
웹 공간의 확장

공헌택¹, 고성범^{1*}

¹공주대학교 컴퓨터공학부

Expansion of the Web Space

Heon-Tag Kong¹, Sung-Bum Ko^{1*}

¹Division of Computer Science & Engineering, Kongju University

요약 현재의 웹은 표준화된 통합 환경을 지원하지 않는다. 이로 인해 작업 규모가 클 경우 작업 효율을 떨어뜨리는 원인이 될 수 있다. 본 논문에서는 복수의 행위 주체들이 함께 일을 할 때 서로의 일처리 프로세스를 공유하는 것이 가능한 통합 공간(SB공간)을 제안한다. SB공간은 웹 사이트의 하나로 설치될 수 있으며, 웹 공간과의 효율적인 연계사용이 가능하다. 이는 사실상 웹 공간의 확장을 의미하는 바, 이를 통해 웹 공간의 생산성을 현저하게 개선할 수 있다. 본 논문에서는 구체적인 사례 연구를 통해서 이를 입증해 보인다.

• **주제어** : 통합 환경, SB공간, 프로세스 공유, 웹, 공간

Abstract Current Web space does not provide standard integrated environment. This, especially in case of large task, can be a cause of low performance of task processing. We proposed an integrated space(called SB space) in this paper through which various actors can share their mutual process in processing a task together. SB space can be installed as a Web site and can be used effectively with Web space. This actually means the expansion of the Web space, which allows us to improve the productivity of the Web space notably. In this paper, this is proved through a concrete case study.

• **Key Words** : Integrated Environment, SB space, Process sharing, Web, Space

1. 서론

현재의 웹은 일처리를 위한 표준화된 통합 환경을 제공하지 않는다. 그 이유는 웹을 개발할 당시 그 목표가 정보 공유에 집중되어 있었기 때문이다[1]. 이로 인해, 특히 규모가 큰 작업의 경우, 적지 않은 불편이 따른다. 예를 들어, 서로 다른 작업 환경에서 생성된 일처리 결과들이 자연스럽게 연계되지 못한다.

웹 환경을 개선하려는 주목할 만한 시도의 하나로 웹 2.0을 들 수 있다[2][3]. 웹 2.0을 통해서 주로 공유, 개방, 집단 지성 같은 개념들이 추구되고 있는 데, 이들 중에 통합 환경 개념은 포함되어 있지 않다.

최근 second life 공간이 웹의 보완으로 개발되었으며

그 주민이 800만에 이를 정도로 인기를 끌고 있다[4]. 그러나 second life의 목표는 3차원 가상공간을 지원하는 데 있고, 일처리를 위한 통합 환경은 여기서도 여전히 간과되고 있다.

본 논문에서는 다수의 행위 주체들이 모여서 효율적으로 상호작용 할 수 있는 통합 환경(SB공간)을 제안하였다. SB공간 모델은 기존의 CPM 모델을 개선한 것이다[5]. SB공간은 웹 사이트의 하나로 설치할 수 있으며, 웹 공간과의 효율적인 연계 사용이 가능하다. 즉, 하나의 일을 다수가 공유하는 영역과 나머지 영역으로 구별하고, 각각을 SB공간과 웹 공간에서 나누어 처리하면 효율적인 분업이 가능해진다. 이 방법으로 다수가 참여하는 일

*교신저자 : 고성범(sbko@kongju.ac.kr)

처리 과정에 대한 실시간적 프로세스 공유가 가능해진다. 본 논문에서는 구체적인 사례 연구를 통해서 이를 입증해 보인다.

본 논문의 구성은 다음과 같다. 우선 2장에서는 SB공간의 구조와 기능을 소개한다. 3장에서는 SB공간을 구체적인 사례에 적용해 보인다. 4장에서는 SB공간이 갖는 특징을 분석한다. 5장에서는 SB공간을 평가한다. 마지막으로 6장에서는 결론과 함께 향후 과제를 살펴본다.

