Imperial College London

Session Types and their Applications

Code Mesh November 5th/6th 2020 Nobuko Yoshida

Session types serve as both **theoretical** and **engineering foundations** of **communication-centred programming languages and systems** and **software**, so that the society can use **safe** **concurrent systems** and **distributed services.**

Communications are Ubiquitous

- Increasingly, **communications** are the way to organise software and systems.
- Industry trend – programming languages with **explicit message-passing primitives.**

microservices






Problems: Ambiguity

- Protocol descriptions are **ambiguous**
- **SMTP: simple mail transfer protocol**
  - They are written in English, often very long

RFC 821 August 1982
Simple Mail Transfer Protocol

TABLE OF CONTENTS

Problems: Ambiguity

- Protocol descriptions are **ambiguous**
- **SMTP: simple mail transfer protocol**
  - They are written in English, often very long

```
3.1.  MAIL

   There are three steps to SMTP mail transactions.  The transaction
   is started with a MAIL command which gives the sender
   identification.  A series of one or more RCPT commands follows
   giving the receiver information.  Then a DATA command gives the
   mail data.  And finally, the end of mail data indicator confirms
   the transaction.

      The first step in the procedure is the MAIL command.  The
      <reverse-path> contains the source mailbox.

         MAIL <SP> FROM:<reverse-path> <CRLF>

      This command tells the SMTP-receiver that a new mail
      transaction is starting and to reset all its state tables and
      buffers, including any recipients or mail data.  It gives the
      reverse-path which can be used to report errors.  If accepted,
      the receiver-SMTP returns a 250 OK reply.

      The <reverse-path> can contain more than just a mailbox.  The
      <reverse-path> is a reverse source routing list of hosts and
      source mailbox.  The first host in the <reverse-path> should be
      the host sending this command.

      The second step in the procedure is the RCPT command.

         RCPT <SP> TO:<forward-path> <CRLF>

      This command gives a forward-path identifying one recipient.
      If accepted, the receiver-SMTP returns a 250 OK reply, and
      stores the forward-path.  If the recipient is unknown the
      receiver-SMTP returns a 550 Failure reply.  This second step of
      the procedure can be repeated any number of times.
```

Problems: Concurrency Bugs

- Communications increase **concurrency bugs**
  - Survey of 4K users [golang.org]
  - Analysis of 6 large software systems [ASPLOS 19]

deadlock

channel errors

More than a half of concurrency bugs in Go are caused by communications.

The Go Gopher

Problems: Concurrency Bugs

- Communications increase **concurrency bugs**
  - Survey of 4k users **[golang.org]**
  - Analysis of 6 large software systems **[ASPLOS 19]**

More than a half of concurrency bugs in Go are caused by communications.

Session Types

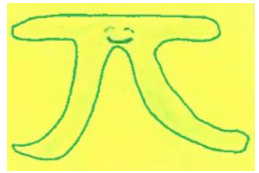
- Prevent concurrency bugs.
- Can abstract, implement and manage communications as **Protocols.**
- **Clean, Cheap** and **Retrofittable.**

Why Session Types, Why Now?

Significant academic and industry interests via fundamental breakthroughs

Milner, Honda, NY

Binary Session Types — ESOP'98

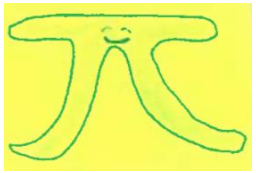


Joined W3C Standardization — 2002

Why Session Types, Why Now?

Significant academic and industry interests via fundamental breakthroughs

Binary Session Types — ESOP'98



↓

Joined W3C Standardization — 2002

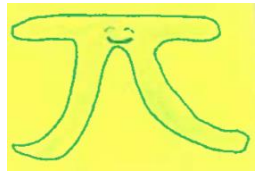
↓

Multiparty Session Types — POPL' 08

Why Session Types, Why Now?

Significant academic and industry interests via fundamental breakthroughs

Binary Session Types — ESOP'98



↓

Joined W3C Standardization — 2002

↓

Multiparty Session Types — POPL' 08

↓

largest open source company in the world

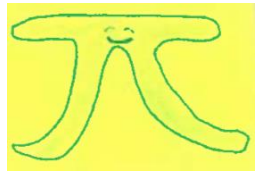
Red Hat

Scribble

Why Session Types, Why Now?

Significant academic and industry interests via fundamental breakthroughs

Binary Session Types — ESOP'98



Joined W3C Standardization — 2002

Multiparty Session Types — POPL' 08

25M US$ large cyber infrastructures

OCEAN OBSERVATORIES INITIATIVE

Red Hat

Scribble

middleware standardisations

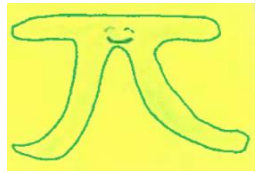
AMQP Advanced Message Queuing Protocol

RabbitMQ Messaging that just works

Why Session Types, Why Now?

Significant academic and industry interests via fundamental breakthroughs

Binary Session Types — ESOP'98



Joined W3C Standardization — 2002

Multiparty Session Types — POPL' 08







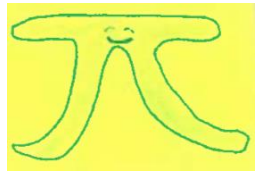



Why Session Types, Why Now?

Significant academic and industry interests via fundamental breakthroughs

ETAPS Test Time Award 2019

Binary Session Types — ESOP'98



↓

Joined W3C Standardization — 2002

↓

POPL Influential Paper Award 2018

Multiparty Session Types — POPL' 08

↓

Why Session Types, Why Now?

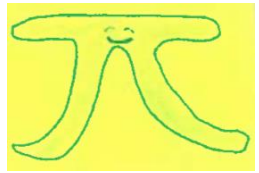
Significant academic and industry interests via fundamental breakthroughs

ETAPS Test Time Award 2019

PLDI'19, ICSE'18, CC'18, CAV'19
POPL'19, POPL'19, POPL'19,
ECOOP'19, ESOP'19, ESOP'18
CONCUR'19, FOSSACS'19,
LICS'18, FOSSACS'18

POPL Influential Paper Award 2018

Binary Session Types — ESOP'98



Joined W3C Standardization — 2002

Multiparty Session Types — POPL' 08