

TDL FOR TESTING COLLABORATION IT SERVICES: THE NETRESULTS EXPERIENCE

Presented by Francesco Oppedisano



ETS

© All rights reserved





NetResults presentation

- Born in 2006 from the research group in telecommunication networks at University of Pisa (Italy)
- Official spin-off of Pisa's University
- SME with strong skills in VoIP/MoIP and network performance testing
 - People: 20
 - with degree: 16/20
 - with PHDs: 3/16

User Conference on Advanced Automated Testing







The REBEC System

- REBEC stands for REALTIME E_LESSON BROADCASTING ENGINE FOR CITTAEDUCANTE
- It is a distance learning tool developed for «La Città Educante» project.
- «La Città Educante» is a project cofinanced by MIUR (Italian Ministry of Research and University), which aims to define an infrastructure to make a city a «long life learning place»
- Our job in the project is to make the learning experience for remote students of the city as effective as possibile.
- REBEC currently can stream a whiteboard by means of a lossless codec and audio via a VoiP Network
- Our goal is to have a reliable system in almost any network condition.
- This is achieved by adaptive sampling and coding (audio and video) and extensive system testing







User Conference on Advanced Automated Testing





The REBEC System



User Conference on Advanced Automated Testing







Testing of Multimedia Services: challenges

- In most cases we develop applications that are distributed, real time and synchronous
- No matter the coverage of your unit tests you will always have situations like the following:
 - 1. When we use the software in more than 25 persons after 30 minutes some people perceive a strong delay
 - 2. When we use the software in VPN the meeting drops after 45 minutes
 - 3. If we show a youtube video the meeting gets slow
 - 4. After 99 hours of continuos streaming the flow SEEMS slower
- We absolutely need objectivation!
- We also gather information about system use cases and we have to describe them!
- Testing challenges:
 - 1. Multi-user
 - 2. Synchronization
 - 3. Fuzzing
 - 4. Data exchange among agents
 - 5. Benchmarking

User Conference on Advanced Automated Testing







How we test our systems: the P-BOT Framework

- PBOT is mainly a test orchestration platform:
 - Distributed
 - Coordination
 - Synchronization
 - Topology management
 - Controlled load
 - Ecc..
- Moreover it also acts as an actuator for some features:
 - Performance testing
 - Environment sensing
- It often runs on production networks
- In some way P-BOT has the same carachteristics of the TDL «Universe»: synchronous, real time ditributed
- Currently the entire system is hosted in the Rebec Infrastucture and comprises 33 Windows 10 Agents

User Conference on Advanced Automated Testing







The REBEC Testing Environment



User Conference on Advanced Automated Testing



© All rights reserved





A REBEC Test Case









A REBEC Test Case









Why TDL

- Communication between R&D and TCoE
- Description of bugs
- Application «template» i.e. same type of testing but different application.
 E.g. rebec vs teamviewer
- So we can also make benchmarking!
- So we needed a layer of abstraction to separate test desing from SUT and/or from attuator.
- Mixed scenarios like «activate blind transfer», test if it really works making a phone call
- First step towards a completely automated testing environment







TDL test example: test configuration









TDL test example: test behaviour

- Test Objective: Determine if a session can be started.
- Step: PBOT1 sends «StartSession» through g1
 - Waits for a session code for 5secs
 - Interrupts if timeout
- Step: PBOT1 send the received code through g3
- Step: PBOT1 plots specific code on the screen
- Step: PBOT2 receives the session code through g3
- Step: PBOT2 joins the session through g2
- Step: PBOT2 read the verification code through g3
- Step: PBOT2 looks on the screen for the verification code
 - Interrupts if not found after T seconds







Impressions about TDL

- We think TDL is perfect for our needs. Nonetheless...
- Our experience is not yet mature and our use of TDL is still primitive
- Our use of TDL is still experimental in the sense that it is not yet part of our official Quality Managemnete System. Il May 2017 we certified our QMS for the new ISO9001:2015 norm but in our processes there is not yet TDL. We hope for the next year.
- Still difficult to use for non computer science people
- TDL is in experimental usage. So we are here to learn and to share information.







Conclusions

- There is still a lot of work to be done
- What we are working on:
 - Better study of TDL
 - More formal test description
 - Understanding how to map performance
 - Understanding how to describe random behaviour (fuzzing)
 - Making P-BOT understand TDL
 - Mapping between graphic and abstract syntax (one is better for info exchange the other for formalization for P-BOT)







Greetings

Ricerca

2007-2013

PON

We want to thank «La Città Educante» Project \bullet



User Conference on Advanced Automated Testing

stere dell' Istre

dell'Università e della Ricerca





Q&A Contacts

- Q&A
- Thank you
- Authors Contacts:
 - Francesco Oppedisano <u>F.oppedisano@netresults.it</u> (speaker)
 - Sergio Borghese <u>s.borghese@netresults.it</u>
 - Francesco Lamonica <u>f.lamonica@netresults.it</u>
 - Enrico La Vela <u>e.lavela@netresults.it</u>



