Институт программных систем имени А.К. Айламазяна РАН

~~Способ~~ Проблема повышения качества алгоритмов~~,~~ ~~базирующихся на метрике Левенштейна или родственных понятиях~~ выделения различий в текстовых файлах

С.В. Знаменский

E-mail: svz@latex.pereslavl.ru

Прикладная сторона проблемы

- Выделение различий для анализа
- Выделение различий для автоматической обработки

Математические модели

- Неадекватность LCS
- Неадекватность patiencediff
- Неадекватность Bdiff

Проблема:

Найти алгоритм, лаконично, адекватно и за приемлемое время выделяющий различия между двумя версиями текстового файла.

Выделение различий между двумя версиями используется вручную множеством программных продуктов для просмотра и редактирования

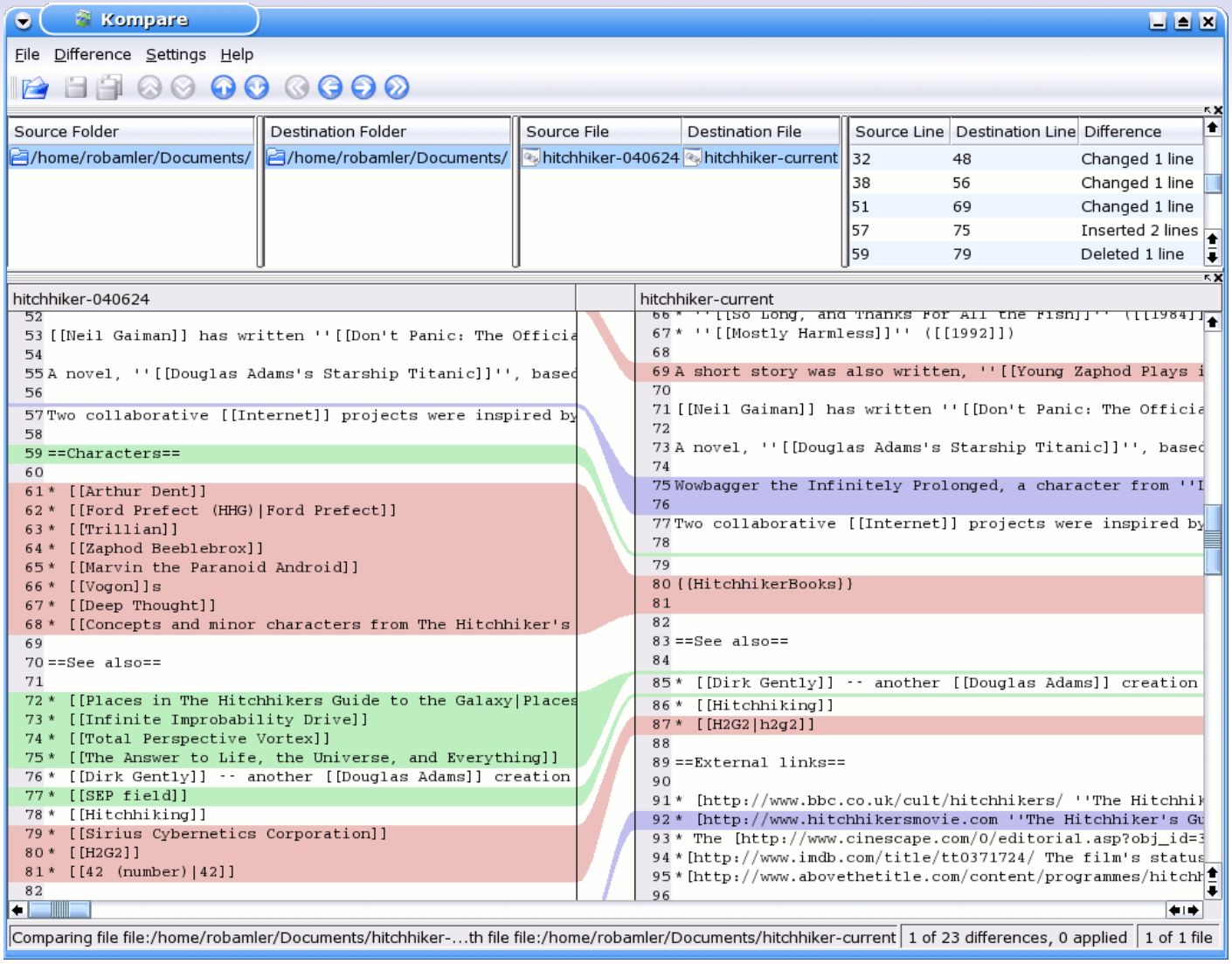
Beyond Compare	Scooter Software[1]	Proprietary	$US30 (std), 50 (pro)	1996	2014 (v4.0)
Compare++	Coode Software[2]	Proprietary	$29.95, 30-day free trial[3]	2010	2013 (v2.15)
diff, **diff3**	AT&T	BSD 3-clause, BSD 4-clause, CDDL, GPL, Proprietary	-	1974	
Ediff	Michael Kifer[5]	GPL	Free	1994	2.81.4
ExamDiff Pro	PrestoSoft[6]	Proprietary	$35, free file-only version[7]	1998	2014 (v7.0.1.6)
Far Manager (compare)	Eugene Roshal (original); FAR Group	Revised BSD license	Free	1996	August 2014 (v 3.0 build 4040)
fc	Microsoft[8]	Proprietary	Part of OS	1987	
