

## TROCHE DEFINICJI

Każdy dokument XML musi być *poprawny składniowo* (*well-formed*) – być zgodny z gramatyką XML i spełniać *well-formedness constraints* wymienione w rekomendacji XML.

Dodatkowo dokument XML może być *poprawny strukturalnie* (*valid*) – być zgodny z definicją struktury. Jeśli nie jest powiedziane inaczej, chodzi o DTD, a dokument ma spełniać *validity constraints* wymienione w rekomendacji XML. Tych samych terminów używa się także do określenia zgodności ze schematem XML Schema bądź innych standardów.

Proces sprawdzania poprawności strukturalnej nazywa się *walidacją*. Po wykonaniu walidacji mamy pewność, że dokument posiada właściwą strukturę. Dzięki temu, np. we własnych programach czy w arkuszach XSLT, nie musimy już rozważać przypadków błędnych i od razu wiemy, że np. wymagane elementy rzeczywiście występują.

Informacje oparte o <https://www.mimuw.edu.pl/>

## JAK TO ZROBIĆ?

A co zrobić, jeśli chcemy sprawdzić poprawność wygenerowanego przez nas xml'a, w szczególności czy zawiera wszelkie wymagane elementy?

Z pomocą wówczas przychodzi nam klasa DOMDocument oraz schemat opisu struktury xml w postaci DTD (Document Type Definition).

DTD pozwoli nam określić strukturę dokumentu (w naszym przypadku dokumentu xml) i listę poprawnych elementów. Samo DTD możemy zdefiniować bezpośrednio w dokumencie xml, jednak na potrzeby wymiany danych bardziej eleganckim rozwiązaniem będzie skorzystanie z DTD zdefiniowanego w oddzielnym pliku. Jeżeli definiujemy DTD w oddzielnym pliku, wówczas musimy w pliku xml przekazać informację o tym DTD.

Poniżej przykład w jaki sposób w PHP możemy sprawdzać poprawność naszego pliku.

Do walidacji poprawności xml zarówno składniowej jak również pod kątem poprawności poszczególnych elementów pliku xml możemy wykorzystać poniższy kod:

poszczególnych elementów pliku xml możemy wykorzystać:

```
1 | libxml_use_internal_errors( true );  
2 | $dom = new DOMDocument();  
3 | $dom->load( "filename.xml" );  
4 | $dom->validate();  
5 | $errors = libxml_get_errors();
```

Pierwszym krokiem jest wyłączenie wewnętrznego raportowania błędów przez libxml i pozwolenie użytkownikowi na ich przechwycenie. Następnie tworzymy sobie instancję

DOMDocument, do której ładujemy plik o wskazanej nazwie. Na instancji tego obiektu wywołujemy metodę *validate()*, która dokonuje walidacji pliku. Ostatni krok to przechwycenie znalezionych błędów, co wykonujemy przy użyciu funkcji *libxml\_get\_errors()*. Funkcja ta zwróci nam pustą tablicę w przypadku braku błędów lub tablicę zawierającą elementy typu *libXMLError* w przypadku wystąpienia błędów walidacji.

Należy zwrócić uwagę tylko na jedną kwestię. Jeżeli plik xml nie jest poprawny pod kątem składniowym, wówczas walidacji przy użyciu DTD jest pomijana. Tak więc odbywa się to niejako dwuetapowo: w pierwszym kroku sprawdzana jest poprawność składniowa pliku, a następnie jego walidacja w odniesieniu do wskazanego pliku DTD.

Informacje oparte o <http://blog.tulikowski.com>