

문제해결의 히스토리 : Concept Evolution Diagram

한승헌 · 권민경* · 이희춘**

Problem Solving History : Concept Evolution Diagram

Seung-heon Han¹, Min-kyung Kwon² and Hee-choon Lee³

Key Words: TRIZ, POSCO, Concept Evolution Diagram, 해결안 Map

Abstract

본 논문에서는 문제해결 수단계에 있어서, 기존문제의 히스토리를 정리하고, 이 히스토리 안에서 기술적인 모순을 도출하면서, 추후 문제 해결안과 향후 예상 문제를 모두 포함하는 문제 해결의 히스토리, 즉 “ Concept Evolution Diagram ” (이하 CED)을 제안한다. 이를 활용하면, 문제 초창기에 기존의 문제해결안과 그 부작용의 분석을 통해서, 문제의 기본적인 기술적인 모순을 도출할 수 있으며, 문제의 해결안과 부작용의 정리를 통하여, 프로젝트 전체 흐름 내에서 상시적으로 기술적인 모순을 도출 할 수 있게 된다. 또한, 사내 전문가 집단의 집단지성을 활용하여 문제를 해결하는 T-Hub⁽¹⁾등을 진행할 때, 모든 참여인원이 가장 쉬운 방법으로 문제를 이해할 수 있는 Guide 가 되기도 하며, 전략 TRIZ⁽²⁾ 과제를 수행할 경우, 모순 문제 정의가 힘든 경우에도, 비교적 쉽게 기본적인 모순을 도출 할 수 있는 방법이 될 수 있다.

논문의 마지막에, 실제 CED 을 활용한 사례를 통하여, 그 유용성을 확인토록 한다.

1. 서 론

TRIZ 를 활용해 아이디어를 도출, 또는 아이디어발상회의 진행(Facilitation)을 해달라는 요청이 있을 때, 한두 번의 미팅과 참고자료만 가지고 문제를 이해해야 하는 경우가 많다. 문제를 해결해 가는 과정에서 각 문제의 문제해결 히스토리 및 바운더리 컨디션 (Boundary Condition)을 이해하는 것은 올바른 문제해결에 있어서 가장 중요한 것 중 하나인데, 실제로는 컨설턴트가 접하게 되는 문제에 이러한 정보를 모두 다 공유하기 쉽지 않은 경우가 많다.

이렇게 기본적인 정보가 부족한 상황에서 문제를 정의하게 되고, 잘못된 문제 정의를 바탕으로 문제를 해결하는 컨셉을 도출하게 되면, 그 결과는 “ 이미 3년 전에 해봤던 방법입니다. ”, “ 10년 후에나 가능한 기술입니다. ” 등의 차가운 평가를 받게 되기 일수이다.

또한, 컨설턴트가 문제해결안을 도출하여 현업에서 그 유용성에 대하여 동의한 경우에도, 이후에 문제가 어떻게 해결되고 있는지, 부가적인 문제는 없는지 하는 모든 수행관련 정보는 컨설턴트의 손을 떠나 연구원이나 엔지니어들의 손에 맡겨지게 되며, 수많은 개선 후에 실현되었을 때, 컨설턴트의 성과라고 입증하기가 매우 어려워진다.

기존의 TRIZ에서는 문제를 정의하기 위한 방법으로, 문제 해결의 필요성을 입증하기 위한 What I want? (Purpose scale), 제대로 된 목표설정을 위한 ENV model 등이 존재하지만, 이들 역시 상기 숨어있는 기존의 지식들을 모두 들춰내는 데에는 한계가 있다.

본 논문에서는 이러한 문제해결의 히스토리를 정리함으로써 문제 해결의 바운더리 컨디션을 이해하기 쉽게 도와주고, 이를 쉽게 TRIZ의 기술적 모순으로 변환할 수 있는 CED을 제안하고, 사례를 통해 유용성에 대하여 확인해보고자 한다.