2. SB공간

2.1 SB공간

지구의 모든 사물들이 SB(Smart Block)라고 부르는 표준 형식으로 변신했다고 가정하자. 이렇게 바뀐 지구를 인터넷상에서 구현한 것을 SB공간이라고 부른다. 즉, SB공간은 SB라는 단위 요소들로 구성된 추상화된 지구이다.

2.2 지구_SB

이처럼 SB공간에서는 모든 사물을 SB화 한다. SB화된 사물은 당연히 SB를 통해서만 상호 작용이 가능하다. SB화된 사물은 이름 뒤에 ‘_SB’를 붙여서 원래의 사물과 구별한다. 예를 들어, 지구를 SB화하면 ‘지구_SB’가 된다.

2.3 지구_SB와 SB공간

SB공간은 추상화된 지구이며, 지구_SB가 바로 SB공간이 된다. SB공간이 완전하게 구현될 경우 지구의 모든 사물들은 SB공간에도 있어야 한다.

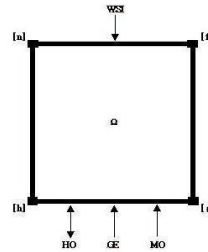
2.4 SB의 구조

SB는 서비스 단자, 탐색 단자, 제어 단자 등 세 그룹의 연결단자와 한 개의 서비스 창으로 구성된다. 그림 1은 SB의 구조를 보여준다.

(1) 제어 단자 : HO, GE, MO

특정한 SB와 상호작용 하려면 그 SB를 방문해야 한다. 방문을 위해서는 인증 과정이 필요하다. 방문자는 주인(Host)과 고객(Guest)으로 분류할 수 있는 데, 주인에 대응하는 인증 단자가 HO이고, 고객에 대응하는 인증 단

자가 GE이다. MO는 SB의 내부 정보를 입력하거나 수정하는 단자이다.



[Fig. 1] SB structure

(2) 출력 단자 : HO

HO는 정보신호 출력 단자이다. 즉, HO단자에서는 (1)에서 언급한 <Host 인증>과 여기서 소개한 <정보신호 출력> 등 두 가지 기능을 동시에 수행한다.

(3) 서비스 단자 : WSI

WSI는 서비스 단자이다. WSI 단자는 외부의 SB들과 서비스를 주고받을 때 사용한다.

(4) 탐색 단자 : [f], [n], [h], [g]

하나의 SB는 [f], [n], [h], [g] 등 4개의 탐색 단자(즐거찾기 단자)를 갖고 있다. 첫째, 가족단자인 [f]는 ‘가족 SB’ 들을 부를 때 사용한다. 둘째, 이웃단자인 [n]은 ‘이웃 SB’들을 부를 때 사용한다. 셋째, 역사단자인 [h]는 ‘역사 SB’를 부를 때 사용한다. 여기서 역사 SB란 SB공간에서 이루어진 작업의 역사를 SB 형식으로 표현한 것이다. 넷째, 보증 단자 [g]는 ‘보증인 그룹’을 부를 때 사용한다. 여기서 보증인 그룹이라 함은 SB를 법적, 윤리적, 경제적으로 보증해주는 SB들을 말한다.

(5) 서비스 창 : Q

SB의 내부 공간을 ‘서비스 창’이라고 부르며 ‘Q’로 표현한다. SB는 외부에 서비스를 지원하거나 외부로부터 서비스를 받을 때 서비스 창을 이용한다. SB가 지원하는 중요한 서비스 중 하나가 웹 서비스이다. 이 경우 서비스 창에는 웹 페이지가 나타난다.

