InnovateFPGA
2018
Grand Final

EM083

BondMachine

Reconfigurable Computing

Mirko Mariotti,
Loriano Storchi,
Daniele Spiga

Dept of Physics and Geology – University of Perugia
INFN Perugia

Headline Sponsor
Organizer
University Progam Sponsor
Sponsors
Distribution Partner
University Progam Sponsor

InnovateFPGA 2018 Grand Final

BondMachine (BM)

EM083

Objectives

- Create a computational system where hardware and software are **co-designed**
- Guarantee the **full exploitation** of the FPGA hardware capabilities
- Provide a **user friendly** abstraction

- A **dynamic computer architecture: adapting** to specific computational problems

Strategy

InnovateFPGA 2018 Grand Final

BM Computer Architecture Components

EM083

Connecting Processor (CP)

The atomic computing core

- **Simple "register machines"**
- **Interconnected via special IO registers and opcodes**
- **Local ROM and RAM**
- **Specialized with many customization possibilities**
- **Building a Heterogeneous architecture with Many cores in a single BM**

Shared Object (SO)

Non-computing objects

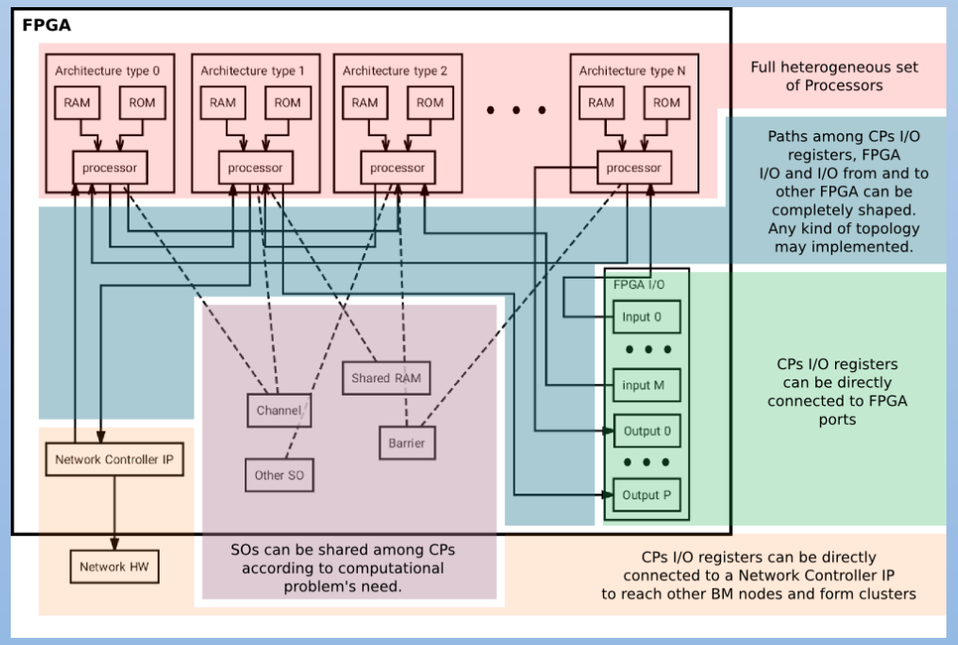
- **Objects like: memories, channels, barriers, pseudo random generators. Etc..**
- **Shared possibly among CPs**
- **Used by CPs via extra instructions to allow synchronization, storage, communication**

InnovateFPGA 2018 Grand Final

EM083

The Ecosystem

BM computer architecture



Handling tools

FPGA

The Bondgo compiler

API for specific problems

Network Connector

InnovateFPGA 2018 Grand Final

Handling the BM computer architecture

EM083

The BM computer architecture is managed by a set of tools to:

- Build a specific architecture
- Modify a pre-existing architecture
- Simulate or Emulate the behavior
- Generate the Register Transfer Level (RTL) code

Processor Builder (procbuilder)

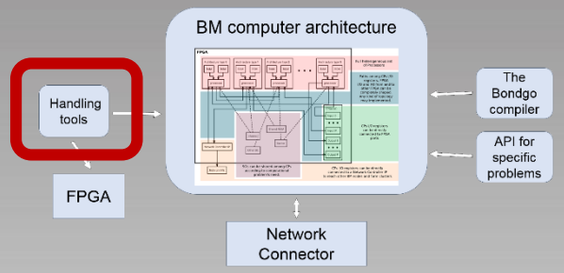
Selects the single processor, assemble and disassemble, saves on disk as JSON, creates the RTL code of a CP.

