Think in Erlang!

Viktória Fördős
Erlanger @ Cisco NSO Core
5 March 2020, Code BEAM SF

Which programming language was the first you learnt?

Basic

Java

Pascal

C

Which programming language was the first you used professionally?

Java

Cobol

Python

C/C++

PHP

Which programming language do you work with?

Elixir

Erlang

Other

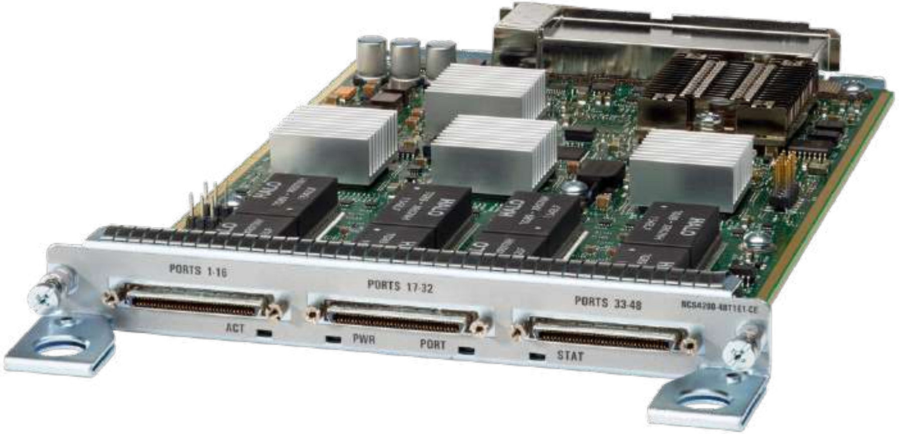
Imperative paradigm

Imperative

Statements, describing how to execute the program.

Goal is achieved through state changes.

Focus is on the control flow.



Is Erlang an imperative programming language?

Declarative paradigm

What we learn first

- High level
- Quick prototyping & testing
- No mutable objects

Next, we learn

- Lambda calculus
  - Function application
  - Function abstraction
  - Function composition
  - Higher order functions
  - Fix point theorem

