Hello,

thanks to Volker for this detailed information and outlook!

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

This strategy seems reasonable to me. At least it's worth trying it out :-)

Only remark: I think, this strategy is not fully "compatible" with the

(~"Development process for railML partial Schemes"),
 http://www.railml.org/documents/general/developmentprocess_g er.pdf (only
available in German?)
But facing the fact that this development process was never fully applied in
past, it might not be a big problem, or what do you think?

> To help railML spreading, we should update the documentation. We should
> provide the following documents
>
>   * An Overview about railML-infrastructure (exists, needs update)

Do you mean the schema documentation which was generated with XMLSpy?
(http://www.railml.org/genesis/infrastructure/v100/Reference/ HTML/Infrastruc
ture_V100_Reference_EN.html)
This is good for reference, but I think we also need some more detailed
specifications, which cannot be covered by the schema.
Or is there anything like
 http://www.railml.org/schemes/timetable/timetable_1.00_en.pd f available for
infrastructure?

>
>   [...]
>
> V) Perspective
>
> Finally, I want to give you an example how the development of the next
> versions can look like if we use the above described development process.
>

>   Roadmap:
>   -------------
>   Version 1.01: Bugfixes, updated documentation
>   Version 1.02: Introduction of complex types, harmonization
>   Version 1.1x: First introduction of new features or elements
>   (development versions)
>   Version 1.20: First introduction of new features or elements
>   (stable version)

Have you any ideas of time horizonts? E.g. What should be done until september (next meeting)?

>   Tasks:
>   ------------
>   * Report and collect bugs
>
>   * Collect ideas and contents for the documentation

In my opinion, the documentation of the timetable schema
(http://www.railml.org/schemes/timetable/timetable_1.00_en.pd f) is a good example.
I propose that we define the format/structure of the documenation first, and then it can be filled with contents from (I hope) several developers.

>
> [...]
>

Best regards
Matthias Hengartner

--
****
Matthias Hengartner

hengartner@ivt.baug.ethz.ch
++ 41 44 633 68 16
****