BondMachine Builder (bondmachine)

Connects CPs and SOs together in custom topologies, loads and saves on disk as JSON, create BM's RTL code.

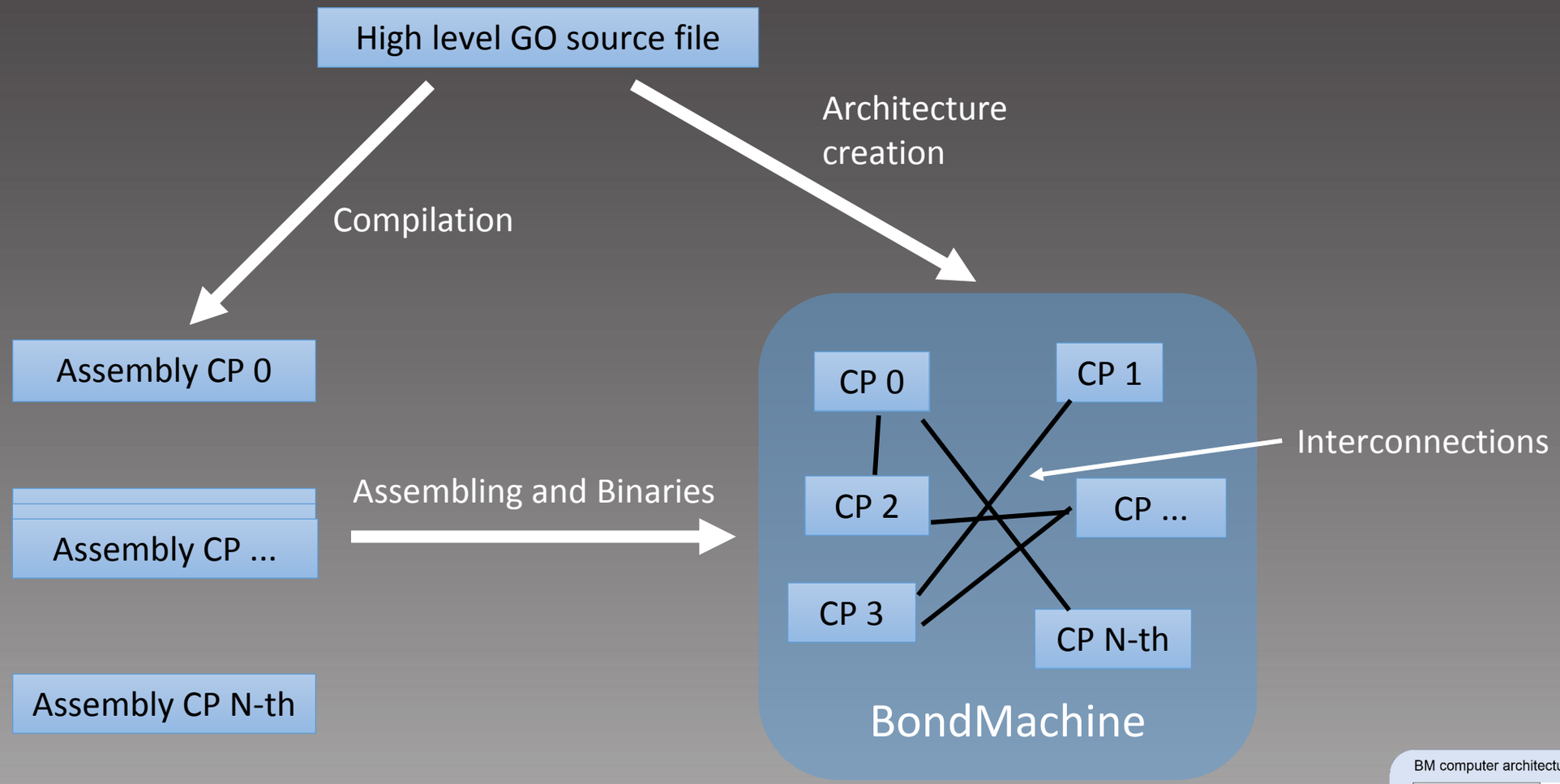
Simulation Framework (simbox)