FileMerge	Apple Inc.	Proprietary	Free (part of Apple Developer Tools)	1993 (part of NEXTSTEP 3.2[9])	2014 (v2.8)
Guiffy SureMerge	Guiffy Software[10]	Proprietary	$37.50(Pro), 75(eXpert)	2000	July, 2014 (v10.6)
IntelliJ IDEA (compare)	JetBrains[11]	Proprietary	Part of application		2007 (v6.0.5)
jEdit JDiff plugin	Various[12]	GPL	Free	1998	2012 (3.3.0)
Kompare	Otto Bruggeman[13]	GPLv2+	Free		
Lazarus Diff	Lazarus (software)	GPL	-	2000	2009
Meld	Stephen Kennedy[15]	GPLv2+	-	2002	2014-1-23 (v1.8.4)
Perforce P4Merge	Perforce	Proprietary[17]	-		2013 (2013.2)
Pretty Diff	Austin Cheney[18]	GPL-compatible	Free	2009	2012-07-05
Tkdiff	Tkdiff[19]	GPLv2+	Free	2003 (or before)	2011-11-27 (v4.2)
Total Commander (compare)	Christian Ghisler[20]	Proprietary	Part of application		2010-12-17 (v7.56a)
twdiff (TextWrangler Diff Helper)	Bare Bones Software, Inc.[21]	Proprietary	free (with TextWrangler [free])		2012 (1.0 (v22))
vimdiff	Bram Moolenaar et al.	GPL-compatible	-	2001	2006-05-07 (v7.0)
WinDiff	Microsoft[22]	Proprietary	Part of Platform SDK	1992	2003 ? (v6.1)
WinMerge	Dean Grimm[23]	GPL	Free	1998	2013-02-03 (v2.14.0)

для отслеживания изменений в исходных файлах

ЗАЧЕМ и КАКИЕ нужны суперкомпьютеры ~~эксафлопсного~~ эксафлопсного класса? Предсказательное моделирование свойств и многомасштабных процессов в материаловедении

