2000 추계학술발표회 논문집
한국원자력학회

공급자 조사 방법에 의한 원전 상용소프트웨어 인정 프로세스
Commercial off-the-shelf Software Dedication Process based on the Commercial Grade Survey of Supplier

김장열, 이장수, 천세우, 이기영, 박종균
한국원자력연구소
대전광역시 유성구 덕진동 150번지

요 약

상용 소프트웨어 인정 프로세스는 하드웨어 상용기기 인정 및 승인 프로세스와 마찬가지로 4가지 방법이 있는데 실제 적용시 이들의 방법을 상호 조합하여 사용한다. 일반적으로 상용 소프트웨어 인정(dedication) 방법에는 크게 4가지가 있는데 method 1은 특수실험 및 감사이며, method 2는 공급자 조사, method 3는 소스검증, method 4는 qualification vendor list 및 운전이력 데이터에 의한 인정 방법이다. 본 논문은 method 2인 공급자 조사 방법에 의한 상용 소프트웨어 인정 프로세스에 관하여 NUREG/CR-6421의 기본 개념을 바탕으로 하고 ERPI/TR-106439 기준을 tailoring policy를 적용하여 절차적 관점에서 commercial grade survey 방법에 의한 상용 소프트웨어 인정 프로세스를 제안하였다. 제안한 상용 소프트웨어 인정 프로세스는 소프트웨어 품질보증 관점에서 보았으며 하드웨어 상용기기 인정 및 승인을 제외한 상용 소프트웨어 인정 프로세스에 국한하여 절차적 관점에서 적용할 수 있도록 하였다.

Abstract

Commercial Off-The-Shelf(COTS) Software dedication process can apply to a combination of methods like the hardware commercial grade item dedication process. In general, these methods are : methods 1(special test and inspection), method 2(commercial grade survey of supplier), method 3(source verification), and method 4(acceptance supplier/item performance record). In this paper, the suggested procedure-oriented dedication process on the basis of method 2 for COTS software is consistent with EPRI/TR-106439 and NUREG/CR-6421 requirements. Additional tailoring policy based on Code and Standards related to COTS software may be also founded in the suggested commercial software dedication process. Suggested commercial software dedication process has been developed for a commercial I&C software dedicator who performs COTS qualification according to the dedication procedure.

1. 서 론

commercial grade survey 방법에 의한 상용 소프트웨어 인정 프로세스는 첫째, 조사 절차(survey procedure) 확립, 둘째 조사 스케줄 설정, 셋째, 조사 수행(survey performance), 넷째, 조사결과 및 발견(survey results and findings) 순으로 크게 나누어 생각할 수 있다. 즉, 상용 소프트웨어 승인 절차를 확립한 다음, 스케줄 일정을 설정하고 상용 소프트웨어 승인을 위한 일련의 활동을 수행한다. 마지막으로, 상용 소프트웨어 조사를 통하여 발견된 내용과 결과를 종합하여 상용 소프트웨어 평가기준에 부합하는지 여부를 판단한다.
이때 도구들(tools), 기술(techniques), 그리고 방법론(methodologies)을 적용하는데 본 논문은 NUREG/CR-6421을 기본 개념으로 설정하고 EPRI/TR-106439의 현실적 개념을 추가한 다음 현재 개정중인 SRP 7, Branch Technical Position(BTP)중 상용 소프트웨어 관련 draft version 기준을 적용하여 commercial grade survey 방법에 의한 상용 소프트웨어 인정 프로세스를 그림1과 같이 제안하였다.

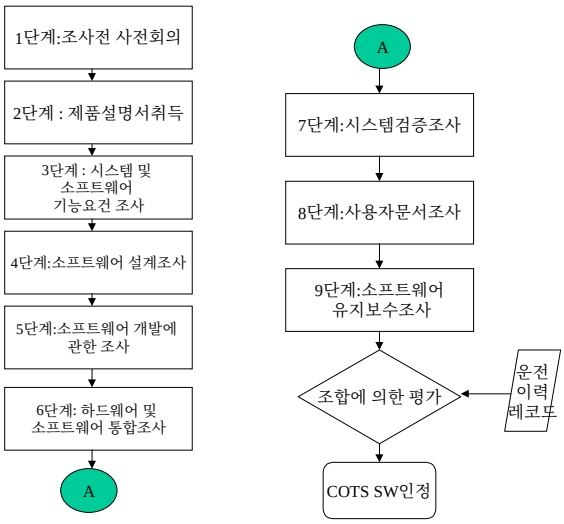

그림 1. 상용 소프트웨어 인정 프로세스

2. 상용 소프트웨어 상세 인정 프로세스

Step 1 : 조사전 사전회의(pre-survey meetings)

