The Future of TTCN-3 in China

Liu Chao & Wu Ji
Software Engineering Institute
School of Computer Science and Engineering
Beihang University, China
Agenda

• Why is the future a problem?
• Testing in China: State-of-art
• Test automation in China: State-of-art
• Overview of TTCN-3
• SEI/BUAA: Research and Practices on TTCN-3
• Opportunities of TTCN-3 in non-telecommunication domains
• Challenges of TTCN-3 in non-telecommunication domains
• Conclusions
Software Engineering Institute (SEI/BUAA)

- 1988-Now

Diagram:
- R & D
- Software Industry
- Education

International Cooperation
Software Testing in SEI/BUAA: Look Back

- **1984-1989:**
  - Software Engineering Environment
    + Software Testing Tools for C: BUAA
  - China-USA Cooperation Project, Sponsored By MOST

- **1995-2002:**
  - BUAA-Lotus Software Quality Engineering Center
    + About 100 Software Testing Engineers
    + 24-hour collaboration around the world (Beijing-Boston-…)

- **2001**
  - Zhong Guan Cun Software Park
    + Beijing Software Testing and Evaluation Center
2002-Now:

- Code based Inspection and Testing
- Software Test Process Management
- Model-Driven Test (MDT)
  + TTCN-3

- BUAA Software Testing and Evaluation Laboratory
  + Software Testing Service
    - Beijing Olympic Game-2008
    - Tests: Kinds of software & information system
TTCN-3 Contributors within China

More …
Why is the future a problem?

- We google from the tooling view:
  - TTCN-3 测试 (ce shi, testing): 4,860 items
  - QTP 测试: 2,050,000 items
  - Loadrunner 测试: 1,850,000 items
  - Rational Robot 测试: 54,500 items
  - Winrunner 测试: 451,000 items
  - Function Tester 测试: 624,000 items
  - SilkTest 测试: 56,100 items
  - TestComplete 测试: 64,000 items

Date when searching by Google: June 8, 2010
Why is the future a problem?

- We google from the language view:
  - TTCN-3 测试语言 (ce shi yu yan, testing language) + 13,900 items
  - VBScript 测试语言: 1,520,000 items
  - JSP 测试语言: 1,010,000 items
  - Shell 测试语言: 3,670,000 items
  - Ruby 测试语言: 2,530,000 items
  - Python 测试语言: 4,480,000 items
  - Perl 测试语言: 4,640,000 items
  - TCL 测试语言: 1,900,000 items
Software GDP Distribution

Data source: MIIT, 2010, 01
China Software Parks

中西部IT产业集群
成都 (电子产业)
重庆 (通信设备)
武汉 (光电子产业)
长沙 (软件)
西安 (光通讯、软件)

集成电路产业和软件产业发展迅速，拥有光宝电子、南方高科和金鹏集团等一批高水的骨干企业

集成电路产业规模全国第三，设计、专用材料和制造方面处于技术领先地位

中国最大的电子元器件生产基地和移动通信手机三大基地之一

环渤海地区IT产业集群
北京 (集成电路)
天津 (电子元器件和移动通信手机)
大连 (软件业)
青岛 (电子家电)

中国唯一的“软件产业国际化示范城市”、“国家软件产业基地”

中国内地规模最大、最完善的IT设备制造基地

国家级IC设计产业基地
国家级软件基地和全国集成电路芯片产业的龙头

长江三角洲IT产业集群
上海 (IC设计)
杭州 (IC制造、通信制造)
苏州 (IT设备制造)

珠江三角洲IT产业集群
广州 (软件)
深圳 (通信、微电子)
东莞 (电脑资讯)

形成了以通信、计算机、微电子为代表，较为完整的电子信息产业配套体系

电脑整机零部件配套率达95%：电脑磁头、电脑主机板和电脑显示器等产品产量均位居世界前列
Testing in China: State-of-art

- **Strong support from Government**
  - Since 2001, government asked all software products MUST be registered with the evidence of testing before ship into the market
  - Since 2003, government pushes the profession certificate expanding to software testing engineer, “Software Testing Profession”

