래를 수행하는 스프레드-벡터 해상도 프로그램 모듈에 의해 처리되는 모든 탐색 반대편 정합을 나타내는 도면이다.
- [0029] 도 7b는 본 발명의 일 실시예에 따라, 스프레드-벡터 해상도 프로그램 모듈에 이용할 수 있는 시스템으로의 자본 유입을 나타내는 도면이다.
- [0030] 도 8a 및 도 8b는 본 발명의 일 실시예에 따른 스프레드-벡터 네팅(Spread-Vector Netting)을 도시한다.
- [0031] 도 9는 본 발명의 일 실시예에 따른 코드 분할 다중 거래 프로그램 모듈의 기능을 도시한다.

- [0032] 도 10은 본 발명의 일 실시예에 따른 코드 분할 다중 거래 프로그램 모듈의 기능을 도시한다.
- [0033] 도 11a 및 도 11b는 본 발명의 일 실시예에 따른 전통적인 간단한 거래의 동작을 나타내는 흐름도이다.
- [0034] 도 12는 본 발명의 일 실시예에 따른 사용자 인터페이스의 거래 작업패드의 화면이다.
- [0035] 도 13은 본 발명의 일 실시예에 따른 사용자 인터페이스의 거래 작업패드의 화면이다.
- [0036] 도 14는 본 발명의 일 실시예에 따른 사용자 인터페이스의 거래 작업패드에서의 비결합형 전통적인 간단한 거래의 화면이다.
- [0037] 도 15는 본 발명의 일 실시예에 따른 사용자 인터페이스의 거래 작업패드에서의 결합형 전통적인 간단한 거래의 화면이다.
- [0038] 도 16은 본 발명의 일 실시예에 따른 승인된 전통적인 간단한 거래의 거래 구문의 화면이다.
- [0039] 도 17은 본 발명의 일 실시예에 따른 전통적인 간단한 거래에 참가한 실시간 시뮬레이션된 수행 데이터의 화면이다.
- [0040] 도 18은 본 발명의 일 실시예에 따른 결합형 전통적인 복잡한 거래의 화면이다.
- [0041] 도 19는 본 발명의 일 실시예에 따른 결합형 협조적인 간단한 거래의 화면이다.
- [0042] 도 20은 본 발명의 일 실시예에 따른 협조적인 복잡한 거래를 수행하는 방법의 흐름도이다.
- [0043] 도 21a는 본 발명의 일 실시예에 따른 협조적인 복잡한 거래에서의 코드 분할 다중 거래 프로그램 모듈에 의해 유지되는 매크로 장부(macro ledger)를 나타내는 표이다.
- [0044] 도 21b는 본 발명의 일 실시예에 따라, 스프레드-벡터 해상도 프로그램 모듈에 의해 처리되는 협조적인 복잡한 거래에 대한 당사자들을 나타내는 도면이다.
- [0045] 도 21c는 본 발명의 일 실시예에 따른 협조적인 복잡한 거래에서의 코드 분할 다중 거래 프로그램 모듈에 의해 유지되는, 경험적으로 감소된 매크로 장부를 나타내는 표이다.