포스코 프로젝트지원실 TRIZ그룹 한승헌

E-mail : heon92@posco.com

TEL : (02)3457-1674

* 포스코 프로젝트지원실 TRIZ그룹 권민경

** 포스코 프로젝트지원실 TRIZ그룹 이희춘

2. CED

2.1 문제 해결의 길

2.1.1 부분(Partial) 해결안s

흔히들 TRIZ는 문제해결의 최적 루트를 제공하여, 시행착오를 줄일 수 있다고 한다. 따라서 처음 TRIZ를 접하는 엔지니어의 경우, 처음 도출된 초기 모순으로부터 완벽한 해결안이 나오지 않는 경우, TRIZ가 동작하지 않는다고 비판하는 경우도 있다.

하지만, TRIZ로 문제를 해결하더라도, 단지 시행착오를 줄일 뿐이지 한번의 TRIZ 원리를 사용해서 문제를 해결하는 경우는 극히 드물다.

TRIZ를 통하여 문제를 단순화하고, 이에 대한 솔루션 아이디어를 도출하고 나면 이 솔루션은 또 다른 부가적인 문제를 야기하게 되는 경우가 많으며, 따라서 초기 도출된 해결안은 보통 부분 해결안이라고 칭하고 부가 문제 역시 해결해야 할 문제로 TRIZ를 사용하던가, 최적화 기법을 사용하던가 하여 부가적인 문제들도 해결해야 최종적인 솔루션이 나오게 된다.

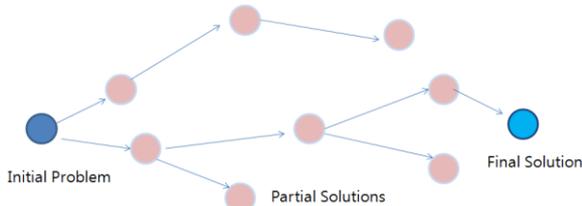


그림1: 부분 해결안..

2.1.2 부분 해결안s와 기술적인 모순

또한, 부분 솔루션과 부가적인 문제의 관계는 TRIZ의 기술적인 모순 즉 “~~을 진행한다면 (부분 해결안), A라는 파라미터는 좋아지지만(원 문제의 해결), B라는 파라미터는 나빠진다(부가 문제)”의 정의에 대입하여 바로 기술적인 모순으로 활용이 가능하다.

2.1.3 부분 해결안s 활용 시도

2000년대 초반, 삼성종합기술원에서는 이러한 부분 해결안을 Transition Action 로 정의하고, 이에 따른 부가적인 문제를 Undesirable Effect로 정의하여, 문제 문제 정의 단계에서 활용하는 시도를 하였다. 하지만, 이 역시 로직이 다소

복잡하고, 문제의 흐름을 한눈에 표현하는 데에는 불편함이 있었다.*(3)

이에 본 논문에서는 문제 해결에 있어서 필수적으로 도출되는 중간단계의 해결안, 즉 부분 솔루션을 시각화하여 누구나 이해하기 쉬운 형태로 만드는 방법을 제안하고, 이러한 방법의 유용성을 검증해 본다.

2.2 CED

2.2.1 CED 정의

CED이란 주요 문제에 대한 기존의 해결책, 신규 도출 해결책 등과 이에 따라 추가되는 부가 문제점(단점)들을 트리형태로 정리하여 아이디어 도출의 흐름 및 문제 해결과정에서 도출되는 모순 상황을 한눈에 알 수 있도록 하는 틀이다.

2.2.2 CED 작성 방법

① CED의 시작은 해결하고자 하는 문제를 최대한 간단한 문장으로 기술하는 데에서 시작한다. 문제가 기술된 후, 기존에 문제를 해결하기 위한 방법을 역시 간단한 문장으로 기술한다. 몇몇 해결안의 경우는 기존 문제 해결책이고, 프로젝트를 진행하면서 도출된 기본 문제에 대한 해결안도 주요문제의 옆에 기술한다. 필요한 경우, 각 해결안 밑에 그림을 삽입하여 추후 내용을 이해하는데 도움이 되게 하거나, 참고 문헌을 링크하여 이론적인 백그라운드를 확인한다.

