

(1/1) ArgumentCountError

Too few arguments to function TYPO3\CMS\Core\ExpressionLanguage\

ProviderConfigurationLoader::__construct(), 0 passed in

/var/www/html/vendor/typo3/cms-core/Classes/Utility/GeneralUtility.php on line 2969 and exactly 3 expected in /var/www/html/vendor/typo3/cms-core/Classes/ExpressionLanguage/

ProviderConfigurationLoader.php line 29

```
* This class resolves the expression language provider configuration and store in a cache.
*/
class ProviderConfigurationLoader
{
    public function __construct(
        private readonly PackageManager $packageManager,
        private readonly PhpFrontend $coreCache,
        private readonly string $cacheIdentifier,
    ) {
```

at TYPO3\CMS\Core\ExpressionLanguage\ProviderConfigurationLoader->__construct()

in /var/www/html/vendor/typo3/cms-core/Classes/Utility/GeneralUtility.php line 2969

```
        return self::$container->get($className);
    }

    // Create new instance and call constructor with parameters
    $instance = new $finalClassName(...$constructorArguments);
    // Register new singleton instance, but only if it is not a known PSR-11 container service
    if ($instance instanceof SingletonInterface && !(self::$container !== null && self::$container-
>has($className))) {
        self::$singletonInstances[$finalClassName] = $instance;
    }
```

at TYPO3\CMS\Core\Utility\GeneralUtility::makeInstance('TYPO3\CMS\Core\ExpressionLanguage\ProviderConfigurationLoader')

in /var/www/html/vendor/typo3/cms-core/Classes/ExpressionLanguage/Resolver.php line 47

```
    // by config.
    // The entire construct should be reviewed at this point and most likely
    // declared final as well.

    $providers = GeneralUtility::makeInstance(ProviderConfigurationLoader::class)-
>getExpressionLanguageProviders()[$context] ?? [];
    // Always add default provider
    array_unshift($providers, DefaultProvider::class);
    $providers = array_unique($providers);
    $functionProviders = [];
```

at TYPO3\CMS\Core\ExpressionLanguage\Resolver->__construct('site', array())

in /var/www/html/vendor/typo3/cms-core/Classes/Utility/GeneralUtility.php line 2969

```
        return self::$container->get($className);
    }

    // Create new instance and call constructor with parameters
    $instance = new $finalClassName(...$constructorArguments);
    // Register new singleton instance, but only if it is not a known PSR-11 container service
    if ($instance instanceof SingletonInterface && !(self::$container !== null && self::$container-
>has($className))) {
        self::$singletonInstances[$finalClassName] = $instance;
```

```
}
```

at TYPO3\CMS\Core\Utility\GeneralUtility::makeInstance('TYPO3\CMS\Core\ExpressionLanguage\Resolver', 'site', array())

in **/var/www/html/vendor/typo3/cms-core/Classes/Site/Entity/Site.php** line 158

```
if (!empty($baseVariants)) {
    $expressionLanguageResolver = GeneralUtility::makeInstance(
        Resolver::class,
        'site',
        []
    );
    foreach ($baseVariants as $baseVariant) {
        try {
            if ((bool)$expressionLanguageResolver->evaluate($baseVariant['condition'])) {
```

in **/var/www/html/vendor/typo3/cms-core/Classes/Site/Entity/Site.php** line 105

```
    ],
    ];
    $baseUrl = $this->resolveBaseWithVariants(
        $configuration['base'] ?? '',
        $configuration['baseVariants'] ?? null
    );
    $this->base = new Uri($this->sanitizeBaseUrl($baseUrl));

    foreach ($configuration['languages'] as $languageConfiguration) {
```

in **/var/www/html/vendor/typo3/cms-core/Classes/Configuration/SiteConfiguration.php** line 159

```
        $configuration['contentSecurityPolicies'] = $this->getContentSecurityPolicies($identifier);

        $rootPageId = (int)($configuration['rootPageId'] ?? 0);
        if ($rootPageId > 0) {
            $sites[$identifier] = new Site($identifier, $rootPageId, $configuration, $siteSettings);
        }
    }
    $this->firstLevelCache = $sites;
    return $sites;
```

