Galapagos:

a generic distributed parallel
genetic algorithm development platform

Nicolas Kruchten

4th year Engineering Science

(Infrastructure Option)

Roadmap

- What is Galapagos?

- EAs, GAs & PGAs

- Distributed Computing

- Distributed GAs & PGAs

- Case study: dGAID

G a l a p a g o s

What is Galapagos?

• a software framework for building distributed parallel genetic algorithms

• defines and steps through a basic flowchart with 'black-box' steps

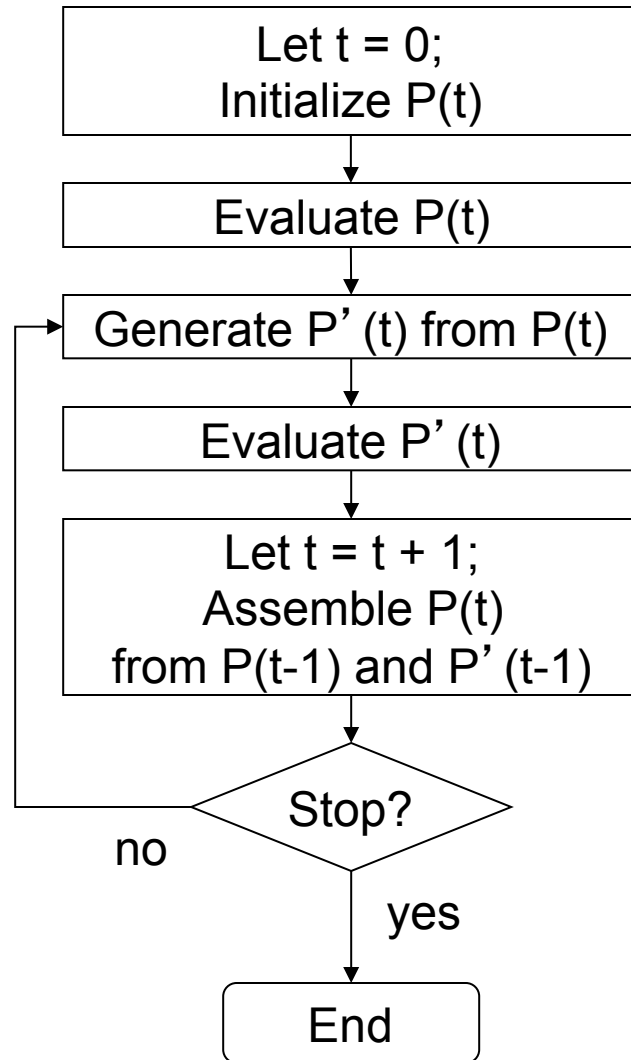• provides basic black boxes (modules), templates for the creation of new modules

Genetic Algorithms

- Many ITS problems are or can be formulated as optimization problems

- Traditional methods often don't suffice

- Genetic Algorithms are powerful but can require a lot of intuition or original coding

Evolutionary Algorithms

- EA = GA + EP + ES

- Evolutionary Programming
  - GA minus the G (single parents)
  - not same as 'genetic programming'

- Evolutionary Strategies
 - very similar to GA with adaptive mutation

Galapagos

```
┌─────────────────────┐
│      Let t = 0;      │
│    Initialize P(t)   │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│     Evaluate P(t)    │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│ Generate P'(t) from P(t) │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│     Evaluate P'(t)   │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│    Let t = t + 1;    │
│    Assemble P(t)     │
│ from P(t-1) and P'(t-1) │
└─────────────────────┘
           │
           ▼
        ◇ Stop? ◇
     no           yes
                   │
                   ▼
              ( End )
```

Let t = 0;
Initialize P(t)

Evaluate P(t)

Generate P'(t) from P(t)

Evaluate P'(t)

Let t = t + 1;
Assemble P(t)
from P(t-1) and P'(t-1)

Stop?

no

yes

End

Parallel Genetic Algorithms

- 'regular' GA aka 'panmictic' GA

- PGAs have multiple interacting 'demes' or sub-populations (also: islands)

- Interaction occurs through 'migration'

- Can converge faster, to better optima

G a l a p a g o s

Parallel Genetic Algorithms

- Many different topologies are possible:
    - a few large demes
    - many small demes

