

Jupiter - Zadanie #1731

Zadanie # 1694 (Rozwiązany): Obsługa pełnych tekstów

obsłużyć publikacje załączników infonowych

2017-06-08 09:26 - Łukasz Waśkiewicz

Status:	Rozwiązany	Start date:	2017-06-09
Priority:	Normalny	Due date:	2017-06-09
Assignee:	Michał Komar	% Done:	100%
Category:	backend	Estimated time:	0.00 hour
Target version:	0.3	Tester:	Michał Komar
Developer:	Łukasz Waśkiewicz		
Description			
nie pobierać zawartości załącznika jeśli się nie zmienił			
w innych miejscach pobierać zawartość tylko jeśli jest potrzebna			
Related issues:			
Related to Jupiter - Zadanie # 1809: Optymalizacja komunikacji z infoną podcz...		Nowy	2017-08-22
Follows Jupiter - Zadanie # 1730: Nie zapisywać danych załączników z infony w...		Rozwiązany	2017-06-08

Associated revisions

Revision 74fd58db - 2017-08-03 15:15 - Łukasz Waśkiewicz

refs #1731 poprawa mapowania document attachemntEntity-> Bean

History

#1 - 2017-06-08 09:26 - Łukasz Waśkiewicz

- Follows Zadanie #1730: Nie zapisywać danych załączników z infony w jupiterze added

#2 - 2017-06-08 12:18 - Łukasz Waśkiewicz

- Target version set to 0.3

#3 - 2017-06-23 12:02 - Michał Komar

W ramach commita z #1717 poprawiłem mapowania - integrationdocument nie tzyma już binarnej wartości pliku. Trzeba popracować nad mapowaniem pól:

- identifier - id dokumentu
- attachmentPath - path w ycontentfile
- fileId - id z ycontentfile
- attachmentId - id z attachment

Wydaje mi się, że trzeba to uspoźnić w ten sposób, żeby ostatecznie attachmentPath, fileId i attachmentId były takie same. W ten sposób unikniemy błędów. Niestety trzeba też obsłużyć istniejące elementy, które nie mają ujednoczonej postaci identyfikatorów. W takim przypadku należy do momentu zapisu posługiwać się attachmentPath, a dopiero przy zapisie / publikacji ujednoczyć identyfikatory.

#4 - 2017-07-05 13:05 - Łukasz Waśkiewicz

- Status changed from Nowy to Testowanie
- Assignee changed from Łukasz Waśkiewicz to Michał Komar
- % Done changed from 0 to 100

documentId - niepersystowane, ustawiane przy mapowaniu, używane jest dla infony

pozostałe identyfikatory zastąpione **location**

jeśli jest to upload location odpowiada FileId i pozostaje bez zmiany przy zapisaniu do Jupiter, można ew. usupólnić i w TempFileStorage używać location zamiast fileId

natomiast załączniki z infony location z YContentFile -> location

może być problem dla testowych załączników już zapisanych w jupiterze (cocos lub lokalny), nie pisałem skryptów kasujących takie, bo najlepiej było by zdropować bazę bo serwery już zawierają dziwne dane testowe

