

On Temporal Abstractions of Web Service Protocols

Boualem Benatallah
boualem@cse.unsw.edu.au
CSE, UNSW
Sydney NSW 2052, Australia



Fabio Casati
casati@hp1.hp.com
Hewlett-Packard Laboratories
Palo Alto, CA, 94304 USA



Julien Ponge, Farouk Toumani
{ponge,ftoumani}@isima.fr
Laboratoire LIMOS - CNRS UMR 6158
ISIMA - Campus des Cézeaux
63173 Aubière cedex, France



Introduction

- Web services are increasingly gaining acceptance as a framework for **facilitating application-to-application interactions** within and across enterprises.
- Tools supporting service development today provide little support for **high level modeling and analysis** of abstractions at higher level of services stack.
- When developing our framework for service protocols modeling, analysis, and management, we identified the need for representing **temporal abstractions** in protocol descriptions.

Modeling temporal abstractions in business protocols

- Timed business protocols are deterministic finite state machines.
- Two kinds of temporal abstractions:
 - **must-invoke**: implicit transitions
 - **can-invoke**: explicit transitions
- Formal semantics based on timed traces.

Compatibility and replaceability in timed protocols

Protocol management operators can be identified to perform the following type of analysis.

1. **Compatibility analysis**: when can two services interact correctly ?
2. **Replaceability analysis**: when can a given service replace another one ?