- Other possibilities:
    - synchronous vs asynchronous
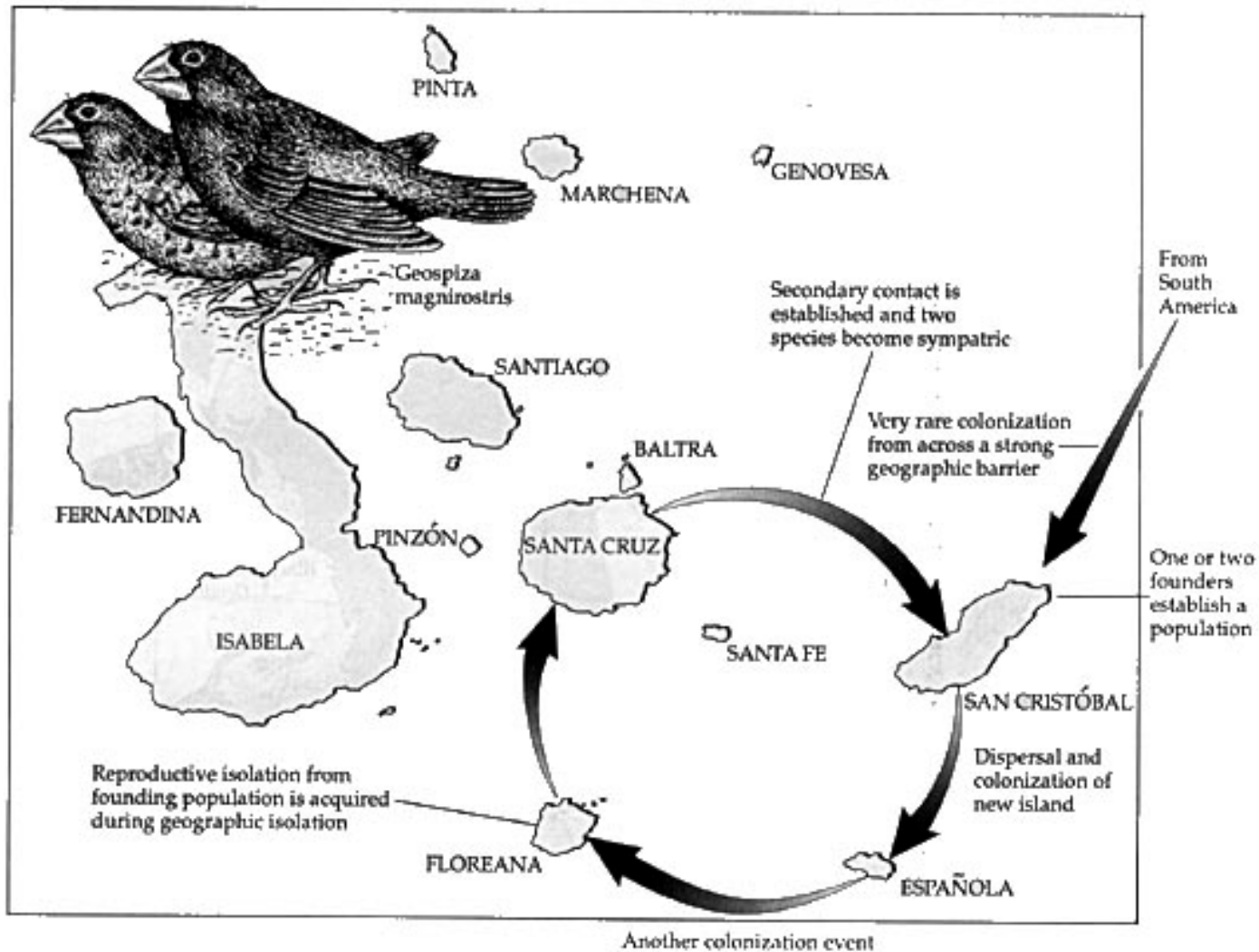   - heterogeneous PGAs
    - n-level PGAs
