



WIPO Sequence Validator

Руководство

Версия 2.3.0

Настоящий документ призван помочь ведомствам интеллектуальной собственности во внедрении и настройке веб-сервиса WIPO Sequence Validator.

Содержание

1. Введение.....	3
1.1. Общая информация о WIPO Sequence Validator.....	3
1.2. Структура файловой системы WIPO Sequence Validator.....	5
2. Внедрение WIPO Sequence Validator.....	6
2.1. Запуск WIPO Sequence Validator в виде файла Spring Boot JAR.....	6
2.1.1. Внедрение WIPO Sequence Validator в виде исполняемого приложения.....	7
2.2. Внедрение в виде веб-службы WAR.....	8
3. Отчет о проверке.....	9
4. Запрос от конечной точки обратного вызова.....	10
4.2. Отчет о проверке.....	14
5. Конфигурация.....	15
5.1. Настройки по умолчанию.....	15
5.2. Настройка правила проверки «VXQV_49».....	17
5.3. Проверка работоспособности конечной точки.....	17
5.4. Локализованные сообщения.....	18
5.5. Названия пользовательских организаций.....	18
5.6. Указание файлов DTD стандарта ST.26.....	18
5.6.1. Указание альтернативной версии DTD для проведения проверки.....	19
6. REST API WIPO Sequence Validator.....	20
6.1. Проверка файла стандарта ST.26.....	20
6.2. Запрос статуса проверки.....	22
Приложение I: Пример отчета о проверке в формате XML.....	23
Приложение II. Полная спецификация API (YAML).....	24
Приложение III. Названия свойств (JSON).....	24
Приложение IV. Пример отчета о проверке в формате HTML.....	24

1. Введение

Основная цель разработки WIPO Sequence Validator (далее по тексту — «WIPO Sequence Validator» или «программа») заключается в предоставлении ведомствам ИС (ВИС) веб-сервиса для проверки файлов в формате XML стандарта ВОИС ST.26 на предмет соответствия указанному стандарту. Если для подготовки перечня последовательностей используется приложение WIPO Sequence, перечень будет соответствовать требованиям стандарта ВОИС ST.26, но пользователи могут использовать любой наиболее подходящий инструмент.

В настоящем документе объясняется, как внедрить, настроить и использовать программу. В случае возникновения сбоев обратитесь к справочной вики-странице по следующему адресу: <https://www3.wipo.int/confluence/display/WSVAL/Troubleshooting>

1.1. Общая информация о WIPO Sequence Validator

Программа позволяет выполнять следующие четыре операции:

- проверку файла стандарта ST.26;
- запрос состояния выполняемой проверки;
- обновление файлов конфигурации (эта операция может быть выполнена только администратором ВИС); и
- вызов конечной точки обратного вызова с передачей результата проверки по завершении процесса.

Примечание: конечная точка обратного вызова¹ находится вне программы WIPO Sequence Validator. Ведомства должны создать конечную точку сами при настройке микрослужбы.

Программа состоит из файла JAR, который выполняется в качестве веб-службы, или из файла WAR, который запускается на сервере Tomcat.

В обоих случаях для проверки перечня последовательностей стандарта ВОИС ST.26 программа получает файлы из локальной файловой системы и генерирует отчет о проверке, а также может возвращать результат проверки, то есть отчет о проверке, путем вызова конечной точки обратного вызова.

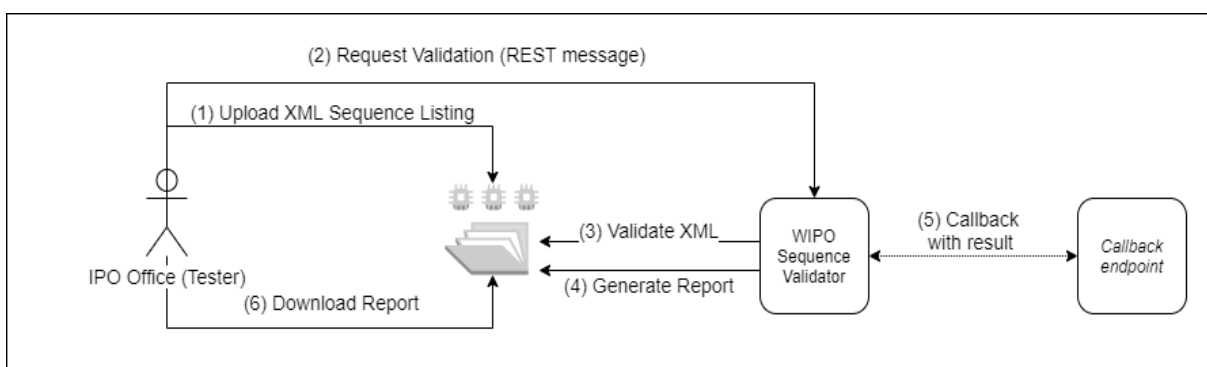
WIPO Sequence Validator функционирует следующим образом:

- а) компьютерная система соответствующего ВИС сохраняет файл XML стандарта ST.26 в заданной по умолчанию папке «Inbox» или в указанной в запросе папке;
- б) система ВИС инициирует запрос HTTP Post на проверку файла. В зависимости от конфигурации система ВИС может инициировать проверку файла в режиме «full» (полная проверка) или «formality» (формальная проверка). При проверке файла в режиме «formality» проверяется, представлен ли файл стандарта ST.26 в формате XML и соответствует ли определение типа документа в файле требованиям стандарта ST.26. При проверке файла в режиме «full» проверяется соответствие файла правилам, прописанным в стандарте ВОИС ST.26, и выполняется проверка в режиме «formality»;

¹ Конечная точка обратного вызова в данном случае означает уникальный URI-адрес, на который можно отправлять запросы.

