Session 1: Mobility

Nobuko Yoshida

(Mobius)

Department of Computing

Imperial College London

Mobius Project

- to develop the technology for establishing trust and security for the next generation of global computers, using the Proof Carrying Code paradigm
- Theoretical well-founded technologies for mobility and security

  Types, Logics, Secure Information Flow, Certificates, Resource and Alias Controls, Distributed PCCs...

  - Can offer security technologies for other projects

    Grid, Service Orientation, Global Computing, ...
  - Experiment in other projects will lead to new topics

Mobius Project

- to develop the technology for establishing trust and security for the next generation of global computers, using the Proof Carrying Code paradigm
- Theoretical well-founded technologies for mobility and security

  Types, Logics, Secure Information Flow, Certificates, Resource and Alias Controls, Distributed PCCs...

  - Can offer security technologies for other projects

    Grid, Service Orientation, Global Computing, ...
  - Experiment in other projects will lead to new topics

Mobile Computing is Ubiquitous

- Computing Machines, Network Infrastructure, Applications in Internet, Mobile Robots, Bio-Info, ...
- Basic Questions
  - How to understand mobile behaviour systematically?
  - What is this abstract (mathematical) entity called mobility?
  - Given some system/software, how can we formally specify/describe/control its mobile behaviour?

Mobile Computing is Ubiquitous

- Computing Machines, Network Infrastructure, Applications in Internet, Mobile Robots, Bio-Info, ...
- Basic Questions $\Longrightarrow$ Model/Calculus
  - How to understand mobile behaviour systematically?
  - What is this abstract (mathematical) entity called mobility?
  - Given some system/software, how can we formally specify/describe/control its mobile behaviour?

Mobility: Theories and Applications

- Basic Mobility Communication and Name Passing

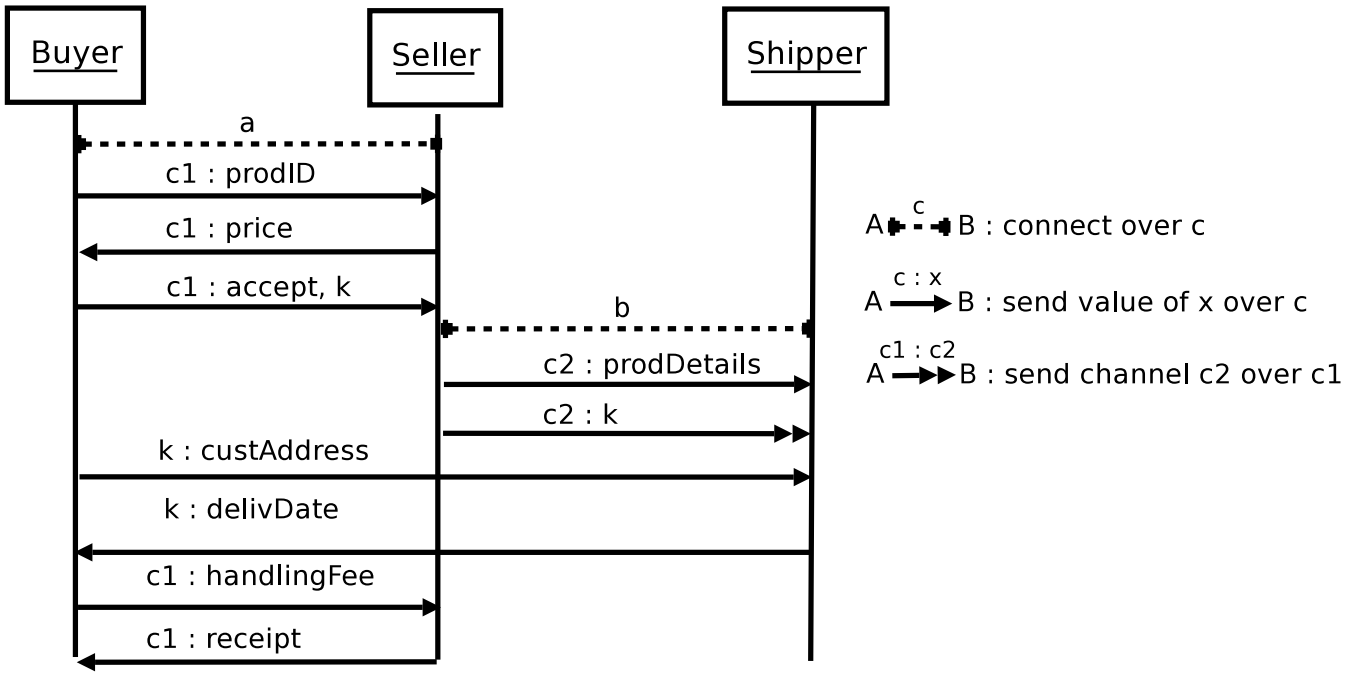
  $\Longrightarrow$ Applications W3C WS-CDL via Session Types

- Distributed Mobility Code and Proof Distribution

  $\Longrightarrow$ Applications A distributed multi-threaded Java and Distributed PCC via Fine-Grained Types

- An integrated framework via typed mobile processes

Protocol Example



Scenario: Item Purchasing (Typical W3C example)

Challenges

- How can we design languages for Web Services?

