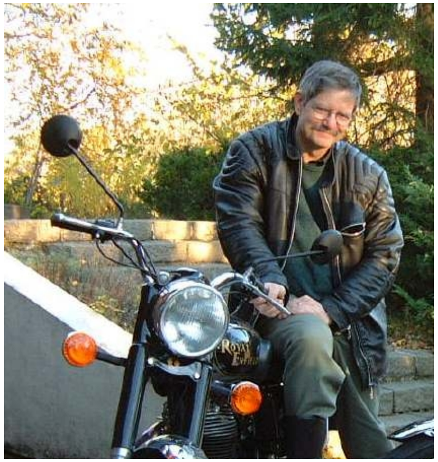
Practical Look at Erlang

Concurrent, Fault Tolerant and Distributed Programming Language

Tehran Linux User Group
Hamidreza Soleimani
February 18, 2014

History & Philosophy

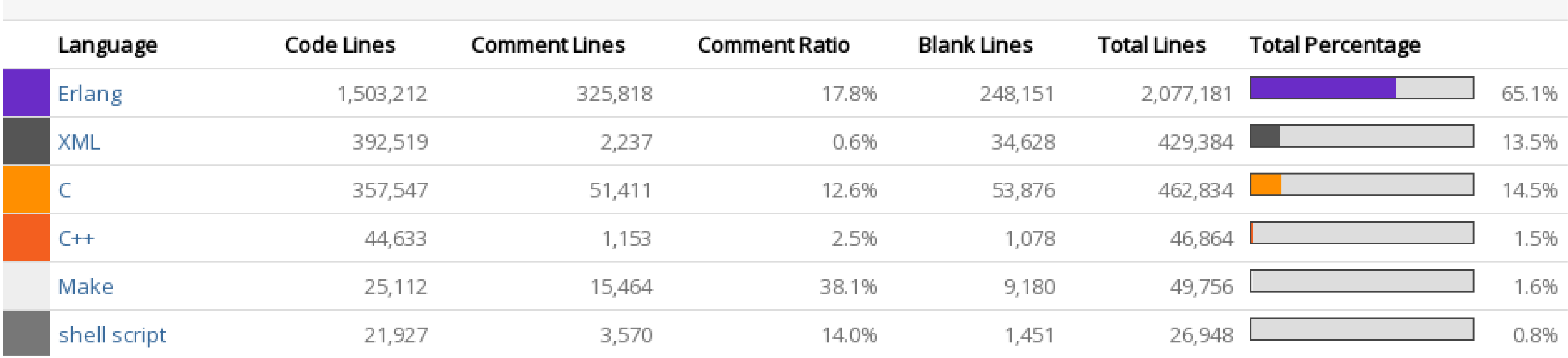
Functional
Runtime System
Garbage Collected
Strong Typing



ERLANG

ERICSSON

Erlang Birthday: 1986



Language	Code Lines	Comment Lines	Comment Ratio	Blank Lines	Total Lines	Total Percentage
Erlang	1,503,212	325,818	17.8%	248,151	2,077,181	65.1%
XML	392,519	2,237	0.6%	34,628	429,384	13.5%
C	357,547	51,411	12.6%	53,876	462,834	14.5%
C++	44,633	1,153	2.5%	1,078	46,864	1.5%
Make	25,112	15,464	38.1%	9,180	49,756	1.6%
shell script	21,927	3,570	14.0%	1,451	26,948	0.8%

