

게임 기획 산출물의 일관성 유지를 위한 변경 추적 기법

박재현[†] · 이두원^{**} · 류성열^{***}

요약

한국게임산업개발원에서 연구 발표한 게임제작 프로세스와 게임 기획서 표준양식은 게임 산업을 위한 중요한 초석이 되었다. 그러나 게임 기획은 기획자의 새로운 아이디어와 변경으로 정확한 산출물을 작성할 수 없어 소프트웨어의 품질을 보장할 수 없다.

본 연구에서는 기획단계에서 생성되는 산출물들을 일관성 있게 유지 관리하고 변경된 내용을 추적할 수 있는 체계를 정립하였다. 첫째, 각 산출물간의 연관관계를 표현하는 ACT그래프, 산출물의 변경을 추적·관리를 위한 ACT 테이블 및 산출물 변경 이력 테이블을 정의하였다. 또 이를 활용한 게임 기획 ACT 프로세스를 제시하였다. 제시한 추적 기법의 유효성을 검증하기 위하여 사례연구를 통해 게임 기획 산출물의 변경 발생 시 게임 기획자가 관련 산출물과 항목을 신속히 파악하고 변경 가능한 것을 확인함으로써 게임 기획 산출물의 일관성 유지 노력과 시간을 절약할 수 있음을 보였다.

키워드 : 게임, 산출물, 일관성, 변경 추적 기법, 게임 기획, ACT

A Change Tracking Technique for Maintaining Consistency of Game Design Artifacts

Park Jae-Hyun[†] · Yi Du-Won^{**} · Rhew Sung-Yul^{***}

ABSTRACT

The Game development process that the Korea Game Development Infinity researches and publishes has laid important groundwork for the game industry. However, the game design cannot guarantee a software quality because an artifact is made incorrect by a game planner's new idea and modification.

In this paper theorized the system to maintain artifacts consistently that is generated in planning phase and make changed contents traceable. First, it defined an ACT graph to represent relation of between artifacts, a ACT table that traces and manages it changes and a table of artifacts change history. Also, it suggested the ACT process applying as we mentioned above. Then we worked a case study to verify the change tracking technique which is presented. In a case study, we find game designer is capable to grip and modify a change of artifacts when it occurs a change of game design artifacts. As a result of that, we can save the time and effort to maintain the consistency among game design artifacts.

Key Words : Game, Artifacts, Consistency, Change Tracking Technique, Game Design

1. 서론

게임소프트웨어산업은 IT 산업 중에 다른 산업과의 연계를 통해 큰 부가가치를 창출하며 침체된 현재 IT 산업을 활성화시킬 수 있는 분야로 빠르게 성장하고 있다. 또한 게임 소비자 인구가 폭발적으로 증가하고 소비층이 폭넓어짐에 따라 엔터테인먼트의 핵심적인 분야로 각광 받고 있으며, 크고 작은 다수의 게임이 제작되어지고 있다[1, 2]. 이러한

게임산업의 특징은 산출물 지향적, 산출물 기반, 내용 기반의 산업이라 할 수 있으며, 게임제작은 타 응용프로그램 개발과는 달리 게임만의 독특하고 다양한 산출물들이 만들어진다[3-6].

그러나 게임 기획 시 제작되는 산출물들은 그 특성상 잦은 변경이 발생하게 되고, 각 산출물들은 서로 깊은 연관관계를 가지고 있기 때문에 특정 산출물의 변경이 다른 산출물에게 미치는 파급효과는 클 수밖에 없다. 이로 인해 산출물간의 일관성 유지가 어려운 뿐만 아니라 유지보수 및 재사용의 어려움으로 이어지게 된다. 또한, 게임 소프트웨어 개발 과정의 여러 기획 산출물 간의 일관성 유지는 최종 게임 소프트웨어의 품질을 결정하는 중요한 요소가 되며, 운

※ 본 연구는 숭실대학교 교내연구비 지원으로 이루어졌음.

† 정회원 : 숭실대학교 컴퓨터학과 석사과정

** 준회원 : 숭실대학교 컴퓨터학과 석사과정

*** 종신회원 : 숭실대학교 정보과학대학 컴퓨터학부 교수

논문접수 : 2005년 7월 4일, 심사완료 : 2006년 1월 23일

영단계의 유지보수 효율성에도 큰 영향을 미친다. 특히 게임 기획 후반부 활동 진행 시 발견된 하나의 오류는 그 이전 단계의 다수의 산출물 오류를 가져오므로, 발견된 오류의 원인이 되는 이전 단계의 오류를 효과적으로 추적하고 수정할 수 있는 체계가 필요하다[7]. 따라서 게임 기획 산출물의 일관성 유지는 선택사항이 아닌 필수사항이다.

게임개발프로세스는 게임 개발의 전체 프로세스이며 실제적인 기획 즉 디자인의 대부분이 이루어지는 단계는 총 6단계 중에서 요구사항의 분석과 설계에 해당하는 요구사항수집과 게임요소 분석 및 설계 단계이다[4, 6]. 물론 게임개발프로세스의 게임요소 구현과 테스트 단계에서 기획서의 수정이 일어날 수도 있고, 반복에 의하여 수행되는 프로세스의 특성상 특정단계에서만 디자인이 일어난다고 규정지을 수는 없지만 본 연구에서는 디자인 즉 기획의 핵심 아이디어 대부분이 도출되는 요구사항 수집 단계와 게임요소 분석 및 설계단계에 포커스를 맞추어 본다.

