

СТАНДАРТ VOIB ST.97

РЕКОМЕНДАЦІЇ ЩОДО ОБРОБКИ ДАНИХ ПРО ІНТЕЛЕКТУАЛЬНУ ВЛАСНІСТЬ З ВИКОРИСТАННЯМ JSON (JAVASCRIPT OBJECT NOTATION)

Схвалено Комітетом зі стандартів VOIB (КСВ)
на його десятій сесії
25 листопада 2022 року

ЗМІСТ

СТАНДАРТ VOIB ST.97	1
1. ВСТУП.....	2
2. ВИЗНАЧЕННЯ ТА ТЕРМІНИ	2
3. ЗАГАЛЬНІ ПОЗНАЧЕННЯ.....	2
3.1 Ідентифікатори правил	3
3.2 Приклад структури даних JSON	3
4. СФЕРА ЗАСТОСУВАННЯ.....	3
5. ЗАГАЛЬНІ ПРАВИЛА ПРОЄКТУВАННЯ JSON.....	4
5.1 Огляд.....	4
5.2 Угоди про імена JSON.....	4
6. ПРАВИЛА ПРОЄКТУВАННЯ СХЕМИ JSON (JSON SCHEMA)	6
6.1 Огляд.....	6
6.2 Модульність.....	7
6.3 Документація.....	15
6.4 Ім'я файлу.....	17
6.5 Структурування властивостей схеми JSON	17
7. ПРАВИЛА ПРОЄКТУВАННЯ КОНСТРУКЦІЙ СХЕМИ JSON	18
7.1 Огляд.....	18
7.2 Властивості.....	18
7.3 Визначення	19
7.4 Визначення типів.....	19
7.5 Базові (primitive) типи JSON	20
7.6 Списки кодів	20
7.7 Масиви	21
7.8 Об'єкти	21
8. ІДЕНТИФІКАТОРИ СХЕМИ JSON.....	22
8.1 Огляд.....	22
9. ПРАВИЛА ПРОЄКТУВАННЯ ЕКЗЕМПЛЯРА JSON (JSON INSTANCE)	22
9.1 Порядок властивостей	22
9.2 Перевірка екземпляра JSON	22
10. ПОСИЛАННЯ	23
Стандарти VOIB.....	23
Стандарти та галузеві специфікації	23
ДОДАТОК I - ПРАВИЛА ПЕРЕТВОРЕННЯ XSD СТАНДАРТУ VOIB ST.96 НА СХЕМУ JSON ТА РЕКОМЕНДАЦІЇ ЩОДО ВИКОРИСТАННЯ	24
ДОПОВНЕННЯ - ІНСТРУМЕНТ ПЕРЕТВОРЕННЯ СХЕМИ XSD В JSON	54
ДОДАТОК II - СХЕМА JSON	55
ДОДАТОК III - ПРИКЛАДИ JSON	56
ДОДАТОК IV - ПЕРЕЛІК АКРОНІМІВ ТА АБРЕВІАТУР	57
ДОДАТОК V - ТЕРМІНИ ПРЕДСТАВЛЕННЯ	60

ЗМІСТ (ТАБЛИЦІ)

Таблиця 1: Елементи документації заголовка схеми JSON.....	17
Таблиця 2: Перетворення простих типів даних XSD	27
Таблиця 3 Перетворення обмежувальних аспектів XSD (X - числове значення обмеження).....	54

1. ВСТУП

Цей стандарт містить рекомендації щодо проектування, створення або оновлення ресурсів нотації об'єктів JavaScript (JSON) для використання під час подання, обробки, обміну або публікації даних щодо всіх типів об'єктів інтелектуальної власності (ІВ). У цьому стандарті також розглядаються правила перетворення схем розширюваної мови розмітки (XML), які відповідають стандарту BOIB ST.96 (XSD), у схеми JSON, які відповідають вищезазначеним рекомендаціям.

Цей Стандарт призначений для:

- Надання рекомендацій стосовно найкращих методів проектування та розроблення даних ІВ у форматі JSON;
- Забезпечення узгодженості шляхом надання схем та екземплярів JSON на основі стандарту BOIB ST.96 для обміну даними про ІВ;
- Надання рекомендацій стосовно принципів проектування для розширення наданих схем JSON або створення нових сумісних схем JSON; і
- Підвищення ефективності обміну даними шляхом сприяння повторному використанню ресурсів JSON відомствами інтелектуальної власності (ІВ), а також даних, доступних для громадськості.

2. ВИЗНАЧЕННЯ ТА ТЕРМІНИ

У цьому Стандарті використовуються такі терміни:

- Термін "ресурси JSON" призначений для позначення будь-якого з компонентів, що використовуються для створення та використання JSON відповідно до цього Стандарту;
- Терміни "object", "object type", "property", "member", "property name", "property value", "property type", "keyword" та "definition" у цьому Стандарті мають розумітися так само, як і в документі JSON Schema Core, версія draft-2020-12¹;
- Термін "конструкція" у цьому Стандарті слід інтерпретувати як "будівельний блок", з якого будуються схеми JSON;
- Термін "глобальне визначення" використовується для визначення, на яке можуть посилатися інші визначення в тій самій схемі або визначення в інших схемах; і
- У цьому Стандарті терміни "ПОВИНЕН", "НЕ ПОВИНЕН", "СЛІД", "НЕ СЛІД", "РЕКОМЕНДУЄТЬСЯ", "МОЖЕ" і "НЕ ОБОВ'ЯЗКОВО" слід розуміти так, як описано в RFC 2119². Форми цих слів, що не пишуться з великої літери, використовуються у звичайному значенні.

3. ЗАГАЛЬНІ ПОЗНАЧЕННЯ

У цьому стандарті використовуються такі позначення:

¹ Версія схеми JSON може бути змінена, оскільки вона не набула статусу RFC; і не була прийнята робочою групою IETF. Ця версія Стандарту заснована на останній версії, тобто 2020-12, доступній за адресою

<https://json-schema.org/draft/2020-12/json-schema-core.html>

² <https://www.ietf.org/rfc/rfc2119.txt>

- `<>`: вказує описовий термін-заповнювач, який під час реалізації буде замінено конкретним значенням;
- `" "`: вказує на те, що текст, взятий в лапки, повинен використовуватися дослівно при реалізації;;
- `{ }`: вказує на те, що пункти не є обов'язковими для виконання при реалізації; і
- Шрифт `Courier New` використовується для написання ключових слів JSON, назв властивостей JSON, а також елементів і атрибутів XSD.

3.1 Ідентифікатори правил

Усі правила проектування є нормативними. Правила проектування позначаються префіксом [JXX-nn].

- Значення "JXX" є префіксом для класифікації типу правила наступним чином:
 - (a) JGD для загальних правил проектування;
 - (b) JSD для правил проектування схеми JSON;
 - (c) JCD для правил проектування конструкцій; і
 - (d) JID для правил проектування екземплярів
- Значення "nn" вказує наступне доступне число в послідовності певного типу правила. Слід зазначити, що цифра не означає позицію правила, зокрема, для нового правила. Нове правило буде розміщене у відповідному контексті. Наприклад, ідентифікатор правила [JGD-10] визначає десяте загальне правило проектування. Правило [JGD-10] можна розмістити між правилами [JGD-05] і [JGD-06] замість [JGD-09], якщо це найбільш підходяще місце для цього правила.
- Ідентифікатор видаленого правила буде збережено до заміни опису правила текстом "Видалено".

3.2 Приклад структури даних JSON

Приклади структур даних JSON відображаються в текстових полях з використанням шрифту фіксованої ширини. Зразок синтаксису структури даних JSON виділяється для зручності читання.

4. СФЕРА ЗАСТОСУВАННЯ

Цей Стандарт призначено для опису ресурсів JSON, що використовуватимуться для подання, публікації, оброблення та обміну даними IB та супутньою інформацією. Цей стандарт призначений для надання рекомендацій для ВІВ та інших організацій, які працюють з даними та документами, що стосуються патентів, торговельних марок, промислових зразків, географічних зазначень та/або «творів-сиріт», що охороняються авторським правом.

Цей стандарт покликаний надати рекомендації для ВІВ та інших організацій, які створюють або змінюють дані про IB у вигляді ресурсів JSON. Відповідність цьому стандарту потрібна для обміну даними між ВІВ з використанням ресурсів JSON, таких як схеми, приклади, повідомлення та корисні дані для інтерфейсів прикладного програмування (API). Правила проектування та правила трансформації написані з

урахуванням правил та угод щодо правил проектування стандарту BOIB ST.96. Однак правила проектування та угоди стандарту BOIB ST.96 не зіставляються один в один із правилами та угодами проектування JSON, і тому правила проектування стандарту BOIB ST.96 дублюються і в деяких випадках трохи змінені, де це може бути застосовано.

Цей Стандарт включає такі Додатки:

- Додаток I: Правила перетворення схем XML стандарту BOIB ST.96 у схеми JSON, які містять Доповнення: інструмент перетворення для перетворення XSD стандарту BOIB ST.96 у схеми JSON;
- Додаток II: Схеми JSON, перетворені зі схем XML стандарту BOIB ST.96 версії 5.0³;
- Додаток III: Приклади випадків JSON;
- Додаток IV: Список акронімів та абревіатур; і
- Додаток V: Представлення термінів.

Цей Стандарт не включає:

- (a) Проблеми архітектури програмного забезпечення; і
- (b) Мови реалізації.

5. ЗАГАЛЬНІ ПРАВИЛА ПРОЄКТУВАННЯ JSON

5.1 Огляд

Розділ містить загальні високорівневі правила та рекомендації з проектування JSON, які застосовуються до всього обміну даними JSON і зусиль із розроблення JSON, а не до конкретної мови програмування, що виконує маршалінг/демаршалінг даних у/з JSON. Загальні правила та рекомендації, перелічені нижче, забезпечують загальну основу для розроблення схеми JSON, екземпляра JSON і структури даних JSON для всіх даних, включно з даними IB і даними, які не стосуються IB, наприклад дані змішаного вмісту. Рівні вкладеності СЛІД зводити до мінімуму при створенні нових схем JSON, екземплярів JSON і розробці структур даних JSON, які не доступні в стандарті BOIB ST.96 або яким не обов'язково бути сумісним зі стандартом BOIB ST.96.

5.2 Угоди про імена JSON

Ці угоди необхідні для забезпечення узгодженості, однаковості та повноти іменування і визначення всіх ресурсів JSON.

Угоди про імена JSON засновані на рекомендаціях і принципах, описаних у документі [ISO 11179](#), частина 5⁴

- Принципи іменування та ідентифікації. Імена об'єктів та імена властивостей складаються з таких термінів:
- *Об'єктний клас (Object Class)* стосується дії або об'єкта в контексті виробничої діяльності і представляє собою логічні групування або агрегування даних (у логічній моделі даних), до якої належить властивість. Об'єктний клас виражається терміном об'єктного класу.

³ Перетворені схеми JSON мають такі самі імена тегів і структуру даних, як визначено в стандарті BOIB ST.96, Додаток III, включно з XML-компонентами змішаного вмісту і зовнішніми стандартами, тобто MathML і Oasis Table, для взаємодії з даними у форматі стандарту BOIB ST.96.

⁴ <https://www.iso.org/standard/60341.html>

- *Властивість (Property Term)* визначає характеристики об'єктного класу.
- *Кваліфікатор (Qualifier Term)* – це слово або слова, які допомагають визначити та відрізнити елемент даних від інших пов'язаних елементів даних, і, за необхідності створення унікального імені, можуть бути приєднані до об'єктного класу або властивості.
- *Термін представлення (Representation Term)* класифікує формат даних елемента відповідно до наборів типів. Слід використовувати терміни представлення, наведені в Додатку V.

- [JGD-01] Імена типів об'єктів і властивостей ПОВИННІ складатися зі слів англійської мови з використанням основних варіантів написання англійської мови, наведених в Оксфордському словнику англійської мови. Допустимими винятками є акроніми, аббревіатури та інші скорочення слів, наведені в Додатку IV.
- [JGD-02] Для типу об'єкта та імен властивостей СЛІД обирати тільки іменники, прикметники та дієслова в теперішньому часі, за винятком акронімів, аббревіатур та інших скорочень слів, наведених у Додатку IV.
- [JGD-03] Символи, які використовуються в іменах властивостей, ПОВИННІ міститися в такому наборі: 'a-z, A-Z і 0-9'.
- [JGD-04] Максимальна довжина імен типів об'єктів і властивостей ПОВИННА не перевищувати 35 символів.
- [JGD-05] Імена типів об'єктів і властивостей ПОВИННІ бути лаконічними і такими, що не потребують пояснень.
- [JGD-06] В іменах об'єкта та імена властивостей ПОВИННА використовуватись угода lowerCamelCase (LCC). Наприклад, "currencyCode": "EUR".
- [JGD-07] В іменах типів об'єктів ПОВИННА використовуватись угода LCC і суфікс Type. Наприклад, applicantType.
- [JGD-08] Акроніми і скорочення, перераховані в Додатку IV, ПОВИННІ завжди використовуватись замість повного розширеного імені.
- [JGD-09] Акроніми та аббревіатури ПОВИННІ використовуватись відповідно до переліку Додатку IV для імен властивостей і типів об'єктів.
- [JGD-10] Термін об'єктного класу ПОВИНЕН завжди мати одне й те саме семантичне значення в усьому конкретному домені IB, такому як патенти, торговельні марки, промислові зразки, географічні зазначення або авторське право, і МОЖЕ складатися з більш ніж одного слова. Наприклад, `contactInformation`.
- [JGD-11] Термін властивості в імені ПОВИНЕН бути унікальним у контексті об'єктного класу, але МОЖЕ повторно використовуватись в різних об'єктних класах.
- [JGD-12] Кваліфікатор МОЖЕ бути приєднаний до терміна об'єктного класу або терміна властивості, якщо необхідно зробити ім'я унікальним.
- [JGD-13] Якщо ім'я містить термін об'єктного класу, термін властивості та термін

представлення, термін об'єктного класу ПОВИНЕН передувати терміну властивості, а термін властивості ПОВИНЕН передувати терміну представлення. Кваліфікатор ПОВИНЕН передувати пов'язаному терміну об'єктного класу або терміну властивості. Наприклад, `claimTotalQuantity`.

- [JGD-14] Якщо властивість закінчується тим самим словом, що й термін представлення (або еквівалентним словом), тоді термін представлення ПОВИНЕН бути видалений.
- [JGD-15] Там, де потрібен термін представлення, в іменах основних компонентів ПОВИННІ використовуватися терміни представлення з Додатку V.
- [JGD-16] У домені IB усі типи об'єктів та імена властивостей ПОВИННІ бути унікальними.
- [JGD-17] Слово (слова) в імені СЛІД використовувати в однині, якщо саме поняття не у множині. Наприклад, `totalMarkSeries`.
- [JGD-18] До імені властивості або типу об'єкта, яке містить набір контекстуально пов'язаних компонентів, СЛІД додавати суфікс "Bag". Наприклад, `emailAddressBag` представляє колекцію елементів `emailAddress`.
- [JGD-19] Сполучні слова, такі як "and", "of" і "the", НЕ СЛІД використовувати в іменах типів об'єктів і властивостей, якщо тільки вони не є частиною ділової термінології.
- [JGD-20] Тип об'єкта та імена властивостей НЕ ПОВИННІ бути перекладені, змінені або замінені для будь-яких цілей.
- [JGD-21] Тип об'єкта та імена властивостей НЕ ПОВИННІ посилатися на номери статей і правил. Наприклад, `PCTRule702C` для PCT.
- [JGD-22] Рівні вкладеності СЛІД зводити до мінімуму при створенні нових схем JSON, примірників JSON і фрагментів структур даних JSON, які недоступні в стандарті BOIB ST.96.
- [JGD-23] Для нового типу об'єкта та імен властивостей, не визначених в стандарті BOIB ST.96 або сумісність яких з ім'ям компонента, визначеним в стандарті BOIB ST.96 не передбачається, СЛІД використовувати вбудовані описові терміни чи імена, а не загальні чи короткі типи об'єктів або імена властивостей, що є об'єктами з однією властивістю. Наприклад, замість `"inventor": {"fullName": "Thomas Edison" }` краще використовувати `"inventorFullName": "Thomas Edison"`.

6. ПРАВИЛА ПРОЄКТУВАННЯ СХЕМИ JSON (JSON SCHEMA)

6.1 Огляд

Схема JSON описує структуру екземпляра JSON, який виражає обмеження на структуру та вміст документа. Цей стандарт має бути приведений у відповідність до галузевої специфікації схеми JSON. Останньою версією, доступною на момент публікації

цього Стандарту, є проєкт 2020-12⁵, і ця версія Стандарту відповідає зазначеному проєкту специфікації.

