ITACA: Integrated Toolbox for the Automatic Adaptation of Web Services

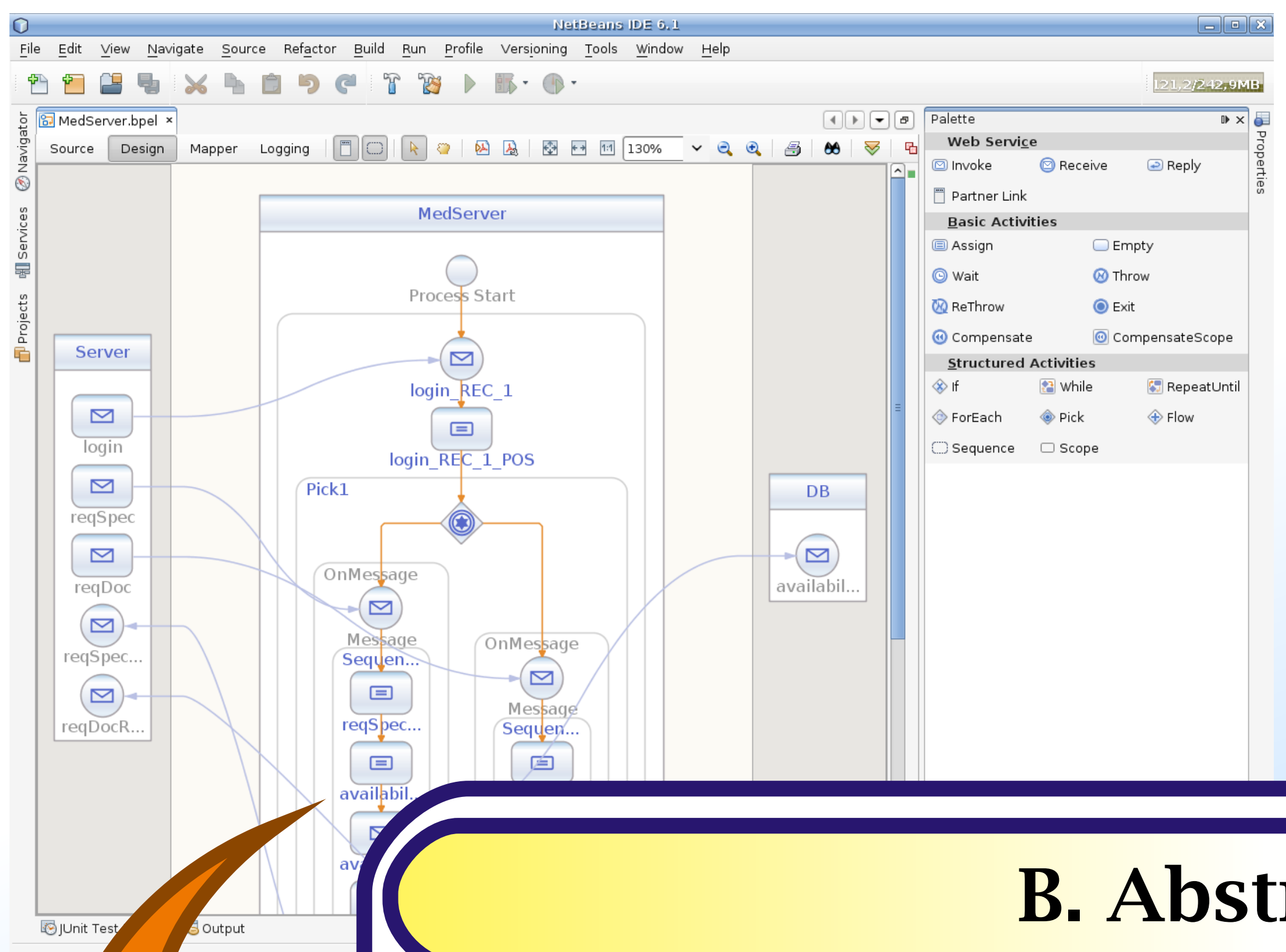
J. Cámara, J. A. Martín, G. Salaün, J. Cubo, M. Ouederni, C. Canal, E. Pimentel
{jcamara,jamartin,salaun,cubo,meriem,canal,ernesto}@lcc.uma.es
Dpto. de Lenguajes y Ciencias de la Computación, Universidad de Málaga, Málaga, Spain