도면

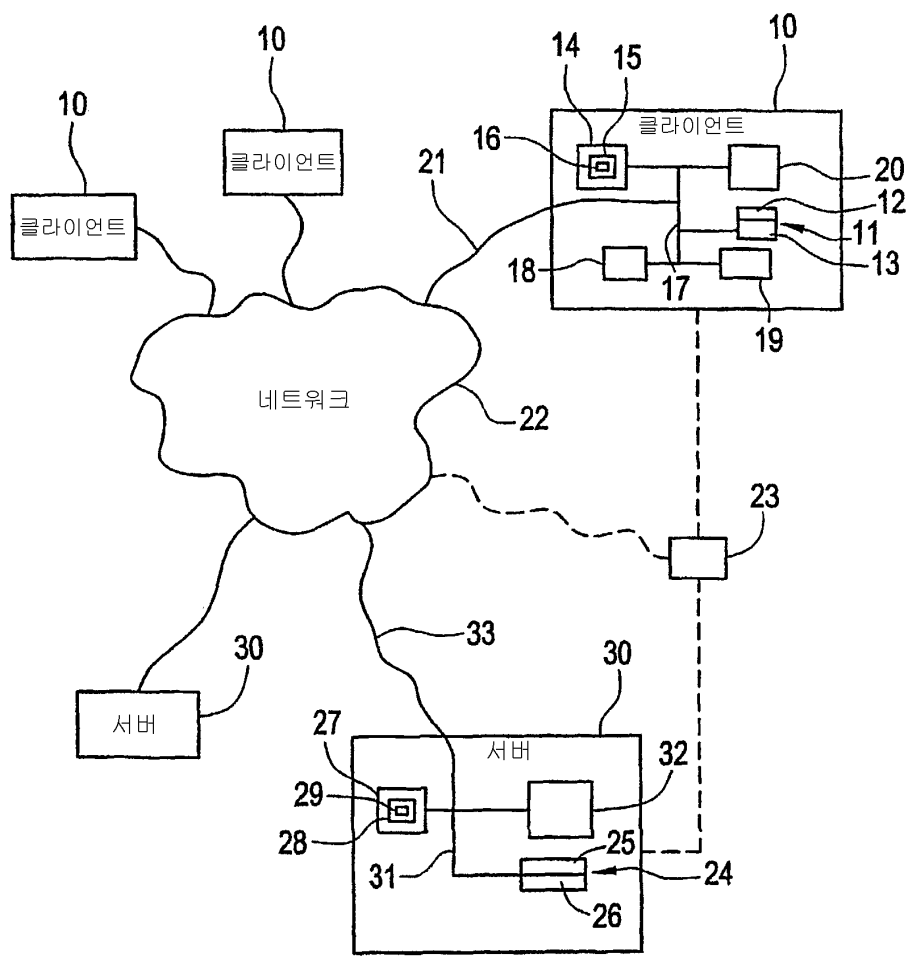
도면1



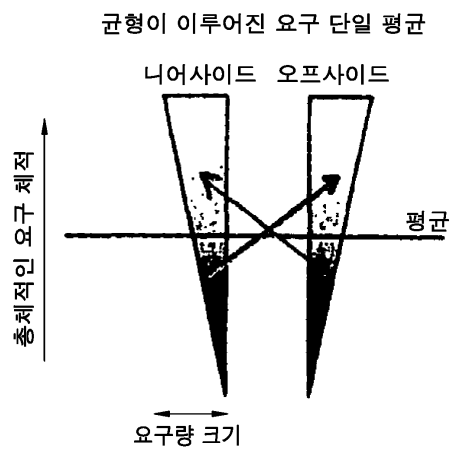
도면2



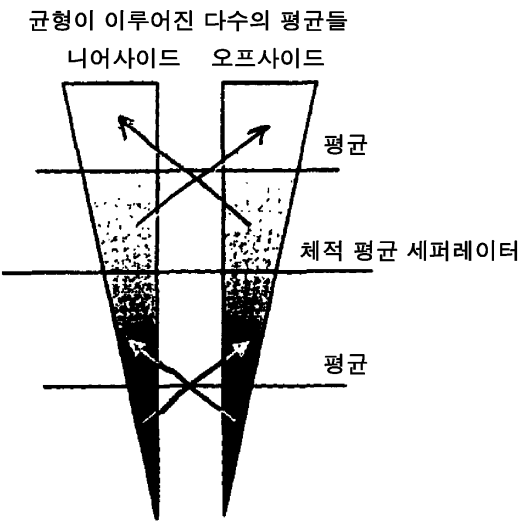
도면3



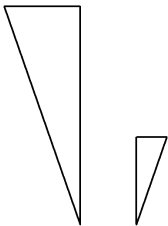