#5 - 2017-07-24 08:49 - Michał Komar

- Status changed from Testowanie to Odpowiedź

- Assignee changed from Michał Komar to Lukasz Waśkiewicz

Leci błąd przy zapisywaniu infonowych załączników. Spowodowany jest brakiem documentId w DocumentAttachmentBean, przez co nie może wyszukać parta w infonie.

```
java.lang.NullPointerException at
pl.edu.icm.jupiter.integration.services.JupiterSynatStorageIntegrationService.findPartData(JupiterSynatStorageIntegrationService.java:144) at
pl.edu.icm.jupiter.integration.services.JupiterSynatStorageIntegrationService$$FastClassBySpringCGLIB$$16c3869e.invoke(<generated>) at
org.springframework.cglib.proxy.MethodProxy.invoke(MethodProxy.java:204) at
org.springframework.aop.framework.CglibAopProxy$CglibMethodInvocation.invokeJoinpoint(CglibAopProxy.java:738) at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:157) at
net.bull.javamelody.MonitoringSpringInterceptor.invoke(MonitoringSpringInterceptor.java:73) at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:179) at
org.springframework.aop.interceptor.ExposeInvocationInterceptor.invoke(ExposeInvocationInterceptor.java:92) at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:179) at
org.springframework.aop.framework.CglibAopProxy$DynamicAdvisedInterceptor.intercept(CglibAopProxy.java:673) at
pl.edu.icm.jupiter.integration.services.JupiterSynatStorageIntegrationService$$EnhancerBySpringCGLIB$$a56273b0.findPartData(<generated>) at [sun.re
>) at
sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method) at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62) at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43) at
java.lang.reflect.Method.invoke(Method.java:498) at
org.springframework.aop.support.AopUtils.invokeJoinpointUsingReflection(AopUtils.java:333) at
org.springframework.aop.framework.ReflectiveMethodInvocation.invokeJoinpoint(ReflectiveMethodInvocation.java:190) at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:157) at
org.springframework.cache.interceptor.CacheInterceptor$1.invoke(CacheInterceptor.java:52) at
org.springframework.cache.interceptor.CacheAspectSupport.invokeOperation(CacheAspectSupport.java:345) at
org.springframework.cache.interceptor.CacheAspectSupport.execute(CacheAspectSupport.java:408) at
org.springframework.cache.interceptor.CacheAspectSupport.execute(CacheAspectSupport.java:327) at
org.springframework.cache.interceptor.CacheInterceptor.invoke(CacheInterceptor.java:61) at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:179) at
org.springframework.aop.framework.JdkDynamicAopProxy.invoke(JdkDynamicAopProxy.java:213) at
com.sun.proxy.$Proxy168.findPartData(Unknown Source) at
pl.edu.icm.jupiter.services.storage.attachments.InfonaDocumentAttachmentResolver.resolveData(InfonaDocumentAttachmentResolver.java:27) at [pl.edu.i
) at
pl.edu.icm.jupiter.services.storage.attachments.DocumentAttachmentResolveManagerImpl.resolveData(DocumentAttachmentResolveManagerImpl.java:25
rImpl.java:25) at
pl.edu.icm.jupiter.services.integration.mapper.CurrentDocumentToIntegrationDocumentConverter.lambda$2(CurrentDocumentToIntegrationDocumentConv
cumentConverter.java:48) at
java.util.stream.Collectors.lambda$toMap$58(Collectors.java:1321) at
```

```
java.util.stream.ReduceOps$3ReducingSink.accept(ReduceOps.java:169) at
java.util.stream.ReferencePipeline$2$1.accept(ReferencePipeline.java:175) at
java.util.ArrayList$ArrayListSpliterator.forEachRemaining(ArrayList.java:1374) at
java.util.stream.AbstractPipeline.copyInto(AbstractPipeline.java:481) at
java.util.stream.AbstractPipeline.wrapAndCopyInto(AbstractPipeline.java:471) at
java.util.stream.ReduceOps$ReduceOp.evaluateSequential(ReduceOps.java:708) at
java.util.stream.AbstractPipeline.evaluate(AbstractPipeline.java:234) at
java.util.stream.ReferencePipeline.collect(ReferencePipeline.java:499) at
pl.edu.icm.jupiter.services.integration.mapper.CurrentDocumentToIntegrationDocumentConverter.convertTo(CurrentDocumentToIntegrationDocumentConv
cumentConverter.java:47) at
pl.edu.icm.jupiter.services.integration.mapper.CurrentDocumentToIntegrationDocumentConverter.convertTo(CurrentDocumentToIntegrationDocumentConv
cumentConver
```

#6 - 2017-08-03 15:20 - Łukasz Waśkiewicz

- *Status changed from Odpowiedź to Testowanie*
- *Assignee changed from Łukasz Waśkiewicz to Michał Komar*

powinno działać (miałem problem ze znalezieniem zasobu z trunk-portal który ma załącznik [wszystkie bwmety źródłowe które patrzyłem mają location "yar://..."])

jest problem z zapisywanie jupiterowych załączników, ale to jest w kwestii #1775

#7 - 2017-08-18 08:57 - Michał Komar

Czekam, aż lukwas naprawi zapisywanie.

#8 - 2017-08-22 09:00 - Michał Komar

- *Related to Zadanie #1809: Optymalizacja komunikacji z infoną podczas procesu publikacji added*

#9 - 2017-08-22 10:58 - Michał Komar

- *Status changed from Testowanie to Rozwiązany*

Załączniki nie są pobierane automatycznie. Dzieje się to dopiero podczas zapisu lub publikacji.

OK