UNIVERSIDAD DE MÁLAGA

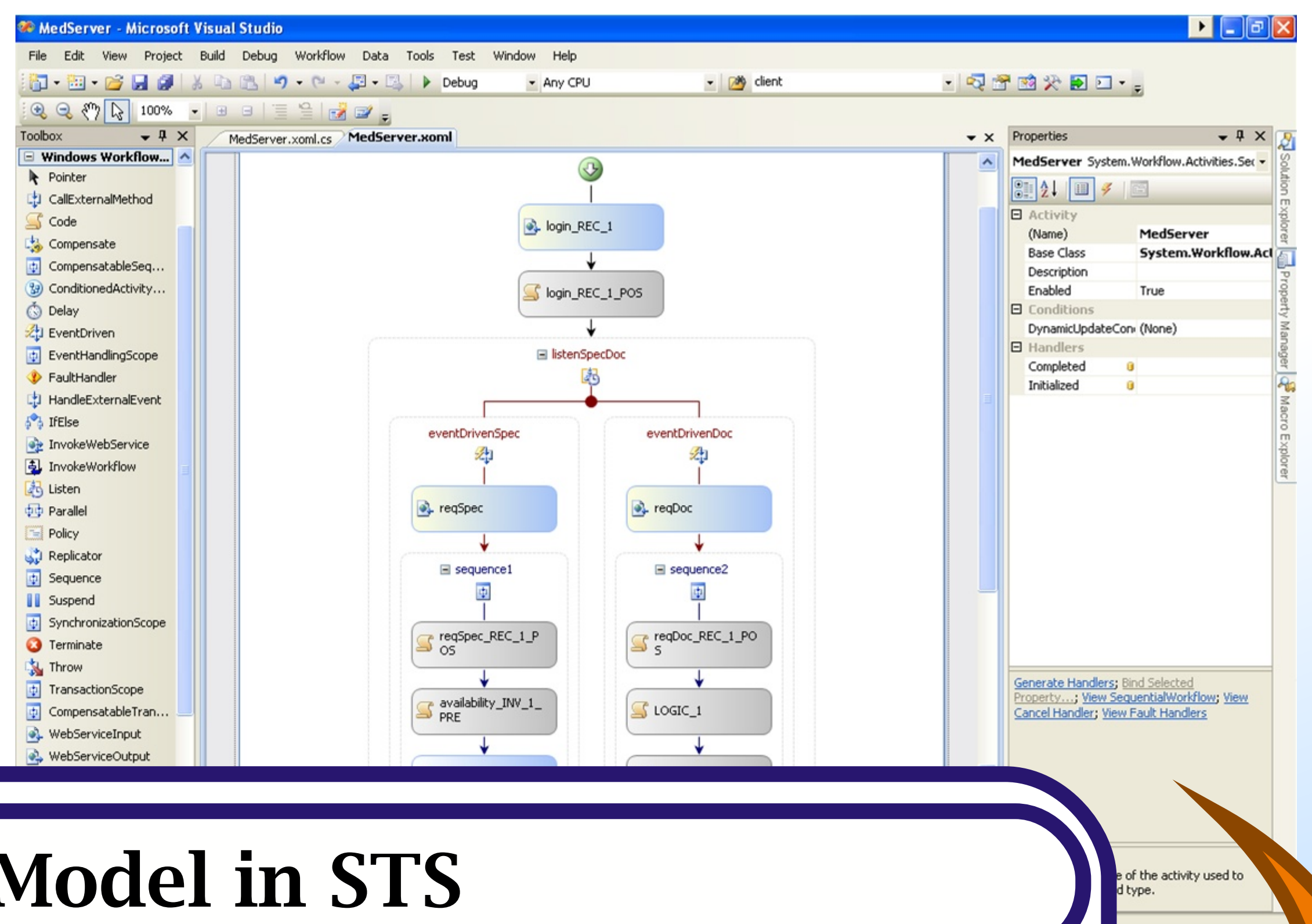
1. Objectives

- Automatic Web Service orchestration and choreography through contract-based adaptation.
- Both signature and behavioral incompatibilities are solved avoiding erroneous executions derived from the order of the messages and parameters exchanged.

