

DIE WORDPRESS-HERAUSFORDERUNG

krone.at - learned lessons

5.174.628

Unique Clients

ÖWA Basic 02 2018

32.141.875

Unique Clients

ÖWA Basic 02 2018

99.504.198

Page Impressions

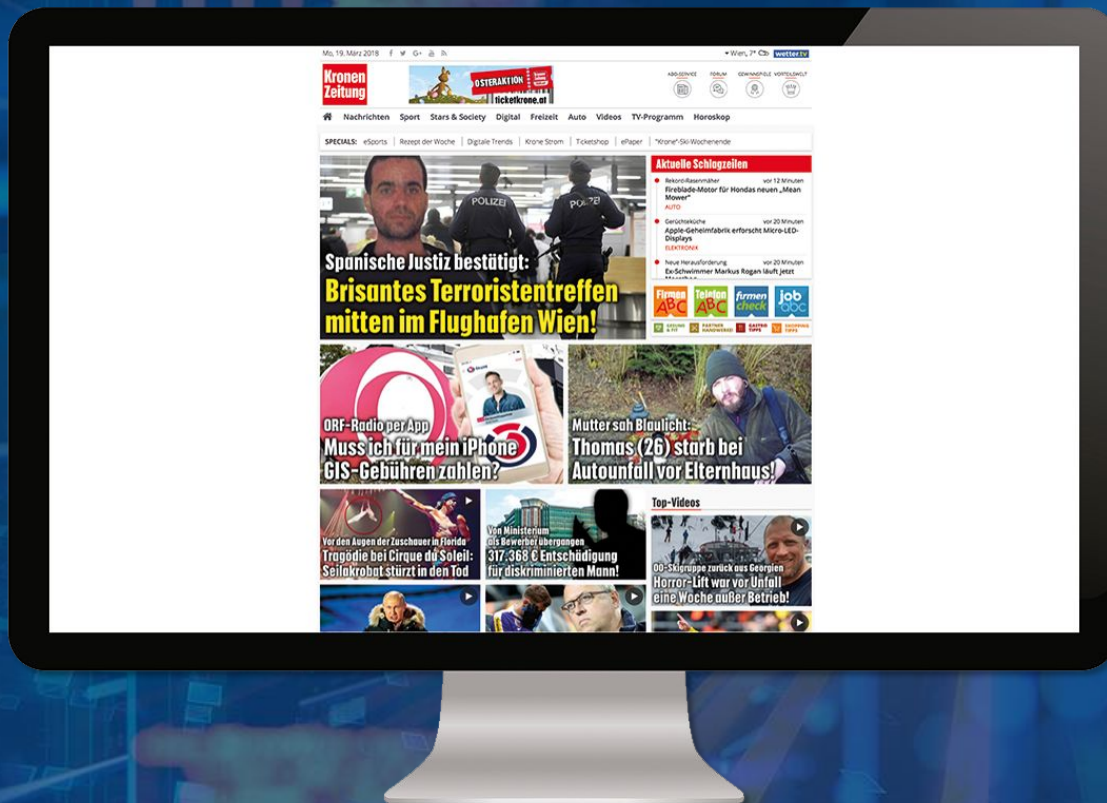
ÖWA Basic 02 2018



~ 850.000
Lines of Code

~ 1.8
Deploys / Tag

4-6
TB / tag



- CMS aus dem Jahr 2001/2002
- Proprietäre closed source Script-Sprache
- Oracle Datenbank
- gigantischer Monolith
- skaliert schlecht

