Design and Development of the ICEACE Simulator

Roni Bülent Özel

Reykjavik University
School of Science and Engineering
bulent.ozel@gmail.com
bulent@ru.is **Einar Jon Erlingsson, Marco Raberto, Hlynur Stefansson**

June 22, 2013

WEIHA 2013, Reykjavik, *n*Iceland

ICEACE Project

- Home: `http://iceace.github.io/home`
- Matlab: `http://iceace.github.io/MATLAB`
- FLAME: `http://iceace.github.io/FLAME`

ICEACE Model

- Agent Types:
  - Household
  - Firm
  - Bank
  - Equity Fund
  - Central Bank
  - Government
- Markets:
  - Labour Market
  - Production Markets (Consumption Goods, Housing Units)
  - Consumption Goods Market
  - Housing Market
  - Credit Market
- Communication Schemes:
  - Direct Messaging
  - Balance Sheet Flows
  - Agent-Agent Links

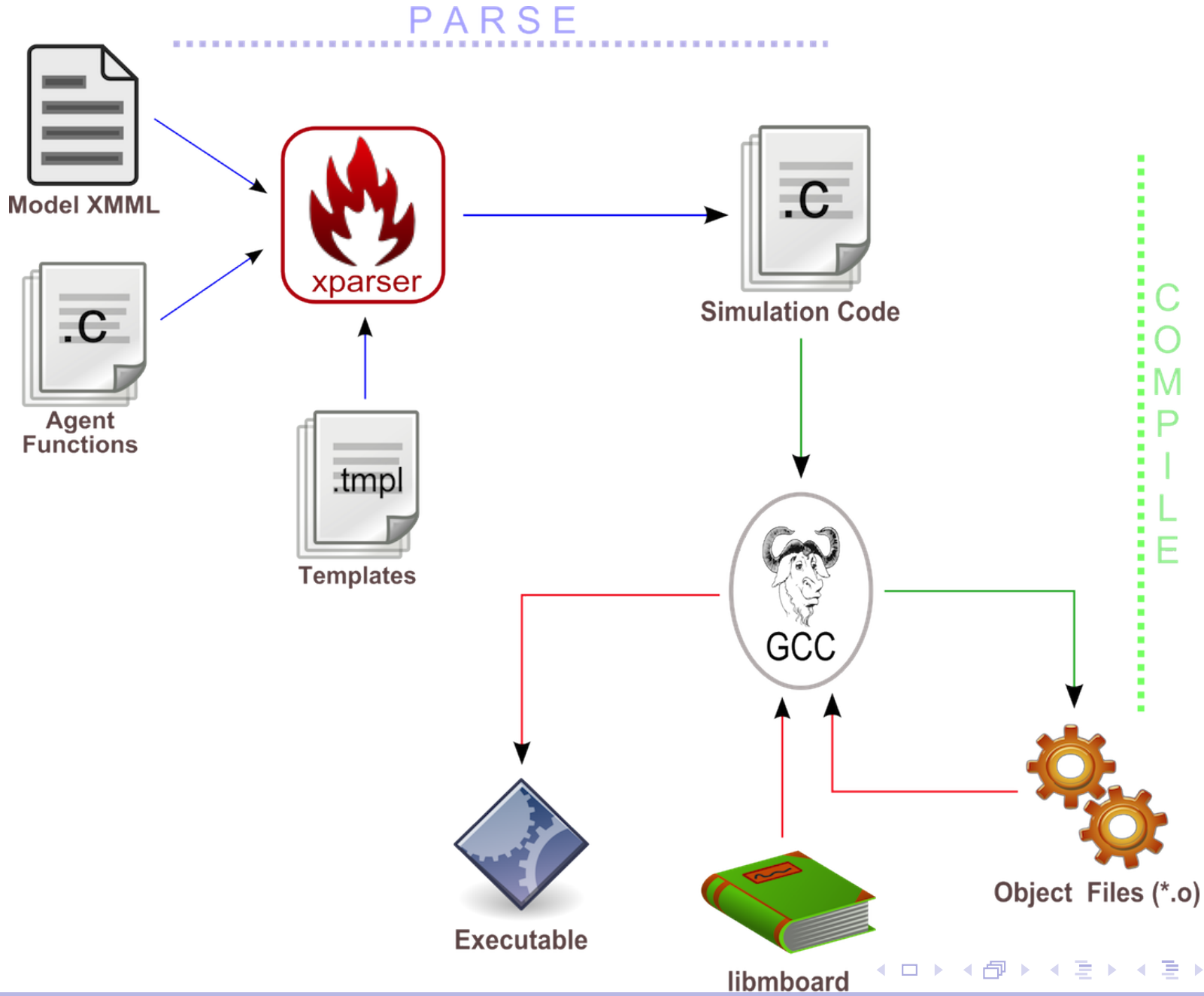
Multi-agent Design Challanges

- Agents
  - Role Multiplicity
  - Beliefs, Desires, Intentions
  - Autonomity
- Environment
  - Context
  - Influence
- Communication
- Scalability
- Initialization