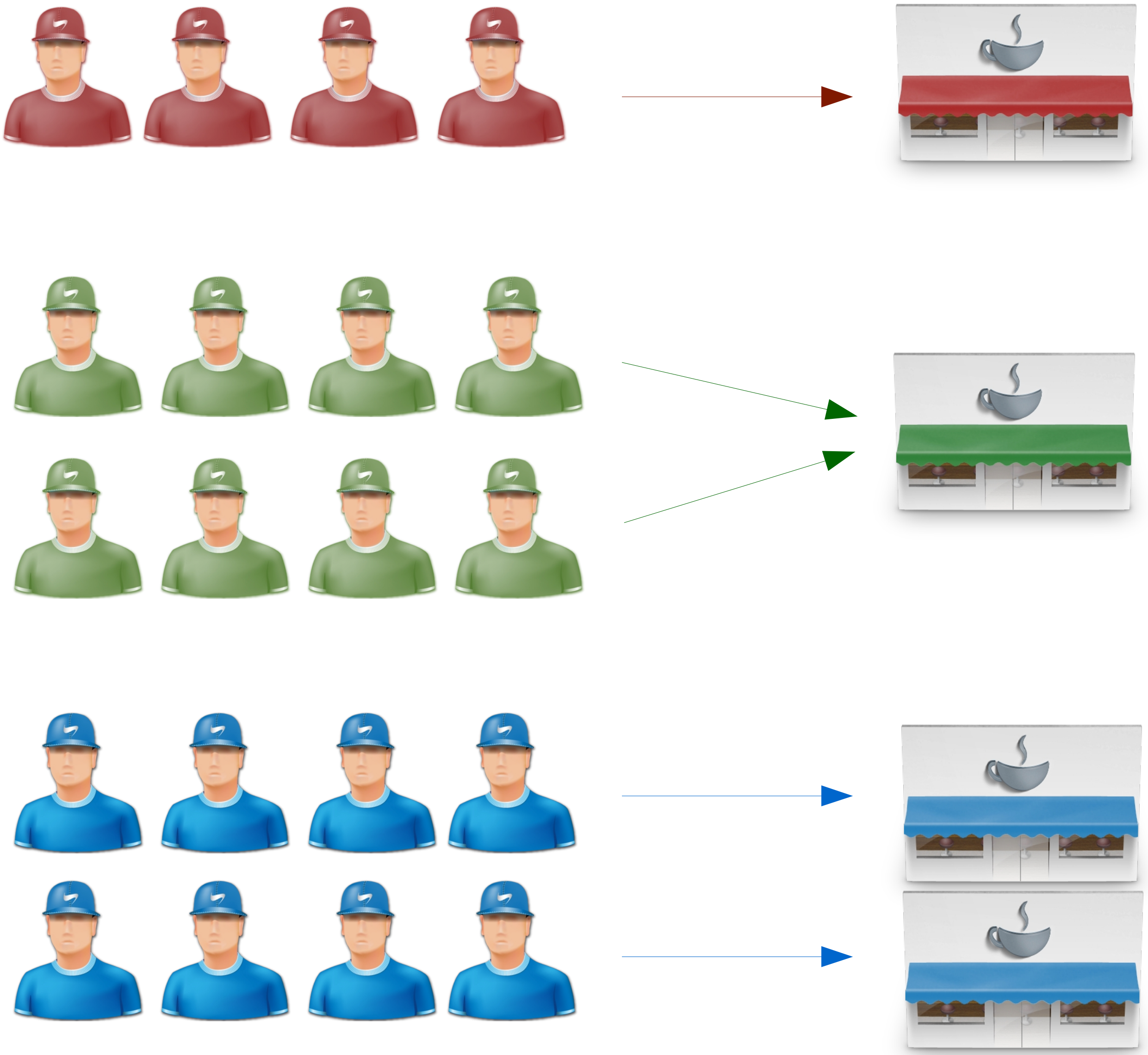
History & Philosophy

Concurrent

Fault Tolerant

Distributed

1. Concurrent

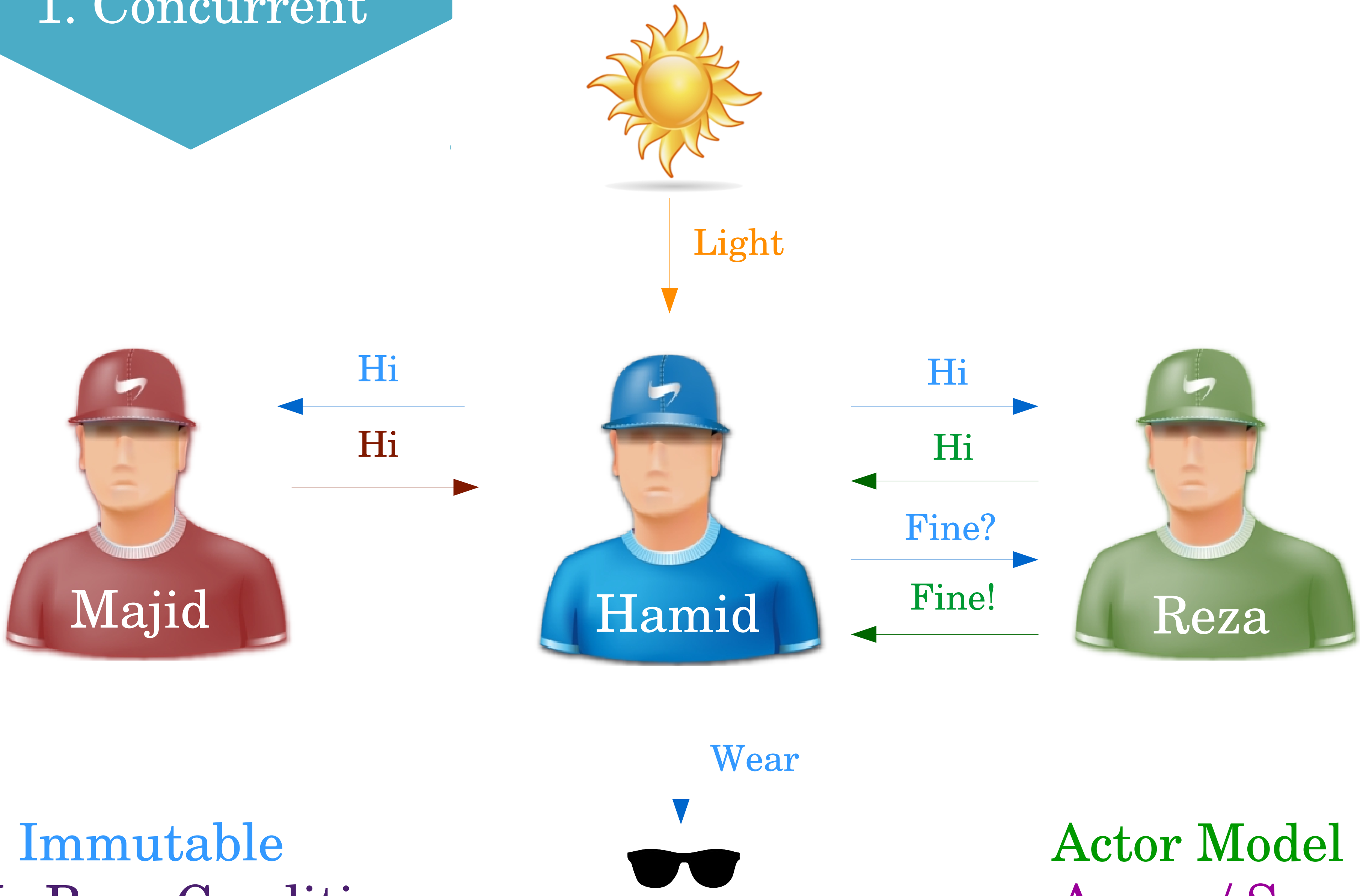


Sequential

Concurrent

Parallel

1. Concurrent

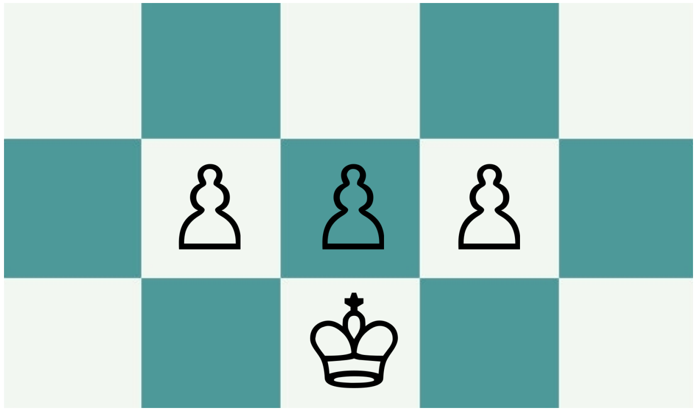


Immutable
No Race Condition
Reenterancy

Actor Model
Async / Sync
Light Process

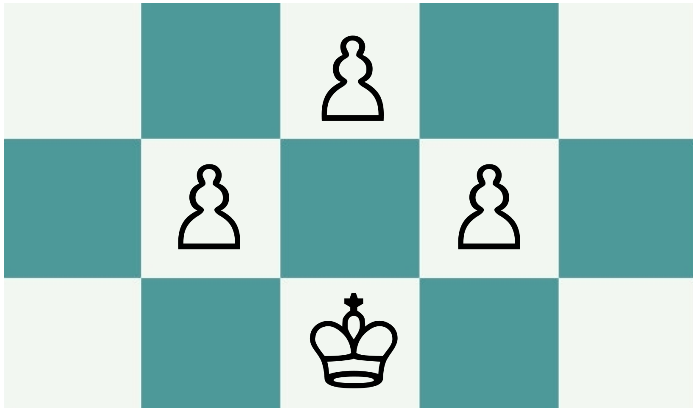
2. Fault Tolerant

Defensive



“ Try to survive! ”

Corrective



“ Let it crash! ”

2. Fault Tolerant

In Failure

“ Needs Restart ”

