

DOI 10.34216/2587-6147-2020-1-47-32-36

05.13.01 Системный анализ, управление и обработка информации
УДК 004.912

Барило Илья Иванович

кандидат технических наук

Костромской государственной университет, г. Кострома, Россия

mail@barilo.ru

ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ НАГРУЗОЧНОГО ТЕСТИРОВАНИЯ ПРОГРАММНЫХ СРЕДСТВ ЛОКАЛИЗАЦИИ ПУТЕМ ИСПОЛЬЗОВАНИЯ ФУНКЦИИ ПОСТРОЕНИЯ ТЕСТОВОГО СЛОВАРЯ ПЕРЕВОДА

В статье рассмотрены особенности внедрения программного обеспечения (ПО) для локализации разрабатываемого ПО на начальном этапе его разработки; составлен список проблем, необходимых к рассмотрению при выборе ПО для локализации; предложен вариант решения основных проблем путем разработки функции построения тестового словаря, позволяющего масштабировать его под разрабатываемое ПО; представлены результаты исследования ряда существующих словарей локализации популярного ПО; сделан вывод о необходимости расчета значений параметров функции построения в зависимости от типа ПО; проведен анализ словарей локализации и подготовлен ряд параметров функции построения тестового словаря под требования разработчика.

Ключевые слова: локализация, словарь, параметры, разработка, нагрузочное тестирование, программный продукт, длина текстового фрагмента.

Локализация программного продукта является важной частью его разработки, позволяет сделать интерфейс более понятным и удобным пользователю, а также расширяет целевую аудиторию пользователей программы.

Перед разработчиками программного обеспечения (ПО) встает вопрос выбора оптимального инструмента локализации для своего ПО, которое должно быть интегрировано в процесс его разработки, а также корректно работать в составе разрабатываемого ПО.

Для этого разработчикам необходимо провести тестирование различных инструментов локализации, которые, как правило, объединяет наличие словаря, содержащего локализованные фразы, подставляемые в ресурсы разрабатываемой программы и, в частности, ее интерфейс.

Одной из проблем тестирования является отсутствие на начальном этапе разработки ПО достаточного объема кода и словаря локализации, которые не позволяют корректно провести нагрузочное тестирование [1].

Еще одной проблемой в ряде случаев является замедление скорости подстановки слов из увеличенного словаря. Так, некоторые инструменты локализации производят интеграцию

своих модулей в разрабатываемое ПО и выполняют подстановку необходимых локализованных фраз при работе программы, что при дальнейшем увеличении количества фраз в словаре приведет к замедлению работы разрабатываемой программы.

Скорость процесса локализации при работе программы зависит от количества фраз в программе для перевода и объема словаря локализации. В случае подстановки слов в процессе работы программы ключевым фактором замедления работы программы является объем словаря.

Из перечисленных проблем, влияющих на скорость локализации, основными являются:

- количество фраз в исходном коде;
- количество фраз в словаре;
- количество фраз в программе для локализации.

В ходе исследования последняя из перечисленных проблем была исключена из задачи разработки параметров масштабируемого словаря, так как она зачастую зависит от языка программирования, формы представления ресурсов и пр. Кроме того, данный пункт обычно можно легко линейно интерполировать.

Для нагрузочного тестирования программы локализации необходимо проверить ее работу при большом количестве фраз в словаре, до

которого теоретически может вырасти объем словаря в будущем.

Ключевым фактором, влияющим на объем словаря, кроме количества слов в нем, являются длины фраз, так как в интерфейсе программы могут быть как короткие фразы, так и длинные пояснения, которые повлияют на скорость подбора локализованного фрагмента.

Для решения перечисленных проблем предлагается использовать функцию с набором параметров, позволяющую формировать тестовый словарь для нагрузочного тестирования средств локализации разрабатываемого ПО.

Для определения параметров функции были выбраны крупные известные проекты с открытым исходным кодом. В качестве объекта исследования были использованы файлы перевода в формате *.po, в которых не только перечислены исходные строки и строки переводов, но и количество ссылок на них из программы, а также другая информация.