도면4



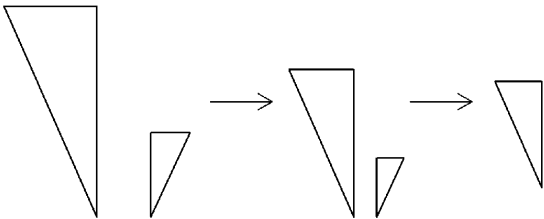
도면5



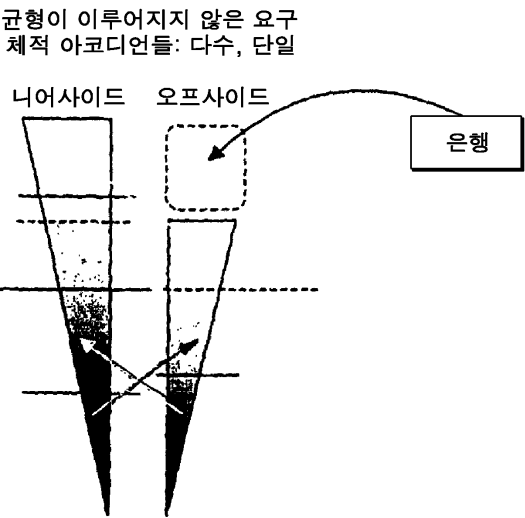
도면6



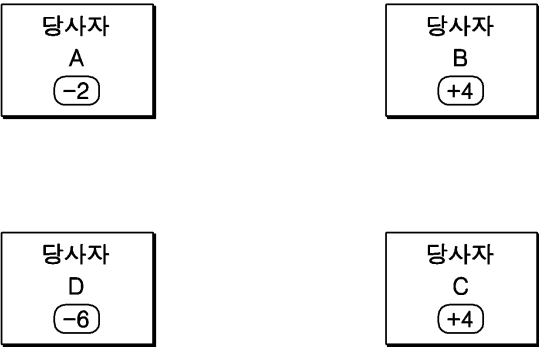
도면7a



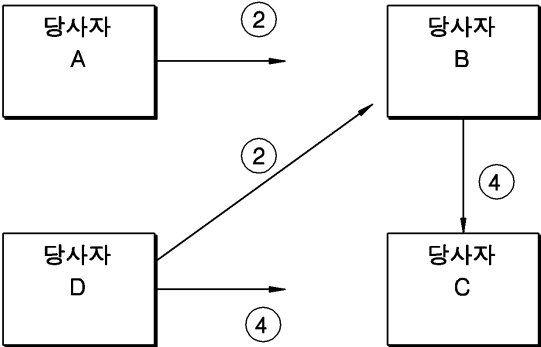
