

정보처리기사 필기

소프트웨어 개발 9 인터페이스 구현 ①

양문자 선생님

출처 : ncs 학습모듈(NCS능력단위 인터페이스구현)

소프트웨어 개발

차례

1 데이터 입출력 구현

2 통합 구현

3 제품소프트웨어 패키징

4 애플리케이션 테스트 관리

5 인터페이스 구현

1) 인터페이스 설계 확인

2) 인터페이스 기능 구현

3) 인터페이스 구현 검증

인터페이스 설계 확인

인터페이스 기능 확인

- 인터페이스 기능 : 이 기종 시스템 또는 컴포넌트 간 데이터 교환 및 처리를 위한 목적의 기능
- 인터페이스 기능 확인 : 인터페이스 설계서, 인터페이스 정의서
- 인터페이스 설계서
 - 각 시스템의 교환 데이터 및 업무, 송수신 주체 등이 정의되어 있다.
 - 일반적인 내용이 포함된 인터페이스 설계서(정의서) 외에도 다양한 다이어그램 및 데이터 포맷을 포함한 형태의 인터페이스 설계서가 있다.

소프트웨어 개발

- 인터페이스 설계서(정의서)
시스템의 인터페이스 현황을 한눈에 확인하기 위하여 한 시스템이 갖는 인터페이스 목록 및 각 인터페이스의 상세 데이터 명세와 각 기능의 세부 인터페이스 정보를 정의한 문서

(1) 시스템 인터페이스 정의서

(가) 인터페이스 목록

인터페이스 번호 및 인터페이스 되는 시스템의 정보 및 관련 요구 사항 ID(요구 사항 정의서의 요구 ID)를 리스트(목록) 형태로 보여 준다.

(나) 인터페이스 명세

인터페이스 목록에 있는 각 인터페이스의 상세 정보. 각 인터페이스 번호당 인터페이스 되는 데이터, 데이터 형식, 송수신 시스템의 정보 등을 구체화 한다.

소프트웨어 개발

(2) 상세 기능별 인터페이스 정의서

인터페이스를 통한 각 세부 기능의 개요, 세부 기능이 동작하기 전에 필요한 사전 조건, 사후 조건 및 인터페이스 파라미터(데이터), 호출 이후 결과를 확인하기 위한 반환값 등을 정의한 문서이다.

인터페이스 ID	IF-HR-001	인터페이스명	인사정보 전송 인터페이스
오퍼레이션명	인터페이스 대상 선정		
오퍼레이션 개요	관계사와 인터페이스 할 대상(정보)를 선택		
사전조건	과장이상 정규직만 선택		
사후조건	전송 이후 상대시스템의 결과값을 업데이트		
파라미터	발령구분(입사, 이동, 겸직, 승진), 발령정보(발령일, 직급, 소속)		
반환값	Success / Fail		