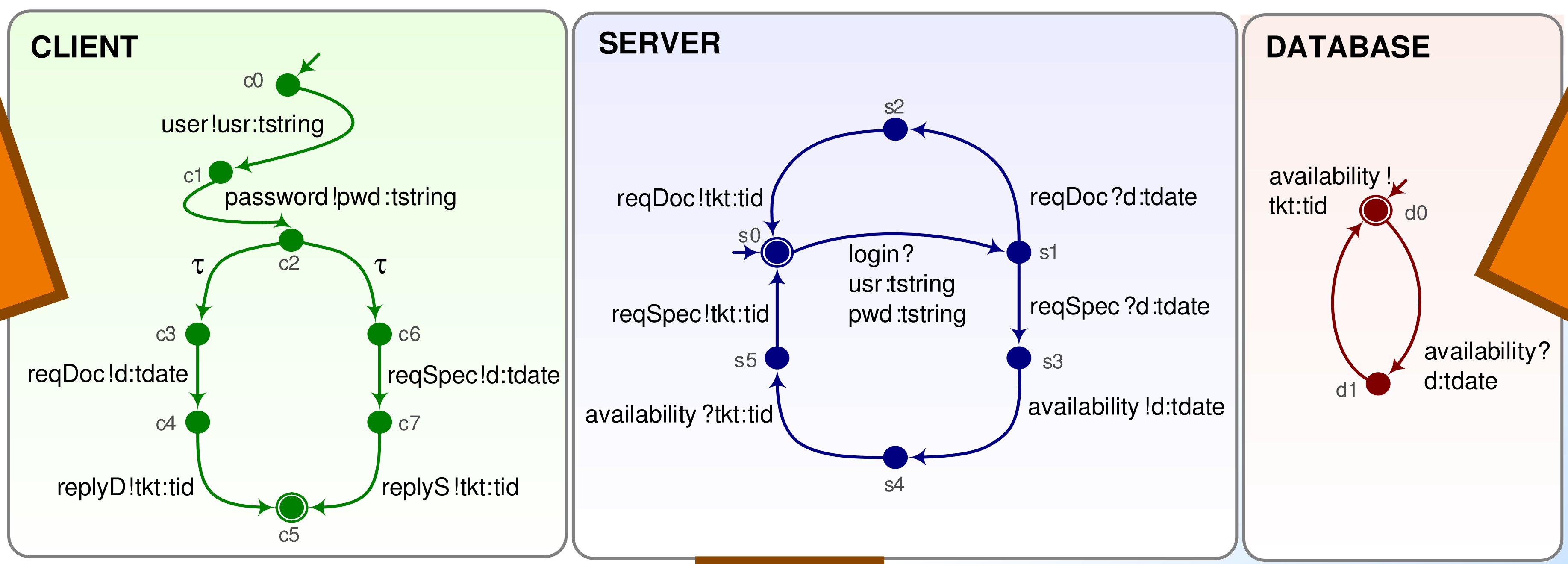
A1. Services in (A)BPEL



A2. Services in Microsoft-WF



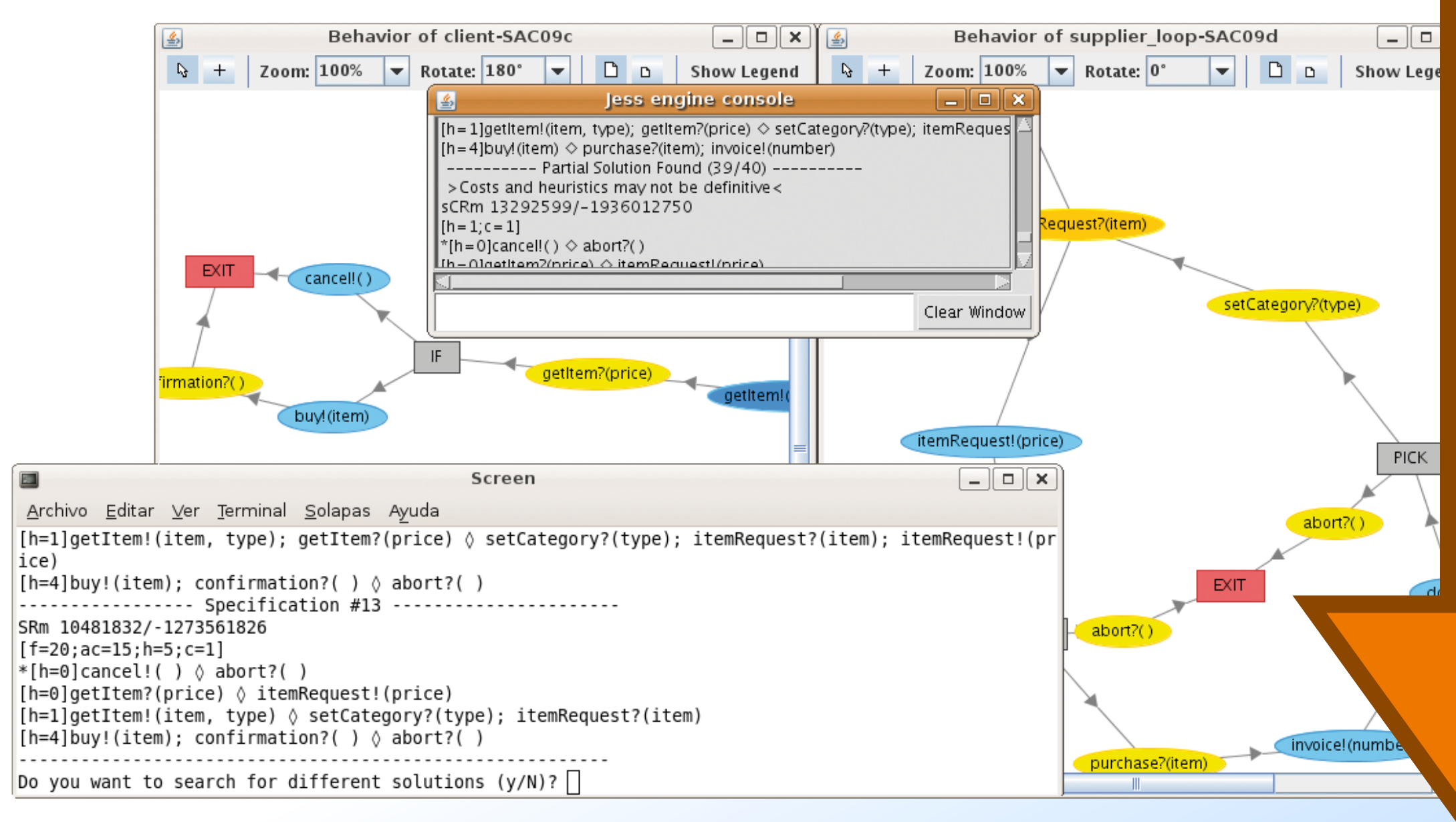
B. Abstract Model in STS



2. Adaptation Contracts

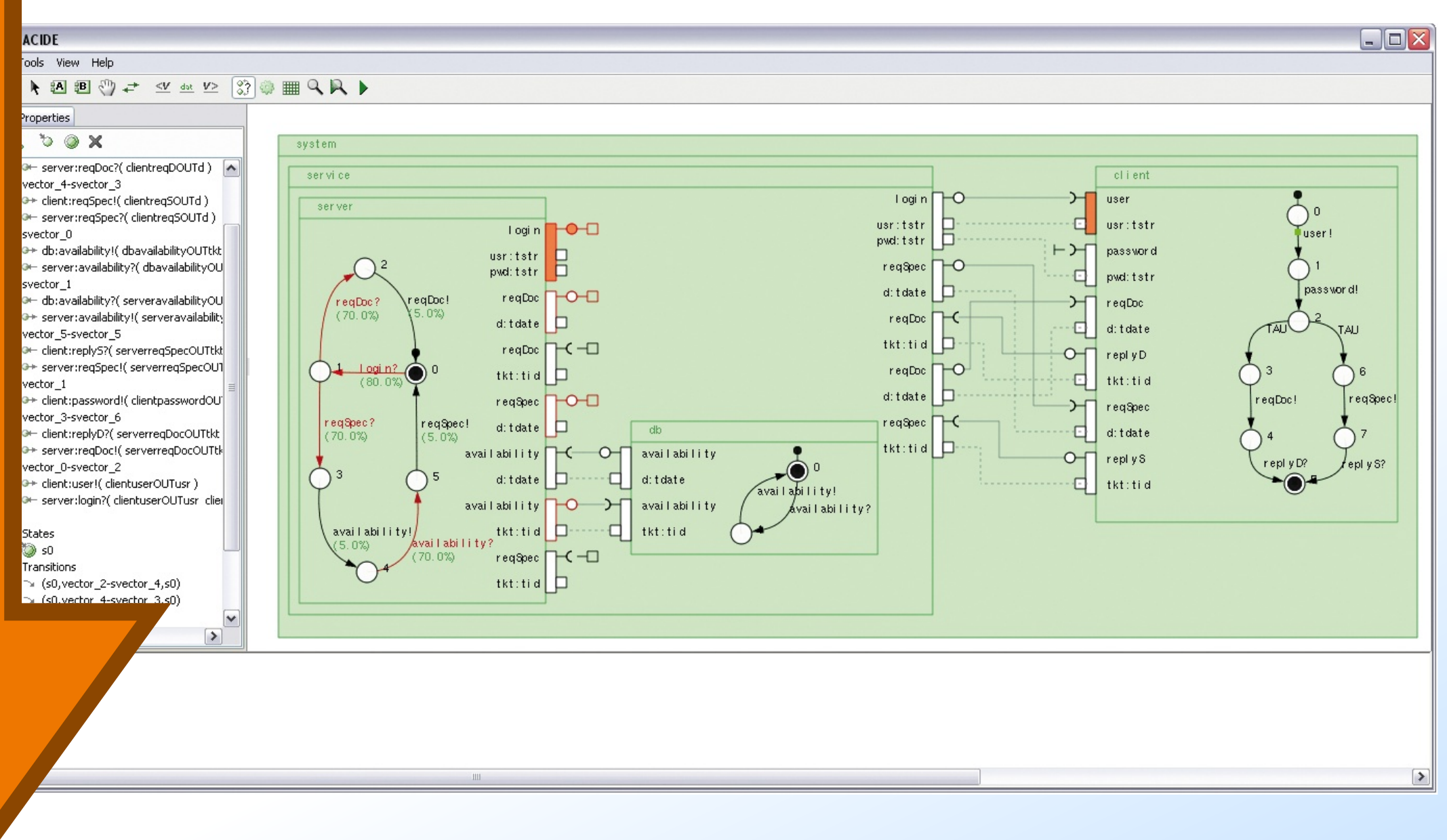
- Web Service interfaces are extracted from WSDL, whereas their protocols are obtained from (A)BPEL or WF code.
- Web Services are modeled as Symbolic Transition Systems (STS).
- Transitions and arguments are mapped across services using an adaptation contract which can either be designed in an interactive environment or automatically generated using similarity metrics.
- Contracts can be validated through interactive simulation and different kinds of trace checks on the choreography/orchestration.

C1. Automatic Contract



```
[h=1]getItem!(item, type); getItem?(price) ◊ setCategory?(type); itemRequest?(item); itemRequest!(pr
ice)
[h=4]buy!(item); confirmation?( ) ◊ abort?( )
---------------- Specification #13 ---------------------
SRm 10481832/-1273561826
[f=20;ac=15;h=5;c=1]
*[h=0]cancel!( ) ◊ abort?( )
[h=0]getItem?(price) ◊ itemRequest!(price)
[h=1]getItem!(item, type) ◊ setCategory?(type); itemRequest?(item)
[h=4]buy!(item); confirmation?( ) ◊ abort?( )
---------------------------------------------------------
Do you want to search for different solutions (y/N)? 
```

