

협업을 지원하는 분산 페어 프로그래밍 시스템 설계/구현 및 검증

The Design, Implementation and Verification of Distributed Pair Programming System for Supporting Collaboration

노효원*, 박진호**, 곽훈성*

Hyo-Won Noh*, Jin-Ho Park**, and Hoon-Sung Gwak*

요 약

소프트웨어 개발의 주요 동향 중 하나는 소프트웨어 산업의 세계화에 있다. 글로벌 소프트웨어 개발은 여러 가지 문제에 직면해 있다. 이러한 문제를 해결하기 위해 새로운 소프트웨어 개발 방법론과 프로세스가 필요하다. 최근 XP 동향에서의 개발 단계로 본다면 기존의 개발자와 개발자간의 협업을 위한 도구에서 벗어나 조직 구성원 전체가 협업할 수 있는 시스템이 필요하다. 텍스트 기반의 IDE 플러그인 형태나 단순한 화면공유, 채팅 기능을 제공하는 것이 아닌 구성원 전체가 협업할 수 있는 분산 페어 개발을 지원하는 시스템의 설계하고 개발하였다.

Abstract

The dominant trend in software development is the globalisation of the software industry. This development is faced with diverse problems, which require solution by the adoption of new processes and development techniques. eXtreme Programming (known as XP) is one methodology which is now at the leading edge of software development. This recent trend in XP allows organisation members to cooperate towards the development of new software independently of the existing developers. This is achieved functionally between the members by the development of distributed pair programming, this is not IDE plug-in shape of text, simple screen sharing or chatting function based

Key words : Pair Programming, eXtreme Programming, XP, CDDP System, Distributed Pair Programming

I. 서 론

소프트웨어 산업이 세계적인 경쟁력을 갖기 위해서는 글로벌 소프트웨어 플랫폼을 확보하는 것이 무엇보다 중요하다. 소프트웨어 플랫폼은 일반적인공통의 것들을 미리 설계하고, 그 위에 고객의 요구사

항에 맞는 소프트웨어를 개발하기 때문에 비용과 기간을 절약할 수 있다. 이러한 소프트웨어 플랫폼을 개발하기 위해서는 공통적으로 활용이 가능한 부분을 디자인 할 수 있는 환경을 지원해 주는 것이 매우 중요하다[1].

클라우드 환경으로 통합되고 있는 소프트웨어 산

* 전북대학교 영상공학과 (Dept. of Image Engineering, Chon-Buk University)

** 경남대학교 컴퓨터공학과 (Dept. of Computer Science and Engineering, Kyungnam University)

· 제1저자 (First Author) : 노효원(Hyo-Won Noh, Tel : +82-10-2846-9579 email : flash200@hanmail.net)

· 접수일자 : 2013년 5월 31일 · 심사(수정)일자 : 2013년 5월 31일 (수정일자 : 2013년 6월 25일) · 게재일자 : 2013년 6월 30일

<http://dx.doi.org/10.12673/jkoni.2013.17.3.346>

업환경에서 국내 시장을 대상으로 개발한 소프트웨어는 글로벌 시장을 대상으로 개발된 소프트웨어를 이기기 어렵다. 우리 소프트웨어 기업들이 어플리케이션 소프트웨어 및 콘텐츠 글로벌 시장을 대상으로 개발할 수 있도록 글로벌리제이션을 강화할 필요가 있다. 글로벌 소프트웨어 플랫폼 확보를 위한 노력이 필요하다[1]. 애자일 소프트웨어 개발은 클라이언트와의 상호작용과 반복적인 개발을 통해 작동하는 코드를 생성하는 것이며, 기존의 계획 기반의 SW개발 방법의 대안으로 떠오르고 있다[2].

최근 구글, HP, 선마이크로시스템즈 등 세계적인 기업들도 소프트웨어의 근간이 되는 프로그램의 소스코드를 공개하는 오픈소스 개념을 활용하고 있다. 오픈소스는 다수의 기술자가 소프트웨어를 공동으로 개발하여 더 나은 소프트웨어를 단기간에 개발할 수 있게 한다. 애자일 소프트웨어 개발의 성공은 분산된 팀의 상호작용에 따라 달라진다[3].