Аннотация. Рассматривается подход, позволяющий выявить, для каких задач нужны суперкомпьютеры эксафлопсного класса. Возможности подхода рассмотрены на примерах актуальных задач материаловедения, физики конденсированного вещества и плотной плазмы, для решения которых необходимо атомистическое моделирование на современных и создаваемых в настоящее время суперкомпьютерах. Для каждой задачи проведено соответствие между набором изучаемых явлений и требуемым уровнем быстродействия (числа ядер) вычислительной системы. Показана масштабируемость параллельных программ моделирования и перспектива расширения предсказательной способности методов по мере увеличения числа вычислительных ядер и/или использования специализированных архитектур (графические ускорители). Рассмотрена иерархия методов моделирования, необходимых для адекватного описания свойств веществ на различных пространственных и временных масштабах. На наиболее глубоком нанометровом/пикометровом масштабе для моделирования электронной динамики и построения эффективных потенциалов взаимодействия частиц применяется теория функционала плотности (квантовая молекулярная динамика). Классический метод молекулярной динамики позволяет явно рассмотреть системы движущихся атомов вплоть до ~~микро-масштабов.~~ микромасштабов. Выход на ~~макро-масштабы~~ макромасштабы осуществляется с помо-

или в иных текстах при совместной подготовке



— как построчно,

Some widely used file comparison programs are [[diff]], [[cmp (Unix)|cmp]], [[Apple Developer Tools#FileMerge|FileMerge]], [[WinMerge]], [[Beyond Compare]], and [[Microsoft File Compare]].

Some widely used file comparison programs are [[di (Unix)|cmp]], [[Apple Developer Tools#FileMerge|Fil [[WinMerge]], [[Beyond Compare]], and [[Microsoft F

Line 8:

Line 8:

== Method types ==

== Method types ==

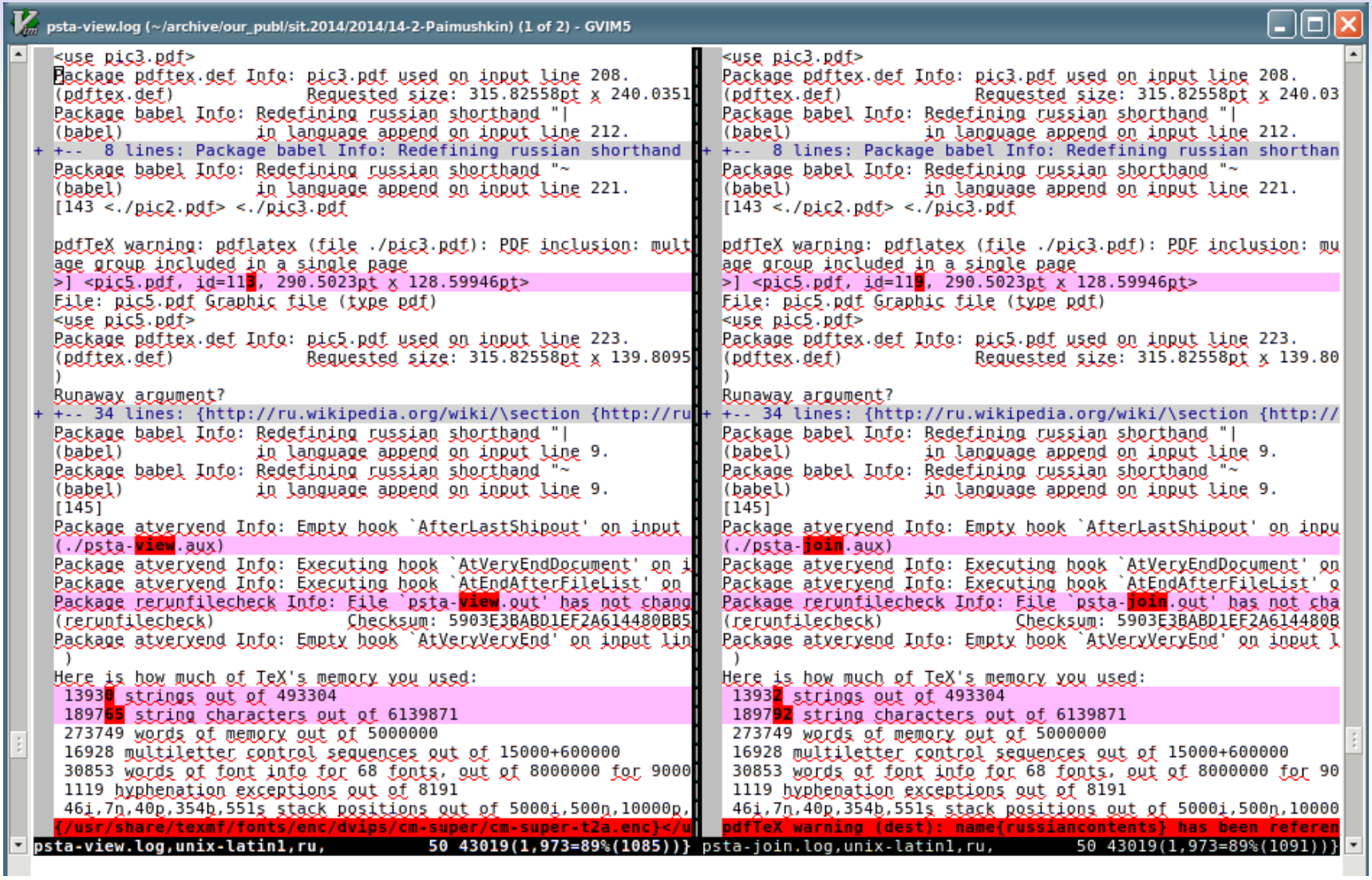
Most file comparison tools find the [[longest common subsequence]] between two files. Any data not in the longest common subsequence is presented as an insertion or deletion.

