

## Definiowanie typów dokumentów

### Część 2. Przestrzenie nazw, XML Schema

## Globalne i lokalne deklaracje elementów i atrybutów

- Deklaracje lokalne:

```
<xsd:complexType name="OsobaTyp">
  <xsd:sequence>
    <xsd:element name="imie" type="xsd:string"
      minOccurs="1" maxOccurs="2"/>
    <xsd:element name="nazwisko" type="xsd:string"/>
  </xsd:sequence>
  <xsd:attribute name="NIP" type="NIPTyp"
    use="required"/>
</xsd:complexType>
```

## Globalne i lokalne deklaracje elementów i atrybutów

- Deklaracje globalne i odwołania do nich:

```
<xsd:element name="imie" type="xsd:string"/>
<xsd:element name="nazwisko" type="xsd:string"/>
<xsd:attribute name="NIP" type="NIPTyp"/>

<xsd:complexType name="OsobaTyp">
  <xsd:sequence>
    <xsd:element ref="imie" minOccurs="1"
      maxOccurs="2"/>
    <xsd:element ref="nazwisko"/>
  </xsd:sequence>
  <xsd:attribute ref="NIP" use="required"/>
</xsd:complexType>
```

## Elementy czy atrybuty?

- Informacje modelowane zwykle w elementach:
  - zawartość (elementy semantyczne, np. nazwisko, nazwa leku, adres),
  - struktura (np. rozdział, akapit, tytuł, lista),
  - wyszukiwanie (np. termin do indeksu, glosariusza, a także elementy dedykowane),
  - odesłania (np. hiperlinki, noty, przypisy).
- Informacje modelowane zwykle w atrybutach:
  - metainformacje (np. jednostka miary, waluta, język, priorytet),
  - identyfikatory wartości i odwołania do nich.

## Elementy czy atrybuty?

- Zalety elementów:
  - mogą zawierać inne elementy i atrybuty,
  - mogą się powtarzać,
  - można bezpiecznie rozszerzać ich model zawartości, np. dodając atrybuty,
  - można dokładnie sterować sposobem ich występowania (kolejnością, liczbą wystąpień, itp.).
- Zalety atrybutów:
  - są bardziej zwarte,
  - mogą być automatycznie dodawane do elementów, jeśli mają wartości domyślne.

## Wartości domyślne i ustalone

```
<xsd:element name="ilość" type="xsd:positiveInteger"
  default="1"/>
<xsd:element name="cena" type="xsd:positiveInteger"
  fixed="5"/>
<xsd:attribute name="waluta" type="xsd:string"
  default="PLN"/>
<xsd:attribute name="VAT" type="xsd:string"
  fixed="22%"/>
```

<i>Przed przetworzeniem</i>	<i>Po przetworzeniu</i>
<code>&lt;towar&gt;&lt;cena/&gt;&lt;/towar&gt;</code>	<code>&lt;towar&gt;&lt;cena&gt;5&lt;/cena&gt;&lt;/towar&gt;</code>
<code>&lt;towar&gt;&lt;ilość&gt;3&lt;/ilość&gt;&lt;/towar&gt;</code>	<code>&lt;towar&gt;&lt;ilość&gt;3&lt;/ilość&gt;&lt;/towar&gt;</code>
<code>&lt;kwota&gt;20&lt;/kwota&gt;</code>	<code>&lt;kwota waluta="PLN" VAT="22%&gt;20&lt;/kwota&gt;</code>
<code>&lt;kwota waluta="EUR"&gt;20&lt;/kwota&gt;</code>	<code>&lt;kwota waluta="EUR" VAT="22%&gt;20&lt;/kwota&gt;</code>

## Przestrzenie nazw

- Problem:
  - ta sama nazwa oznacza dwa różne byty w różnych dokumentach,
  - dokumenty te są powiązane (np. wspólnie przetwarzane, jeden zanurzony w drugim, itp.)
- Rozwiązanie: przestrzeń nazw (ang. *namespace*):
  - grupa nazw oddzielona (składniowo i semantycznie) od innych nazw.
- Status:
  - rekomendacja W3C z 14 stycznia 1999 r,
  - XML 1.0 (w tym DTD) nie wspierają przestrzeni nazw,
  - nowsze standardy (XML Schema, XSL, XLink, ...) wspierają przestrzenie nazw.

## Nazwy przestrzeni nazw

- Identyfikatory URI, np:
  - <http://www.w3.org/XSLT/Transform/1.0>
  - <http://szcz.mimuw.edu.pl/osoby>
  - <urn:szcz:mimuw>
- Nazwy przestrzeni nazw:
  - nie muszą wskazywać konkretnych zasobów,
  - pełnią funkcję unikatowych identyfikatorów,
  - są porównywane jako napisy (istotna jest wielkość liter).
- Wykorzystanie w dokumentach XML:
  - odwzorowanie na kilkunastkowy prefiks,
  - poprzedzanie nazw prefiksem.