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

Data

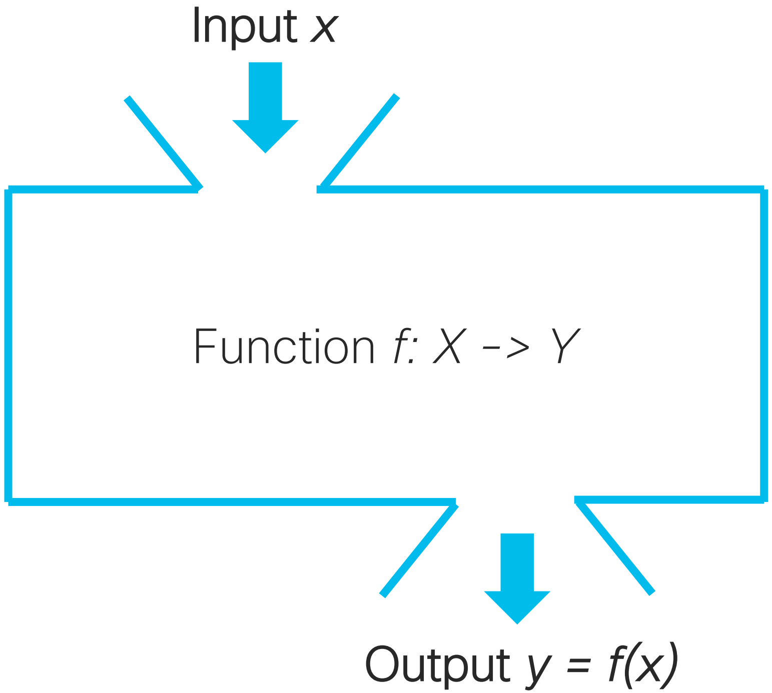
Data

Data

Data

Data

IO model



Data Transformation Function

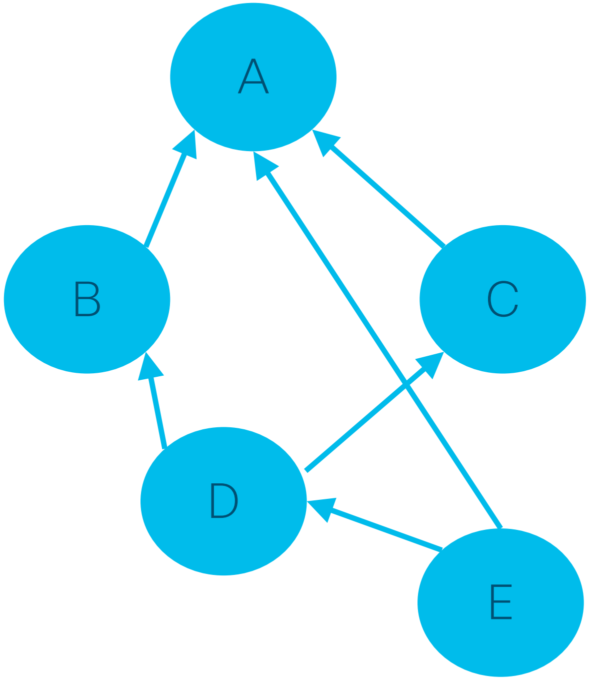
i(A, h(f(A), g(A)))

B = f(A),

C = g(A),

D = h(B, C),

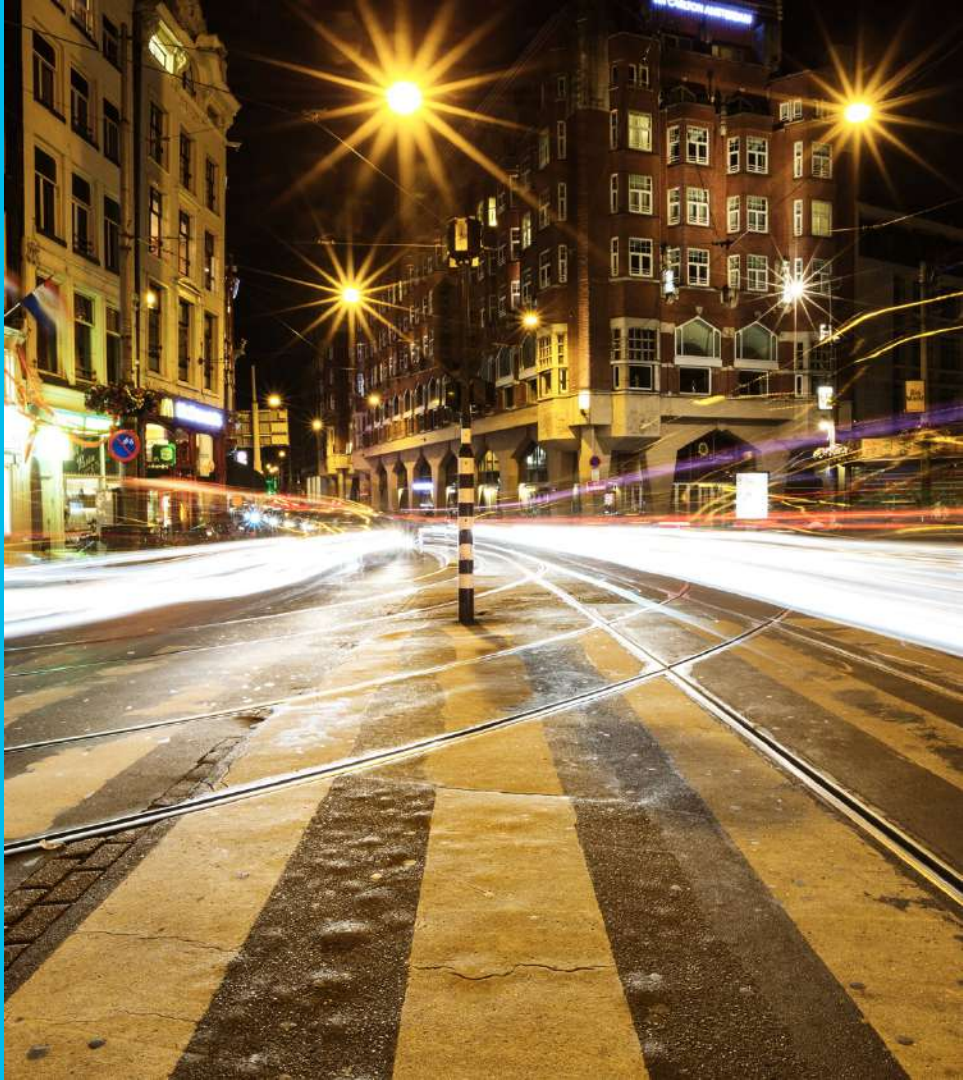
E = i(A, D)