  $\Longrightarrow$ use the π-calculus as an underlying formal model

- What are good programming and type disciplines for Web Services?

  $\Longrightarrow$ use the type theory of the π-calculus (session types) for structured programming of communication and concurrency

- How can we correctly implement global scenarios?

  $\Longrightarrow$ propose a semantics, type and structured preserving End-Point-Projections from Web Service languages to the π-calculus

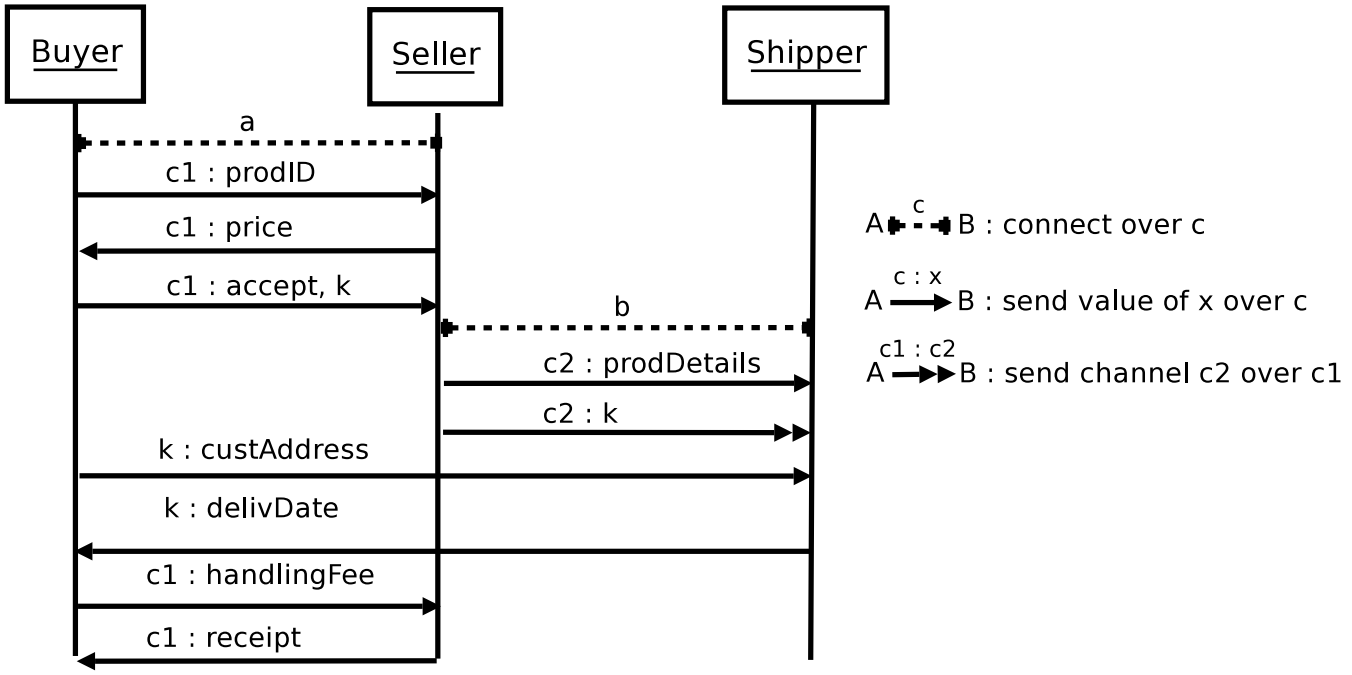
WS-Choreography Description Language

- XML-based description language for business protocols.
- Developed by W3C's CDL WG (2003∼, chaired by Steve Ross-Talbot and Martin Chapman).
- Central idea: **choreography** (cf. orchestration).

  Dancers dance following a global scenario without a single point of control.

- Pi-calculus experts (Kohei Honda, Robin Milner and Nobuko Yoshida) invited in 2004.
- Now Candidate Recommendation, reaching a W3C standard soon.