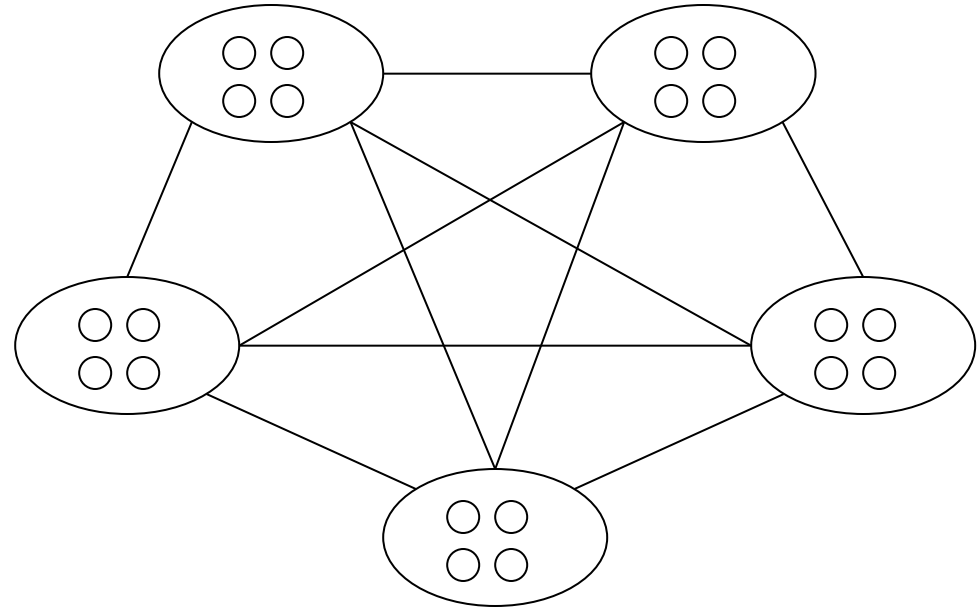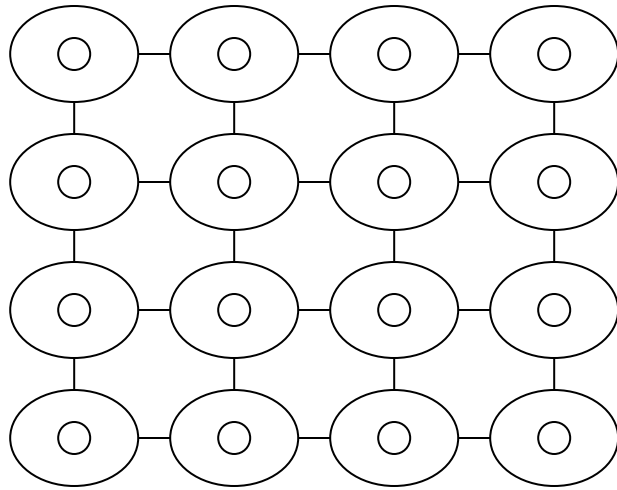
Galapagos
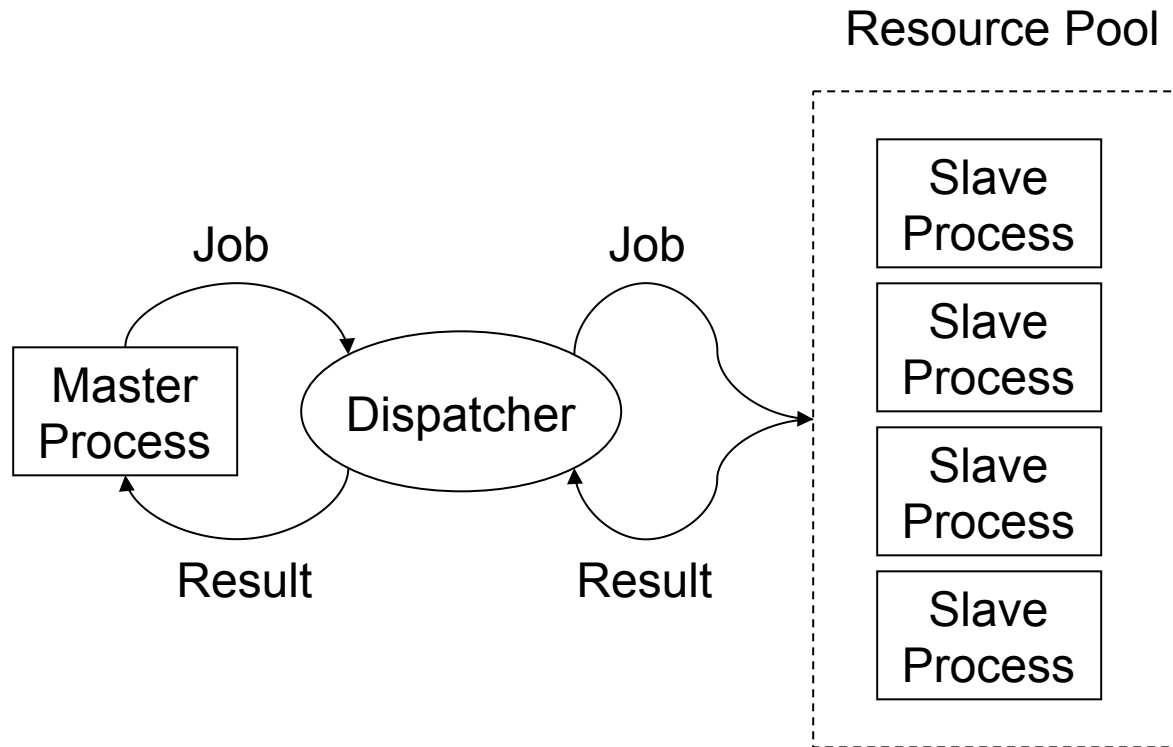
Migration Path   ○ 5 Genomes   Deme

