The LIEBRE Stream Processing Engine

Since I started by Ph.D. back in 2008, I spent a considerable amount of time digging into the internals of stream processing engines because of my research topics. Pioneer engines like Aurora and Borealis were not easy to learn. During the Ph.D. I designed and developed StreamCloud, a parallel and distributed stream processing engine that allows for the parallel execution of streaming operators. The parallelization techniques in StreamCloud are now used in engines such as Storm and Flink.

Once the Ph.D. was over, I continued my research in data streaming and learned to use new systems, like the aforementioned Storm and Flink. Things got way easier to get started with them than they were before, and a simple hello world example is actually a matter of minutes. Still, I always had (and still have) mixed feelings with these systems. On the positive side, they are very powerful and supported by a large community. On the less positive side, they force you to design and implementation decisions that you could find not optimal (remember my perspective is the one of a researcher in data streaming...).

Of course, one could always modify the internal architecture of such systems, but the engineering effort is considerable, and would consume time I usually do not have.