도면7b



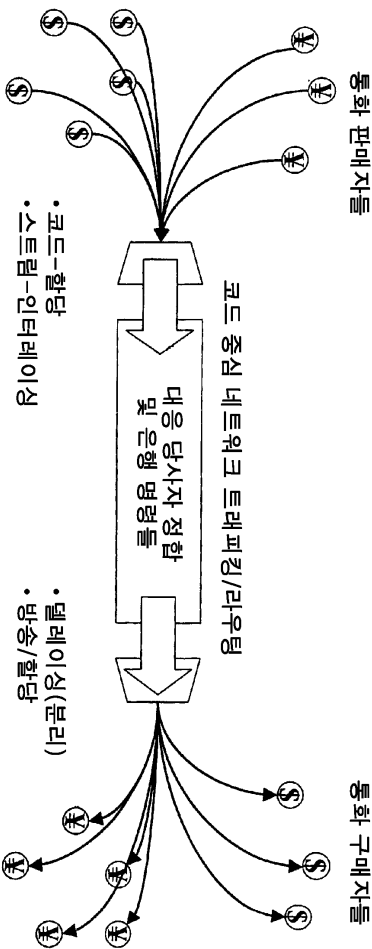
도면8a



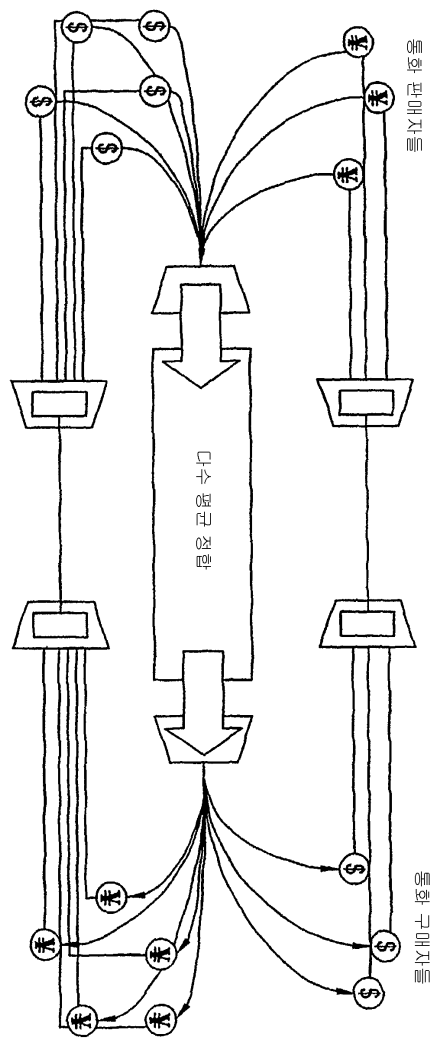
도면8b



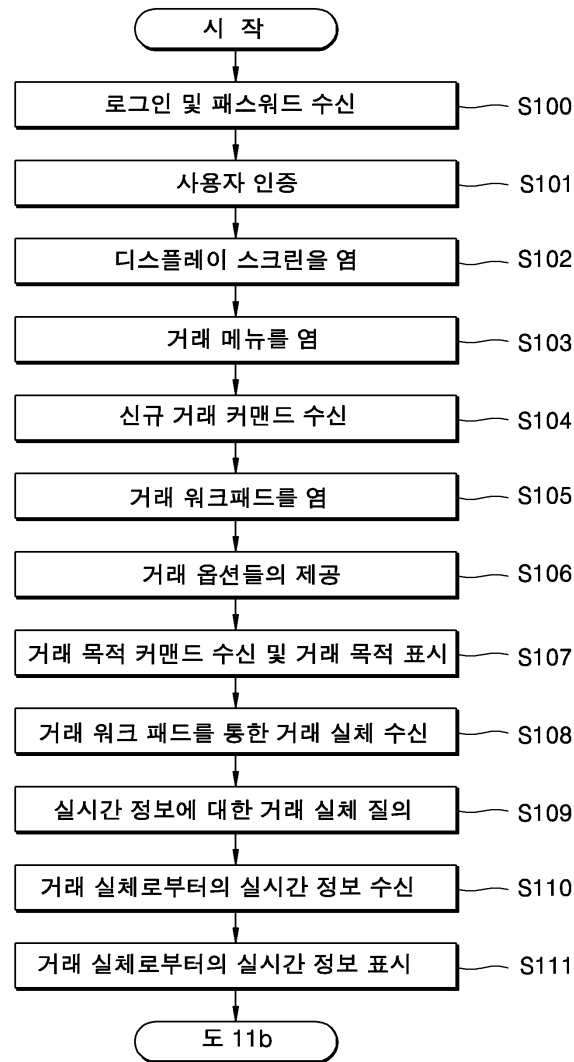
도면9



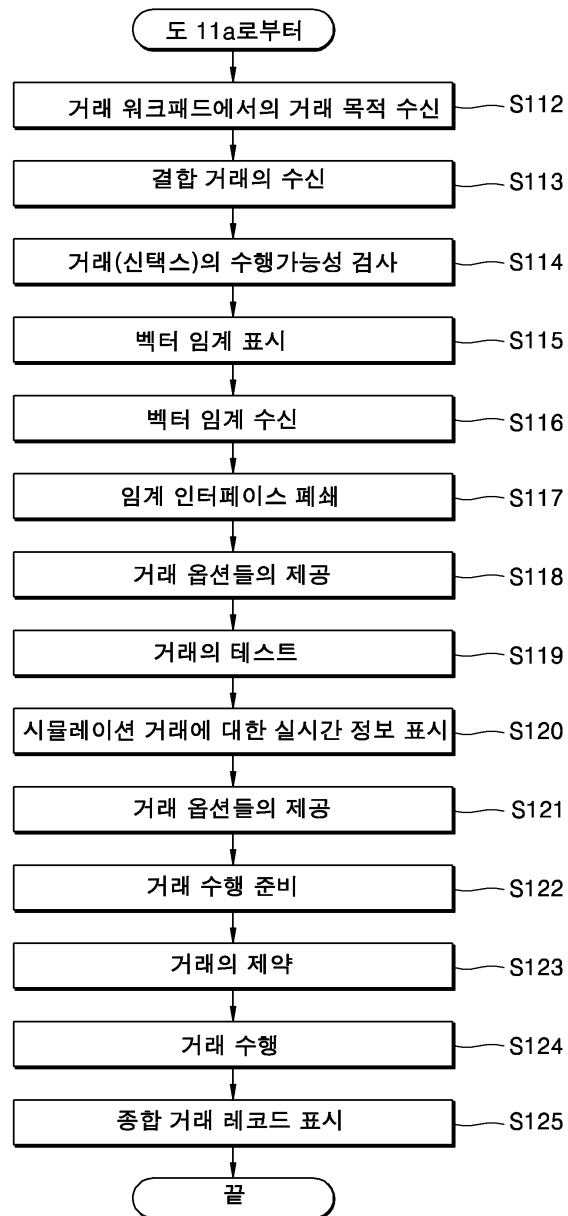