분산 페어 시스템은 글로벌리제이션과 오픈소스 개념을 도입하여 세계적인 플랫폼 또는 소프트웨어를 개발하기 위한 환경을 제공할 수 있는 방안이다.

하지만 이를 지원하는 도구는 명확히 정립되어 있지 않고 사용자 각자가 원하는 기능의 어플리케이션을 활용하여 이용하는 경우가 대부분이다[4].

최근 XP(eXtreme programming) 동향에서의 개발 단계로 본다면 기존의 개발자와 개발자간의 협업을 위한 도구에서 벗어나 조직 구성원 전체가 협업할 수 있는 시스템이 필요하다. 기존의 분산 페어 프로그래밍 지원 시스템은 대다수 개발자를 위해 제작되었다. 최근 사용자경험이 중요시되면서 디자이너가 기획 및 UI설계, 개발 부분에 개발자와 함께 참여하고 이 두 그룹 사이에 소통하며 개발하는 영역의 비중이 커지고 있다. 이를 위해 디자이너와 개발자가 함께 공유하며 시스템을 설계하고 변경할 수 있도록 확장성 높은 협업 분산 페어 소프트웨어 아키텍처 모델을 제시하고, 화면 공유를 이용한 분산 페어 프로그래밍을 위한 시스템을 설계하고 구현한다.

II. 관련연구

2-1 페어 프로그래밍

페어 프로그래밍(Pair Programming)은 동일한 컴퓨터에서 나란히 두 명의 프로그래머가 공동으로 코드를 설계, 알고리즘, 코드, 테스트 등을 하는 것으로 면대면으로 협력해 상호작용하면서 완성해 가는 프로그래밍 방식이다. 두 명 중 한 명인 드라이버(Driver)는 컴퓨터 앞에서 타이핑을 하면서 알고리즘을 적는다. 상대인 네비게이터(Navigator)는 그것을 관찰하며 오류를 발견하거나 대안을 생각한다. 둘은 대화를 나누어 합의된 내용들을 구현한다. 중간에 드라이버와 네비게이터는 역할을 교체하기도 한다. 페어 프로그래밍은 개발 시 오류를 빨리 검출할 수 있으며 코드 완성도를 높이는 결과를 가져온다. 그림 1은 개발 코드 라인수의 차이를 보여주고 있다.

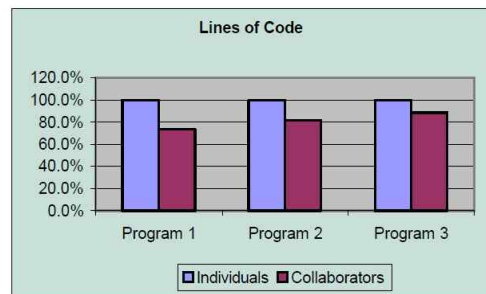


그림 1. 개발 코드 라인 비교[3]
Fig 1. Comparing of lines of code developed

2-2 분산 페어 프로그래밍

분산 페어 프로그래밍은 가상 페어 프로그래밍, 원격 제어 프로그래밍이라고도 하며, 네트워크로 연결된 사용자끼리 원격으로 페어 프로그래밍 하는 것을 뜻한다. 기존 연구에서는 분산 페어 프로그래밍이 페어 프로그래머의 생산성을 유지하면서 높은 품질의 코드를 생산하며, 높은 작업 만족도를 가지고 제품에 더 많은 자신감을 가지고 것을 발견했다[5],[6].

1) 분산 페어 프로그래밍 도구

지금까지 인터넷을 통한 협업을 위해 다양한 툴들이 사용되었다. 현재의 분산 페어 프로그래밍 도구는 크게 화면 공유 도구(Screen-Sharing Applications)와

협업을 지원(Collaborative Work Support)하는 도구로 분류할 수 있다[6]. 표 1은 이러한 도구를 분류하여 보여주고 있다.

분산 페어 프로그래밍을 이용한 개발 시 도구를 이용하여, 프로그램 개발자는 협업을 통한 원거리의 사용자들이 원하는 기능들을 설계 및 구현할 수 있다. 또한 이에 대한 접근권한을 받아 허가된 기능들의 속성정보를 사용하여 프로그램을 작성할 수도 있다.

