

Библиотека конкурентных структур данных

- Разработчик: [Максим Хижинский](#)
- Репозиторий: [github](#)
- Continuous integration: [jenkins](#)

Доработки

Архитектурные


[Рефакторинг SMR-алгоритма cds::gc::DHP](#)

Технические

- [Bounded object pool](#)
- [Concurrent Data-Structures Through Explicit Timestamping](#)
- [Flat Combining](#)
- [Алгоритмы для доработки в libcds](#)

Инфраструктурные

Тестирование

1. Сделать 1 тест для примера с использованием *gtest*
2. Разделить тесты через суффикс в имени на:
 - Unit: произвольное имя без суффикса
 - Нагрузочные: имя теста с суффиксом *-stress*
 - Интеграционные: 
3. По умолчанию тестирование осуществлять без нагрузочных тестов через использование системы фильтрации тестов *Google Test*
4. Разделить тесты по исполняемым файлам по признаку тестируемой сущности, если где не разделено

Continuous Integration

1. Запустить ветку *integration* в автотестирование на архитектуре *x86_64-suse-linux-4.8.3*
2. Срастить *ctest* с *jenkins* для ведения истории тестов и их удобного представления
3. Добавить статический анализ кода *cppcheck*
4. Добавить анализ покрытия кода тестами
5. Расширить тестирование архитектурой сборки *i686-linux-gnu-4.8.4*
6. Полностью успешные сборки завершать построением пакетов под целевые платформы и

выкладывать в что-нибудь типа *artifactory*

7. Добавить возможность выпуска релиза с созданием тэга в *git* и выкладыванием собранных пакетов в что-нибудь типа *artifactory*
8. По стабилизации прохождения тестов создать почтовую группу и внедрить рассылку
9. Активировать профилирование модульных тестов *valgrind* с выкладыванием результатов
10. Актуализировать документацию по библиотеке в автоматическом режиме выкладывая результаты прогона *doxygen*
11. Валить сборку при непроходящих тестах (сейчас видно, что тесты упали, однако сборка сама считается успешной, когда тесты стабилизируются — по хорошему нужно валить)
12. Ограничить время выполнения 1 теста 15 часами
13. Анализировать предупреждения gcc

From:

<http://wiki.osll.ru/> - **Open Source & Linux Lab**

Permanent link:

<http://wiki.osll.ru/doku.php/projects:libcds:start?rev=1477866957>

Last update: **2016/10/30 22:35**