- **Continuous growing of ‘IV&V Service Providers’**
  - Almost every software industry park has at least one such provider
  - CCID

- **The software testing master degree is popularly set in software schools at 35 universities in China since 2004**
Testing in China: State-of-art

- 51testing.com made a survey in 2008

Test-Domain distribution

Sources: www.51testing.com
Testing in China: State-of-art

The ratio of having Testing Branch

- 51% Have Technique Support Branch
- 38% Have Tool R&D Branch
- 11% No such branch

The Independence of the Testing Branch

- 46% Under Quality Branch
- 35% Under R&D Branch
- 19% Independent branch

Sources: www.51testing.com
Testing in China: State-of-art

Number of Full-time Test Engineers

- A: None
- B: 1-10
- C: 10-50
- D: 50-100
- E: 100-500
- F: 500-1000
- G: 1000-2000
- H: 2000 or more

Levels of Test Process Strictness

- A: Very poor
- B: Poor
- C: Fair
- D: Good
- E: Very good
Testing in China: State-of-art

Testing Skills Expected When Hiring
- Others
- Test Analysis
- Performance Test
- Whitebox Test
- Automated Functional Test
- Test Management
- Test Case Design
- Manual Functional Test

Language Skills Required When Hiring
- Others
- Not Specified
- Delphi
- Visual Basic
- C#
- Java
- C/C++
Testing automation in China

Currently used Automation Tools

- A: HP Mercury Quicktest Pro (38%)
- B: HP Mercury Winrunner (10%)
- C: IBM Rational Robot (9%)
- D: IBM Function Tester (10%)
- E: Borland Silktest (9%)
- F: AutomatedQA TestComplete (31%)
- G: Watir (4%)
- H: Others

Currently used Performance Testing Tools

- A: HP Mercury Loadrunner (51%)
- B: IBM Rational Robot (6%)
- C: IBM Performance Tester (6%)
- D: Borland SilkPerformer (5%)
- E: Radview Webload (30%)
- F: Parasoft Webking (2%)
- G: Jmeter (3%)
- H: OpenSTA (2%)
- I: Others
Testing automation in China

**Script Language used**

- Others: 31%
- VBScript: 36%
- JSP: 22%
- Shell: 14%
- Ruby: 3%
- Python: 5%
- Perl: 8%
- TCL: 9%

**Framework used in Unit Testing**

- Other Script Lang: 23%
- Other: 35%
- SQL Unit: 8%
- JUnit: 27%
- CUnit: 6%
- CppUnit: 7%
Evolving Techniques over 10 Years

- Model-based design
- MDA
- Web application (web service)
- Rich Web (web 2.0)
- Telecommunication
- Virtualization
- Cloud computing
Overview of TTCN-3

• The standardised (black-box) test specification and test implementation language.
• Developed
  – based on the experiences from previous TTCN versions.
• Standards
  – ES 201 873-1 (Z.140): TTCN-3 Core Language
  – ES 201 873-2 (Z.141): TTCN-3 Tabular Presentation Format (TFT)
  – ES 201 873-3 (Z.142): TTCN-3 Graphical Presentation Format (GFT)
  – ES 201 873-4 (Z.143): TTCN-3 Operational Semantics
  – ES 201 873-5: TTCN-3 Runtime Interface (TRI)
  – ES 201 873-6: TTCN-3 Control Interfaces (TCI)
  – ES 201 873-7: import ASN.1, XML, IDL, C/C++ to TTCN-3
  – …
testcase myTestcase () runs on MTCType system TSIType
mydefault := activate (OtherwiseFail);
verdict.set(pass);
connect(PTC_ISAP1:CP_ISAP1, mtc:CP_ISAP1);
map(PTC_ISAP1:ISAP1, system:TSI_ISAP1);
PTC_ISAP1.start(func_PTC_ISAP1());
PTC_MSAP2.start(func_PTC_MSAP2());
Synchronization();
all component.done;
log(.Correct Termination.);
TTCN-3 Language

