

웹기반 बैं킹컴포넌트 시스템에서 승인시스템의 설계 및 구현

안 태 광[†] · 김 병 기^{††}

요 약

금융환경의 변화속도가 빨라지고 금융시스템으로의 접속채널이 다양해짐에 따라 신속한 상품개발과 배포 기능, 용이한 유지보수성을 지닌 새로운 बैं킹시스템이 요구되고 있는 요즘 컴포넌트 기반 개발방법은 이런 요구를 충족시킬 수 있는 소프트웨어 개발방법으로 주목받고 있다. 본 논문에서는 EJB बैं킹컴포넌트의 구성 및 기능을 소개하고 बैं킹컴포넌트 시스템에서 사용할 수 있는 승인시스템을 설계하고 구현하였다. 승인시스템의 구현을 위하여 बैं킹시스템에서 사용되는 승인사항들을 분류하여 등록하였으며 등록된 승인조건을 이용하여 트랜잭션의 승인필요여부를 판단할 수 있도록 하였다. 또 승인클라이언트를 웹기반으로 구현함으로써 웹환경으로의 클라이언트 표준화 원칙을 준수하고 클라이언트 프로그램의 배포 및 버전관리 문제를 해결하도록 하였다.

Design and Implementation of Approval System for Web-based Banking Component System

Tae-Kwang Ahn[†] · Byung-Ki Kim^{††}

ABSTRACT

As changing financial environment and increasing the number of access channel to banking system, new banking system is needed which can be developed financial commodity rapidly, can be deployed easily, and can be maintained easily. Component based development method is believed to meet these demands. In this paper, we propose EJB banking component and explain its system architecture and functionality. And we design and implement approval system on EJB banking component system. To implement approval system, approval conditions are classified and registered. Using registered approval condition, transactions are classified whether approval is needed or not. Approval client for web based client standardization has an advantage of no need to deploy and manage software version.

키워드 : बैं킹컴포넌트(Banking Component), 승인컴포넌트(Approval Component), 대화형 트랜잭션(Interactive Transaction)

1. 서 론

बैं킹시스템은 동시다발적인 금융트랜잭션을 처리하는 시스템으로 시스템의 안정성과 신속한 처리가 매우 중요시되는 시스템이다. 이런 이유로 बैं킹시스템을 구성하고 관련 응용프로그램을 개발할 때에는 안정성과 빠른 응답속도를 최우선목표로 설정하여 개발하여 왔다. 그러나 세계경제의 글로벌화, 인터넷의 활성화 등으로 인한 사회와 시장환경의 급격한 변화로 고객들의 요구사항과 금융시스템의 접근채널이 다양해짐에 따라 안정성과 신속한 처리 외에도 은행상품과 응용프로그램의 신속한 개발, 배포, 용이한 유지보수성도 필수적인 목표가 되었다. 실제로 과거에 창구담당과 자동화기기의 지원 정도에 그치던 बैं킹시스템이 최근 몇 년사이에 폰 बैं킹, 인터넷 बैं킹, 모바일 बैं킹 등으로 확대

되고 있고 은행내부적인 경영정보시스템 역시 리스크관리(RMS), 신고객관리(CRMS) 등 새로운 프로그램의 개발을 필요로 하게 되었으며 이로 인해 각종 클라이언트 프로그램들의 배포 및 표준화, 그리고 버전관리 문제가 대두되고 있는 실정이다.

이러한 필요조건을 충족시키기 위해 서버측 구현방법으로는 파라미터에 기반한 시스템[1], 컴포넌트에 기반한 시스템[2] 등이 제시되고 있다. EJB (Enterprise JavaBeans)는 컴포넌트 트랜잭션 모니터를 위한 표준 서버측 컴포넌트 모델로서 분산 트랜잭션을 보장하고 확장성, 이식성, 안정성 등이 우수하여 बैं킹시스템과 같은 복잡한 시스템을 구현하는데 적합하다. 또 재사용성이 우수하여 새로운 어플리케이션을 개발할 경우에도 기존의 컴포넌트를 재사용함으로써 보다 빠른 개발이 가능하며 웹기반의 클라이언트 개발이 용이하여 클라이언트 프로그램의 버전관리 문제를 해결할 수 있다. 은행이 사회의 변화와 시장환경에 민감하게 대처할 수 있는 신속한 상품개발과 배포 기능, 용이한

[†] 정 회 원 : 兩廣주은행 정보지원부
^{††} 총신회원 : 전남대학교 컴퓨터정보학부 교수
 논문접수 : 2001년 10월 5일, 심사완료 : 2001년 12월 13일

유지보수성을 지닌 시스템을 요구하고 있음을 감안할 때 객체지향/컴포넌트 기반의 EJB 백킹컴포넌트는 유력한 해결 방안 중의 하나라고 볼 수 있다[3].