게임의 기획은 일반적인 소프트웨어의 요구사항분석 단계에 해당한다. 물론 다루는 범위는 일반의 소프트웨어 보다는 훨씬 크지만, 아이디어를 구체화한다는 관점에서 동일하게 다루는 것이 가능하다. 따라서 게임 기획 산출물의 일관성 및 변경관리 기법을 제안하기 위하여, IEEE의 소프트웨어 요구사항명세서의 작성 권고안의 평가기준을 따라 적용하였다[8]. 본 연구가 적용한 평가기준은 위에서 언급한 8가지 중에서 일관적이고(Consistency), 수정가능하고(Modifiable), 추적가능(Traceable)해야 한다는 것이다[8].

2. 관련 연구

2.1 게임제작프로세스[3, 4]

게임에 있어서 산출물은 일반적인 방법론의 산출물과는

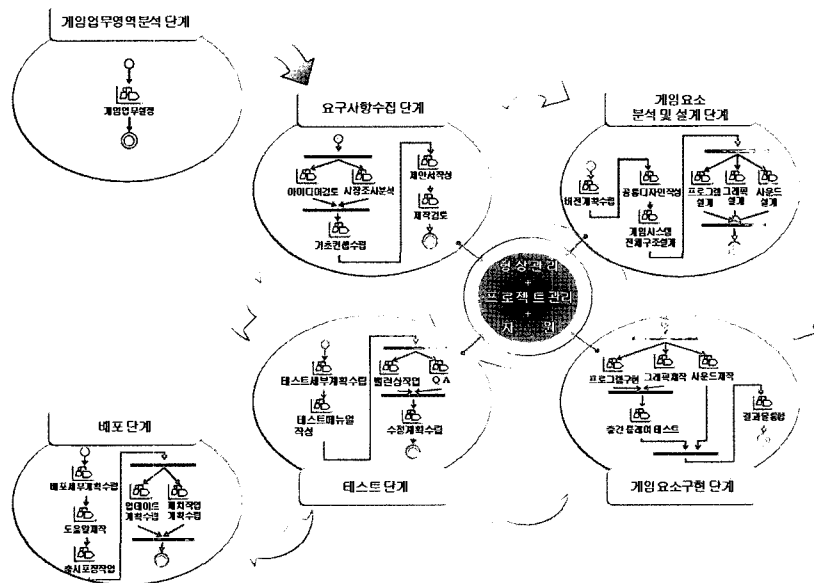
그 성격이 다르다. 개발 프로세스도 다를 뿐 만 아니라 업체마다 각각의 개발 프로세스를 가지고 있다. 또한 각 프로세스에 따른 그에 따른 관리방법도 달라져야 한다.

실제 일반 시스템 개발을 위한 방법론은 이미 학계나 시장에 상당수가 존재하지만 이러한 방법론들이 목표로 하고 있는 시스템은 게임과는 성격이 상이하기 때문에 게임제작에 직접적으로 적용한다는 것은 상당한 무리가 따른다[3,4].

지난 2004년, 한국게임산업개발원에서 발표한 게임제작프로세스 개선안은 초기 게임제작프로세스를 바탕으로 이를 정제하여 (그림 1)과 같이 총 6개의 단계와 27개의 활동으로 구성되어 있으며, 약 50여 개 이상의 산출물을 필요로 하고 있다[4]. 여기에서는 일반 응용프로그램 개발 시 제작되는 산출물과는 매우 다른 형태의 게임에 특화된 개발 프로세스와 산출물들을 제시하고 있다.

2.2 게임기획서표준양식[5]과 작성

‘게임기획서표준양식’이란 게임 디자이너가 게임을 개발하는 과정에서 생각해낸 디자인 내용을 체계적으로 정리하고, 이를 각 파트별 작업자들에게 올바르게 전달할 수 있도록 하기 위해서 만든 게임 디자인 문서 양식을 말한다[5]. 이 양식은 게임 개발과정 중 발생하는 게임 디자인 업무의 체계를 정립하고, 게임 디자인 과정에서 나타날 수 있는 오류 또는 실수를 미연에 방지하며, 작업자들이 문서를 통해서 디자인 내용을 정확히 이해하고 그에 맞게 작업하여 결과물을 생성할 수 있도록 하기 위해 사용된다. 여기서 지금까지 업체들 사이에 널리 사용되던 ‘기획’이라는 개념을 ‘디자인’이라고 칭한 것은, 기존의 ‘게임 기획’이라는 단어가 가진 매우 광범위하고 포괄적인 의미보다는 ‘게임 디자인’이라는 특화된 단어와 의미를 채택함으로써 본 양식의 전문성과 방향성을 좀 더 확고하게 다지기 위함이다.



(그림 1) 게임제작프로세스 개선안[4]

게임기획서표준양식의 진행순서는 순차적인 일련의 단계들을 반복적으로 수행함으로써 진행하는 것이 특징이다. 이것은 많은 개발사들이 게임을 완성해나가는 과정에 있어서 일련의 순환적인 프로세스를 따르고 있기 때문에, 이러한 보편적인 공통요소들을 적용하여 표준양식의 진행단계를 구축한 것이며 한편으로는 RUP(Rational Unified Process)의 특징적인 진행방식을 연구하여 도입하였기 때문이기도 하다. 각 단계들은 하나의 버전이 완성되는 과정에서 반복적으로 작업되며, 모든 단계들이 모여서 이루어진 전체적인 진행흐름 또한 게임이 완성되는 과정에서 반복적으로 진행된다.

