

Änderungsantrag zu XPersonenstandsregister

Zeichen 14/2015
Status erledigt
Umsetzung Version 1.8
Priorität



Umstellung auf Generierung WSDL-Inhalte aus dem UML Modell

Problemstellung

Die Inhalte von Kapitel 8 (Webservices) und die WSDL-Datei (xpersonenstandsregister-wsregister_1_8.wsdl) wurden bislang manuell bzw. teil-automatisiert erstellt.

Die KoSIT hat nun ihre Produktionsumgebung dahingehend erweitert, dass beides konsistent aus dem UML-Fachmodell generiert werden kann. Inkonsistenzen zwischen den Inhalten aus Kapitel 5 (Nachrichten), Kapitel 8 (Webservices) und der WSDL-Datei für XPSR können damit zukünftig größtenteils vermieden werden.

Im Zuge der Umstellung wurden bestehende Inkonsistenzen identifiziert, die bereinigt werden sollten.

Lösung

Das Expertengremium hat die festgestellten Inkonsistenzen erörtert und aufgelöst. Sie empfiehlt, den Webservice wie vorgeschlagen anzupassen.

Um auch die verbleibenden Möglichkeiten für Inkonsistenzen zwischen den Inhalten der Nachrichten aus Kapitel 5 und den Webservices aus Kapitel 8 auszuräumen, befürwortet das Expertengremium die Methode in Zukunft weiterzuentwickeln.

Antragsdetails

Antragsteller: KoSIT (Hannes Weber)

Efasst am: 04.11.2015

Bezug: Spezifikation 1.8 – Kapitel 8 und xpersonenstandsregister-wsregister_1_8.wsdl

Analyse des Änderungsantrags

Die Inhalte von Kapitel 8 (Webservices) und die WSDL-Datei (xpersonenstandsregister-wsregister_1_8.wsdl) wurden bislang manuell bzw. teil-automatisiert erstellt.

Die KoSIT hat nun ihre Produktionsumgebung dahingehend erweitert, dass beides konsistent aus dem UML-Fachmodell generiert werden kann. Inkonsistenzen zwischen den Inhalten aus Kapitel 5 (Nachrichten), Kapitel 8 (Webservices) und der WSDL-Datei für XPSR können damit zukünftig größtenteils vermieden werden.

- a) Im Zuge der Umstellung hat sich allerdings herausgestellt, dass es derzeit Inkonsistenzen gibt. Neben einigen „redaktionellen“ Inkonsistenzen sehen die Beschreibungen der Webservices in Kapitel z. T. andere Exceptions vor, als dies die technisch maßgebliche WSDL-Datei tut.
 - 1) DelUser: SemanticException in WSDL-Datei aber nicht in Kapitel 8
 - 2) GetEintragStatus:
 - a. NacherfassungNotYetCompletedException in WSDL aber nicht in Kapitel 8
 - b. NotFoundException in Kapitel 8 aber nicht in WSDL
 - 3) GetRegisterEintragInNacherfassung: LockedException in WSDL-Datei aber nicht in Kapitel 8
 - 4) GetRegisterEintragStillgelegt: NacherfassungNotYetCompletedException in Kapitel 8 aber nicht in WSDL
 - 5) InsertEintrag: NotLockedException in WSDL-Datei aber nicht in Kapitel 8
 - 6) InsertEintragNacherfassung: NotLockedException in WSDL-Datei aber nicht in Kapitel 8
- b) Außerdem sieht die neue Methodik im Interesse der konsistenten Dokumentation vor, dass die Beschreibung der Parameter eines Webservices der Beschreibung der Kindelemente der entsprechenden „Request“-Nachricht entspricht. D. h. dass bspw. die Dokumentation des Parameters registerJahrId aus Kapitel 8.2.1.2 entspricht der Dokumentation des Kindelements registerJahrId 5.3.1.1.

Lösungsvorschlag im Änderungsantrag

Zu a) Die Inkonsistenzen zwischen den Beschreibungen aus Kapitel 8 und der WSDL-Datei müssen analysiert und behoben werden.