Формат файлов переводов и особенности локализации выбранных проектов не имеют значения, так как в ходе исследования определяются параметры функции, которые влияют на содержание словаря, а не на его форму представления и особенности его использования.

Кроме определения параметров, необходимо рассчитать их значения, а также проверить влияние на них особенностей и назначения разрабатываемой программы.

Для расчета параметров масштабируемого словаря локализации были использованы словари популярных программ, в которых процент перевода фраз выше 90 %:

- Lazarus (Fast Pascal) [2];
- Gimp [3];
- Inkscape [4];
- FileZilla [5];
- Dia [6];
- PostgreSQL [7].

Для дальнейшего анализа словарей проведено их объединение в единый словарь с удалением дублирующей информации. Распределение количества фрагментов слов в зависимости от их длины отражено на рис. 1.

Все рассмотренные в исследовании словари для ключевых слов используют английский язык, и именно по нему были рассчитаны параметры словаря.

Следует отметить, что в используемых форматах файла словарей перевода ключевые слова совпадают с исходным текстом фраз, поэтому формат файла не влияет на расчеты.

Из рис. 1 видно, что для построения тестового словаря перевода можно использовать диапазон от 1 до 500 слов, игнорируя остальные для упрощения параметров функции его создания.

Из графика (рис. 2) видно, что наибольшее значение в распределении длин фрагментов текстов имеет диапазон от 1 до 100.

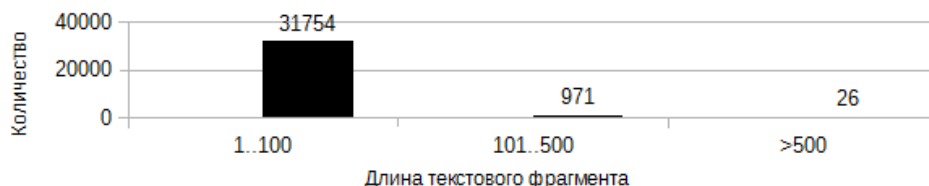


Рис. 1. Количество слов, учитываемых в исследовании

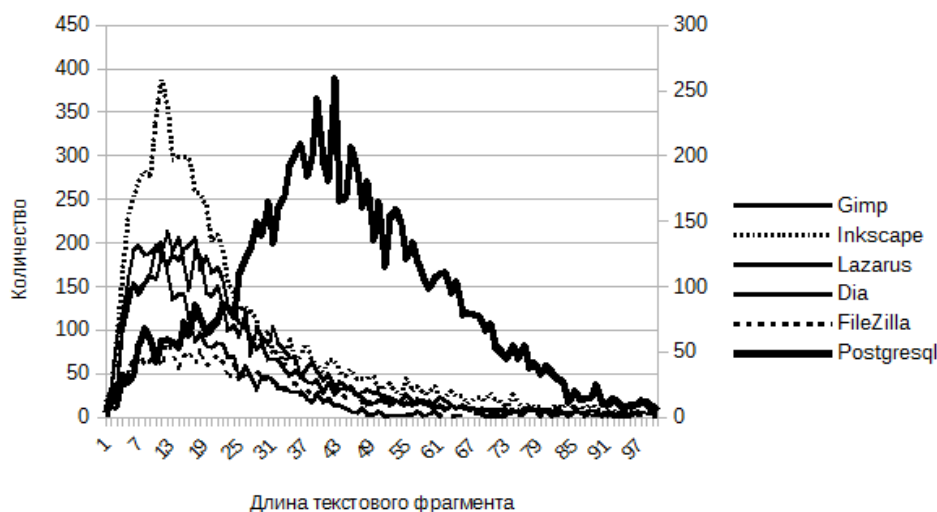


Рис. 2. Распределение количества фрагментов текста в словарях с учетом их длины

Из графика на рис. 3 видно, что словарь для программы PostgreSQL существенно отличается от словарей других программ. Для PostgreSQL характерно минимальное количество с наименьшей длиной и наибольшее количество переводимых фрагментов в диапазоне длин 50...80. Рассмотренные программы являются яркими представителями программ с оконными интерфейсами, обладающих большим количеством элементов форм и подсказок, в то время как PostgreSQL имеет консольную часть и содержит более длинные текстовые сообщения.

