Pictures at QRS 2015

Day 2 (August 4)


STOP


lenovo
Marriott
VANCOUVER AIRPORT


HRML: a hybrid relational modelling language
He Jifeng

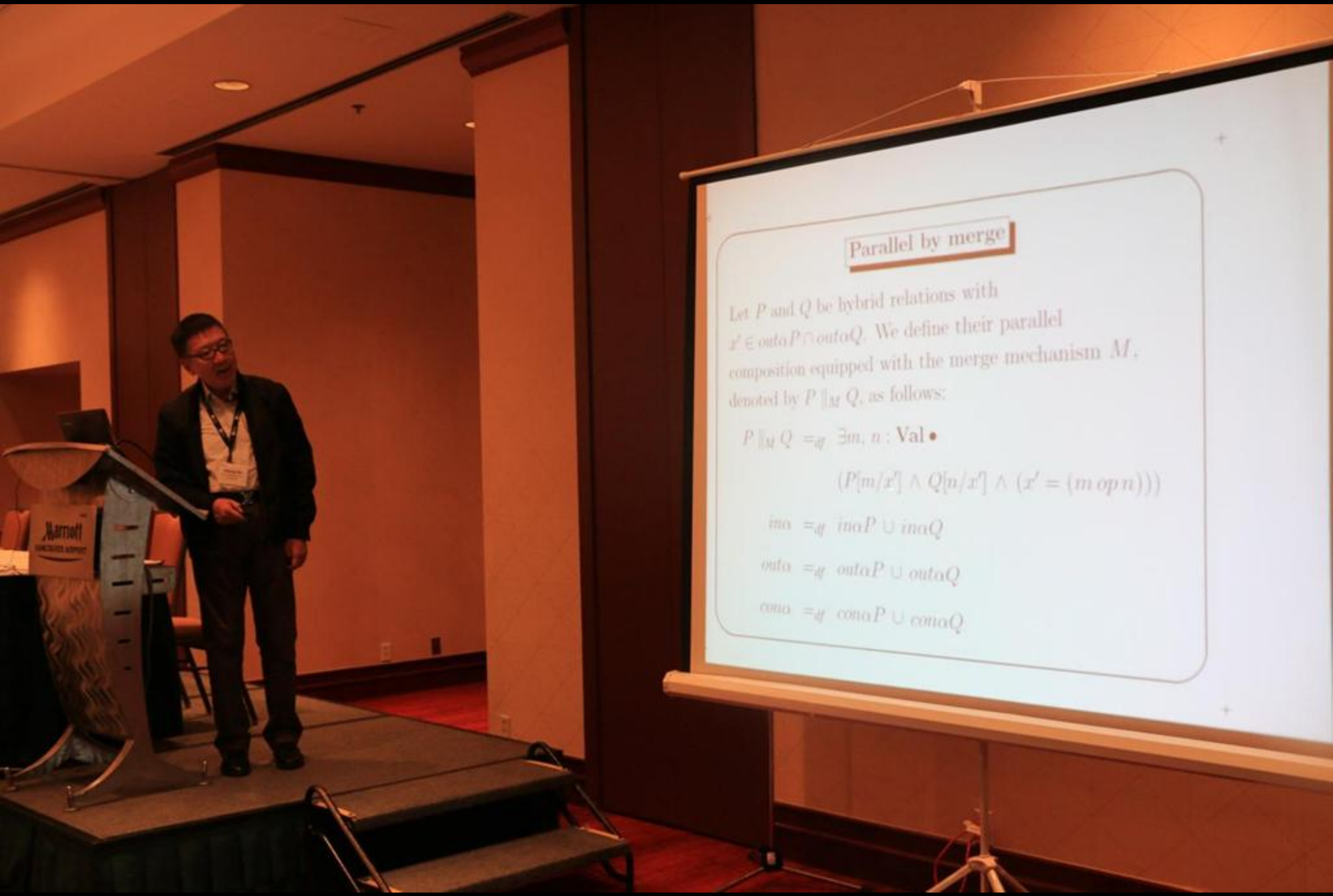

lenovo


Marriott
VANCOUVER AIRPORT
Hybrid Systems


EXIT


lenovo
Marriott
VANCOUVER AIRPORT


lenovo
Marriott


Parallel by merge
Let P and Q be hybrid relations with $x' \in outa P \cap outa Q$. We define their parallel composition equipped with the merge mechanism M, denoted by $P \parallel_M Q$, as follows:
$P \parallel_M Q =_{df} \exists m, n : \mathrm{Val} \bullet (P[m/x'] \wedge Q[n/x'] \wedge (x' = (m\ op\ n)))$
$ina =_{df} ina P \cup ina Q$
$outa =_{df} outa P \cup outa Q$
$cona =_{df} cona P \cup cona Q$


EXIT


Closure of healthy hybrid relati
Theorem
(1) $\mathbf{H}(P) \sqcap \mathbf{H}(Q) = \mathbf{H}(P \sqcap Q)$
(2) $\mathbf{H}(P) \lhd b \rhd \mathbf{H}(Q) = \mathbf{H}(P \lhd b \rhd Q)$
(3) $\mathbf{H}(P); \mathbf{H}(Q) = \mathbf{H}(P; \mathbf{H}(Q))$
(4) If P and Q lie in the complete lattice L,
$(P \|_M Q)$
where the merge mechanism
Marriott
VANCOUVER AIRPORT


HRML: a hybrid relational modelling language
$AP ::= \mathbf{skip} \mid \mathbf{chaos} \mid \mathbf{stop} \mid x := e \mid !s \mid \mathbf{delay}(\delta)$