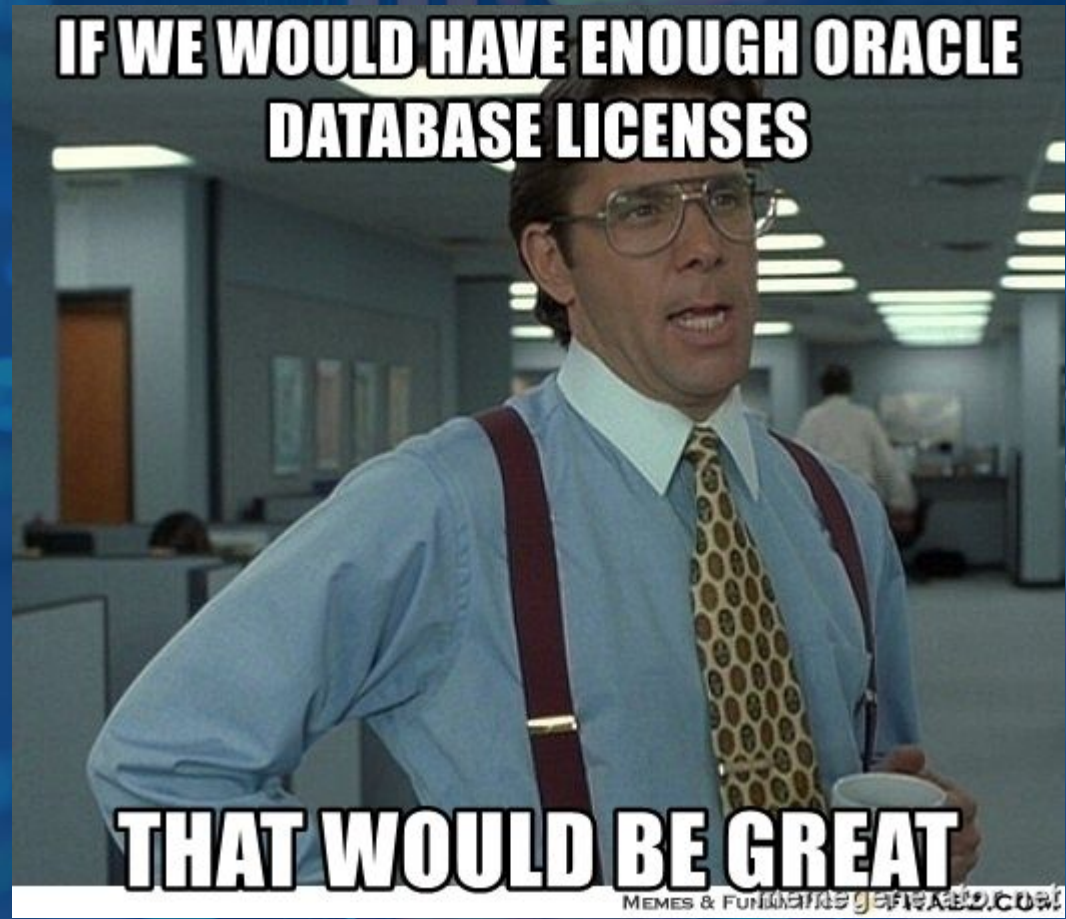
```
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
elif_operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
#mirror_ob.select = 0
None = bpy.context.selected_objects[0]
bpy.data.objects[mirror_ob.name].select = 1
```




modifier object

active ob



ected mirror modifier object
b
er ob is the active ob

- Neues CMS
 - von 0 auf! legacy support
- Offene Standards
 - PHP, node, ...
- Skalierbar
 - 0 single-point-of-failures
- state-of-the art
- Oracle Lizenz einsparen

```
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
elif_operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
#mirror_ob.select = 0
print = bpy.context.selected_objects[0]
#the object is now selected
```


- self-made 🚫
- Typo 3 🚫
- Thunder 🚫
- Wordpress 🤔

```
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
elif_operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
#mirror_ob.select = 0
name = bpy.context.selected_objects[0]
bpy.data.objects[name].select = 1
```