На основе такого явного отличия параметров текста в словарях был сделан вывод, что для разных типов ПО следует использовать различные значения параметров функции построения тестового словаря для каждого типа ПО, поэтому из дальнейшего исследования исключен PostgreSQL.

В связи с этим график исходных данных приобрел вид, представленный на рис. 4.

Из рис. 4 видно, что для основного анализа текста подходят те же диапазоны, которые были указаны ранее. Рассмотрим графики по основным выявленным периодам со значениями для каждой программы.

График на рис. 5 показывает, что соотношение длин текстовых фрагментов в словарях для анализируемых программ стало более схожим.

На основании полученных результатов исследования был построен усредненный график (см. рис. 5) по всем программам с указанием процентного соотношения количества фрагментов и их периода длины фрагментов текста.

График на рис. 6 также подтверждает схожую для всех программ зависимость количества текстовых фрагментов от их длины.

На рис. 6 видно, что для всех программ средние длины текстовых фрагментов по выбранным периодам примерно совпадают, поэтому был построен усредненный график (рис. 7).

Для каждого диапазона был проведен расчет средних значений длины для дальнейшего масштабирования. Рассчитанные значения представлены на рис. 8.

Рассчитанные диапазоны, представленные на рис. 7, позволяют использовать их в качестве параметров функции для построения словаря произвольного размера для нагрузочного тестирования разрабатываемого ПО, учитывая дальнейший рост объема словаря локализации.

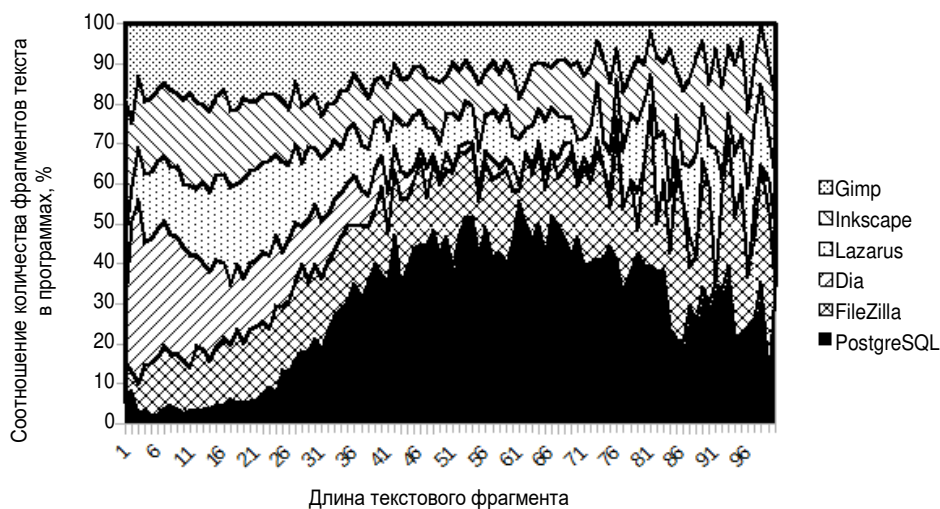


Рис. 3. Соотношение количества переводимых текстовых фрагментов в программах в зависимости от их длины