Most file comparison tools find the [[**longest commo problem**|longest common subsequence]] between tw not in the longest common subsequence is presented or deletion.

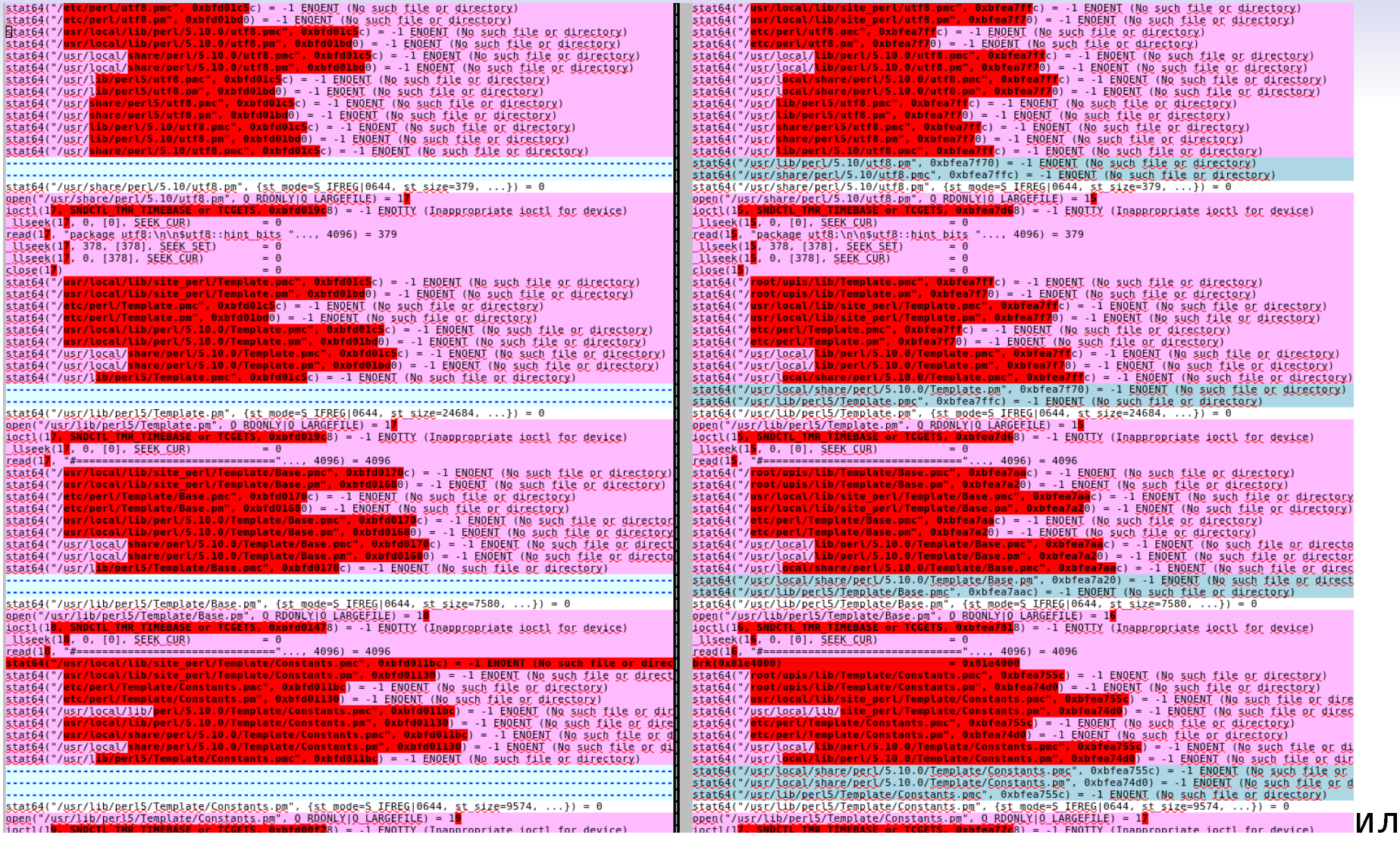
In 1978, Paul Heckel published an algorithm that identifies most moved blocks of text.<ref> {{Citation|last1=Viégas|first1=Paul|last1=Heckel|year=1978|title=A Technique for Isolating Differences Between Files|journal=Communications of the ACM|volume=21|pages=264–268|url=http://documents.scribd.com/docs/10ro9oowpo1h81pgh1as.pdf|accessdate=2011-12-04|doi=10.1145/359460.359467}}</ref> This is

In 1978, Paul Heckel published an algorithm that iden blocks of text.<ref> {{Citation|last1=Viégas|first1=Paul|last1=Heckel|yea Technique for Isolating Differences Between Files|journal=Communications of the ACM|volume=2 268|url=http://documents.scribd.com/docs/10ro9oow f|accessdate=2011-12-04|doi=10.1145/359460.3594

так и более детально (википедия)