소프트웨어 개발

- 정적·동적 모형, 데이터 포맷 형태에 따른 인터페이스 설계서

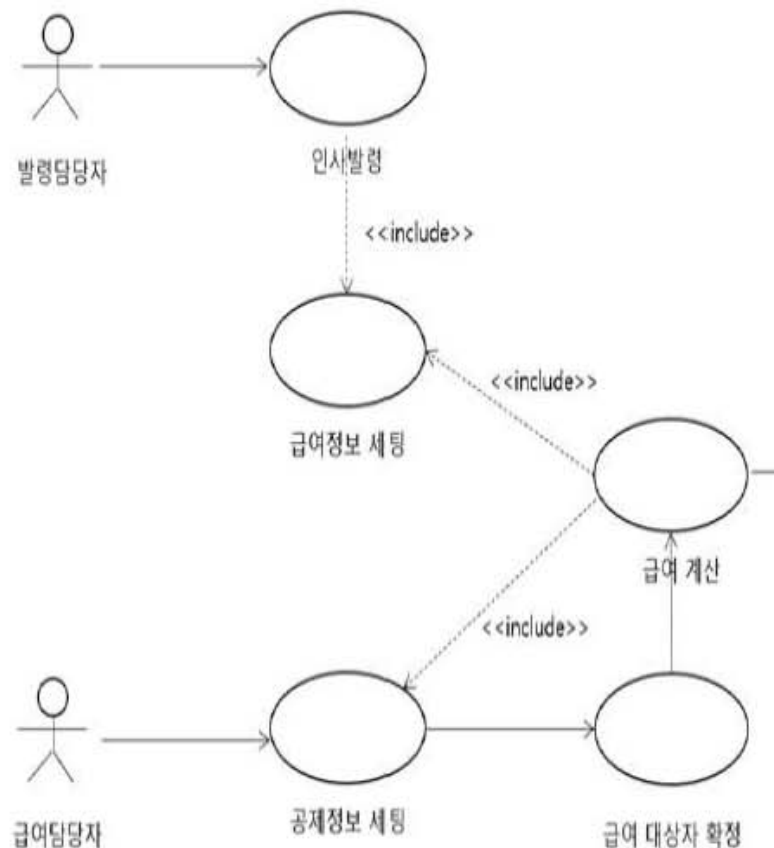
다이아그램을 통해 시스템, 컴포넌트별 인터페이스와 요구 조건을 확인할 수 있다.

(1) 정적, 동적 모형을 통한 인터페이스 설계서

시스템을 구성하는 주요 구성 요소 간 트랜잭션을 보여 주고, 어떤 트랜잭션이 인터페이스를 통해 상호 교환되는지 확인할 수 있다.

(2) 데이터 정의를 통한 인터페이스 설계서

제공하는 인터페이스 서비스에 대한 상세 명세를 표현하는 산출물. 개발 수준에서 인터페이스를 어떻게 구현해야 할지 잘 명시되어 있다.



인터페이스 표준 확인

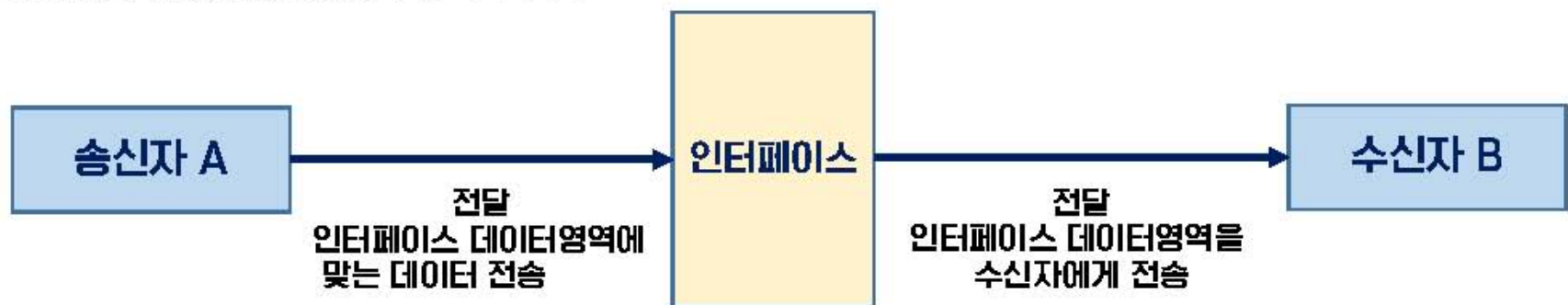
외부 및 내부 모듈 간 데이터를 교환하고 상호 호환이 되게 하기 위해서 인터페이스 데이터 표준을 정의하고 이를 관리하여야 한다.

- 인터페이스 데이터 표준 개념
 - 인터페이스를 위해 인터페이스가 되어야 할 범위의 데이터들의 형식과 표준을 정의하는 것이다.
 - 기존에 있던 데이터 중 공통의 영역을 추출하여 정의 하는 경우도 있고
 - 인터페이스를 위해 한쪽의 데이터를 변환하는 경우도 있다.