$EQ ::= R(v, \dot{v}) \mid EQ \,\mathbf{init}\, v_0 \mid EQ \| EQ$
$P ::= AP \mid P \sqcap P \mid P ; P \mid P \lhd b(x) \rhd P \mid P \| P \mid$
$EQ \,\mathbf{until}\, g \mid \mathbf{when}(G) \mid \mu X \bullet P(X)$
$\mathbf{timer}\, c \bullet P \mid \mathbf{signal}\, s \bullet P$
$g ::= \mathbf{skip} \mid s \mid test \mid g \cdot g \mid g + g$
$test ::= \mathbf{true} \mid v \geq e \mid v \leq e \mid test \wedge test \mid test \vee test$
$G ::= g \& P \mid G | G$
Marriott
VANCOUVER AIRPORT


EXIT
lenovo
Marriott
VANCOUVER AIRPORT


Marriott
VANCOUVER AIRPORT


lenovo
Marriott


Adjust the sampling rate
The following theorem is used to reduce the sampling rate of the controller by estimating the change speed of physical state
Theorem Let $l < m < n$.
If $R \supseteq \{0 < \dot{v} \leq v\}$ and $\delta < (n-m)/v$, then
$\{R\,\mathbf{init}\,m\,\mathbf{until}\,\{v \geq n\}\}$
$= \mathbf{signal}\,s,\,u \bullet \left(\begin{array}{l} \{R\,\mathbf{init}\,m\,\mathbf{until}\,s\} \\ \left[\!\left] \left(\begin{array}{l} \mathbf{delay}(\delta); \\ \mathbf{when}\{(v \geq u)\&!s\} \end{array} \right) \right]\!\right] \{(v \leq l)\&!u\}) \end{array} \right)$
Marriott
VANCOUVER AIRPORT
EXIT