Zu b) Die vor der Umstellung bestehenden Dokumentationen der Parameter müssen analysiert und bei Bedarf in die Dokumentation der „Request“-Nachrichten übernommen werden.

Bearbeitung

Wartend auf:

Aktivität	Status	Wer

Bearbeitet durch: KoSIT (HW)

Bearbeitet am: 2015-11-05

Zu a) Inkonsistenzen zw. Kapitel 8 und WSDL-Datei

Im Folgenden werden die in der Analyse festgestellten Inkonsistenzen bewertet und jeweils ein Vorschlag gemacht, ob die Exception erforderlich ist:

- 1) DelUser: Die Gründe für eine SemanticException sind in Kapitel 8.3.4 nicht weiter ausgeführt. → SemanticException nicht erforderlich
- 2) GetEintragStatus:
 - a. Die Gründe für eine NacherfassungNotYetCompletedException sind in Kapitel 8.2.27 nicht ausgeführt und die Information, dass der Eintrag sich in der Nacherfassung befindet ist in der Rückgabe enthalten → NacherfassungNotYetCompletedException nicht erforderlich
 - b. Die NotFoundException scheint erforderlich → NotFoundException erforderlich
- 3) GetRegisterEintragInNacherfassung: Die Gründe für eine LockedException sind in Kapitel 8.2.18 nicht ausgeführt und andere „lesende“ Zugriffe werfen auch keine LockedException → LockedException nicht erforderlich
- 4) GetRegisterEintragStillgelegt: Die Gründe für NacherfassungNotYetCompletedException sind in Kapitel 8.2.31 nicht ausgeführt und da stillgelegte Einträge eh nicht für die „normale Bearbeitung geöffnet werden können → NacherfassungNotYetCompletedException nicht erforderlich
- 5) InsertEintrag: Gründe für NotLockedException in Kapitel 8.2.2 beschrieben → NotLockedException erforderlich
- 6) InsertEintragNacherfassung: Gründe für NotLockedException in Kapitel 8.2.16 beschrieben → NotLockedException erforderlich

Die Vorschläge wurden in der aktuellen Entwurfsfassung von XPSR mit Stand 2015-11-05 bereits umgesetzt.

Zu b) Analyse und Übernahme der Parameter-Dokumentation

Im Folgenden werden die Parameter der Webservices benannt, deren Dokumentation über die Dokumentation der entsprechenden Kindelemente in den „Request“-Nachrichten hinausgehen. An den genannten Stellen wurde die Dokumentation der Kindelemente der „Request“-Nachrichten entsprechend fortgeschrieben.

- 1) Doku aus UndoJahresAbschluss/Aufhebung (Kapitel 8.2.22) wurde in UndoJahresAbschluss/ JahresAbschlussDokument/JahresAbschluss (Kapitel 6.7.5) integriert
- 2) Doku aus GetEvidenceDokument/ArchivId (Kapitel 8.2.23) wurde in GetEvidenceDokument/ArchivId (Kapitel 5.3.45) integriert
- 3) Doku aus CreateUser/ Berechtigungsnachweis (Kapitel 8.3.1) wurde in CreateUser/ berechtigungsnachweis (Kapitel 5.4.1) integriert.
- 4) Doku aus SetBerechtigungsnachweis/ Berechtigungsnachweis (Kapitel 8.3.3) wurde in SetBerechtigungsnachweis/ berechtigungsnachweis (Kapitel 5.4.5) integriert

Bearbeitet durch: EG XPSR

Bearbeitet am: 2015-11-11

Die oben dargestellten Änderungen im Vorgehen findet Zustimmung im EG. Eine zukünftige Weiterentwicklung der Methode (in einem separaten CR) wird befürwortet, damit auch das

noch verbleibende Potential für Inkonsistenzen zwischen den Nachrichtenbeschreibungen in Abschnitt 5 und den Webservicebeschreibungen in Abschnitt 8 eliminiert werden kann. Dabei sollte die in XTA verwendete Methode zur Dokumentation von Webservices berücksichtigt werden.