- [JSD-01] Схеми JSON ПОВИННІ відповідати специфікаціям схеми JSON під назвою: JSON Schema Core, версія 2020-12, яка доступна за адресою <https://json-schema.org/latest/json-schema-core.html>, та JSON Schema Validation, версія 2020-12, доступна за адресою <https://json-schema.org/latest/json-schema-validation.html>.
- [JSD-02] Схеми JSON ПОВИННІ вказувати, що вони відповідають версії 2020-12 схеми JSON, використовуючи ключове слово `$schema` зі значенням "<https://json-schema.org/draft/2020-12/schema>".

Приклад: зазначення версії схеми JSON

```
"$schema": "https://json-schema.org/draft/2020-12/schema"
```

Схема має використовувати кодування UTF-8 для максимальної сумісності.

- [JSD-03] Схеми JSON ПОВИННІ використовувати набір символів Unicode (UCS) відповідно до стандарту ISO/IEC 10646. Для кодування символів Unicode ПОВИННА використовуватися система кодування UTF-8.

6.2 Модульність

Модульність дає змогу створювати компоненти схеми для підтримки гнучкості проєктування та повторного використання. Під час проєктування рекомендується уникати визначення всіх властивостей і логічних компонентів у єдиній монолітній схемі JSON, що перешкоджає можливості спільного використання та повторного використання окремих властивостей або логічних компонентів, визначених як група в схемі.

Нижче наведено схему, яка **не** дотримується принципу модульності. Подібне **НЕ РЕКОМЕНДУЄТЬСЯ** цим Стандартом.

applicationNumber.json (невірний приклад складеного документа схеми)

```
{
  "$id" : "applicationNumber.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "applicationNumber" : {
      "$ref" : "#/$defs/applicationNumber"
    }
  },
  "required" : [ "applicationNumber" ],
  "$defs" : {
    "applicationNumber" : {
      "description" : "Description" : "Опис: Номери, які використовуються ВІВ для ідентифікації кожної отриманої заявки; Version: V5_0",
      "type" : "object",
      "additionalProperties" : false,
      "properties" : {
```

⁵ <https://json-schema.org/draft/2020-12/schema>

```

"ipOfficeCode" : {
  "anyOf" : [ {
    "type" : "string",
    "enum" : [ "AD", "AE", "AF", "AG", "AI", "AL", "AM", "AO",
"AP", "AR", "AT", "AU", "AW", "AZ", "BA", "BB", "BD", "BE", "BF", "BG",
"BH", "BI", "BJ", "BM", "BN", "BO", "BQ", "BR", "BS", "BT", "BV",
"BW", "BX", "BY", "BZ", "CA", "CD", "CF", "CG", "CH", "CI", "CK", "CL",
"CM", "CN", "CO", "CR", "CU", "CV", "CW", "CY", "CZ", "DE", "DJ",
"DK", "DM", "DO", "DZ", "EA", "EC", "EE", "EG", "EH", "EM", "EP", "ER",
"ES", "ET", "EU", "FI", "FJ", "FK", "FO", "FR", "GA", "GB", "GC",
"GD", "GE", "GG", "GH", "GI", "GL", "GM", "GN", "GQ", "GR", "GS", "GT",
"GW", "GY", "HK", "HN", "HR", "HT", "HU", "IB", "ID", "IE", "IL",
"IM", "IN", "IQ", "IR", "IS", "IT", "JE", "JM", "JO", "JP", "KE", "KG",
"KH", "KI", "KM", "KN", "KP", "KR", "KW", "KY", "KZ", "LA", "LB", "LC",
"LI", "LK", "LR", "LS", "LT", "LU", "LV", "LY", "MA", "MC", "MD", "ME",
"MG", "MK", "ML", "MM", "MN", "MO", "MP", "MR", "MS", "MT", "MU", "MV",
"MW", "MX", "MY", "MZ", "NA", "NE", "NG", "NI", "NL", "NO", "NP", "NR",
"NZ", "OA", "OM", "PA", "PE", "PG", "PH", "PK", "PL", "PT", "PW", "PY",
"QA", "QZ", "RO", "RS", "RU", "RW", "SA", "SB", "SC", "SD", "SE",
"SG", "SH", "SI", "SK", "SL", "SM", "SN", "SO", "SR", "SS", "ST",
"SV", "SX", "SY", "SZ", "TC", "TD", "TG", "TH", "TJ", "TL", "TM",
"TN", "TO", "TR", "TT", "TV", "TW", "TZ", "UA", "UG", "US", "UY", "UZ",
"VA", "VC", "VE", "VG", "VN", "VU", "WO", "WS", "XN", "XU", "XV",
"XX", "YE", "ZA", "ZM", "ZW" ],
    "description" : "Опис: Цей перелік кодів відповідає кодам стандарту
ВОІВ ST.3 (двобуквенні коди представлення держав, інших адміністративних
одниць та організацій), опублікованого вересні 2019 року.; Версія: V5_0;
AD: Andorra; AE: United Arab Emirates; AF: Afghanistan; AG: Antigua And
Barbuda; AI: Anguilla; AL: Albania; AM: Armenia; AO: Angola; AP: African
Regional Intellectual Property Organization (ARIPO); AR: Argentina; AT:
Austria; AU: Australia; AW: Aruba; AZ: Azerbaijan; BA: Bosnia and
Herzegovina; BB: Barbados; BD: Bangladesh; BE: Belgium; BF: Burkina Faso;
BG: Bulgaria; BH: Bahrain; BI: Burundi; BJ: Benin; BM: Bermuda; BN: Brunei
Darussalam; BO: Bolivia (Plurinational State of); BQ: Bonaire, Sint
Eustatius and Saba; BR: Brazil; BS: Bahamas; BT: Bhutan; BV: Bouvet
Island; BW: Botswana; BX: Benelux Office for Intellectual Property (BOIP);
BY: Belarus; BZ: Belize; CA: Canada; CD: Democratic Republic of the Congo;
CF: Central African Republic; CG: Congo; CH: Switzerland; CI: Côte
D'Ivoire; CK: Cook Islands; CL: Chile; CM: Cameroon; CN: China; CO:
Colombia; CR: Costa Rica; CU: Cuba; CV: Cabo Verde; CW: Curaçao; CY:
Cyprus; CZ: Czech Republic; DE: Germany; DJ: Djibouti; DK: Denmark; DM:
Dominica; DO: Dominican Republic; DZ: Algeria; EA: Eurasian Patent
Organization (EAPO); EC: Ecuador; EE: Estonia; EG: Egypt; EH: Western
Sahara; EM: European Union Intellectual Property Office (EUIPO); EP:
European Patent Office (EPO); ER: Eritrea; ES: Spain; ET: Ethiopia; EU:
European Union; FI: Finland; FJ: Fiji; FK: Falkland Islands (Malvinas);
FO: Faroe Islands; FR: France; GA: Gabon; GB: United Kingdom; GC: Patent
Office of the Cooperation Council for the Arab States of the Gulf (GCC
Patent Office); GD: Grenada; GE: Georgia; GG: Guernsey; GH: Ghana; GI:
Gibraltar; GL: Greenland; GM: Gambia; GN: Guinea; GQ: Equatorial Guinea;
GR: Greece; GS: South Georgia and South Sandwich Islands; GT: Guatemala;
GW: Guinea-Bissau; GY: Guyana; HK: Hong Kong, China; HN: Honduras; HR:
Croatia; HT: Haiti; HU: Hungary; IB: International Bureau of the World
Intellectual Property Organization (WIPO); ID: Indonesia; IE: Ireland; IL:
Israel; IM: Isle of Man; IN: India; IQ: Iraq; IR: Iran, Islamic Republic
of; IS: Iceland; IT: Italy; JE: Jersey; JM: Jamaica; JO: Jordan; JP:
Japan; KE: Kenya; KG: Kyrgyzstan; KH: Cambodia; KI: Kiribati; KM: Comoros;
KN: Saint Kitts and Nevis; KP: Democratic People's Republic of Korea; KR:
Republic of Korea; KW: Kuwait; KY: Cayman Islands; KZ: Kazakhstan; LA: Lao

```



```

People's Democratic Republic; LB: Lebanon; LC: Saint Lucia; LI:
Liechtenstein; LK: Sri Lanka; LR: Liberia; LS: Lesotho; LT: Lithuania; LU:
Luxembourg; LV: Latvia; LY: Libya; MA: Morocco; MC: Monaco; MD: Republic
of Moldova; ME: Montenegro; MG: Madagascar; MK: North Macedonia; ML: Mali;
MM: Myanmar; MN: Mongolia; MO: Macao, China; MP: Northern Mariana Islands;
MR: Mauritania; MS: Montserrat; MT: Malta; MU: Mauritius; MV: Maldives;
MW: Malawi; MX: Mexico; MY: Malaysia; MZ: Mozambique; NA: Namibia; NE:
Niger; NG: Nigeria; NI: Nicaragua; NL: Netherlands; NO: Norway; NP: Nepal;
NR: Nauru; NZ: New Zealand; OA: African Intellectual Property Organization
(OAPI); OM: Oman; PA: Panama; PE: Peru; PG: Papua New Guinea; PH:
Philippines; PK: Pakistan; PL: Poland; PT: Portugal; PW: Palau; PY:
Paraguay; QA: Qatar; QZ: Community Plant Variety Office (European
Community) (CPVO); RO: Romania; RS: Serbia; RU: Russian Federation; RW:
Rwanda; SA: Saudi Arabia; SB: Solomon Islands; SC: Seychelles; SD: Sudan;
SE: Sweden; SG: Singapore; SH: Saint Helena, Ascension and Tristan da
Cunha; SI: Slovenia; SK: Slovakia; SL: Sierra Leone; SM: San Marino; SN:
Senegal; SO: Somalia; SR: Suriname; SS: South Sudan; ST: Sao Tome and
Principe; SV: El Salvador; SX: Sint Maarten (Dutch part); SY: Syrian Arab
Republic; SZ: Eswatini; TC: Turks and Caicos Islands; TD: Chad; TG: Togo;
TH: Thailand; TJ: Tajikistan; TL: Timor-Leste; TM: Turkmenistan; TN:
Tunisia; TO: Tonga; TR: Turkey; TT: Trinidad and Tobago; TV: Tuvalu; TW:
Taiwan, Province of China; TZ: United Republic of Tanzania; UA: Ukraine;
UG: Uganda; US: United States of America; UY: Uruguay; UZ: Uzbekistan; VA:
Holy See; VC: Saint Vincent and the Grenadines; VE: Venezuela (Bolivian
Republic of); VG: British Virgin Islands; VN: Viet Nam; VU: Vanuatu; WO:
World Intellectual Property Organization (WIPO) (International Bureau of);
WS: Samoa; XN: Nordic Patent Institute (NPI); XU: International Union for
the Protection of New Varieties of Plants (UPOV); XV: Visegrad Patent
Institute (VPI); XX: Unknown states, other entities or organizations; YE:
Yemen; ZA: South Africa; ZM: Zambia; ZW: Zimbabwe"
    }, {
      "type" : "string",
      "enum" : [ "AN", "CS", "DL", "DD", "DT", "RH", "SU", "YD", "YU" ],
      "description" : "Version: V5_0; AN: Netherlands Antilles; CS:
Czechoslovakia; DL: German Democratic Republic; DD: German Democratic
Republic; DT: Federal Republic of Germany; RH: Southern Rhodesia; SU: Soviet
Union; YD: Democratic Yemen; YU: Yugoslavia/ Serbia and Montenegro"
    } ]
  },
  "st13ApplicationNumber" : {
    "type" : "string",
    "pattern" : "\\d{2}\\d{4}\\d{9}",
    "description" : "Description: Application number format recommended
in WIPO Standard ST.13. The sequence of indispensable elements in the
application number format is IP type (2 digits), year designation (4
digits) and serial number (9 digits).; Version: V5_0"
  },
  "applicationNumberText" : {
    "type" : "string",
    "description" : "Description: Free format of application number;
Version: V5_0"
  }
},
"oneOf" : [ {
  "required" : [ "st13ApplicationNumber" ]
}, {
  "required" : [ "applicationNumberText" ]
} ]

```

```

}
}
}

```

Кращий підхід до проектування полягає в розбитті даних на набір невеликих компонентів, представлених модулями схеми, що показано в новій схемі application number, наведеній нижче. Ця схема JSON побудована на менших модулях схеми JSON, індивідуально визначених у їхніх власних схемах.

applicationNumber.json (приклад модульної схеми)

```

{
  "$id" : "applicationNumber.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "applicationNumber" : {
      "$ref" : "#/$defs/applicationNumber"
    }
  },
  "required" : [ "applicationNumber" ],
  "$defs" : {
    "applicationNumber" : {
      "$ref" :
        "applicationNumberType.json#/$defs/applicationNumberType",
      "description" : "Description: Numbers used by IPOs in order to
identify each application received; Version: V5_0"
    }
  }
}

```

applicationNumberType.json (приклад модульної схеми, продовження)

```

{
  "$id" : "applicationNumberType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "applicationNumberType" : {
      "description" : "Version: V5_0",
      "type" : "object", "additionalProperties" : false,
      "properties" : {
        "ipOfficeCode" : {
          "$ref" : "ipOfficeCode.json#/$defs/ipOfficeCode"
        },
        "st13ApplicationNumber" : {
          "$ref" :
            "st13ApplicationNumber.json#/$defs/st13ApplicationNumber"
        }
      },
      "applicationNumberText" : {
        "$ref" :
          "applicationNumberText.json#/$defs/applicationNumberText"
      }
    }
  },
  "oneOf" : [ {
    "required" : [ "st13ApplicationNumber" ]
  }, {
    "required" : [ "applicationNumberText" ]
  } ]
}
}

```

ipOfficeCode.json (приклад модульної схеми, продовження)

```

{
  "$id" : "ipOfficeCode.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "ipOfficeCode" : {
      "$ref" : "#/$defs/ipOfficeCode"
    }
  },
  "required" : [ "ipOfficeCode" ],
  "$defs" : {
    "ipOfficeCode" : {
      "$ref" : "extendedWIPOST3CodeType.json#/$defs/extendedWIPOST3CodeType",
      "description" : "Description: Two-letter alphabetic codes which represent the names of states, other entities and intergovernmental organizations the legislation of which provides for the protection of IP rights or which organizations are acting in the framework of a treaty in the field of IP; Version: V5_0"
    }
  }
}

```

extendedWIPOST3CodeType.json (Продовження прикладу модульної схеми)

```

{
  "$id" : "extendedWIPOST3CodeType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "extendedWIPOST3CodeType" : { "description" : "Version: V5_0",
      "anyOf" : [ {
        "$ref" : "wipoST3CodeType.json#/$defs/wipoST3CodeType"
      }, {
        "$ref" : "wipoFormerST3CodeType.json#/$defs/wipoFormerST3CodeType"
      } ]
    }
  }
}

```

wipoST3CodeType.json (Продовження прикладу модульної схеми)

```

{
  "$id" : "wipoST3CodeType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "wipoST3CodeType" : {
      "description" : "Description: This code list is inline with WIPO Standard ST.3 (two-letter codes for the representation of states, other entities and organizations) published on September, 2019.; Version: V5_0; AD: Andorra; AE: United Arab Emirates; AF: Afghanistan; AG: Antigua And Barbuda; AI: Anguilla; AL: Albania; AM: Armenia; AO: Angola; AP: African Regional Intellectual Property Organization (ARIPO); AR: Argentina; AT:

```

Austria; AU: Australia; AW: Aruba; AZ: Azerbaijan; BA: Bosnia and Herzegovina; BB: Barbados; BD: Bangladesh; BE: Belgium; BF: Burkina Faso; BG: Bulgaria; BH: Bahrain; BI: Burundi; BJ: Benin; BM: Bermuda; BN: Brunei Darussalam; BO: Bolivia (Plurinational State of); BQ: Bonaire, Sint Eustatius and Saba; BR: Brazil; BS: Bahamas; BT: Bhutan; BV: Bouvet Island; BW: Botswana; BX: Benelux Office for Intellectual Property (BOIP); BY: Belarus; BZ: Belize; CA: Canada; CD: Democratic Republic of the Congo; CF: Central African Republic; CG: Congo; CH: Switzerland; CI: Côte D'Ivoire; CK: Cook Islands; CL: Chile; CM: Cameroon; CN: China; CO: Colombia; CR: Costa Rica; CU: Cuba; CV: Cabo Verde; CW: Curaçao; CY: Cyprus; CZ: Czech Republic; DE: Germany; DJ: Djibouti; DK: Denmark; DM: Dominica; DO: Dominican Republic; DZ: Algeria; EA: Eurasian Patent Organization (EAPO); EC: Ecuador; EE: Estonia; EG: Egypt; EH: Western Sahara; EM: European Union Intellectual Property Office (EUIPO); EP: European Patent Office (EPO); ER: Eritrea; ES: Spain; ET: Ethiopia; EU: European Union; FI: Finland; FJ: Fiji; FK: Falkland Islands (Malvinas); FO: Faroe Islands; FR: France; GA: Gabon; GB: United Kingdom; GC: Patent Office of the Cooperation Council for the Arab States of the Gulf (GCC Patent Office); GD: Grenada; GE: Georgia; GG: Guernsey; GH: Ghana; GI: Gibraltar; GL: Greenland; GM: Gambia; GN: Guinea; GQ: Equatorial Guinea; GR: Greece; GS: South Georgia and South Sandwich Islands; GT: Guatemala; GW: Guinea-Bissau; GY: Guyana; HK: Hong Kong, China; HN: Honduras; HR: Croatia; HT: Haiti; HU: Hungary; IB: International Bureau of the World Intellectual Property Organization (WIPO); ID: Indonesia; IE: Ireland; IL: Israel; IM: Isle of Man; IN: India; IQ: Iraq; IR: Iran, Islamic Republic of; IS: Iceland; IT: Italy; JE: Jersey; JM: Jamaica; JO: Jordan; JP: Japan; KE: Kenya; KG: Kyrgyzstan; KH: Cambodia; KI: Kiribati; KM: Comoros; KN: Saint Kitts and Nevis; KP: Democratic People's Republic of Korea; KR: Republic of Korea; KW: Kuwait; KY: Cayman Islands; KZ: Kazakhstan; LA: Lao People's Democratic Republic; LB: Lebanon; LC: Saint Lucia; LI: Liechtenstein; LK: Sri Lanka; LR: Liberia; LS: Lesotho; LT: Lithuania; LU: Luxembourg; LV: Latvia; LY: Libya; MA: Morocco; MC: Monaco; MD: Republic of Moldova; ME: Montenegro; MG: Madagascar; MK: North Macedonia; ML: Mali; MM: Myanmar; MN: Mongolia; MO: Macao, China; MP: Northern Mariana Islands; MR: Mauritania; MS: Montserrat; MT: Malta; MU: Mauritius; MV: Maldives; MW: Malawi; MX: Mexico; MY: Malaysia; MZ: Mozambique; NA: Namibia; NE: Niger; NG: Nigeria; NI: Nicaragua; NL: Netherlands; NO: Norway; NP: Nepal; NR: Nauru; NZ: New Zealand; OA: African Intellectual Property Organization (OAPI); OM: Oman; PA: Panama; PE: Peru; PG: Papua New Guinea; PH: Philippines; PK: Pakistan; PL: Poland; PT: Portugal; PW: Palau; PY: Paraguay; QA: Qatar; QZ: Community Plant Variety Office (European Community) (CPVO); RO: Romania; RS: Serbia; RU: Russian Federation; RW: Rwanda; SA: Saudi Arabia; SB: Solomon Islands; SC: Seychelles; SD: Sudan; SE: Sweden; SG: Singapore; SH: Saint Helena, Ascension and Tristan da Cunha; SI: Slovenia; SK: Slovakia; SL: Sierra Leone; SM: San Marino; SN: Senegal; SO: Somalia; SR: Suriname; SS: South Sudan; ST: Sao Tome and Principe; SV: El Salvador; SX: Sint Maarten (Dutch part); SY: Syrian Arab Republic; SZ: Eswatini; TC: Turks and Caicos Islands; TD: Chad; TG: Togo; TH: Thailand; TJ: Tajikistan; TL: Timor-Leste; TM: Turkmenistan; TN: Tunisia; TO: Tonga; TR: Turkey; TT: Trinidad and Tobago; TV: Tuvalu; TW: Taiwan, Province of China; TZ: United Republic of Tanzania; UA: Ukraine; UG: Uganda; US: United States of America; UY: Uruguay; UZ: Uzbekistan; VA: Holy See; VC: Saint Vincent and the Grenadines; VE: Venezuela (Bolivian Republic of); VG: British Virgin Islands; VN: Viet Nam; VU: Vanuatu; WO: World Intellectual Property Organization (WIPO) (International Bureau of); WS: Samoa; XN:

Nordic Patent Institute (NPI); XU: International Union for the Protection of New Varieties of Plants (UPOV); XV: Visegrad Patent Institute (VPI); XX: Unknown states, other entities or organizations; YE: Yemen; ZA: South Africa; ZM: Zambia; ZW: Zimbabwe",

```
"type" : "string",
"enum" : ["AD", "AE", "AF", "AG", "AI", "AL", "AM", "AO", "AP", "AR",
"AT", "AU", "AW", "AZ", "BA", "BB", "BD", "BE", "BF", "BG", "BH", "BI",
"BJ", "BM", "BN", "BO", "BQ", "BR", "BS", "BT", "BV", "BW", "BX", "BY",
"BZ", "CA", "CD", "CF", "CG", "CH", "CI", "CK", "CL", "CM", "CN", "CO",
"CR", "CU", "CV", "CW", "CY", "CZ", "DE", "DJ", "DK", "DM", "DO", "DZ",
"EA", "EC", "EE", "EG", "EH", "EM", "EP", "ER", "ES", "ET", "EU", "FI",
"FJ", "FK", "FO", "FR", "GA", "GB", "GC", "GD", "GE", "GG", "GH", "GI",
"GL", "GM", "GN", "GQ", "GR", "GS", "GT", "GW", "GY", "HK", "HN", "HR",
"HT", "HU", "IB", "ID", "IE", "IL", "IM", "IN", "IQ", "IR", "IS", "IT",
"JE", "JM", "JO", "JP", "KE", "KG", "KH", "KI", "KM", "KN", "KP", "KR",
"KW", "KY", "KZ", "LA", "LB", "LC", "LI", "LK", "LR", "LS", "LT", "LU",
"LV", "LY", "MA", "MC", "MD", "ME", "MG", "MK", "ML", "MM", "MN", "MO",
"MP", "MR", "MS", "MT", "MU", "MV", "MW", "MX", "MY", "MZ", "NA", "NE",
"NG", "NI", "NL", "NO", "NP", "NR", "NZ", "OA", "OM", "PA", "PE", "PG",
"PH", "PK", "PL", "PT", "PW", "PY", "QA", "QZ", "RO", "RS", "RU", "RW",
"SA", "SB", "SC", "SD", "SE", "SG", "SH", "SI", "SK", "SL", "SM", "SN",
"SO", "SR", "SS", "ST", "SV", "SX", "SY", "SZ", "TC", "TD", "TG", "TH",
"TJ", "TL", "TM", "TN", "TO", "TR", "TT", "TV", "TW", "TZ", "UA", "UG",
"US", "UY", "UZ", "VA", "VC", "VE", "VG", "VN", "VU", "WO", "WS", "XN",
"XU", "XV", "XX", "YE", "ZA", "ZM", "ZW"]
}
}
}
```

st13ApplicationNumber.json (Приклад модульної схеми, продовження)

```
{
  "$id" : "st13ApplicationNumber.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "st13ApplicationNumber" : {
      "$ref" : "#/$defs/st13ApplicationNumber"
    }
  },
  "required" : [ "st13ApplicationNumber" ],
  "$defs" : {
    "st13ApplicationNumber" : {
      "$ref" :
"st13ApplicationNumberType.json#/$defs/st13ApplicationNumberType",
      "description" : "Description: Application number format recommended in
WIPO Standard ST.13. The sequence of indispensable elements in the
application number format is IP type (2 digits), year designation (4 digits)
and serial number (9 digits).; Version: V5_0"
    }
  }
}
```

wipoFormerST3CodeType.json (Приклад модульної схеми, продовження)

```
{
  "$id" : "wipoFormerST3CodeType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "wipoFormerST3CodeType" : {
      "description" : "Version: V5_0; AN: Netherlands Antilles; CS:
Czechoslovakia; DL: German Democratic Republic; DD: German Democratic
Republic; DT: Federal Republic of Germany; RH: Southern Rhodesia; SU: Soviet
Union; YD: Democratic Yemen; YU: Yugoslavia/ Serbia and Montenegro",
      "type" : "string",
      "enum" : [ "AN", "CS", "DL", "DD", "DT", "RH", "SU", "YD", "YU" ]
    }
  }
}
```

st13ApplicationNumberType.json (Приклад модульної схеми, продовження)

```
{
  "$id" : "st13ApplicationNumberType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "st13ApplicationNumberType" : {
      "description" : "Version: V5_0",
      "type" : "string",
      "pattern" : "\\d{2}\\d{4}\\d{9}"
    }
  }
}
```

applicationNumberText.json (Приклад модульної схеми, продовження)

```

{
  "$id" : "applicationNumberText.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "applicationNumberText" : {
      "$ref" : "#/$defs/applicationNumberText"
    }
  },
  "required" : [ "applicationNumberText" ],
  "$defs" : {
    "applicationNumberText" : {
      "type" : "string",
      "description" : "Description: Free format of application number; Version:
v5_0"
    }
  }
}

```

У схемах JSON слід використовувати ключове слово "\$defs" для створення глобальних визначень властивостей і їхнього вмісту, які можна використовувати повторно, як показано в наведеному вище прикладі. Це приблизно еквівалентно створенню глобальних оголошень елементів та іменованих типів у схемі XML.

- [JSD-04] У схемах JSON СЛІД використовувати ключове слово "\$defs", яке включає повторно використовуване визначення для кожної властивості та типу властивості.
- [JSD-05] Розробники ПОВИННІ використовувати існуючі схеми JSON, визначені в Додатку II до цього проєкту стандарту, де це може бути застосовано, перш ніж створювати нові схеми JSON.
- [JSD-06] Розробникам СЛІД створювати нові схеми JSON тільки після того, як буде встановлено, що жодна з існуючих схем JSON не описує конструкцію належним чином.

6.3 Документація

Схеми JSON мають бути самоописовими. Розробникам слід давати значущі імена для конструкцій JSON. Крім того, схемі JSON слід мати документацію, що описує схему та конструкції JSON.

Для сприяння повторному використанню, залишаючи його загальним, у схемі JSON не слід надавати документацію щодо конкретних деталей реалізації системи.

- [JSD-07] У документації НЕ СЛІД описувати деталі реалізації або іншу інформацію, що не має прямого відношення до значення конструкції.

Заголовок схеми JSON дає змогу розробнику схеми легко визначити призначення, використання та вміст схеми. Ця інформація дуже корисна, коли розробнику схеми необхідно вибрати схему для використання в якості шаблону під час створення іншої схеми.

- [JSD-08] До схем JSON СЛІД включати документацію заголовка схеми JSON з використанням ключового слова "description" (опис).
- [JSD-09] Елементи, перелічені в таблиці 1 нижче, СЛІД включати в розділ заголовка всіх схем JSON.

Таблиця 1: Елементи документації заголовка схеми JSON

Назва елемента заголовка	Опис	Обов'язково/необов'язково
Description	Простий текстовий опис інформації, що описується схемою	Обов'язково, за винятком схем JSON, що походять із простих XSD Type ST.96, для яких не надано опис, наприклад DateType
Version	Основний і додатковий номер версії схеми	Обов'язково
SchemaCreatedDate	Дата створення схеми	Необов'язково
SchemaLastModifiedDate	Дата останньої зміни схеми	Необов'язково
SchemaContactPoint	Назва організації, до якої можна звернутися з питаннями про схему	Необов'язково
SchemaReleaseNoteURL	Місце, де публікуються примітки про випуск схеми	Необов'язково

[JSD-10] Елементи документації заголовка, такі як вищезазначені «Опубліковано» і «Номер версії», СЛІД відокремлювати крапкою з комою, з допустимими пробілами після крапки з комою, і наводити значення, пов'язане з ключовим словом "description" (опис). Якщо значення для елемента заголовка недоступне, слід включити тільки мітку, як показано в наступному прикладі:

Приклад документації заголовка для applicationBody (схема рівня документа)

```
"description" : "Description: Body of a patent application; Version: V5_0;
SchemaCreatedDate: 2012-07-13; SchemaLastModifiedDate: 2021-10-01;
SchemaContactPoint: xml.standards@wipo.int; SchemaReleaseNoteURL:
http://www.wipo.int/standards/XMLSchema/ST96/V5\_0/ReleaseNotes.pdf"
```

Приклад документації заголовка для ipOfficeCode (схема не на рівні документа)

```
"description" : " Description: Two-letter alphabetic codes which represent
the names of states, other entities and intergovernmental organizations
the legislation of which provides for the protection of IP rights or which
organizations are acting in the framework of a treaty in the field of IP;
Version: V5_0"
```

Приклад документації заголовка для appellateBodyCategoryType.json (схема визначень типу переліків)

```
"description" : "Version: V5_0; Office appeal board: Appeal board within
the IP office; Court: Court; Appeal Court: Second instance court; Supreme
Court: Highest appellate court"
```


Приклад документації заголовка для схеми визначень типів Type definitions Schema

```
"description" : "Version: V5 0"
```

6.4 Ім'я файлу

Для імені файлу схеми JSON дотримуються правил відповідно до стандарту BOIB ST.96, за винятком того, коли повинна використовуватись угода LCC.

Імена файлів схем та імена схем часто є парними. Імена файлів схеми засновані на відповідних іменах схеми. Наприклад, ім'я файлу `postalAddressType.json` походить від імені схеми `postalAddressType`. Таким чином, угоди про іменування файлів схеми пов'язані з правилами для угод про іменування JSON у цьому стандарті.

Файл схеми МОЖЕ мати інформацію про версію. Схема, що знаходиться на стадії проекту, може бути переглянута. Проекти схем повинні бути позначені як такі у назві файлу схеми, з зазначенням літери «D» та номера версії.

- [JSD-11] Символи, які використовуються в іменах файлів схеми, ПОВИННІ належати до такого набору: a-z, A-Z, 0-9, знак підкреслення "_" і крапка ".".
- [JSD-12] Ім'я файлу схеми ПОВИННЕ складатися з двох обов'язкових частин з одним роздільником і необов'язкової інформації про версію з двома додатковими роздільниками, наприклад: `<ім'я компонента>{"_" "V"< номер основної версії>"_"<номер додаткової версії>"}`. "`<розширення файлу>` "; Наприклад, `emailAddressType.json`, `languageCode.json`, `applicationBody_V1_0.json`.
- [JSD-13] Ім'я файлу проекту схеми ПОВИННЕ складатися з чотирьох обов'язкових частин з двома роздільниками і необов'язкової інформації про версію з двома додатковими роздільниками, наприклад: `<ім'я компонента>{"_" "V"<основний номер версії>"_"<додатковий номер версії>" }_" "D"<номер версії>"."` "`<розширення файлу>`", наприклад, `tradeApplication_V1_1_D1.json`. Якщо проект схеми базується на існуючій схемі і в імені файлу має інформацію про версію, то номери основної та додаткової версій у проекті ПОВИННІ бути такими ж, як вказані у файлі схеми, на якому базується проект схеми. Якщо проект схеми є новим, то номер основної версії в імені файлу проектної схеми ПОВИНЕН бути таким же, як зазначено у відповідному домені IB, а номер додаткової версії у файлі проектної схеми ПОВИНЕН дорівнювати нулю «0».

6.5 Структурування властивостей схеми JSON

Схемам JSON слід мати властивість "type": "object" щоб забезпечити використання JSON тільки для вкладених структур, а не для окремих значень. Взято з прикладу applicationNumber.json нижче:

```
{
  "$id" : "applicationNumber.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "applicationNumber" : {
      "$ref" : "#/$defs/applicationNumber"
    }
  },
  "required" : [
    "applicationNumber" ], "$defs"
: {
  "applicationNumber" : {
    "$ref" :
    "applicationNumberType.json#/$defs/applicationNumberType",
    "description" : "Description: Numbers used by IPOs in order to
    identify each application received; Version: V5_0"
  }
}
}
```

- [JSD-14] Зовнішній об'єкт схеми ПОВИНЕН мати ключове слово "type", зі значенням "object".
- [JSD-15]. Зовнішній об'єкт схеми ПОВИНЕН мати ключове слово "\$defs", значенням якого є властивість зовнішнього об'єкта схеми.
- [JSD-16] Зовнішній об'єкт схеми ПОВИНЕН мати ключове слово "required", значенням якого є масив, що містить один елемент, тобто властивість зовнішнього об'єкта схеми.