## Użycie przestrzeni nazw w XML-u

```
<os:osoba
  xmlns:os="http://szcz.mimuw.edu.pl/osoby"
  xmlns:inst="http://szcz.mimuw.edu.pl/instytucje"
  xmlns:xhtml="http://www.w3.org/1999/xhtml">
  <os:imie>Jan</os:imie>
  <os:nazwisko>Kowalski</os:nazwisko>
  <os:NIP>123-456-78-90</os:NIP>
  <os:opis>To jest <xhtml:b>bardzo</xhtml:b> fajny
  facet!</os:opis>
  <os:pracuje-w>
    <inst:firma>
      <inst:nazwa>Business Consulting</inst:nazwa>
      <inst:NIP>987-654-32-10</inst:NIP>
    </inst:firma>
  </os:pracuje-w>
</os:osoba>
```

## Domyślna przestrzeń nazw

```
<osoba
  xmlns="http://szcz.mimuw.edu.pl/osoby"
  xmlns:inst="http://szcz.mimuw.edu.pl/instytucje"
  xmlns:xhtml="http://www.w3.org/1999/xhtml">
  <imie>Jan</imie>
  <nazwisko>Kowalski</nazwisko>
  <NIP>123-456-78-90</NIP>
  <opis>To jest <xhtml:b>bardzo</xhtml:b> fajny
  facet!</opis>
  <pracuje-w>
    <inst:firma>
      <inst:nazwa>Business Consulting</inst:nazwa>
      <inst:NIP>987-654-32-10</inst:NIP>
    </inst:firma>
  </pracuje-w>
</osoba>
```

## Widoczność przestrzeni nazw

```
<os:osoba xmlns:os="http://szcz.mimuw.edu.pl/osoby">
  <os:imie>Jan</os:imie>
  <os:nazwisko>Kowalski</os:nazwisko>
  <os:NIP>123-456-78-90</os:NIP>
  <os:pracuje-w>
    <os:firma
      xmlns:os="http://szcz.mimuw.edu.pl/instytucje">
      <os:nazwa>Business Consulting</os:nazwa>
      <os:NIP>987-654-32-10</os:NIP>
    </os:firma>
  </os:pracuje-w>
</os:osoba>
```

## Nazwy z prefiksem i bez

- Nazwy elementów:
  - kwalifikowane – należą do pewnej przestrzeni nazw:
    - poprzedzone prefiksem,
    - nie poprzedzone prefiksem, jeśli są w zasięgu deklaracji domyślnej przestrzeni nazw;
  - niekwalifikowane – nie należą do żadnej przestrzeni nazw:
    - nie poprzedzone prefiksem, poza zasięgiem deklaracji domyślnej przestrzeni nazw.
- Nazwy atrybutów:
  - poprzedzone prefiksem (globalne) – należą do pewnej przestrzeni nazw,
  - nie poprzedzone prefiksem – nie należą do żadnej przestrzeni nazw.

## Nazwy elementów bez prefiksów

```
<osoba xmlns="http://szcz.mimuw.edu.pl/osoby">
  <imie>Jan</imie>
  <nazwisko>Kowalski</nazwisko>
  <NIP>123-456-78-90</NIP>
  <pracuje-w>
    <firma xmlns="">
      <nazwa>Business Consulting</nazwa>
      <NIP>987-654-32-10</NIP>
    </firma>
  </pracuje-w>
</osoba>
```

## Nazwy atrybutów

```
<osoba xmlns="http://szcz.mimuw.edu.pl/osoby"
  xmlns:xlink="http://www.w3.org/1999/xlink">
  <imie>Jan</imie>
  <nazwisko>Kowalski</nazwisko>
  <NIP>123-456-78-90</NIP>
  <pracuje-w xlink:type="simple"
    xlink:href="..." rodzaj="etat">Business
    Consulting</pracuje-w>
</osoba>
```

## Atrybuty: ta sama nazwa czy nie?

- Poprawne:
 

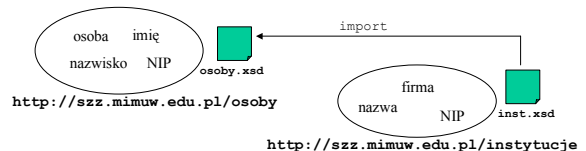
```
<x xmlns:n1="http://szcz.mimuw.edu.pl/n1"
  xmlns:n2="http://szcz.mimuw.edu.pl/n2">
  <good n1:a="1" n2:a="2"/>
</x>
```
- Niepoprawne:
 

```
<x xmlns:n1="http://szcz.mimuw.edu.pl/n1"
  xmlns:n2="http://szcz.mimuw.edu.pl/n1">
  <bad n1:a="1" n2:a="2"/>
</x>
```
- Poprawne:
 

```
<x xmlns:n1="http://szcz.mimuw.edu.pl/n1"
  xmlns="http://szcz.mimuw.edu.pl/n1">
  <good n1:a="1" a="2"/>
</x>
```

## Przestrzenie nazw i schematy

- Schemat zapisuje się przy pomocy jednego lub więcej dokumentów schematów.
- Schemat deklaruje nazwy należące do zero lub więcej przestrzeni nazw.
- Dokument schematu deklaruje nazwy należące do zero lub jednej przestrzeni nazw.
- Przestrzeń nazw zawiera nazwy zadeklarowane w zero lub więcej schematach.



