

## 비정형화된 문서간 추적성 분석 및 그 가시화 방안 제시

김은희<sup>1)</sup> 송덕용<sup>2)</sup> 황진상<sup>3)</sup> 정재천<sup>4)</sup>

1) 세종대학교, 2) 주식회사 부품디비, 3) 주식회사 부품디비, 4) 한전국제원자력대학원대학교

### A Study on the Methodology of Traceability Analysis and Visualization between Non-standardized documents

EunHee Kim<sup>1)</sup>, Duck Yong Song<sup>2)</sup>, Jin Sang Hwang<sup>3)</sup>, Jea Cheon Jung<sup>4)</sup>

1) *Sejong University*, 2) *PARTDB. Co., Ltd.*, 3) *PARTDB. Co., Ltd.* 4) *KEPCO International Nuclear Graduate School*

**Abstract** : We propose a methodology to automatically extract the requirements from the documents and check the traceability between them. The documents include not only the text file but also PDF or image files. We also suggest a method to visualize the result with maps, numbers, and graphs. By comparing the results with those of expert reviews, we show that it is necessary to use knowledge-based method in future instead of the word-based method for improving the reliability. The results give more values when they are applied in already existing documents than those of newly developed product.

**Key Words** : Requirement, Traceability, Requirement Tools, Requirement Process, Traceability Tools

---

\* 교신저자 : [elliee.kim@gmail.com](mailto:elliee.kim@gmail.com)

\* This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

### 1. 서론

원전, 철도, 항공, 우주 분야에 적용되는 소프트웨어는 안전과 결부되어 그 성능 및 안전을 보장해야 하므로 관련 개발방법론을 준수해야 한다. 이에 대한 소프트웨어를 안전등급 소프트웨어라 명명하고 있으며 관련된 개발방법론 및 관련 도구들을 활용하여 개발함으로써 생산성과 신뢰성을 향상 시키는데 중점을 두고 있으며 관련 연구들이 지속적으로 진행되고 있다. 결과적으로 소프트웨어 등급 분류, 산업분야별 기능 및 수명주기, 고품질 달성을 병행하고 있다.

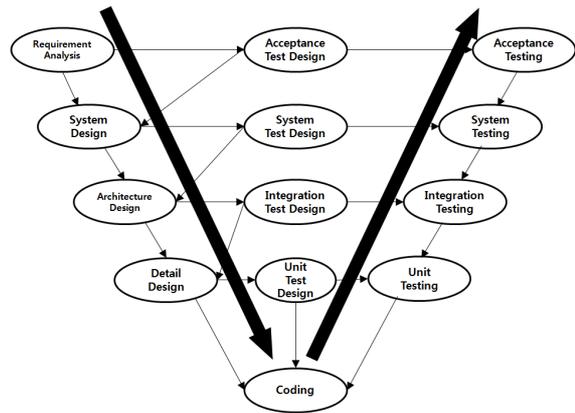
원자력 규제 위원회(NRC: Nuclear Regulatory Commission)에서는 이와 관련된 규제지침(Regulatory Guide)을 제시하고 있으며 소프트웨어 또는 시스템 개발시 이를 준용하도록 하고 있다. 이 규제 지침은 IEEE 산업표준을 근거로 하고 있으며 산업표준 지침을 활용하여 개발하는 것은 해당 분야의 신뢰성을 인정하는 것을 의미한다. 각 산업에서 개발될 소프트웨어는 자체 검증 및 인허가를 취득하여 최종 운용하게 되며 결과적으로 생명을 담보로 하는 제품에 필수 요건으로 자리매김을 하고 있다. Table 1은 각 산업군에서 제시하고 있는 안전등급 소프트웨어에 대한 개발표준의 일례를 제시하고 있다.