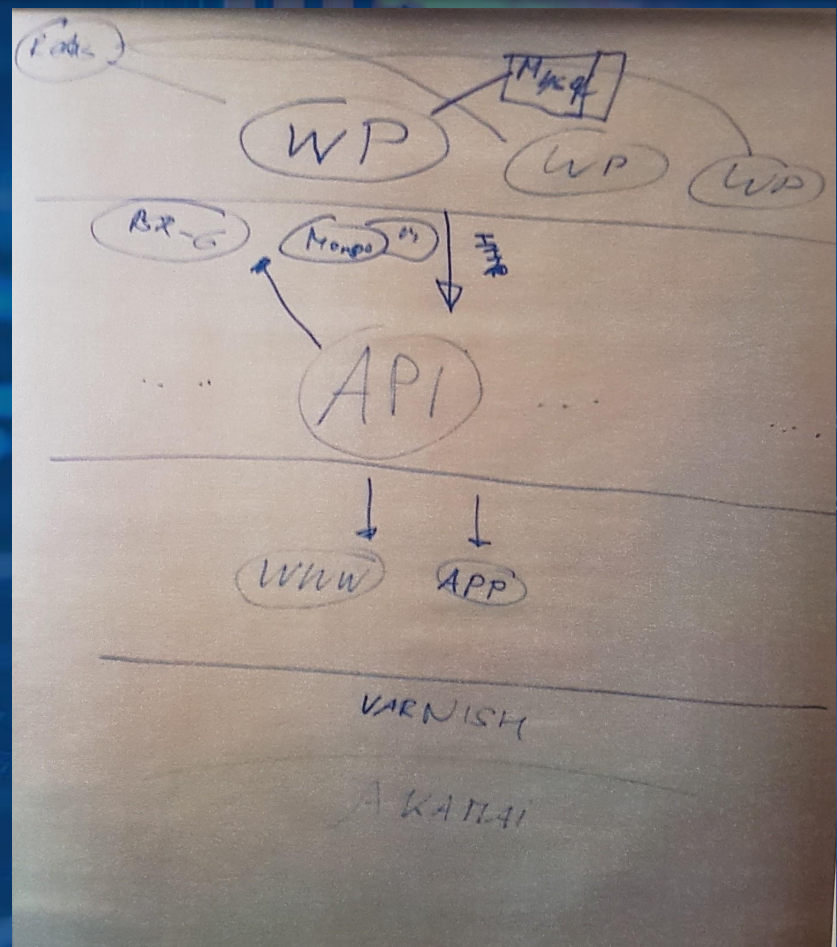

the deselected mirror modifier object

modifier_ob
modifier ob is the active ob

- wp_postmeta 🙄
- coding standards and code quality 😂
- MySQL ❤️
- Filters/Actions ❤️

```
elif_operation == "MIRROR":
    mirror_mod.use_x = False
    mirror_mod.use_y = True
    mirror_mod.use_z = False
elif_operation == "MIRROR":
    mirror_mod.use_x = True
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
#mirror_ob.select = 0
None = bpy.context.selected_objects[0]
bpy.data.objects[mirror_ob.name].select = 1
```



the deselected mirror modifier object

modifier_ob
modifier ob is the active ob



ror modifier object

the active ob

- Nie wieder Monolith
- Jeder Teil - austauschbar
- Kein kompletter rewrite
- learn and adapt

```
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
elif_operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
#mirror_ob.select = 0
None = bpy.context.selected_objects[0]
bpy.data.objects[mirror_ob.name].select = 1
```

- **CDN - Akamai** ⚡
- **Origin - Varnish**
- **Redis**
- **Database** 💣



- **BACKEND**
 - **WordPress**
 - **admin**
 - **json**
- **API**
 - **Symfony**
- **FRONTEND**
 - **Symfony**

```
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
elif_operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
#mirror_ob.select = 0
None = bpy.context.selected_objects[0]
bpy.data.objects[mirror_ob.name].select = 1
```

- **NDB Cluster -> Fail!**
- **Master -> Slave, Slave, Slave**
 - **DB-dropin**
 - **insert -> Master**
 - **select -> Slave**
- **BACKUP Node**





- Build und deployment via GitLab-CI
- Reproduzier/Rebuild-bar
- Revertbar
- Confidence
- Rolling Release
- Slack





ted mirror modifier object

ob is the active ob

- **Grafana (influxdb, telegraf)**
- **Sentry**
- **Blackfire**
- **netdata**

```
operation == "MIRROR_Y":
    mirror_mod.use_x = False
    mirror_mod.use_y = True
    mirror_mod.use_z = False
elif operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
#mirror_ob.select = 0
name = bpy.context.selected_objects[0]
bpy.data.objects[name].select = 1
```


- postmeta exploded (12.772.835 records)
- WPQuery runtime > 90s
- wp-admin unbenutzbar

```
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
elif operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
#mirror_ob.select = 0
None = bpy.context.selected_objects[0]
bpy.data.objects[mirror_ob.name].select = 1
```



eselected mirror modifier object

er_ob
difier ob is the active ob

- **ElastiPress Plugin**

- **ElasticSearch**
- **+ Anpassungen für Wildcards usw.**

- **Plugin-flattable**

- **pro Post-Type eine flache Tabelle**
 - mit Datentypen (date, int usw.)
 - statt WPQuery - direkt \$wpdb verwenden
- **OSS:**

<http://github.com/KroneMultimedia/plugin-flattable>

```
mirror_mod.use_x = false
mirror_mod.use_y = True
mirror_mod.use_z = False
elif _operation == "MIRROR Z":
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
mirror_ob.select= 0
```




```
MySQL [wp]> desc wp_flattable_article;
```

