monadWS: A Monad-Based Testing Tool for Web Services

College of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Yingzhou Zhang, Wei Fu, Changhai Nie

Email: zhangyz@njupt.edu.cn

南京郵電大學
Nanjing University of Posts and Telecommunications

Agenda

- **About *monadWS***
- **Monads Techniques**
- **The monads in *monadWS***
  - **HSP monad**
  - **Gen monad**
  - **TestM monad**
- **Tool Snapshots**
- **Conclusion**

南京郵電大學
Nanjing University of Posts and Telecommunications

About *monadWS*

- An automatic WS testing tool based on monads techniques

- *monadWS* adopts
  - HSP monad[1] for describing WS test cases
  - Gen monad[2] for generating test data automatically,

  and
  - *TestM* monad for building the whole WS testing.

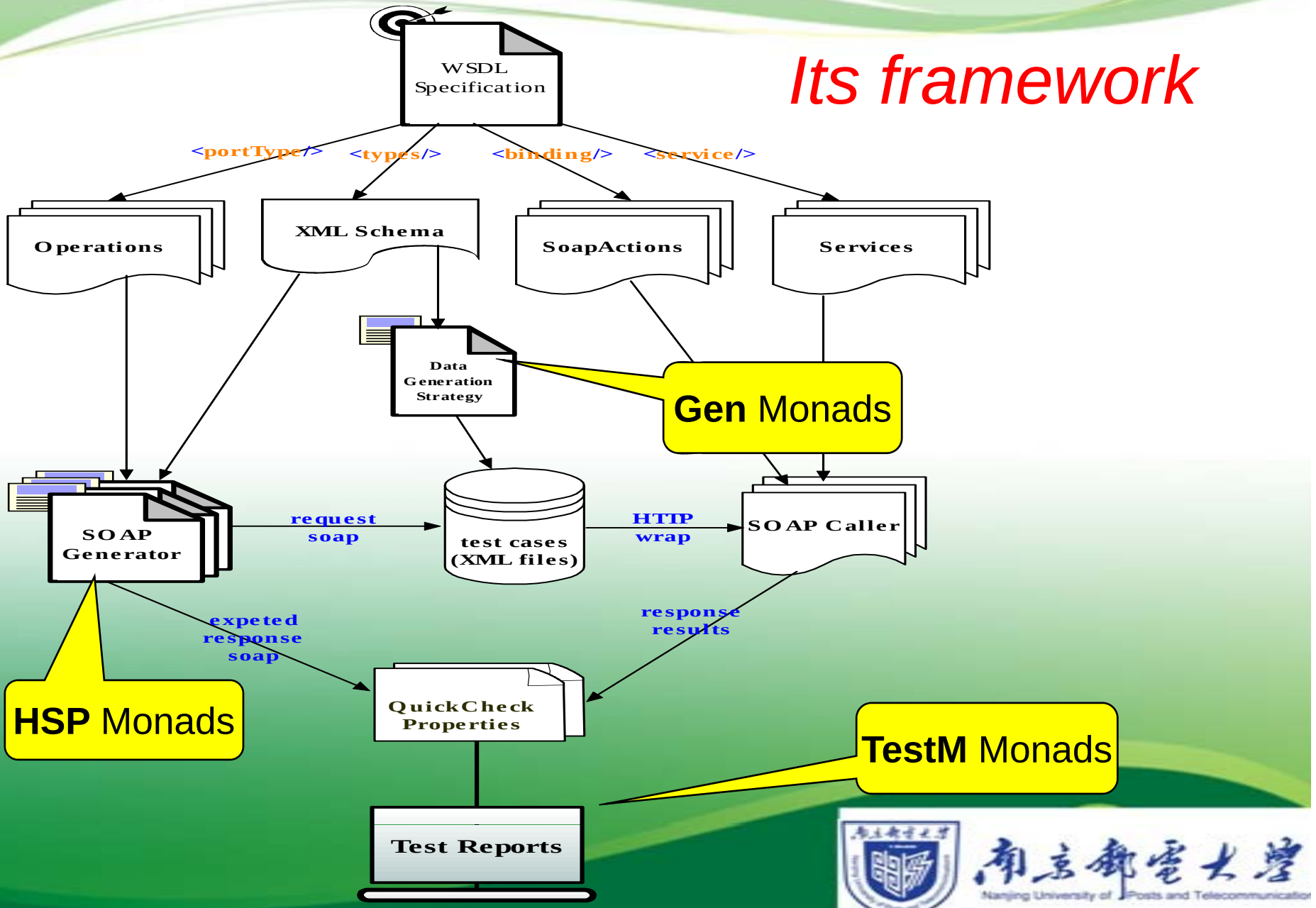
---

[1] HSP (Haskell Server Pages): http://code.google.com/p/hsp/

[2] Claessen K, Hughes J. QuickCheck: a lightweight tool for random testing of Haskell programs. In *Proceedings of the Fifth ACM SIGPLAN international Conference on Functional Programming*, New York, 2005: 268~279.

南京郵電大學
Nanjing University of Posts and Telecommunications

Its framework



Agenda

- About *monadWS*
- **Monads Techniques**
- The monads in *monadWS*
  - HSP monad
  - Gen monad
  - TestM monad
- Tool Snapshots
- Conclusion

南京郵電大學
Nanjing University of Posts and Telecommunications

Monads Intro.

- A general functional notation

  f : a -> b

- Its monadic version of this notion

  f: a -> M b

- Operations on monad M

  return :: a -> M a

  bind :: M a -> (a -> M b) -> M b

Therefore, a monad can be thought as a strategy for combining computations into more complex computations.