상용 소프트웨어 평가팀과 상용 소프트웨어 관련 업체 담당자(이들 중에는 상용 소프트웨어를 개발한 개발자들이 포함되어야 한다.)와 사전에 조사(survey) 방법에 대하여 회의를 개최할 수 있어야 한다. 상용 소프트웨어 평가팀은 정보공학 방법론 따라 제대로된 소프트웨어 엔지니어링 절차를 준수하여 소프트웨어를 개발하였는지 여부를 가장 먼저 논의할 수 있어야 하며 상용 소프트웨어는 일반 산업계 표준을 준수한 만큼 ISO 9001 Part 3 의 준수 여부를 중요한 판단의 기준으로 삼아야 한다. 이때 상용소프트웨어 평가팀은 승인 우선순위(acceptable priority)를 사전에 설정하고 있어야 한다. 조사전 사전회의(pre-survey meetings)에서는 다음과 같은 사항들을 순서적으로 점검하여야 한다.

(1) 계획문서 검토(planning documentation review)
(2) 생명주기 모델 및 단계별 활동사항 검토
(3) ISO 9001 Part 3에 따른 조사
  - 자체 qualification 여부

- 외부조직에 의한 requalification 여부

(4) 우선순위 정의(priority define)

- priority 1 : most critical(system crashes)
- priority 2 : critical(function specification 에 따라 수행되지 않음)
- priority 3 : minor error(functionality에 영향을 주지 않음)
- priority 4 : 기성 functionality에 따른 다른 해석

상기 priority중 priority 1 ~ priority 2는 배포(release) 해서는 안되는 기준으로 설정한다.

Step 2 : 제품설명서 취득

이 단계에서는 회사가 제공할 수 있는 모든 제품설명서를 수집한다. 제품설명서중 배포 보고서(release report)를 점검하여 다음의 자료들을 수집한다.

- 새로운 기능의 추가 및 변경정의
- 개정레벨(revision level), 최소한의 하드웨어 요건 및 개정 이전에 확인된 오류들의 교정을 포함한 소프트웨어 개정이력(software revision history)
- 하드웨어 및 소프트웨어 제품 설명자료 취득
- 새로운 콤포넌트(component)에 대한 소프트웨어 개발 프로세스와의 링크 여부

Step 3 : 시스템 및 소프트웨어 기능 요건 조사

소프트웨어 요구사양명세 조사에서는 상위의 개념인 시스템 명세로부터 소프트웨어 요구사양 명세에 이르기까지 추적성 분석의 개념을 근간으로 다음과 같은 5가지 사항들에 대하여 분석하고 조사한다.

첫째, 시스템 및 소프트웨어 기능요건명세서가 존재하는가?

- 기능설명서(functional design description)
- 요구사양명세서
- 개발환경, CASE 도구들

둘째, 기능요건명세 목록의 취득 및 조사

셋째, 제품에의 기능요건(functional requirement) 반영여부?

- 새로운 제품에 새로운 기능요건사양의 반영여부

넷째, 이외의 요구사양 존재유무?

다섯째, 소프트웨어 요구사양명세의 검토 수행여부

Step 4 : 소프트웨어 설계 조사

설계단계의 주요 점검항목들은 소프트웨어 설계 요건을 작성하였는지를 조사하게 되는데 이때 만약 소프트웨어 설계 요건을 취득할 수 없다면 어떤 설계 정보를 이용할 수 있는지 여부와 이들 설계 결과물에 대한 충분한 검토가 개발과정에서 이루어 졌는지 여부를 평가 한다.

- 소프트웨어 설계 프로세스의 검토
  . 예비 및 상세설계의 검토가 있었는가?
  . 예비 및 상세설계의 검토가 있었다면, 이들 검토에 참여한 사람들은 누구이며 자격 요건을 갖춘 요원이었는지?
- 소프트웨어 설계, 구현, 통합, 테스팅 및 최종사용에 대한 검토

. 공정입력 변수들(ranges, accuracy, sampling intervals, engineering unit conversion)
. 운영체제(operating system), utility, routines 또는 보조프로그램
. postulated abnormal inputs의 처리(handling)
. engineering unit, symbolic names 등의 알고리즘에 의해서 요구되는 데이터 화일 및 데이터 검토

