StratMaster

A language for algorithmic trading.
The StratMaster Team

Project Manager: Jintack Lim
Language Guru: Vincent Mierlak
System Architect: Xingying Liu
System Integrator: Moning Zhang
Testing & Validation: Enrui Liao
The StratMaster Team

Project Manager: Jintack Lim
Language Guru: Vincent Mierlak
System Architect: Xingying Liu
System Integrator: Moning Zhang
Testing & Validation: Enrui Liao
Motivation:

- Computerized trading accounts for over half of all trading volume
- Yet, no domain-specific language for algorithmic trading
- Proprietary software written mostly in C++ (and Java)
Introducing: StratMaster

- Concise and intuitive syntax
- Concurrent
- Modular blocks for algorithms and functions
- Designed for data
- Accounting interface for seamless integration into trading platforms
StratMaster: The Brain Of A Trading Platform
Computational Model

- Process-driven
- Data-driven
- Action-oriented
Demo 1: “Hello World”

/* Single Order in a STRATEGY */

USE ACCOUNT ac_master;

STRATEGY hello
{
    BUY { WHAT: SECURITY(EQTY(ZBRA)).AMOUNT(10000).PRICE(USD(85.17)); } 
}
StratMaster Output: Orders

++++++++++++++++++STRATMASTER CONFIRMATION++++++++++++++++++
[2015-05-10 16:16:01] YOU BOUGHT: 10,000 SHARES OF ZBRA AT USD 85.17

>>>>>> ORDER PLACED BY hello

++++++++++++++++++END CONFIRMATION+++++++++++++++++++++++++

ACCOUNT summary for ac_master:
[  1] ZBRA: Total Shares: 10,000 Current Value: 907,200.00

Total Value of Securities: 907,200.00
Cash Balance Remaining: 99,148,300.00
-------------------
Total Account Value: 100,055,500.00
Demo 2: Buy Low, Sell High

- Start with $100 million
- BUY when price < $10
- SELL when price > $15
- Don’t let cash drop below $5 million

➢ 2 STRATEGYs, 2 ALGORITHMs in parallel
# Building Blocks: Types & Objects

<table>
<thead>
<tr>
<th>Process Blocks:</th>
<th>Procedures:</th>
</tr>
</thead>
<tbody>
<tr>
<td>● STRATEGY</td>
<td>● FUNCTION</td>
</tr>
<tr>
<td>● ALGORITHM</td>
<td>Objects:</td>
</tr>
<tr>
<td>USE Elements:</td>
<td>● CURRENCY</td>
</tr>
<tr>
<td>● ACCOUNT</td>
<td>● SECURITY</td>
</tr>
<tr>
<td>● DATAFEED</td>
<td>● POSITION</td>
</tr>
<tr>
<td>● DATABASE</td>
<td>● INT, DOUBLE, PRICE</td>
</tr>
<tr>
<td>● EXCHANGE</td>
<td>Orders: BUY / SELL</td>
</tr>
</tbody>
</table>
Building Blocks: Types & Objects

STRATEGY
Building Blocks: Types & Objects

- ALGORITHM
- STRATEGY
Building Blocks: Types & Objects
Building Blocks: Types & Objects

DATA FEED → ALGORITHM → STRATEGY → BUY / SELL ORDERS
Demo 3: Moving Average

- Start with $100 million
- BUY when price > 50-day moving average
- SELL when price < 120-day moving average

➢ 1 STRATEGY with 2 concurrent processes
➢ 2 ALGORITHMs
➢ 1 FUNCTION
The StratMaster Team

Project Manager: Jintack Lim
Language Guru: Vincent Mierlak
System Architect: Xingying Liu
System Integrator: Moning Zhang
Testing & Validation: Enrui Liao
Translator architecture

Source file: mystrat.sm

Lexer → Tokens

Parser → AST

Symbol table

Syntactic Check

Semantic Check

Symbol table Creation

Interpreter

process handler

algorithm handler

order handler

create thread for each strategy, process, and algorithm

Run time

manage account

manage order queue
Syntactic Structure
The StratMaster Team

Project Manager: Jintack Lim
Language Guru: Vincent Mierlak
System Architect: Xingying Liu
System Integrator: Moning Zhang
Testing & Validation: Enrui Liao
Run Time Environment

Main

<table>
<thead>
<tr>
<th>Strategy 1</th>
<th>WHEN-UNTIL 1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(emit order)</td>
</tr>
<tr>
<td></td>
<td>Call</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Strategy 2</th>
<th>WHEN-UNTIL 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(emit order)</td>
</tr>
<tr>
<td></td>
<td>Signal</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>WHEN-UNTIL 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(emit order)</td>
</tr>
</tbody>
</table>

Algorithm

(order queue)

(order handler)

(put order)
The StratMaster Team

Project Manager: Jintack Lim
Language Guru: Vincent Mierlak
System Architect: Xingying Liu
System Integrator: Moning Zhang
Testing & Validation: Enrui Liao
Semantic Checking

- What do we do?
  - ID conflict
  - Type error
  - id1 + num < id2

- How do we do that?
  - Explicit Stack
  - Type synthesis
  - Postorder traversal
## Test Plan:

- **What do we check**

<table>
<thead>
<tr>
<th>Category</th>
<th>Cases</th>
</tr>
</thead>
<tbody>
<tr>
<td>header: account &amp; datafeed</td>
<td>case 1 - case 15</td>
</tr>
<tr>
<td>block: strategy &amp; algorithm</td>
<td>case 16 - case 33</td>
</tr>
<tr>
<td>set-if statement</td>
<td>case 34 - case 45</td>
</tr>
<tr>
<td>when-until statement</td>
<td>case 46 - case 80</td>
</tr>
<tr>
<td>syntax error</td>
<td>case 8-13, case 20, case 52 - case 55</td>
</tr>
<tr>
<td>semantic error</td>
<td>case 55 - case 66</td>
</tr>
</tbody>
</table>
Test Plan:

- Interactive test model
  - Partial / Full
  - Log the fail case
  - Easy to debug
  - Example

  case 79 : /*<79>two WHEN statment in one strategy
          --------> pass
  case 80 : /*<80>two WHEN with until statment in one strategy
          --------> pass

  /*************summary*************/
  pass:  70 / 80
  fail case 3 : /*<3>Missing account
  fail case 4 : /*<4>Account not found
  fail case 9 : /*<9>Wrong Account
The StratMaster Team

Project Manager: Jintack Lim
Language Guru: Vincent Mierlak
System Architect: Xingying Liu
System Integrator: Moning Zhang
Testing & Validation: Enrui Liao
Project Management Strategy

- Team != sum of its parts
- Fast prototyping
- Micro Management

<table>
<thead>
<tr>
<th>Date</th>
<th>Task</th>
<th>Test file</th>
<th>Status</th>
<th>Person in charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>4/23/2015</td>
<td>multiple orders</td>
<td>misc/progs/orders.sm</td>
<td>Done</td>
<td>Michelle</td>
</tr>
<tr>
<td>4/23/2015</td>
<td>algorithm with empty body</td>
<td>devel/empty_algo.sm</td>
<td>Done</td>
<td>Jintack, Moning</td>
</tr>
</tbody>
</table>
Conclusions

● what you learned
  ○ How to work as a team
  ○ Read and think before you do something

● what you would have done differently
  ○ To have the more detailed realistic the project plan and keep it strictly
The StratMaster Team

Project Manager: Jintack Lim
Language Guru: Vincent Mierlak
System Architect: Xingying Liu
System Integrator: Moning Zhang
Testing & Validation: Enrui Liao

Thank You!