Agenda

- About *monadWS*
- Monads Techniques
- The monads in *monadWS*
  - HSP monad
  - Gen monad
  - TestM monad
- Tool Snapshots
- Conclusion

南京郵電大學
Nanjing University of Posts and Telecommunications

HSP Monad: represent TC

- represents a WS test case as a SOAP message
- can encapsulate any side effects, and lets all XML expressions be computed safely

***makeTC_serv* :: InputType -> HSP XML**

The input type of a service operation (say *serv*), such as String, Double …

HSP Monad: represent TC

***makeTC_serv* :: InputType -> HSP XML**

For example:

```
makeTC_getOffesetUTCTime :: Double -> HSP XML
makeTC_getOffesetUTCTime d =
  <soap:Envelope xmlns:soap=soapURI xmlns:xsd=xsdURI>
     <soap:Body>
        <getOffesetUTCTime xmlns=operURL>
             <hoursOffset> <% show d %> </hoursOffset>
        </getOffesetUTCTime>
     </soap:Body>
   </soap:Envelope>
 where
     soapURI ="http://schemas.xmlsoap.org/soap/envelope/"
     xsdURI = "http://www.w3.org/2001/XMLSchema"
     operURL = "http://www.Nanonull.com/TimeService/"
```

HSP Monad: represent TC

```
makeTC_getOffesetUTCTime :: Double -> HSP XML
makeTC_getOffesetUTCTime d =
  <soap:Envelope xmlns:soap=soapURI xmlns:xsd=xsdURI>
     <soap:Body>
        <getOffesetUTCTime xmlns=operURL>
             <hoursOffset> <% show d %> </hoursOffset>
        </getOffesetUTCTime>
     </soap:Body>
   </soap:Envelope>
 where
     soapURI ="http://schemas.xm      p.org/soap/envelope/"
     xsdURI = "http://www.w3.or       L/XMLSchema"
     operURL = "http://www.Nan        om/TimeService/"
```

HSP provides interfaces for other computations through the escapes (with “<%” and “%>”)