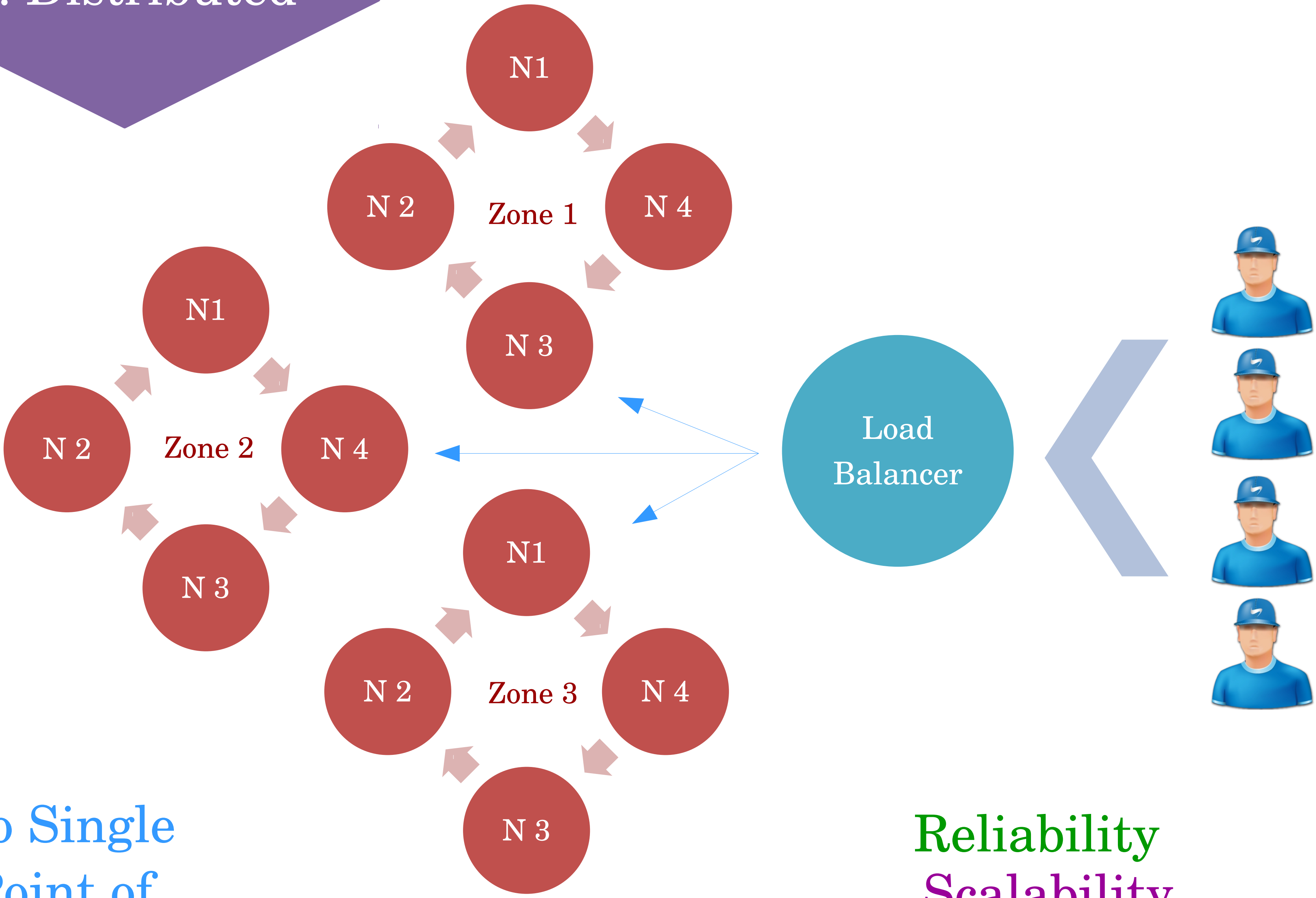
Example: Apache

“ Doesn't Need Restart ”

Example: Yaws

Hot Swapping Feature

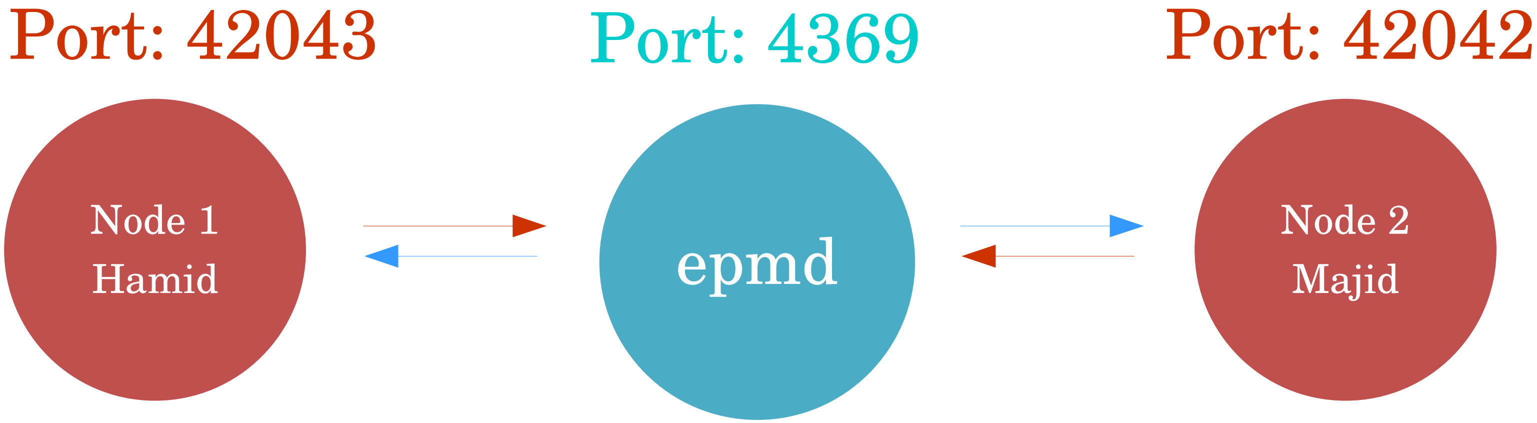
3. Distributed



No Single Point of Failure

Reliability
Scalability
Performance

3. Distributed



Erlang Port Manager Daemon

Erlang Distributed Protocol

No Need to IDL

Question?

- Hamidreza.s@gmail.com