Field	Type	Null	Key	Default	Extra
id	int(12)	NO	PRI	NULL	auto_increment
post_id	int(12)	YES	UNI	NULL	
post_type	varchar(100)	YES		NULL	
post_date	datetime	NO		0000-00-00 00:00:00	
post_date_gmt	datetime	NO		0000-00-00 00:00:00	
post_status	varchar(20)	NO		publish	
post_modified	datetime	NO		0000-00-00 00:00:00	
post_modified_gmt	datetime	NO		0000-00-00 00:00:00	
frontend_time_gmt	datetime	NO		0000-00-00 00:00:00	
offline_time	datetime	NO		0000-00-00 00:00:00	
article_format	varchar(100)	YES		NULL	
post_title	varchar(1024)	YES		NULL	
post_author	int(12)	YES		NULL	
comment_status	varchar(100)	YES		NULL	
first_published_gmt	datetime	NO		0000-00-00 00:00:00	

mirror modifier object

is the active ob



```
//FlatTable  
add_filter('krn_flattable_enabled_article', [$this, "flattable_enabled"], 10, 2);  
add_filter('krn_flattable_columns_article', [$this, "flattable_columns"], 10, 2);  
add_filter('krn_flattable_values_article', [$this, "flattable_values"], 10, 2);  
add_filter('krn_flattable_pre_write_article', [$this, 'flattable_pre_write'], 10, 2);  
add_filter('krn_flattable_pre_delete_article', [$this, 'flattable_pre_delete'], 10, 2);
```

```
public function flattable_enabled($state, $postObject = null)
{
    if ($postObject->post_type == "revision") {
        return false;
    }
    if ($postObject->post_password != "") {
        $this->wpdb->suppress_errors(true);
        $this->wpdb->query("delete from wp_flattable_article where post_id = " . $postObject->ID);
        $this->wpdb->query("delete from wp_flattable_articles_in_reSORT where post_id = " . $postObject->ID);
        $this->wpdb->suppress_errors(false);
        return false;
    }
    return true;
}
```



```
public function flattable_pre_delete($postObject)
{
    $table_name = $this->wpdb->prefix . 'flattable_articles_in_ressort';
    $this->wpdb->suppress_errors(true);
    $this->wpdb->query("delete from " . $table_name . " where post_id = " . $postObject->ID);
    $this->wpdb->suppress_errors(false);
}
```

```
public function flattable_pre_write($columns, $postObject)
{
    $new_columns = [
        ["column" => "post_id", "type" => "int(12)", "printf" => "%d"],
        ["column" => "post_ressort", "type" => "int(12)", "printf" => "%d"],
    ];
    //Check that the posts<->ressort table exists
    do_action('krn_flattable_check_table', 'articles_in_ressort', $new_columns);

    $table_name = $this->wpdb->prefix . 'flattable_articles_in_ressort';
    //Delete all current relations:
    $this->wpdb->query("delete from " . $table_name . " where post_id = " . $postObject->ID);

    //Insert new Relations
    $ressort_repeater = get_field('field_58512668ff1d2', $postObject->ID);
    if (is_array($ressort_repeater)) {
        foreach ($ressort_repeater as $rep) {
            if (isset($rep["ressort_id"]->ID) && $rep["ressort_id"]->ID != '') { // empty field or deleted ressort ... $rep["ressort_id"] = false
                $sql = "insert into $table_name (post_id, post_ressort) values(" . $postObject->ID . ", " . $rep["ressort_id"]->ID . ")";
                $this->wpdb->query($sql);
            }
        }
    }
}
```

```
public function flattable_columns($columns, $postObject)
{
    return [
        ["column" => "post_date", "type" => "datetime NOT NULL DEFAULT '0000-00-00 00:00:00'", "printf" => "%s"],
        ["column" => "post_date_gmt", "type" => "datetime NOT NULL DEFAULT '0000-00-00 00:00:00'", "printf" => "%s"],
        ["column" => "post_status", "type" => "varchar(20) NOT NULL DEFAULT 'publish'", "printf" => "%s"],
        ["column" => "post_modified", "type" => "datetime NOT NULL DEFAULT '0000-00-00 00:00:00'", "printf" => "%s"],
        ["column" => "post_modified_gmt", "type" => "datetime NOT NULL DEFAULT '0000-00-00 00:00:00'", "printf" => "%s"],
        ["column" => "frontend_time_gmt", "type" => "datetime NOT NULL DEFAULT '0000-00-00 00:00:00'", "printf" => "%s"],
        ["column" => "offline_time", "type" => "datetime NOT NULL DEFAULT '0000-00-00 00:00:00'", "printf" => "%s"],
        ["column" => "article_format", "type" => "varchar(100)", "printf" => "%s"],
        ["column" => "post_title", "type" => "varchar(1024)", "printf" => "%s"],
        ["column" => "post_author", "type" => "int(12)", "printf" => "%d"],
        ["column" => "comment_status", "type" => "varchar(100)", "printf" => "%s"],
        ["column" => "first_published_gmt", "type" => "datetime NOT NULL DEFAULT '0000-00-00 00:00:00'", "printf" => "%s"],
    ];
}
```