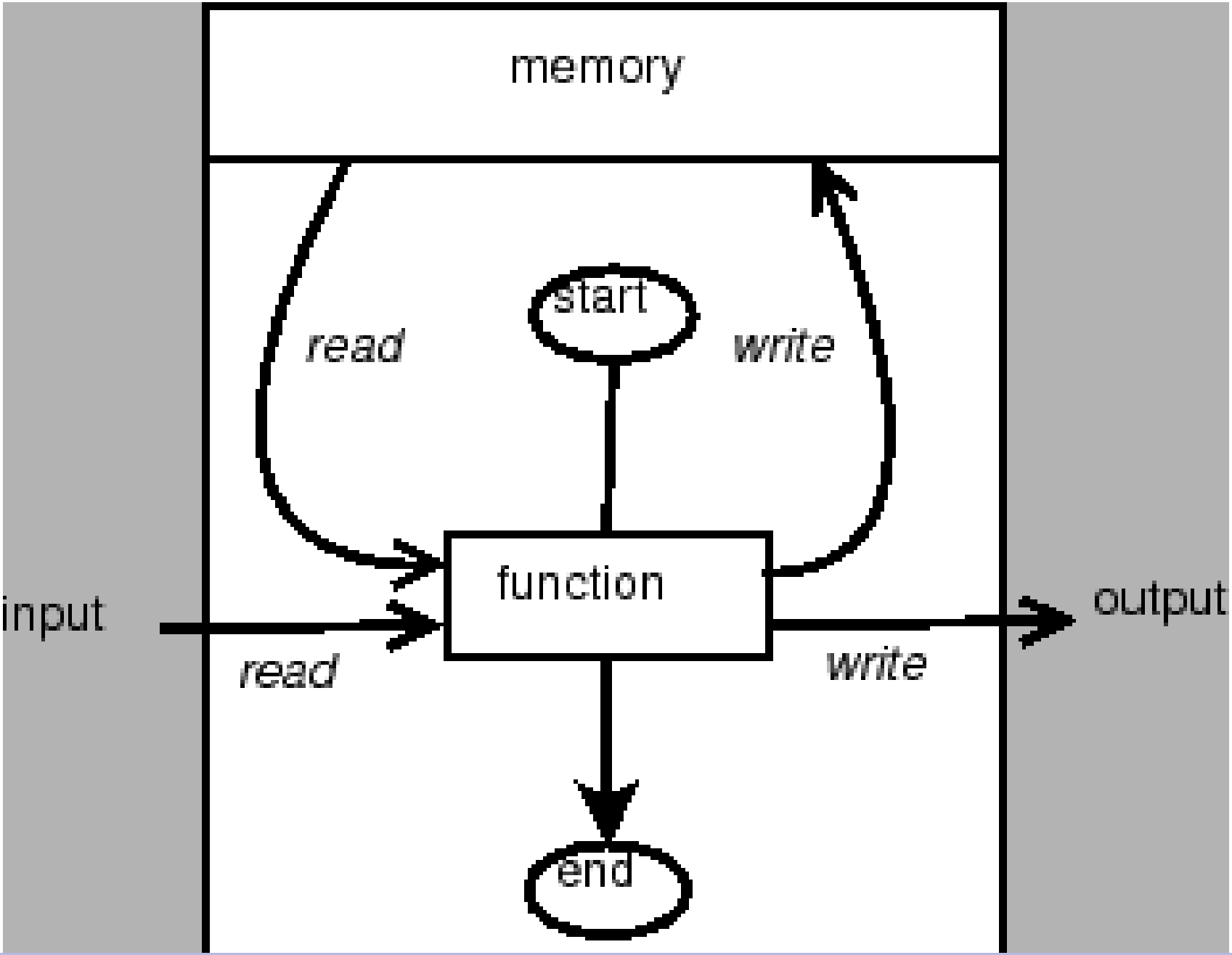
ICEACE Implementation Choices (FLAME)

- Distributed Computing
  - –> XMachine
- Object Oriented Programming Paradigm
  - –> XMachine Markup Language (XMML)
- Message Passing
  - –> Message Boards (Broadcasting)
  - –> Message Filtering (Links)
- Synchronization
  - –> Time Units: Day (1), Week ($5xD$), Month ($4xW$), Quarter ($3xM$), Year ($12xM$)
- Acyclic Dependencies
  - –> Exclusive State Transitions
- High Performance Computing
  - –> MPI Protocal
- Initialization
  - –> Pythonic Agent Initialization Description Language (PyAIDL)

