Netzwerk / SVN-Anbindung

Die Verzeichnisstruktur von DokuWiki kann in TortoiseSVN abgebildet werden. Neue Seiten werden geadded, Änderungen committet. Sollten Seiten später umbenannt werden, ist der vorgenannte Ablauf unbedingt zu beachten.

Mit Anlage von neuen Seiten und Überarbeitungen werden weitere Verzeichnisinhalte, z. B. der attic oder Medien bei Nutzung von integrierten Bildern inhaltliche Änderungen erfahren oder interne index-Dateien angepasst. Über das SVN-Archivflag gesteuert erfolgt eine Aktualisierung auf Basis aller Änderungen in DokuWiki-Unterverzeichnissen. Neben der Wiki-Source für das öffentliche DokuWiki (iX-Wiki-Trunk-Version) kann eine Neufassung gepflegt werden, die in separatem Bereich liegt (iX-Wiki-Branch-Version) Der Browserpfad srv-dev/xampp/DokuWiki/trunk/doku.php?id=berichtscenter:excel_berichte:system hat ein Pendant im Serverpfad: \\srv-dev\DokuWiki\Trunk\data\pages\berichtscenter\excel_berichte\system.txt ⇒ nach `id=` ist im Browserpfad der Pfad innerhalb des pages-Verzeichnisses auf dem Server abgebildet.

Bei jeder Aktualisierung kann in DokuWiki eine Bemerkung für die Wiki-Historie gesetzt werden. Auch wenn die Anzeige der Historie ausgeblendet ist, werden diese Informationen erfasst. Diese Historieninformation erfassen wir nicht dauerhaft, insofern nur bestimmte Dateien per SVN in TortoiseSVN archiviert werden! Beim SVN-commit wird eine Bemerkung für die SVN-Historie gesetzt. Hier ist zumindest die Wiki-Version als Auslöser zu benennen.

Mit der Branchvariante besteht im Intranet eine parallele Variante. Die Branchversion halten wir bis zu einem bestimmten Zeitpunkt mit der Trunk-Version parallel und werden sie dann gezielt weiterentwickeln. Hierzu planen wir eine Versorgung via TortoiseSVN und mergen nach bestimmten Regeln. Die Branchversion des iX-Wiki wird in der Aufbauphase nicht online gestellt. Dies ist der jeweiligen Trunkversion vorbehalten ⇒ iX-Wiki Onlineversion.

DokuWiki und Serveraktualisierung

Der verwendete Server (s. Kapitel Doku-Wiki-Voraussetzungen) unterstützt PHP und Dateien (Text-Dateien, CSS, PHP, Mediendateien, ...). Durch die Zugangssteuerung im DokuWiki sowie das administrative Ausblenden von nicht öffentlichen Namespaces können auch noch nicht öffentliche Anpassungen an den Webserver übertragen werden. Eine Übertragung an den Webserver ist erforderlich, wenn eine öffentliche Komponente aktualisiert wurde.

Zeitpunkt für Serveraktualisierung

Hierzu könnte eine Meldung in SVN erfolgen, die als Trigger für den FTP-Upload genutzt wird (spezieller Commit-Kommentar). Übertragen werden alle Dateien und Verzeichnisse, welche neu oder mit neuerem Datum im Quellverzeichnis vorliegen. Derzeit wird das Serverupdate manuell gestartet. Nur die Trunk-Version wird so öffentlich. Idealerweise sollte der Trunk wöchentlich aktualisiert werden, parallel zu den Patches. Wegen des manuellen Aufwands wird diese Aktualisierung nur auf Anfrage und nach Abstimmung mit der Redaktion erfolgen.

Die Frage, ob der Branch nicht am besten als Kopie des Trunk erzeugt und erst im letzten Quartal mit

neuerungsspezifischen Anpassungen und Erweiterungen ergänzt wird wurde zusammen mit SMV diskutiert. Das von SMV eingezogene Zwischenverzeichnis T1-Smv ist derzeit von den Redakteuren nicht in separaten Update- und Commit-Prozessen zu berücksichtigen, es wird automatisch bedient, wenn Update- und Commit-Prozesse für das Trunk-Verzeichnis (Version 20.20.0) ablaufen.