Можуть використовуватися розширення (налаштування) схеми JSON. Якщо розширений тип є типом об'єкта, то на нього необхідно посилатися, а не дублювати його властивості, щоб підвищити можливість повторного використання.

- [JSD-17] Розширення (налаштуванн)я схеми JSON для типів об'єктів ПОВИННІ бути реалізовані у вигляді посилання на схему JSON розширеного типу.

7. ПРАВИЛА ПРОЄКТУВАННЯ КОНСТРУКЦІЙ СХЕМИ JSON

7.1 Огляд

У цьому розділі встановлюються правила для конструкцій схеми JSON, зокрема для масивів, об'єктів і базових значень. Стандартизація імен для конструкцій схеми має вирішальне значення для розробки надійної архітектури даних.

7.2 Властивості

Властивості, також відомі як елементи, є основними будівельними блоками конструкції JSON.

- [JSC-01] У визначеннях СЛІД використовувати наявні схеми максимально допустимою можливою мірою.
- [JSC-02] Кілька властивостей, які можуть бути логічно згруповані разом, МОЖУТЬ бути оголошені в одному файлі схеми під глобальним визначенням.

7.3 Визначення

Кожна властивість повинна мати глобальне визначення, яке визначається в її схемі JSON. Це дасть змогу повторно використовувати ім'я властивості в багатьох батьківських елементах і мати узгоджене визначення для всіх них. Див. властивість "applicationNumber" у наведеному нижче прикладі.

[JSC-03] Кожній властивості, зазначеній у ключовому слові властивостей, СЛІД посилатися на глобальне визначення, яке визначено в ключовому слові "\$defs". Глобальне визначення властивості має складатися з назви файлу та опису властивості.

Приклад властивості, що посилається на глобальне визначення

```
{
  "$id" : "applicationNumber.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema", "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "applicationNumber" : {
      "$ref" : "#/$defs/applicationNumber"
    }
  },
  "required" : [ "applicationNumber"
], "$defs" : {
  "applicationNumber" : {
    "$ref" :
      "applicationNumberType.json#/$defs/applicationNumberType",
    "description" : "Description: Numbers used by IPOs in order to
identify each
application received; Version: V5_0"
  }
}
}
```

[JSC-04] Глобальне визначення властивості ПОВИННЕ складатися з імені файлу та опису ("description") властивості.

Властивості повинні мати типи. Їх можна визначити безпосередньо, якщо вони є примітивними типами (крім об'єктів), або обробляти їх через посилання на глобальне визначення властивості в іншій схемі JSON.

[JSC-05] Властивість ПОВИННА мати тип, який задається за допомогою ключового слова "type", або як безпосередньо властивість, або через посилання на глобальне визначення

7.4 Визначення типів

Схеми JSON можуть визначати багаторазові визначення типів, на які посилаються з глобальних визначень властивостей. Ці глобальні визначення типів повинні складатися з ключового слова "type", ключового слова "properties" (якщо тип - "object") і будь-якого іншого значення обмеження.

[JSC-06] Схема МОЖЕ містити визначення глобальних типів для повторного використання моделей вмісту в багатьох властивостях.

Повторно використовуване визначення типу

```

{
  "$id" : "applicationNumber.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema", "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "applicationNumber" : {
      "$ref" : "#/$defs/applicationNumber"
    }
  },
  "required" : [ "applicationNumber" ],
  "$defs" : {
    "applicationNumber" : {
      "$ref" :
        "applicationNumberType.json#/$defs/applicationNumberType",
      "description" : "Description: Numbers used by IPOs in order to
        identify each
        application received; Version: 5_0"
    }
  }
}

```

[JSC-07] Визначення, які представляють типи, ПОВИННІ мати імена, що відповідають умові lower camel case (LCC) + суфікс "Type".

7.5 Базові (primitive) типи JSON

[JSC-08] Для властивості ПОТРІБНО використовувати найбільш підходящий базовий тип JSON. До базових типів належать: "string" (рядок), "number" (число), "integer" (ціле), "object" (об'єкт), "array" (масив), "boolean" (логічний тип) і "null" (нульовий).

Наприклад, якщо значенням властивості буде ціле число, слід використовувати тип "integer", а не більш загальний "number" або ще ширший "string". Для типу "string" слід використовувати вбудовані формати, якщо це можливо, наприклад, "date-time" (дата-час) або "duration" (тривалість).

7.6 Списки кодів

У деяких випадках доцільно обмежити значення переліком кодів, які є стандартними і прийнятними для цілей обміну даними. Переліки кодів - це засіб для створення контрольованого словника дозволених значень для елемента даних (наприклад, стандартний перелік кодів для кодів країн, кодів мов, кодів патентних відомств тощо). Переліки кодів, які вже існують у відкритому доступі та підтримуються відповідними комітетами зі стандартів, такими як ISO.

[JSC-09] Стандарт VOIB [ST.3](#) ПОВИНЕН використовуватися для представлення ВПВ, держав, інших адміністративних одиниць, організацій, а також для пріоритетних і визначених країн/організацій.

[JSC-10] Елементи коду ISO 3166-1-Alpha-2 (двобуквені коди країн) ПОВИННІ використовуватися для представлення назв країн для адресації та громадянства.

[JSC-11] ISO 639-1 (двобуквені коди мов) ПОВИНЕН використовуватися для кодів мов.

[JSC-12] ISO 4217-Alpha (трибуквені коди валют) ПОВИНЕН використовуватися для кодів валют.

- [JSC-13]. Ключове слово JSON `enum` (перерахування) СЛІД використовувати для завдання визначення списків кодів.
- [JSC-14]. Символи, що використовуються в значеннях перерахування, ПОВИННІ бути обмежені таким набором:
{a-z, A-Z, 0-9, крапка (.), кома (,), пробіли, дефіс (-) і знак підкреслення ()}

7.7 Масиви

Термін кардинальність визначається як кількість елементів у масиві. Кардинальність вказується на схемі за допомогою ключових слів `minItems` та `maxItems`. Розробникам схем рекомендується не вказувати значення за замовчуванням для (наприклад, «`minItems`»: 0), оскільки це може призвести до надмірного захаращення схеми.

- [JSC-15] У схемах JSON СЛІД використовувати ключові слова `minItems` і `maxItems` для масивів, за винятком значення за замовчуванням `minItems` (0).

Тип елементів у масиві має бути визначений за допомогою ключового слова `"items"`. Для спрощення всі елементи масиву повинні мати один і той самий тип. Якщо потрібна послідовність об'єктів різних типів, їх слід визначити як окремі властивості об'єкта.

- [JSC-16] Для кожного об'єкта типу масив ПОВИННЕ бути ключове слово `"items"`, і його значенням ПОВИНЕН бути об'єкт схеми, а не масив. Усі елементи масиву ПОВИННІ мати один і той самий тип.

Ключове слово `"additionalItems"` не можна використовувати для масивів, оскільки воно не має значення, якщо значенням `"items"` є один об'єкт схеми.

- [JSC-17] Ключове слово `"additionalItems"` НЕ СЛІД використовувати, якщо `"items"` є одним об'єктом схеми.

7.8 Об'єкти

7.8.1 Властивість "wildcards" (підставні знаки)

Схеми JSON не повинні допускати, щоб довільні властивості були частиною екземпляра JSON, причому коректними, оскільки це може порушити цілісність обміну даними.

Потрібно використовувати ключове слово `"AdditionalProperties"`, значення якого має бути встановлене в `"false"`. В іншому випадку в екземплярах будуть дозволені невизначені властивості.

- [JSC-18] Схема JSON ПОВИННА використовувати `"AdditionalProperties"` і встановити його значення - `"false"` для кожного об'єкта.

Використання ключового слова `"patternProperties"` не допускається. Це ключове слово дає змогу відображати регулярні вирази в схемі. Наприклад, воно допускає неявні визначення схеми на основі імені властивості. Для певного виду імені властивості застосовується певна схема.

- [JSC-19] У схемі НЕ ПОВИННО використовуватися ключове слово `"patternProperties"`.

7.8.2 Порядок розташування властивостей

Схема JSON не вимагає певного порядку для властивостей об'єкта. Однак якщо схема JSON має відповідну схему XML, рекомендується, щоб властивості як у схемі JSON, так і в екземплярі, були перераховані в тому ж порядку, що й у схемі XML.

[JSC-20] В схемі СЛІД використовувати той самий порядок властивостей, що й у відповідній схемі XML, якщо така існує.

8. ІДЕНТИФІКАТОРИ СХЕМИ JSON

8.1 Огляд

Ідентифікатор у схемі JSON надає URI, який ідентифікує категорію інформації на основі бізнес-домену (наприклад, підприємство, патенти та торговельні марки). У цьому проєкті стандарту було ухвалено рішення використовувати зв'язки "багато до одного" між кількома ідентифікаторами і, можливо, сотнями конструкцій JSON. Група пов'язаних конструкцій JSON з унікальними іменами будуть пов'язані з певним ідентифікатором (ID). Для ідентифікації слід використовувати універсальний ідентифікатор ресурсу (URI).

[JID-01] Ідентифікатори ПОВИННІ застосовуватися в схемах із використанням ключового слова "\$id".

9. ПРАВИЛА ПРОЄКТУВАННЯ ЕКЗЕМПЛЯРА JSON (JSON INSTANCE)

Схема JSON визначає структуру й обмеження для екземпляра JSON. Щоб поліпшити якість і забезпечити надійність обміну даними (внутрішнього та міжвідомчого), екземпляри JSON повинні співвідноситися зі схемою JSON з метою перевірки коректності та відповідності.

9.1 Порядок властивостей

Схема JSON не вимагає певного порядку для властивостей об'єктів. Однак, рекомендується, щоб властивості відображалися в екземплярі в тому ж порядку, що й у схемі JSON.

[JIN-01] В екземплярі документа JSON СЛІД використовувати той самий порядок властивостей, що й у відповідній схемі JSON, якщо така існує.

9.2 Перевірка екземпляра JSON

Успішна перевірка екземплярів JSON на коректність гарантує, що його вміст задовольняє всім вимогам, визначеним у відповідних схемах.

[JIN-02] Примірники документів JSON МОЖУТЬ перевірятися на відповідність схемі в процесі обробки.

[JIN-03] Для задоволення вимог продуктивності додатку в середовищі виконання МОЖЕ бути створена схема часу виконання. Наприклад, складена схема для номера додатку може використовуватися як схема часу виконання.

[JIN-04] Усі модифікації, оновлення, виправлення та нові випуски ПОВИННІ бути спочатку надіслані на схвалення Цільовій групі XML4IP, перш ніж зміни можна буде включити до схеми часу виконання.

Бажано, щоб екземпляр JSON відповідав схемі.

[JIN-05] Екземпляр JSON ПОВИНЕН відповідати певній схемі JSON, яка відповідає правилам, описаним у цьому стандарті.

10. ПОСИЛАННЯ

Стандарти ВОІВ

- Стандарт ВОІВ [ST.96](#) : Рекомендації щодо обробки інформації з інтелектуальної власності з використанням XML (розширюваної мови розмітки)
- Стандарт ВОІВ [ST.90](#): Рекомендації щодо обробки та обміну даними -стосовно інтелектуальної власності з використанням вебінтерфейсів API (інтерфейсів прикладного програмування)

Стандарти та галузеві специфікації

Специфікація JSON: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>.

Ядро схеми JSON, проєкт 2020-2012 рр.: <http://json-schema.org/latest/json-schema-core.html>

Перевірка схеми JSON, проєкт 2020-2012 рр.: <http://json-schema.org/latest/json-schema-validation.html>

Специфікація OpenAPI v3.1.0: <https://spec.openapis.org/oas/v3.1.0#schema>

Стандарт ISO 21778 ECMA-404 – стандарт ISO 21778. Синтаксис обміну даними JSON: <https://www.ecma-international.org/publications/standards/Ecma-404.htm>

UBL-2.1 Універсальна ділова мова: <http://docs.oasis-open.org/ubl/UBL-2.1.html>

Коди країн ISO 3166: <https://www.iso.org/iso-3166-country-codes.html>

Коди мов ISO 639: <https://www.iso.org/iso-639-language-codes.html>

Поточні коди ISO 4217: <https://www.iso.org/iso-4217-currency-codes.html>

ISO 11179: <https://www.iso.org/standard/60341.html>

RFC 2119: <https://www.ietf.org/rfc/rfc2119.txt>

[Додаток I наведено далі]

ДОДАТОК I

ПРАВИЛА ПЕРЕТВОРЕННЯ XSD СТАНДАРТУ VOIB ST.96 НА СХЕМУ JSON ТА РЕКОМЕНДАЦІЇ ЩОДО ВИКОРИСТАННЯ

*Схвалено Комітетом зі Стандартів VOIB (КСВ)
на його десятій сесії
25 листопада 2022 року*

ЗМІСТ

ВСТУП	24
АЛГОРИТМ ПЕРЕТВОРЕННЯ ТА РЕКОМЕНДАЦІЇ	25
СФЕРА ЗАСТОСУВАННЯ	25
ПЕРЕТВОРЕННЯ ІМЕНІ ФАЙЛУ	25
ПЕРЕТВОРЕННЯ ПРОСТОРІВ ІМЕН	25
ПЕРЕТВОРЕННЯ ВБУДОВАНИХ ТИПІВ ДАНИХ	26
ВИЗНАЧЕННЯ XSD	27
ПОСИЛАННЯ НА ГЛОБАЛЬНУ СХЕМУ	35
Формувальники XSD	36
ЕЛЕМЕНТИ	41
АТРИБУТИ	43
ПРОСТИЙ ТИП	43
КОМПЛЕКСНИЙ ТИП	44
АННОТАЦІЯ	46
ОБ'ЄДНАННЯ	48
РОЗШИРЕННЯ	48
ОБМЕЖЕННЯ	49
ОБМЕЖУВАЛЬНІ АСПЕКТИ	50
ГРУПА	52
ПЕРЕТВОРЕННЯ ЗОВНІШНІХ ЗАЛЕЖНОСТЕЙ XSD	53

ВСТУП

Хоча XML (eXtensible Markup Language, розширювана мова розмітки), заснована на стандартах XML VOIB, все ще широко використовується, відомства інтелектуальної власності (ВІВ) та індустрія інтелектуальної власності (ІВ) поступово приймають та використовують формат JSON. Стандарт VOIB ST.96 рекомендує використовувати ресурси XML для подання, публікації, обробки та обміну інформацією щодо різних видів ІВ, а саме інтелектуальної власності (ІВ), тобто патентів, торговельних марок, промислових зразків, географічних зазначень і авторських прав. ВІВ впроваджують стандарт VOIB ST.96 у тому вигляді, в якому він опублікований, або адаптований відповідно до вимог.

Для полегшення обміну даними між ВІВ та розповсюдження даних ВІВ у двох форматах, тобто XML та JSON, важливе значення має узгодженість даних, структура даних та сумісність імен тегів між цими двома форматами. Цієї узгодженості та сумісності даних можна досягти за допомогою сумісних схем XML та JSON, які використовуватимуться для валідації екземплярів XML та JSON відповідно.

Додаток I призначений для надання правил і рекомендацій щодо перетворення XSD відповідно до стандарту VOIB ST.96 у відповідні схеми JSON. Схеми JSON, представлені в Додатку II стандарту VOIB ST.97, є результатом застосування цих правил перетворення, і ВІВ рекомендується дотримуватися їх під час розроблення своїх схем JSON, які будуть узгоджені та сумісні з їхніми налаштованими XSD відповідно до стандарту VOIB ST.96.

Як в Доповненні, так і в Додатку I міститься інструмент перетворення, який застосовує ці правила для використання тими BIB, які бажають розробити власні схеми JSON на основі їхніх XSD налаштованими відповідно до стандарту BOIBST.96.

На момент підготовки, версія 2020-12 є останнім проектом версії схеми JSON, а версія 5.0 є останньою версією стандарту BOIB ST.96. Таким чином, правила та настанови щодо перетворення ґрунтуються на версії JSON 2022-12 і версії 5.0 стандарту BOIB ST.96, у якій використовується XSD 1.1. Надані правила мають узгоджуватися з правилами, визначеними в основній частині стандарту BOIB ST.97. Правила перетворення визначені в цьому Додатку за допомогою префікса 'TR.

