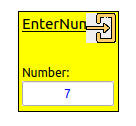
Существует 3 основных типа блоков:

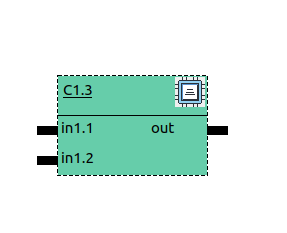
1. Basic - базовый блок ( простейший ). Внутри блока пользователь может описать сценарий работы блока на языке JavaScript.

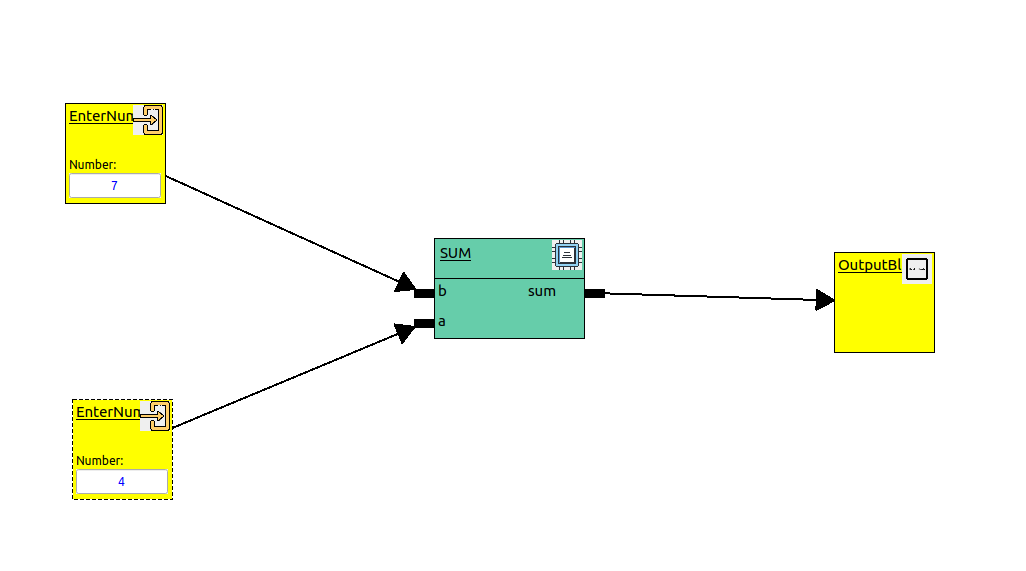


2)SPARQL - блок для запроса к RDF-хранилищам данных на основе Jena Fuseki Sparql Server. Сам запрос формируется в редакторе SPARQL. JavaScript генерируется по запросу Модуля Исполнения Блоков или Модуля Генерации Скрипта Композитного Блока на основе сетевого запроса.



3)Composite - блок для возможности создания сборных блоков. Создаётся в специальном редакторе для создания композитных блоков. Перед выполнением программы заменяется на блоки, которые он содержит в себе. Таким образом в схеме выполнения остаются только SPARQL и Basic блоки.





1. Разберём тестовый пример с использованием Based ( Sparql не отличается исполнением от базового за исключением использованием сетевого подключения ) и Composite блоков.

И так на схеме находится 4 блока:

1. Based блок для ввода числа
2. Based блок для ввода числа
3. Composite блок для суммирования
4. Based блок для вывода

Перед исполнением программы начнётся первичная генерация JavaScript кода. Далее начнётся запуск. Запуск работает по принципу LILO (last in — last out: последним пришёл — последним вышел). Программа циклически ищет блоки для исполнения и затем выполняет. Блоки для исполнения подбираются по принципу готовности ( все входящие в них блоки уже отработали ).

Принцип исполнения блоков:

1. Запускается бесконечный цикл
2. Запускается цикл перебора блоков
3. Если блок готов к работе, то запускается выполнение блока, иначе следующий
4. Если выполнился хоть 1 блок за цикл, то возвращаемся на пункт 2, иначе заканчивает выполнение блоков.

Трасировка на примере:

1. В первом запуске цикла выполнятся блоки 1 и 2.
2. При втором запуске цикла выполнится Basic блок SUM внутри композитного SUM.
3. При третьем запуске цикла выполнится блок 4.
4. При четвёртом запуске цикла не будет выполнено ни одного блока, следовательно процесс выполнения завершается.

Принцип генерации скрипта Sparql блоков.

1. На диаграмме находятся области, которые можно соединять создавая при этом связи MINUS, UNION и тд.
2. В каждой области связи создаются по принципу :
3. Блок начала стрелки - Объект
4. Блок конца стрелки - Субъект
5. Стрелка - Отношение между ними
6. Далее области соединяются друг с другом.
7. Полученный Sparql запрос конвертируется в сетевое представление (URL-encode).
8. Добавляются строки для создания сетевого запроса с полученным URL-текстом
9. Полученный скрипт возвращается из функции