- Vorteile sind geringere Konfliktpotentiale bei Commit und Update.
- Nachteil ist begrenzteres Zeitfenster für diese redaktionellen Arbeiten
- Spezielle Anpassungen für Branch können in Branchverzeichnis generiert und per Commit gesichert werden. Alternativ können relevante Abschnitte im Trunk in einer `showif-Klammer` vorbereitet werden. Sie sollten dann entsprechend intern auffindbar kommentiert sein (für Volltextsuche vorbereitet). Beispiel:

<showif isloggedin>FIXME Version BRANCH 20.20.2 .... </showif>

Datei- und Verzeichnislöschungen

Dateilöschungen sollten durch entsprechend sorgfältiges Arbeiten zur Ausnahme gehören. Löschungen sind daher von Anfang an zu kommentieren und müssen manuell begleitet werden. Betrifft die Löschung einer Datei/eines Verzeichnisses eine schon upgeloadete Komponente (Datei/Verzeichnis), muss diese administrativ auch auf dem Server analog gelöscht werden.

SVN-Anbindung

Wiki-Versionen

Redaktionell arbeiten wir Neuerungen vornehmlich in der Branchversion im Intranet ein - vorbereitend für die Veröffentlichung von Servicepacks an spezielle Kunden und für das kommende Jahresrelease. Nur absolute Basics werden rückwirkend in Vorversionen, dann i. d. R. in der Trunk-Version angepasst. Diese müssen dann bis zum Branch durchgemerged werden. Durch Veröffentlichung des Servicepack 1 zu Version 20.21 Ende Juni 2021 wurde z. B. die Version 20.21.1 zur neuen Trunk-Version und die neue Branchversion 20.21.2 wurde eingerichtet. Eine Angabe der Kalenderwoche nach der Programmversion (letzer Upload von KW...) auf der Startseite kann intern helfen, zu erkennen welche Informationen ggf. zusätzlich per Patchnotes zu berücksichtigen sind, wenn man eine aktuellere Version einsetzt. Da Releasenotes inhaltlich synchron mit dem Wiki sind bzw. das Wiki aktueller als die Releasenotes, dient die aktuelle Patchnotes-Pdf als Ergänzung zu Releasenotes-Pdf. Beide liefern dem Anwender jeweils thematisch und kalendarisch sortierte Neuerungen. Das veröffentlichte iX-Wiki bietet hingegen die Hilfen zur Programmbedienung auf Basis des jeweiligen Release.

Internet und Intranet

Intranet	**Internet**	**aktuelle Ebene**
http://srv-dev/xampp/DokuWiki/Version.20.25.3/doku.php	keine Online-Variante	Branch
http://srv-dev/xampp/DokuWiki/Version.20.25.2/doku.php	https://wiki.crem-solutions.de/Version.20.25.2/doku.php	Trunk
http://srv-dev/xampp/DokuWiki/Version.20.25.1/doku.php	https://wiki.crem-solutions.de/Version.20.25.1/doku.php	Trunk-1
http://srv-dev/xampp/DokuWiki/Version.20.25.0/doku.php	https://wiki.crem-solutions.de/Version.20.25.0/doku.php	Trunk-2
http://srv-dev/xampp/DokuWiki/Version.20.24.3/doku.php	https://wiki.crem-solutions.de/Version.20.24.3/doku.php	Trunk-3
http://srv-dev/xampp/DokuWiki/Version.20.24.2/doku.php	https://wiki.crem-solutions.de/Version.20.24.2/doku.php	Trunk-4

