Управление версиями в CM6

# Термины

Документ - любой прикладной объект, для которого требуется поддержка версионирования;

Версия документа - прикладной объект, сохраняющий вариант прикладных данных документа ("реквизитов", включая контент) по инициативе пользователя или самой СЭД (в соответствии с её прикладной логикой);

Реквизиты Документа или Версии – прикладные атрибуты (поля) соответствующего «бизнес-объекта», реализуемого платформой AF5 как дерево связанных ДО.

# Основные требования

Основное назначение версии документа – фиксация в системе "слепка" его основных реквизитов. Решения по документу в СЭД всегда принимаются и документируются в системе относительно какого-то значимого состояния в его ЖЦ, которое и должно быть зафиксировано как версия.

Операция фиксации документа создает «актуальную» фиксированную версию документа, соответствующую его состоянию на момент фиксации.

Даже после фиксации (в зависимости только от прав доступа) документ может быть отредактирован и тогда его ранее созданная актуальная версия перестает быть таковой. Получается, что актуальной версии у документа может не быть, если его версии вообще не создавались, или если он отредактирован позже последней фиксации. Документ в таком состоянии можно считать «своей» редактируемой версией.

Всё это необходимо, чтобы при работе с документом можно было как редактировать его, так и выполнять какие-то работы, принимать решения и т.п., для которых документ является контекстом. Например, пользователь открыл документ, который направлен ему на подпись, и он на данный момент зафиксирован той версией, которую на подпись отправил автор. Подписант может подписать сразу, при этом новая запись о подписании будет связана с актуальной на данный момент версией документа. А может сначала отредактировать и сохранить документ, а потом подписать. Но тогда в процедуре подписания нужно зафиксировать его "почкованием" новой версии и ссылаться уже на неё.

С другой стороны, кроме изначально зафиксированных версий, создаваемых по реальным состояниям документа, возможны «альтернативные» версии, предлагаемые участниками подготовки или согласования документа, создаваемые не как «слепки», а скорее как «заготовки» или «вариации» для возможного в будущем включения в документ. Впрочем, в итоге они тоже должны фиксироваться и только тогда могут рассматриваться как «реальная альтернатива», то есть, с возможностью вернуть документ в состояние такой его версии.

Самый значимый пример применения к документам версионирования – это их Согласование. Документ, отправляемый на согласование, обязательно должен быть «зафиксирован», и конкретные решения (визы) принимаются согласующими относительно уже зафиксированной версии документа. С другой стороны, возможность для участника предложить в процессе согласования свою редакцию документа, может быть реализована как создание им «альтернативной» его версии – на основе согласуемой (но за основу можно взять и любую другую). Новая альтернативная версия создается как «черновик» с доступом только для автора-согласующего, но для включения в его визу как результат решения задачи согласования она обязательно должна быть зафиксирована.