가벼운 클라이언트 구현과 클라이언트 프로그램의 배포 및 버전관리 문제를 해결하는 방법으로는 웹기반의 클라이언트 개발 방법이 유력한 방법으로 제시되고 있다. HTML, JSP, ASP등을 이용하여 웹기반으로 클라이언트를 개발하면 프로그램의 배포 및 버전관리문제에서 벗어날 수 있으므로 실제로 많은 응용프로그램들이 웹환경을 기반으로 개발되고 있다. 하지만 웹기반의 클라이언트는 비연결지향의 HTTP프로토콜의 특성상 사용자인증, 세션관리 등의 어려움으로 인해 하나의 트랜잭션을 처리하는 도중 사용자와의 상호작용을 요하는 대화형 트랜잭션의 처리가 곤란하다는 단점이 있다.

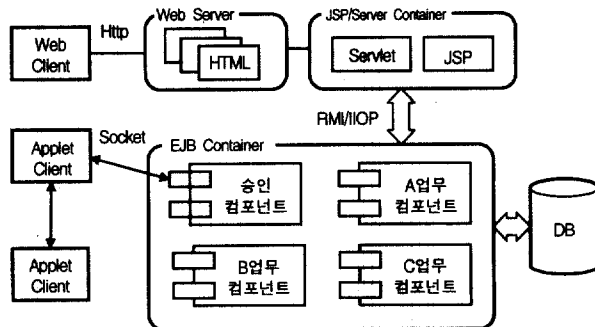
뱅킹시스템에서 사용되는 승인처리란 은행에서 처리되는 제반업무 중 확인 또는 주의를 요하는 업무를 별도로 지정하여 해당 트랜잭션이 발생할 때마다 그 취급의 타당성을 승인자(주로 상위책임자)가 검사하고 확인하게 하는 것으로 금융사고 발생을 미연에 방지하고 보다 신중하게 처리하도록 유도한다. 따라서 은행창구에서 발생하는 모든 트랜잭션은 승인을 요하는지 반드시 확인하여야 하고 승인을 요하는 트랜잭션은 승인자의 승인여부에 따라 트랜잭션의 유효성이 결정된다. 이런 이유로 승인처리는 트랜잭션의 처리도중 승인자의 인증을 받아야 하는 대화형 트랜잭션으로 처리되어야 하는데 연결지향의 클라이언트어플리케이션에서와는 달리 비연결지향의 웹환경에서는 승인처리와 같은 대화형 트랜잭션의 구현에 어려움이 있다.

본 논문에서는 EJB에 기반한 백킹컴포넌트 시스템의 구조 및 각 기능에 대해서 설명하고, 승인처리를 하나의 트랜잭션으로 포함하여 처리할 수 있는 승인컴포넌트 및 승인 클라이언트를 설계하고 구현하였다. 승인컴포넌트를 구현하기 위하여 기존의 승인처리방법 및 승인대상거래를 분석하여 EJB 컴포넌트로 구현하였으며 승인클라이언트는 웹브라우저와 인증애플릿(signed applet)을 연동하여 구현함으로써 웹환경으로의 클라이언트 표준화 원칙을 준수하고 클라이언트 프로그램의 배포 및 버전관리 문제를 해결하도록 하였다.

2. 백킹컴포넌트 시스템의 구성 및 기능

백킹컴포넌트 시스템은 JSP/서블릿/애플릿 위주의 클라이언트측과 업무컴포넌트 위주의 서버측으로 나눌 수 있다. 텔러는 웹브라우저에서 JSP를 통해 서비스를 요청하고 이 요청은 해당 업무컴포넌트에 전달되어 트랜잭션의 유효성 검사, 승인처리를 거친 후 다른 업무컴포넌트들의 협조를 받아 처리된다. 모든 처리가 끝나면 해당 업무컴포넌트는 클라이언트측에 처리결과를 반환한다. 각 구성요소들과 그

관계를 그림으로 표현하면 (그림 1)과 같다.



(그림 1) 백킹컴포넌트 시스템 구성도

2.1 클라이언트

텔러가 웹브라우저를 통해 로그인을 하면 로그인 JSP는 영업일자, 영업점ID, 텔러ID, 단말유형 등의 기본정보를 세션에 저장하고 있다가 각각의 트랜잭션 발생시 텔러의 입력사항을 추가하여 서블릿으로 보낸다. 서블릿은 JSP로부터 받은 기본정보와 사용자의 입력사항을 조합하여 각 업무처리에 필요한 자바빈 (Java Bean)을 생성한 후 해당 업무컴포넌트에게 서비스를 요청한다. 모든 처리가 끝난 후 업무컴포넌트는 그 처리결과를 출력하기 위해 자바빈을 생성하여 다시 서블릿으로 보낸다. 서블릿은 처리결과를 출력하기 위해 처리결과가 들어있는 자바빈을 Request 객체에 담아 JSP로 보내고 JSP에서는 이 자바빈을 화면형식에 맞게 출력한다. 해당 컴포넌트로부터 예외가 던져 진다면 해당 예외에 대한 적절한 오류메시지를 여러 JSP에 출력한다.