Erlang

Concurrency

Based on data dependencies



Process

Evaluates a function.

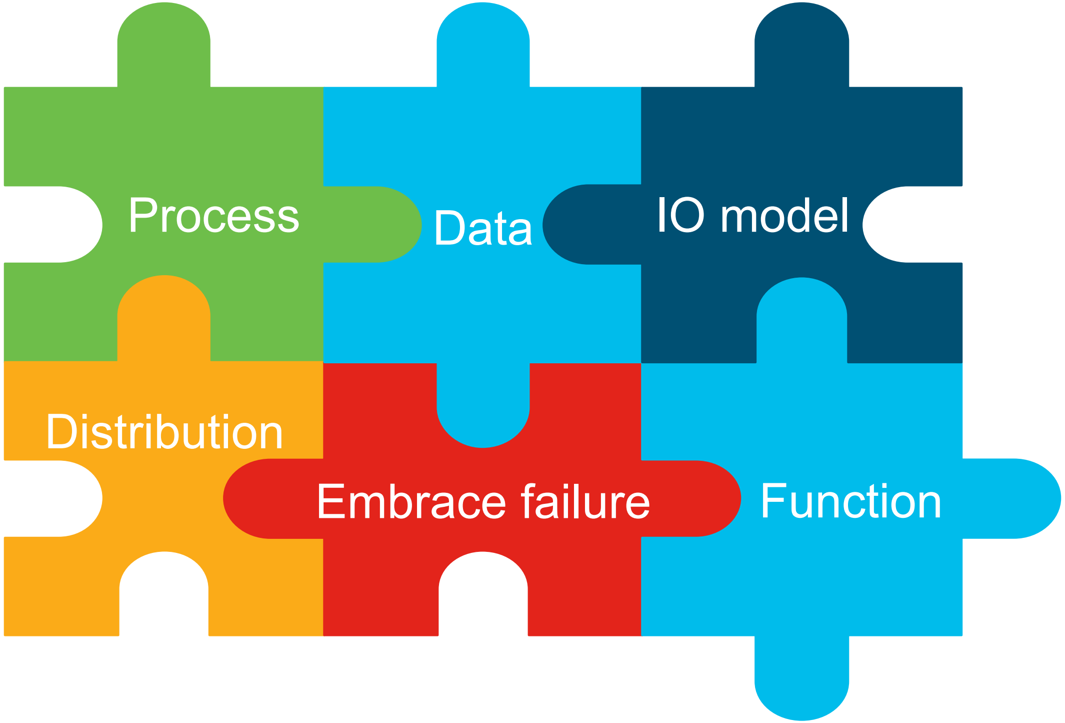
Has its own world.

Uses signals to communicate with the outside world.

Distributed

Network is not always reliable.
Things will fail. “Let it crash!”