# Решение

1. Сущности Документ и Версия реализуются как иерархия связанных ДО с корневым ДО соответствующего типа. Типы ДО Документа и его Версий должны быть «родственными». Как основной вариант, выбрано наследование типа Версия от типа Документ.   
   Но рассматривались и другие:   
   - Версия и Документ одного типа ДО;  
   - Версия и Документ – наследники одного общего базового абстрактного типа;  
   - Версия – базовый тип, Документ – наследник;  
   - Версия и Документ – разные типы.
2. Зафиксированная Версия никогда не изменяется по крайней мере в части прикладных данных.
3. Документ может редактироваться и сохраняться независимо от наличия и фиксированности его версий.
4. У Версии есть обязательная ссылка на Документ, в Документе эта ссылка пустая, а лучше – ссылка на самого себя, что может быть удобно при вычислениеях чего-либо по документу, но независимо из какого контекста – версии или документа (например, для контекстных ролей документа).
5. У Версии есть номер, генерируемый автоинкрементом в пределах множества Версий Документа.
6. И у Документа и у Версии есть необязательная ссылка "версия-источник" (VersionSource) на Версию, которая логически предшествовала текущему объекту.
7. Основные операции с Документами и Версиями:   
   - Фиксация Документа версией - создает новую фиксированную Версию по текущему состоянию Документа;  
   - Возврат Документа к состоянию одной из имеющихся Версий (по выбору пользователя);  
   - Создание новой "альтернативной" Версии "с нуля" или на основе одной из имеющихся;   
   - Сброс актуальности версии-источника (при редактировании и сохранении Документа);  
   - …
8. Сразу после фиксации Документа или возврата Документа к выбранной Версии его реквизиты равны реквизитам его соответствующей Версии. И это актуальное (соответствующее) состояние пары Документ = Версия (по ссылке VersionSource) отслеживается установкой специального «флага актуальности версии" и в Документе и в Версии. Флаг актуальности версии автоматически сбрасывается (и в Версии и в Документе) в момент сохранения Документа после редактирования.
9. Сбрасывать флаг актуальности версии Документа следует только при "существенных" его модификациях, то есть, изменениях тех его реквизитов (назовем их контрольными), для которых действительно важно версионирование (по умолчанию можно ограничиться "Содержанием"). Этим можно управлять, например, через указание таких реквизитов в конфигурации.
10. Также в конфигурации, возможно, следует указывать реквизиты Документа, сохраняемые в Версии (включая вложения), но по умолчанию можно считать, что это все реквизиты Документа. Контрольные реквизиты (см. в предыдущем пункте) - это подмножество сохраняемых реквизитов.
11. Системный Статус Документа, от которого зависят права доступа, для Документа устанавливается как обычно - в соответствии с прикладной логикой. Режим закрытия Документа от редактирования с точки зрения механизма версионирования не нужен. Возможность редактирования документа определяется правами доступа к нему в соответствии с текущим статусом, который в свою очередь зависит от бизнес-процессов, происходящих с документом.
12. Для фиксации Версии достаточно одного статуса "VersionFixed", при котором права к Версии предоставляются только на чтение. Таким образом, поле Статус не копируется из Документа в Версию при ее создании, но сохранить в Версии его текущее значение может быть полезно (для истории) - в дополнительном поле. Для «альтернативных» Версий возможны дополнительные статусы и права доступа.
13. Прикладные объекты, имеющие смысл каких-либо принимаемых решений по Документу (согласования, подписания, утверждения,...) могут быть привязаны только к фиксированной Версии. Если действия, создающие такие объекты, выполняются в контексте самого Документа, в создаваемые объекты-решения должна прописываться ссылка на актуальную Версию Документа. Если в этот момент такой версии нет (флаг актуальности версии не установлен), то по Документу создается новая Версия, прописывается как актуальная в Документ и всё-таки передается для включения в новое решение. Это основное правило автоматического порождения версий. Версия создается при условии наличия в Документе важных изменений в моменты вызова операций, фиксирующих в системе решения, принятые по документу. Надо бы как-то задавать в конфигурации, для каких операций и/или связанных объектов должно действовать такое правило.
14. С другой стороны, решения не насчет Документа, а в его контексте, например, резолюции/поручения могут всё-таки привязываться к самому Документу ???
15. Состав вложений может меняться от версии к версии, одно и то же вложение может модифицироваться в более новой версии, а также вложение может оставаться неизменным в следующей версии. В последнем случае желательно не создавать свою копию файла вложения для каждой версии. Это должно поддерживаться подсистемой управления вложениями.

## Поля «Статус» и «Состояние» у Документа и Версии

* Статус – особое поле (в соответствии с принципами платформы AF5), выделяющее текущую стадию в ЖЦ объекта, в зависимости от которой определяются права доступа в соответствии с конфигурацией.
* Состояние – дополнительное «прикладное» поле объекта, отслеживающее текущее состояние среди его промежуточных состояний в пределах одного статуса (стадии ЖЦ).

Например (в формате Статус [сосотояние1, состояние 2,...]):

* Черновик [на редактировании];
* Проект [на редактировании, на согласовании, согласование завершено, на подписи, подписан/утвержден];
* Документ [зарегистрирован, исполнен].