Galapagos

Distributed Computing

- GAs and PGAs can be very slow

- We can use a faster computer

- Or we can use more than one computer
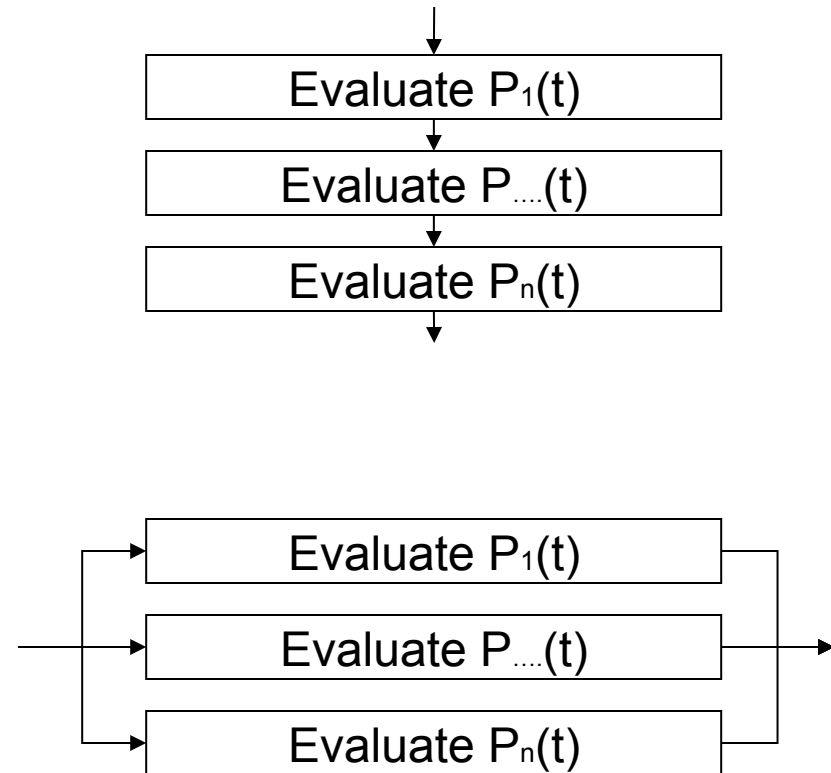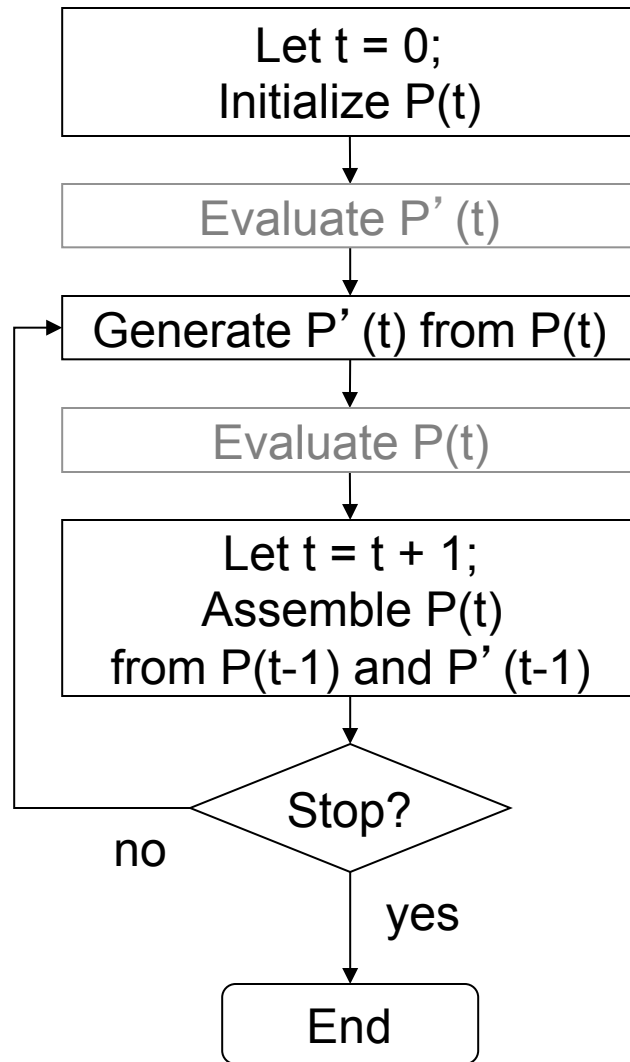 - Multi-processor machines
  - Clusters
  - Grid computing
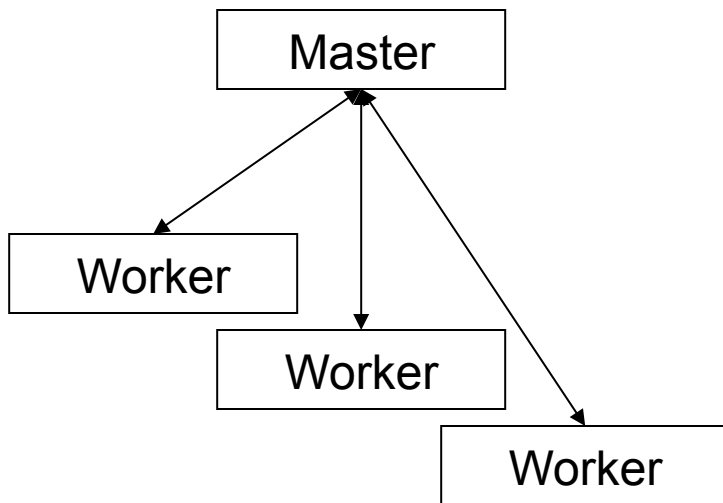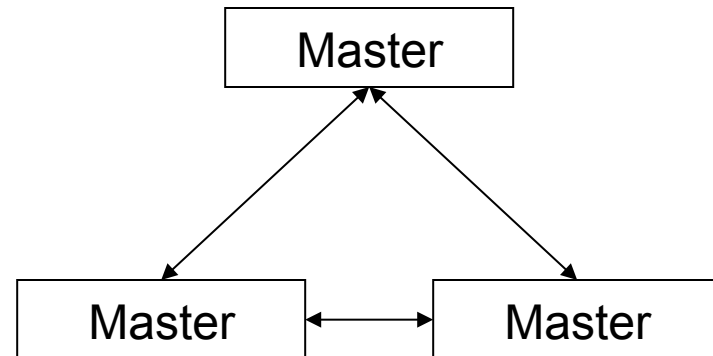
Galapagos: a distributed PGA platform

Resource Pool

Master Process → Job → Dispatcher → Job → Slave Process

Slave Process ← Result ← Dispatcher ← Result ← Master Process

Slave Process

Slave Process

Slave Process

Slave Process

Distributed GAs & PGAs

- 'Master/Worker' architecture:
  - GAs are 'embarassingly parallel'

- 'Peered' architecture:
  - PGAs are uniquely suited to this sort of distribution

- 'Hybrid' architecture:
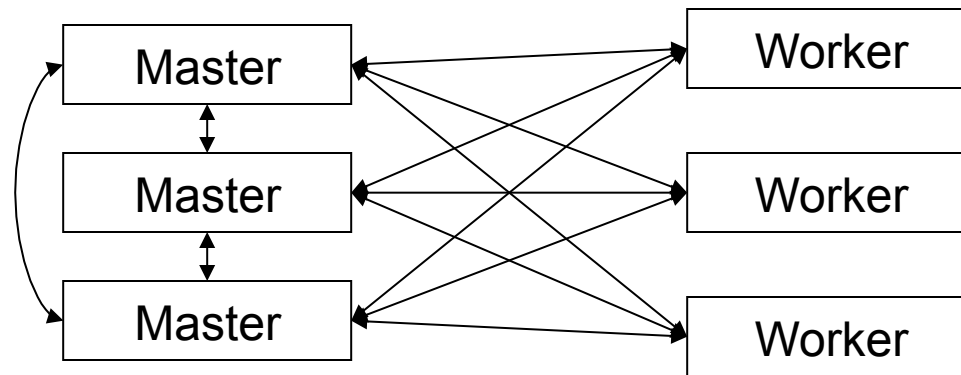 - Peered Masters with a Worker pool

