Sviluppare DSL in modo pragmatico

Federico Tomassetti

VOXXED DAYS TICINO

Ciao, sono Federico

Prima…

- Dottorato in Language Engineering
- Ricerca fra Italia e Germania
- Lavorato a TripAdvisor
- Lavorato a Groupon

Ora sono un consulente indipendente su Language Engineering.

Progetto e costruisco:

- Parser
- Interpreti
- Compilatori
- Editor

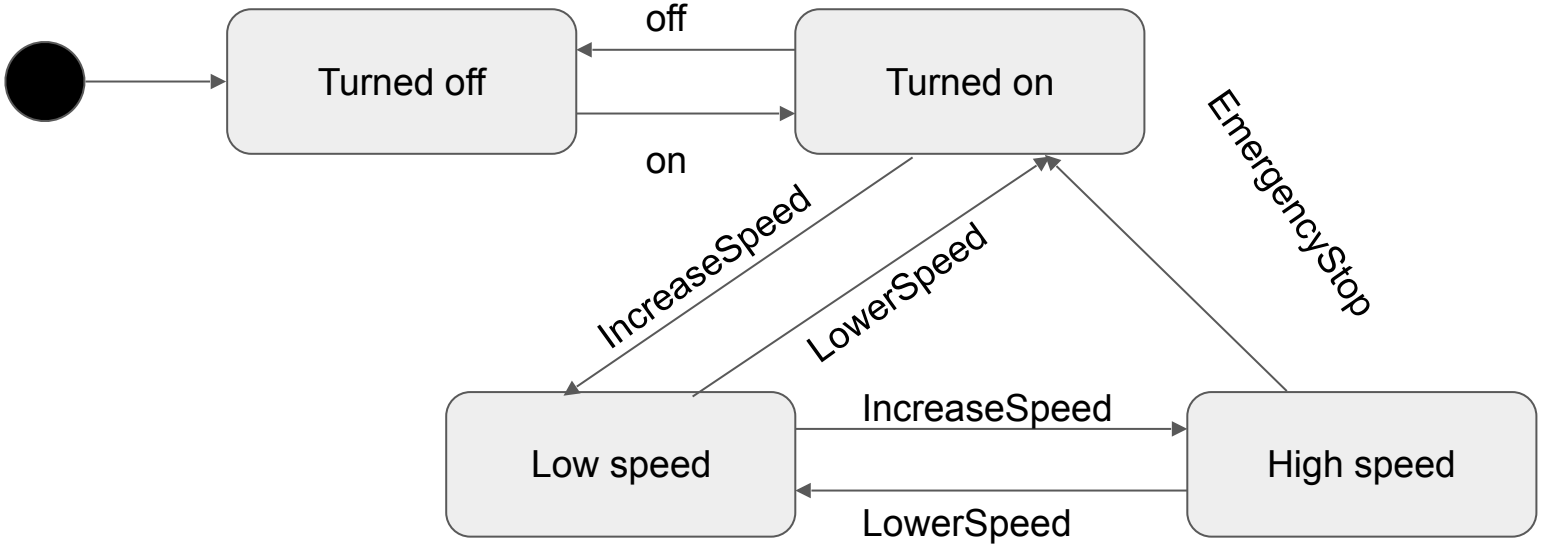
FEDERICO TOMASSETTI
Software Architect

Perché creare linguaggi?

The language or notation we are using to express or record our thoughts, are the major factors determining what we can think or express at all!

Edsger W. Dijkstra - *The Humble Programmer*

Macchine a stati



TinyFSM

```
void Idle::entry() {
  send_event(MotorStop());
}
void Idle::react(Call const & e) {
  dest_floor = e.floor;
  if (dest_floor == current_floor) return;
  /* lambda function used for transition action */
  auto action = [] {
    if (dest_floor > current_floor)  send_event(MotorUp());
    else if(dest_floor < current_floor) send_event(MotorDown());
  };
  transit<Moving>(action);
};
```

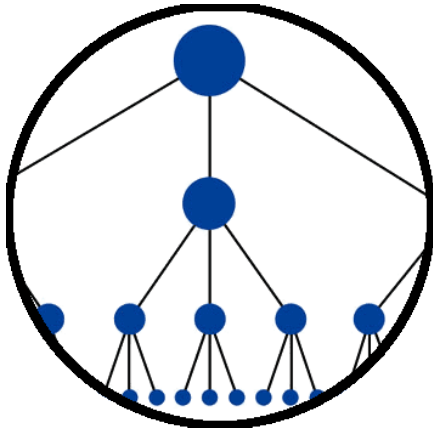
http://digint.ch/tinyfsm/doc/usage.html

FEDERICO
TOMASSETTI
Software Architect

Un po’ di conti

Cosa	Numero di linee
Lexer	~50
Parser	~40
AST	~120
Mapping	~90
Typesystem	~30
Validation	~80
Interpreter	~100
Editor	~300
Total	~810