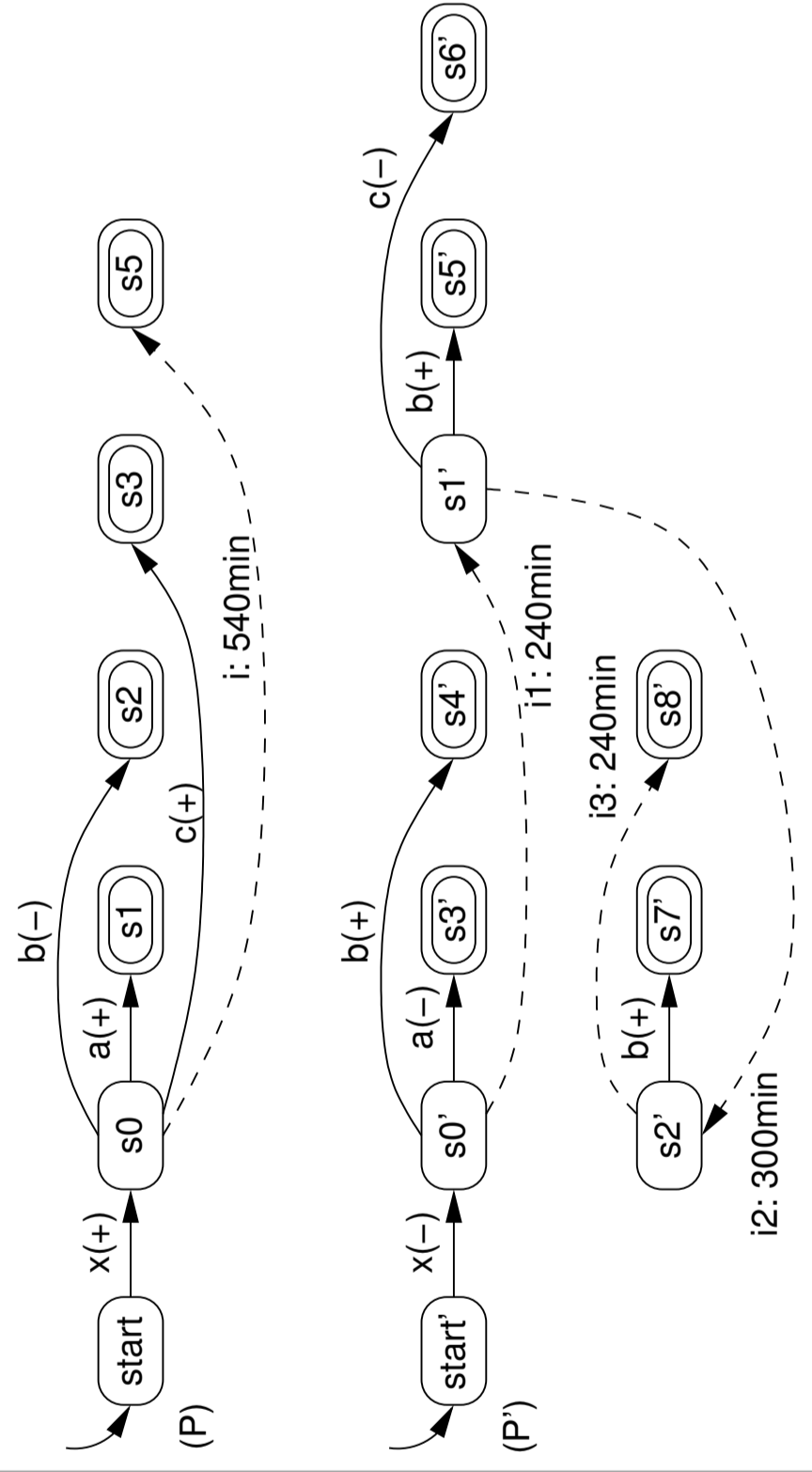


Figure 2: Two compatible timed protocols.

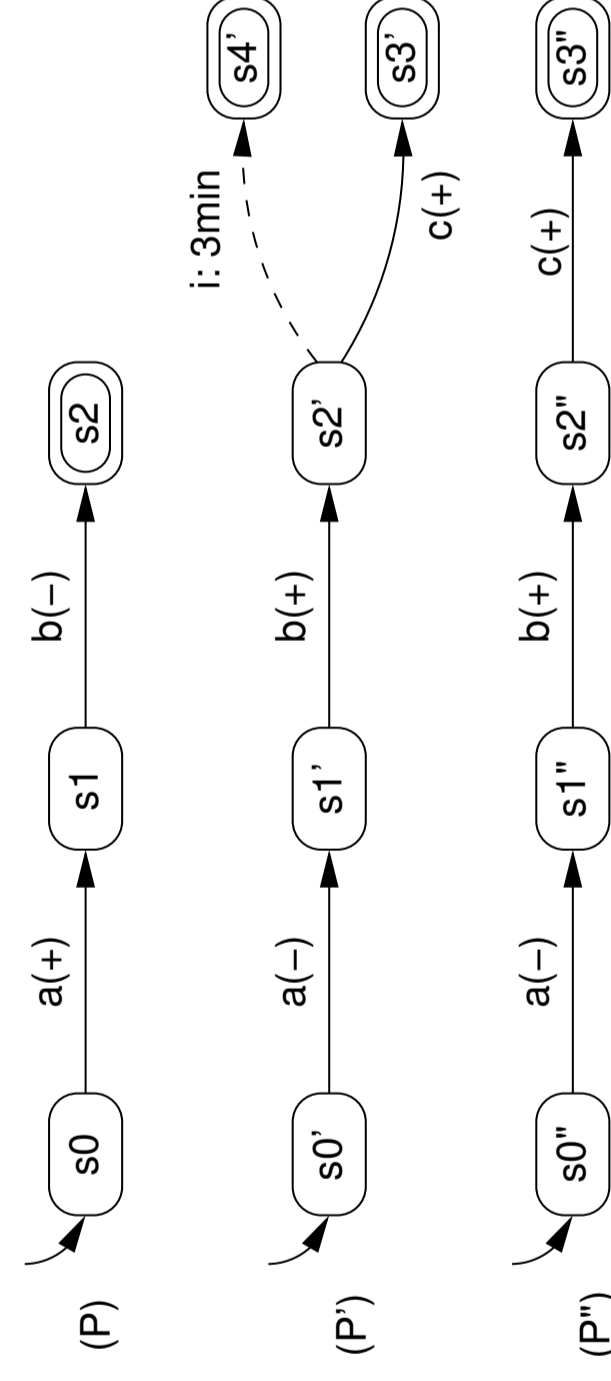


Figure 3: Another compatibility problem illustrated.

