Tangle: A General Purpose, Concurrent, Object Oriented, Actor Based Programming Language

Chris Wailes and Graham Price

Buzzword Soup

- General Purpose
  - Not designed for a specific application domain or machine architecture
- Concurrent and Object Oriented
  - Every object is viewed as being executed concurrently
- Actor Based
  - Message passing acts as serialization point
Why a New Language?

- Concurrency is a new concern for the average programmer
  - Fortran, C, Java
- Fighting to add concurrency to a serial paradigm is a loosing battle
- Concurrency is HARD
- Execution environment
  - How many cores do you write your code for?

Current Concurrent Languages Issues

- Erlang (1986)
  - Actor model
  - Functional language tradition
- Fortress (2008)
- Chapel (2010)
- Clik (2009)
- Orc (2009)
- Titanium (2005)
- Kilim (2008)
  - Extended Java
- x10 (2004)
- ZPL (2005)
- HPF(1993)
Current Concurrent Languages Issues

- Erlang (1986)
  - Actor model
  - Functional language tradition
- Fortress (2008)
- Chapel (2010)
- Clik (2009)
- Orc (2009)
- Titanium (2005)
- Kilim (2008)
  - Extended Java
- x10 (2004)
- ZPL (2005)
- HPF (1993)

1. Annotations (Death)
2. Concurrency Constructs
3. Limited Parallelism

The Tangle Architecture

Tangle Source → Lex & Parse → Type Analysis → LLVM

Tangle To LLVM Bytecode
The Tangle Architecture

Runtime to LLVM Bytecode

The Tangle Architecture

Clang + Response Closure Linker Pass -> Native Binary
Concurrency in Tangle

- Parallelism extracted by the runtime environment
  - Asynchronous message sends
  - Futures
  - Hints can be given to the runtime
- Runtime is smart (or it will be)
  - Load balancing
  - Data locality
  - Can identify *hot* objects

The Tangle Runtime Environment
The Tangle Runtime Environment
Tangle Example

class DSApp
  state
    :Peer NextPeer
  end

  def skip
    NewPeer = NextPeer get next; get next
    print (NewPeer get name)
    transition NextPeer -> NewPeer
  end
end

The Only Serial Tangle Program

class Serial
  state
  end
  def main
    1 + 1
  end
end
Future Work

- Embedded Objects
- Tail Recursion Optimization
- Object Clustering
- Object Load Balancer
- Compile Time Pattern Matching
- Standard Library
- New Type System
- Interfaces

Conclusion

- Tangle is a new parallel language
  - Implicit Parallelism
  - Runtime Management System
  - LLVM Compiler Infrastructure
- Get Involved
  - Code hosted at www.launchpad.net/tangle
  - Make your own runtime
    - Windows
    - Mac
    - Cell