



■ WIPO 서열 검증기 웹 서비스

웨비나 교육

오늘 다룰 내용

- 검증기 서비스의 목적
- 데스크탑 도구와 비교
- 웹 서비스란 무엇인가?
- 검증기 엔드포인트 소개 (주 기능)
- 웹 서비스의 운영 및 설치
- 요청/응답 매개변수
- 웹 서비스의 구성

WIPO 서열 검증기: 목적

- **목표:** 제출한 서열 목록이 WIPO ST.26을 준수하는지 확인
- **구현:** 제출한 서열 목록을 특허청이 검증하기 위해 생성된 웹 서비스
- **특허청:** 각 특허청 IT 환경에 통합
- **출원인:** 오류/경고를 표시하는 동일한 보고서를 수신

WIPO 서열 유효성 검증기 대 WIPO 서열 데스크탑

검증기 서비스	데스크탑 도구
특허청을 위한 설계	출원인을 위한 설계
웹 서비스	데스크탑 애플리케이션
<u>인터페이스 없음</u>	인터페이스
IT 환경에 통합	사용자에 의한 실행
두 가지 검증 수준	상세 검증만
둘 이상의 DTD	최신 DTD만

웹 서비스란 무엇인가?

- 인터넷 또는 인트라넷을 통해 통신하는 두 전자 장치 사이에서 작동하는 서비스
- 표준 XML/JSON 요청/응답 형식을 사용합니다
- 특정 프로그래밍 언어에서 구현될 필요가 없습니다(언어 애그노스틱)
- 모듈식, 동적 및 분산
- SOAP 또는 REST를 통신 수단으로 사용하여 작동하는 API 유형
- 웹 서비스 계약은 "무엇"/ "어떻게" 및 "어디"를 정의합니다

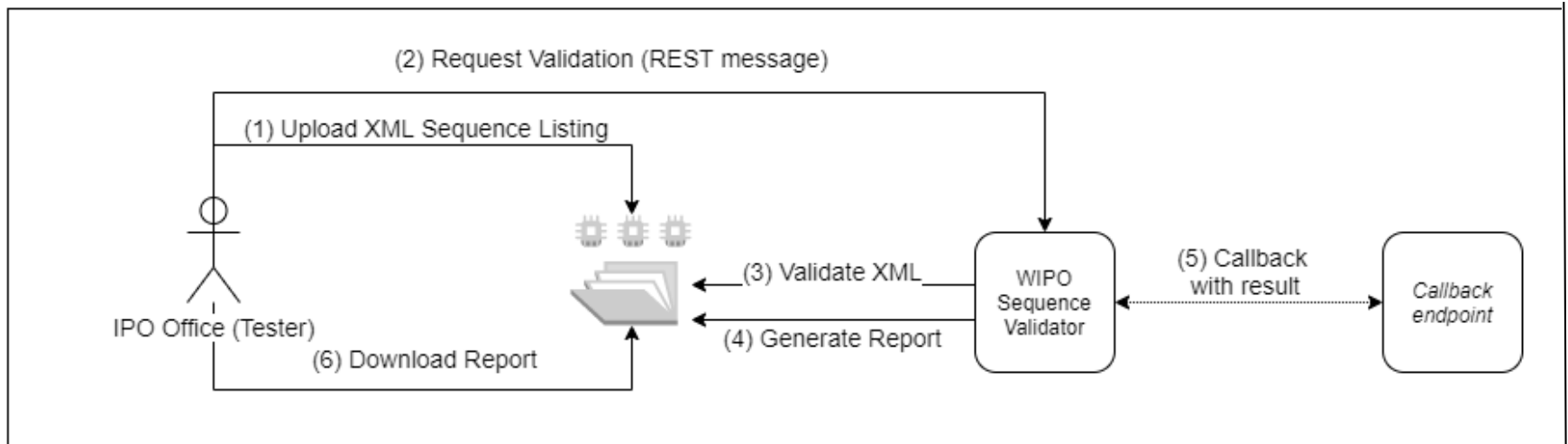
OAS API 사양

- RESTful API에 대한 사양
- REST는 HTTP의 서브세트를 사용하는 아키텍처인 REpresentational State Transfer를 의미합니다
- 이 사양을 사용하면 사람과 컴퓨터 모두 소스 코드에 액세스하지 않고도 API의 기능을 이해할 수 있습니다
- 프로그래밍 언어 애그노스틱

WIPO 서열 검증기: 기본사항 (1)