## Dokument schematu (1)

```
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://szcz.mimuw.edu.pl/osoby"
  targetNamespace="http://szcz.mimuw.edu.pl/osoby">
  <xsd:complexType name="OsobaTyp">
    ...
  </xsd:complexType>
  <xsd:element name="osoba" type="OsobaTyp"/>
  <xsd:element name="numer" type="xsd:integer"/>
  ...
</xsd:schema>
```

## Dokument schematu (2)

```
<schema
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:os="http://szcz.mimuw.edu.pl/osoby"
  targetNamespace="http://szcz.mimuw.edu.pl/osoby">
  <complexType name="OsobaTyp">
    ...
  </complexType>
  <element name="osoba" type="os:OsobaTyp"/>
  <element name="numer" type="integer"/>
  ...
</schema>
```

## Łączenie dokumentów schematów

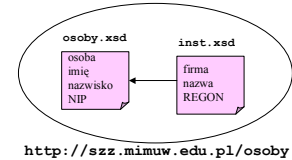
- Schemat zapisuje się przy pomocy jednego lub więcej dokumentów schematów.
- Metody budowania schematu z dokumentów schematów:
  - instrukcje `include`, `import` i `redefine`,
  - lokalizacje dokumentów schematów są określone w egzemplarzu,
  - procesor łączy dokumenty schematów znajdujące się w predefiniowanych (skonfigurowanych) lokalizacjach,
  - lokalizacje dokumentów schematów są przekazywane jako parametry wiersza poleceń.

## include

- Dołączanie dokumentu schematu do docelowej przestrzeni nazw głównego dokumentu schematu.

```
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://szz.mimuw.edu.pl/osoby"
  targetNamespace="http://szz.mimuw.edu.pl/osoby">
  <xsd:include schemaLocation="inst.xsd"/>
  ...
</xsd:schema>
```

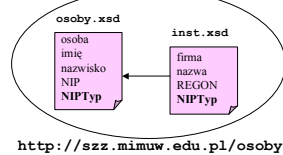
- Dołączany dokument musi:
  - mieć taką samą docelową przestrzeń nazw jak dokument główny, lub
  - nie mieć docelowej przestrzeni nazw.



## redefine

- Dołączanie dokumentu schematu do docelowej przestrzeni nazw głównego dokumentu schematu z możliwością przedefiniowania:
  - typów prostych i złożonych,
  - nazwanych grup modeli,
  - grup atrybutów.

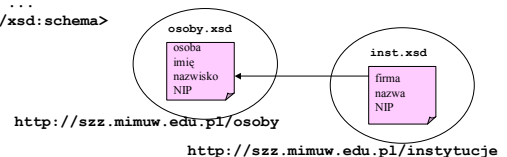
```
<xsd:redefine schemaLocation="inst.xsd"/>
```



## import

- Odwołanie do komponentów w innej przestrzeni nazw.

```
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://szz.mimuw.edu.pl/osoby"
  xmlns:inst="http://szz.mimuw.edu.pl/instytucje"
  targetNamespace="http://szz.mimuw.edu.pl/osoby">
  <xsd:import schemaLocation="inst.xsd"
    namespace="http://szz.mimuw.edu.pl/instytucje"/>
  ...
</xsd:schema>
```



## Egzemplarz

- Przestrzeń nazw egzemplarzy XML Schema: `http://www.w3.org/2001/XMLSchema-instance` zawiera atrybuty:
  - `nil`,
  - `type`,
  - `schemaLocation`,
  - `noNamespaceSchemaLocation`.

## schemaLocation

```
<?xml version="1.0"?>
<osoba
  xmlns="http://szz.mimuw.edu.pl/osoby"
  xmlns:inst="http://szz.mimuw.edu.pl/instytucje"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://szz.mimuw.edu.pl/osoby
    http://szz.mimuw.edu.pl/instytucje
    inst.xsd">
  <imię>Jan</imię><nazwisko>Kowalski</nazwisko>
  <pracuje-w>
    <inst:firma>
      <inst:nazwa>Business Consulting</inst:nazwa>
      <inst:NIP>987-654-32-10</inst:NIP>
    </inst:firma>
  </pracuje-w>
</osoba>
```

## noNamespaceSchemaLocation

```
<?xml version="1.0"?>
<osoba
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation=
    "osoby.xsd">
  <imie>Jan</imie>
  <nazwisko>Kowalski</nazwisko>
</osoba>
```

## Wartości puste w egzemplarzach

- Sposoby oznaczenia braku wartości:
  - brak elementu/attributu,
  - element/attribut występuje, ale jest pusty,
  - element/attribut występuje i zawiera specjalną wartość, np. N/A,
  - element ma wartość nil.

```
<?xml version="1.0"?>
<osoba
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <imie>Jan</imie>
  <drugie-imie xsi:nil="true"/>
  <nazwisko>Kowalski</nazwisko>
</osoba>
```

- Dopuszczenie wartości nil w deklaracji elementu:

```
<xsd:element name="drugie-imie" type="xsd:string"
  nillable="true"/>
```