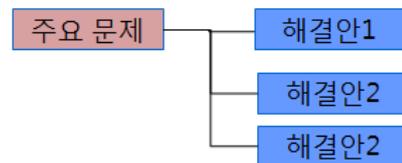


그림2: CED 1

② 1단계에서 정의한 문제를 해결하여야 하는 이유는, 아직 해결안들이 완전하지 못해서인 경우가 많다. 대개 해결안들은 주요문제는 해결할 수 있으나, 부가적인 문제를 만들어내기 때문이다. CED에서는 이러한 부가적인 문제를 해결안 아래에 하나의 문장으로 기술한다. 기술하는 형식은 아래 그림과 같다.

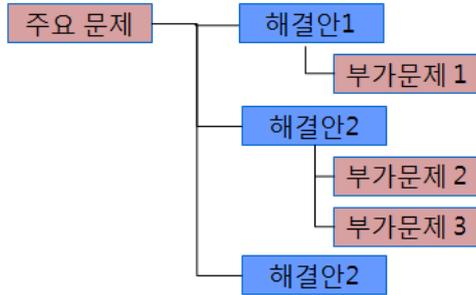


그림3 : CED 2

③ 부가 문제들에 대해서도 역시 2에서 기술한 바와 같은 방법으로 해결안을 기술한다.

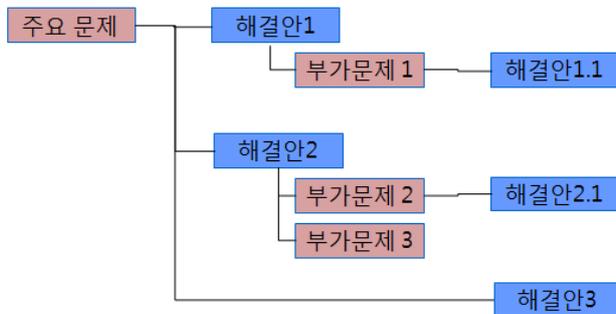


그림4 : CED 3

④ 문제의 완전한 해결책이 나올 때까지 반복한다.

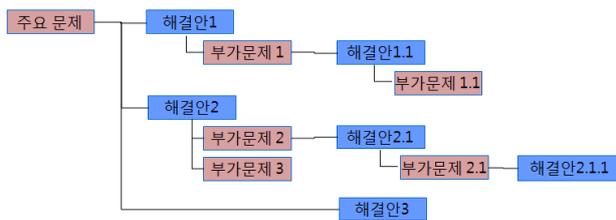


그림5 : CED 4

이상으로, CED을 작성하는 방법을 알아보았다.

CED작성의 로직은 실제로 매우 간단하며, 1시간 정도의 교육으로 충분히 숙지 할 수 있다.

2.2.2 CED과 기술적 모순

CED을 작성하는 목적 중 중요한 것은, CED 작성을 통하여 아이디어 발상과 더불어 지속적으로 기술적인 모순을 도출하여 이를

TRIZ를 활용하여 해결할 수 있음이다.

CED에서는 다음과 같은 방법으로 쉽게 기술적인 모순을 정의할 수 있다. 즉, 그림처럼 “해결안1을 도입하면, (상위)문제는 해결되지만, 부가문제1이 발생한다.”로 정의된다.

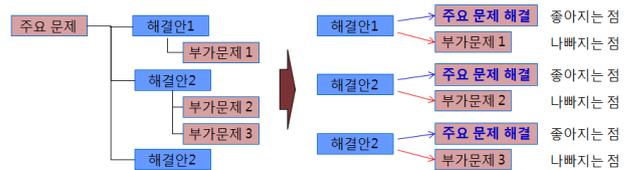


그림6 : CED 5

물론 그림6에서처럼 모든 해결안들이 기술적 모순에서의 컨트롤 파라미터로 정의할 수 있는 것은 아니다. 예를 들어 해결안2를 적용한다고 하더라도, 부가문제2로 기술되는 것이 “주요 문제를 해결하는데 부족하다”라는 점이 도출되거나, “너무 많은 비용이 들어간다.” 등의 비용 문제 등이 기술되는 경우에는 기술적인 모순으로 정의되기 힘들게 된다.

주요 문제를 해결하는데 부족한 경우에는 TRIZ의 물질-장모델을 사용하여 표준해를 활용, 문제 해결책을 도모해 볼 수 있으나, 비용 문제의 경우, TRIZ로 문제해결 할 수 있는 방법은 제한적이다.