도면10



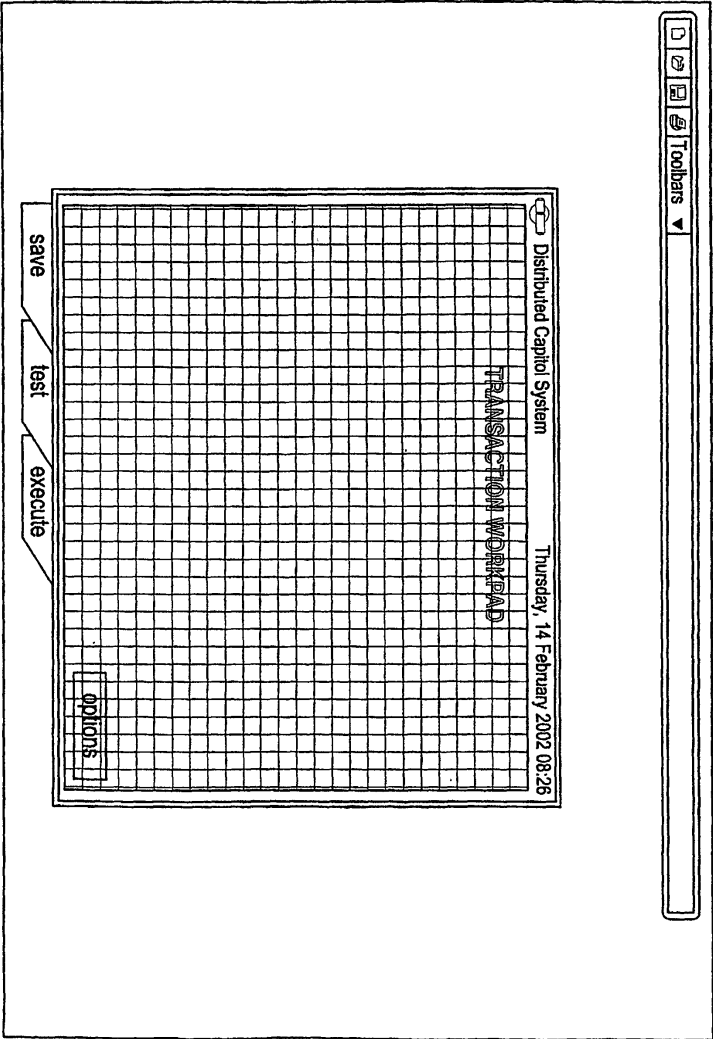
도면11a



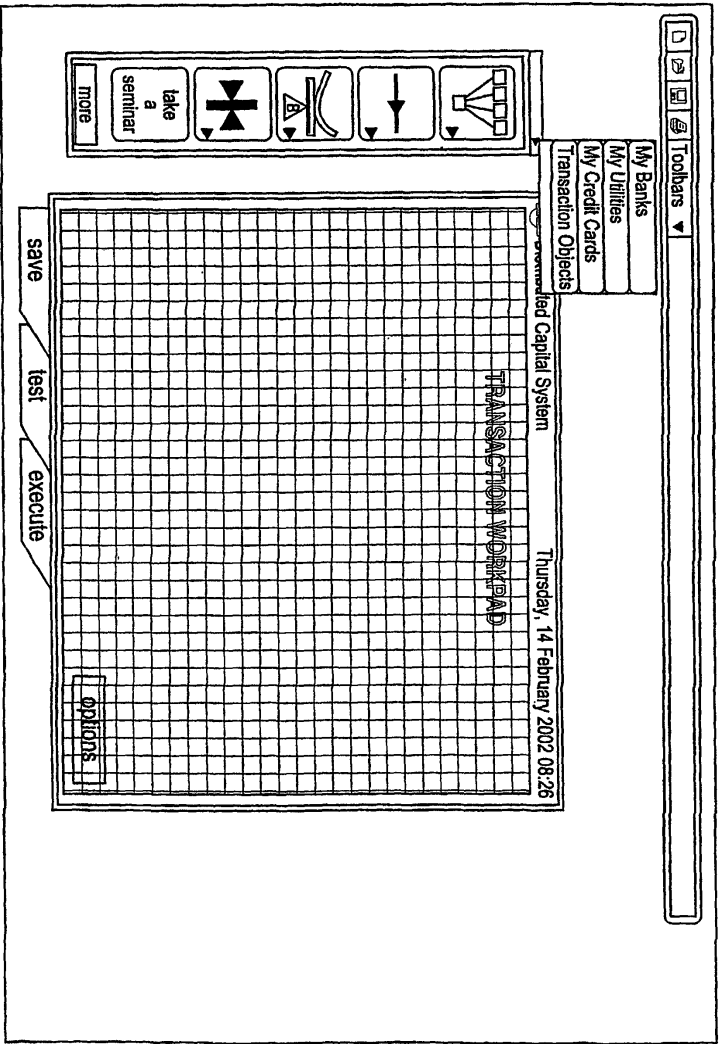
도면11b



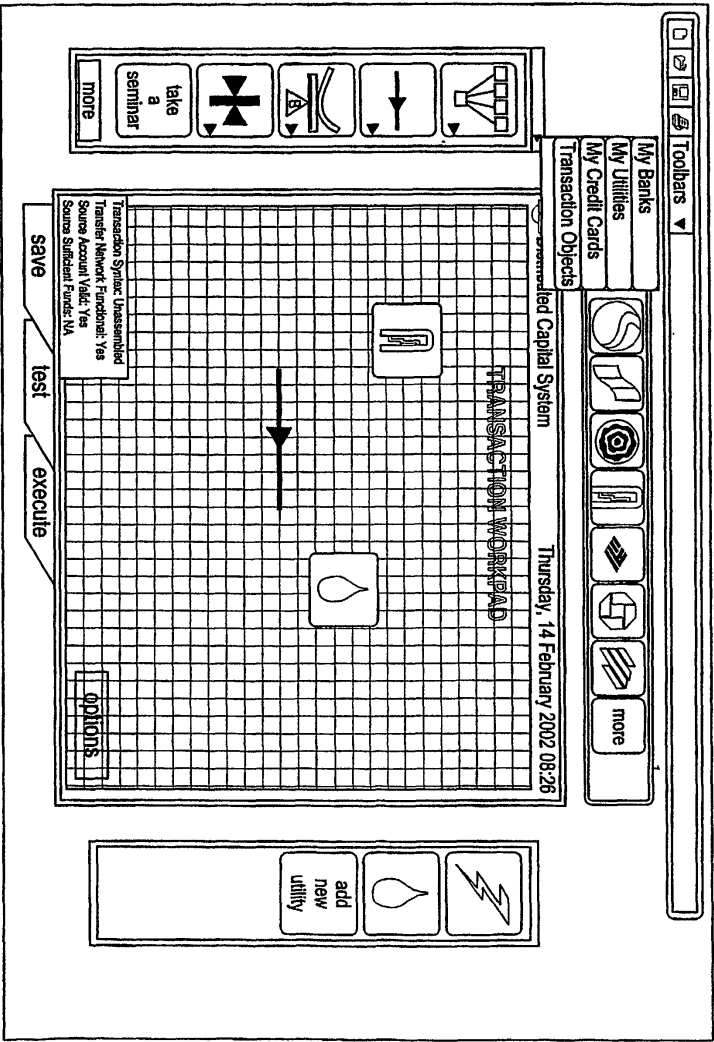
도면12



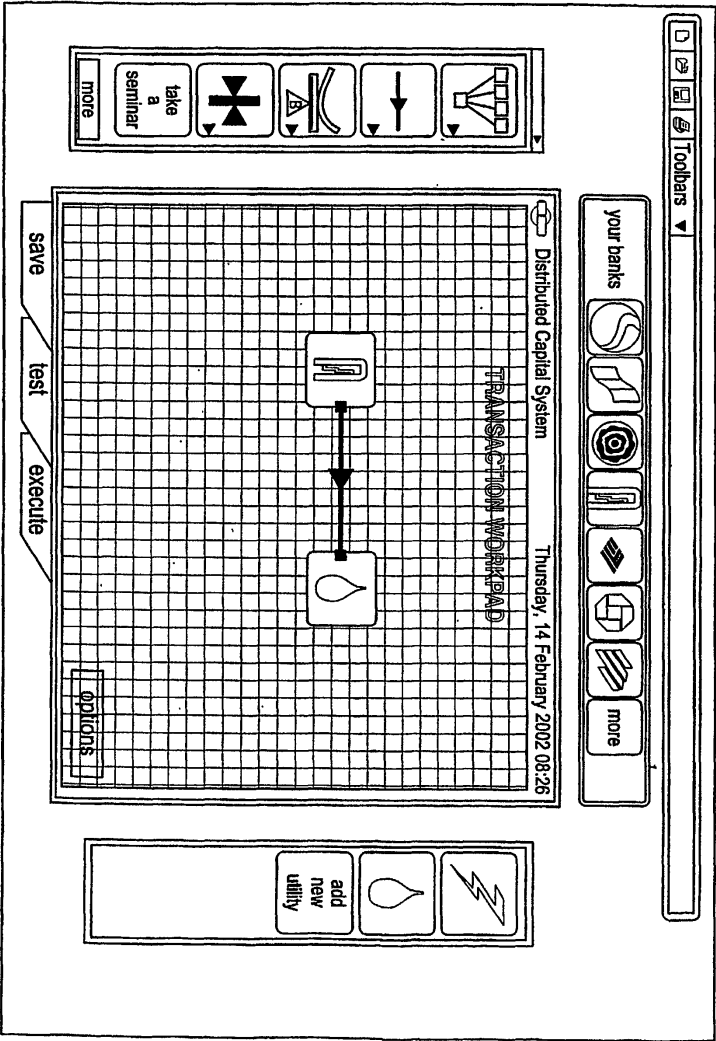