<Table 1> 산업군별 소프트웨어 개발 기술표준

산업군	기술표준	형상관리 도구	요건관리 도구
국방/항공	DO-178	Synergy, Change	DOORS, Cradle, Modelino
원자력	IEC60880	iSAS	
자동차	ISO26262, AUTOSAR	-	
철도	EN50128 (IEC6229)	-	

Figure 1은 소프트웨어 개발에서 주로 사용되거나 인용되는 V 모델(위키피디아 참조)을 도식화 한 것으로 요구사항을 기반으로 관련 설계문서 및 소프트웨어 검증 및 테스트와 밀접하게 수행하여 품질

및 성능을 보증하는 보편화된 방법론이다. 즉 안전 등급 소프트웨어 개발시 일반적인 방법론이며 설계와 코딩, 시험의 세 가지가 일치하고 정확한지를 검증하기 위한 모델로서 개발시간 단축 및 시간을 감소시킨다고 알려져 있다.



[Figure 1] V 모델 예시

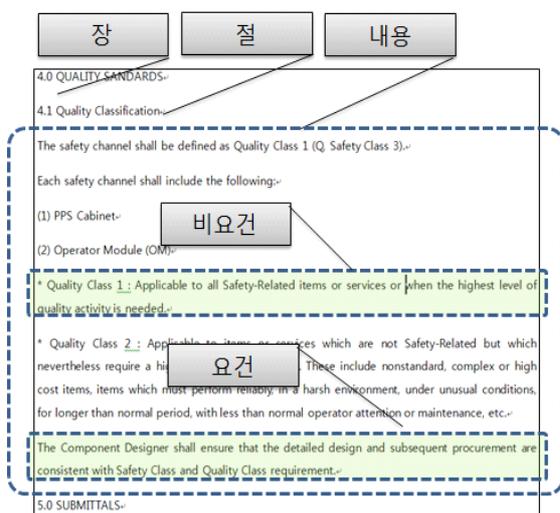
현재 운영되고 있는 원전 안전등급, 그에 준하는 산출물들은 추적성 분석 등의 가치화 방법 등이 정형화되어 있지 않기에 원전 운영사가 이를 확인함에 있어 많은 어려움이 존재한다. 특히 문서기반으로 작성된 설계문서 등은 과거에 PDF, 이미지 등과 같은 저장방식을 채택하고 있어 설계변경 발생시 이를 개정하기 위해서는 모든 문서를 텍스트 형태로 변환하고 추적성을 관리해야 하는 어려움이 존재한다. 본 연구에서는 이미 개발된 소프트웨어 및 문서 간의 추적성을 수행하기 위해 서로 다른 2개의 문서에 대해 요건의 일치성과 추적성을 분석할 수 있도록 하기 위한 업무 프로세스를 제시하고자 한다. 즉, 상기 방법론에 근거하여 자동화된 시스템으로 개발하였을 경우, 요건 및 추적성을 원활히 관리하는 프로세스 및 도구 개발 방안을 제시하고자 한다. 근래에 개발된 소프트웨어는 요구사항에서부터 구체화된 개발절차에 따라 개발되나 과거에 개발된 문서들은 이러한 기술적 근거나 요건관리, 추적성 분석이 제대로 수행되지 않고 그 근거를 확인할 수 있는 방안, 기술적 방안이 모호한 실정이

다. 상기에 기술된 기술표준을 근거하여 요건관리를 통해 형상관리에 사용될 수 있는 방법론을 제시하고 형상관리를 원활히 수행할 수 있는 절차 및 프로그램 개발에 대한 내용을 제안함으로써 이미 활용되고 있는 소프트웨어 품질보증에 활용될 수 있기 위함이다. 본 논문에서는 상하위 문서간의 일치성을 자동적으로 검사하여 일치횟수를 사용자에게 가시화 시켜주고 두 개 문서간의 일치성이 있는지를 사용자가 판단할 수 있는 방안을 제안하였다.

## 2. 요건관리 및 추적성분석 자동화 방법론

### 2.1 설계문서 구조

일반적인 안전등급 소프트웨어 분야의 설계문서는 장, 절, 항으로 구분되어 가독성 및 인식성, 일관성이 높은 편이다. Figure 2에는 장, 절, 요건으로 구성된 요건문서 구조의 일부 예이다. 요건문서의 경우에는 shall, should, must 등과 같은 키워드를 포함하고 있어 이를 기준으로 요건과 비요건(Non-Requirement)으로 구분하기도 한다. 본 연구에서는 일반적인 문서구조를 포함하는 각 문서의 구조화 및 상하위 문서간의 일치성을 검사하여 자동화 하는 방안을 중점으로 연구하였다.