- 다음과 같은 기능:
 - ST.26 서열 목록 검증
 - 검증 상태 제공
 - 검증 보고서의 메시지로 엔드포인트를 콜백
- WIPO ST.90 준수 ("참조" 섹션을 참조)
- 웹 API 사양: OAS 2.0 ("참조" 섹션을 참조)
- 외부에 노출되도록 설계되지 않았습니다

WIPO 서열 검증기: 기본사항 (2)



검증기의 설치

- Java 8 설치 필요 (향후 Java 11 지원)
- 폴더 구조는 웹 서비스가 있는 위치의 상위 위치에 설정되어야 합니다
- 고유한 application.properties 파일을 제공하여 근본적인 기본 구성을 조정할 수 있습니다
- WIPO 서열 검증기 작동 메뉴얼에서 사용할 수 있는 전체 지침:
https://www.wipo.int/export/sites/www/standards/en/sequence/wipo_sequence_validator_operations_manual.pdf

기본 폴더 구조

temp/st26

temp/st26/inbox

temp/st26/outbox

temp/st26/params

temp/st26/process

temp/st26/reports

웹 서비스 배포

- 두 가지 유형의 서비스: WAR 및 JAR이 제공됩니다
 - JAR – 서버에 내장
 - WAR – 특허청은 Tomcat 서버 또는 설치된 Spring Boot 2 및 Servlet Spec 3.1+와 호환되는 애플리케이션 서버가 필요합니다
- `http://[host-name]:8080/swagger-ui.html`: Swagger UI를 사용하여 기본적인 매개변수를 제공하고/테스트를 수행할 수 있습니다
- 콜백 엔드포인트가 도구의 일부를 형성하지 않기 때문에 특허청은 콜백 엔드포인트를 제공해야 합니다

서비스를 JAR로 배포

■ **java -D"file.encoding=UTF-8" -jar wipo-sequence-validator.jar**

■ 기본 포트는 8080이지만 다음을 사용하여 서버 포트를 변경할 수 있습니다:

java -D"file.encoding=UTF-8" -jar wipo-sequence-validator.jar --

```

Command Prompt - java -D"file.encoding=UTF-8" -jar wipo-sequence-validator.jar
5 Dir(s) 121,863,659,520 bytes free

C:\dev>java -D"file.encoding=UTF-8" -jar wipo-sequence-validator.jar
#####
WIPoSequenceValidator
#####
Application Version: 1.1.0-beta4-SNAPSHOT

=====
2021-04-14 09:43:14 [main] INFO c.w.st26.ipotool.IPOToolApplication - Starting IPOToolApplication v1.1.0-beta4-SNAPSHOT on LAP536593 with PID 9988 (C:\dev\wipo-sequence-validator.jar started by DevFrancis in C:\dev)
2021-04-14 09:43:14 [main] INFO c.w.st26.ipotool.IPOToolApplication - No active profile set, falling back to default profiles: default
2021-04-14 09:43:14 [main] INFO o.s.b.w.s.c.AnnotationConfigServletWebServerApplicationContext - Refreshing org.springframework.boot.web.servlet.context.AnnotationConfigServletWebServerApplicationContext@de0a01f: startup date [Wed Apr 14 09:43:14 CEST 2021]; root of context hierarchy
2021-04-14 09:43:16 [main] INFO o.s.c.s.PostProcessorRegistrationDelegate$BeanPostProcessorChecker - Bean 'asyncConfiguration' of type [com.wipo.st26.ipotool.services.async.AsyncConfiguration$$EnhancerBySpringGLIB$$c683b031] is not eligible for getting processed by all BeanPostProcessors (for example: not eligible for
  
```

서비스를 WAR로 배포

- Tomcat 8.5의 경우:
 - 서버 중지: "\$TOMCAT_ROOT\bin\catalina.bat stop"
 - WAR를 "\$TOMCAT_ROOT\webapps\wipo-sequence-validator.war"에 복사
 - 서버 시작: "\$TOMCAT_ROOT\bin\catalina.bat start"
- "\$TOMCAT_ROOT"는 Tomcat 애플리케이션 서버의 루트 폴더를 의미합니다
- 애플리케이션 서버를 시작할 때, 파일 인코딩을 UTF-8로 설정해야 합니다
- 기본 포트는 다시 8080입니다. 변경하기 위해, Tomcat 구성 파일의 "포트"속성을 설정해야합니다. 참조
[:https://tomcat.apache.org/tomcat-8.5-doc/config/http.html#Common_Attributes](https://tomcat.apache.org/tomcat-8.5-doc/config/http.html#Common_Attributes)