Galapagos

```
┌─────────────────────┐
│     Let t = 0;      │
│   Initialize P(t)   │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│   Evaluate P' (t)   │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│ Generate P' (t) from P(t) │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│    Evaluate P(t)    │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│   Let t = t + 1;    │
│    Assemble P(t)    │
│ from P(t-1) and P' (t-1) │
└─────────────────────┘
           │
           ▼
        ◇ Stop? ◇
   no            yes
           │
           ▼
        ( End )
```

$$\text{Evaluate } P_1(t)$$
$$\text{Evaluate } P_{\ldots}(t)$$
$$\text{Evaluate } P_n(t)$$

$$\text{Evaluate } P_1(t)$$
$$\text{Evaluate } P_{\ldots}(t)$$
$$\text{Evaluate } P_n(t)$$

Galapagos

Master / Worker

Master

Worker

Worker

Worker

Peered

Master

Master

Master

Galapagos

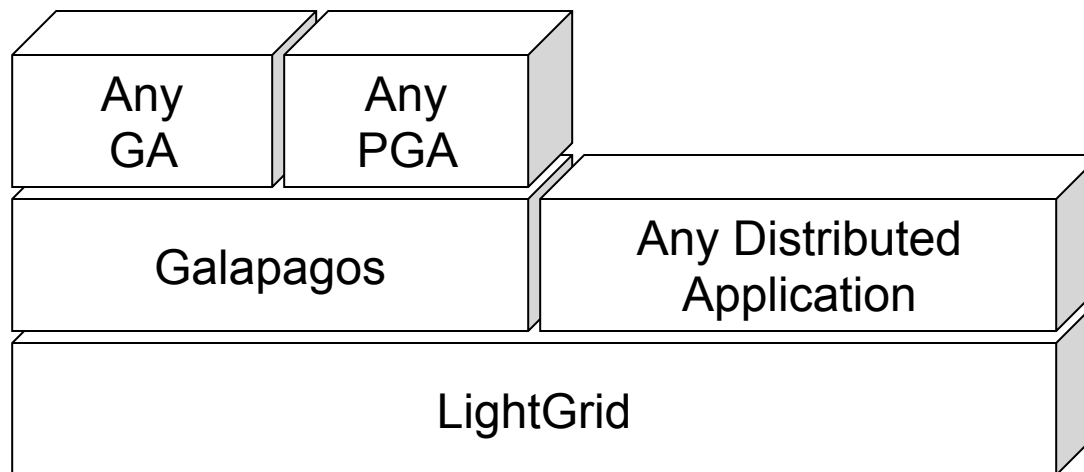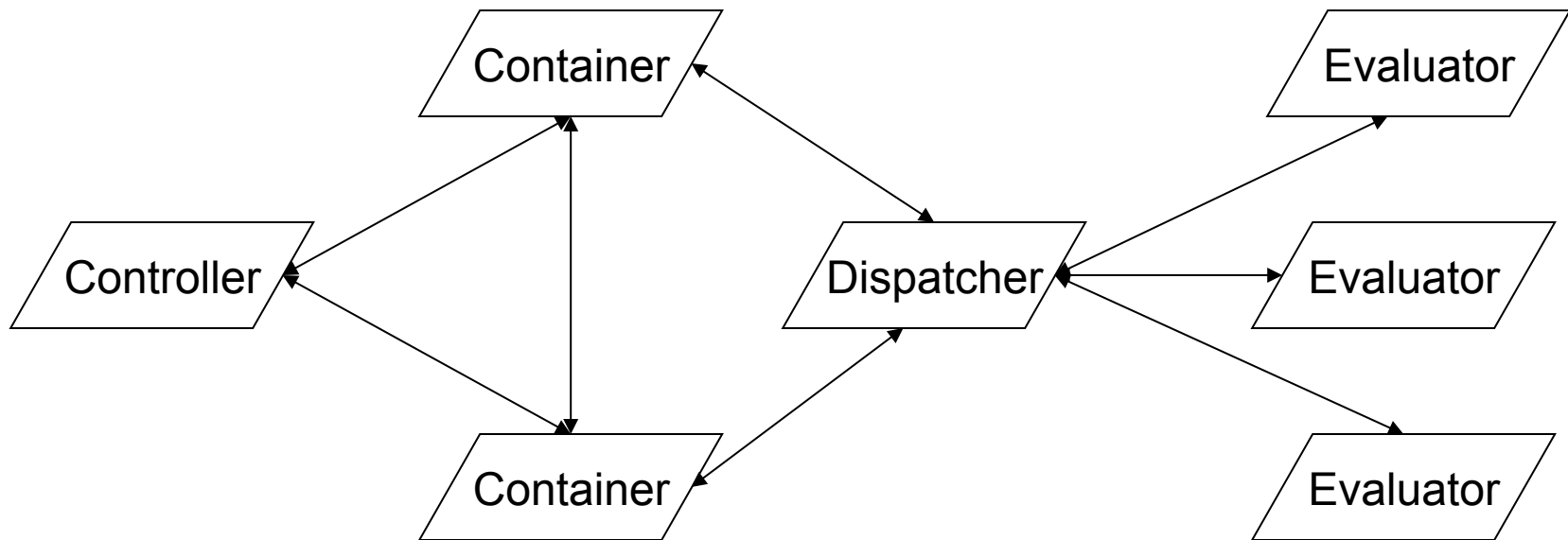Master ↔ Worker
Master ↔ Worker
Master ↔ Worker