하지만, 대부분의 경우 문제 해결안에는 다른 문제를 야기하는 부가 문제가 존재하고, 이를 기술적인 모순으로 정의하여 TRIZ를 활용하여 문제 해결을 하면, 실제 문제 해결에 있어 매우 유용하게 활용 가능하다.

2.2.3 CED의 활용

TRIZ를 활용하여 타인의 문제를 해결해야하는 컨설턴트의 경우, 본 논문의 초반에 설명하였듯이 기존의 기술을 이해하는데 많은 도움을 준다. 문제 의뢰자의 기존 기술을 이해하는 데에는 한계가 있는데, CED를 활용하여 공동으로 워크샵을 진행하면, 비교적 단기간에 기존 해결안의 문제점과 문제해결 히스토리를 쉽게 알아낼 수 있다.

CED의 가장 큰 활용성은 문제해결 히스토리

문제해결의 히스토리 : Concept Evolution Diagram

맵으로서의 활용이다. 연구 개발에 있어서 도출되는 무수히 많은 아이디어들은 채택이 되지 않는 순간 실행하기로 정해진 주요 아이디어에 의해 묻혀져서 실제로 기억에서 사라지기도 한다. CED 기법을 활용하여 연구개발과정에서의 모든 아이디어를 일목요연하게 정리한다면, 채택된 주요 아이디어의 실행이 실패했을 때, 기존의 아이디어를 다시한번 돌아보게 하는 역할을 할 수 있다.

CED를 활용하면, 연구개발에 있어서 도출되는 문제를 TRIZ로 활용하여 해결하기 쉽게 된다. 현 단계에서 막혀있는 문제를 가장 쉬운 방법으로 TRIZ의 기술적인 모순으로 정의하므로, 연구원이 직접 TRIZ를 활용하거나, 컨설턴트에 의뢰를 하기에 매우 용이하다.

CED는 기술문제뿐만 아니라, 사무간접 분야에도 매우 요긴하게 활용될 수 있다. 문제 문제 정의할 수 있는 여러 노우하우가 쌓여 있는 기술문제의 경우에 비해, 사무간접 분야에 TRIZ를 적용할 때에는 문제가 워낙 다양하고, 분야도 넓어서 가장 간단한 형태로 문제를 문제 정의하기에 매우 어려운 경우가 있는데, CED를 활용하면 복잡한 사무간접 분야 문제에서도 비교적 쉽게 기술적인 모순(전략 모순) 도출이 가능해진다. POSCO에서는 동양의 36계 원리를 활용한 POSCO고유의 “전략TRIZ”를 개발하여 사무간접 분야에 활용하고 있는데, 실제로 전략TRIZ 프로젝트를 진행하는 초입단계에 CED를 활용하여 전략모순을 도출하는 방법으로 활용하고 있다.

마지막으로, CED를 활용하여 얻을 수 있는 잇점은 T-HUB같은 집단 지성 활용 문제해결 프로그램에 활용하면 불특정 다수의 사람들이 문제를 이해하는데 매우 도움이 된다.

T-HUB란 전사적으로 이슈가 되는 문제를 TRIZ교육을 받은 다수의 인력을 대상으로 공유하고, 이 다수 인력의 지적능력의 합으로 문제를 해결해나가는 집단지성 프로그램의 일종이다.

T-HUB를 진행하여 문제를 공유할 때, 단순히 문제를 설명하는 것 보다, CED같은 상호 약속된 틀을 활용하여 시각화를 하면 좀 더 효과적으로 문제를 이해시킬 수 있을 뿐 아니라, 기존의

기술/아이디어를 제외한 아이디어를 도출하게 되어 프로그램을 좀 더 효과적으로 이끌 수 있게 된다.

2.2.4 CED와 기술 트리

문제에 대한 해결방법을 트리 구조로 전개한다는 점에서 CED와 기술 트리는 유사성이 있다.