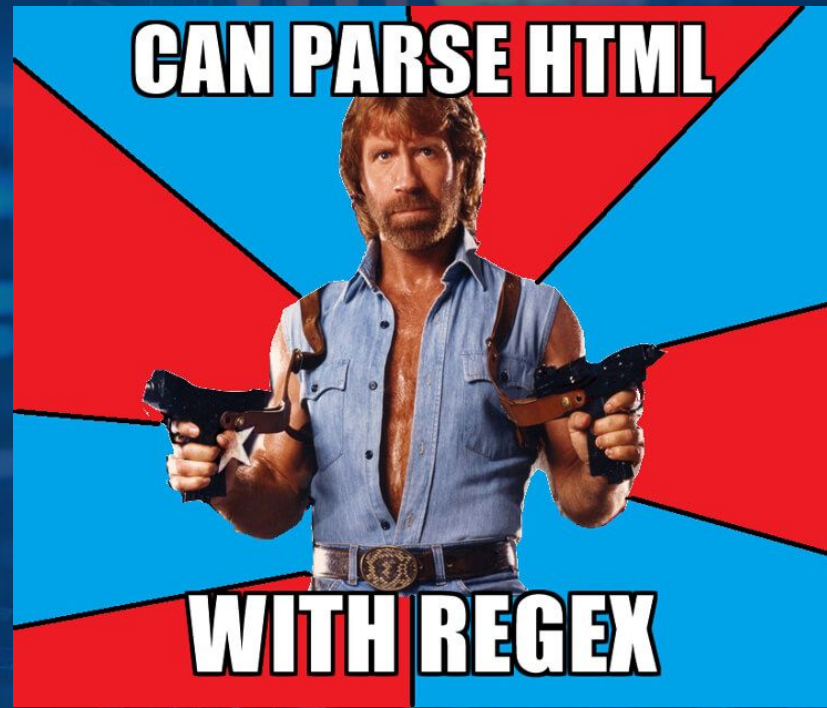
```
public function flattable_values($data, $postObject)
{
    $frontend_time_gmt = get_post_meta($postObject->ID, 'frontend_time_gmt', true);
    $article_format = get_post_meta($postObject->ID, 'article_format', true);
    $offline_time = get_post_meta($postObject->ID, 'offline_time', true);
    $first_published_gmt = get_post_meta($postObject->ID, 'first_published_gmt', true);
    return [
        "post_date" => $postObject->post_date,
        "post_date_gmt" => $postObject->post_date_gmt,
        "post_status" => $postObject->post_status,
        "post_modified" => $postObject->post_modified,
        "post_modified_gmt" => $postObject->post_modified_gmt,
        "frontend_time_gmt" => $frontend_time_gmt,
        "offline_time" => $offline_time,
        "article_format" => $article_format,
        "post_title" => $postObject->post_title,
        "comment_status" => $postObject->comment_status,
        "post_author" => $postObject->post_author,
        "first_published_gmt" => $first_published_gmt
    ];
}
```


Ein riesiger html string, im legacy system!

- native-app 🔥
- externe partner 🔥
- ist ein bild vorhanden? 🔥

```
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
elif_operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
#mirror_ob.select = 0
name = bpy.context.selected_objects[0]
obj.data.object.name = name + "select"
```



```
the deselected mirror modifier object  
  
modifier_ob  
# modifier ob is the active ob
```

- **Strukturierter Inhalt**

- **JSON**

- **utf8**

- **block basiert**

- **html nur in limitierter form (,<i>)**

- **jede box kann custom option values haben.**

```
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
elif_operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select= 1
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
```

Artikel teilen

Kommentare 0 Drucken

Test HTML - sing a song.