Think in Erlang



Process

Data

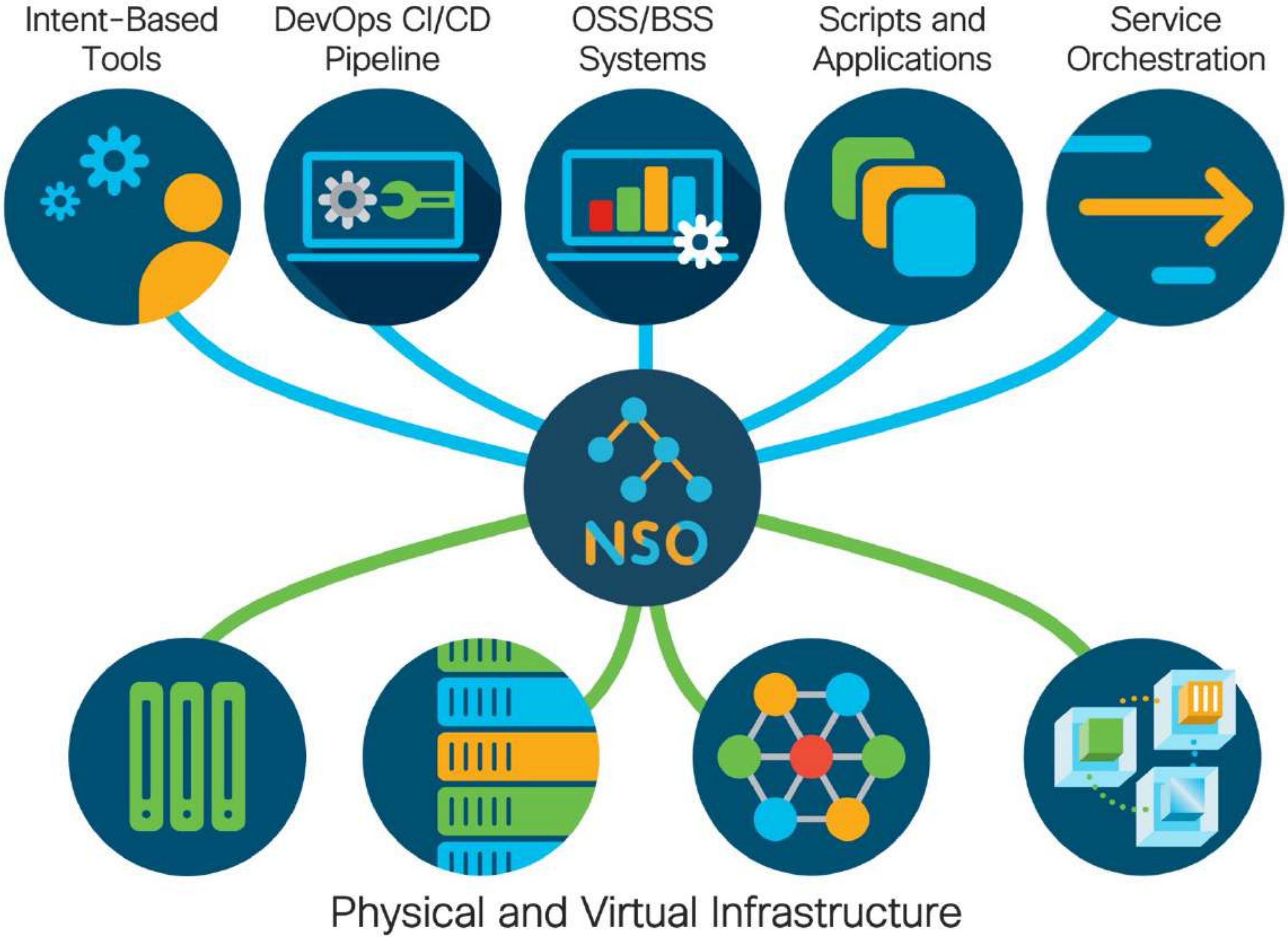
IO model

Distribution

Embrace failure

Function

Case Study


Intent-Based Tools
DevOps CI/CD Pipeline
OSS/BSS Systems
Scripts and Applications
Service Orchestration
NSO
Physical and Virtual Infrastructure

Network Services

VPNs, ACLs.

The problem

Service lifecycle management.