Das EG beschließt, die in der Bearbeitung vom 5.11.15 unter a) dargestellten Inkonsistenzen wie folgt aufzulösen:

- Zu 1) DelUser: Die derzeit nur in der WSDL-Datei genannte SemanticException soll erhalten bleiben, d. h. zukünftig auch in Abschnitt 8 genannt werden. In Abschnitt 8 wird keine gesonderte Dokumentation der Gründe aufgenommen, die zu einer SemanticException führen. Die grundsätzliche Dokumentation der Exception selbst ist ausreichend.
Begründung: Im Zweifel keine Schnittstellenänderung – es ist unklar, ob die Exception in der Praxis benutzt wird oder nicht.
- Zu 2) GetEintragStatus:
 - o Die derzeit nur in der WSDL-Datei genannte NacherfassungNotYetCompletedException kann entfernt werden, falls eine Prüfung der Hersteller keine Verwendung der Exception ergibt.
 - o Die derzeit nur in Abschnitt 8 genannte NotFoundException ist nicht erforderlich, da in diesem Fall „null“ zurückgeliefert wird. Die Methode kann auch benutzt werden, um zu prüfen, ob es einen bestimmten Eintrag gibt. Das Nicht-Finden ist in diesem Fall kein Fehler.
 - o Prüfauftrag für die KoSIT: Gibt es eine allgemeine Regel, wann „null“ und wann „NotFoundException“ verwendet wird (These: alle get-Methoden geben „null“ zurück)? Falls ja, sollte diese Regel in Abschnitt 8 grundsätzlich erläutert werden.
- Zu 3) GetRegisterEintragInNacherfassung: Die derzeit nur in der WSDL-Datei genannte LockedException kann entfernt werden, falls eine Prüfung der Hersteller keine Verwendung der Exception ergibt.
- Zu 4) GetRegisterEintragStillgelegt:
 - o Die derzeit nur in Abschnitt 8 genannte NacherfassungNotYetCompletedException kann entfernt werden. Unabhängig davon, ob es für stillgelegte Einträge noch eine Markierung „in Nacherfassung“ geben kann oder nicht, müssen diese Einträge von GetRegisterEintragStillgelegt zurückgeliefert werden.
 - o Offene Frage: Kann es stillgelegte Einträge mit der Markierung „in Nacherfassung“ geben?
VfSt: nein, XSPR wird so interpretiert, dass mit Stilllegung immer auch die Nacherfassung abgeschlossen wird.
Accenture sollte dies ebenfalls prüfen.
 - o Falls sich die Registerverfahren an dieser Stelle gleich verhalten, kann die Aussage in der Spezifikation unter den Abschnitten 4.3.8 und 8.2.30.4 geschärft werden.
- Zu 5) InsertEintrag: Die derzeit nur in der WSDL-Datei genannte NotLockedException ist auch in Abschnitt 8 aufzunehmen. Die Gründe sind in Abschnitt 8.2.2 beschrieben.
- Zu 6) InsertEintragNacherfassung:

- Die derzeit nur in der WSDL-Datei genannte `NotLockedException` ist auch in Abschnitt 8 aufzunehmen. Die Gründe sind in Abschnitt 8.2.16 beschrieben:
- Redaktionell: „§14 PStV“ → „§ 14 PStV“

Darüber hinaus soll in Abschnitt 8 unter „Überprüfungen bei allen Webservices“ aufgenommen werden, dass `checkVerfuegung` als Ausnahme der Regel keine `SemanticException` werfen kann, obwohl sie ebenfalls unter Registerinhalte fällt.

Begründung: Die Fehlermeldung für den Benutzer ist bereits im Rückgabewert enthalten und wird nicht über eine Exception transportiert.

