Introduction to FSP

SFWRENG 3BB4:

Software Design III — Concurrent System Design

A Foreword on Models

Definition A model is a simplified representation of the real world.

Purpose Engineers use models to gain confidence in the adequacy and validity of a proposed design.

Characteristics

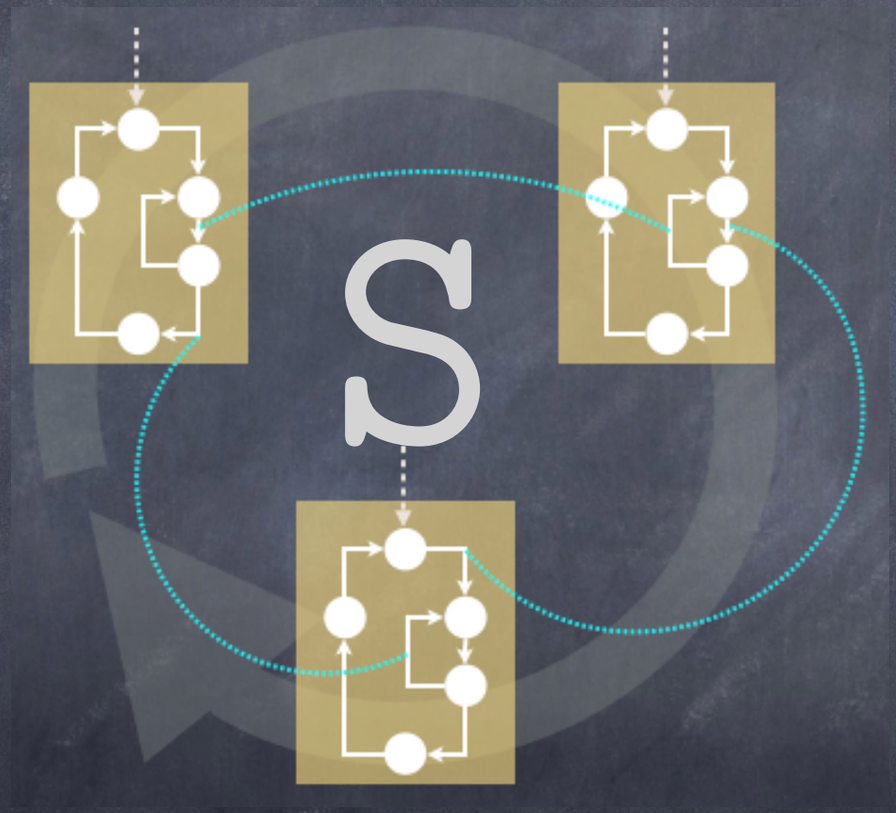
- Focus on some aspects of interest (concurrent behaviour).
- Tool for visualizing behaviour.
- Provide support for technical verification of properties of designs.

For us

- Models describe state machines known as Labelled Transition Systems (LTSs).
- LTSs are described textually as finite state processes (FSPs) and displayed and analyzed by the CLTSA analysis tool.

A Foreword on Models

The Big Picture Does $\mathbb{S}$ satisfy Φ?



Φ

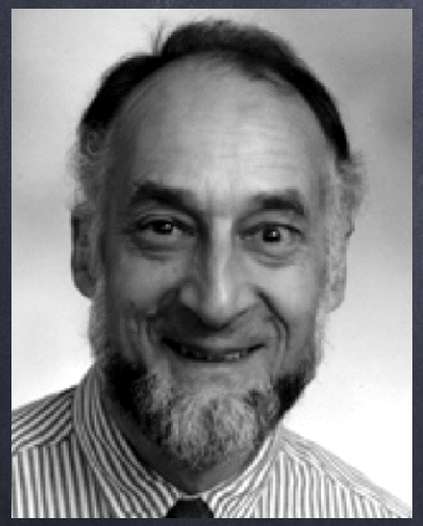
In short

- Language for describing **S** FSP
- Language for stating the properties we want **S** to satisfy FSP + Temporal Logic
- Automatic checking mechanism CLTSA