Примечание. в системе с онлайн-отправкой файлов на проверку рекомендуется использовать только режим «formality», поскольку этот режим позволяет выполнять проверку синхронно, а режим «full» рекомендуется использовать для пакетной обработки файлов, поскольку такая проверка занимает намного больше времени.

- c) после выполнения проверки программа отправляет ответ с указанием того, прошел ли файл проверку в режиме «formality» или нет, а если компьютерная система ВИС выполняет проверку в режиме «full», в ответе дополнительно указывается, корректно ли запустился процесс проверки файла на соответствие правилам, прописанным в стандарте;
- d) если выполняется проверка в режиме «full», программа загружает файл XML из папки «Inbox» и запускает проверку на соответствие правилам, после чего выполняются нижеследующие операции;
- e) программа создает файл отчета в формате XML («report_<filename>.xml») на основе XSD Schema в указанной папке «Input» и перемещает проверенный файл XML стандарта ВОИС ST.26 в папку «Outbox». В качестве дополнительной опции Validator также может одновременно создать HTML-версию отчета о проверке.
- f) Как только файл отчета в формате XML создан и проверен на соответствие XSD-схеме, валидатор создает файл отчета в формате HTML в указанной папке под названием «Reports».
- g) по завершении проверки на соответствие правилам из программы осуществляется вызов конечной точки обратного вызова (если конечная точка задана), при этом в запросе передаются дополнительные данные, касающиеся процесса проверки. Структура запроса и пример данных приводятся в разделе 4 ниже;
- h) в ответе конечная точка обратного вызова должна вернуть пустое значение или код успешного выполнения (означает, что ошибок не обнаружено). [Примечание: этот шаг выполняется только в том случае, если работает внешняя веб-служба и вызов настроен в программе]. Необходимо обеспечить интеграцию между программой и конечной точкой обратного вызова. Как уже отмечалось выше, внешняя веб-служба не является частью программы и должна быть создана и настроена ведомством в соответствии с приведенными ниже требованиями;
- i) система ВИС может загрузить отчет о проверке из папки «Reports».



Примечание: в отношении обработки и передачи данных, касающихся интеллектуальной собственности, с использованием интерфейса программирования приложений для веб-служб (Web API) WIPO Sequence Validator соответствует требованиям [стандарта ВОИС ST.90](#).

1.2. Структура файловой системы WIPO Sequence Validator

В программе WIPO Sequence Validator используется пять основных папок:

- **папка «Inbox»** — локальная папка, в которую ВИС загружает файлы стандарта ВОИС ST.26 для проверки;
- **папка «Process»** — локальная папка, в которой временно хранятся файлы из папки «Inbox» во время обработки. В этой папке содержится две вложенных папки:
 - **папка «Full validation»** — в этой папке хранятся файлы, отправленные на проверку в режиме «full»;
 - **папка «Formality validation»** — в этой папке хранятся файлы, отправленные на проверку в режиме «formality»;
- **папка «Outbox»** — по завершении проверки программа сохраняет исходный файл стандарта ВОИС ST.26 в этой локальной папке;
- **папка «Reports»** — локальная папка, в которой хранятся результаты проверки в виде файла (файлов) отчета о проверке;
- **папка «Params»** — локальная папка, в которой хранится файл в формате JSON (.json) со всеми переданными в запросе на проверку параметрами, используемыми для проведения асинхронной глубокой проверки.

Пример структуры файловой системы приводится ниже:

```
/temp/ST26
/temp/ST26/inbox
/temp/ST26/process/full
/temp/ST26/process/formality
/temp/ST26/outbox
/temp/ST26/reports
/temp/ST26/params
```

ВАЖНО: По умолчанию каталог `/temp/ST26` должен быть расположен в родительском каталоге, в котором находится программа. Например, если файл JAR или WAR расположен в `C:/dev`, то структура папок должна быть создана в виде `C:/temp/ST26/...`

2. Внедрение WIPO Sequence Validator

Как указано выше, WIPO Sequence Validator предоставляется в виде файла в одном из двух бинарных форматов (см. ниже). В зависимости от типа инфраструктуры, на базе которой ведомство желает внедрить WIPO Sequence Validator, ВИС может выбрать наиболее подходящий формат.

WIPO Sequence Validator предоставляется в виде следующих двух бинарных файлов:

- **бинарный файл Spring Boot JAR** — исполняемый файл JAR. Требуется установка [Java 8](#);
- **бинарный пакет WAR** — бинарный файл для установки в контейнер сервлетов. Требуется сервер приложений, совместимый с Spring Boot 2 и Servlet Spec 3.1+, например [Tomcat 8.5](#).

В нижеследующих разделах подробно описывается процесс внедрения WIPO Sequence Validator в виде приложения на базе [Spring Boot](#) или в виде приложения WAR на сервере Java-приложений.