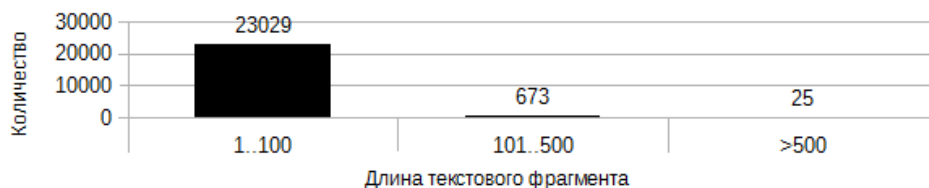


Рис. 4. Количество слов, учитываемое в исследовании

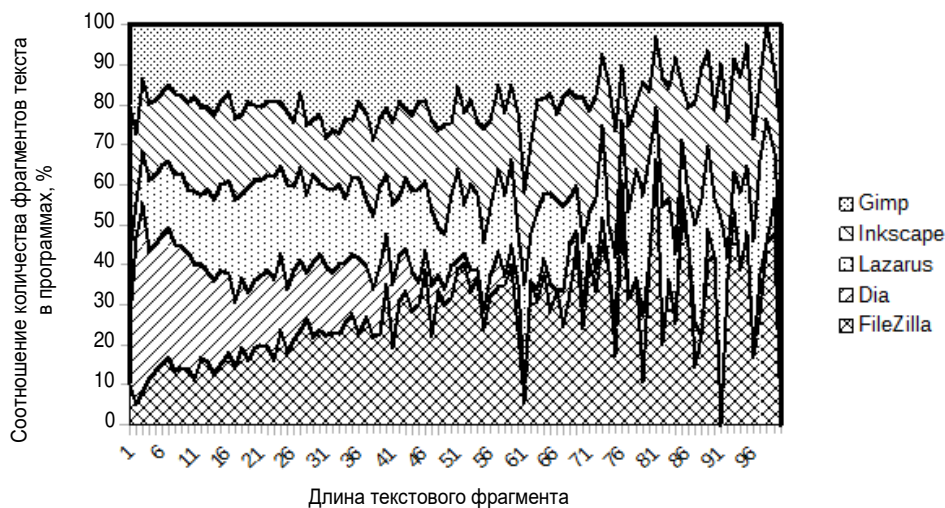


Рис. 5. Соотношение количества переводимых текстовых фрагментов в программах в зависимости от их длины

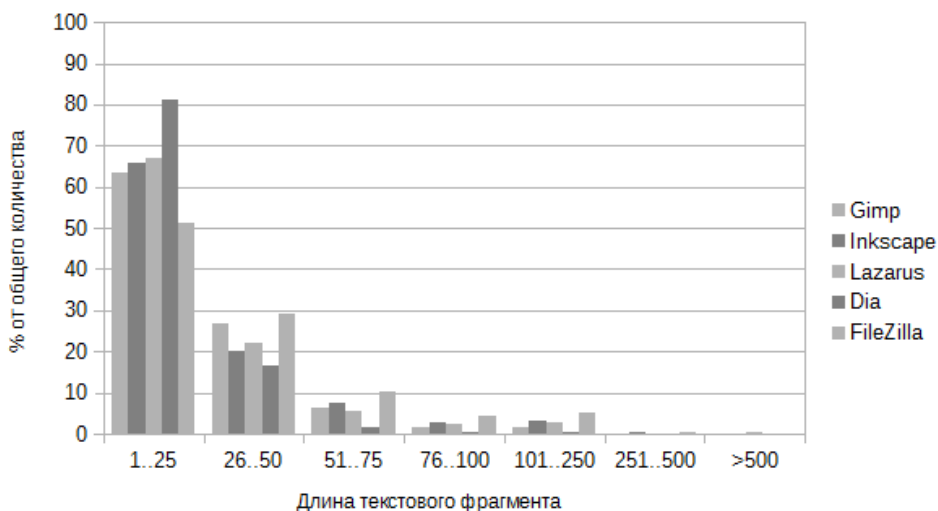


Рис. 6. Распределение количества текстовых фрагментов в зависимости от их длины для каждой программы