[Figure 2] 일반적인 문서구조(예시)

### 2.2 문서 Parsing 방법

원전과 같이 설계기간이 길고 다수의 설계문서가 산출되는 경우, 설계문서 원본을 보존하기 위한 일반적인 방법으로는 PDF 또는 이미지파일 형식으로 저장하여 장기간 보존하는 사례가 다수 존재한다. 특정 문서편집기/작성프로그램에 의존하지 않고 저장하는 장점으로는 원본의 내용을 그대로 전달할 수 있는 반면, 문서 수정사항 발생시 최초 문서를 작성한 프로그램이 항상 같이 존재해야 한다. 운영체제 변경에 따른 과거버전의 프로그램의 실행불가로 인해 많은 문제를 야기하고 있다.

본 연구에서 제시하고 있는 각 문서간의 연계성, 일치성을 확인하기 위해서는 설계문서를 컴퓨터가 인식할 수 있도록 텍스트 화 하여 관리할 수 있는 방법이 필요하다. 이를 해결하기 위해 PDF 파일, HWP, 이미지 형태 문서 등을 Parsing하여 컴퓨터(프로그램)가 인식할 수 있도록 문서화 하였다. 과거에 저장된 대부분의 설계문서는 정형화된 파일형식과 비정형적인 포맷으로 저장된 설계문서로 구분할 수 있다.

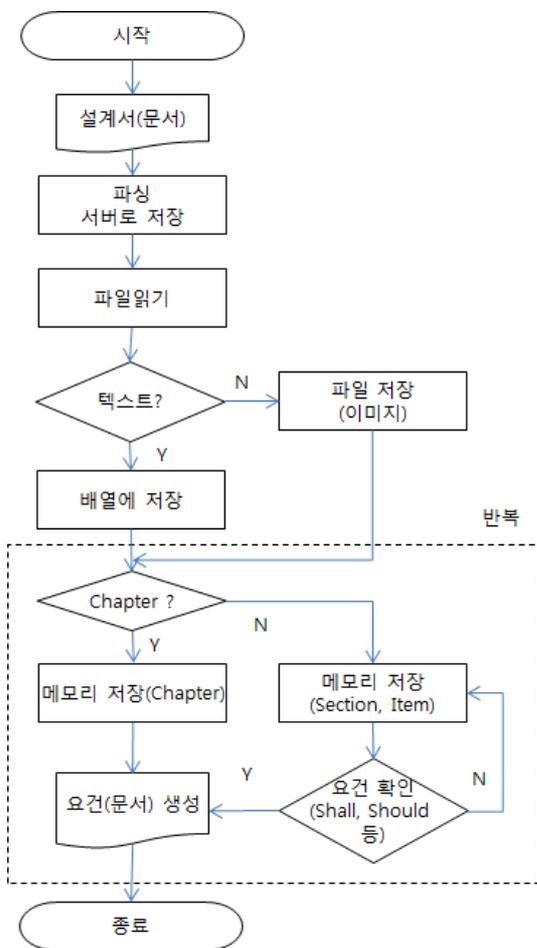
#### 2.2.1 정형화된 문서 Parsing 방법

설계문서는 PDF 파일과 같이 텍스트, XML로 재변환하여 글자를 인식할 수 있는 경우와 이미지 프로그램에 의해 저장된 경우에 바로 인식할 수 없는 것으로 나뉜다. 글자로 인식할 수 있는 경우에는 글자와 그림을 따로 분리하여 저장할 수 있고 “should”, “shall”, “must”와 같이 요건과 비요건 키워드가 포함되는 문장으로 구분도 가능하다.

대부분의 문서는 텍스트 인식이 가능하며 잘 알려진 PDF, 워드, 한글과 같은 파일포맷을 인식할 수 있는 파서들이 많이 개발되고 있다. 파서는 각 문장을 하나의 단락으로 분리하여 문장단위로 ID를 자동으로 부여가 가능하다. 따라서 Parsing된 문서는 고유 ID를 생성 및 요건과 비요건으로 분리하여 저장 및 관리가 가능하다.