<표 1>은 게임 기획의 각 단계별로 작성되는 문서를 나타낸다.

<표 1> 게임 기획 단계별 작성문서

단계	작성문서
의사결정단계	<ul style="list-style-type: none"> ○ 아이디어제안서 ○ 브레인스토밍회의록 ○ 시장조사보고서 ○ 프로젝트제안서 ○ 기초컨셉설정문서 ○ 제품소개문서
개발준비단계	<ul style="list-style-type: none"> ○ 버전계획수립문서
게임컨셉설정 단계	<ul style="list-style-type: none"> ○ 주요컨셉설정문서 ○ 스토리설정문서 ○ 세계관설정문서 ○ 게임진행설정문서 ○ 개체컨셉설정문서 ○ 배경컨셉설정문서
게임요소 디자인단계	<ul style="list-style-type: none"> ○ 게임화면디자인문서 ○ 시스템디자인문서 ○ 시스템공식디자인문서 ○ 조작인터페이스디자인문서 ○ GUI디자인문서 ○ 개체디자인문서 ○ 지역디자인문서 ○ 이벤트디자인문서 ○ 사운드디자인문서
세부리스트 작성단계	<ul style="list-style-type: none"> ○ 게임화면리스트문서 ○ 시스템리스트문서 ○ 시스템공식리스트문서 ○ 조작인터페이스리스트문서 ○ 개체리스트문서 ○ 효과리스트문서 ○ 지역리스트문서 ○ 이벤트리스트문서 ○ 사운드리스트문서 ○ GUI리스트문서
개발평가단계	<ul style="list-style-type: none"> ○ 개발평가서

2.3 산출물간 일관성 점검 방법

기존의 시스템에 새로운 요구사항이나 신기술을 반영하여 변경하고자 할 때, 기존의 시스템에 대한 정확한 이해가 필요하다. 기존의 시스템을 정확히 이해하고, 설계문서와 같은 필요한 정보를 추출하기 위해서 역공학을 이용한다. 역공학을 이용한 일관성 점검 방법은 원시코드로부터 설계모델을 추출하여 기존의 설계단계에서 작성한 설계모델과 비교하여 일관성을 검증하는 방법이다[9]. 이러한 방법은 역공학의 목적이 원래 기존 시스템을 이해하고, 이를 통해 요구사항을 만들어 내는 것이기 때문에, 소프트웨어 요구사항명세서의 변경관리 및 추적하는 기법과는 직접적인 관련이 없다. 따라서 설계와 구현사이의 일관성을 검증할 수는 있지만 게임 기획 산출물간의 일관성을 유지하기 위한 방법으로는 적합하지 않다.

계약언어를 이용하는 방법은 객체모델을 생성할 수 있는

객체모델 작성기와 작성된 객체모델을 제약언어를 기반으로 검증할 수 있는 객체모델 검증기를 포함한 객체모델 검증시스템을 이용하는 방법이다[10]. 그러나 객체모델은 시스템의 설계구성요소간의 검증에 사용하기 위해 만들어졌기 때문에, 내용중심의 게임 기획 산출물에 적용하는 것은 어렵다.

지식베이스를 이용하는 방법은 객체모델, 기능모델, 동적모델을 정형명세화하여 정보를 추출하고, 지식베이스에 담겨있는 추론규칙을 이용하여 분석하는 오류검사방법이다[11]. 하지만, 모델들 사이의 일관성 점검을 수행하기에는 제시된 규칙의 완전성 면에서 부족할 뿐만 아니라, 규칙을 만들어 가는 과정 또한 상당한 시간과 비용이 들어가는 일이기 때문에 적용하기에 어려움이 있다. 또한 모델이 아닌 문서에 대한 일관성 유지 방안에 대해서는 언급되어 있지 않아서 추가적인 연구가 필요한 점도 가지고 있다.

지금까지 언급한 기존의 산출물 및 요구사항 구성요소의 일관성 유지에 관한 연구들은 주로 모델위주의 방법이 주류를 이루고 있으며, 개발 문서에 대한 일관성 유지 방법은 기존의 소프트웨어형상관리에서 일부분만 다루어지는 경향을 보이고 있다.

소프트웨어 형상관리는 형상 항목을 식별하여 그 기능적 물리적 특성을 문서화하고, 그러한 특성에 대한 변경을 제어하고, 변경 처리 상태를 기록 및 보고하고, 명시된 요구사항에 부합하는지 확인하는 일련의 사항에 대해 기술적 행정적인 지침과 사후 관리를 적용하는 시스템을 작성하는 과정 전체를 망라하는 원칙이다[12].

그렇기 때문에 형상관리에서는 산출물의 수직적인 관리 측면만을 강조한, 즉 해당 산출물의 시스템 구축과정 전체의 변화과정 자체에 대한 관리만이 중점적으로 다루어지고 있다. 본 연구에서 다루고자하는 기법은 산출물들 사이의 연관관계에 따른 변경 다시 말해 수평적인 관계를 추적·관리하는 기법으로서, 이에 대한 연구는 미흡하며, 게임이라는 특수한 분야의 기획 산출물을 다루는 연구는 현재 제시되지 않고 있다.