Hybrid: Peered Masters with Worker Pool

Galapagos and LightGrid

- LightGrid: lightweight grid engine

- Galapagos: generic EA/GA/PGA platform

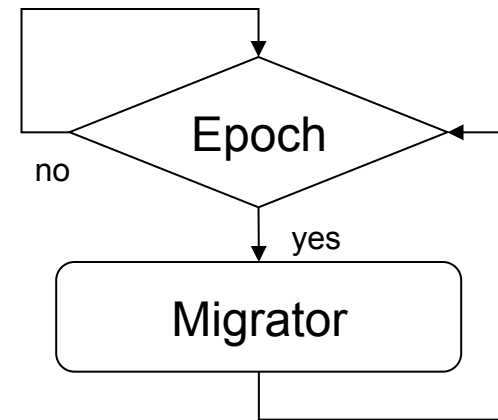- together: very flexible and powerful base onto which a variety of applications can be built!
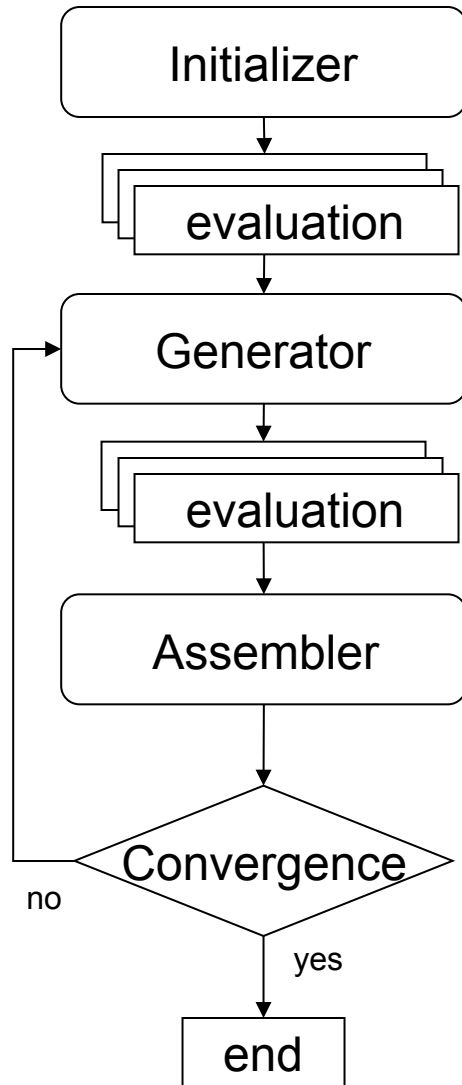
Galapagos

Any
GA

Any
PGA

Galapagos

Any Distributed
Application

LightGrid

Galapagos is Generic

• built with the freedom of the developer
in mind at all times:

- • not representation-specific
- • not problem-specific
- • not GA-specific
- • not platform-specific (written in Java)