at TYPO3\CMS\Core\Configuration\SiteConfiguration->resolveAllExistingSites(true)

in **/var/www/html/vendor/typo3/cms-core/Classes/Configuration/SiteConfiguration.php** line 112

```
{
    if ($useCache && $this->firstLevelCache !== null) {
        return $this->firstLevelCache;
    }
    return $this->resolveAllExistingSites($useCache);
}

/**
 * Creates a site configuration with one language "English" which is the de-facto default language for
 TYPO3 in general.
```

at TYPO3\CMS\Core\Configuration\SiteConfiguration->getAllExistingSites(true)

in **/var/www/html/vendor/typo3/cms-core/Classes/Site/SiteFinder.php** line 128

```
}
```

```

protected function fetchAllSites(bool $useCache = true): void
{
    $this->sites = $this->siteConfiguration->getAllExistingSites($useCache);
    foreach ($this->sites as $identifier => $site) {
        $this->mappingRootPagelIdToIdentifier[$site->getRootPagelId()] = $identifier;
    }
}

```

at TYPO3\CMS\Core\Site\SiteFinder->fetchAllSites()

in **/var/www/html/vendor/typo3/cms-core/Classes/Site/SiteFinder.php** line 52

```

*/
public function __construct(SiteConfiguration $siteConfiguration = null)
{
    $this->siteConfiguration = $siteConfiguration ??
GeneralUtility::makeInstance(SiteConfiguration::class);
    $this->fetchAllSites();
}

/**
 * Return a list of all configured sites

```

at TYPO3\CMS\Core\Site\SiteFinder->__construct(object(Helhum\TYPO3\ConfigHandling\Typo3SiteConfiguration))

in **/var/www/html/vendor/typo3/cms-core/Classes/Utility/GeneralUtility.php** line 3000

```

if (isset(self::$singletonInstances[$finalClassName])) {
    return self::$singletonInstances[$finalClassName];
}
// Create new instance and call constructor with parameters
return new $finalClassName(...$constructorArguments);
}

/**
 * Returns the class name for a new instance, taking into account

```

at TYPO3\CMS\Core\Utility\GeneralUtility::makeInstanceForDi('TYPO3\CMS\Core\Site\SiteFinder', object(Helhum\TYPO3\ConfigHandling\Typo3SiteConfiguration))

in **/var/www/html/var/cache/code/di/**

DependencyInjectionContainer_fa0a0c943435effca7ddc0f30a977c6503df7271.php line 26596

```

* @return \TYPO3\CMS\Core\Site\SiteFinder
*/
protected static function getSiteFinderService($container)
{
    return $container->privates['TYPO3\CMS\Core\Site\SiteFinder'] = \TYPO3\CMS\Core\Utility\
GeneralUtility::makeInstanceForDi(\TYPO3\CMS\Core\Site\SiteFinder::class, ($container->services['TYPO3\
CMS\Core\Configuration\SiteConfiguration'] ?? self::getSiteConfigurationService($container)));
}

/**
 * Gets the private 'TYPO3\CMS\Core\TypoScript\IncludeTree\TreeFromLineStreamBuilder' shared
autowired service.

```

at DependencyInjectionContainer_fa0a0c943435effca7ddc0f30a977c6503df7271::getSiteFinderService(object(DependencyInjectionContainer_fa0a0c943435effca7ddc0f30a977c6503df7271))

in **/var/www/html/var/cache/code/di/**

DependencyInjectionContainer_fa0a0c943435effca7ddc0f30a977c6503df7271.php line 12361

```

    * @return \Localizationteam\L10nmgr\LanguageRestriction\LanguageRestrictionRegistry
    */
    protected static function getLanguageRestrictionRegistryService($container)
    {
        return $container->services['Localizationteam\\L10nmgr\\LanguageRestriction\\
LanguageRestrictionRegistry'] = \TYPO3\CMS\Core\Utility\GeneralUtility::makeInstanceForDi(\
Localizationteam\L10nmgr\LanguageRestriction\LanguageRestrictionRegistry::class, \TYPO3\CMS\Core\Utility\
GeneralUtility::makeInstanceForDi(\Localizationteam\L10nmgr\LanguagesService::class, ($container-
>privates['TYPO3\\CMS\\Core\\Site\\SiteFinder'] ?? self::getSiteFinderService($container)));
    }

    /**
     * Gets the public 'Localizationteam\L10nmgr\Model\L10nBaseService' shared autowired service.