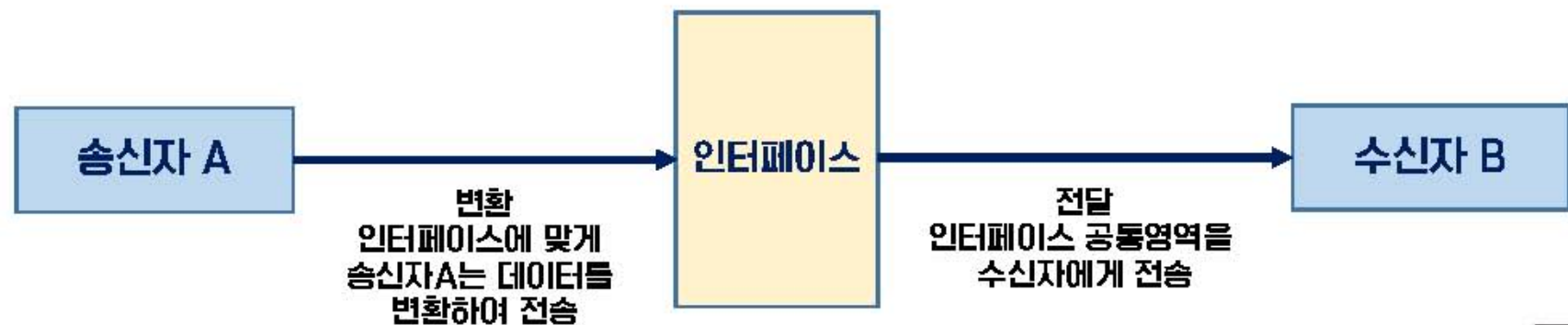
소프트웨어 개발

[표준 데이터 정의에 따른 인터페이스 데이터 변환 방식]

- 인터페이스 데이터 형태가 동일할 경우



- 인터페이스 데이터 형태가 동일하지 않을 경우



소프트웨어 개발

- 외부 및 내부 모듈 간 인터페이스 데이터 표준 확인하기
- 1. **식별된 데이터 인터페이스**를 통해 인터페이스 데이터 표준을 확인한다.
식별된 데이터 인터페이스 및 각 인터페이스의 입력값, 출력값의 의미를 파악하고 이에 따른 데이터 표준을 작성한다.
 - (1) 데이터 인터페이스의 의미를 파악한다.
데이터 표준 작성을 위해 식별된 데이터 인터페이스의 입력값, 출력값이 의미하는 내용을 파악한다. 각 출력값이 의미하는 바와 데이터의 특성 등 참고할 만한 사항을 구체적으로 작성한다.

소프트웨어 개발

데이터 인터페이스 입·출력값의 의미 작성 예시

구분	파라미터(변수)	의미
입력값	급여 코드	- 급여 작업의 단위를 나타내는 Key 값 예시) 2020년 6월 정규직 급여: 202006R001
	급여 일자	- 급여를 지급하는 일자(인터페이스 시 회계 일자로 활용)
	급여 계산 결과	- 각 직원별 급여 계산 결과 - 총금액(급여, 상여 구분), 공제 금액(소득세, 주민세, 국민연금, 건강 보험, 고용 보험, 사내 공제 금액 구분), 직원 계좌 정보
출력정보	전표 일반 정보	- 발생 시기, 전표의 성격(매입/매출 등), 세금 신고 여부, 보조 정보를 통한 전표의 설명
	전표 금액	- 각 부서별 지급, 공제 항목에 대한 금액 - 개인별 지급 금액
	거래처 정보	- 각 개인별 지급을 위한 거래처(직원) 및 계좌 정보

소프트웨어 개발

(2) 데이터 인터페이스의 의미 파악을 통한 데이터 표준을 확인한다.

데이터 인터페이스의 각 항목이 의미하는 바가 무엇인지 이해하고 확인한 후에 이를 기반으로 데이터 표준을 확인한다.