В состояние "на редактировании" проект попадает с момента создания или из Черновика. А еще, если Документ начинает редактироваться из какого-то другого состояния, например, "согласование завершено", то при сохранении его состояние меняется в "на редактировании".

### С учётом версионирования

У Версии свой «статус», т.к. ЖЦ и права доступа у сущности Версия другие. Но в Версии Документа необходимо отслеживать статус Документа, для этого в Версии предусматриваем другое поле (VersionDocStatus). Далее статусом Версии сокращенно называем именно его.

Правила назначения статуса в связке Документ – Версии:

* При создании Версии по Документу поле статус Документа копируется в VersionDocStatus Версии.
* У актуальной Версии VersionDocStatus синхронизируется со статусом Документа.
* При создании Версии по Версии поле VersionDocStatus копируется.
* Возврат Документа к выбранной Версии разрешается только для Версий той же стадии ЖЦ, что и Документ (с тем же "статусом", то есть, равными значениями полей статус Документа и VersionDocStatus Версии).

У Версии должно быть поле «Состояние», одноименное с Документом   
(системное имя поля надо уточнить).

Поле «Состояние» Документа/Версии велет себя следующим образом:

* При создании Версии по Документу поле «Состояние» копируется из Документа.
* В актуальной Версии поле «Состояние» поддерживается одинаковым с Документом:   
  1) изменение только «Состояния» не делает документ/версию неактуальными;   
  2) такое изменение прописывается синхронно в обоих.
* При возврате Документа к выбранной Версии поле «Состояние» берется из Версии в Документ. Поскольку возврат к другой Версии разрешается только для Версий с тем же статусом, проблем быть не должно.
* Уточнение про состояние "на редактировании": если Документ, имея актуальную версию, начинает редактироваться, при его сохранении вместе со сбросом статуса актуальности версии у него (и только у него) задается состояние "на редактировании".

### Пример – согласование документа

1. Согласование начинается с актуальной парой Документ=Версия, но контекстом считается Версия, и все данные конкретного экземпляра согласования остаются навсегда привязанным к этой Версии.
2. В начале процесса обоим ставится состояние "На согласовании". По завершении процесса обоим - "Согласование завершено". Если бывают еще какие-то вариации, например, «Согласование прекращено», то аналогично.
3. Но! Если прямо в процессе, а особенно после него, Документ отредактируют, Версия потеряет актуальность, и у нее останется свое состояние (последнее на момент актуальности, т.е. соответствия документу), а у Документа станет свое "на редактировании".
4. Однако, чисто изменение состояния Документа, не отключает актуальность версии. Поэтому отправка его на подпись ставит состояние "На подписании" обоим! А подписание тоже будет отмечено и в Версии и в Документе, но объекты Подпись (или Утверждение) будут привязаны к Версиям, как и Визы.
5. Если вдруг решат, что подписали какую-то хрень, но есть более ранняя правильная Версия, можно вернуть в Документ Версию с любым состоянием в статусе Проект (как статус у Документа), например, "согласование завершено".
6. Допустим, Документ с актуальной Версией успешно подписан и отправлен на регистрацию. При отказе он останется Проектом, и можно вернуться на несколько шагов назад. Но при регистрации статус Документа меняется в Документ. И статус его актуальной Версии (VersionDocStatus) - тоже. Ну и состояния - аналогично.

## Прикладные поля с особым режимом версионирования

Некоторые реквизиты Документа в его ЖЦ могут задаваться не сразу, а по мере необходимости для текущих операций. Например, предположим, что согласование можно провести, не указывая Подписанта Документа, ведь список согласующих задается отдельно. Но для подписания его придется указать. Еще конкретнее ситуация с рег.номером, ведь заранее его указать вообще нельзя, он определится только в результате процедуры регистрации.

Но что делать, если такие поля являются важнейшими реквизитами Документа, а значит их надо считать «контрольными» (checked), и следовательно, ввод значения такого поля делает Документ/Версию неактуальными, что порождает новую актуальную версию на следующем этапе обработки.

