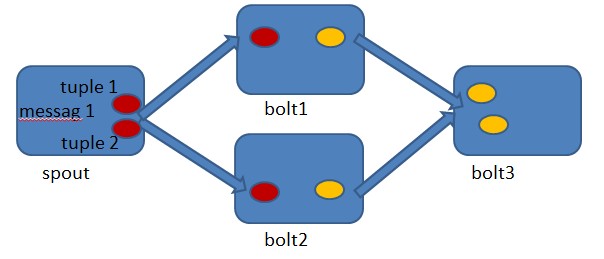
**IdeoEngine Real-time Processing**

IdeoEngine is a C++ based distributed real-time processing system. Different from the batch processing system like Apache Hadoop, IdeoEngine uses stream model to process data. It also supports multi-language interfaces, such as Python, and Java.



**Basic concepts**

**Topology**

The logic for a real-time application is packaged into a IdeoEngine topology. A IdeoEngine topology is analogous to a MapReduce job. One key difference is that a MapReduce job eventually finishes, whereas a topology runs forever.

**Stream**

The stream is an important abstraction in IdeoEngine. A stream is an unbounded sequence of tuples that is processed and created in parallel in a distributed fashion. Streams are defined with a schema that names the fields in the stream's tuples.

**Tuple**

Tuple is the data unit transferred in stream. The spout and bolt need to use tuple to organize the data. Tuples can contain integers, longs, shorts, characters, floats, doubles and strings.

**Spout**

A spout is a source of streams in a topology. Generally spouts will read tuples from an external source and emit them into the topology. Spouts can either be reliable or unreliable. A reliable spout is capable of replaying a tuple if it failed to be processed by IdeoEngine, whereas an unreliable spout forgets about the tuple as soon as it is emitted.

**Bolt**

All processing in topologies is done in bolts. Bolts can do anything from filtering, functions, aggregations, joins, talking to databases, and more.