

(English version follows after the German text; when answering or quoting choose the language you prefer)

Hallo railMLer,

vor ziemlich genau drei Monaten (am 21.06.05) habe ich ein paar Vorschläge für einen verbesserten Entwicklungsprozess in der Infrastruktur-Gruppe eingebracht. Das Ziel war (und ist), die Entwicklung zielgerichteter und transparenter zu machen.

Wir haben über diese Vorschläge während des letzten railML-Treffens gesprochen und uns entschlossen, dass wir sie bei der Entwicklung des neuen Interlocking-Schemas (siehe dazu auch den entsprechenden Beitrag in der Infrastruktur-Gruppe) ausprobieren.

Die nächsten Abschnitte beschreiben die wichtigsten Ideen des neuen Entwicklungsansatzes und sind im wesentlichen aus meiner oben genannten Mail entnommen. Wer also mit der Mail schon vertraut ist, kann die folgenden Zeilen überspringen.

Vor einer Neuentwicklung oder Erweiterung sollten wir zunächst die neuen Features und Elemente, die wir künftig unterstützen wollen, zusammentragen. Im Rahmen der damit verbundenen Diskussion sollten wir noch nicht über Implementierungsdetails diskutieren.

Nachdem wir uns auf eine Reihe von Features geeinigt haben (und auf die Version, in der sie erscheinen sollen) wird die eigentlich Umsetzung in Angriff genommen. Kurz gesagt:

- 1) Vorschläge hinsichtlich Funktionalität, Daten und Elementen sammeln
- 2) Die Vorschläge Versionen zuordnen ("Roadmap", "Feature Freeze")
- 3) Umsetzung beginnen; gleichzeitig: neue Vorschläge sammeln ==> (1)

Dieses Vorgehen wird auch in mehreren erfolgreichen Open-Source-Projekten verwendet und ich denke, dass es nützlich für uns sein könnte. Die vier wichtigsten Vorteile sind:

- * Auch Personen, die nicht Experte in Sachen XML / XSD sind, können Wünsche und Vorschläge einbringen. Momentan lassen sich viele Leser der Newsgroup von der Angst abschrecken, bei Vorschlägen und Diskussionsbeiträgen sofort nach konkreten XML-Beispielen befragt zu werden. Diese Angst ist natürlich unbegründet. Wir brauchen sowohl

die Beiträge von XML- als auch von Eisenbahnexperten!

- * Der Entwicklungsvorgang wird zuverlässiger, vorhersagbarer und transparenter. Anwender unseres Schemas können abschätzen, was die künftigen Schemas bringen und ihre eigene Entwicklung entsprechend anpassen.
- * Die Entwicklung selbst ist zielgerichteter; bis jetzt haben wir immer das implementiert, was uns gerade notwendig erschien. Wenn wir aber anhand einer Liste von Features vorgehen, können wir den Entwicklungsaufwand leichter auf mehrere Personen verteilen und wir können auch genau sagen, wenn die Entwicklung einer neuen Version abgeschlossen ist. Darüber hinaus werden auch andere Dinge wie Testen und Dokumentieren unserer Entwicklung erleichtert.
- * Durch die frühzeitige Festlegung der zu implementierenden Funktionen wird die Qualität der Entwicklung, insbesondere der Schemastruktur und der Datentypen, erhöht.

Ich hoffe sehr, dass wir durch dieses Vorgehen neue Entwickler und mehr Beiträge in den Newsgroups gewinnen können. Schauen wir einfach mal, wie sich Vorgehen bewährt, wenn es auf die Entwicklung des Interlocking-Schemas angewandt wird!

Gruß, Volker Knollmann

----- ENGLISH VERSION -----

Hello railML-fans and developers,

almost exactly three month ago (on 2005-06-21) I posted a set of ideas in the infrastructure-newsgroup, how to improve the development process of new or existing schemas in order make the development more transparent.

We discussed these ideas during the last railML-meeting and decided that they are worth a try. The application will be development of a new interlocking-schema, which will be announced in the infrastructure-group.

The following paragraphs describe the intended process. It mostly cite from my old mail, so if you know already that old postings, you can skip the following lines:

We start with a discussion about new features or elements for our

schema; in this discussion we should spend hardly no time on the implementation. Once we agreed on a set of features (and maybe on the versions they will be implemented in) we can start with the implementation discussion. In short:

- 1) Collect features
- 2) Assign features to versions ("roadmap", "feature freeze")
- 3) Implement; parallel: start again with (1)

This strategy can be found in several successful open-source projects and I think it can be useful for our needs. It has four main advantages:

- * People, especially from the industry, can bring in new ideas regardless of their implementation. Now, many readers of the newsgroup don't dare posting new ideas. They are afraid to be asked for concrete XML- or XSD-examples although they are not familiar with XML.
- * Our development becomes more reliable, predictable and transparent. Users of our schema can estimate which features will be introduced and take it for their development into account.
- * The development itself has a clear target; up to now, we only implement what comes to our minds. If we have a list of features to be implemented, we can distribute the development easily over a group of developers and we know precisely, when the development is finished. Other things like testing or documentation will be improved, too.
- * Due to the early freezing of features and functions, the quality of the resulting schema, especially regarding its structure and data types, will be improved.

I hope, with this procedure we can gain new developers and more transparency in the development. Let's see how it works, when it is applied to the upcoming interlockin-schema!

Best regards,
Volker Knollmann

--

German Aerospace Center (DLR)
Institute of Transportation Systems
Lilienthalplatz 7

38108 Braunschweig, Germany

Dipl.-Ing. Volker Knollmann

Telephone: +49 531 295-3461

Telefax : +49 531 295-3402

Internet: <http://www.dlr.de/fs>

Please use encryption and electronic signatures for secure data exchange. You can download my public key here:

<http://pgpkeys.pca.dfn.de/pks/lookup?op=get&search=0xEE4EB9525CDB6396>