3. 게임 기획 산출물 변경 추적 기법

3.1 게임 기획 산출물의 연관관계 분석

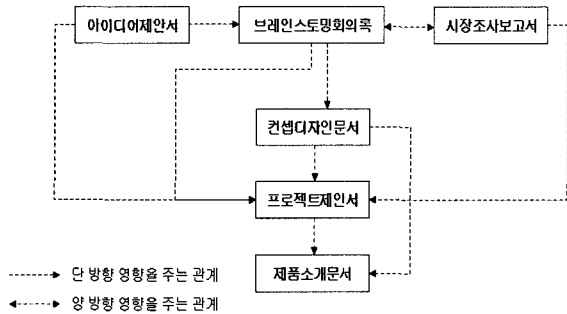
게임 기획 산출물간의 연관관계를 분석은, 게임 기획 산출물의 일관성 유지를 위한 기준을 만드는 것이다. 게임기획서표준양식의 의사결정 단계 산출물 아이디어제안서, 브레인스토밍회의록, 시장조사보고서, 프로젝트제안서, 기초컨셉설정문서, 제품소개문서를 가지고 이들의 연관관계를 분석한다.

산출물들이 작성을 위해 정해진 순서가 따로 존재하지 않기 때문에 산출물의 변경·추적에 직접적인 영향을 주고받는 것과는 상관이 없다. 그러므로 문서작성의 흐름은 본 연구에서는 고려하지 않으며, 상호간에 영향을 주지 않는 산출물도 고려 대상에서 제외한다.

브레인스토밍과 시장조사보고서의 관계는 회의의 영향에

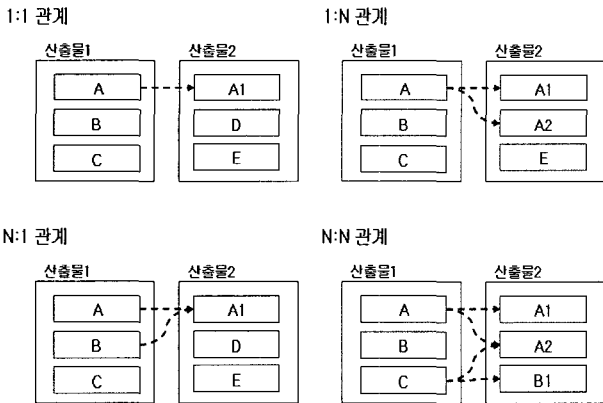
의해 시장조사가 행해질수도 있고, 시장조사 결과를 바탕으로 회의의 결론이 변경될 수 있기 때문에 산출물 상호간에 관계는 순환의 우려가 존재한다. 그러나 상호간에 영향을 주더라도 반드시 먼저 영향을 준 산출물과 영향을 받은 산출물이 존재하고, 이 판단은 작성자에 의해 고려되기 때문에 순환에 빠지진 않는다.

위의 분석결과를 바탕으로 본 연구의 대상이 되는 산출물간의 연관관계를 정의하면 (그림 2)와 같다.



(그림 2) 의사결정단계 산출물간의 연관관계 정리

게임 기획 산출물들의 내부에는 다양한 항목들이 존재하며, 그 항목들은 단어, 문장, 이미지 등의 형태로 다양하게 나타난다. 게임 기획 산출물에서 각 항목들이 가질 수 있는 관계는 (그림 5)와 같이 1:1, 1:N, N:1, N:N의 모든 관계가 나타나며, 각 항목들(단어, 문장, 이미지 등)은 각기 다른 산출물의 다른 항목에 영향을 미치게 된다.



(그림 3) 산출물 항목들의 연관관계

3.2 ACT 그래프

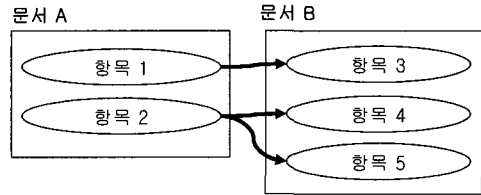
본 연구에서는 게임 기획 산출물의 변경을 추적하기 위해서 ACT(Artifacts Consistency Tracking: 산출물 일관성 추적) 그래프를 제안한다.

ACT 그래프란 산출물의 항목과 항목간의 연관관계를 표현한 그래프로서 산출물의 변경 발생 시 일관성 유지를 변경의 추적을 가능하게 해 준다. ACT 그래프의 구성요소와 활용방법은 <표 2>와 같다.

<표 2> ACT 그래프의 표현방법 및 사용례

구성요소명		표현방법
문서		[]
항목		()
방향선	단방향	→
	양방향	↔

<ACT 그래프의 사용례>



- 문서: 게임 기획 산출물을 표현한다.
- 항목: 게임 기획 산출물내의 항목을 표현한다. 항목은 단어나 문장, 이미지 등과 같은 산출물 추적의 대상이 되는 요소를 의미한다.
- 방향선: 산출물 항목과 항목간의 연관관계를 표현한다. 연관관계에는 단방향 실선과 양방향 실선을 사용하며, 단방향 실선은 한쪽 방향만으로 영향을 미치는 것을 표현하고, 양방향 실선은 산출물 항목 상호간에 영향을 미치는 것을 표현한다.

3.3 ACT 테이블

3.3.1 ACT 테이블 구조

ACT 테이블은 산출물간의 연관관계를 표현하기 위한 테이블이다. 항목에 변경이 가해질 때 각 항목을 자신이 변경되도록 사건을 촉발한 선행산출물의 항목이 존재할 수 있으며, 마찬가지로 자신이 다른 산출물의 변경을 유발할 수도 있다. 전자를 선행산출물이라 하고, 후자를 후행산출물이라 한다. ACT 그래프로 보자면 어떤 항목에 화살표가 닿아있는 상태라면 해당 화살표를 시작한 항목이 선행산출물의 항목이며, 반대로 화살표를 시작한 상태라면 닿아있는 항목이 후행산출물의 항목이 된다. 앞서 연관관계의 종류에서 알 수 있듯이 선행 및 후행산출물 모두 한 개 이상 가능하다.

