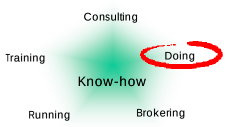
Project "HIT: High-performance Inference engine for Time series"

Keywords Business Rules Approach (BRA), Tailored Inference Engines, Declarative Programming

The Customer

Our customer is a leading global provider of precision instruments and services. The company is one of the world's largest manufacturer of weighing solutions for laboratory, industrial and food retailing applications. The customer's automatic chemistry business unit focuses on accelerating the R&D process and scale-up to production in pharmaceutical and chemical industries, specifically the discovery, development and manufacture of chemical compounds. The automatic chemistry business unit products shorten time-to-market for these customers by accelerating the synthesis, purification and process development of chemical compounds through integration, automation and sophisticated software solutions.

The Project



The automatic lab reactors combined with several in-situ reaction analysis systems support product and process development through systematic and data-based automation of experiments in drug research and development. They allow the measurement and control of all important process parameters and determination of the complete mass and heat balance over the course of the entire chemical process. The lab batches are performed on a milliliter to liter scale under conditions, which closely mirror the (pilot) plant reality. The results are used directly for the development and optimization of chemical production processes. The family of automatic lab reactors is controlled by a controlling software, which allows the chemist to design, run, monitor and evaluate experiments in a very flexible way.

The Challenge

The evaluation of chemical experiments requires complex mathematical calculations of physical quantities over time. Such physical quantities are usually derived from other physical quantities and depending on various conditions and user settings, different formulae must be applied to calculate a physical quantity. Furthermore, it must be possible to easily extend these calculation rules to adapt them to specific needs of an end-user (the chemist).

$$mr(R_r)(t) = \sum_{x=1}^{nSubst} m\_Subst_x(R_r)(t)$$

$$\begin{aligned} m\_Subst_x(R_r)(t) = {} & (If\ \tilde{S}ubst_x \in FirstFill_r\ Then\ FirstFill_r^m\ Else\ 0) + \\ & \sum_{k=1}^{nDos} (If\ Dos_k \in R_r \wedge \tilde{S}ubst_x \in Dos_k \wedge Dos_k{=}active\ Then\ Balance(k,t)\ Else\ 0) + \\ & \sum_{k=1}^{nHDos} (If\ HDos_k \in R_r \wedge \tilde{S}ubst_x \in HDos_k \wedge HDos_k{=}active\ Then\ m^{HDos}(k,t)\ Else\ 0) \end{aligned}$$

$$m^{HDos}(k,t) = \begin{cases} If\ (t < \tilde{t}_k^{start}) : 0 \\ If\ (\tilde{t}_k^{start} \le t \le \tilde{t}_k^{end}) : \left(\dfrac{\tilde{m}_k^{HDos(end)}}{\tilde{t}_k^{end} - \tilde{t}_k^{start}} \right)(t - \tilde{t}_k^{start}) \\ If\ (t > \tilde{t}_k^{end}) : \tilde{m}_k^{HDos(end)} \end{cases}$$

This kind of calculations must be performed over large time series and in a very efficient manner. Since it is possible to evaluate an experiment while it is still running, it is also absolutely necessary that the evaluation software has a limited and deterministic memory and performance footprint.

Our Contribution

People from KnowGravity Inc. are involved in the development of the controlling software for many years, providing services including UML training, architecture consulting and implementation. In order to improve the flexibility of the evaluation of an experiment, KnowGravity Inc. is currently developing a highly specialized inference engine tailored for efficient time series computations. This engine called HIT (High-performance Inference engine for Time series)
offers the following features:

- Declarative statement of mathematical formulae
  - Complex computations of physical values and physical values over time (time series)
  - Declarative statement of rules to choose applicable formula
    (`if condition then X is formula1 else X is formula2`)
- Support of complex expressions
  - Boolean operations: not, and, or, implies, <, =<, =, \=, >=, >, in, subset_of
  - Set operations: set_of, union, intersection, difference, minus, map
  - Numeric operations: +, -, *, /, //, ^, abs, acos, aln, alog, asin, atan, cos, exp, fp, int, ip, ln, log, max, min, mod, rand, sign, sin, sqrt, tan
  - Time series operations: points, sum, average, integral, compress (simple, average, minpeak, maxpeak), expand (hold, linear, quadratic)
- Understandable and easy changeable rule language

The inference engine is implemented as a meta-language on top of LPA WIN-PROLOG, a high-performance implementation of the rule-based language Prolog. Calculations are per-formed in a backward chaining way and supported by extensive caching mechanisms. So, the inference engine is able to handle high data volumes (time series of up to a million points in time and more than 100'000 complex inferences per second on a standard PC). The engine is designed as a component embeddable into Microsoft COM and .NET host applications.

Example Rules

Below, the formulae shown above are represented in HIT's rule language:

```
rule1 :: mr(R)@T is sum(m_Subst(S,R)@T, S in substances).

rule2 :: m_Subst(S,R)@T is first_fill(S,R) +
         sum(mr_dos(S,R,K)@T, K in 1..nDos) +
         sum(mr_hdos(S,R,K)@T, K in 1..nHDos).
rule2a:: if S member_of first_fills then first_fill(S,R) is first_fill_m(R)
                                    else first_fill(S,R) is 0.

rule2b:: if dos(K).reactor = R and dos(K).substance = S and dos(K).active
         then mr_dos(S,R,K)@T is balance(dos(K).measure)@T
         else mr_dos(S,R,K)@T is 0.
rule2c:: if hdos(K).reactor = R and hdos(K).substance = S and hdos(K).active
         then mr_hdos(S,R,K)@T is mr_hdos(K)@T
         else mr_hdos(S,R,K)@T is 0.

rule3a:: if T < hdos(K).t_start then mr_hdos(K)@T is 0.

rule3b:: if T in T hdos(K).t_start..hdos(K).t_end
         then mr_hdos(K)@T is hdos(K).mass /
              (hdos(K).t_end - hdos(K).t_start) * (T - hdos(K).t_start).

rule3c:: if T > hdos(K).t_end then mr_hdos(K)@T is hdos(K).mass.
```

Contact

KnowGravity Inc
Badenerstrasse 808
8048 Zürich
Switzerland

Phone +41 (0) 1 434 20 00
Fax +41 (0) 1 434 20 09
Internet www.knowgravity.com
Email info@knowgravity.com