Gen Monad: generate data

- automatically generates data for simple types (such as Double, Int, String and so on)

```
class Arbitrary a where
    arbitrary :: Gen a
```

instance Arbitrary String …

instance Arbitrary Double …

南京郵電大學 Nanjing University of Posts and Telecommunications

Gen Monad: generate data

```
class Arbitrary a where
    arbitrary :: Gen a
```

```
makeTC_getOffesetUTCTime :: Double -> HSP XML
```

- ***general testing method*:**
  - Step 1. generate a test value
  - Step 2. generate the final test case with the value

```
testTCGen :: IO String
testTCGen = do
    d <- generate (arbitrary :: Gen Double)
    return $ hsp2str (makeTC getOffesetUTCTime d)
```

Gen Monad: generate data

```
makeTC_getOffesetUTCTime :: Double -> HSP XML
```

- ***general testing method*:**

```
testTCGen :: IO String
testTCGen = do
    d <- generate (arbitrary :: Gen Double)
    return $ hsp2str (makeTC_getOffesetUTCTime d)
```

- ***monadic method by* liftM : on-the-fly generate**

```
testTCGen2 :: IO String
testTCGen2 =
    liftM (hsp2str. makeTC_getOffesetUTCTime) $
          generate (arbitrary :: Gen Double)
```

Gen Monad: generate data

◆ We can easily change the generation strategy through the instances of Arbitrary class.

```
class Arbitrary a where
    arbitrary :: Gen a
```

```
instance Arbitrary Double where
    arbitrary =  myDoubleGen
```

```
myDoubleGen :: Gen Double
myStringGen = suchThat arbitrary guard
  where
    guard a = (a > 0) && (a < 1000)
              || elem a [-1, 0, 9999]
```

TestM Monad: testing as a monad

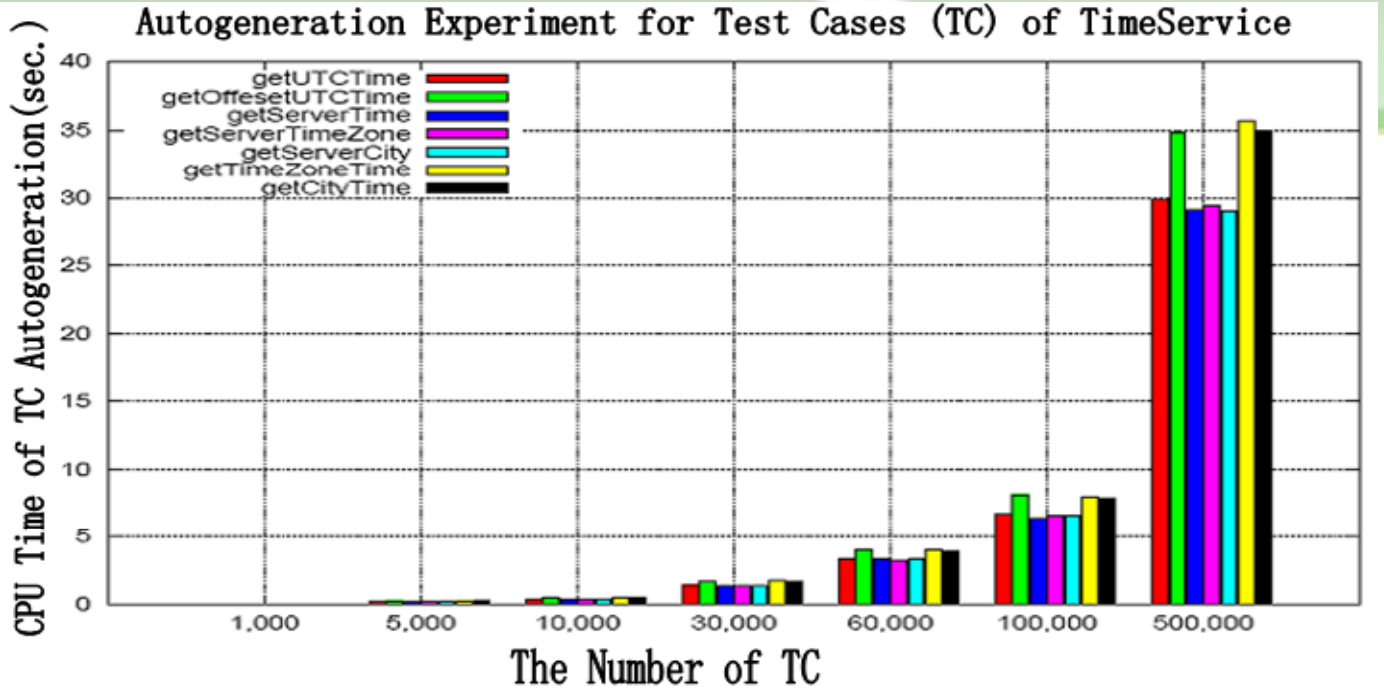
- processes WS testing as the following *TestM* monad:

```
type TestM  a
        = CoAlgMonadT
          (EnvT TestEnv
          (StateT TestState
           HSP
          )) a
```