Для таких полей следует включать особый режим контроля, при котором первичное указание значения (вместо исходного пустого значения) считается не изменением, а как-бы созданием.

Тогда ни впервые указанный подписант, ни присвоенный рег.номер сами по себе не приведут к размножению Версий документа. Но если вдруг их придётся изменять, тогда действительно нужны новые Версии.

Более того, когда такие поля впервые устанавливаются в Документе, они должны также задаваться и в актуальной Версии Документа! Тогда одну и ту же актуальную Версию можно будет сначала согласовать, потом подписать, да ещё и зарегистрировать. Но всё это только по одному разу!

## Примеры сценариев ЖЦ версионируемого Документа

Самый лёгкий вариант – с минимальным числом изменений, Версий, возвратов и т.п.

На старте имеем Документ (статус=проект), Версий пока нет.

1. Автор: создает для проекта Документа Лист согласования (ЛС)  
   (и сохраняет, можно заранее?)
2. Автор: инициирует процесс Согласования.
3. Система: фиксирует Документ созданием актуальной Версии №1, ...
4. Система: меняет «состояние» Документа и Версии1 в "На согласовании".
5. Система: создает экземпляр процесса Согласования, Контекст=Версия1.
6. Система: управляет ходом процесса согласно BPMN-модели и данным ЛС.
7. Система: по завершению процесса, если Версия1 актуальна Документу, опять обновляет «состояние» обоих. Иначе - только состояние Версии1 - контекстной для процесса.
8. Автор: в Документе указывает Подписанта (поле было пустым), сохраняет.
9. Система: определяет, что Подписанта можно задать, но нельзя изменить без потери актуальности версии, поэтому Документ/Версия1 остаются актуальными. Но следует обновить это поле и в Версии1.
10. Автор: отправляет Документ на подписание/утверждение.
11. Система: Поскольку у Документа пока всё та же актуальная Версия1, подписанию подлежит именно она, поэтому система создает ДО «Подпись/Утверждение» в контексте Версии1, а состояние «На подписи» ставит обоим.
12. Подписант: утверждает Документ.
13. Система: ставит Документу и Версии1 состояние «Подписан».
14. Автор: отправляет Документ на регистрацию.   
    Регистрация аналогична подписанию, но при успехе меняет не только «состояние», но и «статус» Документа и актуальной Версии1. А также и рег.номер и дату – и в Документе и в Версии1.

## Дополнительные атрибуты для Версии и Документа

* VersionParentDoc – ссылка в Версии на родительский Документ (а в Документе – на себя);
* VersionNumber – номер Версии;
* VersionSource – ссылка на Версию-источник;
* VersionActual – в Версии - флаг её актуальности, а в Документе – его VersionSource-Версии;
* VersionDocStatus (в Версии) – статус Документа при его фиксации данной версией (это не может быть просто status, т.к. он управляет правами доступа, разными у Документа и Версии).   
  Не путать «статус» с полем «состояние», одноименным в Документе и Версии.

## Расширение конфигурации

Для применения сервисов подсистемы версионирования к различным типам Документов следует особенности их структуры данных с учетом иерархии ДО, составляющих «бизнес-объект» Документа или Версии, описать в расширенной конфигурации.

### Состав конфигурации версионирования

* Тип версионируемого Документа (его корневого ДО)
* Соответствующий тип ДО Версии
* Правила обмена данными полей между ДО Документа и Версии
* Дочерние ДО (типы + поля-ссылки)  
  + правила обмена полями дочерних ДО
* Контрольные реквизиты: атрибуты и дочерние ДО+их атрибуты, разница между которыми означает различие Версий между собой или с Документом.
* Контрольные реквизиты, для которых первичная установка не понимается как различие Версий. Как бы их покороче назвать? Например, «checked-ignore-1st-set»?

Правила обмена данными полей: по-умолчанию - "один-в-один", но возможно с разными именами (как в конфигурации шаблонов, но только самые простые возможности).

