# Introduction to Chor

## A video tutorial

Fabrizio Montesi

IT University of Copenhagen (Denmark)
italianaSoftware s.r.l. (Italy)

# About Chor

- A new programming language for concurrent communicating systems.

- Based on **choreographies**: global descriptions of systems.

- Formally specified.

- Supported by an Eclipse-based IDE.

- Developed following discussions both in the Academia and the Industry.

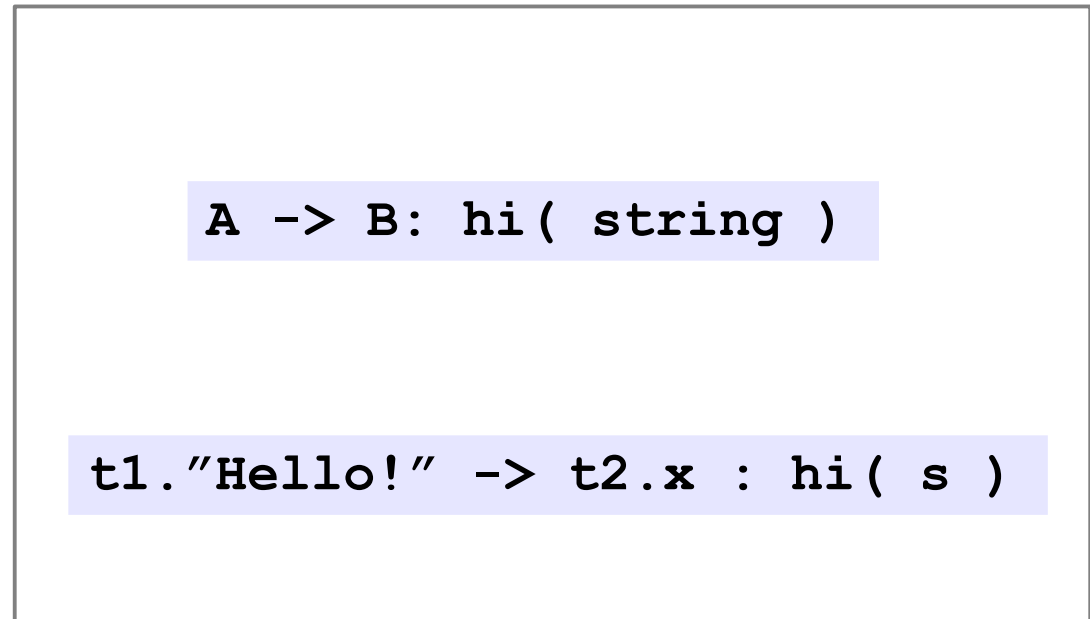- Open source.

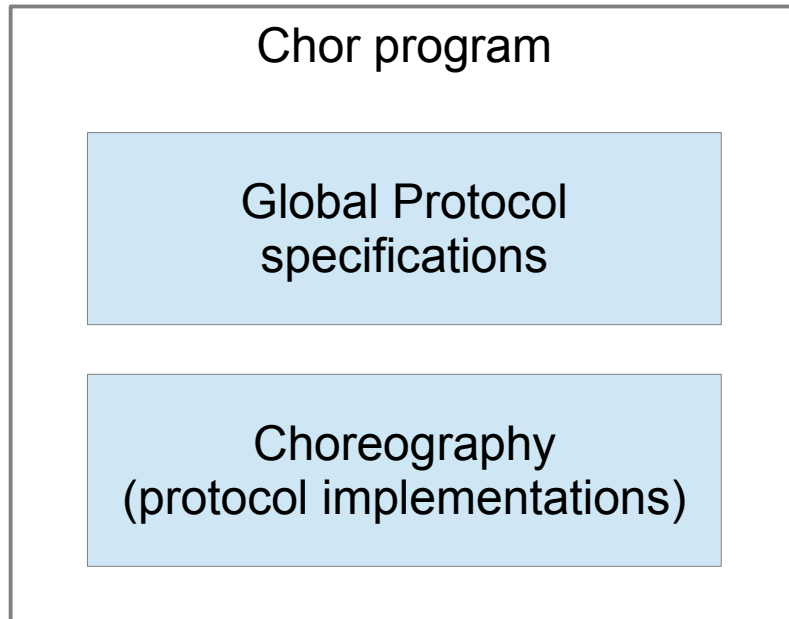- Still a prototype, but rapidly evolving.

# Why Chor? Some problems it solves...

- **Protocols:** multiparty protocols are usually informally specified, and can be hard to implement.
- **Solution:** Chor allows you to write protocol specifications, and check your programs for compliance.

- **Deadlocks:** concurrent programming is prone to writing deadlocks, and these can be very hard to detect.
- **Solution:** Chor programs are statically checked to be deadlock-free.

- **Productivity:** prototyping complex distributed systems can be a long process, when there are many entities to be implemented.
- **Solution:** Chor defines the behaviour of all entities in a single program, and then generates their code automatically.