```

at DependencyInjectionContainer_fa0a0c943435effca7ddc0f30a977c6503df7271::getLanguageRestrictionRegistryService(object(DependencyInjectionContainer_fa0a0c943435effca7ddc0f30a977c6503df7271))
in **/var/www/html/vendor/symfony/dependency-injection/Container.php** line 232

```

        try {
            if (isset($container->fileMap[$id])) {
                return /* self::IGNORE_ON_UNINITIALIZED_REFERENCE */ 4 === $invalidBehavior ?
null : $container->load($container->fileMap[$id]);
            } elseif (isset($container->methodMap[$id])) {
                return /* self::IGNORE_ON_UNINITIALIZED_REFERENCE */ 4 === $invalidBehavior ?
null : $container->{$container->methodMap[$id]}($container);
            }
        } catch (\Exception $e) {
            unset($container->services[$id]);

```

at Symfony\Component\DependencyInjection\Container::make(object(DependencyInjectionContainer_fa0a0c943435effca7ddc0f30a977c6503df7271), 'Localizationteam\\L10nmgr\\LanguageRestriction\\LanguageRestrictionRegistry', 1)
in **/var/www/html/vendor/symfony/dependency-injection/Container.php** line 212

```

    public function get(string $id, int $invalidBehavior = self::EXCEPTION_ON_INVALID_REFERENCE): ?object
    {
        return $this->services[$id]
            ?? $this->services[$id = $this->aliases[$id] ?? $id]
            ?? ('service_container' === $id ? $this : ($this->factories[$id] ?? self::$make ??=
self::make(...))($this, $id, $invalidBehavior));
    }

    /**
     * Creates a service.

```

at Symfony\Component\DependencyInjection\Container->get('Localizationteam\\L10nmgr\\LanguageRestriction\\LanguageRestrictionRegistry')
in **/var/www/html/vendor/typo3/cms-core/Classes/EventDispatcher/ListenerProvider.php** line 96

```

    * @throws \InvalidArgumentException
    */
    protected function getCallable(string $service, string $method = null): callable
    {
        $target = $this->container->get($service);
        if ($method !== null) {
            // Dispatch to configured method name instead of __invoke()
            $target = [ $target, $method ];

```

```
}
```

at TYPO3\CMS\Core\EventDispatcher\ListenerProvider->getCallable('Localizationteam\L10nmgr\LanguageRestriction\LanguageRestrictionRegistry', 'addLanguageRestrictionDatabaseSchema')

in **/var/www/html/vendor/typo3/cms-core/Classes/EventDispatcher/ListenerProvider.php** line 84

```
}
```

```
foreach ($eventClasses as $className) {
```

```
    if (isset($this->listeners[$className])) {
```

```
        foreach ($this->listeners[$className] as $listener) {
```

```
            yield $this->getCallable($listener['service'], $listener['method']);
```

```
        }
```

```
    }
```

```
}
```

```
}
```

at TYPO3\CMS\Core\EventDispatcher\ListenerProvider->getListenersForEvent(object(TYPO3\CMS\Core\Database\Event\AlterTableDefinitionStatementsEvent))

in **/var/www/html/vendor/typo3/cms-core/Classes/EventDispatcher/EventDispatcher.php** line 42

```
// If the event is already stopped, nothing to do here.
```

```
if ($event instanceof StoppableEventInterface && $event->isPropagationStopped()) {
```

```
    return $event;
```

```
}
```

```
foreach ($this->listenerProvider->getListenersForEvent($event) as $listener) {
```

```
    $listener($event);
```

```
    if ($event instanceof StoppableEventInterface && $event->isPropagationStopped()) {
```

```
        break;
```

```
    }
```