기술 트리는 목적기능에 대하여, 그것이 무엇이고(What), 어떻게 해결이 가능한가(How)를 상위 단계와 하위단계가 상호 MECE (Mutually Exclusive and Collectively Exhaustive) 관계가 형성되도록 전개를 하는 방법론이다. 이는 연구개발의 초기단계에서 지금까지의 기술을 모두 도출하고, 연구개발의 방향을 설정하는데 유용하게 활용되고 있다.

CED는 지금까지의 기술을 도출한다는 점에서 기술트리와 매우 유사하다. 하지만, CED에서는 상위 단계와 하위 단계의 관계가 MECE를 충족해야할 필요가 없으며, 상위 단계와 하위 단계의 관계는 단순히 문제와 해결안의 관계이다. 또한 기술 트리에서는 기술의 문제점을 기술하지 않는데, CED에서는 기술의 문제점까지 부가 문제로 기술하고, 이를 TRIZ의 기술적인 모순과 연계한다. 마지막으로 차이점은 문제 해결 과정 전체에 걸쳐서 CED는 지속적으로 업데이트 되어 문제 해결 맵으로 활용하게 된다는 점이 다르며, 이러한 다른 점들이 CED를 좀 더 유용하게 활용할 수 있게 하는 점들이다.

3. CED의 활용 예

CED의 작성방법에 대한 이해를 돕기 위해, 실제 사례를 예를 들어 설명토록 한다. 단, 회사의 보안정책에 의해 자세한 내용은 생략한다.

3.1 활용 사례

크기가 작은 사출물을 제작하는 본 공정은 상/하부에 강화유리가 있으며, 그 사이에 제품의 형상을 만드는 플라스틱 몰드가 있다. 본 방법으로 사출하게 되면, 기존 사출 방식 대비 획기적인 공정비용 절감이 가능한데, 문제는 사출물이 굳고 난 후에는 지렛대를 활용한다면

상판 유리를 제거하기는 용이하지만, 매우 얇은 사출물을 하판 유리에서 분리하기는 매우 어렵다. 작업자가 나이프를 사용하여 사출물을 잘 떼어내어도 가끔 사출물이 찢어지기도 한다.

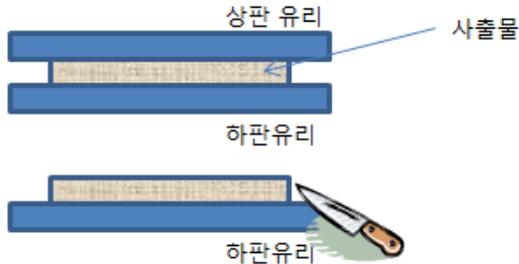


그림 7 사출물 문제

지금까지의 문제상황을 CED로 정리하면 다음과 같다.

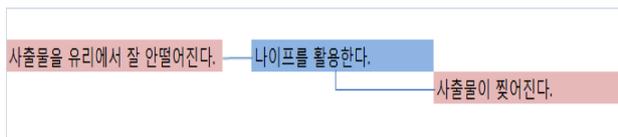


그림 8 사출물 문제 CED 1

이를 기술적인 모순으로 변환하면, “나이프를 활용하면, 사출물을 유리에서 Detach할 수 있으나, 사출물이 찢어진다”로 표현할 수 있다.

40가지 원리등의 TRIZ를 활용할 수도 있으나, 본 프로젝트에서는 FOS(Function Oriented Search) 기법을 활용하여, 첫번째 아이디어들을 도출하였다. 콘택트렌즈 분야에서는 주물을 비틀어 사출물을 제거하는 방법을, 초콜릿 공장에서는 초음파를 활용하여 초콜릿을 떼어내는 방법을, 다른 분야에서는 공기구멍을 뚫어 진공을 없애는 방법을 활용하고 있었다.



그림 9 사출물 문제 CED 2

1차 아이디어 도출을 통해서 두번째 기술적인 모순인 “비틀어서 제거하면, 사출물을 Detach 할 수 있으나, 사출물이 손상된다”라는 두번째 기술적인 모순을 도출하였다.

기 설명한 바대로, 비용이 상승하는 초음파 방법이나 공기구멍 안은 기술적인 모순으로는 정리가 힘들었다.