АЛГОРИТМ ПЕРЕТВОРЕННЯ ТА РЕКОМЕНДАЦІЇ

Для того, щоб перетворити XSD, що відповідають стандарту BOIB ST.96, в схеми JSON, слід застосовувати наступний алгоритм:

1. Перетворити ім'я файлу XSD на ім'я файлу схеми JSON; див. Правила іменування файлів у Розділі 0
2. Перетворити фрагмент XSD на фрагмент, сумісний зі схемою JSON

Слід зазначити, що перетворення XSD у схеми JSON не вимагає суворого порядку. Це означає, що перевірку згенерованих схем JSON слід запустити вже після того, як усі вони будуть згенеровані, оскільки згенерований файл схеми JSON може мати залежності "\$ref", які ще не згенеровані.

СФЕРА ЗАСТОСУВАННЯ

Оскільки схеми JSON, опубліковані як частина цього стандарту, що є результатом перетворення XSD стандарту BOIB ST.96, цей Додаток I охоплює тільки конструкції XSD, ключові слова та інші елементи, що використовуються в стандарті BOIB ST.96.

ПЕРЕТВОРЕННЯ ІМЕНІ ФАЙЛУ

[TR-01] Імена файлів схеми JSON та імена об'єктів або властивостей схеми JSON ПОВИННІ збігатися з відповідними іменами файлів XSD ST.96 або іменами компонентів, але з дотриманням правил, визначених у Розділі 5.2 "Угоди про імена JSON цього стандарту", і використовувати суфікс ".json" замість ".xsd" у випадку імен файлів.

ПЕРЕТВОРЕННЯ ПРОСТОРІВ ІМЕН

Простори імен XSD не повинні зберігатися в схемах JSON. Слід зазначити, що функція власного простору імен не підтримується у схемі JSON 2022-12. Якщо в майбутньому ця можливість буде підтримуватися стандартом схеми JSON, буде переглянуто і перетворення простору імен XSD. Однак, емуляцію простору імен можна виконати тільки в тому випадку, якщо схеми JSON зберігаються в підпапках для кожного домену IB. Структура стандарту BOIB ST.96 передбачає зберігання кожного домену IB в окремих підпапках.

[TR-02] Для емуляції простору імен XSD схеми JSON ПОВИННІ зберігатися в підпапках для кожного домену IB і Common (загальна), тобто Common, Copyright (авторське право), Design (промисловий зразок), GeographicalIndication (географічне зазначення), Patent (патент), Trademark (торговельна марка), та ExternalStandards (додаткові стандарти), так само як відповідні XSD зберігаються в стандарті BOIB ST.96.

ПЕРЕТВОРЕННЯ ВБУДОВАНИХ ТИПІВ ДАНИХ

Згідно з JSC-09, для властивості ПОВИНЕН використовуватися найбільш специфічний базовий тип JSON, який є релевантним. Наприклад, якщо значення властивості буде цілим числом, слід використовувати тип "integer", а не більш загальний "number" або більш розширювальний "string".

Таблиця 2 представляє однозначне зіставлення типу даних XSD з типом даних JSON (вбудованим або призначеним для користувача/вбудованим або в окремому файлі), які використовуються для перетворення XSD ST.96 у схему JSON.

[TR-03] Типи даних XSD, наведені в наступній таблиці, ПОВИННІ бути перетворені у відповідні типи даних схеми JSON, як визначено в Таблиці 2.

Таблиця 2: Перетворення простих типів даних XSD

XSD-тип даних	Тип даних схеми JSON або визначення типу схеми JSON
xsd:string	"type": "string"
Token	"type": "string"
xsd:integer	"type": "integer"
xsd:float xsd:double xsd:decimal	"type": "number"
xsd:Boolean	"type": "boolean"
xsd:positiveInteger	"type": "integer", "minimum": 0, "exclusiveMinimum": true
xsd:negativeInteger	"type": "integer", "maximum": 0, "exclusiveMaximum": true
xsd:nonPositiveInteger	"type": "integer", "maximum": 0, "exclusiveMaximum": false
xsd:nonNegativeInteger	"type": "integer", "minimum": 0, "exclusiveMinimum": false
xsd:date, xsd:dateTime, xsd:time	"type": "string", "format": "date-time"
gYearMonth	Не існує вбудованих типів. Визначено відповідний файл "gYearMonth.json". <pre>"gYearMonth": { "anyOf": [{ "type": "object", "properties": { "year": { "type": "integer" }, "month": { "type": "integer", "minimum": 1, "maximum": 12 }, "timezone": { "type": "integer", "minimum": -1440, "maximum": 1439 } } }] }</pre>

	} }
gYear	Не існує вбудованих типів. Визначено відповідний файл "gYear.json". <pre> "gYear": { "anyOf": [{ "type": "object", "properties": { "year": { "type": "integer" }, "timezone": { "type": "integer", "minimum": -1440, "maximum": 1439 } } }] } </pre>
xsd:anyURI	{ "type": "string", "format": "uri" }

ВИЗНАЧЕННЯ XSD

Щоб перетворити визначення XSD, необхідно додати властивість об'єкта "\$schema" як обгортку для визначення поточного компонента. Для атрибутів кореневого елемента схеми XML "xsd:schema" немає еквівалентних визначень у схемі JSON.

[TR-04] Елемент "xsd:schema" і значення його атрибутів ПОВИННІ бути перетворені у відповідні властивості JSON з вихідних XSD, як зазначено нижче.

XSD	JSON	Примітка
xsd:schema	"\$schema"	
@xmlns	N/A	ігнорується
@targetNamespace	N/A	ігнорується
@elementFormDefault	N/A	ігнорується
@attributeFormDefault	N/A	ігнорується
@version	"description"	Об'єднано зі значенням опису з міткою "Version: " в описі. Див. JSD-10 і TR-20.

Деякі приклади наведено нижче:

Визначення XML-схеми компонента (AbstractNumber.xsd)
<pre> <?xml version="1.0" encoding="UTF-8"?> <xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common" xmlns:xsd="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common" elementFormDefault="qualified" attributeFormDefault="qualified" version="V5_0"> <xsd:element name="AbstractNumber" type="xsd:string"> <xsd:annotation> <xsd:documentation>Number assigned to an abstract published without the full document in a collection of abstracts. This collection can be a journal, conference proceedings, a patent collection of abstracts (e.g. Soviet Patent Abstracts), etc.</xsd:documentation> </xsd:annotation> </xsd:element> </xsd:schema> </pre>

Визначення схеми компонента JSON (abstractNumber.json)

```
{
  "$id" : "abstractNumber.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "additionalProperties" : false,
  "properties" : {
    "abstractNumber" : {
      "$ref" : "#/$defs/abstractNumber"
    }
  },
  "required" : [ "abstractNumber" ],
  "$defs" : {
    "abstractNumber" : {
      "type" : "string",
      "description" : "Description: Number assigned to an abstract published without the full document in a collection of abstracts. This collection can be a journal, conference proceedings, a patent collection of abstracts (e.g. Soviet Patent Abstracts), etc.; Version: V5_0"
    }
  }
}
```

Визначення XML-схеми типу компонента (AdditionalRemarkType.xsd)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
  elementFormDefault="qualified" attributeFormDefault="qualified"
  version="V5_0">
  <xsd:include schemaLocation="P.xsd"/>
  <xsd:include schemaLocation="languageCode.xsd"/>
  <xsd:complexType name="AdditionalRemarkType">
    <xsd:sequence>
      <xsd:element ref="com:P"/>
    </xsd:sequence>
    <xsd:attribute ref="com:languageCode"/>
  </xsd:complexType>
</xsd:schema>
```

Визначення XML-схеми типу компонента (additionalRemarkType.json)

```
{
  "$id" : "additionalRemarkType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "additionalRemarkType" : {
      "description" : "Version: V5_0",
      "type" : "object",
      "additionalProperties" : false,
      "properties" : {
        "languageCode" : {
          "$ref" : "languageCode.json#/$defs/languageCode"
        },
        "p" : {
          "$ref" : "p.json#/$defs/p"
        }
      }
    },
    "required" : [ "p" ]
  }
}
```

```

    }
  }
}

```

Визначення XML-схеми компонента на рівні документа (DesignApplication_V5_0.xsd)

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:dgn="http://www.wipo.int/standards/XMLSchema/ST96/Design"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Design"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
  <xsd:annotation>
    <xsd:appinfo>
      <com:SchemaCreatedDate>2012-07-13</com:SchemaCreatedDate>
      <com:SchemaLastModifiedDate>2021-10-01</com:SchemaLastModifiedDate>
      <com:SchemaContactPoint>xml.standards@wipo.int</com:SchemaContactPoint>
      <com:SchemaReleaseNoteURL>http://www.wipo.int/standards/XMLSchema/ST96/V5_0/ReleaseNotes.pdf</com:SchemaReleaseNoteURL>
    </xsd:appinfo>
  </xsd:annotation>
<xsd:include schemaLocation="DesignApplicationType_V5_0.xsd"/>
  <xsd:element name="DesignApplication" type="dgn:DesignApplicationType">
    <xsd:annotation>
      <xsd:documentation>Details on a design application</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:schema>

```

Визначення схеми JSON компонента рівня документа (designApplication_V5_0.json)

```

{
  "$id" : "designApplication_V5_0.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema", "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "designApplication" : {
      "$ref" : "#/$defs/designApplication"
    }
  },
  "required" : [ "designApplication" ],
  "$defs" : {
    "designApplication" : {
      "$ref" : "designApplicationType_V5_0.json#/$defs/designApplicationType",
      "description" : "Description: Details on a design application; Version: V5_0;
SchemaCreatedDate: 2012-07-13; SchemaLastModifiedDate: 2021-10-01;
SchemaContactPoint: xml.standards@wipo.int; SchemaReleaseNoteURL:
http://www.wipo.int/standards/XMLSchema/ST96/V5_0/ReleaseNotes.pdf"
    }
  }
}

```

Визначення XML-схеми типу компонента на рівні документа

(DesignApplicationType_V5_0.xsd)

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:dgn="http://www.wipo.int/standards/XMLSchema/ST96/Design"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Design"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/RequestSoftware.xsd"/>
  <xsd:include schemaLocation="../../DesignApplicationFormName.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/RequestExamination.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/RegistrationOfficeCode.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/ReceivingOfficeCode.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/ReceivingOfficeDate.xsd"/>
  <xsd:include schemaLocation="../../ReceiptNumber.xsd"/>
  <xsd:include schemaLocation="../../SealedDepositIndicator.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/ApplicationNumber.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/FilingPlace.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/ApplicantFileReference.xsd"/>
  <xsd:include schemaLocation="../../DesignApplicationLanguageCode.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/SecondLanguageCode.xsd"/>
  <xsd:include schemaLocation="../../DesignTotalQuantity.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/CorrespondenceLanguageCode.xsd"/>
  <xsd:include
schemaLocation="../../DesignApplicationCurrentStatusCategory.xsd"/>
  <xsd:include schemaLocation="../../DesignApplicationCurrentStatusDate.xsd"/>
  <xsd:include schemaLocation="../../DesignatedCountryBag.xsd"/>
  <xsd:include schemaLocation="../../DesignBag.xsd"/>
  <xsd:include schemaLocation="../../ApplicantBag.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/RepresentativeBag.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/Authorization.xsd"/>
  <xsd:include schemaLocation="../../DesignApplicationEventBag.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"

```

```

schemaLocation="../../../Common/DocumentIncludedBag.xsd"/>
  <xsd:include schemaLocation="../../DesignApplicationStatementBag.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/PaymentBag.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/ReimbursementBag.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/SignatureBag.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/ApplicationDate.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/ApplicationDateTime.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/CorrespondenceAddress.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/BusinessEntityStatusCategory.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../../../Common/InternationalRegistrationNumber.xsd"/>

  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common
" schemaLocation="../../../Common/operationCategory.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common
" schemaLocation="../../../Common/st96Version.xsd"/>
  <xsd:import
namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common
" schemaLocation="../../../Common/ipoVersion.xsd"/>
  <xsd:complexType name="DesignApplicationType">
    <xsd:sequence>
      <xsd:element ref="com:RequestSoftware" minOccurs="0"/>
      <xsd:element ref="dgn:DesignApplicationFormName" minOccurs="0"/>
      <xsd:element ref="com:RequestExamination" minOccurs="0"/>
      <xsd:element ref="com:RegistrationOfficeCode"/>
      <xsd:element ref="com:ReceivingOfficeCode" minOccurs="0"/>
      <xsd:element ref="com:ReceivingOfficeDate" minOccurs="0"/>
      <xsd:element ref="dgn:ReceiptNumber" minOccurs="0"/>
      <xsd:element ref="dgn:SealedDepositIndicator" minOccurs="0"/>
      <xsd:element ref="com:ApplicationNumber" minOccurs="0"/>
      <xsd:element ref="com:InternationalRegistrationNumber"
minOccurs="0"/>
      <xsd:element ref="com:FilingPlace" minOccurs="0"/>
      <xsd:element ref="com:ApplicantFileReference" minOccurs="0"/>
      <xsd:element ref="dgn:DesignApplicationLanguageCode" minOccurs="0"/>
      <xsd:element ref="com:SecondLanguageCode" minOccurs="0"/>
      <xsd:element ref="dgn:DesignTotalQuantity" minOccurs="0"/>
      <xsd:element ref="com:CorrespondenceLanguageCode" minOccurs="0"/>
      <xsd:element ref="dgn:DesignApplicationCurrentStatusCategory"
minOccurs="0"/>
      <xsd:element ref="dgn:DesignApplicationCurrentStatusDate"
minOccurs="0"/>

```

```

<xsd:element ref="dgn:DesignatedCountryBag" minOccurs="0"/>
<xsd:element ref="dgn:DesignBag"/>
<xsd:element ref="dgn:ApplicantBag"/>
<xsd:element ref="com:RepresentativeBag" minOccurs="0"/>
<xsd:element ref="com:Authorization" minOccurs="0"/>
<xsd:element ref="dgn:DesignApplicationEventBag" minOccurs="0"/>
<xsd:element ref="com:DocumentIncludedBag" minOccurs="0"/>
<xsd:element ref="dgn:DesignApplicationStatementBag" minOccurs="0"/>
<xsd:element ref="com:PaymentBag" minOccurs="0"/>
<xsd:element ref="com:ReimbursementBag" minOccurs="0"/>
<xsd:element ref="com:SignatureBag" minOccurs="0"/>
<xsd:choice>
  <xsd:element ref="com:ApplicationDate" minOccurs="0"/>
  <xsd:element ref="com:ApplicationDateTime"/>
</xsd:choice>
<xsd:element ref="com:CorrespondenceAddress" minOccurs="0"/>
<xsd:element ref="com:BusinessEntityStatusCategory" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute ref="com:operationCategory"/>
<xsd:attribute ref="com:st96Version" use="required"/>
<xsd:attribute ref="com:ipoVersion"/>
</xsd:complexType>
</xsd:schema>

```

Визначення JSON схеми типу компонента на рівні документа (designApplicationType_V5_0.json)

```

{
  "$id" : "designApplicationType_V5_0.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema", "$defs" : {
    "designApplicationType" : {
      "description" : "Version: V5_0", "type" : "object",
      "additionalProperties" : false, "properties" : {
        "operationCategory" : {
          "$ref" : "../../Common/operationCategory.json#/$defs/operationCategory"
        },
        "st96Version" : {
          "$ref" : "../../Common/st96Version.json#/$defs/st96Version"
        }
      },
      "ipoVersion" : {
        "$ref" : "../../Common/ipoVersion.json#/$defs/ipoVersion"
      },
      "requestSoftware" : {
        "$ref" : "../../Common/requestSoftware.json#/$defs/requestSoftware"
      },
      "designApplicationFormName" : {
        "$ref" : "../designApplicationFormName.json#/$defs/designApplicationFormName"
      },
      "requestExamination" : {
        "$ref" : "../../Common/requestExamination.json#/$defs/requestExamination"
      }
    }
  }
}

```



