

Protokoll SD-Tech.-Report-Treffen am 28. Juni

1. Reviews

- User Guide, Installation von Interpreter müsste jemand lesen. Freiwillige?
- Inclusion Links: eingearbeitet
- Complete Example: Jan hat das Review mit Markus besprochen, ist in Arbeit.

2. TODOs

- Todos von Markus und Christian: wie im Protokoll vom 21.6.2012, falls noch nicht erledigt.
- 3.2.3 Link Variables: Typ einer LinkVariable sind ein bis zwei EReferenzen, konzeptionell eigentlich eine Assoziation. Außerdem muss erwähnt werden, dass zwei LinkVariables des gleichen Typs zwischen zwei Objekten auch bedeuten, dass das eine Objekt zwei Mal mit dem anderen verbunden ist. Dazu muss auch der Interpreter angepasst werden. Beides beschreibt Marie.
- 3.2.5 Using Object Attributes: Attribute Constraints müssten in Object Constraints umbenannt werden, weil die Constraints sich nicht zwingend auf ein Attribut beziehen, aber im Gegensatz zu Pattern Constraints den Scope eines Attributs haben. Wer hat den Abschnitt geschrieben und würde das tun?
- 3.2 Story Patterns: Das Matching von Story Patterns soll in einem eigenen Abschnitt am Ende von Abschnitt 3.2 beschrieben werden, z.Z. als 3.2.11. Hier soll vor allem erklärt werden, wie Story Patterns beschaffen sein müssen, damit sie gematcht werden können (Ausgang von gebundenen Variablen, Stichwort: Zusammenhangskomponente, Besonderheiten bei optionalen und negativen Anteilen, ggf. Isomorphie im Zusammenhang mit Collections und maybe-Constraints, Backtracking, etc.) und wie das Matching im Prinzip funktioniert.
- 3.2.10 Pattern Constraints (Christian): Hier muss das Beispiel in Absprache mit Markus (wegen Complete Example) und Jan geändert werden (siehe Todo im Dokument). Vorschlag von Stephan und Jan: mehrere Beispiele, die verschiedene Notationen für den gleichen Zweck verdeutlichen.
- A User Guide: Installation von Editor & Co.; Aljoscha wird dazu etwas schreiben, wenn die Tools so weit fertig sind, dass sie installierbar sind. Vermutlich erst zum Release von Version 1.0.
- Expressions-Abschnitt sollte erweitert werden: Beschreiben, dass wir ein Konstrukt für beliebige textuelle Ausdrücke vorbereitet haben, wodurch OCL und andere Sprachen eingebettet werden können, wenn man entsprechende Editoren und Interpreter registriert. OCL wird von Haus aus unterstützt. Außerdem haben wir eine sehr kleine Sprache für simple logische und arithmetische Ausdrücke (dafür brauchen wir noch einen Namen). Das wird in der kommenden Version vermutlich nicht mehr beschrieben, vielleicht in Version 1.0.

3. Nächste Treffen

- Donnerstag, 5.7. um 10 Uhr (Jan ist nicht da, aber Stephan hat doch noch Zeit)
- Mittwoch, den 11.7. (Tag vor dem Release) um 14 Uhr treffen wir uns für letzte Änderungen am Bericht für 2 bis 3 Std.

4. Weitere Planungen

- Aljoscha arbeitet an Editor weiter. Der Interpreter bedarf einiger Erweiterungen und Anpassungen, um den im Bericht beschriebenen Stand zu unterstützen. Stephan

kann erst nach ca. 2 Monaten dabei helfen. Aljoscha wird (wenn es mit EDapt & Co. klappt) die gesammelten Meta-Modell-Anpassungen durchführen und eine Update Site vorbereiten.

- Der Interpreter unterstützt z.Z. keine Activity Call Nodes, Collections, Inclusion Links, etc. Stephan hat aber den Interpreter so vorbereitet, dass man durch Überschreiben einiger Methoden bei der Behandlung von „Custom Nodes“ und Ähnlichem die Funktionalität ergänzen kann. Ob Aljoscha vor dem Release 1.0 dazu kommt, ist noch unklar. Matching von uni-direktionalen Assoziationen in umgekehrter Richtung funktioniert laut Stephan bereits.
- Die Expressions werden in MUML fast genauso wie bei uns genutzt. Sie verwenden aber mehr Konstrukte, um eine Action Language aufzubauen. Nach Absprache mit Uwe Pohlmann werden die Meta-Modelle für Expressions angeglichen. MUML wird dann unsere Expressions erweitern und es wird für beide Expression-Sprachen Parser/Generatoren auf XText-Basis geben.