표 1. 분산 페어 프로그래밍 도구 분류
Table. 1. Category For Distribute Pair Programming Tools

구분	도구명	
화면 공유	VNC pcAnywhere Milos ASE System	
협업 에디터	Non-DPP	MoonEdit SynchroEdit TalkAndWrite(Skype)
	DPP Designed:	COPPER SubEthaEdit TUKAN
IDE 기반	Non-DPP	Borland CodeWright Borland JBuilder DocSync Collab.NetBeans
	Distributed XP Support1 and DPP Designed:	Moombass MCIDE PEP Sangam Saros Xelicip Xpairtise

2) 분산 페어 프로그래밍 도구 요구사항

협업 및 분산을 지원하기 위한 도구 설계에 대한 요구사항을 Cox와 Greenberg는 모든 참가자에게 일반적이며, 시각적으로 유사한 환경을 제공하며, 작업 공간내에서 모든 작업에 대해 적시에 피드백 제공, 표시와 지시참조 지원, 작업공간에 대한 인식 지원해야 한다고 정의하고 있다[8]-[15].

3-1 기술 수용 모델

Davis et al. (1989)의 기술수용모델인 TAM (Technology Acceptance Model)은 사회 심리학 분야의 합리적 행위 이론(TRA: Theory of Reasoned Action)을 기반으로 한 모델로서 사용자의 행동을 통계적으로

분석하여 과학적으로 예측하려는 것으로 기존의 많은 기술 수용 연구의 근간이 되는 이론이며, 그림 2와 같이 나타낸다.

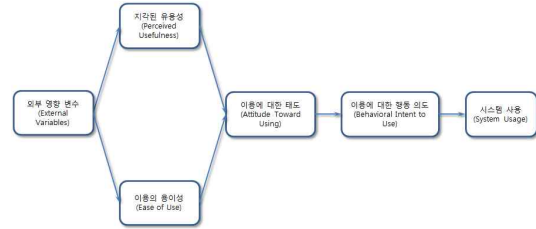


그림 2. TAM 모델(Davis, 1989)
Fig 2. TAM Model

정보기술에 대한 ‘지각된 이용용이성’을 향상시킨다는 것은 수단적인 부분을 향상시킨다는 것을 의미하고, 이러한 ‘이용용이성’이 향상되면 성과도 향상될 수 있다. 즉 정보시스템 이용자가 정보시스템을 쉽게 이용할 수 있도록 이용의 용이성을 향상시키면, 노력을 절감할 수 있고, 동일한 노력에서 더 높은 성과를 낼 수 있다. 지각된 이용용이성이 향상되면 성과를 향상시킨다는 점에서 지각된 이용용이성은 지각된 유용성에 직접적인 효과를 미칠 것이다. 그래서 TAM에서 지각된 유용성과 지각된 이용용이성은 별개이지만 관련된 구성개념으로 보고 있다(Davis et al., 1989).

III. DPP 시스템 설계

분산 페어 프로그래밍을 위한 안정성 및 보안성을 보완하고, 효율적인 데이터 교환 및 논의가 가능한 협업 지원 분산 페어 프로그래밍 시스템을 제안하고자 한다.

3-1 시스템 요구사항

지금까지 논의된 도구들이 분산페어에 미치는 요인을 도출하기 위해 분산 페어 프로그래밍 어플리케이션들을 분석하여 필수적인 요구사항들을 도출하였다[8]-[15].

- 입력장치 공유성

- 커뮤니케이션 기능
- 역할교환 및 제어
- 플랫폼 독립성
- 제한적 화면공유
- 데이터 전송효율
- 데이터 무결성 검사

3-2 시스템 설계

제안한 시스템을 GDPP System(Generic Distributed Pair Programming System)으로 명명하겠다. 그림 2는 제안한 협업 분산 페어 프로그래밍 시스템의 기본 아키텍처이다. 분산 페어 프로그래밍에서 사용자는 드라이버와 네비게이터로 구분한다.