- 범위(range), 정확도 및 갱신구간(accuracy, and update interval)을 포함한 출력물(인간공학 요건 포함)
- 초기화 요건(initialization requirements) 검토
  . 변수 초기값, 기동순서(start-up sequence) 등
- 컴퓨터 시스템에서 탐지된 고장에 반응하는 프로그램 로직의 검토
- 오퍼레이터 인터페이스 검토(keyboard inputs, control panels, display 등)
- 인-서비스 테스트 특성 및 진단의 검토
- 전반적인 컴퓨터 시스템 응답시간을 포함한 타이밍 요건(timing requirement) 검토
- 하드웨어 capability와 일치하는 processing idle time 및 excess memory 검토
- 보안성(security) 요건 검토(password, access control 등)

Step 5 : 소프트웨어 개발에 관한 조사

소프트웨어 개발 절차의 점검에서는 소프트웨어 품질보증 목표와 계획을 세워서 이 계획대로 준수하였는지의 여부를 점검한다. 소프트웨어 프로세스중 개발과 동시에 적절한 검증이 소프트웨어 생명주기 단계별로 이루어 졌는지 여부를 다음과 같이 점검하여야 한다.

- 소프트웨어 품질계획의 검토
  . 산업체 표준 ISO 9000-3 요건의 준수 여부?
- 소프트웨어 개발 문서들에 대한 검토

소프트웨어 개발 절차는 최소한 다음과 같은 사항들이 포함되어 있어야 한다.
소프트웨어 개발 절차에 포함되어 있어야 할 중요 점검 사항들의 내용은 그림 2와 같다.

Step 6 : 하드웨어 및 소프트웨어 통합 조사

하드웨어 및 소프트웨어 통합에서는 하드웨어 및 소프트웨어 통합 계획이 존재했는가, 하드웨어 및 소프트웨어 인터페이스의 적합성을 입증하기 위한 통합시험절차 및 관련 승인기준(acceptance criteria)이 존재했는가, 통합 컴퓨터 시스템에 대한 형상시험(test configuration)을 수행 하였는가, 하드웨어 및 소프트웨어 통합 변경제어(changing control)에 대한 품질보증계획서가 존재했는가, 어떤 프로세스가 시스템 비휘발성 메모리(system non-volatile memory)로 프로그래밍 되었는지를 점검한다.

Step 7 : 시스템 검증 조사

시스템 검증에 대한 시험계획이 존재했는가, 테스트 절차서는 있었는가, 테스트 절차서가 있었다면 적절한 단계로 구성 되었는지와 완결성 여부를 점검한다. 또한, 요건 확인, 기능확인에서 바람직하지 않았던 결과는 없었는지 여부를 재검증한다. 테스트 절차의 이행 여부에서는 테스트될 항목의 기능, 시험 유형의 범위, 테스트할 항목의 선정, 책임사항 및 승인, 테스트 레코드 등의 내용들을 반드시 포함하고 있었는지 여부를 체크한다.
즉, 다음과 같은 항목들을 집중적으로 점검할 수 있어야 한다.

① 일반사항
○ 표준시스템개발프로젝트 모델
② 계획
○ 제품설명, 개발계획 등
③ 기능정의
○ 소프트웨어 시스템 요건 및 기능
④ 시스템 설계
○ 소프트웨어 분석 및 설계 가이드
○ 시스템 설명에 대한 간단한 포맷
⑤ 모듈개발(module development)
○ C/C++ 스타일 가이드
○ C/C++ 이식성 가이드라인
○ 소프트웨어 재사용 가이드라인
⑥ 통합(integration)
○ 시스템사양 관련 문서의 예
⑦ 유형시험(type test)
○ 중요 유형시험 목록
○ 중요 유형시험 설명서
○ 중요 유형시험 기록
○ 중요 검토회의 기록
- 소프트웨어 개발환경 및 소프트웨어 개발도구에 대한 충분한 문서가 존재하는가?
- 조직의 정의 : 개발자와 검토자의 독립성 정도
- 어떤 표준을 이용하는가?
- 어떤 절차서를 이용할 수 있는가? (ex. CM, code attribute : language, modularity, structure, revision history, error handling 등)
- 인-프로세스 감사(in-process audit)의 수행여부?
- 문서화는 잘 되어 있는가?
- V&V 프로세스를 채택하였는가?
- V&V 프로세스의 기록 문서화?
- 코드 검토(code review)를 위해서는 다음과 같은 내용이 포함된 문서를 갖추어야 한다.
- (목 차)
① 계획(planning)
② 책임사항(roles)
③ 개시(initiation)
④ 저작자 준비사항(authors preparations)
⑤ 운영의 기술적 검토(operational walkthrough)
⑥ 소스코드 숙지(reading the code)
⑦ 수집 및 보고서 통계(collecting and reporting statistics)
⑧ 결함기록 회의(defect logging meeting)
⑨ 요약회의(sum-up meeting)
⑩ 점검회의(follow-up meeting)
⑪ 스타일 가이드 및 체크리스트(style guides and checklists)
- 소프트웨어 개발 매뉴얼 검토
- 소프트웨어 개발 계획서를 수립하고 준수하였는가?
- 소프트웨어 설계가 적절히 구현되었는가?
- 품질을 보증하기 위하여 수행한 테스팅 및 절차
- 언제 형상관리를 시작하였는가?
- code walkthrough는 수행하였는가?
. 소스코드 개정이력(source code revision history) 검토