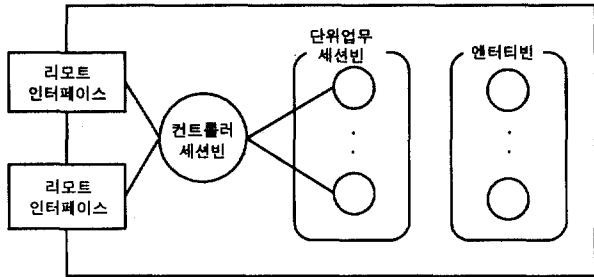
클라이언트에는 트랜잭션 처리 요청을 위한 JSP외에도 승인처리를 위한 애플릿도 존재해야 하는데 이 애플릿은 승인컴포넌트로부터 승인사유가 전달되면 승인자의 애플릿에게 승인요청을 하거나 또는 다른 클라이언트의 애플릿으로부터 승인요청이 전달될 때 승인요청한 클라이언트에 승인응답을 준다. 승인애플릿의 처리흐름에 대해서는 3장에서 다시 다루기로 하겠다.

2.2 서버 컴포넌트

업무컴포넌트는 각 단위업무별로 필요한 엔터티빈과 업무로직을 통해 엔터티빈을 제어하는 세션빈으로 구성된다. 또 세션빈은 크게 클라이언트의 요청을 받아 해당 업무의 흐름(business logic)에 따라 처리해주는 업무별 컨트롤러 세션빈(업무컨트롤러 : business controller)과 이들 업무컨트롤러에 의해 제어되는 단위세션빈으로 구성된다. 특정 메소드를 업무컨트롤러에 두느냐 단위업무 세션빈에 두느냐의 문제는 해당 메소드의 재사용성과 다른 컴포넌트와의 관련성을 고려하여 설정한다. 업무처리로직이 시스템환경이나 클라이언트 환경에 비교적 민감하여 변경요건이 많거나 다른 컴포넌트와 협력해서 처리해야 하는 메소드는 업무컨

트롤러에 포함시키고 변경가능성이 상대적으로 적고 다른 컴포넌트에 의존적이지 않은 메소드는 단위업무 세션빈에 두도록 한다.

이를 그림으로 표현하면 (그림 2)와 같다.



(그림 2) 개별 업무컴포넌트의 구성도

2.2.1 컨트롤러 세션빈(업무컨트롤러)

업무컨트롤러는 클라이언트 서블릿으로부터 입력내용을 넘겨 받은 후 해당업무의 처리순서에 맞게 각각의 단위세션빈에게 서비스를 요청하고 응답을 받는 등의 트랜잭션을 조절하고 관리하는 메소드들을 가지는 세션빈이다. 본 논문에서의 각 업무컴포넌트는 하나의 업무컨트롤러를 가지고 있으며 모든 트랜잭션은 업무컨트롤러의 메소드단위로 관리된다.

업무컨트롤러의 기능에는 각 업무의 흐름에 대한 처리뿐 아니라 트랜잭션의 유효성검사, 감사로그 기록 등 트랜잭션의 공통적인 사항도 포함되는데 뱅킹트랜잭션에서 공통적으로 처리되는 사항은 <표 1>과 같다.

<표 1> 뱅킹 트랜잭션의 기본기능

메소드명	기능
tx_init()	트랜잭션ID생성
tx_checkAuthority()	트랜잭션시간, 영업일자 검사 업무에 따른 사용자권한 검사 단말유형별 트랜잭션금지여부검사 업무별 트랜잭션금지여부 검사 서비스시간별 트랜잭션가능여부 검사
tx_approval()	승인처리
tx_bookKeeping()	일계처리
tx_writeAudit()	감사용 로그 기록

트랜잭션 공통사항과 업무 고유의 흐름은 하나의 트랜잭션으로 동작하게 되며 처리결과가 모두 정상이면 그 결과를 서비스를 요청했던 클라이언트로 반환한다. 처리과정중 예외가 던져지면 업무컨트롤러는 감사용 로그기록을 제외한 이후의 모든 처리를 취소하고 해당 예외를 클라이언트에 던져 예외발생을 알린다.

감사용 로그는 사후의 거래추적을 위해 남기는 로그로 해당 트랜잭션이 정상완료이거나 실패이거나에 관계없이 모두 기록하며 사후감사용으로 사용할 수 있도록 트랜잭션

처리에 관계된 대부분의 정보를 입력하게 된다.

2.2.2 단위업무 세션빈

단위업무 세션빈은 업무컨트롤러로부터 요청받은 서비스를 처리하여 응답해 주는데 주로 개별 엔터빈과 동작하거나 컴포넌트 내 다른 빈들과만 관련을 갖는 메소드들로 구성된다. 단위업무 세션빈은 업무컨트롤러로부터 요청받은 트랜잭션을 처리하기 위해 필요한 세션빈과 엔터빈을 이용하고 처리결과가 정상이면 결과를 업무컨트롤러에 반환하고 예외가 발생하면 해당 예외를 반환한다.

3. 승인시스템의 기능설계

3.1 승인시스템의 처리흐름

뱅킹시스템에서 승인처리를 위해 사용하는 일반적인 방법은 다음과 같다. 승인이 필요한 거래를 규정하고 승인이 발생할 수 있는 업무화면들을 모두 정의한 다음 각각의 업무화면을 처리할 때 승인요건이 있는지 없는지를 검사한다. 즉 클라이언트 화면 코드내 승인필요여부를 검사하는 로직이 포함되는데 승인기준이 변경되면 해당 클라이언트 프로그램은 다시 작성하여 배포되어야 한다. 승인이 필요하다고 판단된 트랜잭션은 해당 클라이언트의 화면에 승인을 요청하는 메시지가 나타나고 승인은 카드키 또는 지문을 이용하거나 승인의 비밀번호를 클라이언트의 단말에 직접 입력함으로써 승인을 처리한다.