DPP System(Distributed Pair Programming System)은 레이어별로 협업모듈, 분산 페어 프로그래밍 기능 그리고, GUI로 구성된다.

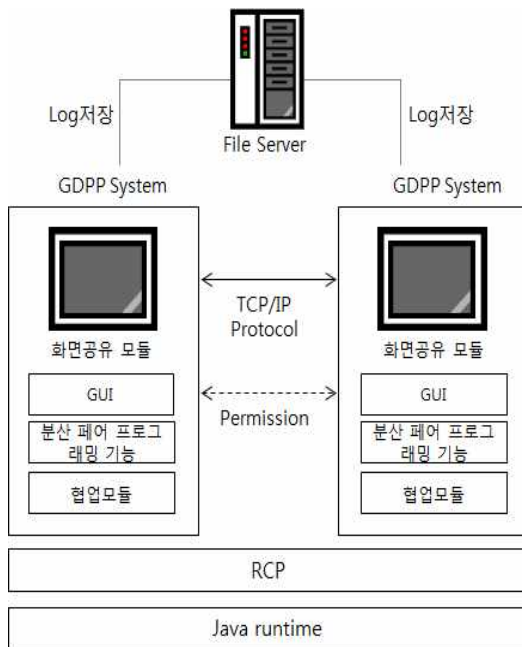


그림 2. 시스템 아키텍처
Fig. 2. The CDPP system for collaboration

협업 분산 페어 프로그래밍 시스템의 구조도는 그림 3과 같다. 제안한 협업 분산 페어 프로그래밍 시스템은 기본적으로 자바와 Rich Client Platform을 기반으로 동작한다. 이는 시스템의 독립성과 구현에 필요한 다양한 클래스를 제공하며, 각각의 이벤트의 액

션을 실행하고, UI를 구성하도록 위함이다. Collaborative Module(협업 모듈)은 분산 페어 프로그래밍에 필요한 다양한 세부 모듈들을 포함하고 있다.

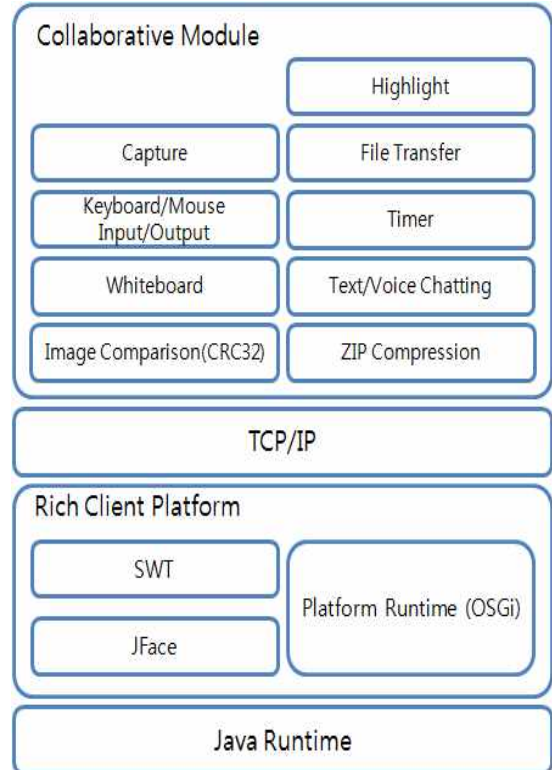


그림 3. 시스템 구조도
Fig 3. The CDPP System Architecture

협업 모듈은 제안한 협업 분산 페어 프로그래밍 시스템의 핵심 요소이며 분산 페어 프로그래밍 시 사용자 간의 원활한 커뮤니케이션, 시스템의 활용, 그리고, 프로그램 개발을 위해 다양한 기능을 제공한다. 총 9개의 세부 모듈로 구성되며, 세부 모듈의 항목은 다음과 같다.

- ① 지정화면 공유모듈
- ② 역할 교환 모듈
- ③ CRC32 검사 및 ZIP압축 모듈
- ④ 하이라이트 모듈
- ⑤ 화이트 보드 모듈
- ⑥ 파일전송 모듈
- ⑦ 타이머 모듈
- ⑧ 음성채팅 모듈
- ⑨ 문자채팅 모듈

세부 모듈의 모든 모듈은 분산 페어 프로그래밍

시 중요한 기능들을 담당하고 있으며, 개별의 스프레드 로 동작한다.