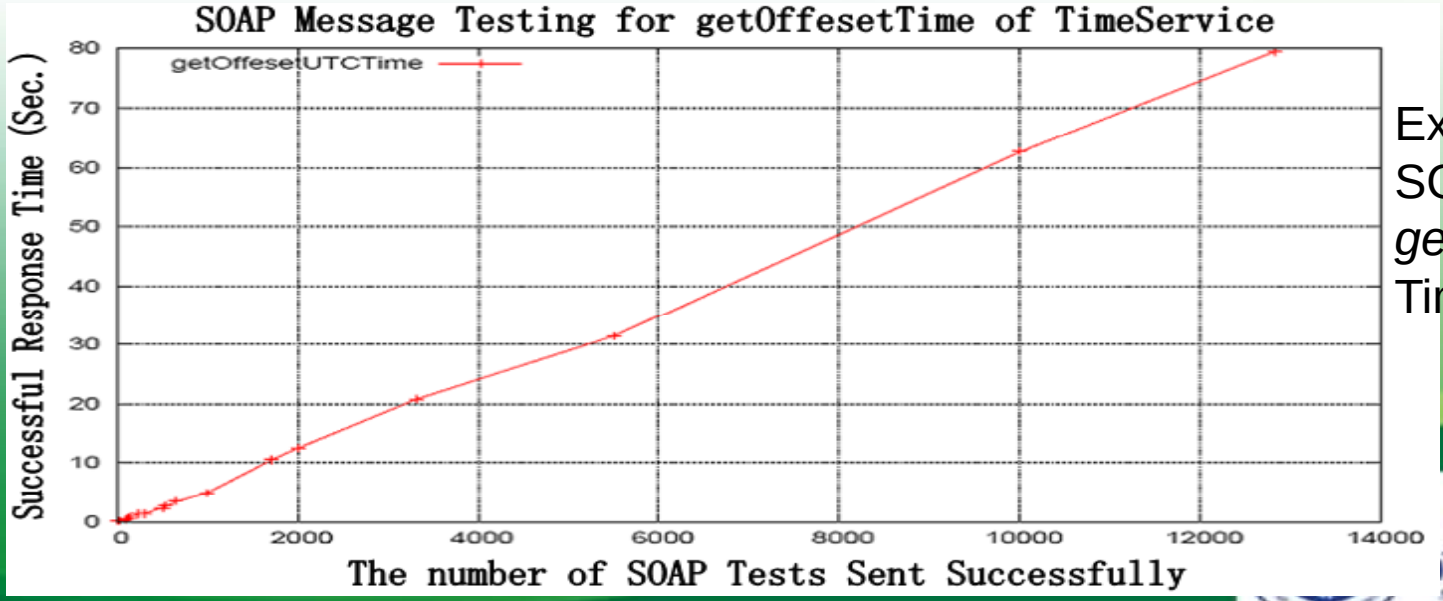
Coalgebraic monad for WS computing

Environment monad for testing environment
(eg. Schema type information, test cases files**)**