본 논문에서 구현한 승인시스템은 승인필요여부 검사를 서버측에서 담당하도록 설계함으로써 승인조건의 변화가 클라이언트 프로그램에 영향을 끼치지 않도록 하였으며 승인이 필요시 승인요청자와 승인의 단말에 필요한 메시지를 직접 출력하도록 하였다. 승인은 주어진 고유의 비밀번호를 자신의 단말에 입력함으로써 승인을 처리한다.

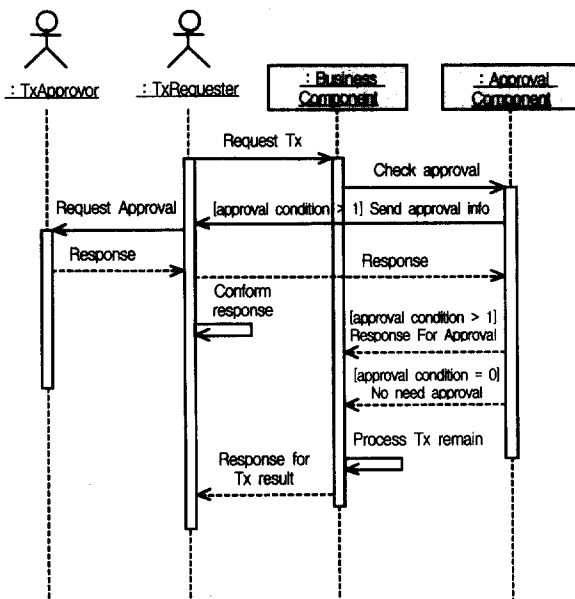
본 논문에서 구현한 승인처리시스템의 처리흐름 및 필요요건을 설명하면 다음과 같다.

- 1) 텔러는 처리해야 하는 트랜잭션을 업무컴포넌트에 처리요청한다.
- 2) 업무컴포넌트는 트랜잭션 ID를 생성하고 트랜잭션의 유효성을 확인한 후 승인컴포넌트에게 승인받을 필요가 있는지 확인요청한다.
- 3) 승인컴포넌트는 확인요청된 트랜잭션의 승인관련정보와 등록된 승인조건을 비교하여 승인필요여부를 확인한다.
- 4) 확인요청받은 트랜잭션이 승인을 요하는 트랜잭션이면(승인조건이 1개 이상 존재) 승인컴포넌트는 승인요청자(트랜잭션을 발생시킨 텔러)에게 승인이 필요함을 알리고 승인결과를 허용된 시간동안 기다린다. 허용된 시간동안 승인요청자로부터 응답이 오지 않으면 승인

요청이 거부된 것으로 간주하며 승인로그의 승인결과는 타임아웃으로 기록한다.

- 5) 승인요청자는 승인컴포넌트로부터 승인에 관련된 정보를 전송받고 승인사유를 확인한 후 승인자를 지정하여 승인을 요청한다. 승인요청자에게 전송되는 승인관련정보는 승인요청자와 승인자가 승인사유를 명확하게 인지할 수 있게 구성되어야 하며 승인요청자에게 승인이 가능한 승인자의 리스트를 보임으로써 승인요청시 승인요청자가 승인자를 선택하여 지정할 수 있어야 한다. 만약 승인요청 후 승인자로부터 허용된 시간내에 응답이 오지 않으면 승인요청이 거부된것으로 간주하며 승인컴포넌트에 승인결과를 타임아웃으로 응답한다.
- 6) 승인자는 승인요청자로부터 승인요청이 발생하면 승인사유 및 승인요청자를 확인한 후 승인여부를 결정하여 승인요청자에게 응답한다. 승인자의 승인이 끝나면 승인된 트랜잭션은 승인요청자가 임의로 중지시키거나 취소시킬 수 없다.
- 7) 승인컴포넌트는 승인요청자로부터 승인결과를 받으면 승인내역을 기록하고(승인로그) 업무컴포넌트에 승인결과와 제어를 돌려준다.
- 8) 업무컴포넌트는 승인결과에 따라 트랜잭션의 나머지를 처리하거나 트랜잭션을 롤백시킨 후 감사로그에 트랜잭션에 관계된 사항을 기록한다.
- 9) 모든 처리가 끝나면 업무컴포넌트는 트랜잭션을 발생시킨 클라이언트에 처리 결과를 응답한 후 종료한다.

이상의 특징을 고려하여 동작하게 될 동기승인의 순차도를 그려보면 (그림 3)과 같다.



(그림 3) 승인처리의 순차도

3.2 승인컴포넌트의 기능

승인컴포넌트는 크게 4가지의 주요 기능을 포함하여야 하는데 그 내용은 다음과 같다.