- 데이터 인터페이스 입·출력값의 의미 데이터 표준 작성 예시

구분	파라미터(변수)	의미
입력값	급여 코드	- 급여 지급 연월을 숫자 6자리(예: 202006)로 명시 - 정규직은 R, 계약직은 T(예: 202006R) - 3자리 숫자로 급여 코드 증가 시 숫자를 증가한다. 해당월에 처음 부여된 코드는 1 번을 부여(예: 202006R001)
	급여 일자	- YYYYMMDD 형태의 8자리로 표현(예: 20200605)
	급여 계산 결과	- 각 직원별 급여 계산 결과 항목: 사번, 근무 일수, 소속, 직급, 급여, 상여, 비과세 급여, 총지급 금액, 소득세, 주민세, 국민연금, 건강 보험, 고용 보험, 회사 공제 금액, 총공제 금액, 실지금액 * 데이터 포맷은 엑셀로 차후 정의

소프트웨어 개발

구분	파라미터(변수)	의미
출력정보	전표 일반 정보	<ul style="list-style-type: none"> - 발생 시기: YYYYMMDD 형식 - 전표 구분: 매입 AP 매출 AR - 세금 신고 여부: TEX_YN - 보조 정보를 통한 전표의 설명: CHAR 형태 500자 이내
	전표 금액	<ul style="list-style-type: none"> - 각 부서별 지급, 공제 항목에 대한 금액: unsigned int(-, 소수점 자리 없음) - 개인별 지급 금액: unsigned int
	거래처 정보	<ul style="list-style-type: none"> - 거래처(직원): 사번_이름 형태로 정의 - 계좌 정보: 은행 코드_계좌 번호로 정의

2. **식별된 인터페이스 기능**을 통해 인터페이스 데이터 표준을 확인한다.
: 추가 보완할 부분이 없는지 확인 및 검증한다.

3. 데이터 인터페이스 및 식별된 인터페이스 기능을 통해 데이터 표준을 최종적으로 확인한다.

소프트웨어 개발

- 외부 및 내부 모듈 연계를 위한 인터페이스 기능 식별

내부, 외부 모듈 연계 방법(EAI, ESB 연계 방법)

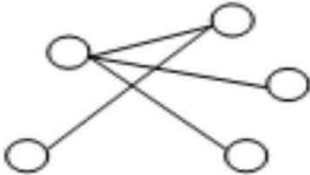
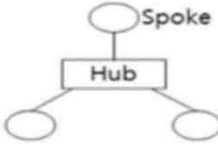
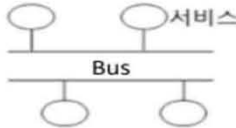
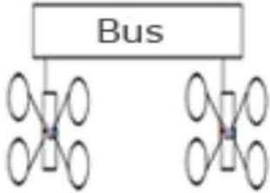
1. EAI(Enterprise Application Integration)

(1) EAI의 개념

- 기업에서 운영되는 서로 다른 플랫폼 및 애플리케이션들 간의 정보 전달, 연계, 통합을 가능하게 해 주는 솔루션이다.
- EAI를 사용함으로써 각 비즈니스 간 통합 및 연계성을 증대시켜 효율성을 높여 줄 수 있으며 각 시스템 간의 확정성을 높여 줄 수 있다.