Figure 3에는 문서 Parsing을 위해 일반적인 설계문서가 포함하고 있는 그림과 글자를 사용자가

원하는 형태로 분리하여 저장하고 각기 고유한 ID를 부여하여 문장단위로 분리하여 새로운 파일, 다른 형태의 문서로 저장 및 관리가 가능한 절차를 도사하였다. 이는 실제 한글, 워드 문서를 분석하기 위해 일반적인 설계문서를 Parsing하여 설계문서 내의 단어 중 shall, should, must와 같이 요건이 되는 문서들만을 추출하고 이를 독립적인 문서로 추출 및 저장하는 기법에 대한 절차를 flow-chart 형식으로 도시한 것이다.



[Figure 3] Parsing Process of Design Document

### 2.2.2 비정형화된 문서 Parsing 방법

비정형화된 문서는 이미지 형태로 저장되어 프로그램이 텍스트를 인식이 불가능 한 경우를 의미한다. 이런류의 문서를 인식하기 위해서 OCR(Optical Character Recognition) 기법을 활용하여 문서를

인식하는 방안을 제시하고자 한다. 본 연구에서 수행한 설계사양서는 이미지 형태로 저장된 PDF 파일을 읽어들이어, OCR 프로그램으로 텍스트화 하여 추적성 분석을 위한 데이터로 활용하였다. OCR 프로그램의 단점으로는 본문 내용 이외에 문서에 포함되어 있는 머리말, 꼬리말, 페이지 번호, 문서명 등이 텍스트로 인식되어 일부 편집을 수작업으로 해야 하는 번거로움이 발생한다.

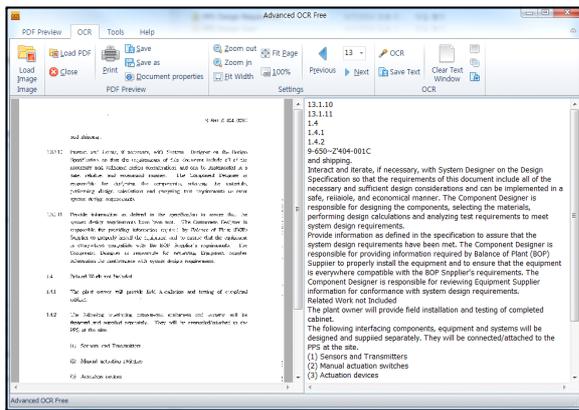
설계사양서를 OCR 프로그램으로 읽어들이어 글자를 추출하는 경우에는 다수의 오타와 글자 폰트 유형, 폰트 크기에 따른 변환이 100% 정확하게 변환되지 않는다. 예를 들어, 문자 “T”는 숫자 “1”로 인식되는 경우가 많고 원본 해상도가 낮은 문서의 경우에는 변환시 오타자에 따른 수정작업이 병행되어야 한다.

설계문서 중 낮은 해상도의 문서는 OCR 프로그램을 활용할 경우, OCR 프로그램 내에 보유하고 있는 확대 기능을 통해 원본 문서를 최대한 확대하여 텍스트 변환시 원본에 근접한 인식이 되도록 하는 것이 변환에 따른 오타를 최소화 할 수 있는 방법이다. OCR 프로그램은 상용소프트웨어가 다수 존재하나, 본 연구에서는 저가형 OCR 인식프로그램을 활용하였고, 일부 웹기반 환경에서 이미지를 Parsing하여 글자로 변환해주는 무료서비스도 있으나 연속적인 파일에는 적합하지 않기에 추천하지 않는다.

본 연구에서는 Figure 4에 도시한 바와 같이 저가형 프로그램인 Advanced OCR Free 소프트웨어를 활용하여 이미지 문서(하위 설계문서)를 Parsing하였으며, 앞서 설명한 바와 같이 해당 문서를 요건과 비요건으로 구분하여 저장하도록 하였다. OCR 프로그램의 좌측 윈도우는 원본 PDF 파일을 가시화하며 우측 윈도우에는 텍스트를 추출하여 가시화 한다. 이는 원본문서와 추출된 텍스트가 일치하는지를 비교하기 위함이기도 하며 추출된 텍스트를 파일로 저장하는 기능을 포함하고 있다.