- 첫째, 승인조건등록은 은행에서 승인을 요하는 트랜잭션을 분류하여 조건검사에 적절한 형태로 등록하고 이를 조회, 수정, 삭제 할 수 있게 하는 기능이다.
- 둘째, 승인필요확인기능은 처리 요청된 트랜잭션이 승인조건에 등록된 사항과 부합하여 승인을 받아야 하는지 또는 해당되는 승인조건이 없어 승인이 필요치 않은 트랜잭션인지를 판단하는 기능이다.
- 셋째, 승인처리를 요청하고 응답 받는 기능은 승인클라이언트와 밀접하게 관련되는 기능으로서 승인이 필요하다고 판정되면 트랜잭션을 요청한 클라이언트에게 승인이 필요함을 알리고 승인에 필요한 정보를 전송한 후 승인결과를 기다리는 기능이다.
- 넷째, 승인로그 기능은 승인관련 내용을 기록함으로써 사후에 특정한 승인트랜잭션의 내역을 조회할 수 있도록 하는 기능이다.

승인조건등록을 위해서는 승인사유를 어떻게 분류하느냐가 중요한데 대부분의 승인트랜잭션을 트랜잭션 유형, 트랜잭션 기준일, 클라이언트 프로그램 ID, 금리, 금액, 타점권 금액, 통장유무 정보를 통해 분류할 수 있다, 다음의 <표 2>는 기존의 승인트랜잭션의 예와 이를 분류한 결과이다.

<표 2> 승인트랜잭션의 예

거래명	거래내용	분류구분
취소거래	취소거래	트랜잭션 유형(취소)
사고해제	계좌사고해제 고객사고해제 부도연장해제 주의코드해제 등	트랜잭션 유형(해제)
기산일거래	기산일거래	트랜잭션 기준일
신규	기준금리초과 CD, 어음발행액 초과 특별이율추가	클라이언트프로그램ID 금액 금리
해지	계좌해지	트랜잭션 유형(해지)
출금	2중주의코드 설정 고객출금 타점권출금 무통장출금	클라이언트프로그램ID 거래금액 타점권금액 통장유무
원장변경	과세코드변경 고객번호변경 양도양수 계리수정 이익지급율변경	클라이언트프로그램ID
기타거래	비밀번호조회 잔액증명발급 개인연금납입증명발급 대출잔액증명발급 영업점마감 이자면제	클라이언트프로그램ID

3.3 승인클라이언트의 기능

승인클라이언트는 (그림 3)에서의 처리흐름에 따라 승인요청자로서의 기능과 승인자로서의 기능을 모두 가진다. 승인요청자로서의 기능은 승인컴포넌트의 승인요구를 받아 승인자에게 승인을 요청한 후 응답을 받고 그 응답결과를 승인컴포넌트에게 돌려주거나 또는 승인컴포넌트에게 승인요청포기를 알리는 기능이다. 승인자로서의 기능은 다른 승인요청자로부터 승인요청 받은 내용을 화면에 출력하고 승인자로부터 받은 승인결과를 승인요청자에게 돌려주는 기능이다.

승인클라이언트를 이런 주요 기능을 수행하기 위해 승인요청자 또는 승인자에게 승인에 관련된 정보를 화면에 보여 주어야 하는데 승인사유, 승인요청가능한 승인자의 성명, 직위 등이 그 예이다. 이런 정보들은 승인컴포넌트로부터 전달받아 승인요청자나 승인자 모두에게 보여져야 한다.

4. 승인시스템의 구현

4.1 구현환경

승인컴포넌트를 비롯한 서버측 컴포넌트는 모두 EJB로 구현하였고 승인클라이언트는 자바소켓을 사용하는 인증애플릿(signed applet)과 웹브라우저를 사용하는데 netscape.javascript.JSObject 클래스를 사용하여 자바스크립트로 인증애플릿과 웹브라우저를 연동하도록 하였다.

인증애플릿은 승인컴포넌트나 또 다른 인증애플릿으로부터 온 메시지를 받고 응답을 전송하는 역할을 담당하고, 이를 화면에 출력하고 사용자로부터 입력을 받는 기능은 모두 웹브라우저를 사용하였다. 승인클라이언트로 웹브라우저를 사용함으로써 웹브라우저의 세션재체에 담겨진 여러 정보를 이용할 수 있으며 또 승인클라이언트를 일반 트랜잭션 클라이언트와 같이 사용할 수 있는 장점이 있다. 승인처리시스템의 구현환경은 <표 3>와 같다.

<표 3> 승인처리시스템의 구현환경

항 목	소프트웨어
EJB 컨테이너	WebLogic 6.0
웹브라우저	MS-Explorer 5.5
인증애플릿구현도구	Microsoft SDK for Java 4.0

4.2 승인컴포넌트 구현

승인컴포넌트는 업무컴포넌트에게 서비스를 요청받으면 일정시간 내에 승인필요여부를 검사하여 해당 클라이언트에 승인요청이 필요함을 알린 후 승인결과를 업무컴포넌트에 응답해야 한다. 승인컴포넌트의 그 주요기능 및 해당 메소드들을 요약하면 <표 4>와 같다.

