Pending Evolution of Grammars
Why the problem is a problem

- Grammar evolution is (meta)model transformation
- MDE doesn’t work
- All MDE tools are crap
- Model transformation bores people
- We need more agility
- [citation needed] ← also XM keynote by Jon Whittle
Consider it done!

http://cheezburger.com/6125443584
Grammar evolution
Grammar evolution

G → G → G → G → G

unfold
Composable default fragments
Composable default fragments
Composable default fragments

\[ \text{defineN} \]
Composable default fragments

two defineN failed

defineN
Asserting preconditions
Asserting preconditions

vertical

deyaccify

deyaccify

vertical
Asserting preconditions

vertical

deyaccify

deyaccify

vertical
Asserting preconditions

vertical

deyaccify

G ➝ G ➝ G ➝ G ➝ G

deyaccify
deyaccify

=all Ok
Bulk processing
Bulk processing

- inject
- defineN
- replace
- deyaccify
Bulk processing

G → G → G → G → G

fix

fix

fix

fix
Bulk processing
Bulk processing
Bulk processing
Pending failure
Pending failure

renameN

introduce

narrow

unite
Pending failure

renameN

report

introduce

narrow

unite
Pending failure

negotiations help!

report

introduce

unite

narrow

cf. XM2012/JOT
Persistent normalisation
Persistent normalisation

G → G → G → G → G

addV

factor

define

introduce
Persistent normalisation

G
G
G
G
G

addV

.factor

define

introduce

:-(
Persistent normalisation

horizontal

define

addV

factor

introduce

:-)
Pre-export normalisation
Pre-export normalisation

G → G → G → G

change
change
change
change
Pre-export normalisation

G → G → G → G → G

Normalise

Change

Change

Change

Change
Also diff:

differ

normalise

change

change

change

change
To summarise

- Don’t do
- Consider it done
- Accumulate failures
- Plan beforehand
- Execute optionally
- ???
- PROFIT!

http://commons.wikimedia.org/wiki/File:Torii_kiyoshige_bando_hikosaburo_ii.jpg
Tutorial on Tuesday
(1 October 2013, 8:30)
Pending Evolution of Grammars

Vadim Zaytsev
Software Analysis & Transformation Team,
Centrum Wiskunde & Informatica,
Amsterdam, The Netherlands

Abstract. The classic approach to grammar manipulation is based on instant processing of grammar edits, which limits the kinds of grammar evolution scenarios that can be expressed with it. Treating transformation preconditions as guards poses limitations on concurrent changes of the same grammar, on reuse of evolution scripts, on expressing optionally executed steps, on batch processing and optimization of them, etc. We propose an alternative paradigm of evolution, where a transformation can be scheduled for later execution based on its precondition. This kind of extreme evolution can be useful for expressing scenarios that are impossible to fully automate within the classic or the negotiated transformation paradigms.

1 Introduction

A colloquial expression ‘consider it done’ means that the subject of the conversation is either indeed already done, or will be done in the very near future — in either case, the receiver of such a message can rest assured that the subject will take place if it has not already, and is expected to act as if it has indeed happened. The technique of pending evolution that we introduce in this paper, is similar to that expression, and the benefits of it are not unlike the subtle differences between considering something done and it having been done.

As it turns out, the pending evolution scheme allows us to efficiently model scenarios of grammar evolution, deployment and maintenance that are impossible to model within the traditional grammar transformation paradigm, which is briefly explained in §2. The method is introduced in §3. Since the most profits hide deep in the details, we spend the rest of the paper (§4) by motivating the use of pending evolution for grammars instead of classical evolution scripts, by concrete examples. §5 concludes the paper by summarizing its contributions and discussing related work.

2 GrammarLab

GrammarLab is a codename for a grammar manipulation project that is currently being migrated from the Software Language Processing Suite\(^1\) initiative