도면13



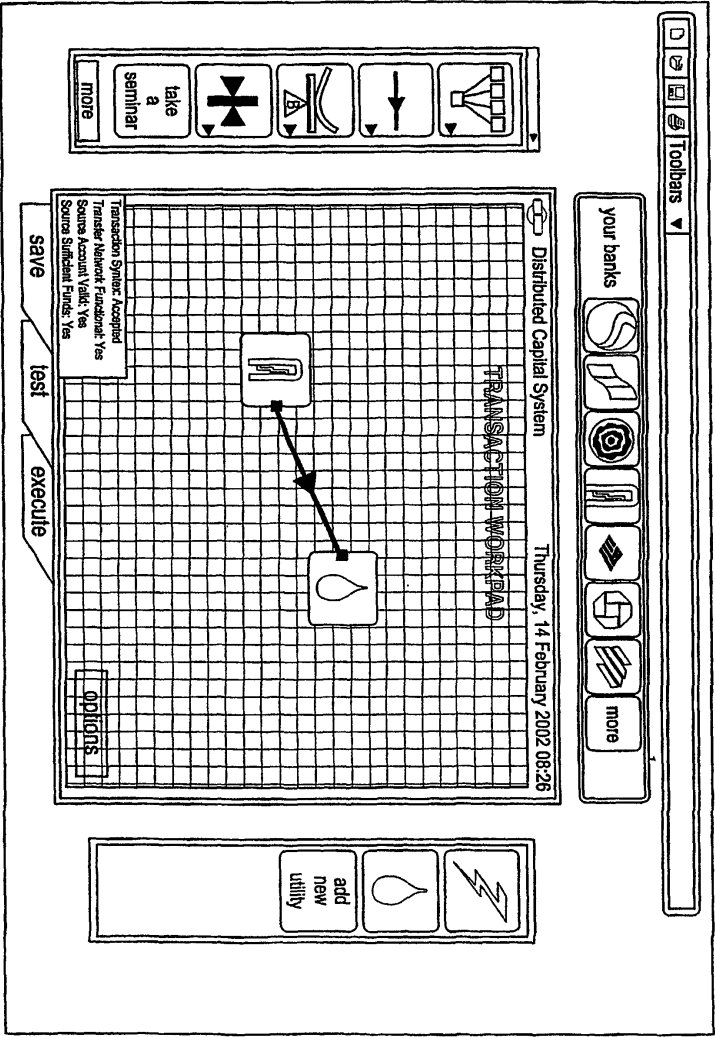
도면14



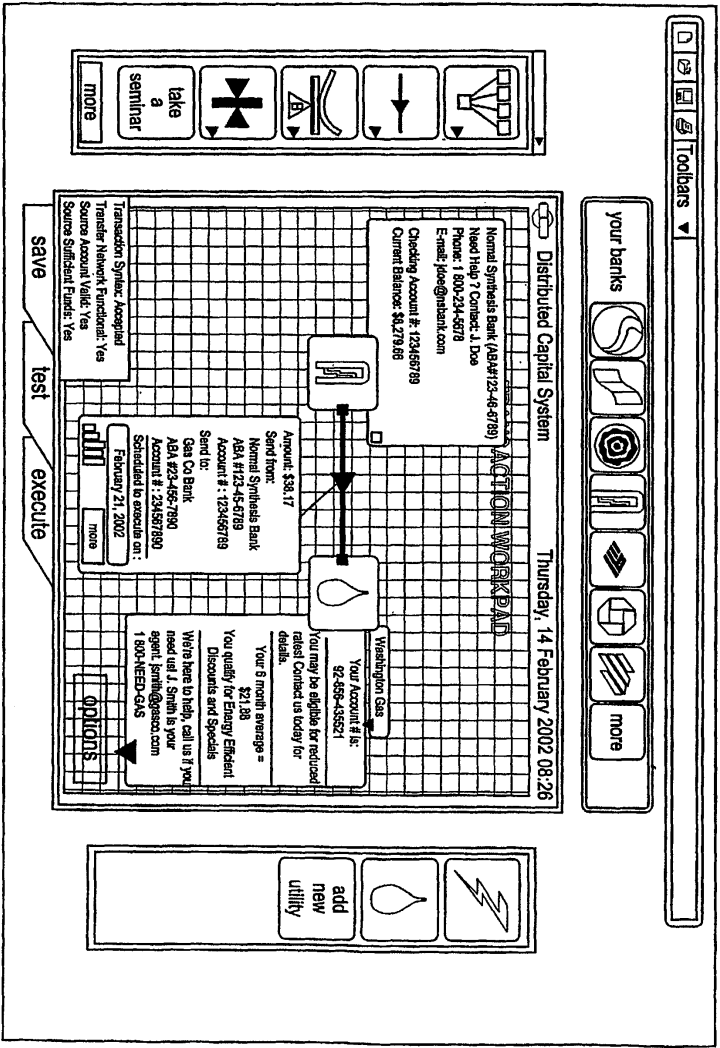
도면15



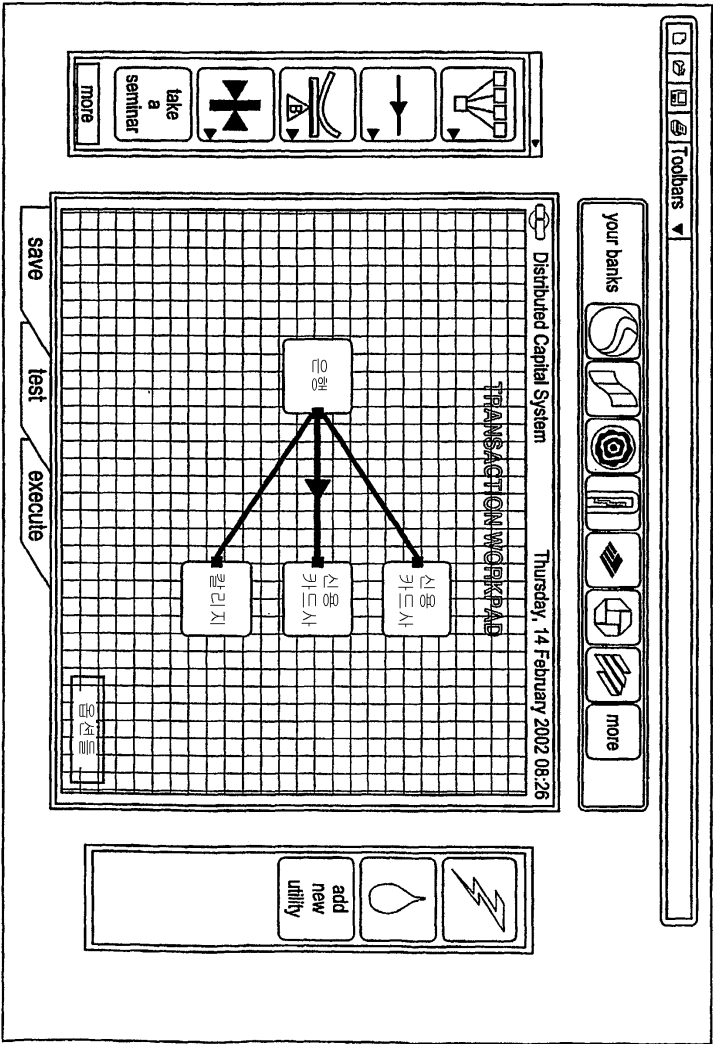
도면16