2.5 Task SB와 SB Network

Task SB는 작업 전용 SB로, 그 안에 여러 SB들이 모여서 주어진 작업을 함께 수행한다. Task SB 안에 모인

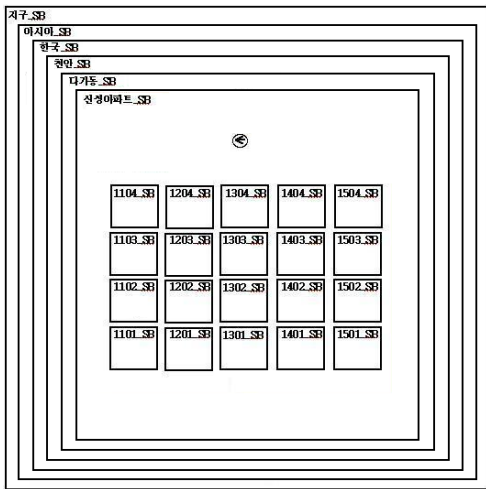
SB들이 작업을 시작하면 SB들 간에 상호작용이 일어나고, 상호작용의 결과로 수많은 연결선들이 생성된다. 이렇게 만들어진 SB간의 연결망을 'SB Network'라고 부른다. SB공간에서의 작업이란 SB Network를 구축하는 과정으로 정의된다.

3. 사례 연구

3장에서는 2장에서 정의한 SB공간을 실제 문제에 적용해 본다.

3.1 SB공간의 탐색

SB공간에서는 특정한 SB와 그 자손들을 표현한다. 이 특정한 SB를 '키 SB'라고 부른다. 또한 자손들을 몇 대까지 표현하느냐로 '표현차원'을 정의한다. 그림 2는 영구가 사는 신성아파트를 SB공간에서 표현한 것인데, 이 경우의 키 SB는 '신성아파트_SB'이고 표현차원은 1이다. SB공간내의 탐색은 그 자체로 흥미로운 주제이지만 본 논문에서는 이 문제를 깊이 다루지 않는다.



주) (사람 아이콘) : 사용자 아바타

[Fig. 2] SB permanent home in space

3.2 Task_SB

Task SB는 작업을 위한 전용 SB이다. 사용자 아바타를 더블클릭하는 방법으로 언제든지 Task SB를 생성할 수 있다. Task SB는 다른 SB와 구별하기 위해 테두리를 이중 네모로 표현한다.

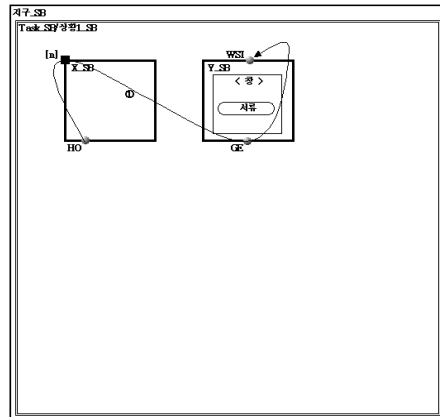
Task SB는 한 개 이상의 상황 SB들로 구성되고, 상황 SB는 한 개 이상의 장면 SB들로 구성된다. 장면 SB는 Task SB 내에서 화살표로 표현하며 생성순서에 따라 일련번호가 매겨진다.

모든 작업은 상황 SB 단위로 수행된다. Task SB가 생성되면 상황1_SB가 자동으로 따라오며 이후 상황2_SB, 상황3_SB 순서로 이어진다.

3.3 Task_SB 에서의 상호작용

그림 3은 Task_SB 내에서, SB간의 전형적인 상호작용 패턴을 보여준다. 번호 ①은 첫 번째 장면이라는 의미이다. [n]은 즐겨찾기 단자의 하나로 이웃관계에 있는 SB를 찾을 때 사용한다. WSI는 서비스 단자를 뜻하고, 서비스는 서비스 창을 통해서 이루어진다. 서비스 창에서는 주로 웹 기능을 지원한다. 즉, 대부분의 세부 작업은 웹 공간에서 수행된다. 그림 3의 상호작용을 단계별로 요약하면 다음과 같다.

- ▶ X_SB의 사용자는 HO단자를 통해서 Host 인증을 한다. Host 인증을 통해서 X_SB의 사용자는 X_SB를 사용할 권리를 얻는다.
- ▶ X_SB의 사용자는 Y_SB의 GE단자를 통해서 Guest 인증을 한다. 여기서 Guest 인증이란 Y_SB에 대한 방문 요청을 의미한다.
- ▶ Y_SB는 Guest 인증을 통과한 방문자에 대해서 서비스 단자인 WSI를 열어준다.
- ▶ X_SB의 사용자는 Y_SB의 WSI단자를 통해서 Y_SB가 지원하는 서비스를 받는다.
- ▶ 두 SB간의 상호작용 과정에서 생성된 서류는 서비스 창에 게시된다.