### 2.3 문서간 일치성 자동화 방법

문서간 일치성을 확인하기 위해서는 두 개의 문서를 텍스트화 하는 것이 선행되어야 하며 앞서 설명한 절차에 준하여 수행하였다. 프로그램에서 인식할 수 있는 텍스트로 변환하고 각 문서에 대해 고유 ID를 부여하였다. 설계요건문서(상위문서)와 설계사양서(하위문서)에 대해 고유 ID를 부여하여 문서간의 일치성을 검사하였다.



[Figure 4] Parsing Process of Image Files(Sample)

본 연구에서는 원자력발전플랜트에서 사용하고 있는 일부 설계요건문서와 설계사양서를 활용하였으며, 요건문서는 392개의 요건(문장)을 설계사양서는 303개의 문장으로 구성된 것을 활용하였다. 두 개 문서 연계(상관) 관계를 분석하기 위해서는 단일요건(문장)에 포함되어 있는 단어를 이용하여 설계사양서의 단일 사양을 단어 단위로 대조하는 역무를 수행하기 위해 단어검색 기법을 활용하여 개발하였다. 일치하는 경우는 상위 요건 ID의 Name을 하위문서(설계사양서)내 일치되는 설계사양서 ID Name의 우측 셀에 표현하도록 개발하였다.

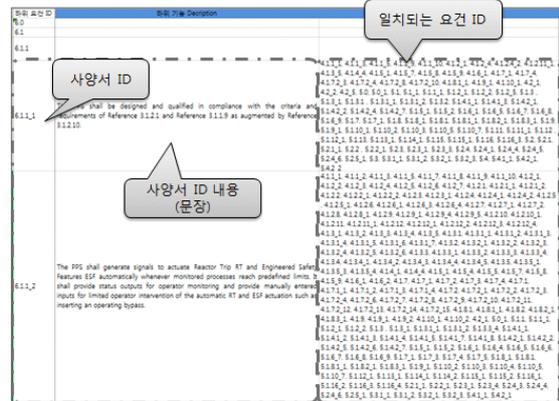
Figure5는 설계사양서 ID Name 이 “6.1.1\_1”인 단일 문장을 상위 요건과 일치하는지 여부를 확인하여 우측 테이블에 표시한 것이다. 대략 셀마다 100개 이상의 상위 요건과 일치되는 것을 확인할 수 있었다. 또한 단어 위주로 일치성을 검사하면 정

관사, 조사, 마침표, 특수문자에 대해서도 일치성을 확인하기 때문에 더 많은 수의 문장이 일치됨을 표시하기 때문에 이를 배제하기 위한 루틴을 포함하도록 개발하였다.

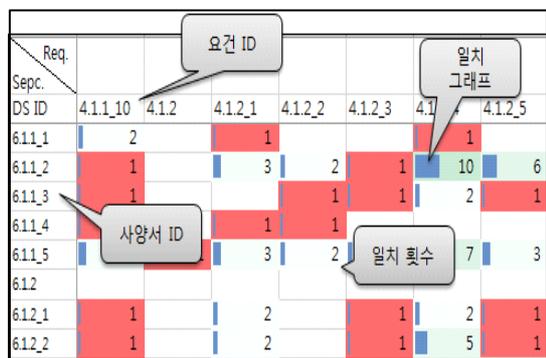
### 2.4 문서간 일치율 가시화 방법

상위문서(요건)와 하위문서(상세설계서) 간의 일치되는 ID를 각 셀에 표시하여 그 연계성을 가시화한 것이 Figure 6이다. 좌측행은 설계사양서의 문장을 고유 ID로 할당하여 정리한 것이며 열은 상위요건에 대해서 부여된 ID를 나열한 것이다.