표 2 협업 모듈 명세

TABLE 2. Functions For Collaborative Module

ScreenShare()	드라이버와 네비게이터의 화면 공유 화면을 고속으로 캡처 하고 분할하여 전송 드라이버/네비게이터로 부터 들어온 키보드, 마우스 데이터를 받아 입출력이 가능
Timer()	지정된 시간이 지나면 역할이 바뀔 수 있도록 키보드, 마우스 입출력을 제어
FileTrans()	파일 전송 기능을 담당하고 있으며, 파일을 선택하여 송수신이 가능하다.
Highlight()	드라이버와 네비게이터로부터 입력된 자료 값에 하이라이트를 표시
WhiteBoard()	투명 레이어를 생성하는 부분과 펜 도구로 구성
Zip_CRC32()	무결성 검사 및 이미지를 ZIP를 이용하여 압축 생성된 캡처 이미지 바이너리를 CRC32로 체크값을 얻어내어 이전에 생성된 이미지의 체크값과 비교
VoiceChat()	컴퓨터에서 사용할 수 있는 마이크 장치를 선택 후 녹음하여 송/수신된 음성은 스피커로 출력
TextChat()	키보드 장치를 선택 후 드라이버 또는 네비게이터로부터 입력받은 텍스트 메시지 전송
RoleChange()	드라이버와 네비게이터간의 역할교환 역할교환 요청을 통해 상대방의 시스템 및 입출력 장치를 제어가능

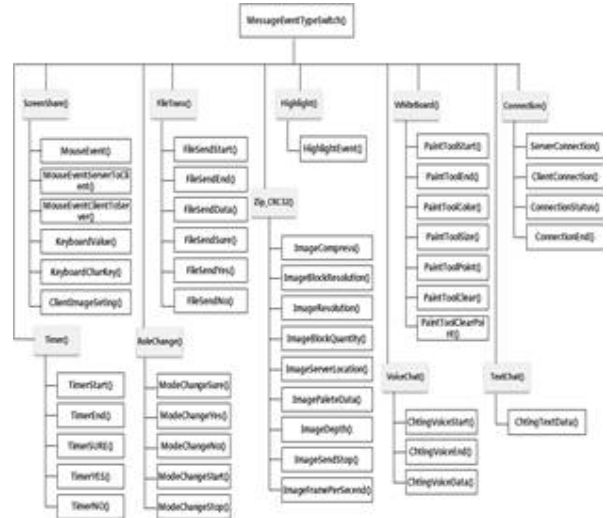


그림 4. 협업 분산 페어 프로그래밍 시스템의 세부 모듈별 구조도

Fig 4. Major Functional Structure of the CDPP System

IV. 협업 분산 페어 프로그래밍 시스템 구현

4-1 시스템 구조

구현된 협업 분산 페어 프로그래밍 시스템의 메소드들을 각 기능별 단위로 분류한 구조도는 그림 4와 같다.

4-2 시스템의 프로토타입 구현

세부 모듈에 대한 항목들을 구현하였으며 일부 구현 화면은 아래와 같다.

① 드라이버&네비게이터 UI

드라이버와 네비게이터는 각각의 프로그램에서 Server는 Port 번호를 지정하고 네비게이터에서는 드라이버의 IP와 Port 입력을 통하여 접속 가능하도록 구현하였다. 초기 접속화면은 그림 5와 같다.



그림 5. 드라이버&네비게이터 UI
Fig 5. Driver&Navigator UI



그림 6 CDPP 시전 화면
Fig 6. CDPP running Screen

② 드라이버의 컨트롤을 위한 트레이 아이콘 화면 캡처가 시작하게 되면, 백그라운드에서 작동 되기 때문에 프로그램을 컨트롤할 수 있는 방법이 없어진다. 이를 해결하기 위해 Server는 트레이 아이콘을 생성하여 여러 작업을 가능하도록 구현하였다.