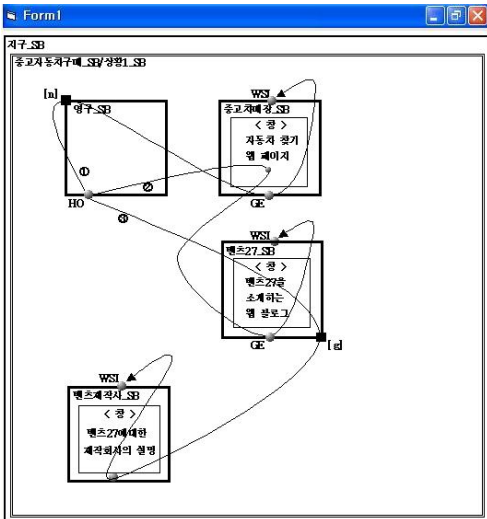
[Fig. 3] Task_SB interactions

3.4 적용

여기서는 SB공간을 구체적인 사례에 적용해 보인다. 적용 도메인은 ‘중고자동차 구매’이다. 즉, 영구가 SB공간에서 중고자동차를 구매한다고 가정한다. 우리는 SB공간의 작동을 연속적인 시각으로 관찰하기 위해서 일종의 시뮬레이터를 구축하였다. 이를 위해 90장 정도의 이미지가 소요되었고, 언어는 Visual Basic을 사용하였다.

3.4.1 상황 1

그림 4는 첫 번째 작업 상황 즉, 상황 1을 보여준다. 상황 1에서 영구는 마음에 드는 자동차를 찾아내고, 찾은 자동차에 대해서 조사한다. 상황 1을 장면 단위로 요약하면 다음과 같다.



[Fig. 4] Interaction Scenario 1

[장면 ①]

영구는 중고차매장_SB를 SB공간으로 불러낸다. 중고차매장_SB는 자신의 서비스 창에서 웹 공간의 자사 홈페이지를 띄운다. 영구는 서비스 창을 확대한 다음, 이곳에서 구매할 자동차를 찾는다(이곳은 SB공간이 아니라 웹 공간이다).

[장면 ②]

영구는 벤츠27을 찾아내고, 벤츠27_SB를 SB공간에 불러낸다. 벤츠27_SB는 서비스 창에서 벤츠 소유자가 만든 웹 블로그를 띄운다. 영구는 자동차 소유자의 웹 블로

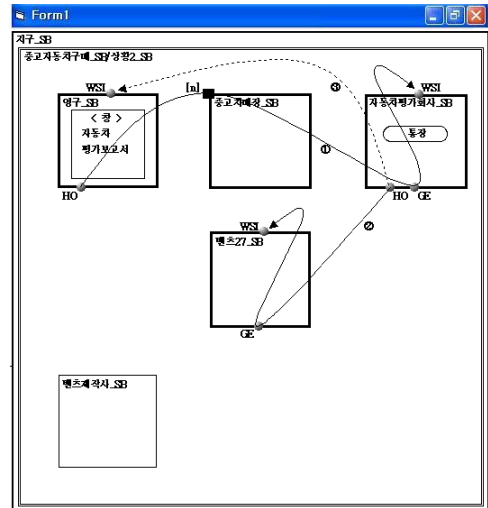
그에서 벤츠27에 대한 상세한 정보를 얻는다(웹 블로그 역시 웹 공간이다).

[장면 ③]

영구는 벤츠27_SB를 만든 벤츠제작사_SB를 불러서 벤츠에 대한 제작사 측의 설명을 듣는다. 여기서 벤츠제작사_SB는 벤츠27_SB에 대한 보증인 중 하나이다.

3.4.2 상황 2

그림 5는 상황 2를 보여준다. 상황 2에서 영구는 중고자동차 평가 회사를 불러서 구매할 중고 자동차에 대한 평가를 의뢰한다. 상황 2를 장면 단위로 요약하면 다음과 같다.



