Transforms By Example: Issues and Workarounds

The main aim of the TBX Beta test phase is to discover any issues or difficulties users have with the TBX tools, and put them right.

Inevitably there is a time lag between discovering some issue (or a user encountering it) and our putting it right. In the intervening time, other Beta test users need to be aware of the issue, and of any way (if there is one) of working around it.

This note contains a running list of the issues that have arisen, any workarounds we can suggest, and the release of the TBX tools in which the issue has been fixed. It will be regularly updated.

Please let us know of any other issues you encounter.

If you need some fix urgently, tell us and we will give it high priority.

|  |  |  |  |
| --- | --- | --- | --- |
| **Issue** | **Description** | **Workaround** | **Fixed in Release** |
| Mac Installations | On selecting a file, you get the message 'Selected file must be in the Eclipse Workspace', even when it is. | This may happen when choosing a file for output of generated Java or FML. Do it by editing the gen file directly, then reload the transform.  Tell us about any other problems on Mac. |  |
| Transforms between FHIR versions | The tools currently require the FHIR version to be the same for source and target - so we cannot yet build transforms between FHIR versions. | None yet - but the fix is simple. |  |
| CDA transforms | To do CDA to FHIR transforms easily (without too many example pairs) the tools need special facilities , to recognise CDA templates | Wait until we have implemented and tested these facilities, before tackling CDA. |  |
| Relational database as target | The tools do not fully support this use case yet. | Create the transform with RDB as source, and something else (X) as target. This creates mappings which allow you to run the transform in both directions, using the native transform engine in the tools. For X=>RDB, you cannot yet generate Java or FML. |  |
| Convenience of RDB as source | Currently you need to define any relational database source structure in two csv files, which you need to create. | Create the files, by hand if necessary.  Facilities to get RDB metadata by jdbc exist, and we will couple them to TBX |  |
| Using HAPI FHIR server facilities | If you want to build a FHIR server facade on a legacy relational database, TBX cannot yet auto-generate SQL to retrieve the required records, from a RESTful FHIR search on a HAPI server | Write the code to generate SQL from RESTful searches yourself.  In a future release, TBX will be able to generate SQL from HAPI server information. |  |
| FHIR ids and resource references | With FHIR as target, you probably need to create FHIR ids for resources obeying some local rules. This is not yet very convenient to do | Modify the generated Java - preferably by subclassing, so you can re-generate. |  |
| Writing Java data conversion functions | You need to set up Eclipse to do this, so you can develop Java code and run it in TBX. In the Help files that describe how to do this, currently the screenshots are missing, so they may be bit cryptic. | Ask for help if necessary |  |
| Tips on building example pairs are not yet in the Eclipse help files | You really need to read these tips, and give us feedback about them. | The tips are in a Word document in the Dropbox you use to download the tools. |  |