at TYPO3\CMS\Core\EventDispatcher\EventDispatcher->dispatch(object(TYPO3\CMS\Core\Database\Event\AlterTableDefinitionStatementsEvent))

in **/var/www/html/vendor/typo3/cms-adminpanel/Classes/Service/EventDispatcher.php** line 38

```
public function dispatch(object $event): object
```

```
{
```

```
    $this->dispatchedEvents[] = $event;
```

```
    return parent::dispatch($event);
```

```
}
```

```
}
```

at TYPO3\CMS\Adminpanel\Service\EventDispatcher->dispatch(object(TYPO3\CMS\Core\Database\Event\AlterTableDefinitionStatementsEvent))

in **/var/www/html/vendor/typo3/cms-core/Classes/Database/Schema/SqlReader.php** line 72

```
    $sqlString[] = (string)file_get_contents($packagePath . 'ext_tables_static+adt.sql');
```

```
    }
```

```
}
```

```
    $event = $this->eventDispatcher->dispatch(new AlterTableDefinitionStatementsEvent($sqlString));
```

```
    $sqlString = $event->getSqlData();
```

```
    return implode(LF . LF, $sqlString);
```

```
}
```

at TYPO3\CMS\Core\Database\Schema\SqlReader->getTablesDefinitionString()

in **/var/www/html/vendor/typo3/cms-install/Classes/Controller/MaintenanceController.php** line 230

```
    $messageQueue = new FlashMessageQueue('install');
```

```
        $suggestions = [];  
        try {  
            $sqlReader = $container->get(SqlReader::class);  
            $sqlStatements = $sqlReader->getCreateTableStatementArray($sqlReader->getTablesDefinitionString());  
            $schemaMigrationService = GeneralUtility::makeInstance(SchemaMigrator::class);  
            $addCreateChange = $schemaMigrationService->getUpdateSuggestions($sqlStatements);  
  
            // Aggregate the per-connection statements into one flat array
```

at TYPO3\CMS\Install\Controller\MaintenanceController->databaseAnalyzerAnalyzeAction(object(TYPO3\CMS\Core\Http\ServerRequest))

in **/var/www/html/vendor/typo3/cms-install/Classes/Middleware/Maintenance.php** line 231

```
                'Unknown action method ' . $action . ' in controller ' . $controllerName,  
                1505216027  
            );  
        }  
        $response = $controller->$action($request);  
    }  
  
    return $response;  
}
```

at TYPO3\CMS\Install\Middleware\Maintenance->process(object(TYPO3\CMS\Core\Http\ServerRequest), object(Psr\Http\Server\RequestHandlerInterface@anonymous/var/www/html/vendor/typo3/cms-core/Classes/Http/MiddlewareDispatcher.php:128\$e0a))

in **/var/www/html/vendor/typo3/cms-core/Classes/Http/MiddlewareDispatcher.php** line 111

```
    }  
  
    public function handle(ServerRequestInterface $request): ResponseInterface  
    {  
        return $this->middleware->process($request, $this->next);  
    }  
};  
}
```

at Psr\Http\Server\RequestHandlerInterface@anonymous/var/www/html/vendor/typo3/cms-core/Classes/Http/MiddlewareDispatcher.php:93\$e09->handle(object(TYPO3\CMS\Core\Http\ServerRequest))

in **/var/www/html/vendor/typo3/cms-core/Classes/Middleware/NormalizedParamsAttribute.php** line 41

```
    */  
    public function process(ServerRequestInterface $request, RequestHandlerInterface $handler):  
ResponseInterface  
    {  
        $request = $request->withAttribute('normalizedParams',  
NormalizedParams::createFromRequest($request));  
        return $handler->handle($request);  
    }  
}
```