따라서 ACT 그래프로 표현된 산출물간의 연관관계는 <표 3>의 테이블로 변경하여 저장하는 것도 가능하며, 물론 그 역도 가능해진다. ACT 그래프를 ACT 테이블로 저장함으로써 추후에 본 연구의 아이디어를 산출물 항목간의 일관성 추적 자동화 도구로 구현하는 것이 가능하다. 이 도구에

<표 3> ACT 테이블

산출물명	항목	선행산출물		후행산출물	
		산출물명	항목	산출물명	항목

서는 ACT 그래프를 먼저 그리고 작성한 후 ACT 테이블을 얻을 수 있으며, 반대로 ACT 테이블을 통해서 ACT 그래프를 얻는 것도 가능해 질 것이다.

3.3.2 ACT 테이블의 활용

게임 기획 산출물의 변경여부는 게임 기획 산출물의 작성자가 판단해야 하며 변경이 필요하다고 판단되는 경우, 해당 항목을 변경 후 변경된 항목이 영향을 미치는 또 다른 항목을 찾아 다시 변경여부를 판단하게 된다.

예를 들어, (그림 4)과 같이 산출물1의 A항목이 변경되면 산출물2의 D항목을 변경여부를 판단해야 하며, 산출물2의 D항목이 변경되면, 산출물3의 G와 H의 변경여부를 판단해야 한다.

산출물4의 K가 변경되면 산출물2의 D항목의 변경여부를 판단하고, 산출물 D가 변경되면 산출물1의 A항목의 변경여부를 판단함으로써 산출물의 연관관계 추적이 가능해 진다.

산출물명	항목	선행산출물		후행산출물	
		산출물명	항목	산출물명	항목
산출물1	A	-	-	산출물2	D
산출물1	B	-	-	-	-
산출물1	C	-	-	-	-
산출물2	D	산출물1	A	산출물3	G
산출물2	E	산출물1	A	산출물3	H
산출물2	F	-	-	-	-
산출물3	G	-	-	산출물4	I
산출물3	H	산출물2	D	산출물4	K

(그림 4) 산출물 일관성 추적 테이블의 활용 예

3.4 산출물 변경 이력 테이블

3.4.1 산출물 변경 이력 테이블의 구조

산출물 변경 이력 테이블은 산출물의 변경 발생 시 이에 대한 이력을 기록하는 것이다. 산출물의 변경 이력을 관리함으로써 얻을 수 있는 이점은 산출물의 변경이 발생했을 경우 하나의 변경에 대해 그 변경이 영향을 미친 산출물들을 추적할 수 있고 그로 인한 변경세부내역을 확인할 수 있다.

산출물의 변경 발생 시 그 이력을 <표 4>와 같이 관리한다. 변경 이력 기록 시 산출물 일관성 추적 테이블과는 달리 선행·후행 산출물은 따로 기록하지 않고, 변경에 영향을 준 산출물만을 기록한다. 변경상태는 변경시작, 변경진행, 변경완료의 3가지로 구분하며, 변경시작은 최초 변경이 일어난 산출물 항목(하나만 존재)을 나타내며, 변경진행은 변경이 일어난 산출물 항목(다수 존재 가능), 변경완료는 변경이 완료된 산출물 항목(다수 존재 가능)을 나타낸다.

시스템 개발을 진행하다 보면 여타 다양한 원인으로 인해 특정 시점으로 돌아갈 경우도 발생한다. 빈번하게 변화하는

<표 4> 산출물 변경 이력 테이블

산출물명	항목	변경에 영향을 준 산출물		변경 내용	변경자	변경 일자	변경 상태
		산출물명	항목				

게임 디자인 또한 마찬가지로이며 이런 경우에 산출물 변경 이력 테이블은 특정 시점의 문서로 되돌리는 것을 가능하게 해주는, 변경의 버전관리 기능을 구현하는 기초 아이디어에 사용할 수 있다.

3.4.2 산출물 변경 이력 테이블의 활용

(그림 5)의 산출물 변경 이력 테이블의 사용 예를 보면 산출물1의 A항목이 변경이 발생되고, 그에 따라 산출물2의 D가 변경되고 산출물3의 G가 변경된 후 더 이상의 변경이 일어나지 않았음을 알 수 있다.

산출물명	항목	변경에 영향을 준 산출물		변경내용	변경자	변경 일자	변경상태
		산출물명	항목				
산출물1	A	-	-	생략	생략	생략	변경시작
산출물2	D	산출물1	A	변경진행
산출물3	G	산출물2	D	변경완료
...							

(그림 5) 산출물 변경 이력 테이블의 활용 예

3.4.3 산출물 변경 이력 기록 절차

산출물의 변경 발생하였을 경우, 그 이력을 산출물 변경 이력 테이블에 기록 시 다음과 같은 절차를 수행한다.

1. 변경이 시작된 산출물의 '변경상태'를 '변경시작'으로 설정하고, '변경에 영향을 준 산출물'은 입력하지 않는다.
2. 다른 산출물의 영향을 받은 변경일 경우, '변경에 영향을 준 산출물'을 기록하고 '변경상태'를 '변경진행'으로 설정한다.
3. 다른 산출물에 변경 영향을 주지 않는 산출물일 경우, '변경상태'를 '변경완료'로 설정한다.
4. 변경된 산출물의 정보(변경내용, 변경자 등)를 입력한다.
5. 변경이 완료 될 때까지 절차1-4를 반복한다.