Intranet	Internet	Status
http://srv-dev/xampp/DokuWiki/Version.20.25.2/doku.php	https://wiki.crem-solutions.de/Version.20.25.2/doku.php	20.25 SP2
http://srv-dev/xampp/DokuWiki/Version.20.25.1/doku.php	https://wiki.crem-solutions.de/Version.20.25.1/doku.php	20.25 SP1
http://srv-dev/xampp/DokuWiki/Version.20.25.0/doku.php	https://wiki.crem-solutions.de/Version.20.25.0/doku.php	20.25 Hauptversion
http://srv-dev/xampp/DokuWiki/Version.20.24.3/doku.php	https://wiki.crem-solutions.de/Version.20.24.3/doku.php	20.24 SP3
http://srv-dev/xampp/DokuWiki/Version.20.24.2/doku.php	https://wiki.crem-solutions.de/Version.20.24.2/doku.php	20.24 SP2
http://srv-dev/xampp/DokuWiki/Version.20.24.1/doku.php	keine Online-Variante	outdated
http://srv-dev/xampp/DokuWiki/Version.20.24.0/doku.php	https://wiki.crem-solutions.de/Version.20.24.0/doku.php	outdated
http://srv-dev/xampp/DokuWiki/Version.20.23.3/doku.php	https://wiki.crem-solutions.de/Version.20.23.3/doku.php	outdated
http://srv-dev/xampp/DokuWiki/Version.20.23.2/doku.php	https://wiki.crem-solutions.de/Version.20.23.2/doku.php	outdated
http://srv-dev/xampp/DokuWiki/Version.20.23.1/doku.php	https://wiki.crem-solutions.de/Version.20.23.1/doku.php	outdated
http://srv-dev/xampp/DokuWiki/Version.20.23.0/doku.php	https://wiki.crem-solutions.de/Version.20.23.0/doku.php	outdated
http://srv-dev/xampp/DokuWiki/Version.20.22.3/doku.php	https://wiki.crem-solutions.de/Version.20.22.3/doku.php	outdated
http://srv-dev/xampp/DokuWiki/Version.20.22.2/doku.php	https://wiki.crem-solutions.de/Version.20.22.2/doku.php	outdated
obsolete http://srv-dev/xampp/DokuWiki/Version.20.22.1/doku.php	https://wiki.crem-solutions.de/Version.20.22.1/doku.php	outdated
obsolete http://srv-dev/xampp/DokuWiki/Version.20.22.0/doku.php	https://wiki.crem-solutions.de/Version.20.22.0/doku.php	outdated
obsolete http://srv-dev/xampp/DokuWiki/Version.20.21.3/doku.php	https://wiki.crem-solutions.de/Version.20.21.3/doku.php	outdated
obsolete http://srv-dev/xampp/DokuWiki/Version.20.21.2/doku.php	keine Online-Variante	outdated
obsolete http://srv-dev/xampp/DokuWiki/Version.20.21.1/doku.php	keine Online-Variante	outdated
obsolete http://srv-dev/xampp/DokuWiki/Version.20.21.0/doku.php	keine Online-Variante	outdated
obsolete http://srv-dev/xampp/DokuWiki/Version.20.20.3/doku.php	keine Online-Variante	outdated
obsolete http://srv-dev/xampp/DokuWiki/Version.20.20.0/doku.php	keine Online-Variante	outdated

Datenhaltung iX-Wiki und SVN

Versionskontrolle allgemein

Durch die Verwendung von TortoiseSVN (Subversion) werden bestimmte Dateien aus dem Wiki-Verzeichnis über `Add` und `Commit` gesichert. Aufgrund der Abfragen und den damit verbunden Merge-Prozessen ist eine bestimmte Abfolge im Umgang mit TortoiseSVN für die iX-Wiki-Varianten zu beachten. Hauptansprechpartner für die SVN-Konfiguration ist SMV.

Da innerhalb von DokuWiki, dem von iX-Wiki verwendeten Wikisystem, ebenfalls eine Versionskontrolle über das php-System operiert, ist zwischen Versionskonflikten in TortoiseSVN und Versionskonflikten in DokuWiki zu unterscheiden. Da wir nur bestimmte Dateien des DokuWikisystems in TortoiseSVN betrachten und nur für diese eine Versionkontrolle vorsehen, ist es möglich, dass spezielle DokuWikifunktionen hierdurch unterbrochen oder nicht optimal ausgeführt werden können. Solange diese Funktionen ebenfalls nur eine Wiki-interne Versionkontrolle betreffen, sehen wir das als unkritisch an. Im laufenden Betrieb sind bislang praktisch keine Versionskonflikte in Dokuwiki aufgetreten.

Welche Dateien oder Verzeichnisse von iX-Wiki unterliegen in TortoiseSVN einer Versionskontrolle?

Versionsspezifische Sicherung

Welche Dateien sollten trunk/branch-spezifisch sein?