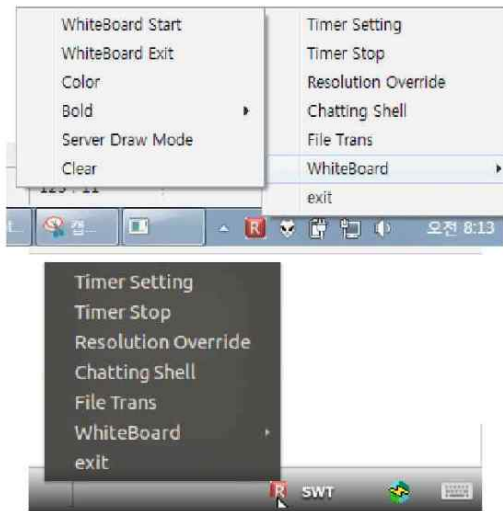


그림 7 윈도우와 리눅스 드라이버 트레이 아이콘
Fig 7. Driver Tray Icon For Windows and Linux

③ 하이라이트 기능

네비게이터가 드라이버의 화면에 잘못된 부분이나 수정이 필요한 부분에 대한 내용을 네비게이터가 지정하여 표기할 수 있도록 구현된 기능이다. 클라이언트에서 전송 받는 View 화면에 Click 이벤트를 발생시키면 서버의 화면의 해당 좌표에 이벤트가 발생하도록 구현하여 클라이언트의 피드백 기능을 제공한다.

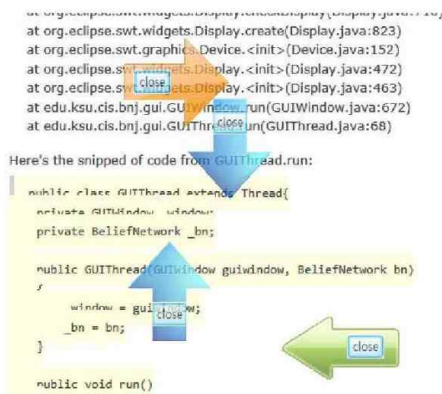


그림 8 하이라이트 기능
Fig 8. Function of Highlight

4-3 기술수용 모델에 의한 평가

단순한 협업 분산 페어 도구가 아닌 사람중심의 XP방법론의 관점에서 시스템을 평가해야 할 필요성이 있다. 이를 위해 인지된 유용성과 인지된 용이성

에 영향을 주는 요인들을 정의하고 평가하였다.

Davis(1986)가 1986년 제안한 기술수용모델은 사용자의 정보기술 수용과 사용행동을 설명하는 간단한 해서 설명력이 높은 모델로 인정받고 있다.

1) 변수정의

연구모형의 독립변수는 공유성, 단순성, 커뮤니케이션 기능, 보안성, 효율성으로 정의하였고, 매개변수는 지각된 유용성, 지각된 사용용이성, 이용의도로 정의하였다.

종속 변수는 업무의 성과를 나타내며, 5개의 독립변수와 3개의 매개변수에 의해 분산 페어 프로그래밍 도입에 의한 업무 성과에 대한 종속변수의 관계를 파악 및 도출한다.

2) 표본선정

표본 선정은 무작위 선발에 의한 표본을 선정하였다. 표본 선정의 지역으로는 수도권 지역은 서울 및 일산 지역의 산업체와 대학을 대상으로 하며, 지방권에서는 광주광역시의 산업체와 연구소 및 대학을 대상으로 표본을 선정하였다

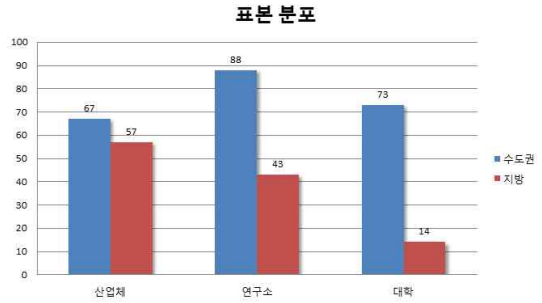


그림 9 업종별 표본 분포의 차트
Fig 9. Sampling Distribution by industry

3) 분석결과

설문분석 결과 그림 10에서 보여지듯이 매개변수와 관련이 있는 변수로는 공유성, 단순성, 커뮤니케이션 기능으로 나타났다. 효율성, 보안성 부분에 취약한 것으로 파악된다.