그림 2. 소프트웨어 개발 절차에 포함되어야 할 사항

- 상세 유형시험 절차(detailed type test procedures)
  . 절차서, 입력자료 및 기대치 결과(input data and expected results)?
  . 동적인 조건 뿐만아니라 정적인 조건하에서 테스팅이 수행되었는가?
- survey system validation testing
  . 개발팀과 tester 사이의 조직관계는?
  . big configuration test는 수행하였는지?
  . 테스트 보고서는 이용 가능한가?
  . 테스트 보고서에 대한 평가는 적절하였는가?
  . 테스트 과정에서 모든 오류들을 발견할 수 있었는가?
  . validation 테스팅시 하드웨어 및 코드 정확성 파급효과를 고려하였는가?

Step 8 : 사용자 문서 조사

사용자 문서 즉, 회사(vendor) 측으로 부터 취득할 수 있는 모든 매뉴얼들에 대하여 요구사양 및 설계 요건에 부합되는지 여부를 평가하고 이들이 일관성, 명확성, 정확성을 유지하는 하고 있는지를 평가한다.

Step 9 : 소프트웨어 유지보수 조사

최초의 배포(original release)가 이루어진 이래로 발생된 문제점들을 어떻게 해결하였는가를 점검한 다음 모든 하드웨어 및 소프트웨어 문제점들에 대한 기록정보를 얻는다. 또한, 제품상에서 발견된 오류들이 사용자에게 어떻게 통지되었는지 여부를 점검하여야 하며 사용자들에 의해서 발견된 오류들의 기록 및 추적은 어떠한 절차를 이용하였는가를 반드시 체크할 수 있어야 한다. 최종적으로 엄격한 형상관리 절차에 의하여 형상관리가 이루어 졌는지 여부를 다음과 같이 점검한다.

- 변경통지 절차에 대한 감사(audit) 수행은?
- 소프트웨어가 선적되기전 기능적 감사(functional configuration audit) 및 물리적 감사(physical configuration audit)가 수행되었는가?
- 소프트웨어 소스코드와 실행코드에 대한 접근 및 형상제어(configuration control)를 적절히 유지하고 있는가?

3. 결 론

commercial grade survey를 근간으로 한 상용 소프트웨어 인정 프로세스를 절차적으로 제안하였다. 제안한 프로세스를 실제 적용하기 위해서는 각 회사의 지적재산권 자료의 접근이 필수적이나 근원적으로 지적재산권 자료 접근의 제한 문제에 직면하여 해당 회사의 적극적인 협조 없이는 사실상 상용 소프트웨어 인정프로세스의 적용은 불가능 하다. 또한, NUREG/CR-6421에서 제시하는 Q class에 해당하는 모든 승인기준(acceptance criteria)을 상용 소프트웨어에 그대로 적용하기에는 현실적으로 불가능에 가깝다. 따라서, NUREG/CR-6421의 상용 소프트웨어 기준을 근간으로 EPRI/TR-106439와 같은 현실적인 개념들을 적절한 수준으로 tailoring 하여야만 한다. 본 논문에서 제시한 상용 소프트웨어 인정 프로세스는 NUREG/CR-6421과 EPRI/TR-106439의 규제 요건을 만족시킬 수 있도록 하였지만 상용 소프트웨어 인정 문제는 워낙 까다롭고 어려운 문제이기 때문에 안전성 분야에 적용할 때 최소한의 범위내에서 사용할 것을 권고하며 어느 한 방법을 단독으로 적용하지 말고 본 논문에서 제시한 방법외에 반드시 특별시험 및 감사 등의 다른 인정 방법을 조합하거나 그 이외 또 다른 방법을 혼용하여 평가하여야 한다.

[참고문헌]

[1] IEEE 7-4.3.2-1993, Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations

[2] IEC 880-1986, First Supplement Rev. 3 Aug. 1996, "Software for Computers in the Safety Systems of Nuclear Power Systems."

[3] NUREG/CR-6421-1996, " A Proposed Acceptance Process for Commercial Off-the-Shelf Software in Reactor Applications."

[4] Guideline on Evaluation and Acceptance of Commercial Grade Digital Equipment for Nuclear Safety Applications (EPRI TR-106439)