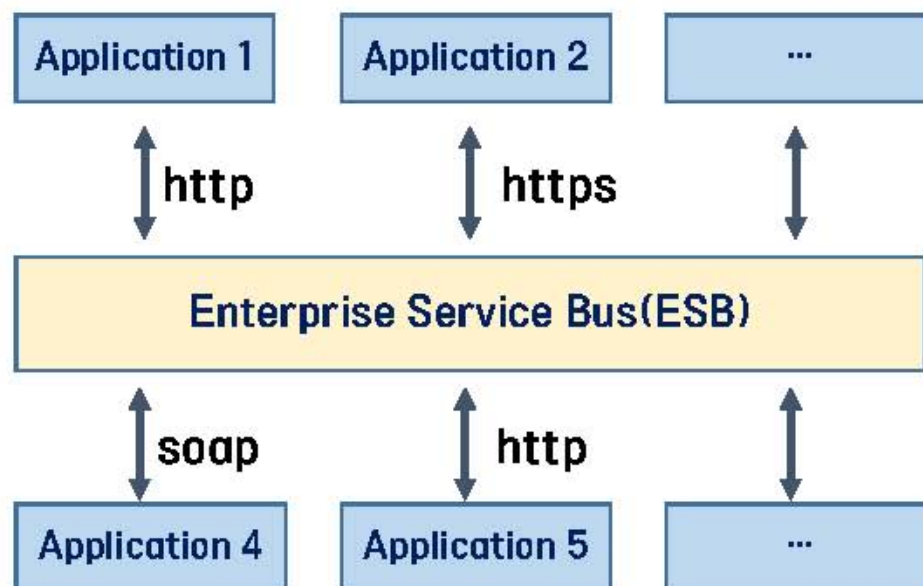
소프트웨어 개발

(2) EAI의 구축 유형

유형	개념도	설명	특징
Point-to-Point		- 중간에 미들웨어를 두지 않고 각 애플리케이션간 Point to Point 형태로 연결	- 솔루션 구매 없이 통합 - 상대적 저렴하게 통합 가능 - 변경, 재사용 어려움
Hub & Spoke		- 단일 접점이 허브 시스템을 통 해 데이터를 전송 하는 중앙 집 중적 방식	- 모든 데이터 전송 보장 - 확장, 유지 보수 용이 - 허브 장애 시 전체 영향
Message Bus		- 애플리케이션 사이 미들웨어 (버스)를 두어 처리 - 미들웨어 통한 통합	- 어댑터가 각 시스템과 버스를 두어 연결하므로 뛰어난 확장성, 대용량 처리 가능
Hybrid		- 그룹 내에는 Hub & Spoke 방식, 그룹 간에서는 메시징 버 스 방식을 사용	- 표준 통합 기술, 데이터 병목 현상 최소화

2. ESB(Enterprise Service Bus)

- ESB는 애플리케이션 간의 통합 측면에서 EAI와 유사하다고 볼 수 있으나 애플리케이션 보다는 서비스 중심으로 통합을 지향하는 아키텍처 또는 기술이다.
- 웹 서비스 중심으로 표준화된 데이터를 버스를 통해, 이 기종 애플리케이션의 결합도를 약하게 (loosely-coupled) 통합하는 핵심 플랫폼(기술)이다.



문제풀이

- EAI(Enterprise Application Integration) 구축유형 중 Hybrid에 대한 설명으로 틀린 것은?
 - ① Hub & Spoke와 Message Bus의 혼합방식이다.
 - ② 필요한 경우 한 가지 방식으로 EAI구현이 가능하다.
 - ③ 데이터 병목현상을 최소화할 수 있다.
 - ④ 중간에 미들웨어를 두지 않고 각 애플리케이션을 point to point로 연결한다.

(2020년 4회 정보처리기사 필기 기출문제 소프트웨어 개발)

문제풀이

- EAI(Enterprise Application Integration)의 구축 유형으로 옳지 않은 것은?
 - ① Point-to-Point
 - ② Hub&Spoke
 - ③ Message Bus
 - ④ Tree

(2020년 1, 2회 정보처리기사 필기 기출문제 소프트웨어 개발)

소프트웨어 개발

- 인터페이스 구현을 위한 도구

여러 가지 방법으로 인터페이스는 구현될 수 있지만 대표적인 방법으로는 **데이터 통신**을 이용한 인터페이스 구현 방법과 **인터페이스 테이블**을 이용한 인터페이스 구현 방법으로 나눌 수 있다.

1. 데이터 통신을 통한 인터페이스 구현

애플리케이션 영역에서 인터페이스 형식에 맞춘 데이터 포맷을 인터페이스 대상으로 전송하고 이를 수신 측에서 파싱(Parsing)하여 해석하는 방식이다. 주로 JSON 및 XML 형식의 데이터 포맷을 사용하여 인터페이스를 구현한다.