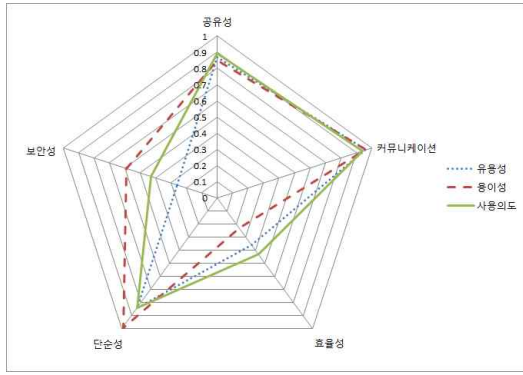


그림 10 독립변수에 의한 매개변수의 추정
Fig 10. Estimates of the parameters by the independent variable

4) 도구평가

정의한 요구사항을 기반으로 하여 기존 분산 페어 프로그래밍 지원 도구와 제안한 시스템의 비교 내용은 표 3과 같다. 대부분의 분산 페어 프로그래밍 도구들은 지정화면 공유기능, 무결성 검사, 전송 속도에 대한 평가가 이루어지지 않고 있다. 화면 공유를 통한 분산 페어를 지원하는 도구들은 화면 공유를 위한 연속된 이미지의 전송에도 불구하고 이를 위한 대처방안이나 알고리즘이 없어 속도가 느려지는 단점이 있다. IDE 기반 도구들은 플러그인 형식으로 설치를 해야 하면 지정 프로그램만 공유하기 때문에 확장성이 낮으며, 프로그래머 간의 코드 공유만 가능하다.

표 3. 분산 페어 프로그래밍 도구 평가
TABLE 3. tool evaluation for DPP

요구사항 도구명	장치 제어	커뮤니케이션 기능	역할 교환	플랫폼 독립적	지정 화면 공유	전송 속도	무결성 검사
Sangam	○	하	○	○	X	X	X
RIPPLE	X	하	X	○	X	X	○
COLLEGE	○	하	○	○	X	X	X
COPPER	○	중	○	○	X	X	X
TUKAN	○	상	○	○	X	X	X
Moomba	○	상	○	○	X	X	X
Xpairtise	○	상	○	○	X	X	X
VNC4DPP	○	하	○	○	X	X	X
제안한 GDPP System	○	상	○	○	○	○	○

V. 결 론

관련 연구 결과를 통해 요구 조건을 도출 하고 이를 토대로 분산 페어 프로그래밍을 위한 시스템을 설계 및 구현함으로써, 기존의 파편화되었던 개발 환경을 정리하여 분산 페어 프로그래밍의 효율을 증진할 수 있게 되었다. 이에 따라 해외에 비해 다소 활성화 되지 않은 분산 페어 프로그래밍의 활성화가 기대된다. 또한 의사소통을 핵심으로 삼고 있는 애자일 모델의 네 가지 가치 - 소프트웨어 개발자들 간의 상호 작용, 작동하는 소프트웨어, 고객과의 협력, 변화에 대응 - 만으로도 적용할 가치는 충분하다.

향후 추가적인 연구 과제로 분산 페어 프로그래밍에 필요한 추가 기능과, 전송되는 패킷에 대한 암호화 처리, 화면 캡처 및 전송에 대한 오버헤드를 줄이는 연구가 필요하다.

Reference

- [1] National IT Industry Promotion Agency. Weekly Technical Trends. Available: http://www.nipa.kr/know/periodicalWebzineView.it?identifier=02-001-130604-000008&menuNo=28&code=B_ITA_01&page=1
- [2] Vijayasarathy, Leo, and Dan Turk. "Drivers of Agile Software Development Use: Dialectic Interplay Between Benefits and Hindrances", *Information and Software Technology*, Vol. 54, Issue 2, pp. 137-148, Feb. 2012.
- [3] Dorairaj, Siva, James Noble, and Petra Malik. "Understanding Team Dynamics in Distributed Agile Software Development", *Agile Processes in Software Engineering and Extreme Programming. Springer Berlin Heidelberg*, Vol. 111, pp. 47-61, May 2012.
- [4] McIntosh, S. Adams, B. Nguyen, T.H.D. Kamei, Y. Hassan, A.E, An empirical study of build maintenance effort, in *33rd International Conference on Software Engineering, Canada*, pp. 141-150, 2011.
- [5] A. Cockburn and L. Williams. "The Costs and Benefits of Pair Programming", *In Extreme Programming*

Examined, ACM SIGCSE Bulletin 33, No. 1, pp. 327-333, 2001.