각 셀은 해당 열의 요건과 해당 행의 설계사양서간의 일치 횟수를 카운트하여 나타낸 것이다. 일치한 횟수가 많은 셀에 대해서는 그래프 및 색을 부여하여 가독성을 높도록 가시화하였다. Figure 6은 다수 요건과 설계사양서간의 일치되는 숫자를 그래프 형태로 표시한 것이다.



[Figure 5] 요건과 사양서간 일치 가시화(예시)



[Figure 6] 일치성 가시화 방안(예시)

### 3. 문서간 일치성 검증

#### 3.1 일치성 검증방법

앞에 제시한 일치 방안은 단어 위주로 일치성을 평가, 확인하기 때문에 해당 문장의 상하위 문서 간 일치여부에 대한 신뢰도 확보가 필요하다. 따라서, 관련 분야의 전문가 판단에 의해 일치여부를 먼저 확인 후 단어위주의 상하위 요건간의 일치성에 대한 평가가 병행되어야 한다. 이를 위해서 상위문서(요건)와 하위요건(설계사양서)에 대해 일치성을 확인하여 자동화(프로그램) 대비 일치성이 신뢰할 만한지 검토하였다.

Table 2에 정리한 바와 같이, 요건 7개(문장)와 하위요건(상세설계서) 390개(문장)를 대상으로 전문가를 통해 두 문장(ID) 간의 일치 여부를 수작업으로 확인하였다. 표 2에 의하면 전문가 검토는 일치되는 장, 절, 항에 대한 전문가 평가를 거쳐 수행한 것이며 프로그램에 의한 평가는 전문가 평가에 비해 문장간 일치수가 매우 높게 분석되었다.

<Table 2> 문서간 일치성 검사결과 비교

요건 ID	전문가 검토 일치 수(개)	단어기반 일치 수(개)
4.1.2_1	4	172
4.1.2_2	3	273
4.1.2_3	3	248
4.1.2_4	4	134
4.1.2_5	5	243
4.1.2_6	5	263
4.1.2_7	5	228
합계	29	1,561
평균	4.1	223

Table 2에 정리한 바와 같이, 전문가 검토에 의한 하위요건(설계사양서)의 일치수는 총 29개이며 평균은 4개로 산출되었다. 단어기반 일치의 경우는 총 1,561개이며 평균 223개로 산출되었다. 단어기반으로 일치, 연관성을 찾을 경우에는 전문가 평가 대비 분석에 비해 53.8배나 많은 연관성을 찾아주

는 반면 정확도는 2% 미만으로 확인되었다.

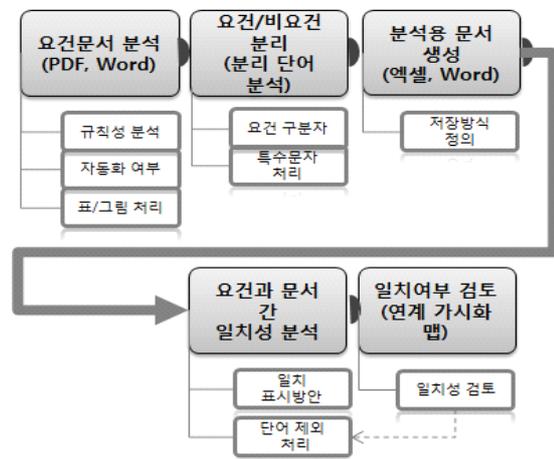
따라서 컴퓨터 알고리즘에 의한 일치, 연계성 자동화 및 추적성 분석은 현재 기술로는 다소 취약성이 많은 편이다. 다만 해당분야 전문가에 문서간의 일치성 평가를 위해 구간을 정해준다면 일치율이 높아질 가능성도 배제할 수 없다.