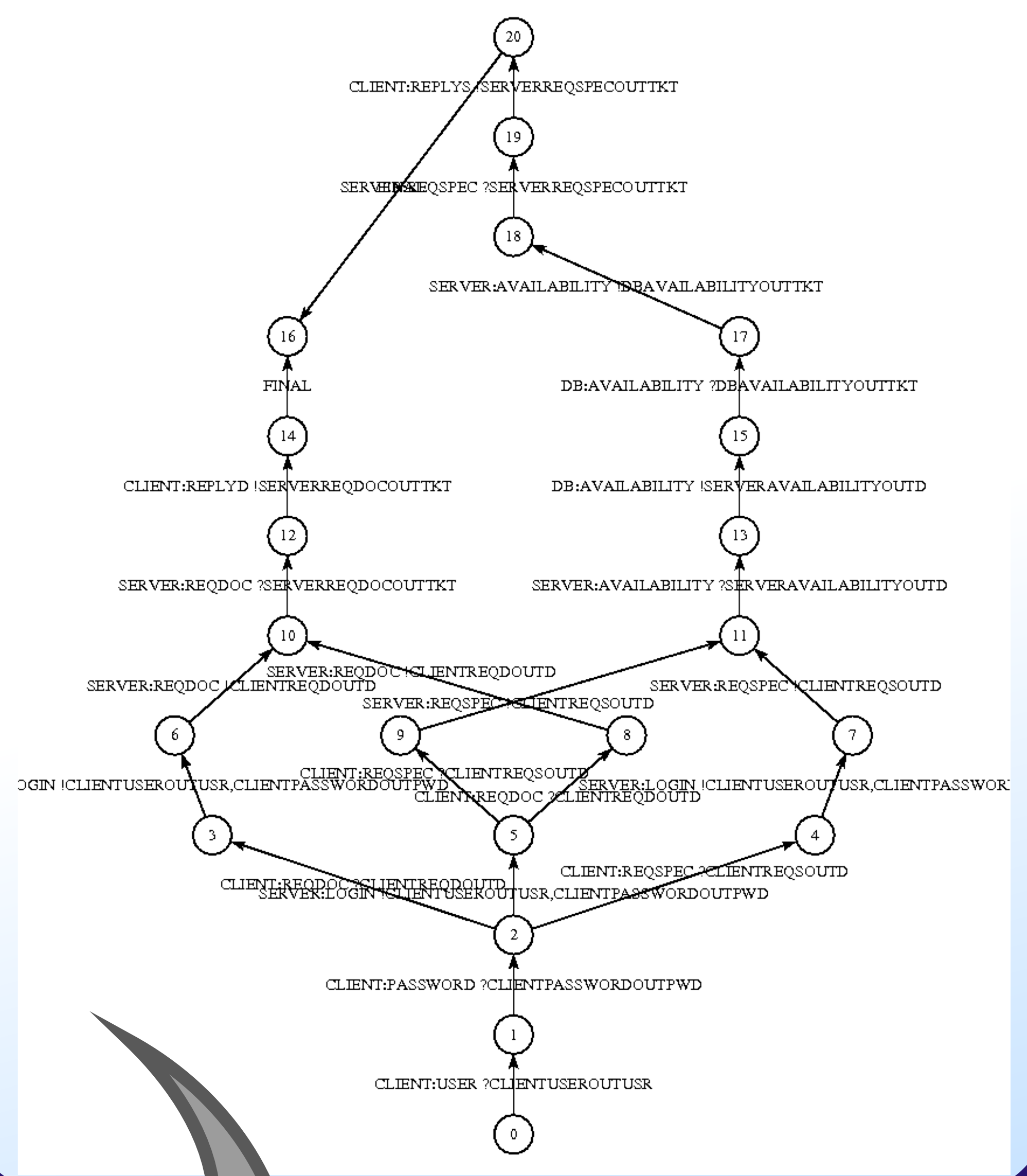
C2. Interactive Contract



3. Adaptor and Wrapper Generation

- Protocols and contracts are encoded in LOTOS where, using CADP and state-of-the-art algorithms, an adaptor protocol is generated.
- The adaptor protocol can be used as monolithic adaptor or be automatically distributed into service wrappers.
- Unfeasible interleavings are pruned in the protocol and the adaptor/wrappers are finally implemented in BPEL.