Galapagos

Initializer

evaluation

Generator

evaluation

Assembler

Convergence

no

yes

end

Epoch

no

yes

Migrator

Galapagos

a generic distributed parallel
genetic algorithm development platform

Case Study: dGAID

- Genetic Adaptive Incident Detection

- Developed here at U of T

- Calibrating an artificial neural network classifier using a GA

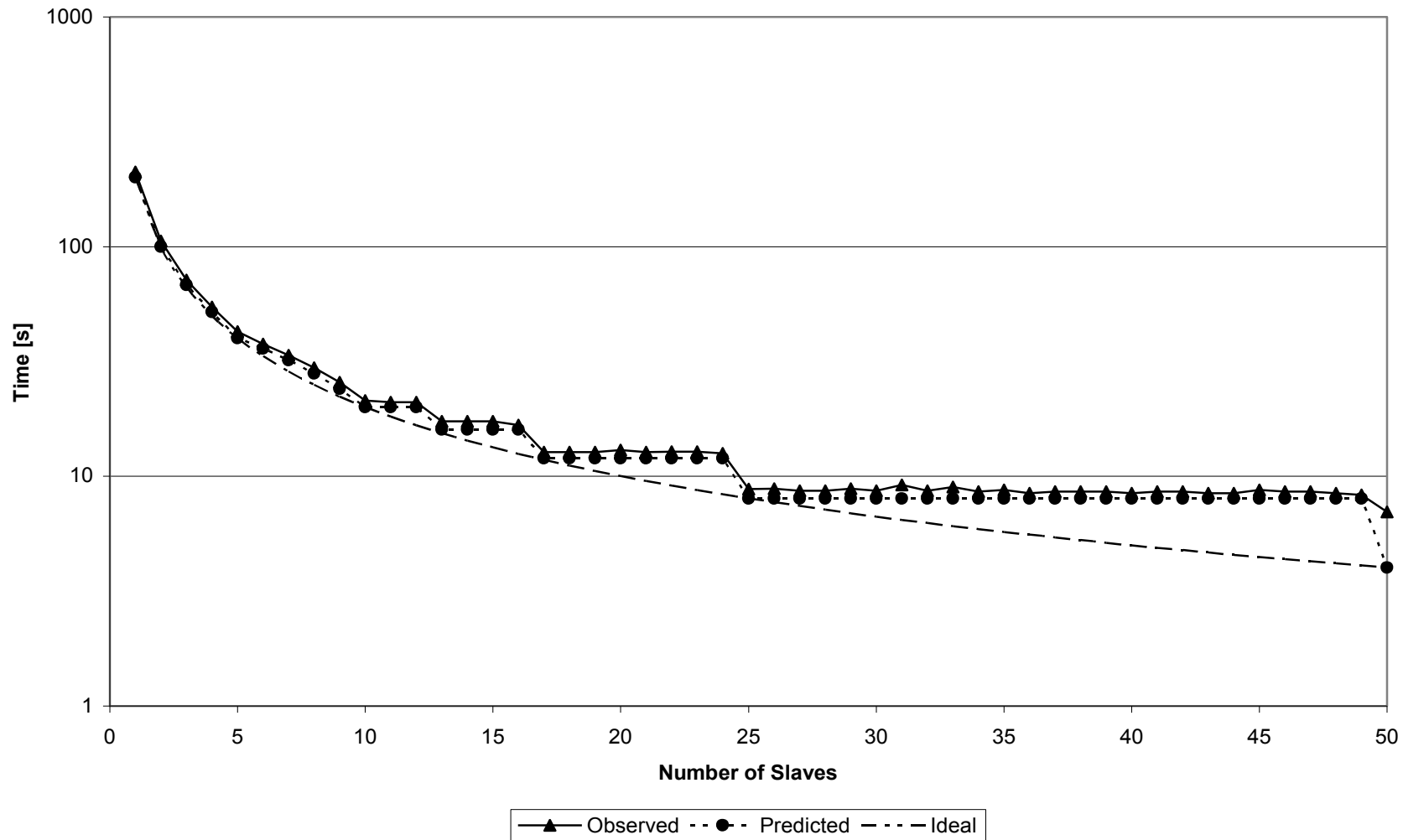- 16-dimensional maximization, non-differentiable

