

Dynamic Years Shortcode Implementation

Project Overview

Replace hardcoded year references ("13 years" and "30 years") throughout the 84EM website with dynamic shortcodes that automatically calculate based on the current year.

- **13 years** = WordPress expertise since 2012
- **30 years** = Programming/web development since 1995

These values were written in 2025 and would become outdated each year without manual updates.

Phase 1: Discovery and Analysis

Database Scan Results

Queried the WordPress database to find all instances of hardcoded year references:

```
SELECT post_type, COUNT(*) as count FROM wp_posts
WHERE post_status = 'publish'
AND (post_content LIKE '%13 year%' OR post_content LIKE '%30 year%'
      OR post_content LIKE '%13+ year%' OR post_content LIKE '%30+
year%')
GROUP BY post_type;
```

Results:

CONTENT TYPE	COUNT
Pages	59
Reusable Blocks (wp_block)	2

CONTENT TYPE	COUNT
Templates (wp_template)	2
Total	63

Patterns Found in Content

13 years (WordPress since 2012):

- "Over 13 years of WordPress plugin development"
- "13 years of building WordPress solutions"
- "**13 years** specializing in advanced WordPress"

30 years (programming since 1995):

- "30 years of experience"
- "nearly 30 years of programming experience"
- "backed by nearly 30 years of programming"
- "30 years of development experience"
- "30 years of web development expertise"
- "Delivered by a senior engineer with 30 years of experience" (hero blocks)

Key Content Items Identified

Reusable Blocks (High Impact - used across multiple pages):

- ID 3835: "84EM Hero (no buttons v2)"
- ID 3840: "84EM Hero (with buttons)"

Templates:

- ID 2500: "Single item: Local Page" (single-local)
- ID 2508: "page-wordpress-development-services-usa"

Core Pages:

- ID 2: WordPress Development & Consulting

- ID 2129: WordPress Services
- ID 2909: Custom WordPress Plugin Development
- ID 6908: Frequently Asked Questions
- ID 6989: Welcome to 84EM

Local SEO Pages:

- ~50 state and city pages (WordPress Development Services in [Location])

Related Code Locations

The 84em-local-pages plugin generates local pages with these references:

- `plugins/84em-local-pages/src/Content/CityContentGenerator.php` (line 436)
 - `plugins/84em-local-pages/src/Content/StateContentGenerator.php` (line 409)
-

Phase 2: Solution Design

Approach Comparison

APPROACH	PROS	CONS
Shortcode (Chosen)	WordPress-native, simple, maintainable	Requires database update
Content Filter	No content changes needed	Magic behavior, potential false positives
Custom Block	Modern, visual in editor	Overkill for this use case
Placeholder Filter	Clean in editor	Non-standard approach

Shortcode Design

Three shortcodes implemented:

```
[years_since year="2012"] → 14 (as of 2026)
[years_since year="1995"] → 31 (as of 2026)
[wp_years]                → 14 (alias for WordPress experience since
2012)
[dev_years]                → 31 (alias for programming experience
since 1995)
```

Usage in content:

- "nearly [dev_years] years of programming"
- "over [wp_years] years of WordPress"
- "[wp_years]+ years specializing"

Migration Strategy

Replace hardcoded text with shortcodes:

- "30 years" → "[dev_years] years"
 - "13 years" → "[wp_years] years"
-

Phase 3: Implementation

Files Created

1. `includes/dynamic-years.php`

```
<?php
/**
 * Dynamic year shortcodes for experience calculations.
 */
```

```

* @package suspended-developer/eightyfourem
*/

namespace EightyFourEM\DynamicYears;

defined( 'ABSPATH' ) || exit;

const WORDPRESS_START_YEAR    = 2012;
const PROGRAMMING_START_YEAR = 1995;

/**
 * Calculate years since a given start year.
 *
 * @param int $start_year The starting year.
 * @return int Years elapsed.
 */
function calculate_years_since( int $start_year ): int {
    return (int) \date( 'Y' ) - $start_year;
}

// Main shortcode: [years_since year="2012"]
\add_shortcode(
    tag: 'years_since',
    callback: function ( array $atts ): string {
        $defaults = [ 'year' => (int) \date( 'Y' ) ];
        $atts      = \shortcode_atts( $defaults, $atts, 'years_since'
    );

        return (string) calculate_years_since( (int) $atts['year'] );
    }
);

// Alias: [wp_years] - WordPress experience since 2012
\add_shortcode(
    tag: 'wp_years',
    callback: function (): string {
        return (string) calculate_years_since( WORDPRESS_START_YEAR
    );
}
);

```

```
// Alias: [dev_years] - Programming experience since 1995
\add_shortcode(
    tag: 'dev_years',
    callback: function (): string {
        return (string) calculate_years_since( PROGRAMMING_START_YEAR
    );
}
);
```

2. `includes/cli-dynamic-years.php`

WP-CLI command class providing:

Commands:

- `wp 84em dynamic-years test` - Test shortcodes are working correctly
- `wp 84em dynamic-years stats` - Display migration statistics
- `wp 84em dynamic-years migrate --dry-run` - Preview changes without saving
- `wp 84em dynamic-years migrate` - Execute the migration

Key Features:

- Searches posts, pages, reusable blocks (`wp_block`), templates
- Case-insensitive replacement using `str_ireplace()`
- Dry-run mode for safe preview
- Progress reporting during migration
- Uses `wp_update_post()` for proper cache invalidation

3. Updated `functions.php`

Added requires:

```
require_once get_template_directory() . '/includes/cli-dynamic-  
years.php';  
require_once get_template_directory() . '/includes/dynamic-  
years.php';
```

4. Updated `AGENTS.md`

Added documentation for:

- Dynamic Content section in Key Include Files
 - Feature-specific testing instructions for Dynamic Years
-

Phase 4: Testing Results

Shortcode Tests

```
wp 84em dynamic-years test
```

Output:

```
=== Testing Dynamic Year Shortcodes ===  
[PASS] [years_since year="2012"] = 14  
[PASS] [wp_years] = 14  
[PASS] [dev_years] = 31  
[PASS] [years_since] (no year) = 0  
  
Success: All 4 tests passed.
```

Migration Statistics

```
wp 84em dynamic-years stats
```

Output:

```
=== Dynamic Years Migration Statistics ===

Shortcode values (calculated for 2026):
- [dev_years] (since 1995): 31
- [wp_years] (since 2012): 14

Search patterns:
- "30 years" -> "[dev_years] years"
- "13 years" -> "[wp_years] years"

Items with hardcoded years:
- Total items: 62
- "30 years" occurrences: 65
- "13 years" occurrences: 3

Affected items:
- [page] WordPress Development & Consulting (ID: 2)
- [page] WordPress Services (ID: 2129)
- [page] WordPress Development Services in Alaska (ID: 2449)
... (59 pages total)
- [wp_block] 84EM Hero (no buttons v2) (ID: 3835)
- [wp_block] 84EM Hero (with buttons) (ID: 3840)
- [template] single-local (ID: 2500)
- [template] page-wordpress-development-services-usa (ID: 2508)
```

Dry-Run Results

```
wp 84em dynamic-years migrate --dry-run
```

Output (abbreviated):










```
DRY RUN: No changes will be saved.
Migrating hardcoded years to dynamic shortcodes...
Found 62 items with potential year references.
[1/62] Would migrate page ID 2 "WordPress Development &
Consulting"... 1 replacement(s)
[2/62] Would migrate page ID 2129 "WordPress Services"... 1
replacement(s)
...
```



```
[37/62] Would migrate wp_block ID 3835 "84EM Hero (no buttons v2)"...
1 replacement(s)
[38/62] Would migrate wp_block ID 3840 "84EM Hero (with buttons)"...
1 replacement(s)
[39/62] Would migrate page ID 3987 "WordPress Consulting &
Strategy"... 2 replacement(s)
[40/62] Would migrate page ID 4041 "AI-Enhanced WordPress
Development"... 3 replacement(s)
[41/62] Would migrate page ID 6908 "Frequently Asked Questions"... 3
replacement(s)
...
[61/62] Would migrate template ID 2500 "single-local"... 1
replacement(s)
[62/62] Would migrate template ID 2508 "page-wordpress-development-
services-usa"... 1 replacement(s)
Success: DRY RUN complete. Would make 68 replacement(s) across 62
item(s).
```

Phase 5: Current Status

Completed Tasks (Staging - 84em.local)

1.  Created `includes/dynamic-years.php` with shortcodes
2.  Created `includes/cli-dynamic-years.php` with WP-CLI migration class
3.  Added requires in `functions.php`
4.  Tested shortcodes on 84em.local - All 4 tests passed
5.  Ran migration dry-run on 84em.local - 68 replacements across 62 items ready
6.  Updated AGENTS.md documentation
7.  Ran actual migration on 84em.local - 68 replacements across 62 items completed
8.  Verified pages display correctly in browser (homepage shows "31 years", Alaska page shows "nearly 31 years")
9.  Updated local-pages plugin prompts for future page generation

Pending Tasks (Production)

1. 🕒 Deploy theme changes to production
 2. 🕒 Run migration on production
 3. 🕒 Clear FlyingPress and Cloudflare caches
-

Verification Plan

After Migration on Local (84em.local)

1. **Homepage Hero** - Check reusable block displays "[dev_years] years" as "31 years"
2. **WordPress Development & Consulting page (ID: 2)** - Verify year displays correctly
3. **Local page example (e.g., Alaska ID: 2449)** - Confirm replacement worked
4. **FAQ page (ID: 6908)** - Check multiple replacements rendered
5. **Block Editor** - Edit a migrated page, confirm shortcode appears as `[dev_years]` text

After Production Deployment

1. Run WP-CLI commands:

```
wp 84em dynamic-years test
wp 84em dynamic-years migrate --dry-run
wp 84em dynamic-years migrate
```

2. Clear caches:

- FlyingPress cache
- Cloudflare cache

3. Verify pages in incognito browser
-

Files Modified Summary

FILE	CHANGE
<code>themes/eightyfourem/includes/dynamic-years.php</code>	CREATED - Shortcodes
<code>themes/eightyfourem/includes/cli-dynamic-years.php</code>	CREATED - WP-CLI migration
<code>themes/eightyfourem/functions.php</code>	Added requires
<code>themes/eightyfourem/AGENTS.md</code>	Added documentation
<code>plugins/84em-local-pages/src/Content/CityContentGenerator.php</code>	Updated API prompt
<code>plugins/84em-local-pages/src/Content/StateContentGenerator.php</code>	Updated API prompt

Future Maintenance

Annual Verification

The shortcodes automatically calculate the correct year difference. No annual updates needed.

Adding New Base Years

To add a new base year calculation:

1. Add a constant to `dynamic-years.php`
2. Add a new shortcode alias following the existing pattern

Local Pages Plugin Update (Completed)

API prompts in the local-pages plugin have been updated so future generated pages will include shortcodes:

- `CityContentGenerator.php` (line 436)
- `StateContentGenerator.php` (line 409)

Changed prompt from:

```
- Programming since 1995, WordPress specialist since 2012
```

To:

```
- Founded in 1995, specializing in WordPress since 2012. When  
mentioning years of experience in the content, output [dev_years] for  
total programming years and [wp_years] for WordPress years (these are  
WordPress shortcodes)
```