4. ACT 프로세스

ACT 프로세스는 크게 2가지 절차로 나누어진다. 변경관리를 위한 베이스를 마련하는 준비 절차와 실제 변경이 발생했을 경우 이를 관리하는 절차인 변경 절차이다.

4.1 준비 절차

산출물의 일관성 추적을 위해서는 이를 위한 기반이 마련되어 있어야 한다. ACT 프로세스의 준비 절차는 다음과 같다.

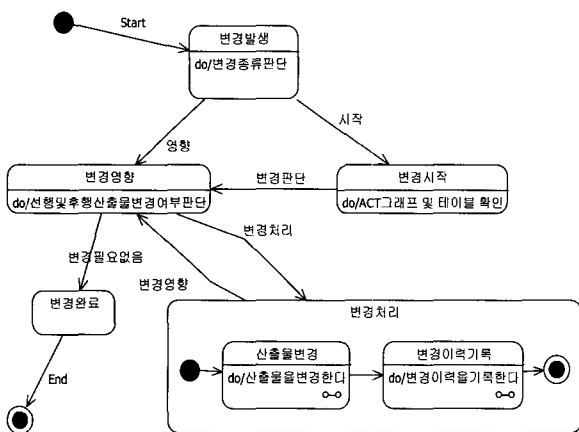
1. 게임 기획에서 사용할 용어들을 정의한다.
2. 게임 기획 산출물을 식별한다.
3. 각 산출물에서 사용될 항목을 식별한다.
4. 각 산출물 항목간의 연관관계를 식별한다.
5. 각 산출물 항목간의 연관관계를 본 연구의 3장에서 제시한 ACT 그래프 또는 ACT 테이블을 이용하여 표현한다.

4.2 변경 절차

산출물의 변경이 발생하였을 경우, 다음의 절차에 따라 변경을 수행한다.

1. 산출물의 변경사항이 발생하면 변경의 시작인지, 다른 산출물 변경의 영향인지를 판단한다. 변경의 시작점이면 절차2를 수행하고, 다른 산출물 변경의 영향인 경우 절차3을 수행한다.
2. 변경의 시작인 경우, ACT 그래프 또는 ACT 테이블의 선행·후행관계에 있는 모든 항목에 대해 변경여부를 판단한다.
3. 다른 변경의 영향인 경우, 선행산출물에서 영향을 받았다면 후행산출물의 변경여부를 판단하고, 후행산출물에서 영향을 받았다면 선행산출물의 변경여부를 판단한다. 선행·후행산출물의 변경이 필요 없다고 판단될 경우 절차 7을 수행한다.
4. 산출물을 변경한다.
5. 본 연구의 3장에서 제시한 산출물 변경 이력 기록 절차에 따라 산출물의 변경 이력을 기록한다.
6. 모든 연관된 산출물의 항목이 변경이 완료될 때까지 절차1-5를 반복한다.
7. 산출물의 변경을 완료한다.

UML의 상태 다이어그램[12]은 유한 상태 기계(Finite State Machine)를 표현하기 위한 다이어그램으로써, 앞에서 제시한 산출물의 변경절차를 표현하는 데 유용한 도구이다. 상태중심의 시스템 모델링은 차후에 자동화 도구를 구현하는데 훌륭한 설계모델을 프로토타입으로 제공한다. (그림 8)은 변경절차를 상태 다이어그램으로 모델링한 것이다.



(그림 6) 변경절차 상태 다이어그램

5. 사례 연구

5.1 연구 범위

게임기획서표준양식 중 가장 대표적이면서도 서로 관련성

이 깊은 캐릭터관련설정문서, 등장캐릭터설정문서, 개체별세부디자인문서를 사례연구의 범위로 선정하고, 이를 본 연구에서 제시한 게임 기획 산출물 일관성 추적 기법을 이용하여 일관성을 유지하고 변경을 추적 관리할 수 있음을 보여 사용성을 검증하고자 한다.

5.2 준비 절차

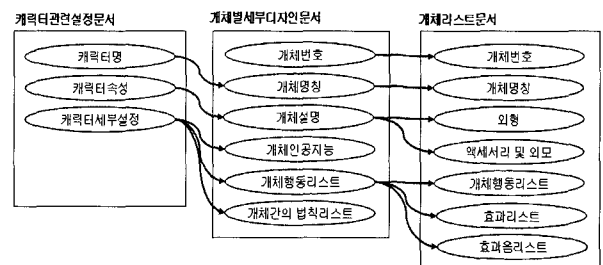
산출물 일관성 유지를 위한 변경 추적 기법의 기반을 마련하기 위해 4장에서 제시한 ACT 프로세스의 준비 절차를 수행한다.

준비 절차에서 요구하는 바와 같이 3가지 문서에서 사용되는 용어를 정의하고, 산출물과 항목을 식별하였다. <표 5>는 산출물과 항목의 식별 결과를 나타낸다.