- [6] L. Williams, R. R. Kessler, W. Cunningham, and R. Jeffries. "Strengthening the Case of Pair Programming", *In IEEE Software*, Vol. 17, No. 4, pp. 19-25, Aug. 2000.
- [7] D Winkler, S Biffl, A Kaltenbach, Evaluating Tools that Support Pair Programming in a Distributed engineering Environmet, *Conference on Evaluation and Assessment in Software Engineering (EASE)*, pp. 54-63, UK, 2010.
- [8] Cox, D. Greenberg, Supporting collaborative interpretation in distributed groupware, *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, pp. 289-298. NY, 2000.
- [9] Gutwin, C. Greenberg, "The Effects of Workspace Awareness Support on the Usability of Real-time Distributed Groupware", *ACM Transactions on Computer-Human Interaction*, Vol. 6, No. 3, pp. 243-281, Sept. 1999.
- [10] Wanfeng Dou, Wei He, A preliminary design of distributed pair programming system, *Second International Workshop on Education Technology and Computer Science*. Wuhan, pp. 256-259, Mar. 2010.
- [11] Ho, Chih-Wei, et al, Sangam: a distributed pair programming plug-in for eclipse, *in Proceedings of the 2004 OOPSLA workshop on eclipse technology eXchange*, NY, pp. 73-77, 2004.
- [12] Elizabeth. K, Dwight. A et al. A development environment for distributed synchronous collaborative programming, *in Proceedings of the 13th annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, NY, pp. 158-162, 2008.
- [13] B. Hanks, "Empirical Evaluation of Distributed Pair Programming", *International Journal of Human-Computer Studies*, Vol. 66, No. 7, pp. 530-544, July 2008.
- [14] T. Schummer, S. Lukosch, "Understanding Tools and Practices for Distributed Pair Programming", *Journal of Universal Computer Science*, Vol. 15, No. 16, 2009.
- [15] Hanks B. F, Empirical Studies of Distributed Pair Programming. *Ph.D. dissertation, University of California Santa Cruz, CA, USA*, 2005.

노 효 원(Hyo-Won Noh)



2002년 2월 : 호남대학교 멀티미디어공학과 공학석사
 2005년 3월 ~ 현재 : 전북대 영상공학과 박사과정
 2011년 3월 ~ 현재 : 호남대학교 인터넷 콘텐츠학과 초빙교수

관심분야 : 영상처리, Web3D, 웹서비스, UI&UX

박 진 호(Jin-Ho Park)



2006년 2월 : 경남대학교 컴퓨터공학과 공학석사
 2010년 1월 ~ 2011년 1월 : The City University of New York, USA, Visiting Student
 2011년 8월 : 경남대학교 컴퓨터공학과 공학박사

2012년 1월 ~ 현재: 한솔그룹 (주)솔라시아 책임연구원
 관심분야 : 스마트카드, 자비카드, 임베디드 시스템, 소프트웨어 개발 방법론

곽 훈 성(Hoon-Sung Gwak)



1979년 : 전북대학교 전자공학과 박사
 1981년~1982년 : 미국 텍사스 주립대학교 연구교수
 1998년 : 과학기술법령정비정책위원
 1994년~1995년 : 국가교육연구 전산망 추진위원
 1999년~현재 : 조달청우수제품(정보통신)

심사위원

1997년~현재 : (사)영상산업연구센터 대표
 현재 : 전북대학교 전자정보공학부 컴퓨터공학 명예교수
 관심분야 : 영상신호처리, 인공지능, 컴퓨터비전, 멀티미디어