

FHIR Mapping Language and FHIR Transform Engine

- FHIR Mapping Language (FML) and FHIR Transform Engine (FTE) both use declarative mappings
- Both do transforms by a runtime mapping engine, minimising code
- FML:
 - maps source DAG => target DAG
 - FHIR can be source or target
 - runs uni-directional transforms, source=> target
 - Mapping style is 'push' from source to target
 - Mappings are text files, or resource instances
- FTE:
 - maps source and target to a common UML class model
 - FHIR can be the class model
 - runs bi-directional transforms
 - Runtime engine is 'pull' to target from source
 - Mappings are viewed as visual trees or spreadsheets
- So FML and FTE appear very different
- FTE mappings are more fine-grained than FML rules
- But they are inter-convertible

FML and FTE: Possible Synergy

- It is now possible to convert one set of mappings to the other:
 - FML to FTE (easy; not coded yet)
 - FTE to FML (harder; demo exists)
- So you will be able to move mappings between the two forms
- Choice for viewing and editing mappings:
 - FML text files
 - StructureMap resource
 - FTE visual mapping editor
 - FTE spreadsheet form
- Choice of runtime engine:
 - FML mapping engine
 - FTE Java engine
 - FTE + HAPI
 - FTE generated XSLT