검증 기능

- **설명:** 받은 문서 폴더에 있는 기존 ST.26 파일의 검증을 요청합니다. 요청 시 검증 요청의 상태를 검색하기 위한 고유한 verificationID를 반환합니다
- **엔드포인트 (JAR):** `http://[host-name]:8080/api/[version]/validate`
- **엔드포인트 (WAR):** `http://[host-name]:8080/wipo-sequence-validator/api/[version]/validate`
- **생성:** application/json
- **입력:** nameFile 및 검증 유형
- **출력:** verificationID 및 검증 보고서

"상세 검증" 대 "형식 검증"

- 검증기가 수행할 수 있는 검증에는 두 가지 유형이 있습니다: *상세 검증* 및 *형식 검증*
 - **형식 검증:** XML 파일이 유효하고 ST.26 DTD (심각도: XML_WARN | XML_ERROR)를 준수합니까?
 - **상세 검증:** 형식 검증 및 WIPO ST.26(심각도: 오류 | 경고)에서 파생된 준수 검증 규칙
- 형식 검증은 제출 시 사용자에게 동시에 제공될 수 있습니다. 상세 검증은 비동시적으로 수행되어야 합니다.

검증 후

- 서열 목록 파일이 로컬 "보낸 문서" 폴더로 이동됩니다
- 검증 보고서는 로컬 "보고서" 폴더에 생성됩니다
- 로컬 "프로세스" 폴더의 임시 폴더들이 삭제됩니다
- 특정 검증 프로세스를 식별하는 검증 ID가 있으며 생성된 폴더에는 이 ID가 표시됩니다
- 검증 보고서가 포함된 응답을 제공하기 위해 엔드포인트를 선택적으로 콜백(특허청에 의해 구성되는 경우)

Swagger UI (데모)

The screenshot shows a web browser window displaying the Swagger UI for the WIPO Sequence Validator API. The browser's address bar shows the URL `localhost:8080/swagger-ui.html`. The Swagger UI header is green and contains the Swagger logo, the text "swagger", and a dropdown menu labeled "Select a spec" with "api-infos" selected. The main content area features the title "WIPO Sequence Validator API" with a version indicator "0.1". Below the title, it shows the base URL `localhost:8080/` and a link to the API documentation. The description reads "API for the WIPO Sequence Validator". A section for "validation-controller" (Validation Controller) is visible, along with a "Models" section, both with right-pointing chevron icons indicating expandable content.

[호스트 이름]: 8080/sw902 -ui.html

(표시) 상태 기능

- **설명:** 특정 ST.26 파일에 대한 검증 상태를 요청
- **엔드포인트 (JAR):** [http://\[호스트 이\]:8080/api/\[version\]/status](http://[호스트 이]:8080/api/[version]/status)
- **엔드포인트 (WAR):** [http://\[host-name\]:8080/wipo-sequence-validator/api/\[version\]/status](http://[host-name]:8080/wipo-sequence-validator/api/[version]/status)
- **생성:** application/json
- **입력:** verificationID
- **출력:** status
(RUNNING/FINISHED_VALID/FINISHED_INVALID/NOT_FOUND/VERIFICATION_ID_ERROR)

콜백 엔드포인트

- 콜백 엔드포인트: 웹 서비스로부터 정보를 수신하는 별도의 API
- 웹 서비스를 호출하는 콜백 엔드포인트에 의해 검증이 수행될 수 있습니다
- 비동시적으로 수행될 것입니다
- 조작 매뉴얼에 제공된 웹 서비스 계약을 준수해야 합니다
- 요청 및 응답 형식도 정의된 표준 형식으로 제공되어야 합니다

요청 형식

■ 검증 기능

```
{  
  "currentApplicationNumber": "string",  
  "currentSEQLVersionNumber": "string",  
  "parentApplicationNumber": "string",  
  "parentSEQLVersionNumber": "string",  
  "seqInputLocation": "string",  
  "verificationReportOutputPath": "string",  
  "nameFile": "file.xml",  
  "type": "full or formality"  
}
```

■ 상태 기능

```
{  
  "verificationID": "1552208288697FNc2"  
}
```

제공된 응답 (1)