<표 5> 산출물과 항목 식별

산출물명	항목명	정의
캐릭터 관련 설정 문서	캐릭터번호	캐릭터의 고유번호를 표기한다.
	캐릭터명	캐릭터의 이름을 표기한다.
	캐릭터속성	게임에서 해당 캐릭터의 역할에 대해 간략히 기술한다.
	캐릭터세부설정	캐릭터의 탄생 배경이나 성격, 나이 등과 같은 세부적인 설정을 기술한다.
등장 캐릭터 컨셉 문서	캐릭터번호	캐릭터의 고유번호를 표기한다.
	캐릭터명	캐릭터의 이름을 표기한다.
	외모	캐릭터의 외모에 대한 컨셉을 기술한다.
	의상	캐릭터의 의상에 대한 컨셉을 기술한다.
	기타외형	캐릭터의 기타외형에 대해 기술한다.
개체별 세부 디자인 문서	개체번호	개체의 고유번호를 표기한다.
	개체명칭	개체의 이름을 표기한다.
	개체설명	개체의 세부적인 내용 전반을 기술한다.
	개체인공지능	개체의 지각능력, 행동선택능력, 상태체크능력 등 인공지능에 대하여 기술한다.
	개체행동리스트	개체의 행동과 제약사항을 나열한다.
	개체간의법칙 리스트	각 개체들 간의 상호작용되는 법칙에 대하여 나열한다.

<표 5>에서 식별된 산출물과 항목을 기반으로 산출물간의 연관관계를 식별하고 (그림 11)과 같이 ACT 그래프로 나타내었다.



(그림 7) 산출물간의 연관관계 설정

<표 6>은 (그림 11)에서 도출된 ACT 그래프를 ACT 테이블로 변경시킨 것이다.

〈표 6〉 ACT 테이블의 적용

산출물명	항목	선행산출물		후행산출물	
		산출물명	항목	산출물명	항목
캐릭터관련설정문서	캐릭터번호	-	-	등장캐릭터설정문서	캐릭터번호
캐릭터관련설정문서	캐릭터명	-	-	등장캐릭터설정문서	캐릭터명
캐릭터관련설정문서	캐릭터세부설정	-	-	등장캐릭터설정문서	외모
캐릭터관련설정문서	캐릭터세부설정	-	-	등장캐릭터설정문서	의상
캐릭터관련설정문서	캐릭터세부설정	-	-	등장캐릭터설정문서	기타외형
등장캐릭터설정문서	캐릭터번호	캐릭터관련설정문서	캐릭터번호	-	-
등장캐릭터설정문서	캐릭터명	캐릭터관련설정문서	캐릭터명	개체별세부디자인문서	개체명칭
등장캐릭터설정문서	외모	캐릭터관련설정문서	캐릭터세부설정	개체별세부디자인문서	개체설명
등장캐릭터설정문서	의상	캐릭터관련설정문서	캐릭터세부설정	개체별세부디자인문서	개체설명
등장캐릭터설정문서	기타외형	캐릭터관련설정문서	캐릭터세부설정	개체별세부디자인문서	개체설명
개체별세부디자인문서	개체명칭	등장캐릭터설정문서	캐릭터명	-	-
개체별세부디자인문서	개체설명	등장캐릭터설정문서	외모	-	-
개체별세부디자인문서	개체설명	등장캐릭터설정문서	의상	-	-
개체별세부디자인문서	개체설명	등장캐릭터설정문서	기타외형	-	-

〈표 7〉 산출물 변경 이력 테이블의 적용

산출물명	항목	변경에 영향을 준 산출물		변경 내용	변경자	변경 일자	변경 상태
		산출물명	항목				
등장캐릭터 설정문서	외모	-	-	생략...	박재현	05.06.23	변경시작
캐릭터관련 설정문서	캐릭터세부설정	등장캐릭터 설정문서	외모	생략...	박재현	05.06.23	변경진행
개체별세부 디자인문서	개체설명	등장캐릭터 설정문서	외모	생략...	박재현	05.06.23	변경완료

5.3 변경 절차

산출물 일관성 추적 준비 절차의 작업이 완료된 후 산출물의 변경이 발생하였고 변경 절차를 적용하여 산출물의 변경을 관리하였다.

등장캐릭터설정문서에서 캐릭터번호와 캐릭터명은 변경이 발생하지 않았지만 외모 항목의 변경이 발생하였고, ACT 그래프와 ACT 테이블을 이용한 변경 추적 기법을 통해 변경에 영향을 미치는 산출물을 찾아보았다. 그 결과 등장캐릭터설정문서의 이전 문서인 캐릭터관련설정문서의 캐릭터세부설정 항목과 이후 문서인 개체별세부디자인문서의 개체설명 항목에 변경의 영향이 미치게 되고 이를 수정해야 하는 것을 발견하였다. 캐릭터관련설정문서와 개체별세부디자인문서에 변경사항을 반영한 후, ACT 그래프와 ACT 테이블을 통해 변경이 영향을 미치는 다른 산출물과 항목이 더 이상 없는 것을 확인하고 변경 절차를 종료하였다.

〈표 7〉과 같이 변경에 대한 이력을 산출물 변경 이력 테이블에 변경 이력 기록절차에 따라 작성하였으며, 이 테이블로 인해 하나의 변경으로 인해 발생되었던 모든 변경들의 추적이 가능하게 되었다.

6. 결론 및 향후과제

게임제작은 타 소프트웨어 개발과는 달리 게임만의 독특하고 다양한 산출물들이 만들어지며, 특히 게임 기획 단계에서 생성되는 산출물들은 게임만의 특성을 지니고 있다[3-5].