#### 3.2 추적성 및 연계절차 개선 방안

##### 3.2.1 현재 업무 프로세스

추적성 및 연계에 대한 자동화 부분은 문장에 대한 의미를 파악하여 일치 여부를 판단하는 것이 가능하나, 지식기반 기법을 적용하는 것은 기술적으로 제약사항이 발생한다. 전문가 의견에 따르면 비교하고자 하는 대상의 구문이 관련된 장, 절, 항으로 구체화된다면 프로그램에 의한 연계성 분석 및 검토가 용이하다는 의견을 제시하고 있다. 특히 Figure 7에 제시한 상위문서(요건)과 하위문서(상세설계서)간의 일치성을 검토하기 위해 검색 대상에 제외할 수 있는 단어처리 편집기를 부과하여 개발함으로써 일치성 평가에 대한 신뢰도를 높일 수 있다. 이는 문서간의 연계성 및 추적성을 검토하고 분석할 때 제외할 단어를 등록하여 요건과 상세설계 문서간 신뢰성을 높일 수 있기 때문이다.

일치성을 가시화 하기 위해서는 상하위 문서간의 일치성을 보여주는 가시화 맵(map)을 통해 전체적인 연계성을 가시화 할 수 있다. Figure 7은 일반적인



[Figure 7] Consistency verification and Analysis Process

인 문서 간 추적성, 연계성 분석 수행에 필요한 절차를 나타낸 것으로써 문서간 연계성 검증 절차를 예시한 것이다.

**3.2.2 개선된 업무 프로세스 및 관련 프로그램 개발제안**

본 연구에서는 앞서 설명한 절차에 이미 작성된 요건과 상세 설계서간의 문서 Parsing을 통해 상호간의 연계성을 파악하는데 중점을 둔 것이다. Figure 8에는 이전에 제시한 방법론을 개선하여 구조화된 방법으로 문서간 추적성분석 및 연계성을 검토하는 자동화된 방법 5개를 절차화 하여 프로그램으로 개발할 수 있는 방안을 제안하고자 한다.

첫째로 워드, 한글, PDF 파일은 XML 형태로 저장하여 각 문서에 존재하는 글자와 이미지(그림)를 분리하여 저장할 수 있어야 한다. 근래에 들어서 문서들은 XML 형태로 저장하고 관리하는 기법을 활용하므로 이를 활용하는 것이 유리하다. 그 이외에 앞서 설명한 이미지 형태의 문서들에서도 텍스트 추출을 OCR 프로그램을 활용하거나 오픈소스를 활용하여 개발이 가능하다. 본 연구에서는 이를 웹(Web) 기반환경으로 개발하여 연구에 적용하였다.

두 번째는 문서내에 포함되어 있는 문장에서 요건과 비요건을 분리해야 한다. 요건문서의 경우 있어 사용자 편의에 따라 임의로 문장 내에 필수 단어들 포함하는 문장에 있어 요건/비요건으로 분리하고자 할 때 활용할 수 있는 단어 편집기가 매우 유용하다고 할 수 있다. 이는 특수문자 등과 같이 배제되어야 하는 단어를 관리(추가, 수정, 삭제)하기 위함이다.

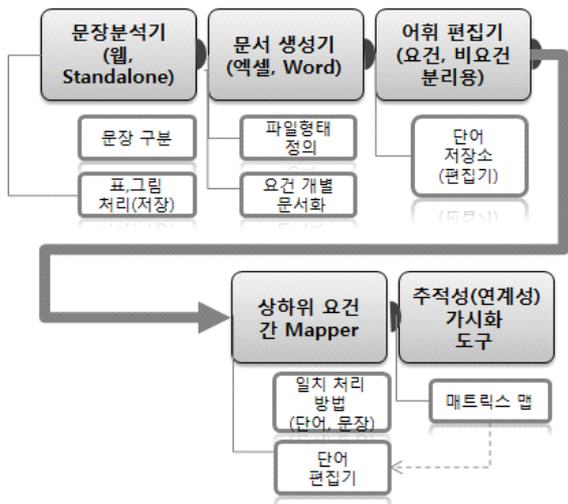
세 번째는 상위요건과 하위문서간의 연계성 분석을 위해서는 문서구조에 맞게 추출된 문장을 구조적으로 저장한다. 각 문장은 고유 ID를 부여하여 식별하고 식별된 문장에서 각 단어를 분리하여 하위 문서(설계사양서)의 문장을 일일이 검색하여 일치성을 확인한다. 하위문서(설계사양서)는 상위문서(요건)이 다수 존재할 수 있으므로 이를 카운트하여 몇회나 일치되는지를 표시할 수 있어야 한다. 동

일한 요건 내에서 중복으로 일치되는 경우에는 카운트를 배제하도록 한다.