[Fig. 5] Interaction Situation 2

[장면 ①]

영구가 자동차평가회사_SB를 불러서 벤츠27에 대한 평가를 의뢰하고 자동차평가회사_SB의 통장을 통해서 수수료를 지불한다.

[장면 ②]

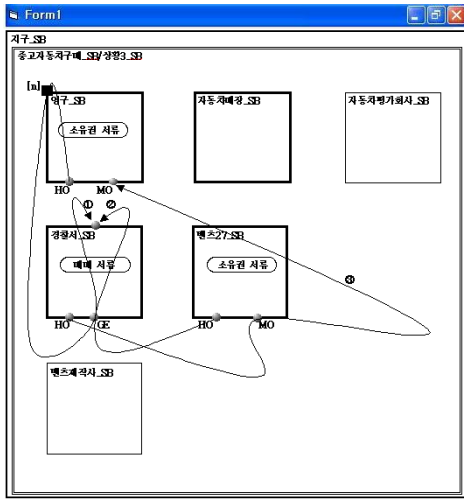
자동차평가_SB는 벤츠27_SB를 방문하여 벤츠27에 대해서 상세하게 조사한다.

[장면 ③]

자동차평가_SB는 평가 결과를 평가 의뢰자인 영구_SB의 서비스 창을 통해 전달한다.

3.4.3 상황 3

그림 6은 상황 3을 보여준다. 상황 3에서는 구매자인 영구와 판매자인 영철이가 매매 서류를 경찰서에 제출하고, 경찰서에서는 매매 서류에 근거하여 자동차의 소유권을 변경시킨다. 상황 3을 장면 단위로 요약하면 다음과 같다.



[Fig. 6] the interaction of the situation 3

[장면 ①]

영구는 경찰서_SB의 서비스 창에 들어가서 매매 서류를 작성한다.

[장면 ②]

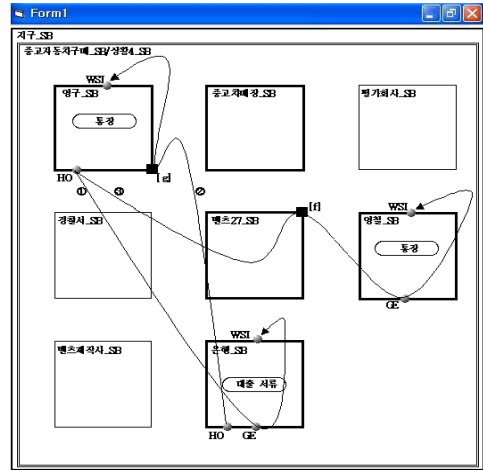
벤츠27_SB의 주인인 영철이가 경찰서_SB의 서비스 창에 들어가서 매매서류를 작성한다.

[장면 ③]

경찰서_SB가 벤츠27_SB와 영구_SB 안에 소유권 서류를 저장한다. 이로서 벤츠27의 소유권이 영철에게서 영구로 넘어간다.

3.4.4 상황 4

그림 7은 상황 4를 보여준다. 상황 4에서 영구는 거래 은행에서 자금을 대출받고, 이 돈으로 영철에게 구매 대금을 지불한다. 상황 4를 장면 단위로 요약하면 다음과 같다.



[Fig. 7] Interaction Situation 4

[장면 ②]

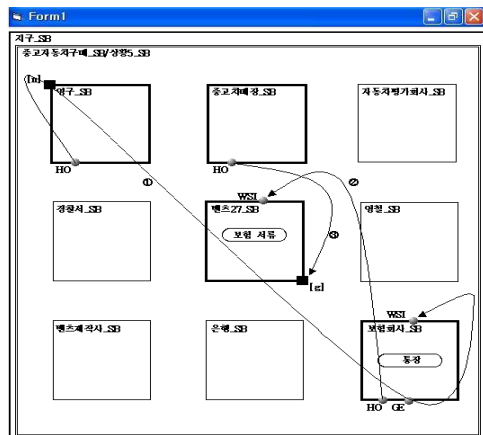
은행_SB는 영구_SB의 보증인 단자를 통해서 보증인_SB들을 조사하고, 영구_SB의 통장을 통해서 자금을 대출해 준다.