G a l a p a g o s

Case Study: dGAID

- Tests run with 1-50 computers,
  1 population of 50 solutions

- 1 computer: ~ 1 h 15 min

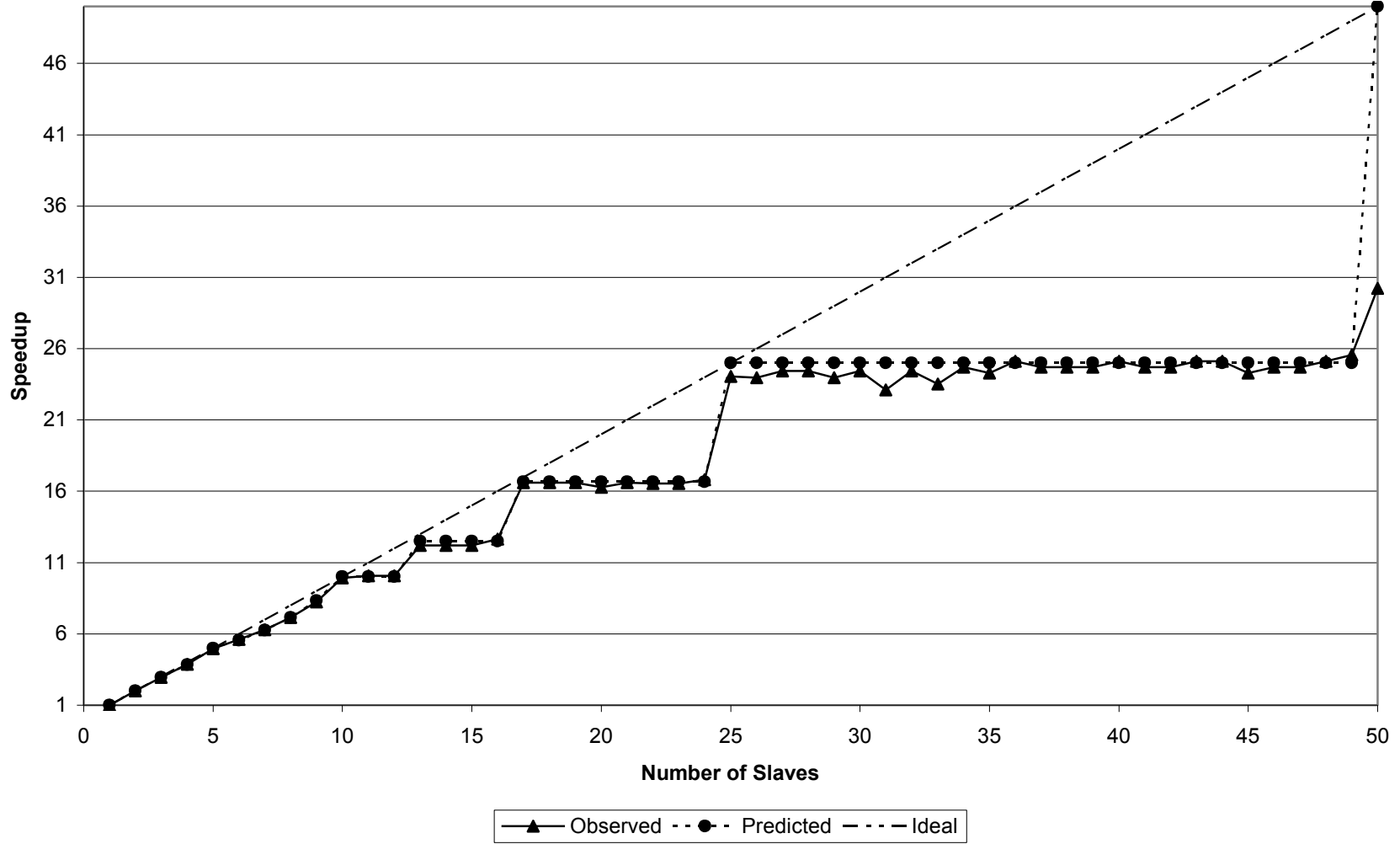- 25 computers: ~ 3.2 min

- 50 computers: ~ 2.3 min
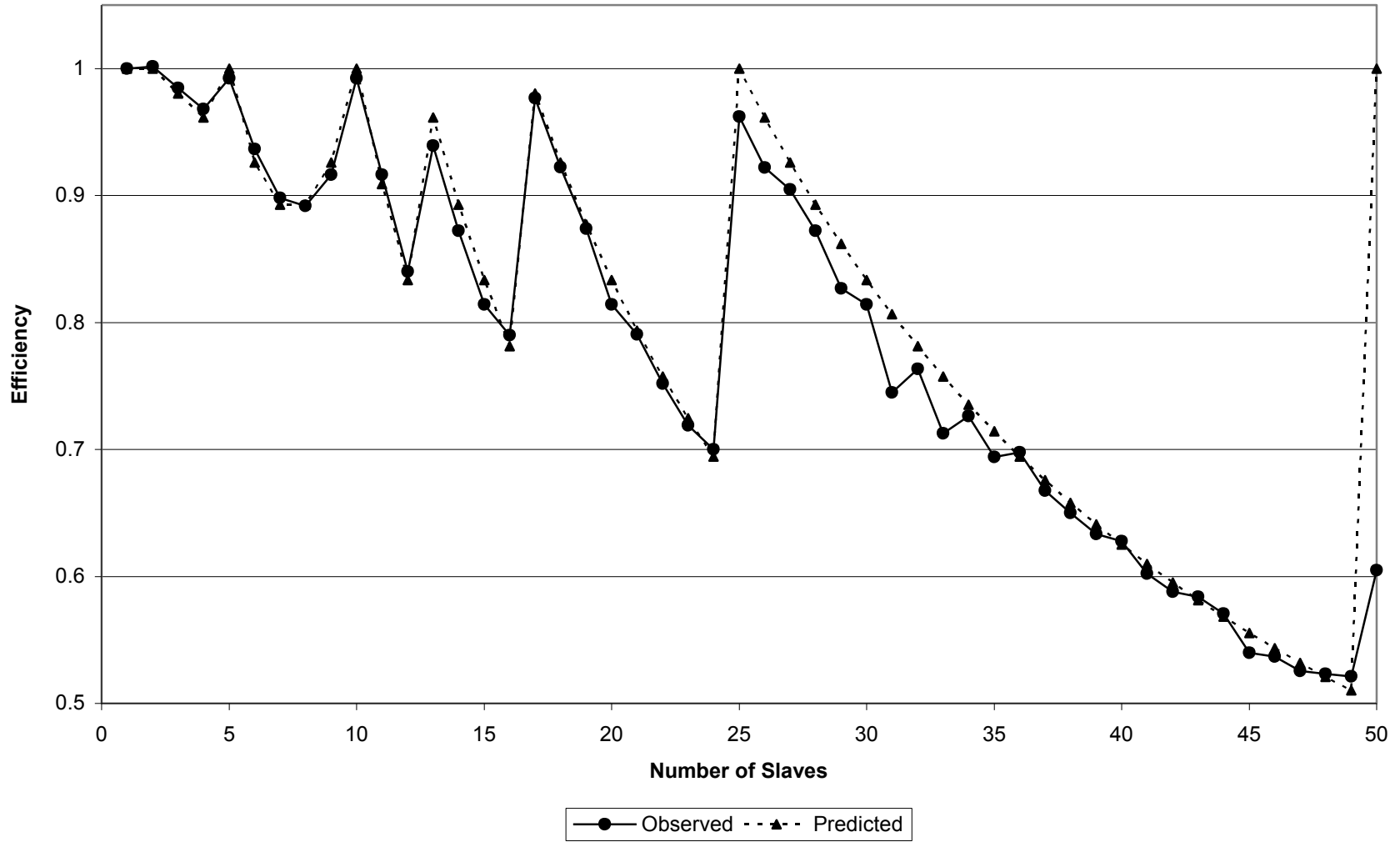
Generation Evaluation Time vs Slaves

Galapagos

Time [s]

Number of Slaves

Observed — Predicted — Ideal

Speedup vs Slaves

Efficiency vs Slaves

Galapagos

Conclusions

- Galapagos brings together GAs, PGAs and distributed computing into one package

- Its use in speeding up solution of ITS problems has been demonstrated

- Come back next week for a more technical look at how Galapagos works and how to use it

*G*alapagos

Questions?

nicolas@kruchten.com

this presentation is available at:
http://nicolas.kruchten.com/

Galapagos:

a generic distributed parallel
genetic algorithm development platform

Nicolas Kruchten

4th year Engineering Science

(Infrastructure Option)