Protocol Example



$\uparrow \mathsf{id}; \downarrow \mathsf{double}; \{\mathbf{accept} : \uparrow \beta; \uparrow \mathsf{double}; \downarrow \mathsf{receipt} \oplus \mathbf{reject}\}$

$\beta = \uparrow \mathsf{address}; \downarrow \mathsf{date}$

Buyer's viewpoint of the Buyer-Seller interaction

End-Point Projection (EPP)

- A notion informally (introduced and) discussed in WS-CDL WG. *How can we project a global description to endpoints so that their interactions precisely realise the original global description?*
- Basis for execution, monitoring, validation, reuse, conformance, interoperability,...
- Demands formalisation of global and end-point descriptions.

$$(I, \sigma) \quad \mapsto \quad A[P]_{\sigma@A} \mid B[Q]_{\sigma@B} \mid C[R]_{\sigma@C} \mid \cdots$$

Scenarios for Distributed PCC

Extending the PCC model with one code producer and one code consumer to more complex scenarios in order to make the MOBIUS technology widely applicable.

Multiple Verifiers

TV: Trusted Verifier

LV: Local Verifier

P: Code Provider

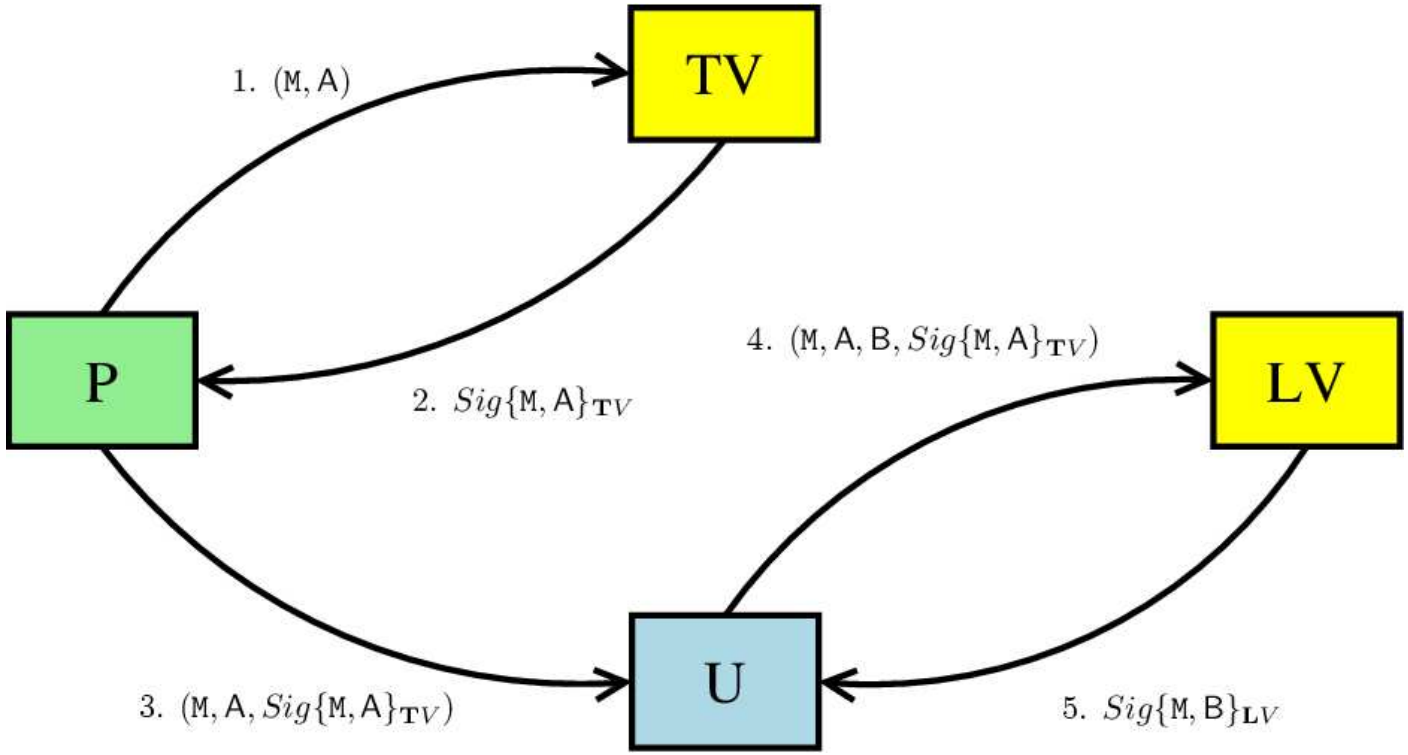
U: Code User

A: Global Security Policy

B: Local Security Policy

U purchases an application program *M* from **P**.