Рис. 7. Распределение количества текстовых фрагментов в зависимости от их длины

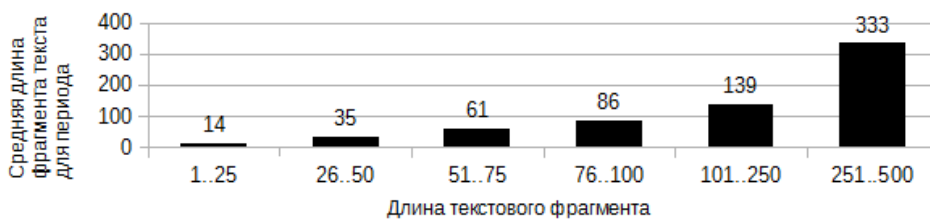


Рис. 8. Средняя длина текстовых фрагментов для каждого периода длин

Значения из графика, представленного на рис. 8, можно использовать в качестве значений параметров функции в случае разрабатываемой программы, схожей с рассмотренными проектами, имеющими оконный интерфейс, обладающими большим количеством элементов форм и подсказок без консольной части.

Определение значений параметров по всем видам ПО и их зависимость от особенностей ПО выходит за рамки данного исследования.

Таким образом, в процессе определения параметров функции построения тестового словаря переводчика был проведен анализ основных характеристик, влияющих на скорость локализации программы, на основе словарей локализации популярных проектов.

Сделан вывод, что для построения тестового словаря функции необходимы 13 парамет-

ров: количество слов в словаре, набор из шести коэффициентов для определенных периодов длин (см. рис. 7) и соответствующих им средних значений (см. рис. 8).

Выявлены зависимости параметров функции построения тестового словаря от особенностей ПО, что указывает на необходимость использования различных коэффициентов и средних значений длин.

В исследовании рассчитаны значения для программ, имеющих оконный интерфейс и обладающих большим количеством элементов форм и подсказок.

Определены параметры функции построения тестового словаря для нагрузочного тестирования ПО локализаторов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Уиттакер Дж., Арбон Дж., Каролло Дж. Как тестируют в Google. – СПб. : Питер, 2014. – 320 с. : ил.
2. All Free Pascal sources. – URL: <https://svn.freepascal.org/svn/lazarus/trunk/languages> (дата обращения: 03.02.2020).
3. Translation of GIMP to Russian. – URL: <https://110n.gnome.org/POT/gimp.master/gimp.master.ru.po> (дата обращения: 03.02.2020).
4. Inkscape. – URL: <https://110n.gnome.org/POT/gimp.master/gimp.master.ru.po> (дата обращения: 03.02.2020).
5. Translation of ru.po. – URL: <https://110n.gnome.org/POT/gimp.master/gimp.master.ru.po> (дата обращения: 03.02.2020).
6. Dia 0.97. – URL: <https://110n.gnome.org/POT/dia.master/dia.master.ru.po> (дата обращения: 04.02.2020).
7. Postgres. – URL: https://github.com/postgres/postgres/releases/tag/REL_11_6 (дата обращения: 03.02.2020).

REFERENCES

1. Uittaker Dzh., Arbon Dzh., Karollo Dzh. Kak testiruyut v Google. – SPb. : Piter, 2014. – 320 s. : il.
2. All Free Pascal sources. – URL: <https://svn.freepascal.org/svn/lazarus/trunk/languages> (data obrashcheniya: 03.02.2020).
3. Translation of GIMP to Russian. – URL: <https://110n.gnome.org/POT/gimp.master/gimp.master.ru.po> (data obrashcheniya: 03.02.2020).
4. Inkscape. – URL: <https://110n.gnome.org/POT/gimp.master/gimp.master.ru.po> (data obrashcheniya: 03.02.2020).
5. Translation of ru.po. – URL: <https://110n.gnome.org/POT/gimp.master/gimp.master.ru.po> (data obrashcheniya: 03.02.2020).
6. Dia 0.97. – URL: <https://110n.gnome.org/POT/dia.master/dia.master.ru.po> (data obrashcheniya: 04.02.2020).
7. Postgres. – URL: https://github.com/postgres/postgres/releases/tag/REL_11_6 (data obrashcheniya: 03.02.2020).