2.1. Запуск WIPO Sequence Validator в виде файла Spring Boot JAR

Файл Spring Boot JAR содержит встроенный сервер, что позволяет внедрить API WIPO Sequence Validator без необходимости использования отдельного сервера. Это существенно упрощает настройку и внедрение на уровне инфраструктуры. Чтобы запустить встроенный сервер, выполните указанную ниже команду.

Примечание: на сервере уже должна быть установлена Java 8. Поскольку Java не гарантирует поддержку кодировки UTF-8, для системного параметра «file.encoding» необходимо задать значение «UTF-8». Сделать это можно следующим образом:

```
java -D -jar wipo-sequence-validator.jar
```

Доступ к API WIPO Sequence Validator можно получить через интерфейс [Swagger UI](#): [http://\[host-name\]:8080/swagger-ui.html](http://[host-name]:8080/swagger-ui.html)

Доступ к API WIPO Sequence Validator возможен в следующих конечных точках:
[http://\[host-name\]:8080/api/\[version\]/status](http://[host-name]:8080/api/[version]/status)
[http://\[host-name\]:8080/api/\[version\]/validate](http://[host-name]:8080/api/[version]/validate)

При этом ВИС необходимо внести следующие изменения:

- вместо [host-name] указать имя хоста сервера; и
- вместо [version] указать версию API WIPO Sequence Validator (например, v1.0).

По умолчанию сервер использует порт 8080, а изменить порт можно при помощи команды «--server.port» следующим образом:

```
java -D"file.encoding=UTF-8" -jar wipo-sequence-validator.jar --server.port=<port-number>
```

WIPO Sequence Validator использует настройки памяти виртуальной машины Java (JVM), заданные по умолчанию. По умолчанию максимальный размер кучи равняется одной четвертой доступного физического объема памяти.

Изменить максимальный размер кучи можно при помощи параметра «-Xmx» из командной строки²:

```
java -D"file.encoding=UTF-8" -Xmx[size]-jar wipo-sequence-validator.jar
```

2.1.1. Внедрение WIPO Sequence Validator в виде исполняемого приложения

WIPO Sequence Validator может быть установлен в виде службы, контролируемой операционной системой, например, чтобы программа запускалась при загрузке операционной системы.

Файл Spring Boot JAR можно настроить соответствующим образом на всех платформах, с которыми совместимо приложение WIPO Sequence: Windows, Linux и Mac OS.

В руководстве, доступном по ссылке ниже, подробно описано, как создать системную службу, выполняющую файл JAR, для каждой операционной системы. В нем содержится также информация о настройке различных параметров службы и параметров выполнения приложения:

<https://docs.spring.io/spring-boot/docs/current/reference/html/deployment-install.html>

² <https://docs.oracle.com/javase/8/docs/technotes/tools/windows/java.html#BABHDAB>

2.2. Внедрение в виде веб-службы WAR

Что касается второго бинарного файла, то пакет WAR можно установить на существующем сервере Java приложений, например Apache Tomcat 8.5.

Примечание: требуется контейнер, совместимый с Servlet 3.1.

Далее приводятся инструкции для сервера приложений Tomcat. В приведенных инструкциях «\$TOMCAT_ROOT» означает корневую папку на сервере Tomcat, и это значение необходимо заменить, указав соответствующий путь к корневой папке:

- a. остановите сервер, выполнив команду: \$TOMCAT_ROOT\bin\catalina.bat stop;
- b. скопируйте файл WAR в папку \$TOMCAT_ROOT\webapps\wipo-sequence-validator.war
- c. запустите сервер, выполнив команду: \$TOMCAT_ROOT\bin\catalina.bat start

Примечание: поскольку Java не гарантирует поддержку кодировки UTF-8, при запуске сервера приложений для системного параметра «file.encoding» необходимо задать значение «UTF-8». Сделать это можно следующим образом:

-D"file.encoding=UTF-8"

Доступ к API WIPO Sequence Validator можно получить через интерфейс Swagger UI, как показано выше:

<http://host-name:8080/wipo-sequence-validator/swagger-ui.html>

Доступ к API WIPO Sequence Validator возможен в следующих конечных точках:

[http://\[host-name\]:8080/wipo-sequence-validator/api/\[version\]/status](http://[host-name]:8080/wipo-sequence-validator/api/[version]/status)

[http://\[host-name\]:8080/wipo-sequence-validator/api/\[version\]/validate](http://[host-name]:8080/wipo-sequence-validator/api/[version]/validate)

При этом ВИС необходимо внести следующие изменения:

- вместо [host-name] указать имя хоста сервера; и
- вместо [version] указать версию API WIPO Sequence Validator (например, v1.0).

По умолчанию сервер использует порт 8080. Чтобы задать другой порт, необходимо внести изменения в файл конфигурации Tomcat, следуя инструкции по ссылке:

https://tomcat.apache.org/tomcat-8.5-doc/config/http.html#Common_Attributes

WIPO Sequence Validator использует настройки памяти виртуальной машины Java (JVM), заданные по умолчанию. По умолчанию максимальный размер кучи равняется одной четвертой доступного физического объема памяти.

Изменить максимальный размер кучи можно при помощи параметра «-Xmx» из командной строки, как показано в разделе 2.1 выше.

