

Uncaught Exception in Flow

last objectManager->get where Security\Context is not the DependencyProxy

Exception thrown in file

C:\Users\afceder\PhstormProjects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3.Flow\Object\ObjectManager.php in line 185.

[Go to the FORGE issue tracker and report the issue](#) - if you think it is a bug!

```
34 TYPO3\Flow\Object\ObjectManager::get("TYPO3\Flow\Security\Context")

C:\Users\afceder\PhstormProjects\acme\Distribution\Data\Temporary\Development\Cache\Code\Flow_Object_Classes\TYPO3_Flow_Security_Aspect_PersistenceQueryRewritingAspect.php:
00607: $this->securityContext = \TYPO3\Flow\Core\Bootstrap::StaticObjectManager->getLazyDependencyByHash('48836470c14129ade5f39e28c4816673', $securityContext_reference);
00608: if ($this->securityContext === NULL) {
00609:     $this->securityContext = \TYPO3\Flow\Core\Bootstrap::StaticObjectManager->createLazyDependency('48836470c14129ade5f39e28c4816673', $securityContext_reference, 'TYPO3\Flow\Security\Context', function() { return \TYPO3\Flow\Core\Bootstrap::StaticObjectManager->get('TYPO3\Flow\Security)
00610:     }
00611:     $session_reference = $this->session;

33 TYPO3\Flow\Security\Aspect\PersistenceQueryRewritingAspect::TYPO3\Flow\Security\Aspect\{closure}()

C:\Users\afceder\PhstormProjects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3.Flow\Object\DependencyInjection\Proxy.php:
00056: /*
00057: public function activateDependency() {
00058:     $realDependency = $this->builder->_invoke();
00059:     foreach($this->propertyVariables as $propertyVariable) {
00060:         $propertyVariable = $realDependency;

32 Closure::_invoke()

C:\Users\afceder\PhstormProjects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3.Flow\Object\DependencyInjection\Proxy.php:
00056: /*
00057: public function activateDependency() {
00058:     $realDependency = $this->builder->_invoke();
00059:     foreach($this->propertyVariables as $propertyVariable) {
00060:         $propertyVariable = $realDependency;

31 TYPO3\Flow\Object\DependencyInjection\Proxy::activateDependency()

C:\Users\afceder\PhstormProjects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3.Flow\Object\DependencyInjection\Proxy.php:
00093: /*
00094: public function __call($methodName, array $arguments) {
00095:     $actualDependency = $this->activateDependency();
00096:     if ($actualDependency instanceof $this) {
00097:         throw new \TYPO3\Flow\Object\Exception("The DependencyProxy representing '" . $this->className . "' is again a DependencyProxy which must never occur.");

30 TYPO3\Flow\Object\DependencyInjection\Proxy::__call("isInitialized", array(0))

C:\Users\afceder\PhstormProjects\acme\Distribution\Data\Temporary\Development\Cache\Code\Flow_Object_Classes\TYPO3_Flow_Security_Aspect_PersistenceQueryRewritingAspect.php:
00142: $result = $joinPoint->getAdviceChain()->proceed($joinPoint);
00143: }
00144: if ($this->securityContext->isInitialized() === FALSE) {
00145:     if ($this->securityContext->canBeInitialized() === TRUE) {
00146:         $this->securityContext->initialize();

29 TYPO3\Flow\Object\DependencyInjection\Proxy::isInitialized()

C:\Users\afceder\PhstormProjects\acme\Distribution\Data\Temporary\Development\Cache\Code\Flow_Object_Classes\TYPO3_Flow_Security_Aspect_PersistenceQueryRewritingAspect.php:
00142: $result = $joinPoint->getAdviceChain()->proceed($joinPoint);
00143: }
00144: if ($this->securityContext->isInitialized() === FALSE) {
00145:     if ($this->securityContext->canBeInitialized() === TRUE) {
00146:         $this->securityContext->initialize();

28 TYPO3\Flow\Security\Aspect\PersistenceQueryRewritingAspect_Original::checkAccessAfterFetchingAnObjectByIdentifier(TYPO3\Flow\Aop\JoinPoint)