State monad for testing states
(**eg.** executing time, testing results)



CPU Time Results of Autogeneration Test Cases for TimeService



Execution Time of the SOAP Messages of *getOffesetUTCTime* in TimeService

南京郵電大學
Nanjing University of Posts and Telecommunications

Comparison of WSTester with Other Test Tools for Web Services

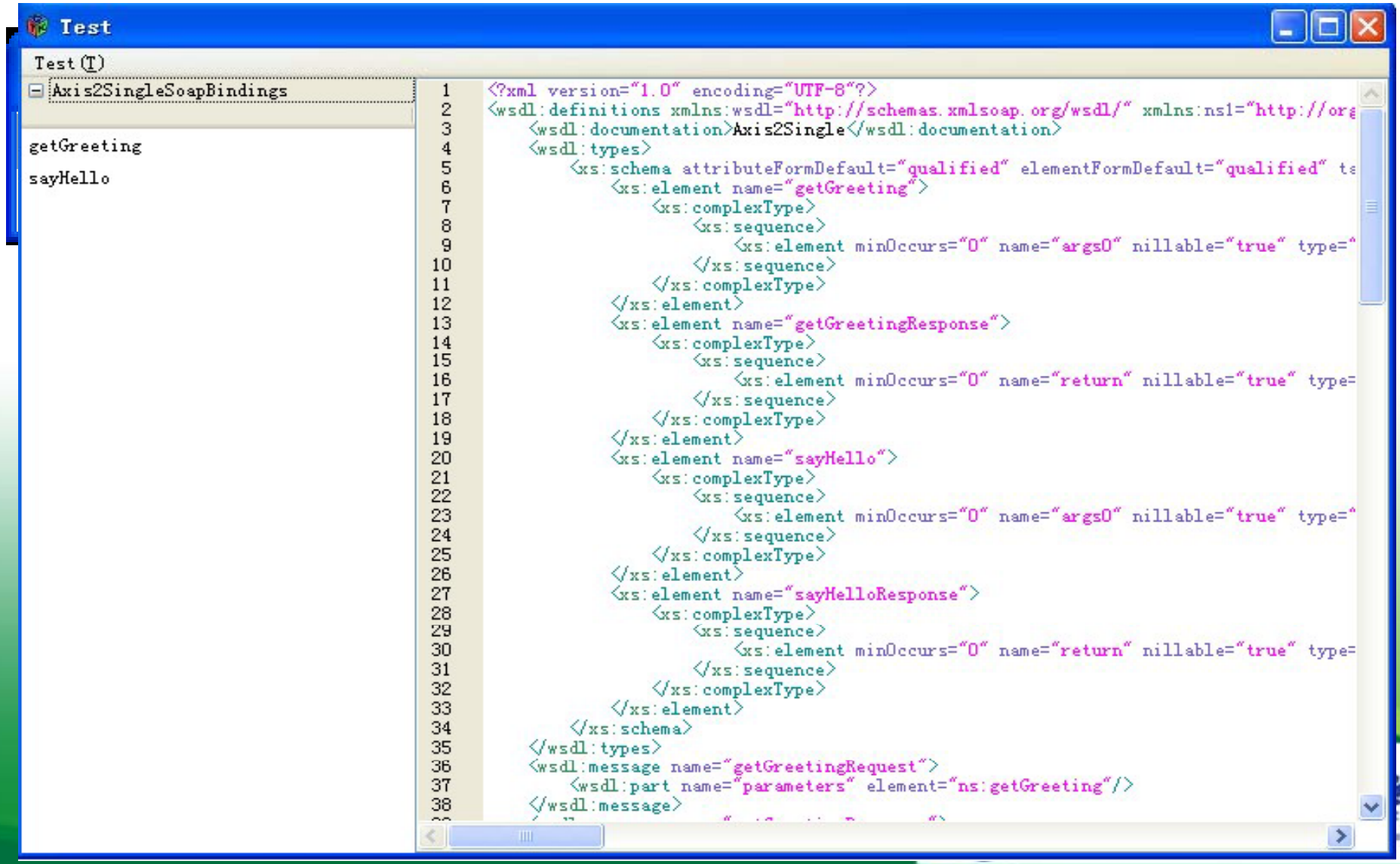
web services		WSTester		soapUI		SoapTest	
websites	oper #	Time (sec.)	fail #	Time (sec.)	fail #	Time (sec.)	fail #
http://www.webservicex.net/country.asmx	10	0.406	0	2.665	0	3.219	2
http://www.webservicex.com/globalweather.asmx	2	0.078	0	1.054	0	0.632	0
http://www.webservicex.net/CurrencyConvertor.asmx	1	0.078	0	0.196	0	0.239	0
http://www.webservicex.com/airport.asmx	4	0.109	0	0.814	0	0.978	0
http://www.webservicex.net/BibleWebservice.asmx	4	0.265	3	44.26	0	1.823	1