```

    "registrationOffice
      Code" : {"$ref" :
"../../Common/registrationOfficeCode.json#/$defs/registrationOfficeCode"
    },
    "receivingOfficeCode" : {
      "$ref" :
"../../Common/receivingOfficeCode.json#/$defs/receivingOfficeCode"
    },
    "receivingOfficeDate" : {
      "$ref" :
"../../Common/receivingOfficeDate.json#/$defs/receivingOfficeDate"
    },
    "receiptNumber" : {
      "$ref" : "../receiptNumber.json#/$defs/receiptNumber"
    },
    "sealedDepositIndicator" : {
      "$ref" :
"../sealedDepositIndicator.json#/$defs/sealedDepositIndicator"
    },
    "applicationNumber" : {
      "$ref" :
"../../Common/applicationNumber.json#/$defs/applicationNumber"
    },
    "internationalRegistrationNumber" :
      {"$ref" :
"../../Common/internationalRegistrationNumber.json#/$defs/internationalRegi
strationNumber"
    },
    "filingPlace" : {
      "$ref" : " ../../Common/filingPlace.json#/$defs/filingPlace"
    },
    "applicantFileReference" : {
      "$ref" :
"../../Common/applicantFileReference.json#/$defs/applicantFileReference"
    },
    "designApplicationLanguageCode" : {"$ref" :
"../designApplicationLanguageCode.json#/$defs/designApplicationLanguageCode
"
    },
    "secondLanguageCode" : {
      "$ref" :
"../../Common/secondLanguageCode.json#/$defs/secondLanguageCode"
    },
    "designTotalQuantity" : {
      "$ref" : "../designTotalQuantity.json#/$defs/designTotalQuantity"
    },
    "correspondenceLanguage
      Code" : {"$ref" :
"../../Common/correspondenceLanguageCode.json#/$defs/correspondenceLanguage
Code"
    },
    "designApplicationCurrentStatusCategory" : {"$ref" :
"../designApplicationCurrentStatusCategory.json#/$defs/designApplicati
onCurrentStatusCategory"
    },
    "designApplicationCurrentStatusDate" : {
      "$ref" :
"../designApplicationCurrentStatusDate.json#/$defs/designApplicationCurrent
StatusDate"
    },

```

```

    "designatedCountryBag" : {
      "$ref" : "../designatedCountryBag.json#/$defs/designatedCountryBag"
    },
    "designBag" : {
      "$ref" : "../designBag.json#/$defs/designBag"
    }
  },
  "applicantBag" : {
    "$ref" : "../applicantBag.json#/$defs/applicantBag"
  },
  "representativeBag" : {
    "$ref" :
      "../../Common/representativeBag.json#/$defs/representativeBag"
  },
  "authorization" : {
    "$ref" : "../../Common/authorization.json#/$defs/authorization"
  },
  "designApplicationEventBag" : {
    "$ref" :
      "../designApplicationEventBag.json#/$defs/designApplicationEventBag"
  },
  "documentIncludedBag" : {
    "$ref" :
      "../../Common/documentIncludedBag.json#/$defs/documentIncludedBag"
  },
  "designApplicationStatementBag"
    : { "$ref" :
      "../designApplicationStatementBag.json#/$defs/designApplicationStatementBag"
    },
  "paymentBag" : {
    "$ref" : "../../Common/paymentBag.json#/$defs/paymentBag"
  },
  "reimbursementBag" : {
    "$ref" :
      "../../Common/reimbursementBag.json#/$defs/reimbursementBag"
  },
  "signatureBag" : {
    "$ref" : "../../Common/signatureBag.json#/$defs/signatureBag"
  },
  "applicationDate" : {
    "$ref" : "../../Common/applicationDate.json#/$defs/applicationDate"
  },
  "applicationDateTime" : {
    "$ref": "../../Common/applicationDateTime.json#/$defs/applicationDate
      Time"
  },
  "correspondenceAddress" : {
    "$ref" :
      "../../Common/correspondenceAddress.json#/$defs/correspondenceAdd
      ress"
  },
  "businessEntityStatusCategory" : {
    "$ref" :
      "../../Common/businessEntityStatusCategory.json#/$defs/businessEntityStatusC
      ategory"
  },
  "oneOf" : [ {
    "required" : [ "applicationDate" ]
  }, {
    "required" : [ "applicationDateTime" ]
  } ]

```

```

    } ],
    "required" : [ "st96Version", "registrationOfficeCode", "designBag",
    "applicantBag" ]
  }
}
}

```

ПОСИЛАННЯ НА ГЛОБАЛЬНУ СХЕМУ

[TR-05] Два звичайних способи глобальних визначень, що застосовуються в XSD, тобто `xsd:import` і `xsd:include`, ПОВИННІ бути перетворені за допомогою властивості «\$ref» в схемі JSON незалежно від типу, що використовується в XSD «import» або «include» Наприклад:

Визначення схеми XML для прикладу `xsd:import` (RelatedApplicationDate.xsd)

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:dgn="http://www.wipo.int/standards/XMLSchema/ST96/Design"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Design"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
  <xsd:import namespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
schemaLocation="../Common/DateType.xsd"/>
  <xsd:element name="RelatedApplicationDate" type="com:DateType">
    <xsd:annotation>
      <xsd:documentation>Application date of the
relatedapplication</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:schema>

```

Визначення схеми JSON для прикладу `xsd:import` (relatedApplicationDate.json)

```

{
  "$id" : "relatedApplicationDate.json",
  "$schema" : "https://json-schema.org/draft/2020-
12/schema", "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "relatedApplicationDate" : {
      "$ref" : "#/$defs/relatedApplicationDate"
    }
  },
  "required" : [ "relatedApplicationDate"
], "$defs" : {
    "relatedApplicationDate" : {
      "$ref" : "../Common/dateType.json#/$defs/dateType",
      "description" : "Description: Application date of the related
application;Version: V5_0"
    }
  }
}

```

Приклад визначення XML-схеми для `xsd:include` (AffectedDesign.xsd)

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:dgn="http://www.wipo.int/standards/XMLSchema/ST96/Design"

```

```

xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Design"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
  <xsd:include schemaLocation="AffectedDesignType.xsd"/>
  <xsd:element name="AffectedDesign" type="dgn:AffectedDesignType">
    <xsd:annotation>
      <xsd:documentation>Design affected by the decision, either all
designs or enumeration</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:schema>

```

Визначення схеми JSON для прикладу xsd:include (actedDesign.json)

```

{
  "$id" : "affectedDesign.json",
  "$schema" : "https://json-schema.org/draft/2020-
12/schema", "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "affectedDesign" : {
      "$ref" : "#/$defs/affectedDesign"
    }
  },
  "required" : [ "affectedDesign" ],
  "$defs" : {
    "affectedDesign" : {
      "$ref" : "affectedDesignType.json#/$defs/affectedDesignType",
      "description" : "Description: Design affected by the decision,
either all designs or enumeration; Version: V5_0"
    }
  }
}

```

ФОРМУВАЛЬНИКИ XSD

Формувальники – це структурні компоненти схеми W3C, що угруповують оголошення елементів. Стандарт схеми W3C передбачає три типи формувальників, а саме, *sequence*, *choice* та *all*. Не допускається для використання `xsd:all`, відповідно до [SD-52] Додатку I до стандарту VOIB ST.96.

[TR-06] Властивості об'єкта або масиву до поточного визначення компонента. ПОВИННА бути додана відповідно до значення «maxOccurs», як вказано нижче.

XSD	JSON	Примітки
<code>xsd:sequence</code>	<code>object</code>	
<code>maxOccurs=1</code>	<code>object</code>	<code>minOccurs=1</code> зробить об'єкт обов'язковим
<code>maxOccurs= unbounded</code>	<code>object</code> або <code>array</code>	<code>minOccurs=1</code> зробить об'єкт обов'язковим
<code>xsd:choice</code>	<code>object</code>	
<code>maxOccurs=1</code>	<code>object</code>	<code>minOccurs=1</code> зробить об'єкт обов'язковим
<code>maxOccurs= unbounded</code>	<code>object</code> або <code>array</code>	<code>minOccurs=1</code> зробить об'єкт обов'язковим

Наприклад:

Приклад визначення схеми XML для прикладу `xsd:sequence` з `maxOccurs=1` (AdditionalRemarkType.xsd)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
<xsd:include schemaLocation="P.xsd"/>
<xsd:include schemaLocation="languageCode.xsd"/>
  <xsd:complexType name="AdditionalRemarkType">
    <xsd:sequence>
      <xsd:element ref="com:P"/>
    </xsd:sequence>
    <xsd:attribute ref="com:languageCode"/>
  </xsd:complexType>
</xsd:schema>
```

Приклад визначення схеми JSON для `xsd:sequence` з `maxOccurs=1` (`additionalRemarkType.json`)

```
{
  "$id" : "additionalRemarkType.json",
  "$schema" : "https://json-schema.org/draft/2020-
12/schema", "$defs" : {
    "additionalRemarkType" : {
      "description" : "Version: V5_0",
      "type" : "object",
      "additionalProperties" : false,
      "properties" : {
        "languageCode" : {
          "$ref" : "languageCode.json#/$defs/languageCode"
        },
        "p" : {
          "$ref" : "p.json#/$defs/p"
        }
      },
      "required" : [ "p" ]
    }
  }
}
```

Приклад визначення XML-схеми для `xsd:sequence` з `maxOccurs=unbounded` (InventionClaimBagType.xsd)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:pat="http://www.wipo.int/standards/XMLSchema/ST96/Patent"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Patent"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
<xsd:include schemaLocation="InventionNumber.xsd"/>
<xsd:include schemaLocation="ClaimNumber.xsd"/>
<xsd:include schemaLocation="ClaimNumberRange.xsd"/>
  <xsd:complexType name="InventionClaimBagType">
```

```

<xsd:sequence maxOccurs="unbounded">
  <xsd:element ref="pat:InventionNumber"/>
  <xsd:choice maxOccurs="unbounded">
    <xsd:element ref="pat:ClaimNumber"/>
    <xsd:element ref="pat:ClaimNumberRange"/>
  </xsd:choice>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>

```

Визначення схеми JSON для прикладу xsd:sequence з maxOccurs=unbounded (inventionClaimBagType.json)

```

{
  "$id" : "inventionClaimBagType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "inventionClaimBagType" : {
      "description" : "Version: V5_0",
      "type" : "object",
      "additionalProperties" : false,
      "properties" : {
        "inventionNumber" : {
          "type" : "array",
          "minItems" : 1,
          "items" : {
            "$ref" : "inventionNumber.json#/$defs/inventionNumber"
          }
        },
        "claimNumber" : {
          "anyOf" : [ {
            "$ref" : "claimNumber.json#/$defs/claimNumber"
          }, {
            "type" : "array",
            "minItems" : 1,
            "items" : {
              "$ref" : "claimNumber.json#/$defs/claimNumber"
            }
          } ]
        },
        "claimNumberRange" : {
          "anyOf" : [ {
            "$ref" : "claimNumberRange.json#/$defs/claimNumberRange"
          }, {
            "type" : "array",
            "minItems" : 1,
            "items" : {
              "$ref" : "claimNumberRange.json#/$defs/claimNumberRange"
            }
          } ]
        }
      ],
      "anyOf" : [ {
        "required" : [ "claimNumber" ]
      }, {
        "required" : [ "claimNumberRange" ]
      } ],
      "required" : [ "inventionNumber" ]
    }
  }
}

```

Визначення схеми XML для прикладу `xsd:choice` з `maxOccurs=1` (`ChemicalFormulaeType.xsd`)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
<xsd:include schemaLocation="Image.xsd"/>
<xsd:include schemaLocation="InlineFormula.xsd"/>
<xsd:include schemaLocation="id.xsd"/>
<xsd:include schemaLocation="chemicalFormulaeNumber.xsd"/>
<xsd:include schemaLocation="ExternalDocumentBag.xsd"/>
  <xsd:complexType name="ChemicalFormulaeType">
    <xsd:choice>
      <xsd:element ref="com:Image"/>
      <xsd:element ref="com:InlineFormula"/>
      <xsd:element ref="com:ExternalDocumentBag"/>
    </xsd:choice>
    <xsd:attribute ref="com:id"/>
    <xsd:attribute ref="com:chemicalFormulaeNumber"/>
  </xsd:complexType>
</xsd:schema>
</xsd:схема>
```

Визначення схеми JSON для прикладу `xsd:choice` з `maxOccurs=1` (`chemicalFormulaeType.json`)

```
{
  "$id" : "chemicalFormulaeType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "chemicalFormulaeType" : {
      "description" : "Version: V5_0",
      "type" : "object",
      "additionalProperties" : false,
      "properties" : {
        "id" : {
          "$ref" : "id.json#/$defs/id"
        },
        "chemicalFormulaeNumber" : {
          "$ref" : "chemicalFormulaeNumber.json#/$defs/chemicalFormulaeNumber"
        },
        "image" : {
          "$ref" : "image.json#/$defs/image"
        },
        "inlineFormula" : {
          "$ref" : "inlineFormula.json#/$defs/inlineFormula"
        },
        "externalDocumentBag" : {
          "$ref" : "externalDocumentBag.json#/$defs/externalDocumentBag"
        }
      },
      "oneOf" : [ {
        "required" : [ "image" ]
      }
    ]
  }
}
```

```

    }, {
      "required" : [ "inlineFormula" ]
    }, {
      "required" : [ "externalDocumentBag" ]
    } ]
  }
}
}
}

```

Приклад визначення XML-схеми для xsd:choice з maxOccurs=unbounded (InventionClaimBagType.xsd)

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"

  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
  elementFormDefault="qualified" attributeFormDefault="qualified"
  version="V5_0">
  <xsd:include schemaLocation="Heading.xsd"/>
  <xsd:include schemaLocation="P.xsd"/>
  <xsd:include schemaLocation="id.xsd"/>
  <xsd:complexType name="ContentType">
    <xsd:choice maxOccurs="unbounded">
  <xsd:element ref="com:Heading"/>
  <xsd:element ref="com:P"/>
    </xsd:choice>
    <xsd:attribute ref="com:id"/>
  </xsd:complexType>
</xsd:schema>

```

Визначення схеми JSON для прикладу xsd:choice з maxOccurs=unbounded (inventionClaimBagType.json)

```

{
  "$id" : "contentType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "contentType" : {
      "description" : "Version: V5_0",
      "type" : "object",
      "additionalProperties" : false,
      "properties" : {
        "id" : {
          "$ref" : "id.json#/$defs/id"
        },
        "heading" : {
          "anyOf" : [ {
            "$ref" : "heading.json#/$defs/heading"
          }, {
            "type" : "array",
            "minItems" : 1,
            "items" : {
              "$ref" : "heading.json#/$defs/heading"
            }
          } ]
        }
      }
    },
    "p" : {
      "anyOf" : [ {

```



```

    "$ref" : "p.json#/$defs/p"
  }, {
    "type" : "array",
    "minItems" : 1,
    "items" : {
      "$ref" : "p.json#/$defs/p"
    }
  } ]
}
},
"anyOf" : [ {
  "required" : [ "heading" ]
}, {
  "required" : [ "p" ]
} ]
}
}

```

ЕЛЕМЕНТИ

Елементи є основними будівельними блоками XML і мають бути перетворені у властивості в JSON.

[TR-07] Властивість відповідного типу **ПОВИННА** бути додана до поточного визначення компонента відповідно до значення `maxOccurs`, як зазначено нижче.

XSD	JSON	Примітки
<code>xsd:element</code>	Property (властивість)	Див. вище розділ перетворення вбудованих типів даних
<code>maxOccurs=1</code>	Властивість з базовим типом з XSD або створений тип	<code>minOccurs=1</code> зробить властивість обов'язковою
<code>maxOccurs= unbounded</code>	Масив базових типів або створений тип	<code>minOccurs=1</code> зробить властивість обов'язковою

Наприклад:

Визначення схеми XML для прикладу `xsd:element maxOccurs=1` (DocumentTotalQuantity.xsd)

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
  <xsd:element name="DocumentTotalQuantity" type="xsd:nonNegativeInteger">
    <xsd:annotation>
      <xsd:documentation>Total number of documents available or provided.
</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:schema>

```

Визначення схеми JSON для прикладу `xsd:element maxOccurs=1` (documentTotalQuantity.json)

```

{
  "$id" : "documentTotalQuantity.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema", "additionalProperties" : false,
  "properties" : {
    "documentTotalQuantity" :
    {
      "$ref" : "#/$defs/documentTotalQuantity"
    }
  },
  "required" : [
  "documentTotalQuantity" ], "$defs" :
  {
    "documentTotalQuantity" : {
      "type" : "integer",
      "minimum" : 0,
      "description" : "Description: Total number of documents available or provided. ;Version: V5_0"
    }
  }
}

```

Приклад визначення XML-схеми для xsd:element maxOccurs=unbounded (IPOfficeCodeBagType.xsd)

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
<xsd:include schemaLocation="IPOfficeCode.xsd"/>
  <xsd:complexType name="IPOfficeCodeBagType">
    <xsd:sequence>
      <xsd:element ref="com:IPOfficeCode" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>

```

Визначення схеми JSON для прикладу xsd:element maxOccurs=unbounded (ipOfficeCodeBagType.json)

```

{
  "$id" : "ipOfficeCodeBagType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "ipOfficeCodeBagType" : {
      "description" : "Version: V5_0",
      "type" : "object",
      "additionalProperties" : false,
      "properties" : {
        "ipOfficeCode" : {
          "type" : "array",
          "minItems" : 1,
          "items" : {
            "$ref" : "ipOfficeCode.json#/$defs/ipOfficeCode"
          }
        }
      }
    }
  },
}

```

```

    "required" : [ "ipOfficeCode" ]
  }
}

```

АТРИБУТИ

Атрибути – це конструкції схеми W3C, пов'язані з елементами, які надають додаткову інформацію про елементи і мають бути перетворені як властивість поточного компонента в JSON.

[TR-08] Властивість відповідного типу **ПОВИННА** бути додана до поточного визначення компонента, як зазначено нижче:

XSD	JSON	Примітки
xsd:attribute	Властивість поточного компонента	Див. розділ "Перетворення вбудованих типів даних"
xsd:annotation	"\$description"	Див. розділ "Анотації"

Наприклад:

Визначення XML-схеми для прикладу `xsd:attribute` (`changeDateTime.xsd`)

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
<xsd:attribute name="changeDateTime" type="xsd:dateTime">
  <xsd:annotation>
    <xsd:documentation>Date and time of change</xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:schema>

```

Визначення схеми JSON для прикладу `xsd:attribute` (`changeDateTime.json`)

```

{
  "$id" : "changeDateTime.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "additionalProperties" : false,
  "properties" : {
    "changeDateTime" : {
      "$ref" : "#/$defs/changeDateTime"
    }
  },
  "required" : [ "changeDateTime" ],
  "$defs" : {
    "changeDateTime" : {
      "format" : "date-time",
      "type" : "string",
      "description" : "Description: Date and time of change; Version: V5 0"
    }
  }
}

```

ПРОСТИЙ ТИП

[TR-09] Властивість відповідного типу **СЛІД** додати до поточного визначення компонента, як зазначено нижче.