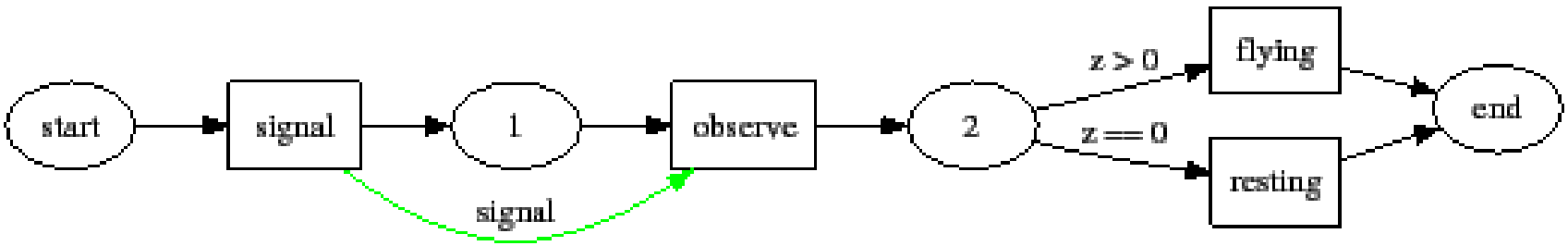
FLAME Multi-agent Design Frame Framework



XMachine - A Single Design Unit



State Transitions



ICEACE Model v0.2

Overall state transition and communication graph of ICEACE model:

`http://iceace.github.io/FLAME/doxy/stategraph_colour.pdf`

Conceptual Design Vs Implementation

- Pseudo Agents
  - Real Estate Agency
  - Job Placement Office
  - Mall
  - *Census Bureau*
- Agent Subtypes
  - Households: Capitalist, Non-capitalist
  - Firm: Constructor, Regular
- Mortgage Durations & Annuity

ICEACE Reference Manual

`http://iceace.github.io/FLAME/doxy/summary.pdf`

- State Variables (memory)
- Functions (behaviours)
- Messages (communication)

Modular and Iterative Design

Model Descriptions:

`https://github.com/ICEACE/FLAME/blob/master/model_v0.1.0.xml`

Labour Market

- Monthly
- Market opens first day of the month
- Payments are done at last day of the month
- Market closes either when all positions are filled or all households are employed.
- Employment turnover is possible
- Skilled households are given priority
- Firing, new hiring, and *wage adjustment* is possible

`http://iceace.github.io/FLAME/doxy/stategraph_colour.pdf`

Production Market

- Monthly
- Regular products are produced monthly
- A housing unit is completed in 12 months
- Production function
- Pricing
- Production planning
- Labour requirements

`http://iceace.github.io/FLAME/doxy/stategraph_colour.pdf`

Consumption Market

- Weekly
- Limited yet monthly adjustable disposable consumption budget
- Unspent budget maybe used in subsequent weeks
- Wealth effect as a mean of shock transmission mechanism from housing markets
- Arrival to mall is random
- Cheaper products have a higher probability to be consumed first

`http://iceace.github.io/FLAME/doxy/stategraph_colour.pdf`

Housing Market

- Monthly
- Housing units or homogenous
- Constructor firms, buyers, sellers
- Fire sale cases
- Pricing
- Mortgage requirements
- Annuity

`http://iceace.github.io/FLAME/doxy/stategraph_colour.pdf`

Credit Market

- Monthly
- Loans
- Mortgage annuity adjustment
- Equity Fund
- Illiquidity
- Insolvency

`http://iceace.github.io/FLAME/doxy/stategraph_colour.pdf`

Policy Making

- Quarterly, monthly, weekly
- Interest rates
- Tax rates and taxing
- Inflation, unemployment
- General benefits, unempoylemt benefits

`http://iceace.github.io/FLAME/doxy/stategraph_colour.pdf`

Computational Challanges

- Initialization
- Load Balancing
- Time Performance, worst case: $O(|AgentCount|)$
- Memory Management

ICEACE Iterative Design Process

- **Theoretical Design**
- **Prototyping**
- Iterative Multi Agent Design Cycle:
  - Model Description (XMML):
    - Memory
    - Action Description
    - State Transitions
    - Activation Conditions
    - Inputs: (filtering, sorting, randomizing)
    - Outputs
  - Behaviors (C Functions)
  - Unit Testing
  - Modular Verificatation
- Initialization (via PyAIDL):
  - Setting policy parameters
  - Instantiating agents
  - Initializing agent memories
- *Validation Experiments*

Validation

- Calibration
- Randomness
- Paramater sensivity
- Empirical Tests

Serial Run Time

- Households: 8000, Firms: 125(regular) + 25(constructor), Banks:2, Central Bank, Government, Job Placement Office, Real Estate Agency, Mall
- Dual Core MacPro OS 10.8.4, CPU 2.26 GHz, RAM 4G 1067MHz
- Debug Mode
- 50 runs
- Avg clock time $\approx 7min$