[장면 ③]

영구는 벤츠27의 소유자인 영철_SB를 불러서, 통장을 통해 자동차 구매 대금을 지불한다.

3.4.5 상황 5

그림 8은 상황 5를 보여준다. 상황 5에서 영구는 자동차 보험에 가입하고, 중고자동차 매장은 벤츠27_SB의 새로운 보증인으로 등록한다. 상황 5를 장면 단위로 요약하면 다음과 같다.



[Fig. 8] Interaction situation 5

[장면 ①]

영구_SB는 보험회사_SB를 불러서, 통장을 통해 보험료를 지불하고 보험에 가입한다.

[장면 ②]

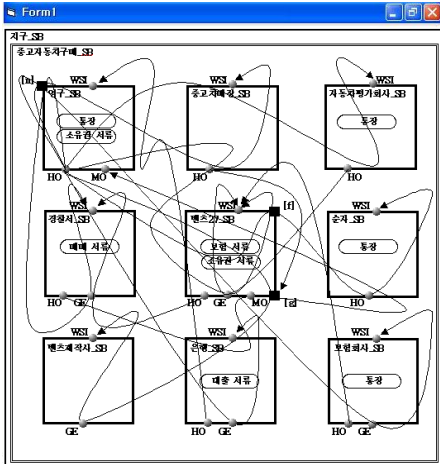
보험회사_SB는 벤츠27_SB의 서비스 창을 통해 보험서류를 전달한다.

[장면 ③]

중고차매장_SB는 거래가 끝났음을 확인하고, 벤츠27_SB의 새로운 보증인으로 등록한다.

3.4.6 Task SB의 생성

상황5_SB를 끝으로 모든 작업이 종결되었다. 작업이 종결되면 Task SB가 자동으로 생성된다. Task SB는 상황 SB들의 단순한 결합이다. 예를 들어, 본 사례의 Task SB는 5개의 상황 SB들로 구성된다. 그림 9는 생성된 Task SB이다



[Fig. 9] Task SB (home purchases _SB) generation

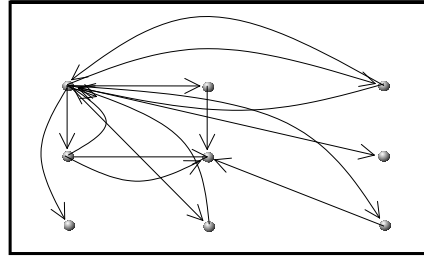
4. 분석

본 장에서는 3장에서 보여준 자동차 구매 작업을 분석해 보인다.

4.1 그래프의 표현

SB공간에서는 SB Network(이하 SN)를 구축하는 방식으로 모든 작업이 수행된다. SN은 그래프로의 자동변

환이 가능하다. 그림 10은 그래프로 표현된 SN(자동차구매_SB)의 예이다. 그림에서 9개 노드는 그림 9의 9개 SB를 상징한다.



[Fig. 10] a graphical representation of the car buying _SB

4.2 그래프 분석

그림 10의 그래프를 통해서 자동차 구매 작업이 갖는 여러 가지 특성을 알 수 있다.

(1) 행위 주체와 상호작용

작업에 참여한 행위 주체의 수는 9이다. 이것은 노드의 수를 조사하면 알 수 있다. 작업 과정에서 발생한 상호작용의 수는 15이다. 이것은 링크의 수를 조사하면 알 수 있다.

(2) 노드간의 평균 거리

15개 상호작용에 있어서, 노드간의 평균 이동 거리는 2.2이다. 여기서 평균 이동거리는 상호 작용 한 개당 평균 클릭 수를 의미한다.

(3) 허브

방향성 그래프에서는 출력 차수가 큰 출력허브와 입력 차수가 큰 입력허브로 나누어서 생각할 수 있다. 그림 4.1에서 출력허브는 영구_SB(차수8)이고 입력허브는 벤츠27_SB(차수 5)이다.

(4) 복잡도

SB공간에서 이루어지는 작업의 복잡도는 그래프 상에서 링크의 수를 노드의 수로 나눈 값으로 간단하게 정의된다. 즉, $X = L/N$ 이다.