이를 바탕으로 2차 아이디어 발상을 진행하였다. 이때는 자원분석을 통하여 공기라는 자원을 활용여 진공상태에서 사출물을 떼어내는 아이디어를 도출하였다.



그림 10 사출물 문제 CED 3

실제 진공챔버안에서 Test를 진행한 결과, 충분한 진공을 만들기 어려웠고, 설사 주변 환경을 중반한 저진공상태로 만들더라도, 장치를 챔버 안에 넣기가 매우 어려웠다. 이에 따라 단순하지만 Suction 장치로 직접 기계적으로 Detach하는 방법을 생각하였으나, 역시 충분한 힘을 가하기 어려웠다.

이에, 부분적으로 진공을 만들어 사출물을 잡고, 라운드 형태를 활용하여 Detach하는 방법을 고안하였으나, 실험 결과 이 역시 진공으로는 사출물을 강하게 잡지 못하는 문제가 있었다.

마지막으로, TRIZ의 사전 반대 조치를 활용하여 사출 전에 유리면에 Tape를 부착하고, 이 Tape를 잡고 사출물을 Detach하는 방법을 생각하게 되었다.

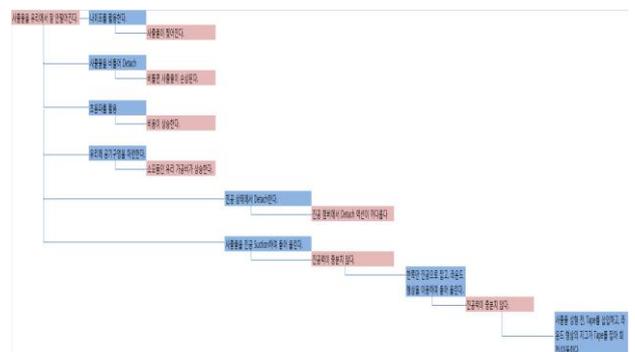


그림 10 사출물 문제 CED 4

문제해결의 히스토리 : Concept Evolution Diagram

실험결과 사출물이 매우 쉽게 떨어지는 것을 확인하였고, 라운드 형상을 활용하여 사출물이 깨지는 것도 방지 할 수 있었다.

이처럼 CED를 활용하면서, 본 프로젝트에서는 향후 진행될 문제를 즉각적으로 파악하고, 그때그때의 문제를 TRIZ로 모델링하여 문제를 해결하였다. 물론, 본 예제에서는 특정한 한 방향으로 잘 진행되어 문제가 해결되었지만, 만약 사전반대조치 아이디어가 불가능했다면, 초기문제 중 하나인 “ 비틀어서 제거하면, 사출물을 Detach 할 수 있으나, 사출물이 손상된다” 라는 문제로 다시 돌아가서 문제해결을 도모하거나, 진공챔버 안을 좀 더 개선하는 아이디어를 발전시킬 수 도 있는 것이다.

4. 결 론

지금까지 CED를 작성하는 방법과 그 활용하는 방법, 마지막으로 실제 예제를 통하여 CED가 어떻게 동작하는지를 살펴보았다.

CED는 연구개발, 혹은 사무간접분야의 문제에 적용하여 문제 해결의 지도로 활용하거나, 컨설턴트의 문제 정의 시 매우 유용하게 활용될 수 있으며, 실제 6여년간 다수의 프로젝트에 직접 적용함으로써 그 유용성을 입증하였다.

향후, 기존에 시도되었던 T-HUB라는 프로그램을 좀 더 발전시킨 회사 집단 문제해결 프로그램에 본 논문에서 소개한 CED를 활용한다면, 개인의 창의성 증진에 집중한 기존 TRIZ에서 나아가 조직 전체의 창의적 문제해결 능력을 향상시킬 수 있으리라 기대하며, 본 논문을 마무리한다.

참고문헌

- (1) 노정은, 한승현 “T-HUB” 2010 TRIZ Future Conference
- (2) 권민경, 김중호 “Case study using Strategy TRIZ” 2013 TRIZ Future Conference
- (3) 송미정, V. Leniachine, 김성철, S. Antonov, 2004 “Hands-on Logic for Inventive Problem Solving – I. Formulating Technical Contradiction from Initial Situation”