Bild: APA/GEORG HOCHMUTH

the deselected mirror modifier object

modifier_ob

) # modifier ob is the active ob


```
[
  {
    "id": "bf8b4530",
    "type": "box",
    "meta": {
      "active": true,
      "name": "tinymce",
      "width-sm": 12,
      "width-md": 12,
      "width-lg": 12,
      "display-family": {
        "xs": true,
        "sm": true,
        "md": true
      },
      "width-xs": 12,
      "_custom": {
        "content": "<p><strong>Test HTML - sing a song.</strong></p>",
        "headline": true
      }
    },
    "box": {
      "type": "reference",
      "href": "http://api.krone.at/v1/boxes/tinymce"
    }
  },
  {
    "id": "9842926a",
    "type": "box",
    "meta": {
      "active": true,
      "width-xs": 12,
      "width-sm": 12,
      "width-md": 12,
      "display-family": {
        "xs": true,
        "sm": true,
        "md": true
      },
      "name": "image",
      "_custom": {
        "slideshow": false
      },
      "content-set": {
        "images": [
          {
            "id": "1683502",
            "type": "image",
            "settings": {
              "assign-time": 1522690811963
            }
          }
        ]
      }
    },
    "box": {
      "type": "reference",
      "href": "http://api.krone.at/v1/boxes/image"
    }
  }
]
```

```
add back the deselected mirror modifier object
```

```
active = modifier_ob  
(modifier_ob)) # modifier ob is the active ob
```

```
(objects[0])
```

```
select 1 1
```

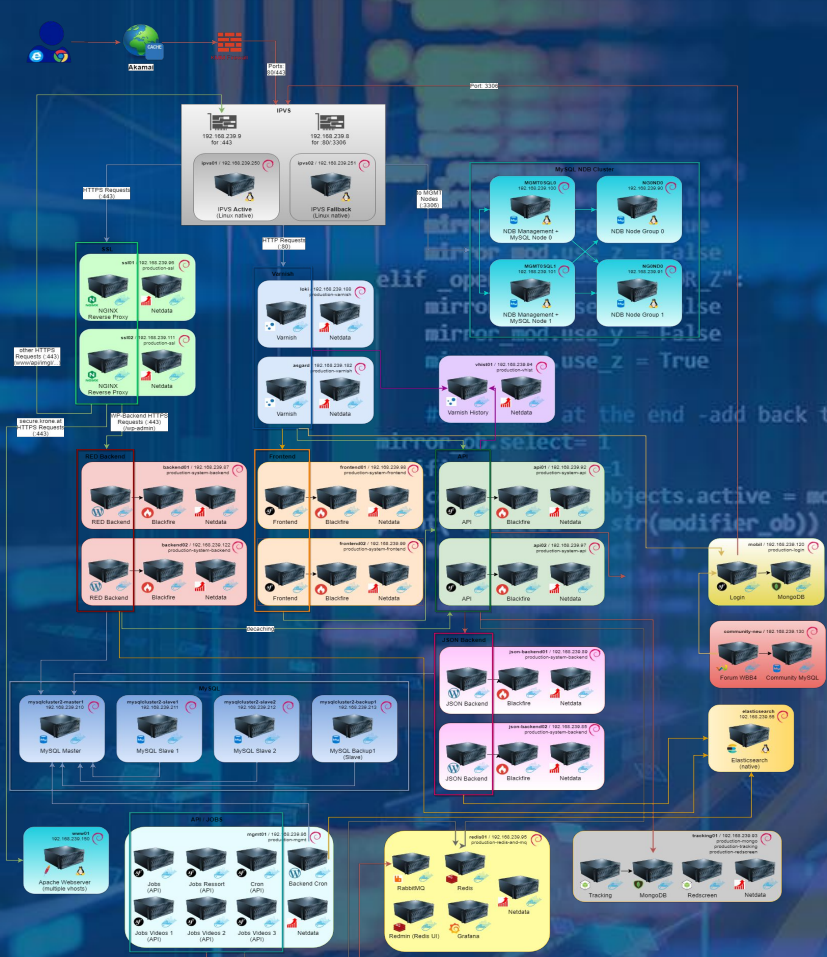
```
with the new objects, the box and image objects
```

- **Beispiel Blöcke**

- **Iframe**
- **Tinymce**
- **Image/Video**
- **Social (Facebook, Twitter usw)**
- **Content aus anderen bereichen.**

```
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
elif_operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select= 1
objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
mirror_ob.select = 0
```



WE ARE HIRING!

Entwickler gesucht

h.januschka@krone.at



@hjanuschka