- **processID**: WIPO 서열 검증기가 이미 제공했습니다
- **seqType**: 고정 값: 'ST.26 '
- **httpStatus**: 서비스의 HTTP 상태 코드 서비스: '성공' / '실패'와 HTTP 오류 코드 및 해당 설명
- **applicationNumber**: 특허 출원의 경우 서열 목록은 예컨대, PCTUS1234567 또는 23456789의 일부입니다.
- **currentSQLVersionNumber**: 이 서열 목록의 버전 번호 (특허청이 내부적으로 할당, 예컨대 1.1).
- **parentApplicationNumber**: 연관된 모출원, 예컨대 12345678.
- **currentSQLVersionNumber**: 모출원의 서열 목록의 수정 번호 (오피스가 내부적으로 할당, 예컨대 3.2).
- **verificationReportOutputPath**: 보고서에 대한 전체 경로를 제공합니다
- **startTime**: 서비스가 검증을 시작한 시간, 예컨대, 2019-07-29 15:59:37.784
- **endTime**: 서비스가 검증을 완료한 시간, 예컨대, 2019-07-29 15:59:37.859
- **elapsedTime**: 서열 목록의 검증에 필요한 총 시간을 제공, 예컨대 0시간 0분 0초 75ms

...

제공된 응답 (2)

- **totalWarningQuantity**: 검증 보고서의 모든 경고의 합, 예컨대 2.
- **totalErrorQuantity**: 검증 보고서의 모든 오류의 합, 예컨대, 0.
- **seqInputQuantity**: 출원인이 입력한 프로젝트에 있는 일반 정보 섹션의 총 서열 수, 예컨대 10.
- **seqIDQuantity**: 출원인이 입력한 숫자, 예컨대, 10에 대해 서열 목록의 서열들의 수를 체크하기 위한 검증 단계.
- **errorSummary** 이하의 세부 정보를 포함하는 검증 보고서에 보고된 오류의 요약:

```
[
{errorIndexID (이는 보고된 각 오류 코드의 순차적인 번호, 예컨대, 1, 2, 3 등
입니다):  errorCode:  (오류에 대한 숫자 또는 알파숫자 식별, 예컨대, 오류의 경우
1(####) 또는 경고의 경우 2(####))  설명 (오류 코드-설명 쌍):,
sequenceNumber (SEQ ID),  XPath:  (이는 XML 문서, 예컨대,
ST26SequenceListing/SequenceData[1]/INSDSeq/INSDSeq_sequence의 요소 및
속성을 탐색하는 데 사용될 수 있음),  characterRange:  (예컨대, 10-20: 요소
INSDSeq_sequence에서 문자 위치 10에서 시작하여 위치 20까지)
},  {{(다음 오류에 대해 보고된 동일한 정보) },  {. ... ..}
```

검증 보고서

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<verificationReport productionDate="YYYY-MM-DD" sourceFileName="[ST.26 filename]">
  <verificationMessages>
    <message>
      <severity>[ERROR | WARN | XML_WARN | XML_ERROR]</severity>
      <dataElement>[ST.26 element]</dataElement>
      <detectedSequence>[Sequence ID]</detectedSequence>
      <detectedValue>[value]</detectedValue>
      <messageKey>[Message key]</messageKey>
      <params>
        | <param key="param key">Param value</param>
      </params>
      <localizedMessage> [Localized message] </localizedMessage>
    </message>
    ...
  </verificationMessages>
</verificationReport>

```

검증 보고서: 예시

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<VerificationReport productionDate="2021-05-10" sourceFileName="EPOShort.xml">
  <VerificationMessageBag>
    <VerificationMessage>
      <Severity>WARNING</Severity>
      <DataElement>PROPERTY_NAMES.FEATURE_QUALS</DataElement>
      <DetectedSequence>2</DetectedSequence>
      <DetectedValue>2'-O-methyladenosine</DetectedValue>
      <MessageKey>X_FEATURE_QUALS_QUAL_NOTE_VAL_OTHER</MessageKey>
      <ParameterBag/>
      <LocalizedMessage>Verify that the 'note' qualifier describes the 'OTHER' value for the 'mod_base' qualifier.</LocalizedMessage>
    </VerificationMessage>
    <VerificationMessage>
      <Severity>WARNING</Severity>
      <DataElement>PROPERTY_NAMES.FEATURE_QUALS</DataElement>
      <DetectedSequence>6</DetectedSequence>
      <DetectedValue>2'-O-methyladenosine</DetectedValue>
      <MessageKey>X_FEATURE_QUALS_QUAL_NOTE_VAL_OTHER</MessageKey>
      <ParameterBag/>
      <LocalizedMessage>Verify that the 'note' qualifier describes the 'OTHER' value for the 'mod_base' qualifier.</LocalizedMessage>
    </VerificationMessage>
    <VerificationMessage>
      <Severity>ERROR</Severity>
      <DataElement>PROPERTY_NAMES.APPLICANT</DataElement>
      <DetectedSequence></DetectedSequence>
      <DetectedValue>{"name":"DAIICHI SANKYO COMPANY, LTD.", "languageCode":""}</DetectedValue>
      <MessageKey>LANGUAGE_CODE_MISSING</MessageKey>
      <ParameterBag/>
      <LocalizedMessage>The language code has not been entered.</LocalizedMessage>
    </VerificationMessage>
  </VerificationMessageBag>
</VerificationReport>