3. Отчет о проверке

Отчет о проверке может быть сформирован в двух форматах: XML и HTML (или оба). Программа генерирует отчет в формате XML; шаблон отчета приводится ниже.

На корневом уровне указаны следующие атрибуты:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<VerificationReport applicationNumberText = "123" productionDate = "YYYY-MM-DD"
filingDate = "YYYY-MM-DD" softwareBuildVersion = "2.3.0.-SNAPSHOT" softwareVersion
="2.3.0" sourceFileName="[ST.26 filename]" xsi:noNamespaceSchemaLocation = "st26-
seq1-verification-report-2.0.0.xsd" xmlns:xsi= "http://www.w3.org/2001/XMLSchema-
instance" >
  <VerificationMessageBag
    >
    <VerificationMessage>
      <Severity>[ERROR | WARN | XML_WARN | XML_ERROR]</Severity>
      <DataElement>[ST.26 element]</DataElement>
      <DetectedSequence>[Sequence ID]</DetectedSequence>
      <DetectedValue>[value]</DetectedValue>
      <MessageKey>[Message key]</MessageKey>
      <ParameterBag
        <Parameter key="param key">Param value</Parameter>
      <ParameterBag>
      <LocalizedMessage> [Localized message] </LocalizedMessage>
    </VerificationMessage>
  ...
```

- «applicationNumberText» – приложение, связанное с данным перечнем последовательностей;
- «productionDate» – дата проведения проверки;
- «filingDate» – дата подачи заявления;
- «softwareBuildVersion» – версия WIPO Sequence Validator, использованная для проверки;
- «softwareVersion» – версия WIPO Sequence, использованная для создания перечня последовательностей; и
- «sourceFileName» – имя XML-экземпляра перечня последовательностей.

Пример отчета о проверке приводится в приложении I к настоящему руководству, а допустимые значения для этих компонентов приводятся в приложении III.

В следующей классификации представлена указываемая степень серьезности сообщений:

- ERROR — ошибка, выданная при проверке в режиме «full»
- WARNING — предупреждение, выданное при проверке в режиме «full»
- XML_ERROR — ошибка, выданная при проверке в режиме «formality»
- XML_WARN — предупреждение, выданное при проверке в режиме «formality»

Отчет о проверке также составляется в формате HTML с использованием той же таблицы стилей, которая используется в WIPO Sequence. Пример отчета о проверке в формате HTML

приведен в приложении IV к настоящему руководству.

4. Запрос от конечной точки обратного вызова

Запрос, отправляемый WIPO Sequence Validator конечной точкой обратного вызова, должен содержать указанные ниже параметры, задающие место хранения необходимых файлов и режим проверки:

```
{
  "currentApplicationNumber": "string",
  "currentSQLVersionNumber": "string",
  "parentApplicationNumber": "string",
  "parentSQLVersionNumber": "string",
  "seqInputLocation": "C:/temp/valid2Warning.xml",
  "verificationReportOutputPath": "C:/temp/report.xml",
  "nameFile": "valid2Warning.xml",
  "type": "full"
}
```

В поле «seqInputLocation» запроса на проверку указывается путь к перечню последовательностей в формате XML, проверку которого необходимо выполнить. Если оставить это поле пустым, программа попытается выполнить проверку файла XML с названием, указанным в поле «nameFile», в используемой по умолчанию папке «Inbox». Параметр «nameFile» указывает файл перечня последовательностей, проверку которого необходимо выполнить.

Параметр «verificationReportOutputPath» указывает путь к месту сохранения файла отчета о проверке (.xml или/и .html) в зависимости от настроек файла application.property, сгенерированного программой. Если оставить это поле пустым или указать неверный путь, отчет о проверке сохраняется в заданной по умолчанию папке «Reports».

4.1. Формат запроса от конечной точки обратного вызова

Если задан параметр «api.URL», WIPO Sequence Validator попытается отправить результаты проверки в конечную точку с указанным URL-адресом.

Для взаимодействия с WIPO Sequence Validator конечная точка обратного вызова должна соответствовать следующему контракту веб-службы (YAML):

https://www.wipo.int/standards/en/sequence/callback_end_point_web_service_contract.yml

Запрос должен представлять собой объект в формате JSON со следующей структурой:

```
{
  "currentApplicationNumber":
  "string",
  "currentSEQLVersionNumber":
  "string", "elapsedTime": 0,
  "endTime":
  "string",
  "errorSummary":
  [
    {
      "dataElement": "string",
      "detectedSequence": "string", "index":
      0,
      "key": "string",
      "locmessage": "string",
      "params": {
        "additionalProp1":
        "string",
        "additionalProp2":
        "string",
        "additionalProp3":
        "string"
      },
      "paramsForXML": [
        {
          "key": "string",
          "value": "string"
        }
      ],
      "reportValue": "string",
      "sequenceIDNumber":
      "string",
      "type": "string"
    }
  ],
  "httpStatus": "string",
  "parentApplicationNumber":
  "string",
  "parentSEQLVersionNumber":
  "string", "processID": "string",
  "seqIDQuantity": 0,
  "seqInputQuantity": 0, "seqType": "string", "startTime": "string",
  "totalErrorQuantity": 0,
  "totalWarningQuantity": 0,
  "verificationReportOutputPath":
  "string"
}
```

Ниже приводится пример объекта в формате JSON, который будет отправлен внешней конечной точке, направившей WIPO Sequence Validator вызов:

```
{
  "processID": "1608194222169dvVE",
  "seqLType": "ST.26",
  "httpStatus": "SUCCESS",
  "currentApplicationNumber": "string",
  "currentSQLVersionNumber": "string",
  "parentApplicationNumber": "string",
  "parentSQLVersionNumber": "string",
  "verificationReportOutputPath":
  "C:/temp/report.xml", "startTime": "2020-12-
  17 09:36:54.000000",
  "endTime": "2020-12-17 09:37:26.000607",
  "elapsedTime": 32607,
  "totalWarningQuantity": 1,
  "totalErrorQuantity": 2,
  "seqInputQuantity": 3,
  "seqIDQuantity":
  3,
  "errorSummary":
  [
    {
      "index": 0, "reportValue": "",
      "type": "WARNING",
      "params": com.wipo.st26.ipotool.models.ServiceRequest@58
      87858, "key": "X_EARLIEST_PRIO_APPLICATION_ID_MISSING",
      "locmessage": "Earliest priority application information is absent. It must be
      provided when a priority claim is made to an earlier application.",
      "detectedSequence": "",
      "dataElement": "PROPERTY_NAMES.EARLIEST_PRIORITY_APPLICATION"
    },
  ],
}
```

```
    "reportValue": "",
    "type": "WARNING",
    "params": com.wipo.st26.ipotool.models.ServiceRequest@5887
858, "key": "X_EARLIEST_PRIO_APPLICATION_ID_MISSING",
    "locmessage": "Earliest priority application information is absent. It must be
provided when a pri ority claim is made to an earlier application.",
    "detectedSequence": "",
    "dataElement": "PROPERTY_NAMES.EARLIEST_PRIORITY_APPLICATION"
},
{
    "index": 0,
    "reportValue":
    "-",
    "type": "ERROR",
    "params": {},
    "key": "INVENTION_TITLE_MISSING",
    "locmessage": "The invention title is missing. At least one invention title
must be entered.", "detectedSequence": "",
    "dataElement": "PROPERTY_NAMES.INVENTION_TITLE_BAG"
},
{
    "index": 1,
    "reportValue":
    "-",
    "type": "ERROR",
    "params": {},
    "key": "INVENTION_TITLE_MISSING",
    "locmessage": "The invention title is missing. At least one invention title
must be entered.", "detectedSequence": "",
    "dataElement": "PROPERTY_NAMES.INVENTION_TITLE_BAG"
}
]
}
```

4.2. Отчет о проверке

Как уже говорилось в разделе 3, по завершении проверки создается отчет о проверке в месте, указанном в параметре «verificationReportOutputPath»; по умолчанию для файла отчета в формате XML это: «/temp/st26/reports/[verificationID]/report.xml», в то время как для файла отчета в формате HTML это: «/temp/st26/reports/[verificationID]/report.html».

Вы можете включить или отключить создание отчетов в формате HTML в файле «application.properties». Для создания отчетов укажите значение «true», а для отключения данной функции используйте «false».

```
#turning on/of printing HTML Report  
app.property.html=true
```

Содержимое отчета отправляется конечной точке обратного вызова в поле «errorSummary» запроса «ServiceRequest». Пример этого поля приводится в примере запросов в разделе 4 выше.

5. Конфигурация

5.1. Настройки по умолчанию

Конфигурация WIPO Sequence Validator задается в файле конфигурации. По умолчанию в файле конфигурации «application.properties» заданы следующие значения³:

```
##### WIPO Sequence Validator properties

## -- FOLDERS --
#Base path to be used by the rest of folders
app.basePath=/temp/st26/
#Folder to put the files to be processed\
app.inboxPath=${app.basePath}inbox/

#Folder to store the ST26 files once validated
app.outboxPath=${app.basePath}outbox/
#Folder to store the validation reports
app.reportsPath=${app.basePath}reports/

#Folder to store the parameters
app.paramsPath=${app.basePath}params/

#Parent folder for full and formality folders
app.processPath=${app.basePath}process/

#Files in process for a full validation are stored in this folder
app.process.fullPath=${app.processPath}full/
#Files in process for a formality validation are stored in this folder
app.process.formalityPath=${app.processPath}formality/

alternativeResourceBasePath=${app.basePath}alt_resources

## --PREFERENCES --
#To enable the rule VXQV49 set this value to true, default value is false.
app.preferences.optionalEnglishQualifierValue=false
# Please enter either: ERROR or WARNING to specify the type of the verification message
for the rule VXQV_49, default value is "WARNING".
app.preferences.optionalRuleType=WARNING

#locale used for the localized messages from the verification report
validator_locale=en
#Setting the location of the endpoint
api.URL=
#turning on/of printing HTML Report
app.property.html=true
#software version from pom.xml
app.version=@project.version@
```

³ Действительны на февраль 2023 года

```
## -- WATCHER --

# These properties control the process looking for files in the folders to be processed
# (see: https://docs.spring.io/spring-framework/docs/current/javadoc-api/org.springframework.scheduling.concurrent.ThreadPoolTaskExecutor.html)

processing.delay=10000
processing.corePoolSize=5
#Max number of files being validated concurrently
processing.maxPoolSize=10
processing.queueCapacity=1000 processing.enabled=true