• It is the standardized test language
  – Not tied to a particular application or its interface(s)
  – Not tied to any specific test execution environment, compiler or operation system

• Powerful language features
  – Rich type system and namespace
  – Template matching mechanism
  – Snapshot semantics on event queue
  – Concept of verdict
  – Concurrent test behaviour
  – Structured decomposition (data and behavior)
  – …
Implementation of TTCN-3 System

Abstract Test Suite

Compile

TE

TTCN-3 Executable

Build

SUT

Executable Test Suite

TTCN-3 Runtime System
TTCN-3 Applications

• Applicable for all kinds of black-box testing for reactive and distributed systems
  – Telecom systems (ISDN, ATM);
  – Mobile (telecom) systems (GSM, UMTS, 3G, LTE);
  – Internet (has been applied to IPv6, SIP, Wimax);
  – CORBA based systems;
  – Java, XML, ...
• Wider scope of application
  – not just conformance, also for development, system, integration, interoperability, scalability... testing
  – applicable in the telecom and datacom domain
  – used both for standardized test suites...
TTCN-3 Activities in Beihang Uni.

- **Researches**
  - Model driven testing
  - Test distribution
  - Robustness testing
  - Supported by NSF, MOST, MIIT, etc.

- **Engineering Projects**
  - Banking system testing in TTCN-3
  - Operating system interoperability testing in TTCN-3
  - Large scale web app. Testing in TTCN-3
  - Web-based Office interface testing in TTCN-3
MDT Methodology Framework

Diagram:

- Test Req.
- Test Design
- Test Behavior
- Iteration
- Test System Model
- Test Scheduling
- Test Result
- Test Adaptation
- Adapters & Deployment

Flow:

1. Test Req. → Test Design → Test Behavior → Iteration
2. Iteration → Test System Model
3. Test System Model → Test Scheduling
4. Test Scheduling → Test Result
5. Test Result → Test Adaptation
6. Test Adaptation → Adapters & Deployment
7. Adapters & Deployment → Test Req.
Model-driven Testing (cont.)

- Test system model language
  - Meta-model definition based on U2TP
- Transformation is the key
  - Stepwise transformation from Test Req to executable test system
- Test iteration
  - Adjust test requirement and test design strategy
MDT Methodology Framework

- **Test Req.**
  - SUT Model + Testing concerns
- **Test Behavior**
  - Test data, test case, test component
- **Adapters & Deployment**
  - Test adapters, test component deployments
- **Test results**
  - Test verdict, test trace
Test System Dev. Metaphor

Program System

Req
UML
Design
Java

Test System

Req
U2TP
Design
TTCN-3

Code
Code

Req
UML
Design
Java

Code
Code

U2TP
TTCN-3
Test System Dev. Platform

- Focus on domain specific research based on the essential MDT features
  - Modeling platform
  - Runtime platform
Visualized Modeling
Test Distribution

- Test Mgmt.
  - Test Campaign
  - Deploy & Schedule
  - Test Analysis

- SUT Specific Plugins
  - Web (CD + SA)
  - Service (CD + SA)
  - Embedded (CD + SA)

- Test Adapter Framework
  - CD Mgmt.
  - SA Mgmt.
  - <SA, SUT> Comm. Mgmt

- Configurable TTCN-3 Runtime

- Test Node Mgmt.
  - Communication Service
    - Test Deployment
    - Test Control
    - Test Result Collection
Robustness Testing

- Apply model driven methodology on robustness testing
  - Focus on abnormal messages
- Provide visualized fault model designer based on sequence model
- Tester can inject typical message faults into the communication scenario
  - Message loss
  - Message re-order
  - Message delay
  - Message content falsify
Robustness Testing

- Fault Model Designer & Visualized Result Viewer
Engineering in TTCN-3

- Banking system testing in TTCN-3
  - Supported by MOST 863 Projects
  - Co-worked with Shandong Computer Science Center
- OS interoperability testing in TTCN-3
  - Supported by MIIT Key Project
  - Co-worked with the Software Test Center at China Electronics Standardization Institute
- Large scale web app. Testing in TTCN-3
- Web-based Office testing in TTCN-3
  - Supported by MOST 863 Projects
Opportunities of TTCN-3 in non-telecommunication domains

