Mehrere DIV Container mit AJAX aktualisieren

== Einleitung ==
Das folgende Script könnt ihr einerseits dazu verwenden mehrere Divs mit einem einzigen AJAX Request zu aktualisieren. Alle Div container können unabhängig aktualisiert werden.
Außerdem ist das Script in der Lage die Daten ständig aktuell zu halten indem es in einem bestimmten Intervall die Daten aktualisiert.

== Benutzung ==
Die Nutzung ist einfach. Unten im Downloadbereich findet ihr zum einen die Scripte die ihr einbinden müsst.
Ansonsten definiert ihr euch ein Updater wie folgt:

Quellcode

```
1. <script type="text/javascript">
2. var up = new UpdateManyDivs('callback.php', 750);
3. </script>
```

== Parameter ==
callback.php ist die Datei welche die Inhalte zur Verfügung stellt. 750 ist das Zeitintervall in Millisekunden zu dem die Updates geladen werden sollen.

== Updatebereiche definieren ==
Welche Container aktualisiert werden sollen macht ihr anhand der ElementID fest. Die ID des Containers und der Parameter beim update.push müssen übereinstimmen.
Eine ID muss natürlich eindeutig sein.

Quellcode

```
1. <div id="update1">Ich werde aktualisiert #1</div>
2. <script type="text/javascript">up.push('update1');</script>
```

== Updates liefern (callback.php) ==
In der Demo Applikation sieht die callback php wie folgt aus:

Quellcode

```
1. $data = array();
2. foreach(array_keys($_GET) as $identfier) {
3. $data[$identfier] = getRandomText($identfier);
4. }
5. echo json_encode($data);
```

statt der getRandomText müsst ihr nur eure sinnvolle Datenbankabfrage einbauen.
=== Beispiel: MySQL ===

Quellcode

```
1. function getTextMYSQL($identfier) {
2. $sql = "SELECT text FROM tabelle WHERE id = $identifier";
3. $res = mysql_query($sql);
4. $row = mysql_fetch_array($res);
5. return $row['text'];
6. }
```

=== Beispiel: PDO ===

Quellcode

```
1. function getTextPDO($identfier) {
2. $sql = "SELECT text FROM tabelle WHERE id = :identifier";
3. $stmt = MyDB::getInstance()->prepare($sql);
4. $stmt->execute(array(
5. ':identifier' => $identfier
6. ));
7. $row = $stmt->fetch();
8. return $row['text'];
9. }
```

== Code ==

Download

Quellcode

```
1. /**
2. * xxx [..] by http://www.easy-coding.de
3. *
4. * @param url
5. * @param poll
6. */
7. function UpdateManyDivs(url, poll) {
8. this.url = url;
9. this.poll = poll ? poll : 750;
10. this.list = [];
11. this.timer = null;
12. /**
14. * adds elem
15. * @param id string as id
16. */
17. this.push = function(id) {
18. this.list.push(id);
19. };
20. /**
22. * sends single request
23. * @param loop boolean
24. */
25. this.fire = function(loop) {
26. ajaxPost(this.url + '?' +this.list.join('&'), 'seed='+ new Date().getTime(), function(up) {
27. return function() {
28. if (this.readyState == 4 && this.status == 200) {
29. if(loop) {
30. // start timer
31. up.start();
32. }
```

```
34. var data = eval('(' + this.responseText + ')');
35. for(var key in data) {
36. document.getElementById(key).innerHTML = data[key];
37. }
38. }
39. };
40. }(this));
41. };
42. /**
44. * stops auto updater
45. */
46. this.stop = function() {
47. window.clearTimeout(this.timer);
48. };
50. /**
51. * starts auto updater
52. */
53. this.start = function() {
54. this.timer = window.setTimeout(function(up) {
55. return function() {
56. up.fire(true);
57. };
58. }(this), this.poll);
59. };
60. }
```

Alles anzeigen

== Steuerung ==
Ihr steuert das Verhalten mit 3 Funktionen. Zum einen könnt ihr das Update manuell erzwingen. Ihr könnt das Update aber auch über einen automatischen Updater realisieren.

Quellcode

```
1. <a href="#" onclick="up.fire()">force single update</a>
2. <a href="#" onclick="up.start()">start autoupdate</a>
3. <a href="#" onclick="up.stop()">disable autoupdate</a>
```

Wenn ihr den automatischen Updater immer aktivieren wollt, dann startet ihr doch einfach beim Laden des Dokuments:

Quellcode

```
1. <body onload="up.start()">
```

== Demo ==
Eine Live Demo mit allen möglichen Formularelementen findet ihr unter demo.easy-coding.de/ajax/mehre…tainer-ajax-aktualisieren. Des weiteren wird der kompletten Code hier als ZIP Archiv zur Verfügung gestellt: download.zip.