XSD	JSON	Примітки
xsd:simpleType	object	Див. розділ "Перетворення вбудованих типів даних".
xsd:union	object - "anyOf"	Див. розділ "Об'єднання"
xsd:restriction	object	Див. розділ "Обмеження"
xsd:enumeration	enum	Див. розділ "Перерахування"

КОМПЛЕКСНИЙ ТИП

[TR-10] Властивість відповідного типу СЛІД додати до поточного визначення компонента, як зазначено нижче.

XSD	JSON	Примітки
Xsd:complexType	object	Див. розділ "Перетворення вбудованих типів даних".
xsd:simpleContent	object	Див. розділ "Простий контент"
xsd:complexContent	object	Див. розділ "Складний вміст"
xsd:sequence	object	Див. розділ "Формувальники XSD".
xsd:choice	object/array	Див. розділ "Формувальники XSD".
@mixed	object	Див. розділ "Змішаний вміст"

Простий контент

[TR-11] Властивість відповідного типу СЛІД додати до поточного визначення компонента, як зазначено нижче.

XSD	JSON	Примітки
xsd:simpleContent	object	
xsd:extension	object	Див. розділ "Розширення"
xsd:restriction	object	Див. розділ "Обмеження"

Комплексний контент

[TR-12] Властивість відповідного типу СЛІД додати до поточного визначення компонента, як зазначено нижче.

XSD	JSON	Примітки
xsd:complexContent	object	
xsd:extension	object	Див. розділ "Розширення"
xsd:restriction	object	Див. розділ "Обмеження"

Змішаний контент

[TR-13] Властивість відповідного типу СЛІД додати до поточного визначення компонента, як зазначено нижче.

XSD	JSON	Примітки
xsd:complexType	object	
@mixed	object	

Наприклад :

**Приклад визначення схеми XML для `xsd:complexType@mixed=true`
(CrossReferenceType.xsd)**

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
<xsd:include schemaLocation="PhraseType.xsd"/>
<xsd:include schemaLocation="id.xsd"/>
<xsd:include schemaLocation="idrefs.xsd"/>
<xsd:include schemaLocation="extRef.xsd"/>
<xsd:include schemaLocation="crossReferenceCategory.xsd"/>
<xsd:include schemaLocation="sourceURI.xsd"/>
<xsd:include schemaLocation="sourceSystemName.xsd"/>
<xsd:include schemaLocation="sourceSystemIdentifier.xsd"/>
  <xsd:complexType name="CrossReferenceType" mixed="true">
    <xsd:complexContent>
      <xsd:extension
        base="com:PhraseType">
        <xsd:attribute ref="com:id"/>
        <xsd:attribute ref="com:idrefs"/>
        <xsd:attribute ref="com:extRef"/>
        <xsd:attribute ref="com:crossReferenceCategory" use="required"/>
        <xsd:attribute ref="com:sourceURI"/>
        <xsd:attribute ref="com:sourceSystemName"/>
        <xsd:attribute ref="com:sourceSystemIdentifier"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:schema>

```

Приклад визначення схеми JSON для xsd:complexType@mixed=true (crossReferenceType.xsd)

```

{
  "$id" : "crossReferenceType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "crossReferenceType" : {
      "description" : "Version: V5_0",
      "type" : "object",
      "additionalProperties" : false,
      "properties" : {
        "phraseType" : {
          "$ref" : "phraseType.json#/$defs/phraseType"
        },
        "id" : {
          "$ref" : "id.json#/$defs/id"
        },
        "idrefs" : {
          "$ref" : "idrefs.json#/$defs/idrefs"
        },
        "extRef" : {
          "$ref" : "extRef.json#/$defs/extRef"
        },
        "crossReferenceCategory" : {
          "$ref" : "crossReferenceCategory.json#/$defs/crossReferenceCategory"
        },
        "sourceURI" : {
          "$ref" : "sourceURI.json#/$defs/sourceURI"
        }
      }
    }
  }
}

```

```

    "sourceSystemName" : {
      "$ref" : "sourceSystemName.json#/$defs/sourceSystemName"
    },
    "sourceSystemIdentifier" : {
      "$ref" : "sourceSystemIdentifier.json#/$defs/sourceSystemIdentifier"
    }
  },
  "required" : [ "crossReferenceCategory" ]
}
}
}

```

АННОТАЦІЯ

У стандарті VOIB ST.96 використовуються два типи анотацій, а саме: `xsd:appinfo` та `xsd:documentation`.

xsd:appinfo

Елемент `xsd:appinfo` визначає інформацію, яку має використовувати додаток. Цей елемент повинен знаходитися всередині елемента анотації. Стандарт VOIB ST.96 використовує `xsd:appinfo` в усіх XSD на рівні документа, що містить такі елементи схеми:

- `com:SchemaCreatedDate`
- `com:SchemaLastModifiedDate`
- `com:SchemaContactPoint`
- `com:SchemaReleaseNoteURL`

[TR-14] Інформацію, надану в `<xsd:appinfo>`, СЛІД перетворити на значення "description" у схемі JSON.

Наприклад:

Компонент визначення XML-схеми (ApplicationBodyType.xsd)

```

<xsd:annotation>
  <xsd:appinfo>
    <com:SchemaCreatedDate>2012-07-13</com:SchemaCreatedDate>
    <com:SchemaLastModifiedDate>2021-10-
01</com:SchemaLastModifiedDate>

    <com:SchemaContactPoint>xml.standards@wipo.int</com:SchemaContactPoint
>

    <com:SchemaReleaseNoteURL>http://www.wipo.int/standards/XMLSchema/ST96/
V5_0/ReleaseNotes.pdf</com:SchemaReleaseNoteURL>
  </xsd:appinfo>
</xsd:annotation>
<xsd:include schemaLocation="ApplicationBodyType_V5_0.xsd"/>
<xsd:element name="ApplicationBody" type="pat:ApplicationBodyType">
  <xsd:annotation>
    <xsd:documentation>Body of a patent
application</xsd:documentation>
  </xsd:annotation>...

```

Визначення схеми JSON для прикладу з `xsd:annotation` (applicationBodyType.json)

```

"$defs" : {
  "applicationBody" : {
    "$ref" :
      "applicationBodyType_V5_0.json#/$defs/applicationBodyType",
    "description" : "Description: Body of a patent application;
Version:
V5_0; SchemaCreatedDate: 2012-07-13; SchemaLastModifiedDate:
2021-10-01; SchemaContactPoint: xml.standards@wipo.int;
SchemaReleaseNoteURL:
http://www.wipo.int/standards/XMLSchema/ST96/V5_0/ReleaseNotes.pdf"
  }
}

```

xsd:documentation

У стандарті VOIB ST.96 визначено інформацію, призначену для читання або використання користувачами в елементі анотації. Як зазначено у стандарті VOIB ST.96:

"[SD-58], всі схеми ПОВИННІ включати документацію щодо побудови схеми з використанням елемента `xsd:documentation`".

[TR-15] Інформацію про документацію та інформацію про версію СЛІД передавати до значення "description" у схемі JSON.

Наприклад:

Компонент визначення XML-схеми (AbstractNumber.xsd)

```

<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
  <xsd:element name="AbstractNumber" type="xsd:string">
    <xsd:annotation>
      <xsd:documentation>Number assigned to an abstract published
without the full document in a collection of abstracts. This collection can
be a journal, conference proceedings, a patent collection of abstracts (e.g.
Soviet Patent Abstracts), etc.</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:schema>

```

Визначення схеми JSON для прикладу `xsd:documentation` (abstractNumber.json):

```

{
  "$id" : "abstractNumber.json",
  "$schema" : "https://json-schema.org/draft/2020-
12/schema", "additionalProperties" : false,
  "properties" : {
    "abstractNumber" : {
      "$ref" : "#/$defs/abstractNumber"
    }
  },
  "required" : [ "abstractNumber" ],
  "$defs" : {
    "abstractNumber" : {
      "type" : "string",
      "description" : "Description: Number assigned to an abstract published
withoutthe full document in a collection of abstracts. This collection can be
a journal, conference proceedings, a patent collection of abstracts (e.g.
Soviet Patent Abstracts), etc.; Version: V5_0"
    }
  }
}

```

}

ОБ'ЄДНАННЯ

[TR-16] Властивість відповідного типу СЛІД додати до поточного визначення компонента, як зазначено в таблиці нижче.

XSD	JSON	Примітки
xsd:union	об'єкт - "anyOf"	Див. розділ "Перетворення вбудованих типів даних".

Наприклад:

Приклад визначення схеми XML для xsd:simpleType\xsd:union (DocumentNameType.xsd)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
<xsd:include schemaLocation="DocumentNameCategoryType.xsd"/>
<xsd:simpleType name="DocumentNameType">
  <xsd:union memberTypes="xsd:string com:DocumentNameCategoryType"/>
</xsd:simpleType>
</xsd:schema>
```

Визначення схеми JSON для прикладу xsd:simpleType\xsd:union (documentNameType.json)

```
{
  "$id" : "documentNameType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "documentNameType" : {
      "description" : "Version: V5_0",
      "anyOf" : [ {
        "type" : "string"
      }, {
        "$ref" :
          "documentNameCategoryType.json#/$defs/documentNameCategoryType"
      } ]
    }
  }
}
```

РОЗШИРЕННЯ

[TR-17] Об'єкт СЛІД додати до поточного визначення компонента, як зазначено в таблиці нижче:

XSD	JSON	Примітки
xsd:extension	об'єкт - "anyOf"	Див. розділ "Перетворення вбудованих типів даних".

Наприклад:

Приклад визначення схеми XML для xsd:extension (AmountType.xsd)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
```



```
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
<xsd:include schemaLocation="currencyCode.xsd"/>
<xsd:complexType name="AmountType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:decimal">
      <xsd:attribute ref="com:currencyCode"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
</xsd:schema>
```

Визначення схеми JSON для прикладу з `xsd:extension` (`amountType.json`)

```
{
  "$id" : "amountType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "amountType" : {
      "description" : "Version: V5_0",
      "type" : "object",
      "additionalProperties" : false,
      "properties" : {
        "$" : {
          "type" : "number"
        },
        "currencyCode" : {
          "$ref" : "currencyCode.json#/$defs/currencyCode"
        }
      }
    }
  }
}
```

ОБМЕЖЕННЯ

[TR-18] Об'єкт СЛІД додати до поточного визначення компонента, як зазначено в таблиці нижче.

XSD	JSON	Примітки
<code>xsd:restriction</code>	<code>object</code>	
<code>xsd:pattern</code>	<code>pattern</code>	Див. розділ Шаблон
<code>xsd:перерахування</code>	<code>enum</code>	Див. розділ Перерахування

ПЕРЕРАХУВАННЯ

[TR-19] Властивість `xsd:enumeration` ПОВИННА бути перетворена на властивість `"enum"` у поточному визначенні компонента схеми JSON.

Визначення схеми XML для прикладу з `xsd:simpleType`\`xsd:restriction`\`xsd:enumeration` (`BusinessEntityStatusCategoryType.xsd`)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
  <xsd:simpleType name="BusinessEntityStatusCategoryType">
    <xsd:restriction base="xsd:token">
```

```

<xsd:enumeration value="Undiscounted">
  <xsd:annotation>
    <xsd:documentation>Undiscounted entity</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="Small">
  <xsd:annotation>
    <xsd:documentation>Small entity discount</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="Micro">
  <xsd:annotation>
    <xsd:documentation>Micro entity discount</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
</xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

Визначення схеми JSON для прикладу з `xsd:simpleType`\`xsd:restriction`\`xsd:enumeration` (businessEntityStatusCategoryType.json)

```

{
  "$id" : "businessEntityStatusCategoryType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema", "$defs" : {
    "businessEntityStatusCategoryType" : {
      "description" : "Version: V5_0; Undiscounted: Undiscounted entity; Small: Small entity discount; Micro: Micro entity discount",
      "type" : "string",
      "enum" : [ "Undiscounted", "Small", "Micro" ]
    }
  }
}

```

ОБМЕЖУВАЛЬНІ АСПЕКТИ

У схемах W3C використовуються обмежувальні аспекти, перелічені в таблиці нижче. Слід зазначити, що `xsd:minInclusive`, `xsd:maxInclusive`, `xsd:minExclusive`, `xsd:maxExclusive` і `xsd:minLength` не використовуються в стандарті BOIB ST.96. Отже, ці невикористані аспекти не представлені в Додатку І.

[TR-20] Обмежувальні аспекти XSD ПОВИННІ бути перетворені у відповідні ключові слова JSON разом із відповідною мінімальною або максимальною довжиною, як зазначено в таблиці нижче.

Таблиця 3 Перетворення обмежувальних аспектів XSD (X - числове значення обмеження)

Обмежувальний аспект XSD	Еквівалент схеми JSON
<code><xsd:minLength value="X" /></code>	{ "minLength": X }
<code><xsd:maxLength value="X" /></code>	{ "maxLength": X }
<code><xsd:length value="X" /></code>	{ "minLength": X, "maxLength": X }

<code><xsd:pattern value="X" /></code>	<code>{ "pattern": "X" }</code>
<code><xsd:minExclusive value="X" /></code>	<code>{ "minimum": X, "exclusiveMinimum": true }</code>
<code><xsd:maxExclusive value="X" /></code>	<code>{ "maximum": X, "exclusiveMaximum": true }</code>
<code><xsd:minInclusive value="X" /></code>	<code>{ "minimum": X, "exclusiveMinimum": false }</code>
<code><xsd:maxInclusive value="X" /></code>	<code>{ "maximum": X, "exclusiveMaximum": false }</code>

Наприклад:

Визначення XML-схеми для прикладу з `xsd:length` (ClassType.xsd)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:pat="http://www.wipo.int/standards/XMLSchema/ST96/Patent"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Patent"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
  <xsd:simpleType name="ClassType">
    <xsd:restriction base="xsd:token">
      <xsd:length value="2"/>
      <xsd:pattern value="[0-9][1-9]|[1-9][0-9]"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```

Визначення схеми JSON для прикладу з `xsd:length` (classType.json)

```
{
  "$id" : "classType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "classType" : {
      "description" : "Version: V5_0",
      "type" : "string",
      "maxLength" : 2,
      "pattern" : "[0-9][1-9]|[1-9][0-9]"
    }
  }
}
```

Шаблон (Pattern)

[TR-21] `xsd:pattern` ПОВИНЕН бути перетворений у властивість "pattern" у поточному визначенні компонента.

Приклад:

Приклад визначення схеми XML для `xsd:pattern` (WIPONotificationNumberType.xsd)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:com="http://www.wipo.int/standards/XMLSchema/ST96/Common"
```

```

xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.wipo.int/standards/XMLSchema/ST96/Common"
elementFormDefault="qualified" attributeFormDefault="qualified"
version="V5_0">
<xsd:simpleType name="WIPONotificationNumberType">
  <xsd:restriction base="xsd:token">
    <xsd:pattern value="[A-Z]{3}[0-9]{6}" />
  </xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

Визначення схеми JSON для прикладу з `xsd:pattern` (wipoNotificationNumberType.json)