Introduction to FSP

Background

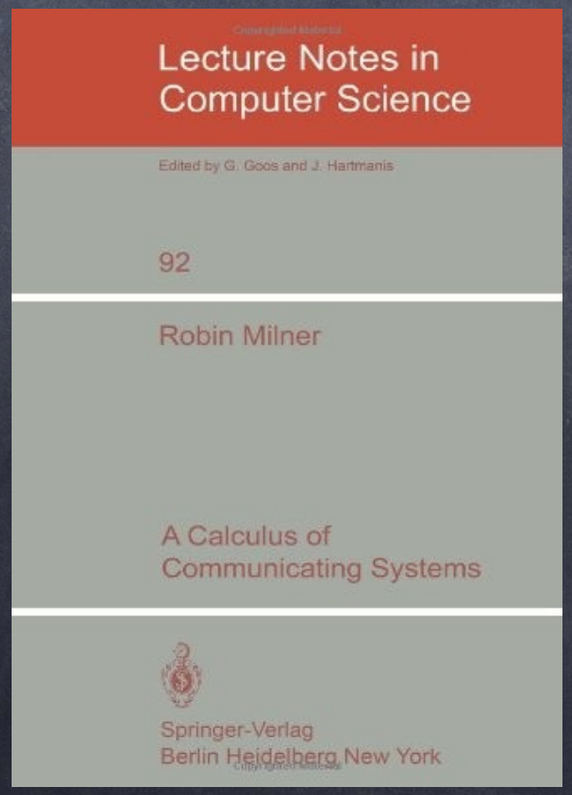
- FSP is a member of the larger family of so-called the process calculi.
- Process calculi provide a language for the high-level description of interactions, communications, and synchronizations between a collection of independent processes acting concurrently.
- Process calculi obey algebraic laws that allow process to be manipulated and analyzed, and facilitating formal reasoning about concurrent behaviour.

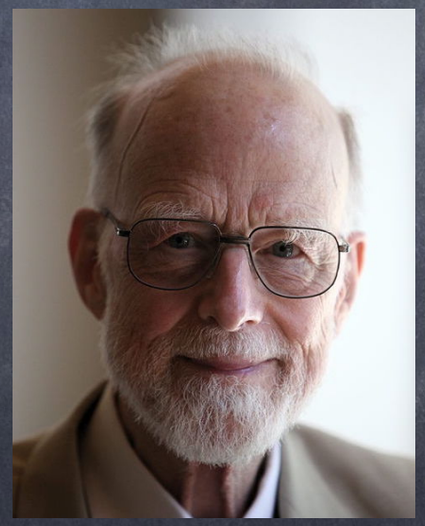


Robin Milner
(Jan 1934 – Mar 2010)

British computer scientist

1991 Turing Award winner

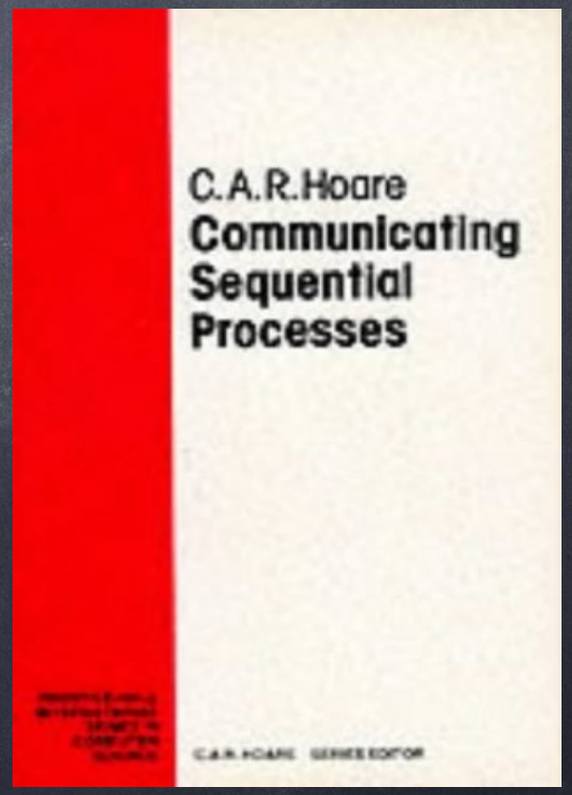




Tony Hoare
(Jan 1934 –)

British computer scientist

1980 Turing Award winner



Introduction to FSP

Background

- Basic modularization units in FSP are called processes.
- Processes are described in terms of actions.
- Actions are indivisible entities.
- Processes in FSP are interpreted as labelled transition systems (LTSs) and behaviour as sets of traces.

