



Programming Models for Data-intensive Applications



Bachelor Project Proposal, WS 2011/2012 – SS 2012
Software Architecture Group, Prof. Dr. Robert Hirschfeld

Objectives

Many of today’s business applications are data-intensive. Application logic is implemented using a general-purpose programming language. Data access is either generated or realized directly using SQL. There is an impedance mismatch between programming languages and data access that becomes obvious in both the interface to the programmer as well as the runtime system: Because of the different languages, programmers need to perform rather complicated mappings between different concepts and their representations, and runtimes need to carry out sophisticated optimizations to achieve reasonable system performance. Hence, programs seem to be much larger and more complex than necessary.

Because of the increasing number of such applications and the huge data volume these applications have to process, this project will revisit and evaluate relevant programming concepts and runtime mechanisms.

Approach and Activities

The goal of this project is to better understand the technical constraints for developing data-intensive applications and to suggest programming language concepts that empower programmers to concisely express their thoughts as well as runtime mechanisms to execute them efficiently.

Following an iterative approach, project participants will explore typical application scenarios and promising programming languages, concepts, and paradigms. Programming models of interest include but are not limited to those of Clojure, Erlang, F-Script, APL, Fortress, Actors, LINQ, MapReduce, and OMeta.

Project participants are encouraged to propose, implement, and evaluate new language constructs and runtime mechanisms.

Organization

A group of about six (6) students may take part in the project. Project participants are expected to be self-organized and have a strong interest in the design and implementation of programming languages. The project will be carried out at the Hasso Plattner Institute in Potsdam. Project participants communicate with our partner on a regular basis. Expected results include working prototypes accompanied by appropriate documentation.

Partners and Contact

Dr. Andreas Raab, Dr. Christian Mathis, Cafer Tosun
SAP Innovation Center Potsdam
<http://www.sap.com/>, cafer.tosun@sap.com

Prof. Dr. Robert Hirschfeld, Bastian Steinert, Michael Perscheid
Software Architecture Group, Hasso-Plattner-Institut, Potsdam
<http://www.hpi.uni-potsdam.de/swa>, hirschfeld@hpi.uni-potsdam.de

```
«SYNTAX PROGRAM
PRIMARY = «ID «OUT('LD ' #) /
«NUMBER «OUT('LDL' #) /
«(' EXP ' )' »»
TERM = PRIMARY S('«' PRIMARY «OUT('HLT' ) »»
EXP1 = TERM S('«' TERM «OUT('ADD' ) /
«-» TERM «OUT('SUB' ) »»
EXP = EXP1 ( '«' EXP1 «OUT('EQU' ) / «EMPTY' »»
ASSIGNST = EXP '=' «ID «OUT('ST ' #) »»
UNTILST = «UNTIL' «LABEL #1 EXP «DD' «OUT('BTP' #2)
ST «OUT('B ' #1) «LABEL #2 »»
CONDITIONALST = «IF' EXP «THEN' «OUT('BFP' #1)
ST «ELSE' «OUT('B ' #2) «LABEL #2
ST «LABEL #2 »»
IOST = «EDIT' («' EXP '«' «STRING
«OUT('EDT' #) ' )' /
«PRINT' «OUT('PNT' ) »»
IDSEQ1 = «ID «LABEL # «OUT('BLK 1' ) »»
IDSEQ = IDSEQ1 S('«' IDSEQ1) »»
DEC = «REAL' «OUT('B ' #1) IDSEQ «LABEL #1 »»
BLOCK = «BEGIN' (DEC '«' / «EMPTY)
ST S('«' ST) «END' »»
ST = IOST / ASSIGNST / UNTILST /
CONDITIONALST / BLOCK »»
PROGRAM = BLOCK «OUT('HLT' )
«OUT('SP 1' ) «OUT('END' ) »»
«END
```