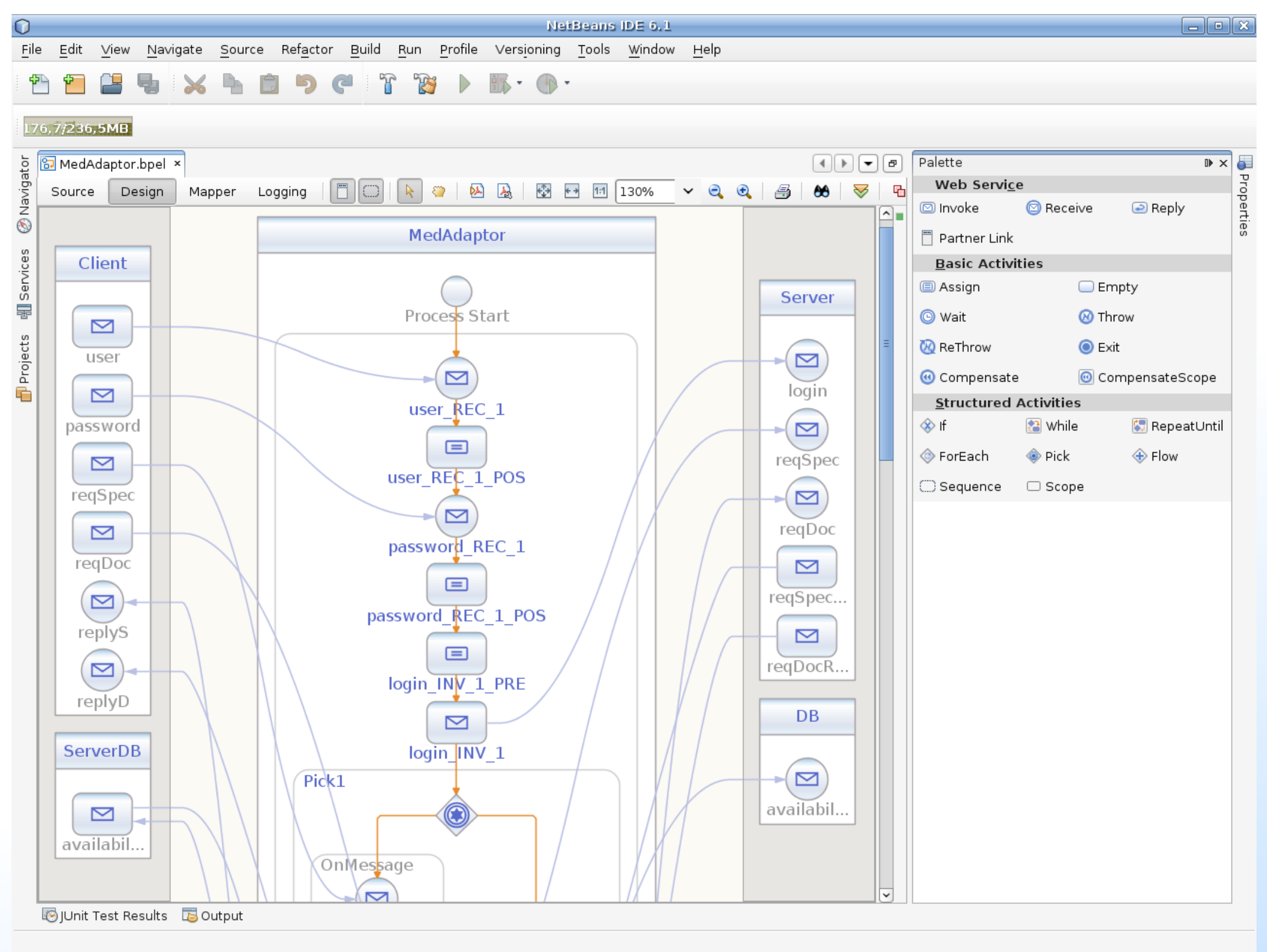
D. Adaptor Protocol



4. Final Remarks

- ITACA is a toolbox that fully supports generative adaptation from beginning to end.
- About 51,000 lines of Python and Java code.
- We plan to extend it with goal-oriented adaptation, system monitoring and self-reconfiguration.

E. Adaptor Implementation



http://itaca.gisum.uma.es

ICSE 2009 VANCOUVER