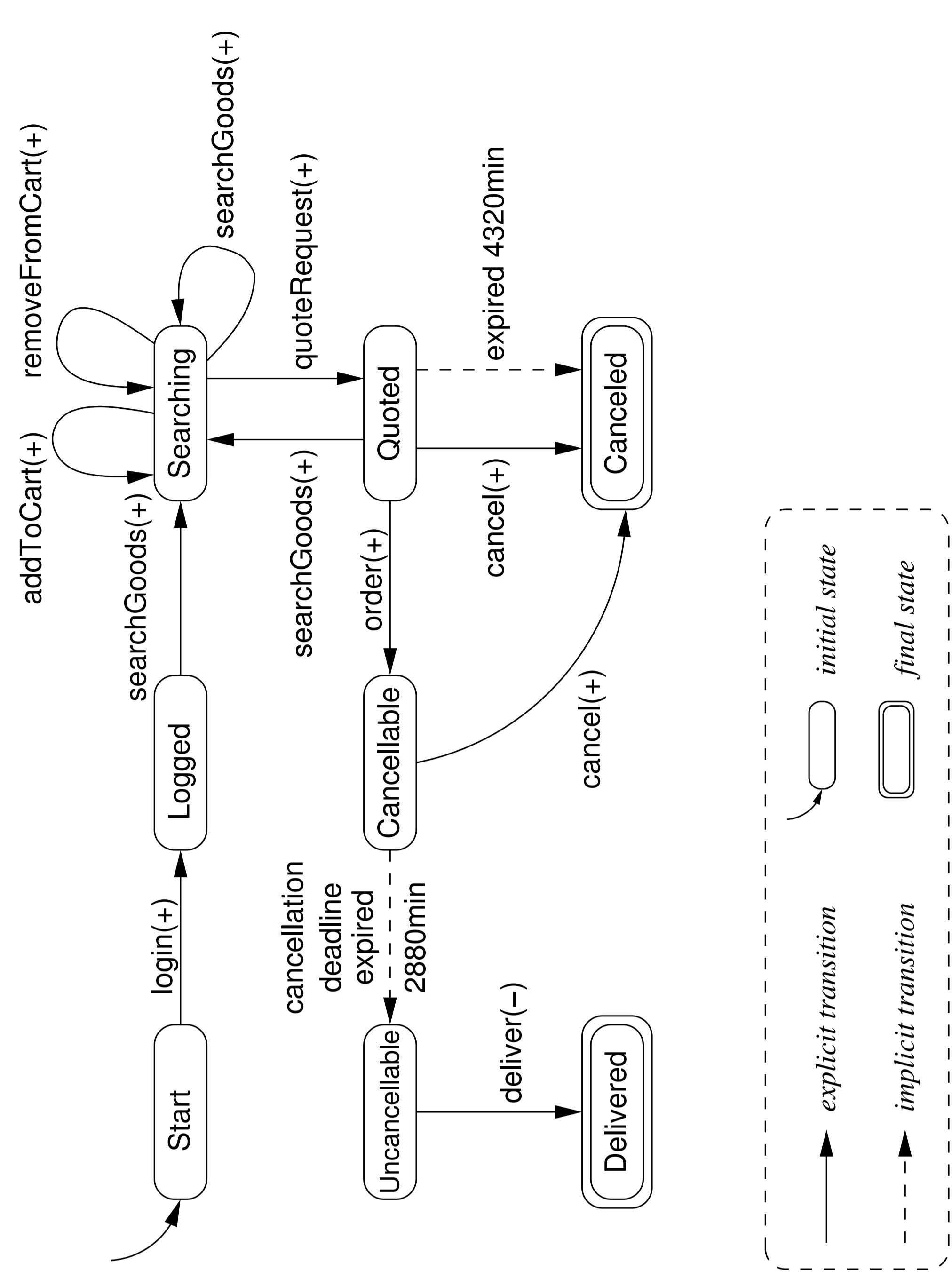


Figure 1: A sample timed business protocol P.

Our approach

- **Compatibility / replaceability classes**:
 - partial timed compatibility / replaceability
 - full timed compatibility / replaceability
 - timed replaceability with respect to a client protocol.
- **Operators** for analysis and management:
 - parallel composition
 - intersection
 - difference.

⇒ **Toward a framework and a CASE tool for protocol analysis and management.**

Figure 4: A protocol P that can replace P'.