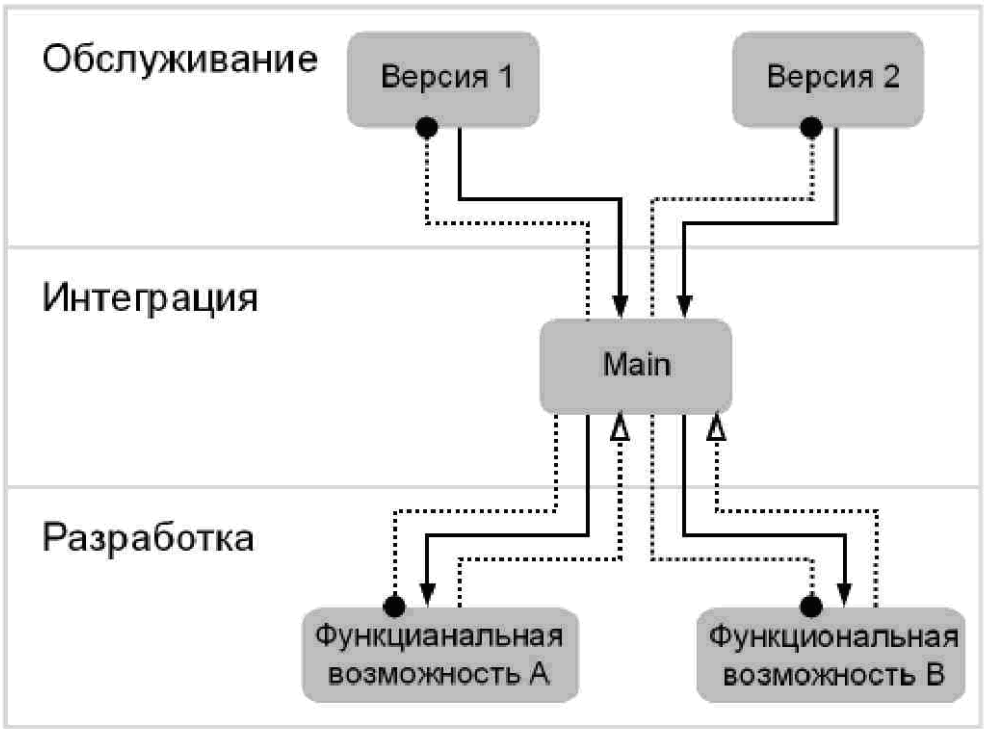
для отслеживания изменений в логах

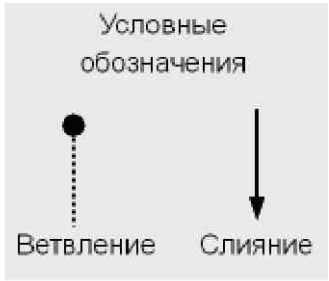


ил

в последовательности системных вызовов.

Используется системами сборки сложного программного обеспечения для выявления конфликтов и наложения изменений,





независимо внесённых разными разработчиками. Качество и сложность сборки сильно зависит от точности локализации изменений.

Основные математические модели:

Широко используются (по Baudiš P. Current concepts in version control systems //arXiv preprint arXiv:1405.3496. – 2014.(2009-09-11)):

1. Longest Common Subsequence LCS (Myers E.: An O(ND) Difference Algorithm and its Variations, Algorithmica 1 (1986) 251–266) — GNU diffutils, perl, ...
2. patiencediff (LCS for lines which occur exactly once on both sides, then recurse between). Cohen B., et al.: Bazaar Source Code, bzrlib/patiencediff.py, `http://www.bazaar-vcs.org/Download`
3. Bdiff longest common continuous substring — Ratcliff J. W., Metzener D. E. Pattern-matching-the gestalt approach //Dr Dobbs Journal. – 1988. – Т. 13. – №. 7. – С. 46- difflib(python), Mercurial, Revlog

не вполне адекватны

Основные математические модели:

Широко используются (по Baudiš P. Current concepts in version control systems //arXiv preprint arXiv:1405.3496. – 2014.(2009-09-11)):

1. Longest Common Subsequence LCS (Myers E.: An O(ND) Difference Algorithm and its Variations, Algorithmica 1 (1986) 251–266) — GNU diffutils, perl, ...
2. patiencediff (LCS for lines which occur exactly once on both sides, then recurse between). Cohen B., et al.: Bazaar Source Code, bzrlib/patiencediff.py, `http://www.bazaar-vcs.org/Download`
3. Bdiff longest common continuous substring — Ratcliff J. W., Metzener D. E. Pattern-matching-the gestalt approach //Dr Dobbs Journal. – 1988. – Т. 13. – №. 7. – С. 46- difflib(python), Mercurial, Revlog

не вполне адекватны

Основные математические модели:

Широко используются (по Baudiš P. Current concepts in version control systems //arXiv preprint arXiv:1405.3496. – 2014.(2009-09-11)):