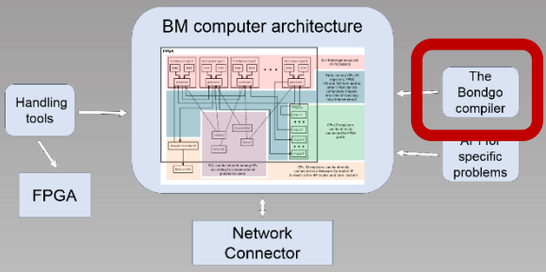
Simulates the behavior, emulates a BM on a standard Linux workstation.



EM083

Bondgo: The BM compiler





InnovateFPGA 2018 Grand Final

Ecosystem API

EM083

Mapping specific computational problems to BondMachines

Symbond

Map symbolic mathematical expressions to BM

Boolbond

Map boolean systems to BM

Matrixwork

Basic matrix computation

Neuralbond

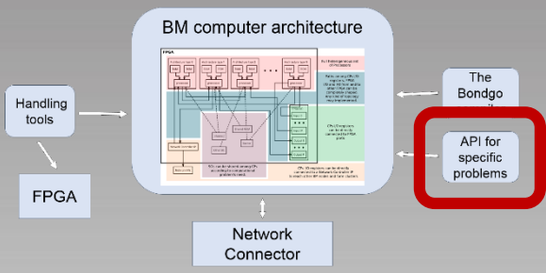
Map neural networks to BM

Evolutive BM

Evolutionary computing to BMs

tf2bm

Mapping TensorFlow graphs to BMs

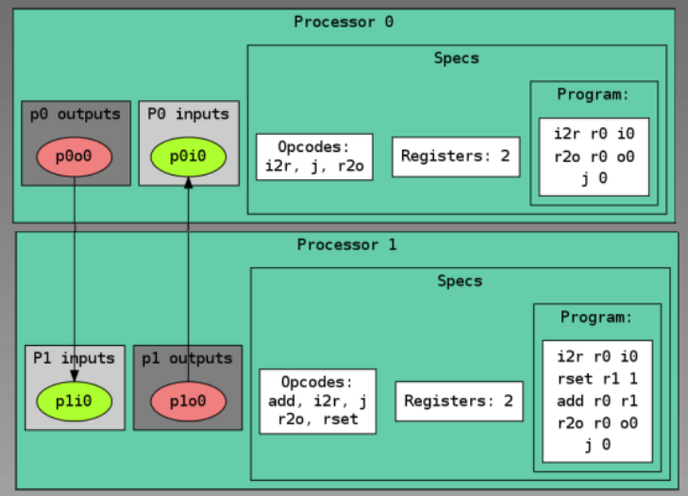


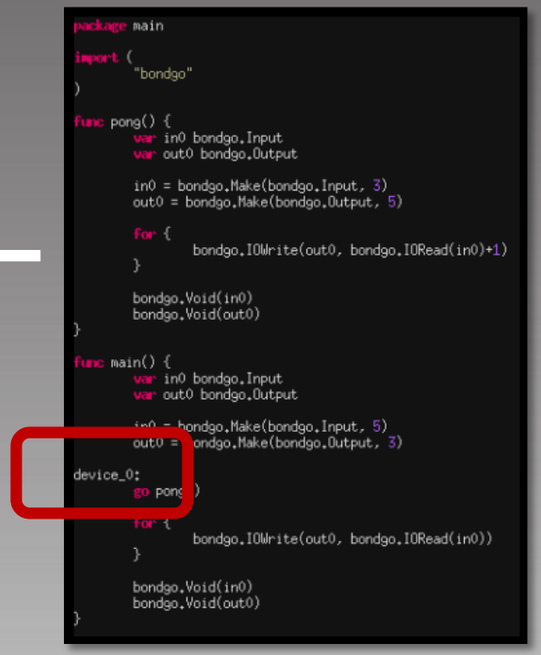
InnovateFPGA 2018 Grand Final

Clustering BondMachines

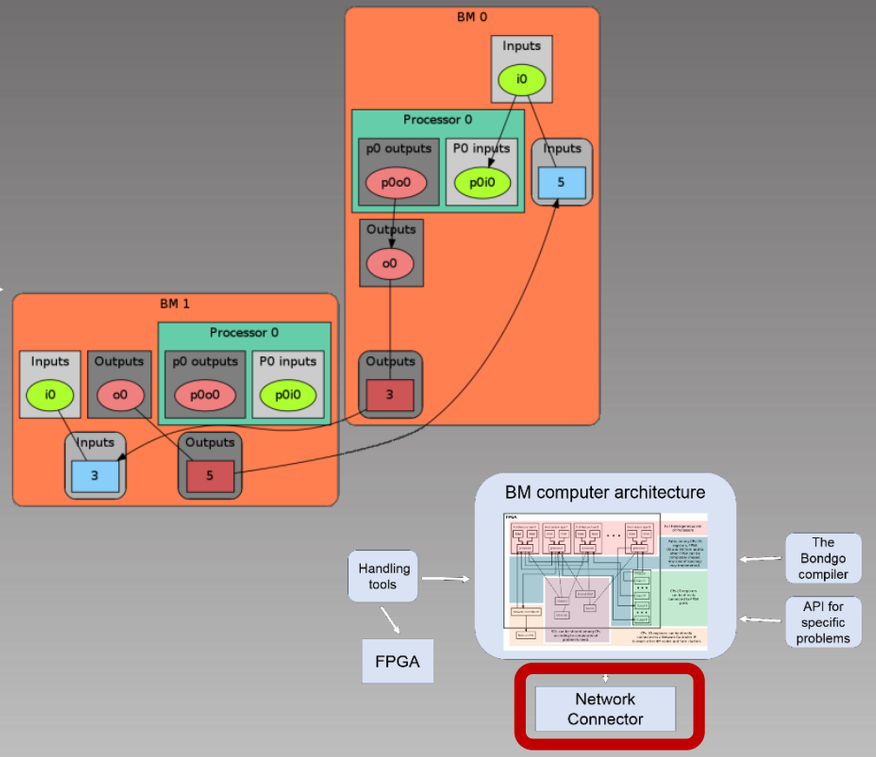
EM083

- The same logic existing among CPs have been extended to different BMs organized in clusters.
- Custom protocols have been created for this purpose.
- FPGA based BMs, standard Linux Workstations and emulated BMs may join a cluster and contribute to a single distributed computational problem.





```
package main

import (
        "bondgo"
)

func pong() {
        var in0 bondgo.Input
        var out0 bondgo.Output

        in0 = bondgo.Make(bondgo.Input, 3)
        out0 = bondgo.Make(bondgo.Output, 5)

        for {
                bondgo.IOWrite(out0, bondgo.IORead(in0)+1)
        }

        bondgo.Void(in0)
        bondgo.Void(out0)
}

func main() {
        var in0 bondgo.Input
        var out0 bondgo.Output

        in0 = bondgo.Make(bondgo.Input, 5)
        out0 = bondgo.Make(bondgo.Output, 3)

device_0:
        go pong()

        for {
                bondgo.IOWrite(out0, bondgo.IORead(in0))
        }

        bondgo.Void(in0)
        bondgo.Void(out0)
}
```



InnovateFPGA 2018 Grand Final

EM083

Conclusion and Future work

- The result of this project is the construction of a computer architecture that is not anymore a static constraint where computing occurs but its creation becomes a part of the computing process, gaining computing power and flexibility
- Over this abstraction is it possible to create a full computing Ecosystem
- Keeping the register machine abstraction it is possible to borrow well known languages and techniques to keep the programming simple

Future work:

- New instructions and SO
- Support for new interconnection devices

Uses in Physics experiments:

- Real time pulse shape analysis in neutron detectors
- Space experiments test beams