http://www.webservicex.net/Statistics.asmx	1	0.125	0	0.976	0	0.312	0
http://www.xignite.com/xcompensation.asmx	10	0.203	0	7.612	0	2.800	0
http://www.webxml.com.cn/WebServices/WeatherWebService.asmx	5	0.172	0	0.745	0	4.297	2
http://www.webservicex.net/sendsmsworld.asmx	1	0.047	0	0.792	0	0.210	0
http://www.webservicex.net/stockquote.asmx	1	0.031	0	0.545	0	0.612	0
http://fy.webxml.com.cn/webservices/EnglishChinese.asmx	6	0.094	0	0.212	0	7.253	3
http://webservice.webxml.com.cn/WebServices/MobileCodeWS.asmx	2	0.047	0	0.149	0	1.484	1
http://webservice.webxml.com.cn/WebServices/WeatherWS.asmx	6	0.140	0	0.147	0	5.320	3
http://webservice.webxml.com.cn/WebServices/StockInfoWS.asmx	1	0.078	0	0.019	0	4.055	0
http://webservice.webxml.com.cn/WebServices/ChinaOpenFundWS.asmx	6	0.328	0	0.107	0	6.301	2
http://webservice.webxml.com.cn/webservices/DomesticAirline.asmx	2	0.109	0	0.523	0	15.05	1

