

## Dodatek k datovým typům

Při implementaci DOM Level 3 od W3.org jsem narazil na několik chybějících prvků které jsou uvedeny v tomto dokumentu.

### Libovolný typ - ANY

V specifikaci DOM, konkrétně DOM Core Level 3 je zahrnuta možnost ukládání uživatelských dat do uzlu. U těchto dat není typ specifikován přesně, ale pomocí klíčového slova *any* v jazyku *IDL*. Implementačně je to uskutečnitelné několika způsoby:

- použije se BLOB (binary large object) a provede se serializace datových struktur
- do uzlu se bude ukládat pouze ukazatel na data (void pointer), který se při čtení musí přetypovat zpět
- vytvoří se variabilní typ, sestávající z
  - ukazatele na data
  - hodnoty která specifikuje typ dat
  - metod na čtení/zápis hodnot různých typů

My tady ale řešíme jenom definici typů, takže nám postačí, když doplníme specifikaci o:

```
<any xmlns="Type" />
```

Ve světě PHP se libovolný typ označuje jako *mixed*.

### Libovolný objekt - OBJECT

Ve W3 specifikaci k DOM Core Level 3 je taky použitý typ *Object* - v IDL:

```
typedef    Object DOMObject;
:
:
interface DOMImplementation {
:
:
// Introduced in DOM Level 3:
DOMObject    getFeature(in DOMString feature,
                        in DOMString version);
};
```

Když jsem vypátral použití:

```
// example 1, without prepending the "+"
if (myNode.isSupported("Events", "3.0")) {
    EventTarget evt = (EventTarget) myNode;
    // ...
}

// example 2, with the "+"
if (myNode.isSupported("+Events", "3.0")) {
    // (the plus sign "+" is irrelevant for the getFeature method itself
    // and is ignored by this method anyway)
    EventTarget evt = (EventTarget) myNode.getFeature("Events", "3.0");
    // ...
}
```

Tak bylo jasno, že *Object* se musí přetypovat na vyšší typ (na něco použitelnějšího). Z tohoto lze taky usuzovat, že typ *Object* je prázdná třída která je zároveň základem pro ostatní třídy. Zápis typu objekt, prázdné třídy, je následovný:

```
<object xmlns="Type" />
```

Formálně by to šlo zapsat taky jako:

```
<class xmlns="Type" id="Object" />
```