C:\Users\afceder\PhstormProjects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3.Flow\Aop\Advice\AroundAdvice.php:
00032: $adviceObject = $this->objectManager->get($this->aspectObjectName);
00033: $methodName = $this->adviceMethodName;
00034: return $adviceObject->$methodName($joinPoint);
00035: }
00036: }

27 TYPO3\Flow\Aop\Advice\AroundAdvice::invoke(TYPO3\Flow\Aop\JoinPoint)

C:\Users\afceder\PhstormProjects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3.Flow\Aop\Advice\AdviceChain.php:
00053: $this->adviceIndex++;
00054: if ($this->adviceIndex < count($this->advices)) {
00055:     $result = $advices[$this->adviceIndex]->invoke($joinPoint);
00056: } else {
00057:     $result = $joinPoint->getProxy()->Flow_Aop_Proxy_invokeJoinpoint($joinPoint);

26 TYPO3\Flow\Aop\Advice\AdviceChain::proceed(TYPO3\Flow\Aop\JoinPoint)

C:\Users\afceder\PhstormProjects\acme\Distribution\Data\Temporary\Development\Cache\Code\Flow_Object_Classes\TYPO3_Flow_Persistence_Doctrine_PersistenceManager.php:
00489: $adviceChain->rewind();
00490: $joinPoint = new \TYPO3\Flow\Aop\JoinPoint($this, 'TYPO3\Flow\Persistence\Doctrine\PersistenceManager', 'getObjectByIdentifier', $methodArguments, $adviceChain);
00491: $result = $adviceChain->proceed($joinPoint);
00492: }
00493: } catch (\Exception $e) {

25 TYPO3\Flow\Persistence\Doctrine\PersistenceManager::getObjectByIdentifier("225d8c7f-clc8-4291-be95-38512912ad9c", "TYPO3\Flow\Security\Account", TRUE)
```

```
C:\Users\afceder\PhpstormProjects\acme\Distribution\Data\Temporary\Development\Cache\Code\Flow_Object_Classes\TYPO3_Flow_Security_Authentication_Token_UsernamePassword.php:
00116: $persistenceManager = \TYPO3\FLOW\Core\Bootstrap::$staticObjectManager->get('TYPO3\FLOW\Persistence\PersistenceManagerInterface');
00117: foreach ($this->Flow\Persistence_RelatedEntities as $entityInformation) {
00118:     $entity = $persistenceManager->getObjectByIdentifier($entityInformation['identifier'], $entityInformation['entityType'], TRUE);
00119:     if (!isset($entityInformation['entityPath'])) {
00120:         $this->$entityInformation['propertyName'] = \TYPO3\FLOW\Utility\Arrays::setValueByPath($this->$entityInformation['entityPath'], $entity);
    }

24 TYPO3\FLOW\Security\Authentication\Token\UsernamePassword::__wakeup()

23 unserialize("a:13:{i:0;O:36:"TYPO3\FLOW\Mvc\FlashMessageContainer\ViewHelpers\Widget\RatingAggregateViewHelper";}}");

C:\Users\afceder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3\FLOW\Cache\Frontend\VariableFrontend.php:
00082:     return FALSE;
00083:     } else {
00084:         return ($this->useIgBinary === TRUE) ? igbinary_unserialize($rawResult) : unserialize($rawResult);
00085:     }
00086: }

22 TYPO3\FLOW\Cache\Frontend\VariableFrontend::get("335b9168-34b2-4f63-a143-49d045199b2a573091c8c4abc13135b03c6da4f6c05a")

C:\Users\afceder\PhpstormProjects\acme\Distribution\Data\Temporary\Development\Cache\Code\Flow_Object_Classes\TYPO3_Flow_Session_Session.php:
00294:     $this->started = TRUE;
00295: }
00296: $sessionObjects = $this->$cache->get($this->storageIdentifier . md5('TYPO3_Flow_Object_ObjectManager'));
00297: if (!is_array($sessionObjects)) {
00298:     foreach ($sessionObjects as $object) {

21 TYPO3\FLOW\Session\Session_Original::resume()

C:\Users\afceder\PhpstormProjects\acme\Distribution\Data\Temporary\Development\Cache\Code\Flow_Object_Classes\TYPO3_Flow_Session_Session.php:
00931:     $this->Flow_Aop_Proxy_fixMethodsAndAdvicesArrayForDoctrineProxies();
00932:     if (!isset($this->Flow_Aop_Proxy_methodIsInAdviceMode['resume'])) {
00933:         $result = parent::resume();
00934:     }
00935:     } else {

20 TYPO3\FLOW\Session\Session::resume()

19 call_user_func_array(array(2), array(0))