- Seiten: data\pages
- Suchindex zu Seiten: data\index
- Medien: data\media
- Plugins in Testphase
- Neuerungen in Branch (inhaltliche Dateiunterschiede ⇒ Differenzierung ist via TortoiseSVN und update/commit-Workflow zu managen

Für eine versionsspezifische Sicherung sind auf jeden Fall `data\pages`, `data\media` relevant zzgl. der Inhalte der Verzeichnisse `conf` und `lib`. Die Konfigurationsdateien aus dem Verzeichnis `conf` sowie der Plugins `lib\plugins` beinhalten ggf. individualisierte Daten, die ebenfalls gesichert werden müssen und im Zweifelsfall versionsspezifisch (trunk oder branch) sind. Wird z. B. ein Plugin im Trunk installiert und liefert spezifische Codes im Seitentext, können diese Codes in der Branch-Variante nur dann sicher interpretiert werden, wenn das gleiche Plugin dort ebenfalls mit der gleichen Konfiguration eingerichtet ist! Daher ist die Liste der verwendeten Plugins sinnvoll, um den Überblick über die das DokuWiki ergänzenden Features zu behalten. Plugins sollten nach einer Testphase möglichst immer in beiden Versionen parallel installiert werden.

SVN-Workflow

Nur Branch-Bearbeitung

Trunk- und Branch-Bearbeitung

TortoiseSVN-Eigenschaften (Beispiele)

TortoiseSVN-Eigenschaften TRUNK

TortoiseSVN-Eigenschaften BRANCH

Konflikte in TortoiseSVN auflösen

Bei Mergekonflikten in der Error-E-Mail auf `Antworten` klicken und in dem Mail-Editor dann mit `Strg` + F nach „conflict(s)“ suchen. Dort sind dann die Dateien benannt, welche konfliktbehaftet sind. Nachfolgende Infos wurde aus der SVN-Hilfe entnommen…

Ab und an werden Sie einen Konflikt erhalten, wenn Sie Ihre Arbeitskopie aktualisieren oder zu einer anderen URL wechseln. Es gibt zwei Arten von Konflikten:

- Dateikonflikte: Ein Dateikonflikt entsteht, wenn zwei (oder mehr) Entwickler dieselben Zeilen einer Datei geändert haben.
- Baumkonflikte: Ein Baumkonflikt entsteht, wenn ein Entwickler eine Datei oder einen Ordner umbenannt, verschoben oder gelöscht hat, den ein anderer Entwickler ebenfalls umbenannt, verschoben, gelöscht oder bearbeitet hat.

Dateikonflikte

Ein Konflikt tritt dann auf, wenn mehrere Personen die gleichen Zeilen in einer Datei verändert haben. Da Subversion nichts über Ihr Projekt weiß, überlässt es in solchen Fällen Ihnen, den Konflikt aufzulösen. Die sich in Konflikt befindlichen Bereiche sind in der Resolve-Ansicht folgendermaßen markiert:

```
<<<<<<< Dateiname
    Ihre Änderungen
=======
    Code aus dem Projektarchiv
>>>>>>> Revision
```

Außerdem werden für jede Datei in Konflikt drei weitere Dateien in Ihrem Verzeichnis erstellt:

filename.ext.mine

Dies ist die Datei, so wie Sie war, bevor Sie Ihre Arbeitskopie aktualisierten. Das heißt, es ist Ihre eigene Originaldatei inklusive der Änderungen, die Sie selbst vorgenommen haben.

filename.ext.rOLDREV

Dies ist die Datei, wie Sie ursprünglich war, ohne jegliche Änderungen, auch ohne die Änderungen, die Sie selbst an der Datei vorgenommen haben.

filename.ext.rNEWREV

Dies ist die Datei, wie sie im Projektarchiv gerade aktuell ist, d. h., diese Datei hat die Änderungen von den anderen Mitarbeitern bereits integriert, jedoch noch nicht die Ihren.

Sie können entweder einen externen Konflikteditor per `TortoiseSVN → Konflikt bearbeiten` aufrufen oder Sie können den Konflikt mit einem beliebigen Texteditor manuell auflösen. Sie müssen entscheiden, wie der Code aussehen soll, die notwendigen Änderungen vornehmen und die Datei speichern. Mit einem Konflikteditor wie TortoiseMerge oder einem der anderen beliebten Programme ist das im allgemeinen einfacher, da diese die betroffenen Dateien normalerweise in einer Drei-Fenster-Sicht anzeigen und Sie sich keine Gedanken über die Konfliktmarken machen müssen. Wenn Sie einen Texteditor verwenden, müssen sie manuell nach Zeilen suchen, die mit «««< beginnen.