<표 4> 승인컴포넌트의 주요기능

주요 리모트인터페이스 메소드	기능
void registerApprovalCondition()	승인조건등록
void updateApprovalCondition()	승인조건수정
void removeApprovalCondition()	승인조건삭제
Collection inquireApprovalCondition()	승인조건조회
void checkApproval()	승인필요여부 검사
void writeApprovalLog()	승인로그기록
void inquireApprovalLog()	승인로그조회

승인조건등록에서는 은행에서 승인을 요하는 트랜잭션들을 분석하여 그 필요조건을 각각의 항목으로 분류하여 클라이언트프로그램 ID를 키로 하여 등록한다. बैं킹시스템에서 사용되는 프로그램은 서버프로그램, 클라이언트프로그램을 가리지 않고 새로 배포되거나 갱신될 때에는 보안상 반드시 일정 형식에 의해 등록, 관리되어야 하는데 새로운 프로그램이 배포되거나 기존의 적용된 프로그램이 변경되었을 때에는 변경된 프로그램의 승인조건도 함께 확인하여 변경될 수 있도록 한다. 승인조건등록에 필요한 항목들을 정리해 보면 <표 5>과 같다.

<표 5> 승인조건등록에 필요한 항목

데이터형	속성이름	설 명
String	approvalId	승인조건ID
String	approvalName	승인조건명
String	txDate	트랜잭션 기준일
String	txType	트랜잭션 유형 0: 일반 1: 취소 2: 해지 3: 해제
String	clientProgramId	클라이언트프로그램ID
String	discharge	면제, 감면
String	noPassbook	무통장거래
String	appAmount	거래금액
String	yourBankAmount	타점권금액
String	appInterest	거래이율
String	appReason	승인사유

승인필요검사에서는 등록된 승인조건을 이용하여 요청된 트랜잭션이 승인을 요하는지를 확인하는데, 업무컴포넌트와의 인터페이스를 담당하고 승인요청을 실시하는 기능을 포함한다. 승인필요검사에 필요한 항목들은 <표 5>의 내용에 트랜잭션 ID와 같은 기본정보를 포함하고 클라이언트 프로그램 ID를 프라이머리키로 하여 해당 엔터티를 찾은 후에 각각의 세부조건을 비교하여 판단한다. 승인이 필요하지 않는 트랜잭션은 업무컴포넌트에 승인 없음을 응답하고 별도의 승인로그는 기록하지 않는다. 승인이 필요한 트랜잭션은 처리 요청한 클라이언트의 IP주소를 참조하여 해당 클라이언트에게 승인에 필요한 정보를 보낸 후 허용된 시간동안 응답을 기다려 응답이 오면 승인로그를 기록한 후 승인의 결과를 업무컨트롤러에 알린다.

승인컴포넌트가 승인클라이언트에 보내주는 정보는 승인

자에 관한 정보와 승인사유에 관한 정보인데 승인자 정보는 승인자의 직위, 성명, IP주소를 포함하며 트랜잭션에 따라 같은 부점내의 모든 텔러가 승인자가 될 수 있다. 승인 사유는 트랜잭션ID, 승인사유, 승인관련 금액, 이율 등이다. 다음 <표 6>은 승인컴포넌트가 승인 클라이언트에 보내는 정보의 형식이다.

<표 6> 승인클라이언트에 보내지는 정보의 형식

데이터형	속성이름	설 명
String	txId	트랜잭션ID
String	employeeId	승인요청자 직원번호
String	employeeName	승인요청자 성명
String	employeeTitle	승인요청자 직위번호
String	employeeIp	승인요청자 IP주소
Collection	approverList	승인자정보(승인요청자의 정보와 같은 형태)
String	basicApprover	기본으로 설정된 승인자
String	appTxName	승인요청 트랜잭션명
String	accountId	계좌번호
String	customerName	고객성명
String	appReason1	승인사유
String	appAmount1	승인관련 트랜잭션금액
String	appInterest1	승인관련 이율
String	appAttention1	사고신고등급
String	appTxDate1	기산일
String	appReason2	승인사유
String	appAmount2	승인관련 트랜잭션금액
String	appInterest2	승인관련 이율
String	appAttention2	사고신고등급
String	appTxDate2	기산일
String	appReason3	승인사유
String	appAmount3	승인관련 트랜잭션금액
String	appInterest3	승인관련 이율
String	appAttention3	사고신고등급
String	appTxDate3	기산일

승인로그기록에서는 승인처리 내역을 그 결과와 함께 기록하고 이후에 그 내역을 조회할 수 있도록 한다.

4.3 클라이언트 프로그램 구현

승인에플릿은 사용자의 로그인과 동시에 서버로부터 다운로드되어 실행된다. 승인에플릿이 실행되면 특정 포트를 서버소켓으로 열고 다른 승인에플릿이나 승인컴포넌트로부터 승인요청을 기다린다. 승인컴포넌트로부터 승인요청이 들어오면 승인에플릿은 주어진 승인정보로부터 승인자에 대한 정보를 추출하여 승인이 가능한 승인자의 리스트와 승인 사유를 추출하여 웹브라우저에 출력한다. 승인요청자가 승인자를 선택하여 승인을 요청하면 승인자의 승인에플릿으로 클라이언트소켓을 생성하여 자신이 승인컴포넌트로부터 받은 승인정보를 보낸 후 응답을 기다린다. 승인자의 애플릿은 전송된 내용 중 승인사유 및 승인 요청자에 관한 정보를 웹브라우저에 출력하고 승인자는 승인여부를 선택

하여 응답한다. 승인요청에플릿은 승인자로부터 응답 받은 결과를 승인컨트롤러에 다시 응답한 후 승인웹브라우저에 승인의 결과를 출력해 준다. 승인 요청자는 승인자로부터의 결과를 볼 수 있을 뿐 승인후의 트랜잭션을 중지하거나 취소할 수 없다.