at TYPO3\CMS\Core\Middleware\NormalizedParamsAttribute->process(object(TYPO3\CMS\Core\Http\ServerRequest), object(Psr\Http\Server\RequestHandlerInterface@anonymous/var/www/html/vendor/typo3/cms-core/Classes/Http/MiddlewareDispatcher.php:93\$e09))

in **/var/www/html/vendor/typo3/cms-core/Classes/Http/MiddlewareDispatcher.php** line 162

```

        if (!$middleware instanceof MiddlewareInterface) {
            throw new \InvalidArgumentException(get_class($middleware) . ' does not
implement ' . MiddlewareInterface::class, 1516821342);
        }
        return $middleware->process($request, $this->next);
    }
};
}
}
}

```

at Psr\Http\Server\RequestHandlerInterface@anonymous/var/www/html/vendor/typo3/cms-core/Classes/Http/MiddlewareDispatcher.php:128\$e0a->handle(object(TYPO3\CMS\Core\Http\ServerRequest))
in /var/www/html/vendor/typo3/cms-core/Classes/Http/MiddlewareDispatcher.php line 70

```

    * Invoke the middleware stack
    */
    public function handle(ServerRequestInterface $request): ResponseInterface
    {
        return $this->tip->handle($request);
    }

    /**
     * Seed the middleware stack with the inner request handler

```

at TYPO3\CMS\Core\Http\MiddlewareDispatcher->handle(object(TYPO3\CMS\Core\Http\ServerRequest))
in /var/www/html/vendor/typo3/cms-core/Classes/Http/AbstractApplication.php line 80

```

    public function handle(ServerRequestInterface $request): ResponseInterface
    {
        try {
            $response = $this->requestHandler->handle($request);
        } catch (ImmediateResponseException $exception) {
            $response = $exception->getResponse();
        }
        return $response;
    }

```

at TYPO3\CMS\Core\Http/AbstractApplication->handle(object(TYPO3\CMS\Core\Http\ServerRequest))
in /var/www/html/vendor/typo3/cms-install/Classes/Http/Application.php line 46

```

    public function handle(ServerRequestInterface $request): ResponseInterface
    {
        $this->initializeContext();
        $request = $request->withAttribute('applicationType',
SystemEnvironmentBuilder::REQUESTTYPE_INSTALL);
        return parent::handle($request)
            ->withHeader('X-Frame-Options', 'SAMEORIGIN');
    }

    /**

```

at TYPO3\CMS\Install\Http/Application->handle(object(TYPO3\CMS\Core\Http\ServerRequest))
in /var/www/html/vendor/typo3/cms-core/Classes/Http/AbstractApplication.php line 92

```

    * Set up the application and shut it down afterwards
    */
    final public function run()
    {

```

```
        $response = $this->handle(ServerRequestFactory::fromGlobals());
        $this->sendResponse($response);
    }
}
```

at TYPO3\CMS\Core\Http\AbstractApplication->run()

in **/var/www/html/public/typo3/install.php** line 19

```
call_user_func(static function () {
    $classLoader = require dirname(dirname(__DIR__)).'/vendor/autoload.php';
    \TYPO3\CMS\Core\SystemEnvironmentBuilder::run(1, \TYPO3\CMS\Core\SystemEnvironmentBuilder::REQUESTTYPE_INSTALL);
    \TYPO3\CMS\Core\Core\Bootstrap::init($classLoader, true)->get(\TYPO3\CMS\Install\HttpApplication::class)->run();
});
```

at {closure}()

in **/var/www/html/public/typo3/install.php** line 20

```
call_user_func(static function () {
    $classLoader = require dirname(dirname(__DIR__)).'/vendor/autoload.php';
    \TYPO3\CMS\Core\SystemEnvironmentBuilder::run(1, \TYPO3\CMS\Core\SystemEnvironmentBuilder::REQUESTTYPE_INSTALL);
    \TYPO3\CMS\Core\Core\Bootstrap::init($classLoader, true)->get(\TYPO3\CMS\Install\HttpApplication::class)->run();
});
```