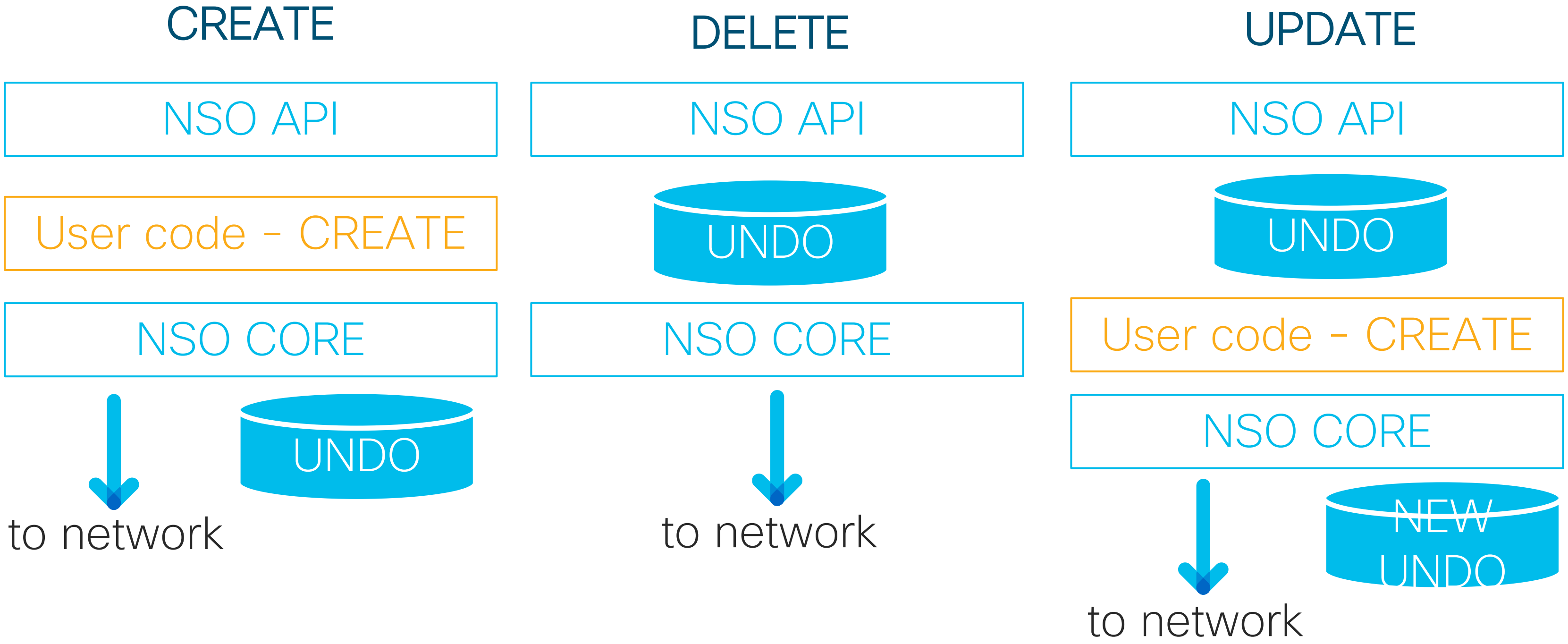
NSO way

NSO needs

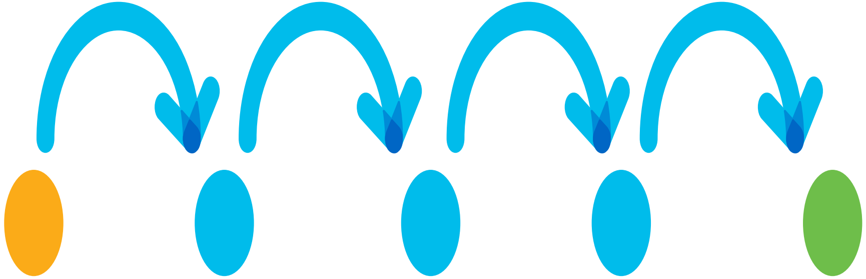
- Service model – IO model
- Mapping from service input parameters to actual device configuration. - Data transformation function.