• Why only non-telecom domains?
  – Simply because TTCN-3 already takes the telecom!
  – Non-telecom is opening its bigger market in China
• Typical non-telecommunication domains in China
  – Avionics Electronics: big plan pushed by government
  – Automobile Electronics: very active
  – Web applications (service): very popular
Aviation and Auto

- Embedded
- Net-centric
  - Bus: CAN or 1553B
  - Protocol: TCP/IP
- Strict real-time behavior
- High dependability requirement
- Model-based development
Opportunities

• For embedded software, TTCN-3 show be outperformed in
  – Testing the standardized interface & protocol
  – Testing the real time behavior
  – Supporting multiple platforms efficiently
  – Integrating with testing device
  – Importing data types in C/C++
Web Applications (Service)

- Rapid development
- Protocol intensive
- Performance sensitive
- Quick evolving of techniques
  - Web 2.0, HTML 5, AJAX, Javascript, cloud,…
- Rapid requirement evolving
Opportunities

• For web application, TTCN-3 should be outperformed in
  – Testing the protocols
  – Testing the workflows/collaborations among distributed modules
  – Testing SOA or cloud computing
Challenges (1)

• TE can be easily handled, while **adapters** (SA and CD) development are tightly coupled with the platform techniques
  – Requires wealth of platform knowledge & experience, even tricks
  – Most testers in China will have difficulties
• For example, web application testing
  – QTP enables direct data object + html tag access and evaluation
  – The CD development has to consider the data schema, the local script like java script, ajax, etc…
Challenges (2)

- Models are massively involved in developing aviation and auto applications
- If TE+SA+CD can be automatically (or partially) generated from the models, TTCN-3 will have more chances to win
- But, how to generate tests from the heterogeneous models?
  - TE generation: maybe coverage based
  - CD+SA generation: ?
  - Correlation btw TE and CD+SA: ?
Challenges (3)

- There still do not have a methodology and tools on the development of TTCN-3 test system.
  - Requirement
  - Design
  - Coding
  - Deployment
  - Optimization
Challenges (4)

- No technique meets all the testing requirements.
- TTCN-3 system needs also to collaborate with other system and techniques.
  - SUT artifacts
    + import not only data types, but the platform knowledge
  - Tools to work with
    + XML-based integration and interoperability
Challenges (5)

• Some SUT input is implicit, like database or configuration file.
  – How to make such input explicitly?
  – How to make the implicit data input consistent with the input selected in TTCN-3 code?
Challenges (6)

- On performance testing
  - How to generate the required load easily and effectively?
  - How to collect the massive logs efficiently?
Challenges (7)

- GUI is the key interface for black-box testing.
- GUI data objects are platform dependent.
  - How to deal with it?
Challenges (8)

- TTCN-3 solution is labor-intensive compared with QTP or Robot
- How to provide smart solution by quickly developing test data and test evaluation (Verdicts)?
Challenges (9)

• How to manage the evolving of TTCN-3 languages?
  – Testers have to keep learning, …
Challenges (10)

- Should we maintain the pureness of TTCN-3 as a general language, or provide calibrated domain specific x-TTCN-3 when dealing with those challenges?
China Specific Challenges

- **The cost**
  - Support tools and services
  - Training
  - Solution cost compared with manual testing

- **Localization**
  - Need success stories in non-telecommunication domains
  - Support tools and services
  - Tools from local vendors

- **Sell to the IV&V testing service providers**
  - Not only business behavior, need government support
  - Professional test development education
Conclusions

- TTCN-3 is successfully adopted in telecom and datacom domains
- Testing in China is still in its developing stage
  - Strong need of test automation
  - Non-telecommunication domains open its big market
  - But, TTCN-3 is not well known
- There are opportunities
  - In Web and Embedded application domains
  - Government encourage standardized solution
- Need more collaborations among
  - Academies
  - Tool vendors
  - Users
Thanks!

谢谢！
• Any Questions?