##-- LOGGING (see https://logback.qos.ch/manual/configuration.html)

logging.level.root=info
logging.level.org.wipo=info
logging.pattern.console=%d{yyyy-MM-dd HH:mm:ss} [%thread] %-
5level %logger{36} - %msg%n

# HEALTH ENDPOINT
management.endpoints.jmx.exposure.include=health #
Show details of health endpoint
management.endpoint.health.show-details=always
```

Чтобы изменить значения параметров, следует использовать другой файл «application.properties». Имеется несколько вариантов, которые подробно описаны в документации по Spring Boot: <https://docs.spring.io/spring-boot/docs/2.0.6.RELEASE/reference/html/boot-features-external-config.html#boot-features-external-config-application-property-files>

Самый простой вариант — использовать новый файл «application.properties», поиск которого осуществляется в следующих папках в указанном порядке:

- a) в папке «/config» в текущем каталоге [*Примечание: при внедрении WIPO Sequence Validator в виде файла WAR на сервере Tomcat, эта папка находится в папке «lib» («/opt/apache-tomcat/lib/config»)];*
- b) в текущем каталоге [*Примечание: при внедрении WIPO Sequence Validator в виде файла WAR на сервере Tomcat, этой папкой является папка «lib» («/opt/apache-tomcat/lib/»)];*
- c) по адресу «\$classpath» или в пакете конфигурации «config», а затем
- d) в корневом каталоге «\$classpath».

Кроме того, путь к файлу конфигурации и его имя можно задать в соответствующем параметре в командной строке при запуске программы:

при внедрении программы в виде файла JAR:

- `java -jar -Dspring.config.location= <PATH_TO_FILE> wipo-sequence-validator.jar`

при внедрении программы в виде файла WAR на сервере Tomcat добавьте следующую запись к «CATALINA_OPTS»:

- `export CATALINA_OPTS="-Dspring.config.location=<PATH_TO_FILE>`

При внедрении программы в виде файла WAR можно также скопировать новый файл «application.properties» в папку «WEB-INF/classes» веб-приложения или отредактировать существующий.

5.2. Настройка правила проверки «VXQV_49»

```
app.preferences.optionalEnglishQualifierValue=false
app.preferences.optionalRuleType=WARNING
```

Значение «optionalEnglishQualifierValue» в файле «application.properties» может быть установлено как «true», если пользователь хочет включить правило «VXQV_49», также пользователь может установить серьезность правила путем изменения значения «optionalRuleType» на «ERROR» или «WARNING». Выше для этих двух параметров показаны значения по умолчанию.

5.3. Проверка работоспособности конечной точки

```
# HEALTH ENDPOINT
management.endpoints.jmx.exposure.include=health
```

В WIPO Sequence Validator была реализована конечная точка «/health», которая предоставляет основную информацию о «здоровье» программы.

Конечная точка «/health» доступна по следующему URL-адресу: <http://localhost:8080/wipo-sequence-validator/actuator/health> URL.

Значение конечной точки должно быть следующим:

- Если работоспособность программы не нарушена, будет указан статус «UP»
- Если работоспособность программы нарушена из-за какого либо сбоя, такого как отсутствие соединения с базой данных или недостаток свободного места на диске и др., будет указан статус «DOWN».

Конечная точка «/health» показывает статус только в простом виде: «UP» или «DOWN». Следующий параметр в файле «application.properties» предоставляет информацию в полной мере, включая статус каждого индикатора работоспособности, проверенного в рамках процесса оценки работоспособности.

```
# Show details of health endpoint
management.endpoint.health.show-
```

Конечная точка «/health» теперь включает данные индикатора «DiskSpaceHealthIndicator», который запускается как часть процесса проверки работоспособности.

Конечная точка оценки работоспособности будет отображаться следующим образом в виде серии пар ключ-значение и включать дополнительную информацию. Ниже приводится пример:

```
{"status": "UP", "details": {"diskSpace": {"status": "UP", "details": {"total": 511123124224, "free": 373225091072, "threshold": 10485760}}}}
```

5.4. Локализованные сообщения

WIPO Sequence Validator может отображать локализованные сообщения, например в отчете о проверке, на каждом из десяти официальных языков PCT (английском, арабском, испанском, китайском, немецком, португальском, корейском, русском, французском и японском).

По умолчанию сообщения отображаются на английском языке. Чтобы WIPO Sequence Validator отображал сообщения на других языках, укажите код соответствующего языка в параметре «validator_locale» в файле «application.properties».

```
#Local used for the localized messages from the verification report
validator_locale=en
```

Примечание: чтобы применить параметры, заданные в новом файле «application.properties», перезапустите WIPO Sequence Validator.

5.5. Названия пользовательских организмов

Чтобы ведомства могли использовать собственные названия пользовательских организмов, которые отсутствуют в предопределенном списке названий организмов, список пользовательских организмов можно задать путем создания нового файла с названием «custom_organism.json» в папке, указанной в параметре «alternativeResourceBasePath». Этот файл должен иметь следующую структуру:

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

Примечание: в отличие от предопределенного списка названий организмов, все организмы хранятся в одном файле JSON, а не в отдельных файлах JSON для каждой буквы алфавита.

5.6. Указание файлов DTD стандарта ST.26

По умолчанию WIPO Sequence Validator использует последнюю версию определения типа документа (DTD) стандарта ST.26. Текущая версия WIPO Sequence Validator основана на DTD версии 1.3 согласно стандарту ВОИС ST.26³.

Копия последней версии файла DTD стандарта ST.26 включена в библиотеку WIPO Sequence Validator, которая находится в папке «/src/main/resources» (этот путь к файлу задан в файле JAR или WAR). Файл DTD указывается в файле «catalog.xml», который находится в той же папке, как показано ниже:

```
<?xml version="1.0" encoding="UTF-8"?>
<catalog xmlns="urn:oasis:names:tc:entity:xmlns:xml:catalog">
  <public publicId="-//WIPO//DTD Sequence Listing 1.3//EN"
    uri="ST26SequenceListing_V1_3.dtd"/>
</catalog>
```

Инструкции относительно того, как указать другую версию DTD, приводятся ниже. Во время проверки будет использоваться версия DTD, указанная в объявлении DOCTYPE в файле XML. Сначала по параметру «publicId» определяется, где находится файл DTD, который будет использоваться. Если «publicId» не указан в файле «catalog.xml», система попытается найти файл DTD в корневой папке, где выполняется процесс Java.

³ Действительно на февраль 2023 года

5.6.1. Указание альтернативной версии DTD для проведения проверки

Чтобы иметь возможность проверять файлы стандарта ВОИС ST.26, которые ссылаются на более старую версию DTD стандарта ST.26, соответствующий файл DTD должен быть доступен WIPO Sequence Validator для выполнения проверки.

Это можно обеспечить двумя способами:

- распакуйте файл JAR и поместите дополнительный или альтернативный файл DTD стандарта ST.26 в папку «src/main/resources».
- Внесите изменения в файл «catalog.xml» и добавьте новую запись для дополнительного файла DTD ST.26 или отредактируйте существующую запись.

Например:

```
<?xml version="1.0" encoding="UTF-8"?>
<catalog xmlns="urn:oasis:names:tc:entity:xmlns:xml:catalog">
<public publicId="-//WIPO//DTD Sequence Listing 1.2//EN"
uri="ST26SequenceListing_V1_2.dtd"/>
<public publicId="-//WIPO//DTD Sequence Listing 1.3//EN"
uri="ST26SequenceListing_V1_3.dtd"/>
</catalog>
```

Вместо внесения изменений в файл JAR можно выполнить следующее:

- a. скопируйте файл «catalog.xml» и все файлы DTD в любую локальную папку;
- b. внесите изменения в файл «catalog.xml» и добавьте запись для дополнительного файла DTD стандарта ST.26; и
- c. установите следующий системный параметр Java при запуске:
«xml.catalog.files=<path_to_catalog.xml>»

Примечание: в Tomcat для Windows это можно сделать, добавив следующую переменную среды:

```
set "JAVA_OPTS=%JAVA_OPTS%
-Dxml.catalog.files=C:\\temp\\tomcat\\sharedclasspath\\catalog.xml"
```

[ВАЖНО: указание другой версии DTD стандарта ST.26 позволит провести проверку файла XML в режиме «formality» — на предмет соответствия указанному DTD стандарта ST.26, а проверка в режиме «full», вероятно, потребует изменения исходного кода для реализации правил проведения проверки. Поэтому несколько файлов DTD рекомендуется использовать только при проведении проверки в режиме «formality».]

6. REST API WIPO Sequence Validator

В этом разделе приводятся примеры использования API WIPO Sequence Validator. Имеется три варианта использования:

- проверка файла в папке «Inbox»; (раздел 6.1)
- проверка файла при получении запроса (раздел 6.1); и
- запрос состояния проверки (раздел 6.2).

Полная спецификация API для этой службы (написана на OAS 3.0 [YAML File]) приводится в приложении II.

6.1. Проверка файла стандарта ST.26

Маппинг запроса	/api/v1/validate
Метод	POST
Входные данные	application/json
Выходные данные	application/json
Операция	Запрос на проверку существующего файла стандарта ВОИС ST.26 в папке «Inbox». Возвращается уникальный «verificationID» для получения статуса запроса на проверку.
Запрос	<pre>{ "currentApplicationNumber": "string", "currentSQLVersionNumber": "string", "parentApplicationNumber": "string", "parentSQLVersionNumber": "string", "sqlInputLocation": "string" (местонахождение файла «Input.xml»), "verificationReportOutputPath": "string" (место сохранения файла «report.xml»), "nameFile": "file.xml", "type": "string" (возможные значения: full formality), }</pre>

<p>Ответы</p>	<ul style="list-style-type: none"> • '202': "Accepted". Файл стандарта ВОИС ST.26 прошел проверку в режиме «formality», и началась его проверка на соответствие установленным правилам. Этот ответ будет также содержать уникальный идентификатор для получения отчета о проверке («verificationID»). Файл стандарта ВОИС ST.26 перемещается в папку «Process» для обработки. • '400': "Bad request". Запрос REST сформирован неправильно, или файл стандарта ВОИС ST.26 не прошел проверку на предмет того, представлен ли файл в формате XML. Этот ответ будет также содержать сведения об ошибке. • '400': "Bad request" – "FILENAME_NOT_VALID". Имя файла содержит недопустимые символы. Хотя имя файла и может включать нелатинские символы, некоторые знаки препинания считаются недопустимыми. • '404': "Not Found". Файл стандарта ВОИС ST.26 не найден в папке «Inbox». • '500': "Internal Server Error". Возникла внутренняя ошибка. Этот ответ будет также содержать сведения об ошибке.
<p>Предварительное условие</p>	<p>Файл стандарта ВОИС ST.26 должен находиться в папке «Inbox», указанной пользователем.</p>
<p>Постусловие</p>	<p>Файл стандарта ВОИС ST.26 перемещается в место, указанное в параметре «verificationReportOutputPath», или в папку «Outbox» в следующее место:</p> <p>“/[outbox]/[verificationID]/[file.xml]”</p> <p>Отчет о проверке создается в следующем месте:</p> <p>“/[reports]/ [verificationID]/report_[file.xml или file.html]”</p>

Примечание: проверенное имя файла может включать нелатинские символы, однако никакие зарезервированные символы приняты не будут.

6.2. Запрос статуса проверки

Маппинг запроса	/api/v1/status
Метод	POST
Входные данные	application/json
Выходные данные	application/json
Операция	Запрос статуса проверки для конкретного файла стандарта ВОИС ST.26.
Запрос	<pre>{ verificationID: {type: string} }</pre>
Ответы	<ul style="list-style-type: none"> • '200': Success. Этот ответ будет также содержать уникальный идентификатор со статусом процесса проверки, который может принимать следующие значения: <ul style="list-style-type: none"> ○ «RUNNING»: файл обрабатывается; ○ «FINISHED-VALID»: файл успешно прошел проверку в режиме «formality», и результат проверки доступен в папке отчетов; ○ «FINISHED-INVALID»: процесс завершен, но файл не прошел проверку в режиме «formality». Результат проверки доступен в папке отчетов; ○ «NOT_FOUND»: идентификатор «verificationID» не найден; ○ «VERIFICATION_ID_ERROR»: в запросе отсутствует идентификатор «verificationID». • '400': Bad request. Запрос REST сформирован неправильно. • '500': Server Error. Возникла внутренняя ошибка. Этот ответ будет также содержать сведения об ошибке.
Постусловие	WIPO Sequence Validator указывает статус проверки.
Допущения	-

[Приложение I следует]

Приложение I: Пример отчета о проверке в формате XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<VerificationReport productionDate="2020-12-17" sourceFileName="valid2Warning.xml">
  <VerificationMessageBag>
    <VerificationMessage>
      <Severity>WARNING</Severity>
      <DataElement>PROPERTY_NAMES.EARLIEST_PRIORITY_APPLICATION</DataElement>
      <DetectedSequence/>
      <DetectedValue/>
      <MessageKey>X_EARLIEST_PRIO_APPLICATION_ID_MISSING</MessageKey>
      <ParameterBag/>
      <LocalizedMessage>Earliest priority application information is absent.It must be provided when a priority claim is made to an earlier application.
    </LocalizedMessage>
    </VerificationMessage>
  </VerificationMessageBag>
  <VerificationMessageBag>
    <VerificationMessage>
      <Severity>ERROR</Severity>
      <DataElement>PROPERTY_NAMES.INVENTION_TITLE_BAG</DataElement>
      <DetectedSequence/>
      <DetectedValue>-</DetectedValue>
      <MessageKey>INVENTION_TITLE_MISSING</MessageKey>
      <ParameterBag/>
      <LocalizedMessage>The invention title is missing. At least one invention title must be entered.</LocalizedMessage>
    </VerificationMessage>
  </VerificationMessageBag>
  <VerificationMessageBag>
    <VerificationMessage>
      <Severity>ERROR</Severity>
      <DataElement>PROPERTY_NAMES.INVENTION_TITLE_BAG</DataElement>
      <DetectedSequence/>
      <DetectedValue>-</DetectedValue>
      <MessageKey>INVENTION_TITLE_MISSING</MessageKey>
      <ParameterBag/>
      <LocalizedMessage>The invention title is missing. At least one invention title must be entered.</LocalizedMessage>
    </VerificationMessage>
  </VerificationMessageBag>
</VerificationReport>
```

Этот экземпляр в формате XML также может быть загружен по ссылке:
https://www.wipo.int/standards/en/sequence/example_verification_report.xml

[Приложение II следует]

Приложение II. Полная спецификация API (YAML)

Полная спецификация API может быть загружена по ссылке:

https://www.wipo.int/standards/en/sequence/complete_validator_api_specification.yml

[Приложение III следует]

Приложение III. Названия свойств (JSON)

Названия свойств, используемых в приложении, могут быть загружены по ссылке:

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

Обратите внимание, что это часть исходного кода, она обновляется по мере необходимости параллельно с внедрением.

[Приложение IV следует]

Приложение IV. Пример отчета о проверке в формате HTML

Verification report

Verification Report Information

Production Date	2023-01-25
Application Number	123456
Sequence listing dated	2021-02-13
Software Version	2.2.0-SNAPSHOT
Validator Version	0.1

Verification Messages

Severity	Data Element	Message Text	Detected Value	Detected Sequence
ERROR	Feature Location	The feature location includes a residue number greater than the length of the sequence, which is invalid.	14219	1
WARNING	Project	A non English free text language code has been entered, but there are no qualifiers with a non-English free text value.	en	

[Конец документа]