- **Clarity:** understanding how distributed entities will interact can be hard, since you have to match their I/O communications manually.
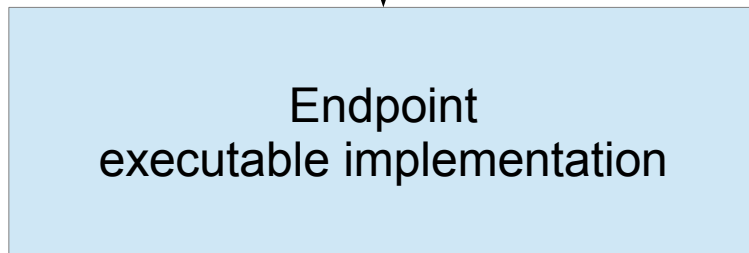- **Solution:** Chor programs make this matching explicit.

# Development methodology

| Artifacts | Examples |
|---|---|

### Chor program

Global Protocol
specifications

Choreography
(protocol implementations)

```
A -> B: hi( string )
```

```
t1."Hello!" -> t2.x : hi( s )
```

Endpoint Projection
(compilation)

Endpoint
executable implementation

# Distributed Authentication

- A distributed authentication protocol, inspired by **OpenID**

- A service (called Relying Party) delegates authentication to another service (called Identity Provider).
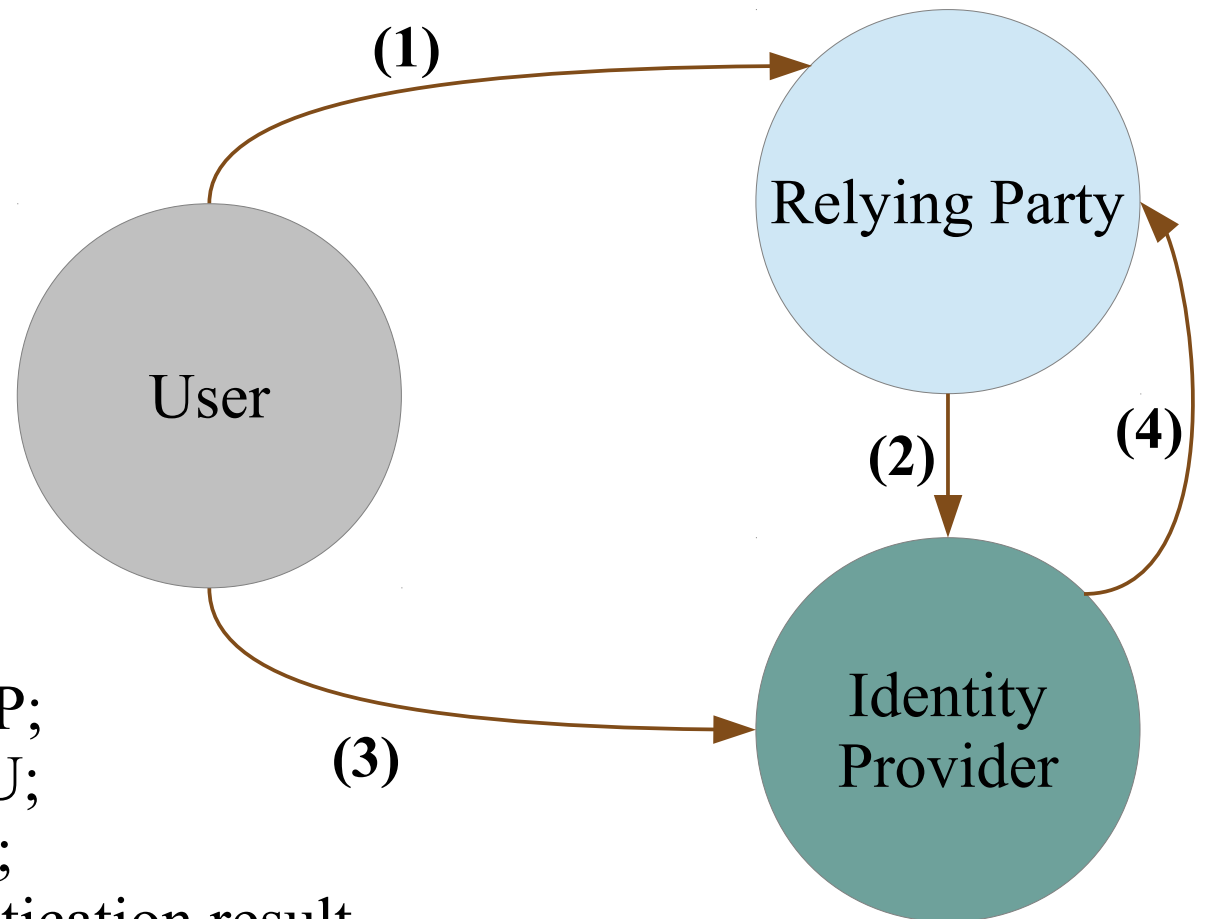


(1) U(ser) requests access to RP;
(2) RP asks IP to authenticate U;
(3) U sends her password to IP;
(4) IP notifies RP of the authentication result.

- Our projection supports the Jolie language.

- You can edit the projected Jolie code to integrate it with many different deployment setups, e.g. HTTP sockets or Bluetooth.

- Through Jolie, we support many application domains, from multicore applications to distributed service-oriented architectures.

- Chor is language independent, so we plan to extend projection to other frameworks in the future.

# More information at...

- Website:      `http://www.chor-lang.org/`

- Mailing List: `chor-devel@lists.sourceforge.net`