1. Longest Common Subsequence LCS (Myers E.: An O(ND) Difference Algorithm and its Variations, Algorithmica 1 (1986) 251–266) — GNU diffutils, perl, ...
2. patiencediff (LCS for lines which occur exactly once on both sides, then recurse between). Cohen B., et al.: Bazaar Source Code, bzrlib/patiencediff.py, `http://www.bazaar-vcs.org/Download`
3. Bdiff longest common continuous substring — Ratcliff J. W., Metzener D. E. Pattern-matching-the gestalt approach //Dr Dobbs Journal. – 1988. – Т. 13. – №. 7. – С. 46- difflib(python), Mercurial, Revlog

не вполне адекватны

Основные математические модели:

Широко используются (по Baudiš P. Current concepts in version control systems //arXiv preprint arXiv:1405.3496. – 2014.(2009-09-11)):

1. Longest Common Subsequence LCS (Myers E.: An O(ND) Difference Algorithm and its Variations, Algorithmica 1 (1986) 251–266) — GNU diffutils, perl, ...
2. patiencediff (LCS for lines which occur exactly once on both sides, then recurse between). Cohen B., et al.: Bazaar Source Code, bzrlib/patiencediff.py, `http://www.bazaar-vcs.org/Download`
3. Bdiff longest common continuous substring — Ratcliff J. W., Metzener D. E. Pattern-matching-the gestalt approach //Dr Dobbs Journal. – 1988. – Т. 13. – №. 7. – С. 46- difflib(python), Mercurial, Revlog

не вполне адекватны

Неадекватность модели LCS

~~о п р е д е л е н и е~~ п е р е м е щ е н и ~~я~~ е с б е р е ж е н и й

п р а к т и к а м а т е м а т и к а ~~п р а к т и к а~~

Удалено:

< < о

п

Добавлено:

> > е

р е

Изменено:

< < д

> > м

е

Изменено:

< < л

> > щ

е н и е

Изменено:

< < п

> > с б

е р е

Изменено:

< < м е щ

> > ж

е н и

Изменено:

< < я

> > й

Строка: с 1 по 1

Изменено:

< п р
<

> м а т е м
>

а

Удалено:

< к
<

т и к а

Изменено:

< м а т е м
<

> п р
>

а

Добавлено:

> к
>

т и к

Строка: с 1 по 1

Удалено:

< >

ab cd ef gh ij kl mn op qr st uv e.g.

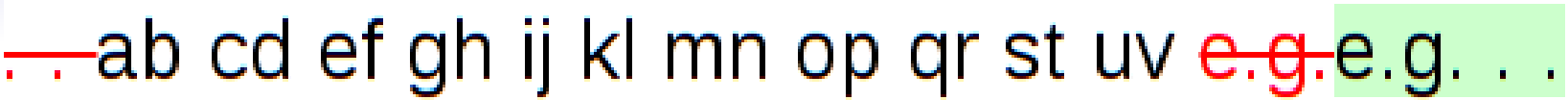
. .

Добавлено:

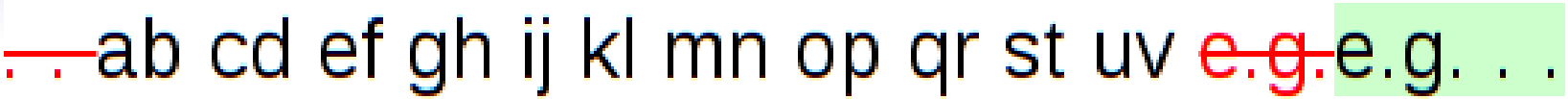
> >

ab cd ef gh ij kl mn op qr st uv e.g.

Неадекватность patiencediff



Неадекватность Bdiff



Smith, Temple F.; and Waterman, Michael S. (1981). «Identification of Common Molecular Subsequences». Journal of Molecular Biology 147: 195-197. DOI:10.1016/0022-2836(81)90087-5. PMID 7265238.