C:\Users\afceder\PhpstormProjects\acme\Distribution\Data\Temporary\Development\Cache\Code\Flow_Object_Classes\TYPO3_Flow_Session_Session.php:
00865: if (!class_exists($joinPoint->getClassName())) return parent::Flow_Aop_Proxy_invokeJoinPoint($joinPoint);
00866: if (!isset($this->Flow_Aop_Proxy_methodIsInAdviceMode[$joinPoint->getMethodName()])) {
00867:     return call_user_func_array(array('self', $joinPoint->getMethodName(), $joinPoint->getMethodArguments()));
00868: }
00869: }

18 TYPO3\FLOW\Session\Session::Flow_Aop_Proxy_invokeJoinPoint(TYPO3\FLOW\Aop\JoinPoint)

C:\Users\afceder\PhpstormProjects\acme\Distribution\Data\Temporary\Development\Cache\Code\Flow_Object_Classes\TYPO3_Flow_Session_Session.php:
00941:     $joinPoint = new \TYPO3\FLOW\Aop\JoinPoint($this, 'TYPO3\FLOW\Session\Session', 'resume', $methodArguments);
00942:     $result = $this->Flow_Aop_Proxy_invokeJoinPoint($joinPoint);
00943: }
00944:     $advices = $this->Flow_Aop_Proxy_targetMethodsAndGroupedAdvices['resume']['TYPO3\FLOW\Aop\Advice\AfterAdvice'];

17 TYPO3\FLOW\Session\Session::resume()

C:\Users\afceder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3\FLOW\Core\Booting\Scripts.php:
00482: static public function initializeSession(Bootstrap $bootstrap) {
00483:     if (FLOW_SAPITYPE === 'Web') {
00484:         $bootstrap->getObjectManager()->get('TYPO3\FLOW\Session\SessionInterface')->resume();
00485:     }
00486: }

16 TYPO3\FLOW\Core\Booting\Scripts::initializeSession(TYPO3\FLOW\Core\Bootstrap)

15 call_user_func(array(2), TYPO3\FLOW\Core\Bootstrap)

C:\Users\afceder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3\FLOW\Core\Booting\Step.php:
00047: */
00048: public function __invoke(\TYPO3\FLOW\Core\Bootstrap $bootstrap) {
00049:     call_user_func($this->callback, $bootstrap);
00050: }
00051: }

14 TYPO3\FLOW\Core\Booting\Step::__invoke(TYPO3\FLOW\Core\Bootstrap)

C:\Users\afceder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3\FLOW\Core\Booting\Sequence.php:
00099: $bootstrap->getSignalSlotDispatcher()->dispatch(__CLASS__, 'beforeInvokeStep', array($step, $this->identifier));
00100: $identifier = $step->getIdentifier();
00101: $step->$bootstrap->getSignalSlotDispatcher()->dispatch(__CLASS__, 'afterInvokeStep', array($step, $this->identifier));
00102: $bootstrap->getSignalSlotDispatcher()->dispatch(__CLASS__, 'afterInvokeStep', array($step, $this->identifier));
00103: if (!isset($this->steps[$identifier])) {

13 TYPO3\FLOW\Core\Booting\Sequence::invokeStep(TYPO3\FLOW\Core\Booting\Step, TYPO3\FLOW\Core\Bootstrap)

C:\Users\afceder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3\FLOW\Core\Booting\Sequence.php:
00103: if (!isset($this->steps[$identifier])) {
00104:     foreach ($this->steps[$identifier] as $followingStep) {
00105:         $this->invokeStep($followingStep, $bootstrap);
    }
}
```

```
00106: }
00107: }

12 TYPO3\FLOW\CORE\BOOTING\Sequence::invokeStep(TYPO3\FLOW\CORE\BOOTING\Step, TYPO3\FLOW\CORE\BOOTSTRAP)

C:\Users\afeder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3\FLOW\Classes\TYPO3\FLOW\CORE\BOOTING\Sequence.php:
00103: if (isset($this->steps[$identifier])) {
00104:     foreach ($this->steps[$identifier] as $followingStep) {
00105:         $this->invokeStep($followingStep, $bootstrap);
00106:     }
00107: }

11 TYPO3\FLOW\CORE\BOOTING\Sequence::invokeStep(TYPO3\FLOW\CORE\BOOTING\Step, TYPO3\FLOW\CORE\BOOTSTRAP)

C:\Users\afeder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3\FLOW\Classes\TYPO3\FLOW\CORE\BOOTING\Sequence.php:
00103: if (isset($this->steps[$identifier])) {
00104:     foreach ($this->steps[$identifier] as $followingStep) {
00105:         $this->invokeStep($followingStep, $bootstrap);
00106:     }
00107: }

10 TYPO3\FLOW\CORE\BOOTING\Sequence::invokeStep(TYPO3\FLOW\CORE\BOOTING\Step, TYPO3\FLOW\CORE\BOOTSTRAP)