(1) JSON(JavaScript Object Notation)

JSON은 속성-값 쌍(attribute-value pairs)으로 이루어진 데이터 오브젝트를 전달하기 위해 사용하는 개방형 표준 포맷이다. AJAX(Asynchronous JavaScript and XML)에서 많이 사용되고 XML을 대체하는 주요 데이터 포맷이다. 언어 독립형 데이터 포맷으로 다양한 프로그래밍 언어에서 사용되고 있다.

```
{  
  "firstName": "Kildong",  
  "lastName": "Hong",  
  "email": "hong1004@hackers.com"  
}
```

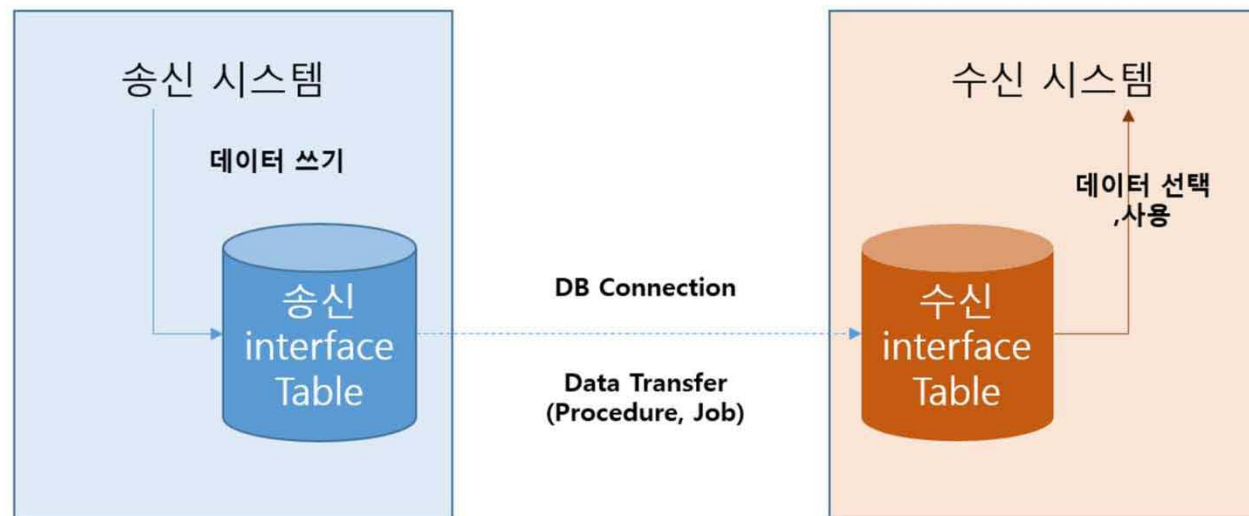
(2) XML(eXtensible Markup Language)

다른 많은 종류의 데이터를 기술하는 데 사용될 수 있으며, 다른 종류의 시스템끼리 데이터를 쉽게 주고 받을 수 있게 하는 목적이 있다.

```
<p><strong>이 구문은 잘못 되었습니다.</p></strong>  
<p><strong>이 구문이 정확합니다.</strong></p>
```

2. 인터페이스 엔티티를 통한 인터페이스 구현

인터페이스가 필요한 시스템 사이에 별도의 인터페이스 엔티티를 두어서 상호 연계한다.
엔티티의 역할은 데이터베이스에서 인터페이스 테이블을 두어 각 시스템 간 데이터 교환에 활용하는 방법이 업계에서 많이 사용된다



문제풀이

- 인터페이스 구현 시 사용하는 기술 중 다음 내용이 설명하는 것은?

JavaScript를 사용한 비동기 통신기술로
클라이언트와 서버간에 XML데이터를 주고 받는 기술

- | | |
|-------------|-----------|
| ① Procedure | ② Trigger |
| ③ Greedy | ④ AJAX |

(2020년 3회 정보처리기사 필기 기출문제 소프트웨어 개발)