문서간의 연계여부를 확인하거나 편집할 수 있는 도구로는 여러 분야에서 활용되고 있는 DOORS 라는 제품이 있다.

네 번째는 상위요건과 하위문서간의 일치성을 맵핑(mapping) 해주거나 찾아주는 프로그램을 개발하는 것이며 2장에 설명한 바와 같다. 현재 기술로는 단어 위주로 문장간의 비교를 수행하나, 향후에는 문장 단위로 비교할 수 있는 추론기법에 의한 개발의 보완이 가능하다. 특히 현재에도 제외 단어, 특수 문자, 마침표 등의 문장간 비교시 제외해야 할 단어들 편집할 수 있는 편집기 개발을 통해 보다 나은 조건의 비교검색이 가능할 것으로 판단된다.

마지막으로 상하위 문서간의 연계성을 확인 후 이를 전체적으로 도시할 수 있는 매트릭스 개발하고 자동화하여 두 개 문서간의 연계성을 심도 있게 가시화하고 해당 연계가 표시된 부분을 선택(클릭)하면 상위 요건정보와 하위상세설계서 정보가 같이 윈도우 상에 표현됨으로써 보다 가독성 및 판단에 도움이 될 것으로 판단된다.



[Figure 8] 일치성 검증 및 분석 도구 예시

#### 4. 결 론

본 연구를 통해 정형화되지 않은 PDF, 이미지 형태로 저장된 문서들로부터 텍스트 형태의 문서정보를 추출하여 각 문장에서 기술된 주요 단어들을 활용하여 문서간 일치성을 자동으로 검사하는 방법론을 제시하였다. 또한 두 개 문서간의 추적성 분석을 수행하기 위해 문장간 일치성을 확인할 수 있는 가시화를 그래프/숫자를 혼합형 그래프로 가시화 하는 방안을 제시하였다. 특히 전문가 검토 결과와의 비교를 통해 문장 간의 일치성/추적성 분석 단어 위주의 문장간 비교 방식의 신뢰성을 높이기 위하여 추후에는 지식기반 관리도구를 개발하는 것이 적절한 것으로 판단된다. 본 연구에서 제안하고자 하는 방향은 설계 초기단계에서 적용하는 상용제품도 있지만 기존 설계문서에 대한 평가를 위한 제품, 프로세스도 활용된다면 그 효과 및 의의가 있다고 할 수 있다.

#### Reference

1. KNS : Safety-related essential software for configuring the system safety analysis research, KAERI, Hansung Sohn, 2004
2. Regulatory Guide 1.152, Revision 2. "Criteria for Use of Computers in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 2006
3. Regulatory Guide 1.168, Revision 1. "Verification, Validation, Reviews and Audits for Digital Computer Software Used in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 2004
4. Regulatory Guide 1.169, "Configuration Management Plans for Digital Computer Software Used in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 1997.
5. Power industry technical standards EME-3100-2005, "Computer software verification and validation"
6. Power industry technical standards EME-3310-2008, "Software Configuration Management"
7. NUREG/CR-6101, "Software Reliability and Safety in Nuclear Reactor Protection Systems", 1993
8. PWR nuclear regulatory guidelines 8.15, "Digital computer software system used in the safety Verification, Validation, Inspection and Audit", 2009
9. PWR nuclear regulatory guidelines 8.20, "Digital computer software used in safety systems life cycle development process", 2009
10. Defense Solution Book, MDS Technology
11. <http://www.pdfcore.com>
12. [http://ko.wikipedia.org/wiki/V\\_%EB%AA%A8%EB%8D%B8](http://ko.wikipedia.org/wiki/V_%EB%AA%A8%EB%8D%B8)
13. KINS, 8.26 Safety Software Review - Software life cycle design results, <http://scale.kins.re.kr/service/search/mobile/view.do?hseq=704>;