

**May 2012**

**MADALGO seminar by Irit Katriel**

**Streamulus - A language for real-time event stream processing**

**Abstract:**

Continuous computations over event streams require a data structure (typically a graph) that describes how the input streams are to be processed. A language for specifying such a computation would ideally hide the construction of the graph from the user.

General purpose programming languages such as Java and C++ are considered to be ill suited for such tasks, prompting event processing systems such as StreamBase and Esper to support one of the variations of StreamSQL, i.e., a language that augments SQL by adding sliding window semantics as well as the possibility to write user-defined operators and embed them in continuous queries. These languages are very useful for many applications. Like SQL, they can be learned by non-programmers and used relatively easily.

Streamulus takes a different approach to the stream programming problem. It is a C++ domain-specific embedded language (DSEL), meaning that the statements of the language are written in valid C++ syntax. The programmer needs to specify only what happens to a single event of the stream. For example, if  $x$  and  $y$  are streams, the expression  $(x + y)/2$  is the stream of point-wise averages of  $x$  and  $y$ . The task of constructing the computation graph is delegated to the C++ compiler. This magic is achieved by use of template metaprogramming techniques via the Boost Proto library, which I will review in the talk.

Streamulus is not intended as a replacement for the StreamSQL based systems. Its target users are C++ programmers who build real-time systems and need a lightweight tool that simplifies the task of writing stream processing code. For those users, it offers an expressive language for efficient stream computations.