X : 복잡도 N : 노드의 수 L : 링크의 수

$$L = \sum_{i=1}^N l_i \quad l_i : \text{각 링크의 길이}$$

본 작업의 경우 복잡도는 표 1과 같다. 표 1에서 상황 SB의 복잡도 평균은 0.88이다. 복잡도가 1.0을 넘으면 상황 파악이 어려워지며, 1.5를 넘으면 거의 식별이 불가능하다.

[Table 1] Calculation complexity

| 종류 | 링크 수/ 노드 수 | 복잡도 |
|------|------------|------|
| 상황1 | 3/4 | 0.75 |
| 상황2 | 3/3 | 1.00 |
| 상황3 | 3/3 | 1.00 |
| 상황4 | 3/3 | 1.00 |
| 상황5 | 3/4 | 0.75 |
| Task | 15/9 | 1.67 |

5. 평가

웹 공간과 비교하여, SB공간의 이점을 다음과 같이 요약할 수 있다.

(1) 투명한 처리

SB공간에서는 다수의 행위 주체들이 같은 화면을 보면서 작업하기 때문에 서로가 무슨 일을 하는지를 쉽게 알 수 있다. 이로 인해 두 가지 이점이 생기는데, 하나는 작업 효율이 개선된다는 점이고 다른 하나는 크로스 체크가 가능하여 서로의 행동을 감시할 수 있다는 점이다.

(2) 용이한 수정

SB공간에서는 상황 SB 단위로 작업이 진행되며, 모든 작업 결과는 [h]단자를 통해서 체계적으로 저장되고 관리된다. 에러가 발견될 경우에는 [h]단자를 통해서 해당되는 상황 SB를 쉽게 불러올 수 있으며, 그 상황에서 작업을 다시 시작할 수 있다. 즉, 작업의 수정이 매우 용이하다.

(3) 자동 분석

SB공간에서는 모든 작업이 SB Network를 구축하는 과정으로 정의된다. SB network는 그래프의 자동변환이 가능하다. SB Network가 그래프로 표현될 경우, 이를 통해 대상 작업에 대한 정량적인 분석이 가능해진다.

6. 결론

SB공간은 웹 공간과는 달리 표준화된 통합 환경을 지원한다. 즉, 다수의 행위 주체들이 하나의 통합된 작업 환경에서 서로의 프로세스를 공유하며 일을 한다.

이런 특징 때문에 SB공간은 웹에 대한 메타 공간적 기능을 수행할 수 있다. 즉, 현재의 웹에서 다루기 어려운 프로세스 공유 부분을 따로 떼어 SB공간에서 처리할 수 있다.

본 논문에서는 간단한 시뮬레이터를 만들어서 SB공간의 작동 과정을 구체적인 사례를 통해서 확인할 수 있게 하였다.

시뮬레이션을 통해서 얻어진 서비스 계층 정보를 이용하여 실제 현장에서 돌아가는 프로토타입을 구현해보는 것이 향후 과제이다.

REFERENCES

- [1] Tim Berners-Lee, WEAVING THE WEB, Publishers, New York, NY, USA, 1999.
- [2] Tim O'Reilly, What Is Web 2.0 Design Pattern and Business Models for the Next Generation of Software, 2005.
- [3] <http://secondlife.com/>

저자소개

공 현 택(Heon-Tag Kong)

[중신회원]



- 1984년 : Northeast Missouri State Univ. 전산학과(학사)
- 1987년 : Utah State Univ. 전산학과(석사)
- 1998년 : 단국대학교 전산통계학과(박사)

· 1990년~현재 : 공주대학교 컴퓨터공학부 교수

<관심분야> : 병렬 알고리즘, 데이터베이스

고 성 범(Sung-Bum Ko)

[정회원]



- 1983년 : 서울대학교 대학원 전기 공학과 졸업(석사)
- 2003년 : 건국대학교 대학원 컴퓨터공학과 졸업(박사)
- 1983년~현재 : 공주대학교 컴퓨터공학부 교수

<관심분야> : 인공지능