

## 1. ДРАКОН

Графический язык программирования, созданный для нужд советской космической программы, но несколько ее переживший. Использует свою отдельную философию программирования, опирающуюся на строгое математическое обоснование каждого шага программы (референс в сторону функциональности). Возможно использование для обучения программированию и написанию алгоритмов. Есть упоминания об интеграциях с другими языками программирования (текстовыми) : ДРАКОН-с, ДРАКОН-java, ДРАКОН-C#, ДРАКОН-PASCAL.

Касательно последнего: блок-схема - основной способ создания программы на ДРАКОНе - конвертируется в код на Pascal'e, но при этом проект имеет неочевидный интерфейс, перегружен блок-схемами - у пользователя нет возможности тонкой настройки рисунка программы, что отталкивает от использования ( в противопоставление Blueprint от Unreal Engine). Конвертируемость в код на Pascal'e все еще не проверена. Полезность документации не проверена. Выглядит колхозно.

В целом, требует проверки основательность связи ИС ДРАКОН и ДРАКОН, который использовался в проекте Буран.

## 2. Blueprint unreal\ Blueprint visual scripting

Безусловный фаворит. Богатый tutorial (правда, на английском), возможно использование для обучения программированию и написанию алгоритмов, большое сообщество пользователей, интуитивно понятный UI, приятный глазу. НО: ограничен движком - проекты, сторонние движки, на нем не попишешь; платная подписка на использование Unreal Engine - стоит денег. Конвертируется в C++, не всеобъемлющ, поэтому в некоторых моментах требует дописывания модулей все на том же C++.

Имеет смысл заимствовать визуальное оформление, используемое в этом языке, т.к. оно user-friendly, более гибкое нежели остальные рассмотренные варианты (ДРАКОН, HiAsm). Уточнение: не совсем корректно говорить о том, что Blueprint конвертируется в C++ - он написан на C++, поэтому любая конструкция, созданная на Blueprint, естественным образом имеет представление на C++.

## 3. HiAsm, среда Hion

Темная лошадка в нашем списке. Известно, что это графический язык, но неизвестно во что он конвертируется (конвертируется ли?) (пока, буду еще искать). Из плюсов: имеет web-версию, что позволяет пользоваться им прямо в браузере (ссылка есть в #useful\_info по поиску Предметная область). Выглядит несколько лучше, нежели ДРАКОН, хотя наличествуют нарекания касательно удобства интерфейса, что, думаю, после некоторого времени работы на нем, пропадет.

## 4. FBD

Графический язык, используемый для программирования PLC. Насколько я понял, вполне справляется с поставленной перед ним задачей. Для нас представляет малый интерес.

## 5. VAX

Среда, позволяющая построить SQL-запрос посредством схемы, т.е. графически. Проект небольшой, написан на JavaScript, требует внесения шаблонов запросов, написанных JavaScript перед применением в конкретной области (общие шаблоны могут не подходить).

## 6. Blockly

Разрабатывается и поддерживается с 2012 года. Написан на JavaScript. Основная и, кажется, единственная цель - язык для обучения программированию школьников. Состоит из графического редактора, где программа составляется из визуальных блоков, и генератора кода. Как мне кажется, хоть язык и обладает отличительной чертой всех визуальных языков - наглядностью, является уж очень простым. Надо делать нечто более серьезное.

Если говорить в общем, то основные плюсы всех подобных языков - это относительная простота освоения, скорость разработки полнофункционального прототипа программы - касательно скорости работы программы, созданной с помощью подобных систем разработки, сказать ничего не могу по понятным причинам; возможность использования для обучения (постараемся сделать в проекте акцент именно на эту особенность). Часто упоминаются СВР и использования некоторых визуальных языков в них. О UI, реализации функционала и пр. будем вести дальнейшие разговоры. Обсуждения и предложения приветствуются.