```

{
  "$id" : "wipoNotificationNumberType.json",
  "$schema" : "https://json-schema.org/draft/2020-12/schema",
  "$defs" : {
    "wipoNotificationNumberType" : {
      "description" : "Version: V5_0",
      "type" : "string",
      "pattern" : "[A-Z]{3}[0-9]{6}"
    }
  }
}

```

ГРУПА

`xsd:group` не використовується в стандарті VOIB ST.96, але використовується в зовнішніх стандартах, на які посилаються XSD стандарту VOIB ST.96. Додаткова інформація про зовнішні стандарти доступна нижче в розділі "Перетворення зовнішніх залежностей XSD".

[TR-22] Тип об'єкта СЛІД додати до поточного визначення компонента.

XSD	JSON	Примітки
<code>xsd:group</code>	<code>object</code>	

Приклад:

Визначення схеми XML для прикладу з `xsd:group` (фрагмент із FlattenedMathML3.json)

```

<xs:group name="anyElement">
  <xs:choice>
    <xs:any namespace="##other" processContents="skip" />
    <xs:any namespace="##local" processContents="skip" />
  </xs:choice>
</xs:group>

```

Визначення схеми JSON для прикладу з `xsd:group` (фрагмент із flattenedMathML3.json)

```

"anyElement" : {
  "type" : "object",
  "additionalProperties" : false,
  "properties" : {
    "other_SKIP" : {
      "patternProperties" : {
        "^@\\w+$" : {
          "type" : [ "string", "number", "boolean" ]
        }
      }
    },
    "additionalProperties" : false,

```

```

    "type" : [ "object", "string", "number", "boolean" ],
    "properties" : {
      "$" : {
        "type" : [ "string", "number", "boolean" ]
      }
    },
    "local_SKIP" : {
      "patternProperties" : {
        "@\\w+$" : {
          "type" : [ "string", "number", "boolean" ]
        }
      },
      "additionalProperties" : false,
      "type" : [ "object", "string", "number", "boolean" ],
      "properties" : {
        "$" : {
          "type" : [ "string", "number", "boolean" ]
        }
      }
    },
    "oneOf" : [ {
      "required" : [ "other_SKIP" ]
    }, {
      "required" : [ "local_SKIP" ]
    } ]
  }
}

```

ПЕРЕТВОРЕННЯ ЗОВНІШНІХ ЗАЛЕЖНОСТЕЙ XSD

Стандарт BOIB ST.96 посилається на наступні схеми галузевих стандартів замість того, щоб перевизначити їх у стандарті BOIB ST.96:

- MathML версії 3⁶ (FlattenedMathML3.xsd); i
- Схема таблиці OASIS⁷ (OASISTable_V1_0.xsd).

На момент розроблення цього стандарту ці зовнішні стандарти не надають еквівалентних схем JSON. Однак, визнаючи, що схеми JSON на основі стандарту BOIB ST.96 неможливі без еквівалентних схем JSON, як програмне рішення, ці дві зовнішні схеми XSD перетворено на достатньо стабільні еквівалентні схеми JSON з використанням інструменту перетворення, представленого в Доповненні до Додатку I цього Стандарту. Стандартні та перетворені схеми JSON включено до набору схем JSON як Додаток II.

[Доповнення до Додатку I наведено далі]

⁶ <http://www.w3.org/TR/MathML3>

⁷ <http://www.oasisopen.org/docbook/xmlschema/1.0b1/calstbl.xsd>

ДОПОВНЕННЯ

ІНСТРУМЕНТ ПЕРЕТВОРЕННЯ СХЕМИ XSD В JSON

У цьому доповненні до Додатка I міститься інструмент для перетворення схем XML у схеми JSON, що являє собою бібліотеку Java, яка допомагає ВІВ перетворювати наданий XSD на основі стандарту BOIB ST.96 в еквівалент схеми JSON відповідно до правил перетворення, викладених у Додатку I. Цей інструмент перетворення надається тут як частина стандарту, тому ВІВ можуть також використовувати цей інструмент для перетворення власних адаптованих XSD на основі стандарту ST.96 у схему JSON.

Вимоги

Середовище виконання Java 1.8 (або новішої версії)

Використання

Надання довідки щодо опцій

```
C:\> java -jar xsd2JsonSchema.jar -help
```

Перетворення одного файлу (XSD)

```
C:\> java -jar xsd2JsonSchema.jar -f "C:\XSD_Folder\Common\AbstractNumber.xsd"
```

Перетворення одного файлу (XSD) і включених/імпортованих схем

```
C:\> java -jar xsd2JsonSchema.jar -f -r "C:\XSD_Folder_Path\Common\AbstractNumber.xsd"
```

Перетворити всі схеми в наданому каталозі (прапорець рекурсивності недоступний для цієї опції)

```
C:\> java -jar xsd2JsonSchema.jar -d "C:\XSD_Folder_Path\Common"
```

Завантажити виконуваний JAR-файл

Інструмент можна завантажити в Доповненні до Додатка I за наступним посиланням:

https://www.wipo.int/standards/en/docs/annexi_appendix_xsd2jsonschema_tool.zip

[Додаток II наведено далі]

ДОДАТОК II

СХЕМА JSON

*Схвалено Комітетом зі стандартів VOIB (КСВ)
на його десятій сесії
25 листопада 2022 року*

Додаток II містить повний набір схем JSON, які відповідають стандарту VOIB ST.96 XSD, версії XSD 5.0. Ці схеми JSON були автоматично створені за допомогою інструменту перетворення, представленого в Доповненні до Додатка I, відповідно до правил і рекомендацій перетворення, визначених у Додатку I. Слід зазначити, що це односторонній процес перетворення, тобто з XSD у JSON. Цей набір схем JSON включає перетворені схеми JSON зовнішніх стандартів XML, тобто таблиці MathML і OASIS. Однак вихідні імена тегів компонентів XSD зовнішніх стандартів XML збережено без змін.

Схеми JSON можна завантажити в Доповненні до Додатка II за наступним посиланням:
https://www.wipo.int/standards/en/docs/annexii_json_schemas.zip

[Додаток III наведено далі]

ДОДАТОК III

ПРИКЛАДИ JSON

*Схвалено Комітетом зі стандартів ВОІВ (КСВ)
на його десятій сесії
25 листопада 2022 року*

Додаток III містить приклади зразків JSON, отриманих зі схеми XSD стандарту ВОІВ ST.96, щоб допомогти ВІВ у створенні аналогічних екземплярів. Кожен із цих екземплярів має пройти перевірку на відповідність схемі JSON, представлений в Додатку II.

Наступні приклади екземплярів відповідають деяким компонентам на рівні документа, які представлені в Додатку VII до стандарту ВОІВ ST.96, і не відображають реальних даних:

- **patchPublication.json** : цей компонент рівня документа використовується для добірки відомостей про публікацію заявки на патент. Цей приклад можна завантажити тут: *patchPublication.json*
- **tradeApplication.json** : цей компонент рівня документа використовується для добірки інформації, що стосується поданої заявки на торговельну марку. Цей приклад можна завантажити тут: *tradeApplication.json*
- **designApplication.json** : цей компонент рівня документа використовується для добірки інформації, що стосується поданої заявки на промисловий зразок. Цей приклад можна завантажити тут: *designApplication.json*

Приклади JSON можна завантажити в Доповненні до Додатка III за наступним посиланням:

https://www.wipo.int/standards/en/json_material/st97/annexiii_json_example_instances.zip

[Додаток IV наведено далі]

ДОДАТОК IV

ПЕРЕЛІК АКРОНІМІВ ТА АБРЕВІАТУР

Схвалено Комітетом зі стандартів VOIB (КСВ)
на його десятій сесії
25 листопада 2022 року

Акроніми та аббревіатури, що зустрічаються на початку типу об'єкта та назви властивості, ПОВИННІ бути в нижньому регістрі, наприклад, "pre", "bioDeposit". У випадку, якщо на початку назви наведено акронім великими літерами, усі символи мають бути у нижньому регістрі, наприклад, "idref", і "wipo" в імені властивості "wipoST3Code". В іншому випадку всі значення переліку, акронімів і аббревіатур ПОВИННІ бути вказані у верхньому регістрі, як зазначено нижче.

Наведені нижче акроніми/аббревіатури НЕ СЛІД розглядати в контексті кодів мов, валют, кодів відомств і кодів країн, як в стандарті VOIB ST.96, де можуть бути повторювані значення. Ці коди базуються на мовних кодах ISO 639-1, кодах валют ISO 4217, кодах стандарту VOIB ST.3 і кодах країн ISO 3166-1 відповідно.

AF	Відомче досьє
Alt	Альтернативний текст для зображення
B	Жирний шрифт
BioDeposit	Біологічне депонування
Br	Розрив рядка
CDX	Формат файлів програмного забезпечення <i>ChemDraw</i> (власність CambridgeSoft)
CPC	Спільна патентна класифікація
DD	Опис визначення
Del	Видалений текст
DL	Список визначень
DOI	Цифровий ідентифікатор об'єкта
DT	Визначення
DTD	Визначення типу документа
DWF	Формат файлів <i>Design Web Format</i>
DWG	Креслення
ECLA	Європейська класифікація
EIDR	Реєстр ідентифікаторів розваг
ExtRef	Посилання, які є зовнішніми для поточного документа XML
GI	Географічне зазначення
H<n>	"n" вказує на рівень заголовка з певним значенням від 1 до 15. Це означає, що ця аббревіатура у перелічуваному значенні представляє собою один з елементів у діапазоні від H1 до H15. Наприклад, "H1" означає "Заголовок 1".

I	Курсивний шрифт
IB	Міжнародне бюро
ID	Ідентифікатор для системної ідентифікації
IDREF	Одне посилання на ідентифікатор
IDREFS	Множина посилань на ідентифікатори
IGES	Формат файлів <i>Initial Graphic Exchange Specification</i>
IGO	Міжурядова організація, нормативна база якої забезпечує захист прав інтелектуальної власності або організації, що діють у рамках договору у сфері інтелектуальної власності
INID	Узгоджені на міжнародному рівні номери для ідентифікації (бібліографічних) даних
Ins	Вставлений текст
IP	Інтелектуальна власність
IPC	Міжнародна патентна класифікація
IPCR	Реформа Міжнародної патентної класифікації
IPO	Відомство інтелектуальної власності
IPR	Право інтелектуальної власності
ISMN	Міжнародний стандартний номер нотного видання
ISNI	Міжнародний стандартний ідентифікатор назв
ISO	Міжнародна організація стандартизації
JSON	Позначення об'єкта Javascript
LCC	Угода <i>Lower Camel Case</i>
LI	Елемент списку
LOR	Правова ліцензія
MathML	Мова розмітки для описання математичних виразів
MPEG	Група експертів з рухомих зображень (кіно)
MOL	Формат файлів для зберігання інформації про атоми, зв'язки, поєднуваність і координати молекули.
NB	Формат файлів записників програмного забезпечення <i>Mathematica</i>
NPL	Непатентна література
NUTS	Номенклатура територіальних одиниць для статистики
O	Закреслений текст
OASIS	Організація з розвитку стандартів у сфері структурованої інформації
OCR	Оптичне розпізнавання символів
OL	Упорядкований список
P	Параграф
PAN	Номер платіжної картки
PCT	Договір про патентну кооперацію

PKCS7	У криптографії ⁸ PKCS - це група стандартів криптографії з відкритим ключем ⁹ , а PKCS #7 ¹⁰ (PKCS7) - це стандарт синтаксису криптографічних повідомлень, який описує загальний синтаксис для даних, до яких може застосовуватися криптографія, така як цифрові підписи та цифрові конверти.
Pre	Попередньо відформатований текст
S	Закреслений текст
SEQL	Перелік послідовностей
SOC	Соціальний код
SPC	Сертифікат додаткової охорони
ST3	Стандарт BOIB ST.3
ST13	Стандарт BOIB ST.13
Sub	Підрядковий текст
Sup	Нарядковий текст
SVG	Формат файлів <i>Scalable Vector Graphic</i>
SWF	Формат файлів <i>Small Web Format</i>
SWIFT	Товариство всесвітніх міжбанківських фінансових телекомунікацій
ThreeDM	Просторове моделювання
ThreeDS	Програмне забезпечення <i>3D Studio</i>
TISA	Буквено-цифровий територіальний інформаційний код CISAC
TISN	Цифровий територіальний інформаційний код CISAC
TSG	Традиційні гарантовані особливості
U	Підкреслений шрифт
UCC	Угода <i>Upper Camel Case</i>
UL	Невпорядкований список
UPOV	Міжнародний союз з охорони нових сортів рослин
URI	Уніфікований ідентифікатор ресурсу
URL	Уніфікований локатор ресурсу
URN	Уніфікована назва ресурсу
W3C	Консорціум Всесвітньої мережі
WIPO	Всесвітня організація інтелектуальної власності
WMV	Формат файлів <i>Windows Media Video</i>

[Далі Додаток V наведено далі]

⁸ <http://en.wikipedia.org/wiki/Cryptography>

⁹ http://en.wikipedia.org/wiki/Public-key_cryptography

¹⁰ http://en.wikipedia.org/wiki/Cryptographic_Message_Syntax

ДОДАТОК V

ТЕРМІНИ ПРЕДСТАВЛЕННЯ

Схвалено Комітетом зі стандартів VOIB (КСВ)
на його десятій сесії
25 листопада 2022 року

Термін	Визначення	Тип даних
Сума	Грошова вартість	number
Категорія	Спеціально визначений розділ або підмножина в системі класифікації, у якій усі елементи мають одну й ту саму концепцію таксономії.	string
Код	Комбінація з однієї або декількох цифр, літер або спеціальних символів, яка замінюється певним значенням. Представляє собою обмежені, попередньо визначені значення або довільний формат.	string
Дата	Позначення певного моменту у часі, представлене роком, місяцем та днем	string, з ключовим словом "format": "date"
Ідентифікатор	Комбінація одного або декількох цілих чисел, букв, спеціальних символів, що однозначно ідентифікують певний екземпляр бізнес-об'єкта, але можуть мати значення, яке не легко визначити.	string
Індикатор	Позначення присутності, відсутності або вимоги чого-небудь. Логічні значення або істинні, або хибні - true або false (без лапок). Ці значення чутливі до регістру.	boolean або string
Міра	Міра - це числове значення, що визначається шляхом вимірювання об'єкта за допомогою певної одиниці вимірювання. <i>MeasureType</i> використовується для вираження фізичних параметрів об'єкта, таких як температура, довжина, швидкість, ширина, вага, об'єм, широта тощо. Іншими словами, <i>MeasureType</i> має використовуватися для вимірювання суттєвих або фізичних властивостей об'єкта в цілому.	number
Назва	Позначення об'єкта, виражене за допомогою слова або виразу.	String
Номер	Рядок цифрових або буквено-цифрових символів, за допомогою яких виражається мітка, значення, кількість або ідентифікація.	integer, number або string
Відсоток	Число, за допомогою якого виражається частина,	number

	поділена на 100.	
Кількість	Кількість є зліченим числом негрошових одиниць, включаючи, можливо, дробі. <code>Quantity</code> використовується для позначення зліченого числа предметів. <code>Quantity</code> має використовуватися для простих властивостей об'єкта, що тлумачиться як комбінація, зібрання або контейнер для визначення кількості або підрахування його складових. <code>Quantity</code> має завжди представляти собою злічене число предметів, а властивість може позначатися як "всього", "відправлено", "завантажено", "зберігається". <code>QuantityType</code> має використовуватись для компонентів, які потребують інформації про одиниці; а <code>xsd:nonNegativeInteger</code> має використовуватись для злічених компонентів, які не потребують інформації про одиниці.	number з ключовим словом "minimum": 0 або integer
Пропорція	Кількість або сума відносно іншої кількості або суми.	number
Текст	Відформатований або неформатований рядок символів, зазвичай слів (включно з аббревіатурою, коментарями).	string
Час	Позначення певного хронологічного моменту в певний період.	string, з ключовим словом "format": "time"
Дата і час	Зареєстрована дата і час, коли сталася подія.	string, з ключовим словом "format": "date-time"
URI	Уніфікований ідентифікатор ресурсу, що вказує, де розташований файл.	string, з ключовим словом "format": зі значенням "uri" або "uri-reference"

[Кінець Додатка V і Стандарту]