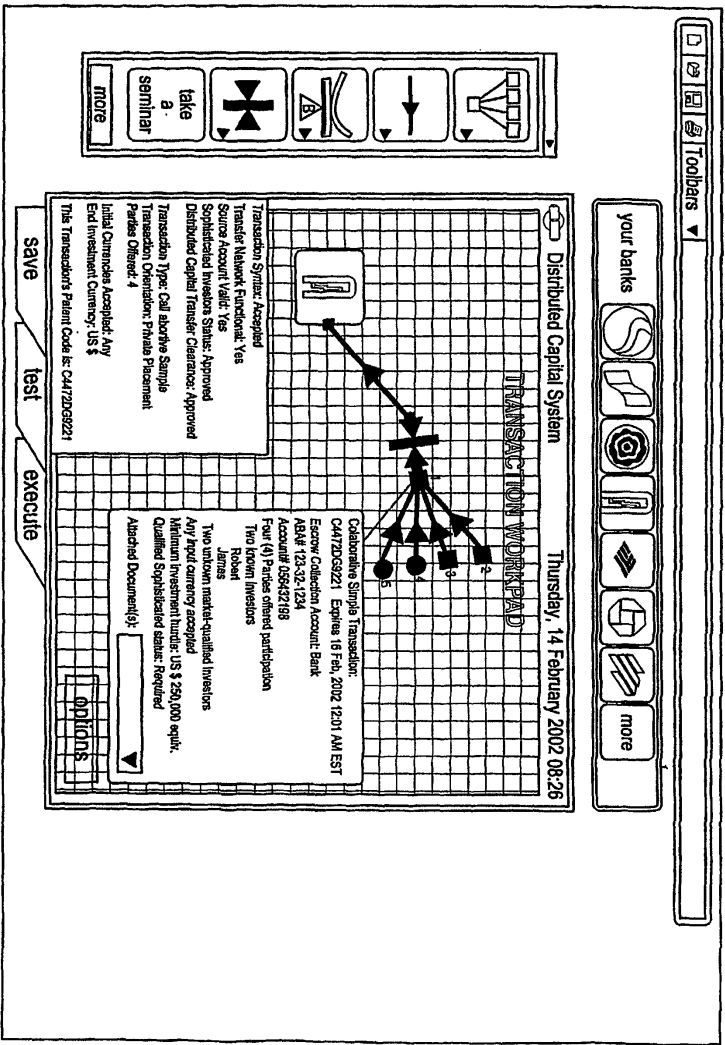
도면17



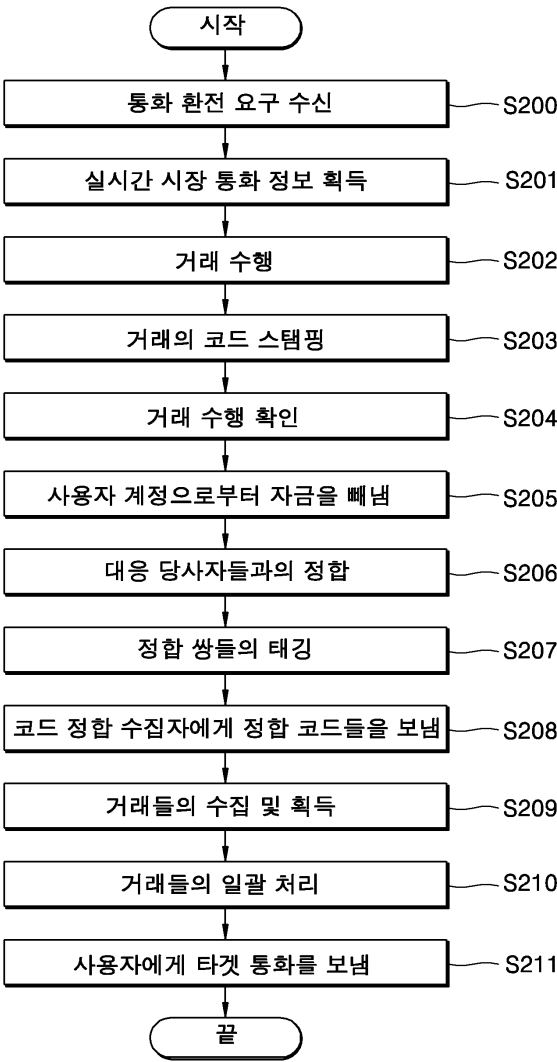
도면18



도면19



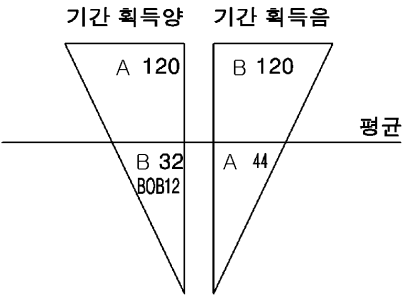
도면20



도면21a

매크로 장부		
정의된 기간: 2002 08 27 09:00:00-11:59:59		
시간	+대응 당사자	-대응 당사자
	A 44	A 44
	B 32	B 32
	BOB 12	B 12
	A 76	B 76

도면21b



도면21c

경험적으로 감소된 매크로 장부		
정의된 기간: 2002 08 27 09:00:00~11:59:59		
시간	+대응 당사자	-대응 당사자
	BOB 12	B 12
	A 76	B 76