Agenda

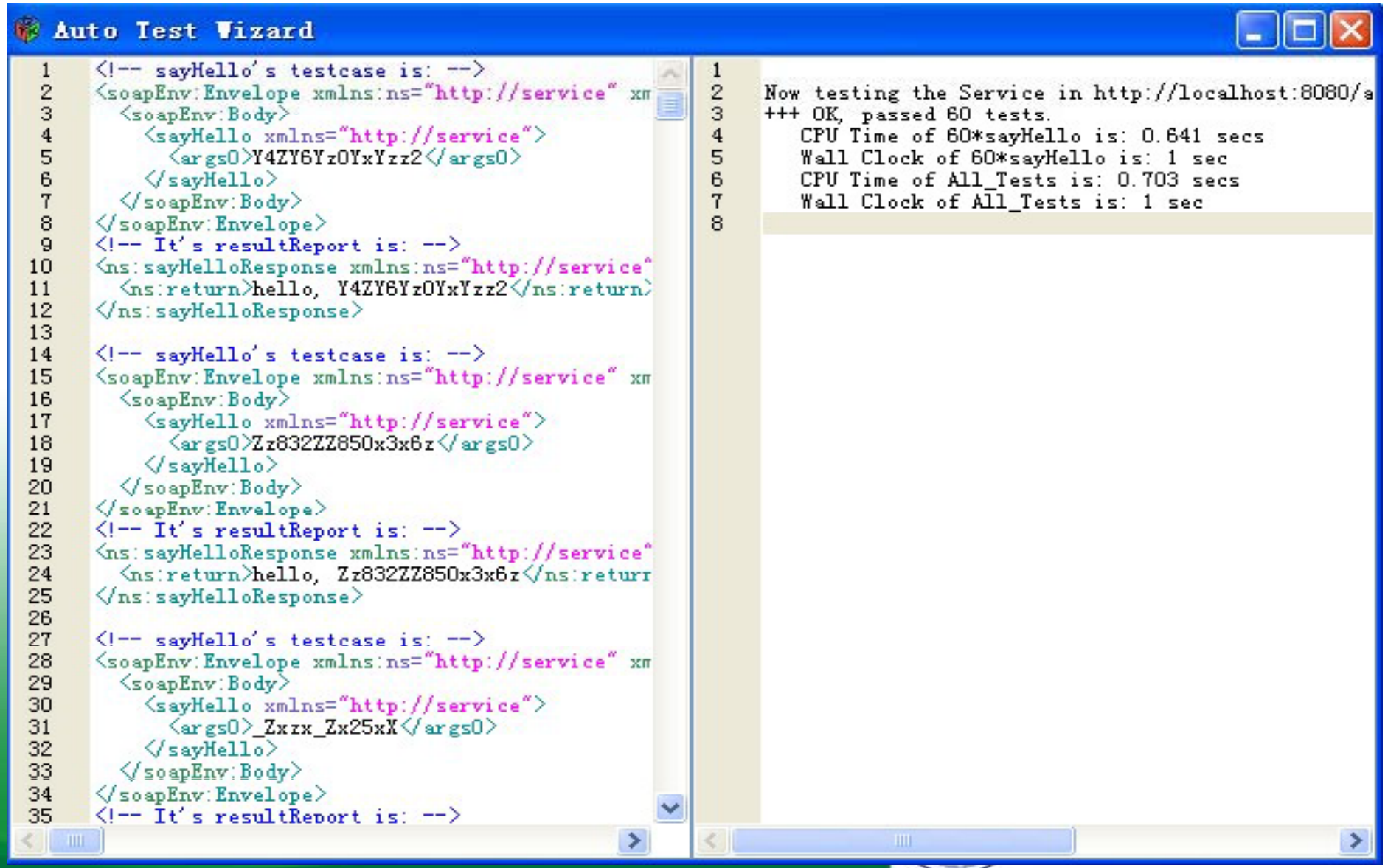
- About *monadWS*
- Monads Techniques
- The monads in *monadWS*
  - HSP monad
  - Gen monad
  - TestM monad
- **Tool Snapshots**
- Conclusion