```


매개변수 구성-기본값

- 나머지 폴더에서 사용할 기본적인 경로=**.../temp/st26/**
- 처리할 파일을 넣을 폴더=**기본적인 경로+/받은 문서/**
- 검증된 ST26 파일을 저장할 폴더=**기본적인 경로+/보낸 문서/**
- 검증 보고서를 저장할 폴더=**기본적인 경로+/보고서/**
- 상세 및 형식 폴더의 상위 폴더=**기본적인 경로+/프로세스/**
- 매개변수를 저장할 폴더=**기본적인 경로+/params/**
- **AlternativeResourceBasePath=기본적인 경로+/alt_resources**
- 검증 보고서의 로컬화된 메시지에 사용되는 로컬화=**en**
- 검증 결과를 알리는 데 사용될 콜백 엔드포인트의 URL=**http://callbackservice/api/endpoint**
- 또한 로깅 등

대체 구성 (1)

- 새로운 application.properties 파일 제공을 통해 구성이 가능합니다
- 대체 application.properties 파일이 제공되면, 도구는 다음 순서로 대체 구성을 검색합니다:
 - 현재 디렉토리 내의 "/config" 폴더
 - 현재 디렉토리
 - \$classpath 또는 config 패키지; 이어서
 - \$classpath 루트 또는
 - 명령 라인의 매개변수를 사용하여 지정된 위치에서
- **참고 : 새 구성을 적용하기 전에 검증기를 다시 시작해야 합니다**

대체 구성 (2)

- 특히, 다음 두 가지 사용자 설정을 수행할 수 있습니다:

- 검증 보고서의 메시지 언어 – 언어를 스페인어로 변경하기 위해 `application.properties` 파일의 `"validator_locale"` 매개 변수는 적절한 언어 코드, 예컨대, `"validator_locale=es"`로 설정되어야 합니다.
- 사용자 정의 유기체 이름: 다음과 같이 패키지 목록의 일부를 형성하지 않는 유기체 이름의 단일 JSON 파일을 제공합니다:

```
[
  {"value": "Custom Organism Sample"},
  {"value": "Custom Organism Sample 2"}
]
```

- 새로운 `custom_organism.json` 파일은 `alternativeResourceBasePath`로 표시되는 위치에 있어야 합니다.

대체 구성: DTD (1)

- 기본적으로, Validator 서비스는 `"/ src / main / resources"` (현재 버전 1.3)에서 제공되는 최신 DTD에 대해 유효성을 검사합니다
- 두 가지 다른 방법을 사용하여 이전 DTD에 대해 유효성을 또한 검사할 수 있습니다:
 - 첫째:
 - JAR 파일을 압축 해제하고 추가 또는 대체 ST.26 DTD 파일에 대한 참조를 `src/main/resources` 폴더에 포함시키십시오;
 - `catalog.xml`을 수정합니다 추가 ST.26 DTD에 대한 새 항목을 추가하거나 기존의 것을 편집하여 파일을 작성합니다

대체 구성: DTD (2)

- 두 번째, JAR 파일을 수정하는 대신:
 - "catalog.xml" 및 모든 DTD를 로컬 폴더에 복사합니다;
 - 추가 ST.26 DTD에 대한 참조를 포함시키기 위해 "catalog.xml"을 수정합니다; 그리고
 - 시작할 때 이 Java 시스템 속성을 설정합니다:
"xml.catalog.files=<path_to_catalog.xml>"

- **참고: 이는 표준 본문에서 파생된 비즈니스 검증 규칙에 필요한 업데이트가 있을 수 있으므로 형식 체크에만 적용됩니다**

Q&A 세션

standards@wipo.int

참조

- WIPO 서열 홈페이지 (검증기 바이너리 및 운영 매뉴얼 (영문)):

<https://www.wipo.int/standards/en/sequence>

- WIPO 표준 ST.26:

<https://www.wipo.int/export/sites/www/standards/en/pdf/03-26-01.pdf>

- WIPO ST.26 FAQ의 구현 :

<https://www.wipo.int/standards/en/sequence/faq.html>

- WIPO 표준 ST.90:

<https://www.wipo.int/export/sites/www/standards/en/pdf/03-90-01.pdf>

- OAS 2.0 규격:

<https://swagger.io/specification/v2/>