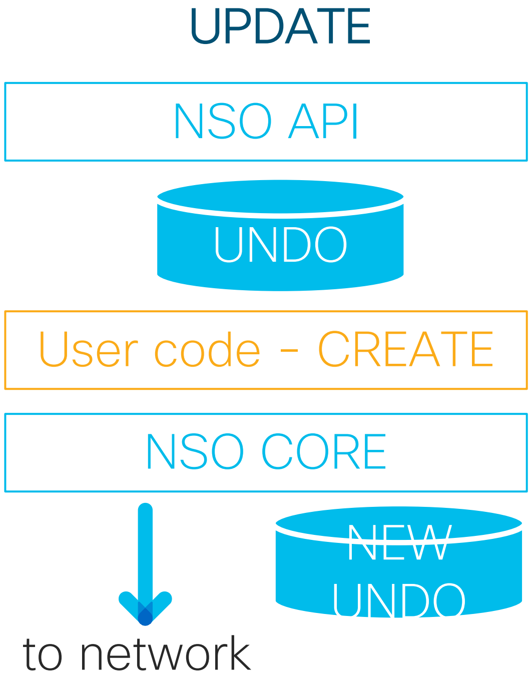
Service lifecycle management: create, update, delete.

Tail-f FASTMAP algorithm



Tail-f Reactive FASTMAP algorithm





UPDATE

NSO API

UNDO

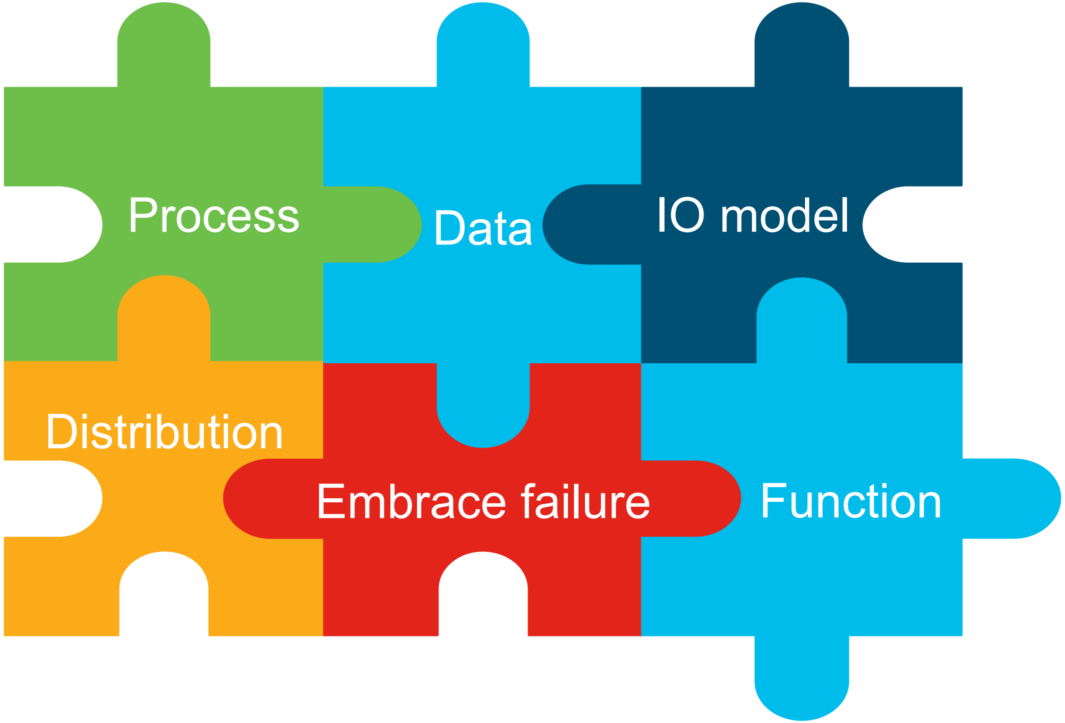
User code - CREATE

NSO CORE

NEW UNDO

to network

Think in Erlang


Process
Data
IO model
Distribution
Embrace failure
Function