그러나 게임 기획 시 제작되는 산출물들은 그 특성상 잦은 변경이 발생하게 되고, 각 산출물들은 서로 깊은 연관관계를 가지고 있기 때문에 특정 산출물의 변경이 다른 산출물에게 미치는 파급효과는 클 수밖에 없다. 이로 인해 산출물간의 일관성 유지가 어려울 뿐만 아니라 유지보수와 재사용성의 어려움으로 이어지게 된다. 또한, 게임 소프트웨어 개발 과정의 여러 기획 산출물 간의 일관성 유지는 최종 게임 소프트웨어의 품질을 결정하는 중요한 요소가 되며, 운영단계의 유지보수 효율성에도 큰 영향을 미치게 된다. 특히 게임 기획 후반부 활동 진행 시 발견된 하나의 오류는 그 이전 단계의 다수의 산출물 오류를 가져오므로, 발견된 오류의 원인이 되는 이전 단계의 오류를 효과적으로 추적하고 수정할 수 있는 체계가 필요하다[7].

그러나, 기존의 게임 기획 산출물의 형상관리는 산출물에 대한 수직적인 관리, 즉 산출물의 버전에 대한 관리만이 이루어지고 있으며, 산출물간의 수평적인 연관관계를 추적·관리하는 방법 및 도구에 대한 연구는 미흡하였다.

본 연구에서는 게임개발프로세스의 요구사항 분석하는 단계라 할 수 있는 요구사항 수집과 게임요소 분석 및 설계 단계에서 게임 기획 산출물의 일관성 유지를 위한 변경 추적 기법으로서 ACT 그래프, ACT 테이블과 산출물 변경 이력 테이블을 정의하고, 이를 활용한 ACT 프로세스를 제시하였으며, 제시된 추적 기법을 검증하기 위한 방법으로 사례연구를 통해 게임 기획 산출물의 변경 발생 시 일관성을 유지하고 변경사항을 추적할 수 있음을 확인하였다.

게임 기획자는 게임 기획 산출물 일관성 유지를 위한 변

경 추적 기법을 통해 게임 기획 산출물의 변경 발생 시 관련 산출물과 항목을 신속히 파악하고 손쉬운 변경·추적이 가능함으로써 게임 기획 산출물간의 일관성 유지 노력과 시간을 최소화하고 산출물의 신뢰성과 정확성을 높일 수 있다.

차후 연구에서는 본 연구에서 다루지 못한 게임개발프로세스의 게임업무영역분석을 제외한 나머지 테스트와 게임요소 구현 단계에까지 산출물의 변경 관리와 추적기법을 적용할 수 있도록 본 연구를 확장할 것이다. 또한 이를 바탕으로 산출물 일관성 유지를 위한 변경 추적 기법을 기초로 하여 이를 게임개발프로세스 전체에 걸쳐 발생하는 산출물을 자동화할 수 있는 도구의 개발의 필요성을 연구할 것이다.

참고 문헌

- [1] '디지털콘텐츠 산업백서 2003', 한국소프트웨어진흥원, 2004.
- [2] '2004 대한민국 게임백서', 한국게임산업개발원, 2004.
- [3] '게임 제작 프로세스에 관한 연구', 한국게임산업개발원, 2003.
- [4] '게임 제작 프로세스 개선안에 관한 연구', 한국게임산업개발원, 2004.
- [5] '게임 기획서 표준양식 연구 보고서', 한국게임산업개발원, 2004.
- [6] 이승훈, "게임 소프트웨어 개발 프로세스에 관한 경험적 연구", 숭실대학교 석사학위논문, 2005.
- [7] K. Narayanaswamy., Neil Goldman., "Lazy consistency : a basic for cooperative software development", Proceeding of the ACM conference on Computer-supported cooperative work table 1992, 1992.
- [8] IEEE 830-1993, Recommended Practice for Software Requirements Specifications, 1993.
- [9] Roger S. Pressman, 'Software Engineering A Practitioner's Approach', 5rd Ed, McGraw Hill, 2001.
- [10] E. Wallmuller, 'Software Quality Assurance A Practical Approach', Prentice Hall, 1994.
- [11] D. Kung, P. Hsia and J. Gao(Eds), "Testing Object-Oriented Software", IEEE Computer Society, 1998.
- [12] IEEE Std 610.12-1990 Standard Glossary of Software Engineering Terminology, 1990.

[13] OMG, UML Specification v1.5, OMG, Inc., March, 2003.



박재현

e-mail : psgrey@gmail.com
 2004년 숭실대학교 컴퓨터학과(공학사)
 2004년~현재 숭실대학교 컴퓨터학과
 석사과정
 관심분야 : 소프트웨어 공학, 프로젝트
 관리, 게임 개발, 웹 서비스



이두원

e-mail : jingagu@selab.ssu.ac.kr
 2004년 광운대학교 행정학과(행정학사)
 2004년~현재 숭실대학교 컴퓨터학과
 석사과정
 관심분야 : 소프트웨어 공학, 디자인 패턴
 애자일 소프트웨어개발,
 게임 개발



류성열

e-mail : syrhw@comp.ssu.ac.kr
 1997년 아주대학교 컴퓨터학부(공학박사)
 1997년~1998년 George Mason
 University 교환교수
 1981년~현재 숭실대학교 정보과학대학
 컴퓨터학부 교수
 1998년~2001년 숭실대학교 정보과학대학원 원장
 1998년~2004년 숭실대학교 전자계산원 원장
 관심분야 : 리엔지니어링, 소프트웨어 유지보수,
 소프트웨어 재사용, 소프트웨어 재공학/역공학,
 소프트웨어 테스트