Processo

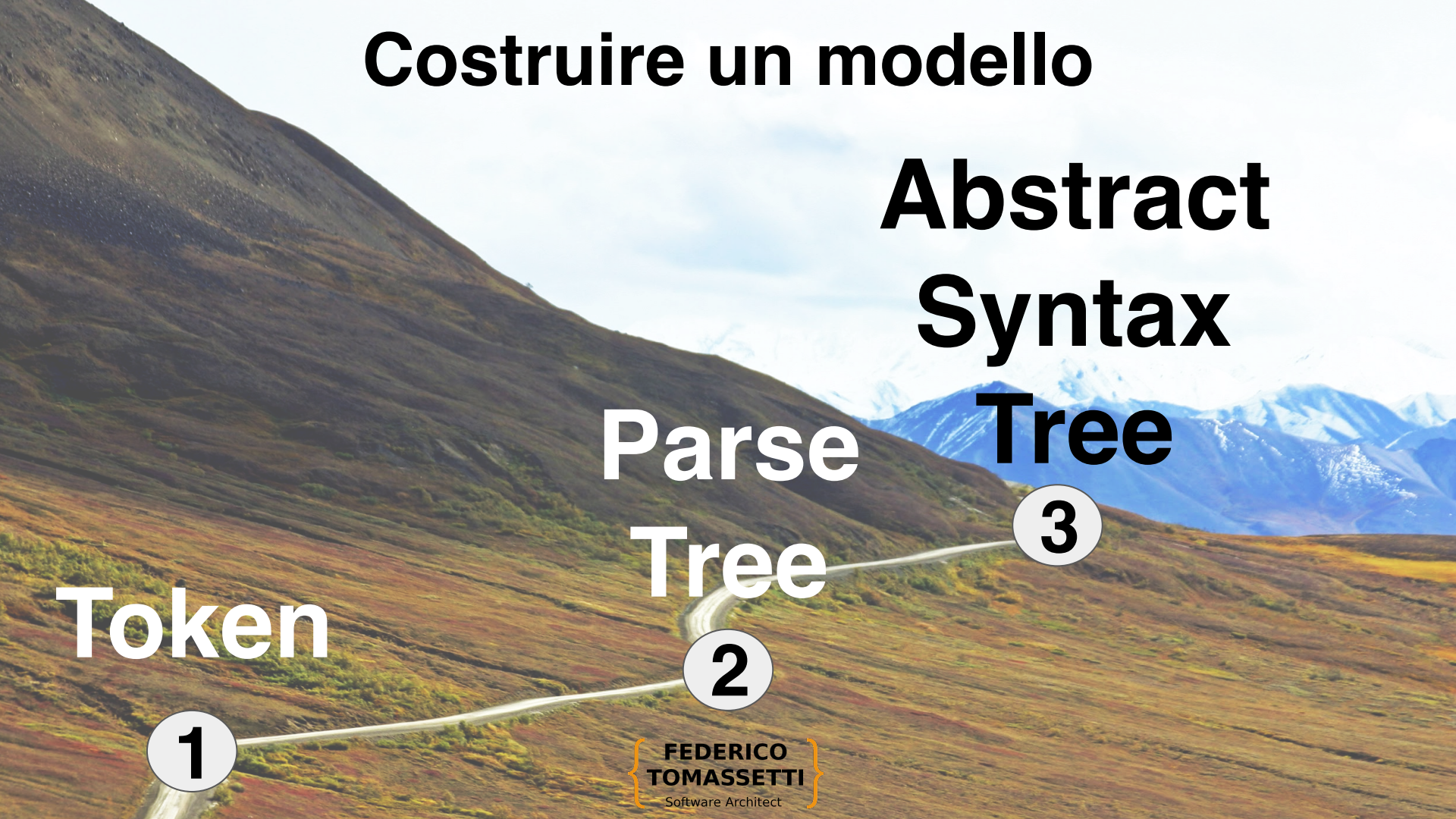


Validazione
Interprete
Compilatore

Codice **Modello**

FEDERICO TOMASSETTI
Software Architect

Costruire un modello



FEDERICO
TOMASSETTI
Software Architect

Qual’é il segreto?

FEDERICO
TOMASSETTI
Software Architect

Vediamo il codice

FEDERICO
TOMASSETTI
Software Architect

Volete saperne di più?

https://tomassetti.me/lugano

*to**m**a**ss**e**tt**i: 1 m, 2 s, 2 t*

Our example (1/2)

```
statemachine mySm

input lowSpeedThroughtput: Int
input highSpeedThroughtput: Int

var totalProduction = 0

event turnOff
event turnOn
event speedUp
event speedDown
event emergencyStop
event doNothing
```

Our example (2/2)

```
start state turnedOff {
   on turnOn -> turnedOn
}

state lowSpeed {
    on entry {
        totalProduction = totalProduction + lowSpeedThroughtput
        print("Producing " + lowSpeedThroughtput + " elements (total "+totalProduction+")")
    }
    on speedDown -> turnedOn
    on speedUp -> highSpeed
    on doNothing -> lowSpeed
}

… more states …
```

Language Workbench

FEDERICO
TOMASSETTI
Software Architect

What pieces do we need

1. Parser
2. Validator
3. Compiler/Interpreter
4. Editor

FEDERICO TOMASSETTI
Software Architect