Jeff Kramer

British computer scientist

Jeff Magee

British computer scientist

The Language of FSP

Process


```
P = (…).
```

Action


```
a -> …
```

The Language of FSP

Relabeling

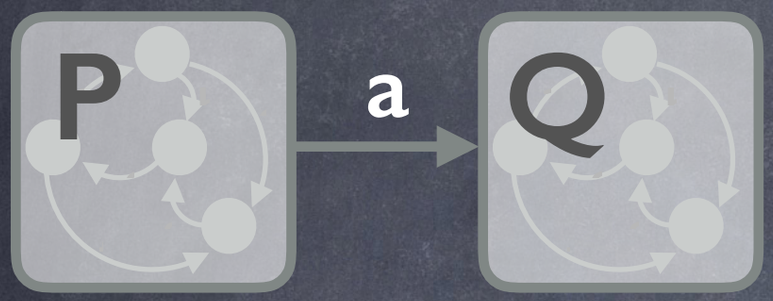

```
/{ new/old }
```

```
... ... ... ... ...
a   a   a   a   a
```

```
... ... ... ... ...
b   b   b   b   b
```

The Language of FSP

Sequencing




```
P = (a -> Q), Q = (… -> …).
```

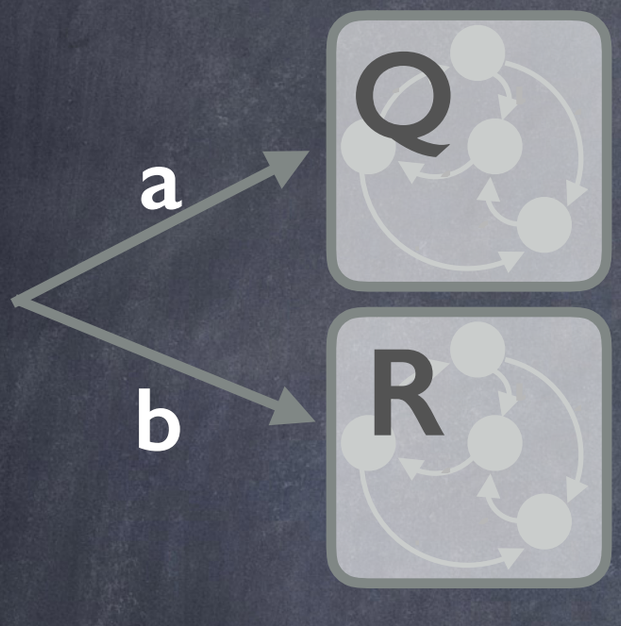
… … … … …
a a a a a

P

Q

The Language of FSP

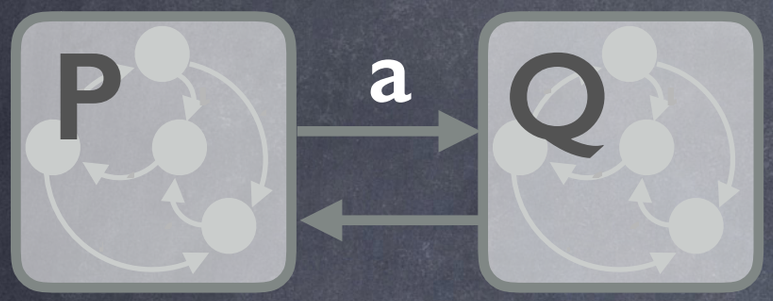
Choice




```
(a -> Q | b -> R)
```

The Language of FSP

Recursion




```
P = (a -> Q), Q = (… -> P).
```

… … … … …

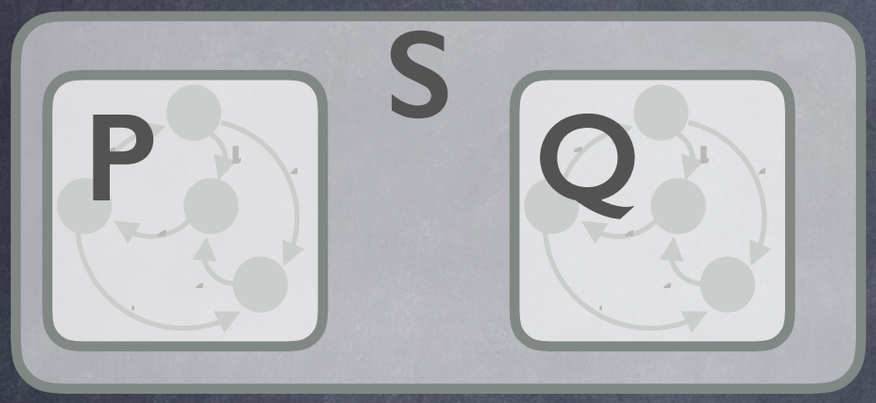
a a a a a

P

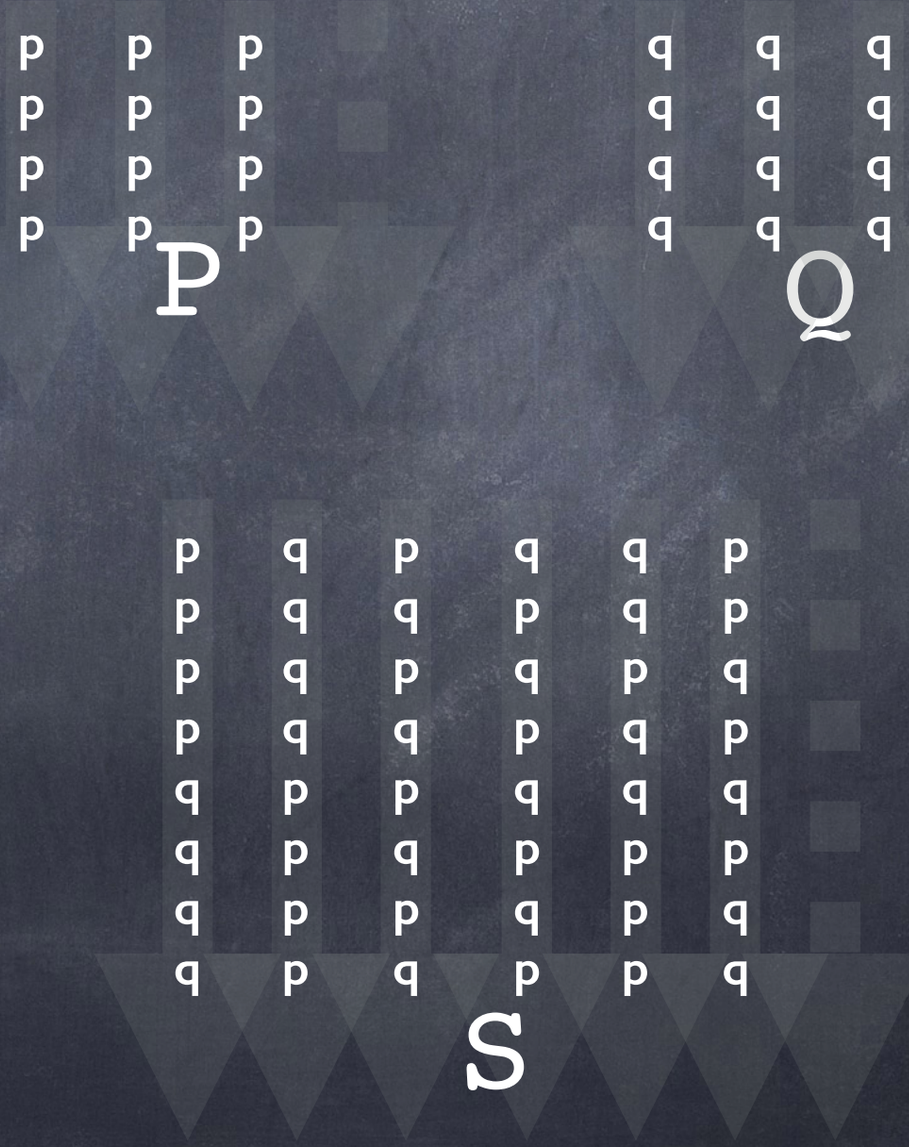
Q

The Language of FSP

Composition




```
||S = (P || Q).
```



Questions?