南京郵電大學
Nanjing University of Posts and Telecommunications

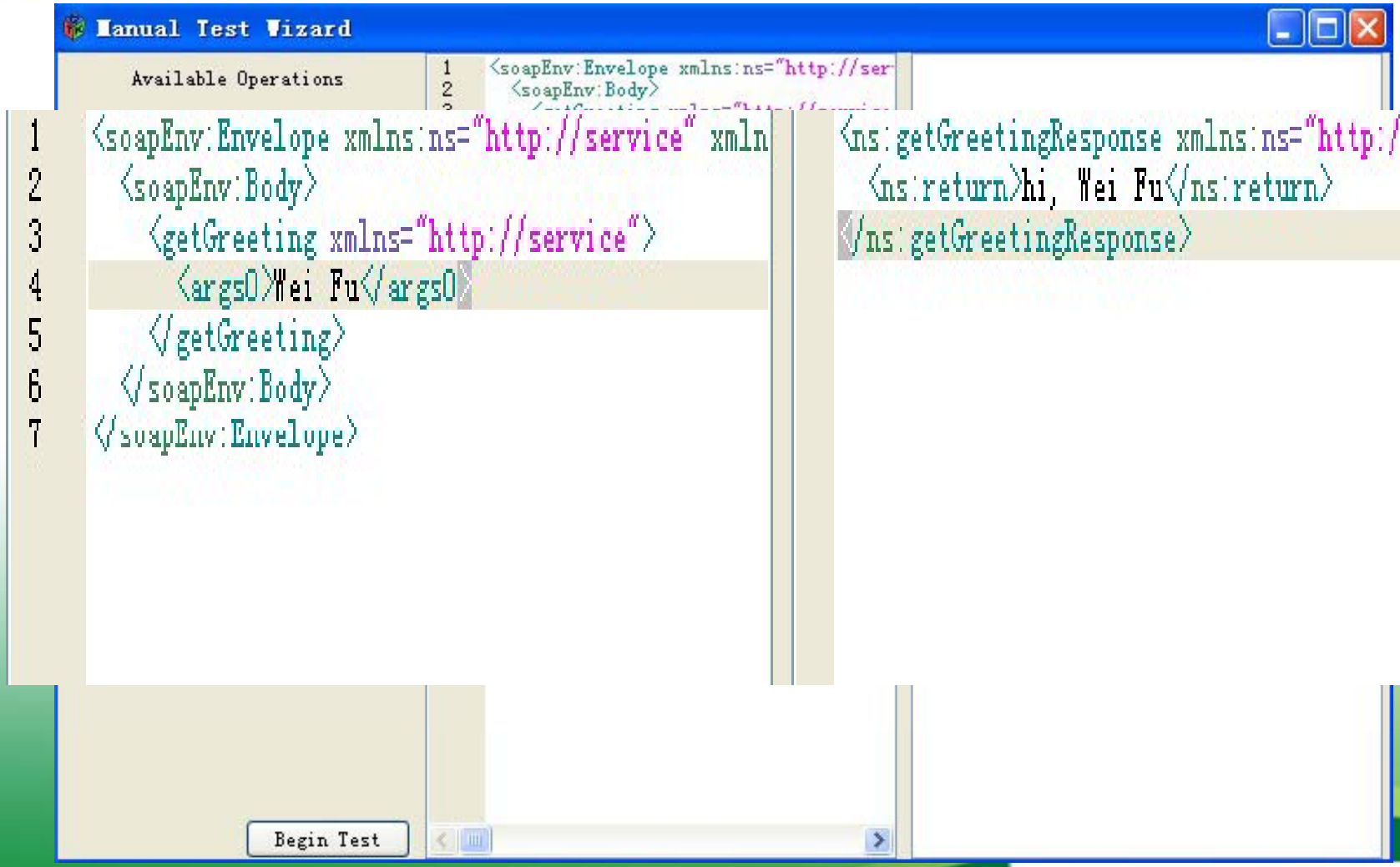
Snapshots of our tool *monadWS*



Snapshots of our tool *monadWS*



Snapshots of our tool *monadWS*



南京郵電大學
Nanjing University of Posts and Telecommunications

Agenda

- About *monadWS*
- Monads Techniques
- The monads in *monadWS*
  - HSP monad
  - Gen monad
  - TestM monad
- Tool Snapshots
- **Conclusion**

南京郵電大學
Nanjing University of Posts and Telecommunications

Conclusion

- **briefly introduce our monad-based testing tool, *monadWS***
- **To automatically test web services.**
- ***monadWS* can use monads related to help**
  - **WS test case representation,**
  - **test data autogeneration, and**
  - **test auto-execution.**

南京郵電大學
Nanjing University of Posts and Telecommunications

Conclusion

- ***monadWS*** **can use monads related to help**
  - **WS test case representation,**
  - **test data autogeneration, and**
  - **test auto-execution.**
- **Future work**
  - **More comparisons with existing tools**
  - **Use more data-generation strategy**
  - **Apply to WS composition (with BPEL)**

南京郵電大學
Nanjing University of Posts and Telecommunications

Thank you!

南京郵電大學

Nanjing University of Posts and Telecommunications