Anschließend müssen Sie Subversion noch mitteilen, dass Sie den Konflikt aufgelöst haben. Dies geschieht mit dem Befehl `TortoiseSVN → Konflikt aufgelöst`. Bitte beachten Sie, dass dieser Befehl nicht den Konflikt selbst löst, sondern nur Subversion mitteilt, dass Sie selbst den Konflikt bereits gelöst haben. Der Befehl macht nichts weiter als die drei zusätzlich erstellten Dateien `filename.ext.mine` und `filename.ext.r*` zu löschen, damit sie Ihre Änderungen in das Projektarchiv übertragen können.

Wenn ein Konflikt zwischen Binärdaten besteht, versucht Subversion nicht, die Daten selbst zusammenzuführen. Die lokale Datei bleibt unverändert (exakt so, wie sie Ihrer letzten Änderung entspricht) und Sie erhalten `Dateiname.ext.r*` Dateien. Wenn Sie Ihre eigenen Änderungen verwerfen wollen, tun Sie das mit dem Befehl `Rückgängig`. Wenn Sie Ihre Version beibehalten und die Version im Projektarchiv überschreiben wollen, verwenden Sie den Befehl `Konflikt aufgelöst` und übertragen anschließend die Daten ins Projektarchiv.

Sie können den Befehl `Konflikt aufgelöst` für mehrere Dateien verwenden, indem Sie den übergeordneten Ordner markieren und `TortoiseSVN → Konflikt aufgelöst…` aus dem Kontextmenü wählen. Dies öffnet einen Auswahldialog, in dem alle konfliktbehafteten Dateien aufgelistet sind. Wählen Sie die Dateien, die Sie als aufgelöst markieren wollen.

Eigenschaftskonflikte

Ein Eigenschaftskonflikt entsteht dann, wenn zwei oder mehr Entwickler dieselbe SVN-Eigenschaft verändert haben. Wie bei Dateikonflikten können nur Entwickler solche Konflikte auflösen.

Wenn eine der Änderungen die andere überschreiben soll, wählen Sie entweder `Mit der lokalen Eigenschaft auflösen` oder `Mit der Eigenschaft aus dem Projektarchiv auflösen`. Wenn die Änderungen zusammengeführt werden müssen, wählen Sie `Revisionseigenschaft bearbeiten`, tragen dort den gewünschten Wert ein und markieren den Konflikt als aufgelöst.

Baumkonflikte

Ein Baumkonflikt entsteht, wenn ein Entwickler eine Datei oder einen Ordner umbenannt, verschoben oder gelöscht hat, den ein anderer Entwickler ebenfalls umbenannt, verschoben, gelöscht oder bearbeitet hat. Es gibt verschiedene Ursachen für Baumkonflikte und alle erfordern unterschiedliche Vorgehensweisen, um den Konflikt aufzulösen.

Wenn eine Datei in Subversion lokal gelöscht wird, wird sie auch aus dem lokalen Dateisystem gelöscht. Das bedeutet, dass kein überlagertes Symbol angezeigt werden kann, wenn sie Teil eines Baumkonfliktes ist und dass Sie den Konflikt nicht mit Hilfe eines Rechtsklicks auflösen können. Verwenden Sie stattdessen den `Auf Änderungen prüfen` Dialog, um den `Konflikt bearbeiten` zu können.

TortoiseSVN kann dabei helfen, Änderungen zusammenzuführen, aber es kann zusätzliche Arbeit erforderlich sein, um die Konflikte aufzulösen. Bedenken Sie, dass nach eine Aktualisierung die BASE der Arbeitskopie dem Inhalt des Projektarchivs entspricht. Wenn Sie eine Änderung nach dem Aktualisieren rückgängig machen, wird das Objekt in den Status des Projektarchivs zurückversetzt und nicht in den Zustand, in dem Sie begonnen haben, Ihre eigenen Änderungen durchzuführen.
</showif>

Baumkonflikt: Lokal gelöscht, eingehende Änderung beim Aktualisieren

Baumkonflikt: Lokal geändert, eingehendes Löschen beim Aktualisieren

Baumkonflikt: Lokal gelöscht, eingehende Löschung beim Aktualisieren

Baumkonflikt: Lokal fehlend, eingehende Änderung beim Aktualisieren

Baumkonflikt: Lokal bearbeitet, eingehende Löschung beim Zusammenführen

Baumkonflikt: Lokal gelöscht, eingehende Löschung beim Zusammenführen

Andere Baumkonflikte

lokale Merge_Unterverzeichnisse