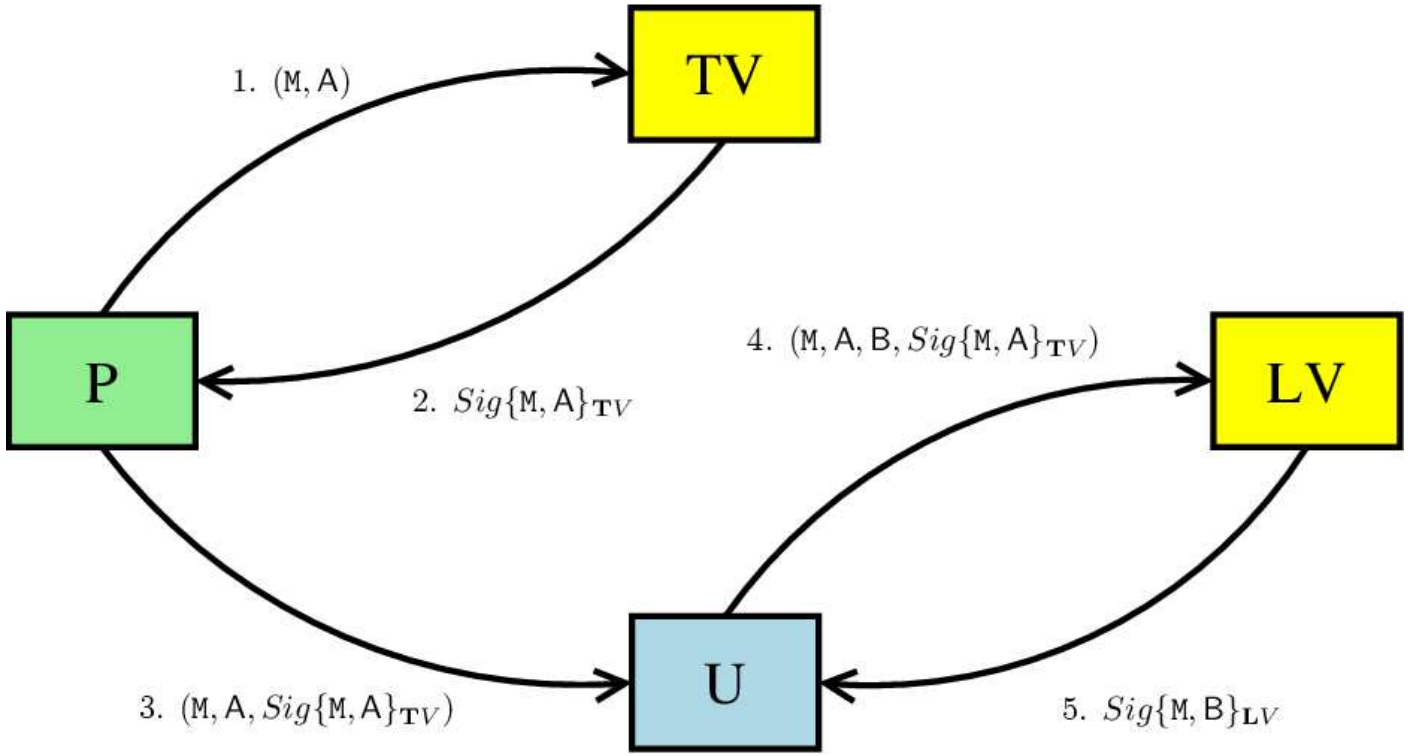
Security Goal: *M* satisfies a security policy *B* local to **U**.



TV: Trusted Verifier **LV**: Local Verifier

P: Code Provider **U**: Code User

A: Global Security Policy **B**: Local Security Policy



TV: Trusted Verifier **LV**: Local Verifier

P: Code Provider **U**: Code User

A: Global Security Policy **B**: Local Security Policy

LV only has to check **A** $\supset$ **B**

Outputs: Basic Mobility

- Session Type Theory (1994∼)

  Bonelli, Carbone, Comagnoni, Dezani, Drossopoulou, Garralda, Gunter, Gay, Hole, Honda, Kubo, Mostrous, Neubauer, Ravana, Takeuchi, Thiemann, Vallecillo, Vasconcelos, Yoshdia

- Languages (Concurrent ML, Haskell, Java, C#)

- Standardisation (W3C CDL) and Industry (Pi4Tech)

Outputs: Distributed Mobility

- Theory

  Higher-Order π-Calculus and Advanced Distributed Calculus based on Locations

  [LICS00,Inf.&.Comp,POPL04,FoSSaCs04,ActaInfo05]

- Language Design and Correctness of existing RMI Java Optimisation [OOPLSA'05,TCS]

- Distributed PCC (Mobius Task 4.1)

Future Topics and Discussions

- The technologies for a transfer from upstream to applied research, or from applied research towards exploitation

  $\Longrightarrow$ from a theory to language/runtime design to standardisation (with feedbacks).

- Merits and missing elements

  $\Longrightarrow$ theory as enabling technologies (e.g. signed code guaranteeing good behaviour)

  $\Longrightarrow$ demand broad experiments for usability

- Common areas for future collaboration between the different disciplines
  - Web Services
  - Code Validations by Types and Logics
  - Design and Development for Secure Languages and Infrastructures for Mobility
  - Protocol Validation and Development in Grid and GC