Das EG beschließt, die in der Bearbeitung vom 5.11.15 unter b) dargestellten Abweichungen zwischen den Elementbeschreibungen (Abschnitt 5) und den Beschreibungen der Parameter (Abschnitt 8) wie folgt aufzulösen:

- Zu 1) UndoJahresabschluss:
 - o Die Integration der Dokumentation des Parameters `Aufhebung` (Kapitel 8.2.22) in `UndoJahresAbschluss/ JahresAbschlussDokument/JahresAbschluss` (Kapitel 6.7.5) ist rückgängig zu machen.
 - o Abschnitt 5.3.43 `UndoJahresAbschluss`, Dokumentation des Kindelements `Jahresabschlussdokument`: „Enthält das ‚Jahresabschlussdokument‘, das die Aufhebung des Jahresabschlusses dokumentiert.“
 - o Abschnitt 5.3.42 `MakeJahresAbschluss`, Dokumentation des Kindelements `Jahresabschlussdokument`: „Dieses Element enthält alle Angaben, die für einen Jahresabschluss benötigt werden.“
 - o Abschnitt 6.7.5.1 `JahresAbschluss (TJahresAbschluss)`: Die Dokumentation ist zu streichen.
- Zu 2), 3) und 4):
 - o Die vorgenommen Angleichungen werden bestätigt.
 - o Redaktionelle Änderung an beiden Stellen: Berechtigungsnachweis.

Die Registerverfahrenshersteller prüfen alle oben genannten Punkte bis zum 25.11.

Bearbeitet durch: Hr. Altmann (Accenture) **Bearbeitet am: 2015-11-22**

Herr Altmann bestätigt, dass sich das Verfahren von Accenture zu der unter „Zu 4“ (11.11.15) aufgeworfenen Frage genauso verhält, wie das Verfahren vom VfSt.

Anmerkung KoSIT: Die Aussagen in den Abschnitten 4.3.8 und 8.2.30.4 können also entsprechend geschärft werden.

Bearbeitet durch: KoSIT (Weber) **Bearbeitet am: 2015-11-25**

Die EG-Beschlüsse vom 11.11.15 wurden wie folgt umgesetzt:

- Zu a1) Für die Methode `DelUser` wurde die `SemanticException` wieder aufgenommen
- Zu a2)
 - o In der Methode `GetEintagStatus` wurde die Exception `NotFoundException` entfernt
 - o Die Prüfung wurde mit dem Ergebnis durchgeführt, dass derzeit einer allgemeinen Regel einige Ausnahmen im Wege stehen. Zu dem Thema wurde der neue CR 15/2015 erstellt.
 - o Im Zuge der Prüfungen wurde festgestellt, dass `CheckVerfuegung` fälschlicherweise den Rückgabewert `null` definiert, obwohl das return-Kindelement mandatorisch ist. Die Tatsache, dass ein RN nicht existiert, wird derzeit über den Schlüssel 5 im Rückgabewert ausgedrückt. Die Regelung wurde aus der Spezifikation entfernt (vgl. Kapitel 8.2.29.4)

- Zu a3) Keine weitere Änderung an *GetRegisterEintragInNacherfassung* erforderlich
- Zu a4) Keine weitere Änderung an *GetRegisterEintragStillgelegt* erforderlich. Die Formulierungen in Abschnitt 4.3.8 wurde geschärft. Die Formulierung in Abschnitt 8.2.30.4 ist ausreichend präzise.
- Zu a5) Keine weitere Änderung an *InsertEintrag* erforderlich.
- Zu a6) Keine weitere Änderung an *InsertEintragNacherfassung* erforderlich. Die redaktionelle Änderung wurde übernommen.
- In Abschnitt 8 unter „Überprüfungen bei allen Webservice“ wurde klargestellt, dass bei *CheckVerfuegung* keine *SemanticException* geworfen werden kann.
- Zu b1)
 - o Die Änderungen in Kapitel 6.7.5 wurden rückgängig gemacht, d. h. die Dokumentation unter 6.7.5.1 wurde gelöscht.
 - o Die Abschnitte 5.3.42 und 5.3.43 wurden angepasst.
- Zu b3 und b4) Die redaktionelle Korrektur wurde übernommen.

Der CR ist damit erledigt.