C:\Users\afeder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3\FLOW\Classes\TYPO3\FLOW\CORE\BOOTING\Sequence.php:
00103: if (isset($this->steps[$identifier])) {
00104:     foreach ($this->steps[$identifier] as $followingStep) {
00105:         $this->invokeStep($followingStep, $bootstrap);
00106:     }
00107: }

9 TYPO3\FLOW\CORE\BOOTING\Sequence::invokeStep(TYPO3\FLOW\CORE\BOOTING\Step, TYPO3\FLOW\CORE\BOOTSTRAP)

C:\Users\afeder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3\FLOW\Classes\TYPO3\FLOW\CORE\BOOTING\Sequence.php:
00103: if (isset($this->steps[$identifier])) {
00104:     foreach ($this->steps[$identifier] as $followingStep) {
00105:         $this->invokeStep($followingStep, $bootstrap);
00106:     }
00107: }

8 TYPO3\FLOW\CORE\BOOTING\Sequence::invokeStep(TYPO3\FLOW\CORE\BOOTING\Step, TYPO3\FLOW\CORE\BOOTSTRAP)

C:\Users\afeder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3\FLOW\Classes\TYPO3\FLOW\CORE\BOOTING\Sequence.php:
00103: if (isset($this->steps[$identifier])) {
00104:     foreach ($this->steps[$identifier] as $followingStep) {
00105:         $this->invokeStep($followingStep, $bootstrap);
00106:     }
00107: }

7 TYPO3\FLOW\CORE\BOOTING\Sequence::invokeStep(TYPO3\FLOW\CORE\BOOTING\Step, TYPO3\FLOW\CORE\BOOTSTRAP)

C:\Users\afeder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3\FLOW\Classes\TYPO3\FLOW\CORE\BOOTING\Sequence.php:
00103: if (isset($this->steps[$identifier])) {
00104:     foreach ($this->steps[$identifier] as $followingStep) {
00105:         $this->invokeStep($followingStep, $bootstrap);
00106:     }
00107: }

6 TYPO3\FLOW\CORE\BOOTING\Sequence::invokeStep(TYPO3\FLOW\CORE\BOOTING\Step, TYPO3\FLOW\CORE\BOOTSTRAP)

C:\Users\afeder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3\FLOW\Classes\TYPO3\FLOW\CORE\BOOTING\Sequence.php:
00103: if (isset($this->steps[$identifier])) {
00104:     foreach ($this->steps[$identifier] as $followingStep) {
00105:         $this->invokeStep($followingStep, $bootstrap);
00106:     }
00107: }

5 TYPO3\FLOW\CORE\BOOTING\Sequence::invokeStep(TYPO3\FLOW\CORE\BOOTING\Step, TYPO3\FLOW\CORE\BOOTSTRAP)

C:\Users\afeder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3\FLOW\Classes\TYPO3\FLOW\CORE\BOOTING\Sequence.php:
00083: if (isset($this->steps['start'])) {
00084:     foreach ($this->steps['start'] as $step) {
00085:         $this->invokeStep($step, $bootstrap);
00086:     }
00087: }

4 TYPO3\FLOW\CORE\BOOTING\Sequence::invoke(TYPO3\FLOW\CORE\BOOTSTRAP)

C:\Users\afeder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3\FLOW\Classes\TYPO3\FLOW\HTTP\RequestHandler.php:
00162: protected function boot() {
00163:     $sequence = $this->bootstrap->buildRuntimeSequence();
00164:     $sequence->invoke($this->bootstrap);
00165: }
00166: }

3 TYPO3\FLOW\HTTP\RequestHandler::boot()

C:\Users\afeder\PhpstormProjects\acme\Distribution\Packages\Framework\TYPO3\FLOW\Classes\TYPO3\FLOW\HTTP\RequestHandler.php:
00114: $this->response = new Response();
00115:
00116: $this->boot();
00117: $this->resolveDependencies();
00118: \TYPO3\FLOW\Var_dump(get_class($this->securityContext), 'in handleRequest, ' . __LINE__);
```

```
2 TYPO3\FLOW\Http\RequestHandler::handleRequest ()
C:\Users\afeder\phpstorm\Projects\acme\Distribution\Packages\Framework\TYPO3.Flow\Classes\TYPO3.Flow\Core\Bootstrap.php:
00111:
00112:     $this->activeRequestHandler = $this->resolveRequestHandler();
00113:     $this->activeRequestHandler->handleRequest();
00114: }
00115:
1 TYPO3\FLOW\Core\Bootstrap::run()
C:\Users\afeder\phpstorm\Projects\acme\Distribution\Web\index.php:
00025: $context = getenv('FLOW_CONTEXT') ? : (getenv('REDIRECT_FLOW_CONTEXT') ? : 'Development');
00026: $bootstrap = new \TYPO3\FLOW\Core\Bootstrap($context);
00027: $bootstrap->run();
00028:
00029: ?>
```