### Пример конфигурации (по аналогии с шаблонами)

<versioning-rule object-type="Document" version-type="Document\_Version">

<fields copy-default="true" checked-default="false" >

<!-- Все поля, которые не упомянуты в этой конфигурации, копируются один в один -->

<!-- checked-default и checked - контрольные поля для версионирования -->

<copy object-field="Creator" checked="true" version-field="Author"/>

<!-- Если вдруг назовут по-разному поля в объекте и в версии -->

<skip object-field="Comment"/><!-- Вообще не копируется -->

</fields>

<child-objects>

<object type="Signature" parent-reference-field="Document" version-type="Document\_Signer">

<!-- Такая же структура, как и у тега template-rule -->

<fields copy-default="false" checked-default="true">

<!-- Не копировать по умолчанию - удобно, если объекты сильно отличаются -->

<copy object-field="Person"/>

<!-- version-field не нужен, если имя поля в объекте такое же, как в версии -->

</fields>

<child-objects>

<!-- Могут быть и свои вложенные объекты... -->

</child-objects>

</object>

</child-objects>

</versioning-rule>

## Алгоритмы основных операций

### Фиксация Документа созданием Версии:

1. Создает для Документа новый объект Версия
2. Копирует в Версию:   
   - прикладные поля ДО Документа  
   - ссылку на Документ
3. Сохраняет Версию (промежуточное сохранение со статусом "VersionDraft", чтобы далее создавать дочерние ДО). Если всё это выполнять под правами системы, то можно сразу ставить конечный статус "VersionFixed".
4. Записывает в Версию:   
   - копию всей иерархии "дочерних" ДО Документа  
   - копию ссылки VersionSource из Документа (она пуста для 1-й Версии)  
   - номер, увеличенный на 1 относительно максимального из имеющихся Версий Документа  
   - флаг актуальности версии = true
5. Сохраняет версию в фиксированном статусе "VersionFixed".
6. Прописывает в Документ:  
   - в VersionSource Документа ссылку на созданную Версию  
   - флаг актуальности версии = true
7. Сохраняет документ

Примечание насчет автоматической нумерации версий: во избежание конфликтов при параллельных транзакциях, возможно следует на время вычисления очередного уникального номера блокировать ДО Документа.

### Возврат Документа к состоянию одной из имеющихся Версий

Вызывается по желанию пользователя, имеющего право на редактирование Документа, если он хочет «вернуться» к одной из ранее зафиксированных его Версий.

1. Пользователь выбирает другую Версию-источник для Документа.
2. Контролируется VersionDocStatus Версии, можно брать только Версию со значением, равным статусу Документа.
3. Если у Документа была другая актуальная Версия, то у неё снимается флаг актуальности версии (VersionActual = false).
4. Если нет, т.е. Документ отличался от своей Версии-источника, то GUI предупреждает пользователя, что изменения в текущей редакции Документа будут потеряны.
5. Ставится флаг актуальности версии (VersionActual = true)  
   - в Документе;  
   - в выбранной Версии.
6. Из выбранной Версии в Документ копируются:   
   - прикладные поля корневого ДО;  
   - копия всей иерархии "дочерних" ДО.

. . .

### Создание новой "альтернативной" Версии

Новая «черновая» версия создается "с нуля" или на основе одной из имеющихся

. . .

### Сброс актуальности Версии-источника

Выполняется при сохранении Документа, имеющего флаг актуальности версии-источника, после модификации (редактированиия) его самого или его «дочерних» ДО.

1. Выполняется проверка состава измененных реквизитов Документа (его собственных полей, а также состава и полей его «дочерних» ДО).
2. Если изменилось хоть что-то, включенное в список «контрольных» согласно правилам версионирования, заданным в конфигурации, то, снимается флаг актуальности версии (VersionActual = false)  
   - в сохраняемом Документе;  
   - в актуальной Версии (по ссылке VersionSource из Документа).

Примечание по реализации: надо специально позаботиться, чтобы снятие флага актуальности и прочие (прикладные/пользовательские) изменения в Документе сохранились в одной транзакции, и чтобы при этом не нарваться на оптимистическую блокировку.