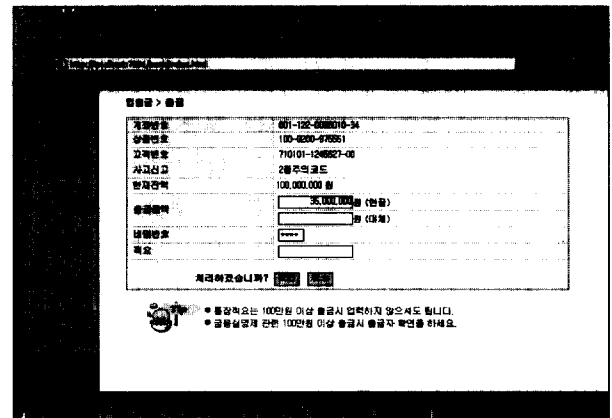
승인에플릿의 주요 메소드 별 기능을 정리하면 <표 7> 과 같다.

<표 7> 승인에플릿의 주요 메소드 및 기능

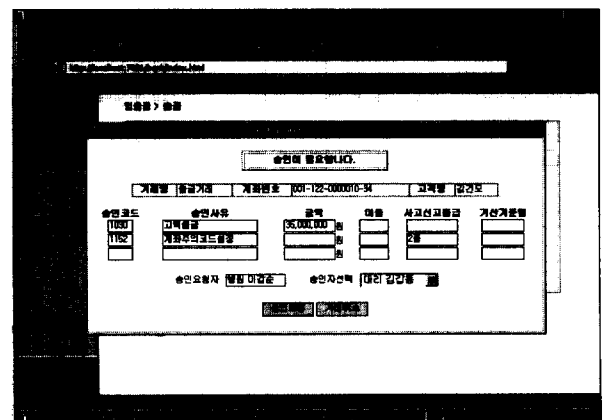
메소드명	주요 기능
init()	* 웹브라우저의 자바스크립트 오브젝트 생성
start()	* Server Socket 생성하고 승인요청 기다림. * 승인요청이 들어오면 승인컴포넌트로부터의 요청인지 다른 승인클라이언트로부터의 요청인지 확인하여 각각의 자바스크립트 함수 호출
requestApproval()	* 자바스크립트함수로부터 호출되는 메소드 * 승인요청자가 지정한 승인자로 client socket을 생성하고 승인정보를 전송. * 승인자로부터 응답을 받아 승인컴포넌트에 응답.

4.4 구현된 승인처리시스템

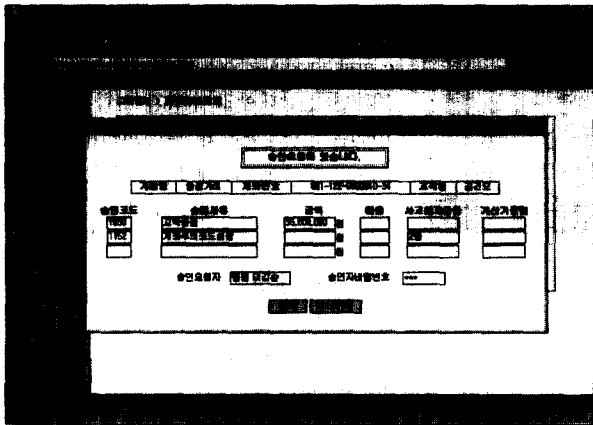
실제 구현된 승인처리시스템을 처리과정별 화면은 다음과 같다.



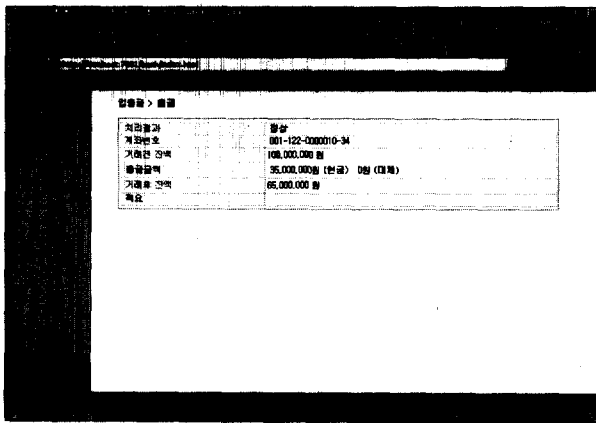
(그림 4) 출금트랜잭션 요청



(그림 5) 승인을 필요시 승인요청자에게 승인원도우를 별도로 띄우고 승인사유 출력



(그림 6) 승인요청자가 승인을 요청하면 승인사유를 승인자의 윈도우에 출력



(그림 7) 승인이 정상적으로 끝나고 거래종료

5. 결 론

컴포넌트 기반 개발방법은 응용프로그램 개발시 기존의 컴포넌트를 재 사용함으로써 보다 쉽게 개발할 수 있으며 이런 장점은 사회의 변화와 시장환경에 민감하게 대처할 수 있는 신속한 상품개발과 배포 기능, 용이한 유지보수성을 요구하는 banking시스템의 요건을 충족시킨다고 볼 수 있다. 또 복잡 다양해지는 응용프로그램으로 인한 클라이언트 프로그램의 배포 및 버전관리문제를 해결하기 위한 방법으로는 웹브라우저를 이용하는 웹클라이언트가 유력하게 제시되고 있으며, 실제로 HTML, JSP, ASP등을 이용한 많은 응용프로그램들이 웹환경을 기반으로 개발되고 있다. 하지만 비연결지향의 웹기반 클라이언트는 승인처리와 같이 트랜잭션의 처리 도중 사용자와의 상호작용에 따라 그 유효성이 결정되는 대화형 트랜잭션을 처리하는 경우에는 그 구현이 어렵다는 단점이 있다.

본 논문에서는 EJB banking컴포넌트와 그 내용을 소개하고 banking시스템에서 사용되는 승인처리시스템을 설계하고 구현하였다. 승인컴포넌트의 구현을 위하여 banking시스템에서 사용되는 승인사항들을 분류하여 이를 승인조건으로 등록하

였고, 등록된 승인조건을 이용하여 트랜잭션의 승인필요여부를 판단할 수 있도록 하였다. 승인사항을 별도의 컴포넌트로 관리함으로써 승인기준의 변화시에도 승인컴포넌트가 관리하는 승인조건을 변경하는 것만으로 승인과 관련된 클라이언트 프로그램의 수정 없이 변화된 기준을 적용할 수 있게 하였다. 또 웹기반으로 구현된 승인클라이언트는 웹환경으로의 클라이언트 표준화 원칙을 준수하고 클라이언트 프로그램의 배포 및 버전관리 문제를 해결하도록 하였으며 웹브라우저와 인증애플릿을 이용함으로써 웹 환경에서 대화형 트랜잭션을 처리할 수 있도록 하였다.

참 고 문 헌

- [1] 박기영, "Parameter Driven 방식과 금융정보기술", 정보처리학회지, Vol.7, No.5, pp.36-40, 2000.
- [2] 김재훈, "자바기반의 엔터프라이즈 이-뱅킹 컴포넌트 모델", 정보처리학회지, Vol.7, No.5, pp.51-57, 2000.
- [3] 안태광, 김병기, "EJB기반의 banking컴포넌트 시스템", 정보처리학회 추계학술대회, 제8권 제2호, pp.497-500, 2001.
- [4] 정승재, 김수동, "CBD와 EJB기반의 banking시스템 설계 및 구현", 정보과학회 학술대회, Vol.27, No.2-1, pp.451-453, 2000.
- [5] Kozaczynski Wojtek, Booch G. "Component-Based Software Engineering," IEEE Software, pp.34-36, Sept./Oct. 1998.
- [6] Erich Gamma, John Vlissides, Ralph Johns, Richard Helm. "Design Patterns," Addison-Wesley, 1995.
- [7] Mark Grand, "Pattern In Java," Vol.1, John Wiley & Sons, 1998.
- [8] Duane K. Fields & Mark A. Kolb, "Web Development with Java Server Pages," Manning, 1999.
- [9] Richard Monson-haefel. "Enterprise Java beans, 2E," O'Reilly, 2000.
- [10] Merlin Hughes, Michael Shoffner, Derek Hammer, Umesh Beller, "Java Network Programming," Manning, 2000.
- [11] Richard Monson-haefel & David A. Chappell, "Java Message Service," O'Reilly, 2001.
- [12] Deepak Alur, John Crupi, Dan Malks, "core J2EE patterns," Prentice Hall, 2001.
- [13] Craig Larman, "Applying UML and Patterns 2E," Prentice Hall, 2001.
- [14] James W. Cooper, "Java Design Patterns : A Tutorial," Addison-Wesley, 2001.
- [15] <http://www.javasoft.com>, "EJB2.0-Proposed Final Draft 2," 2001.
- [16] <http://e-docs.bea.com/wls/docs61/ejb/index.html>, "Programming WebLogic Enterprise JavaBeans," 2001.
- [17] <http://e-docs.bea.com/wls/docs61/ejb/pdf.html>, "Programming WebLogic JMS," 2001.



안 태 광

e-mail : tkahn@kjbank.com

1993년 조선대학교 전자계산학과(이학사)

1995년 전남대학교 대학원 전산통계학과
(이학석사)

1995년~1997년 ㈜광은경영경제연구소

1998년~현재 ㈜광주은행

관심분야 : 분산객체, 소프트웨어 컴포넌트



김 병 기

e-mail : btkim@chonam.chonam.ac.kr

1978년 전남대학교 수학과(이학사)

1980년 전남대학교 대학원 수학과(이학석사)

2000년 전북대학교 대학원 수학과(이학박사)

1981년~현재 전남대학교 컴퓨터정보학부
교수

1998년~2000년 전남대학교 정보통신연구소 소장

2000년~현재 전남대학교 자연과학대 학장

관심분야 : 소프트웨어 공학, CBSE, 병렬처리, 소프트웨어 에이
전트, 객체지향시스템