lenovo
Marriott
VANCOUVER AIRPORT

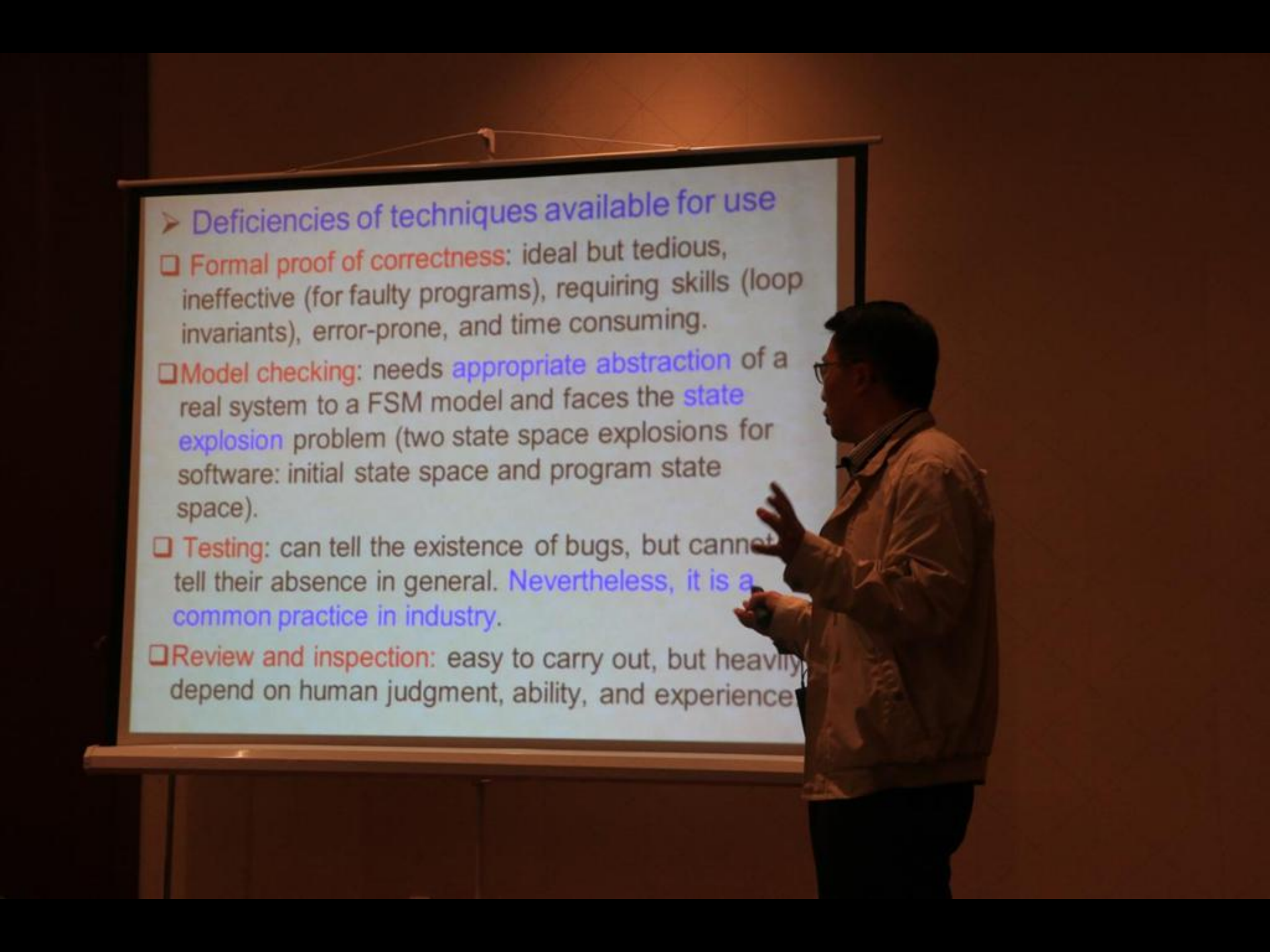

Marriott
VANCOUVER AIRPORT


Integrating Specification Animation
th Specification-Based Program
Testing and Inspection for
ware Quality Assurance
Shaoying Liu
Department of Computer Science
y of Computer and Information Sciences
Hosei University, Japan
Email: shl@hosei.ac.jp
HP: http://cis.k.hosei.ac.jp/~shl/

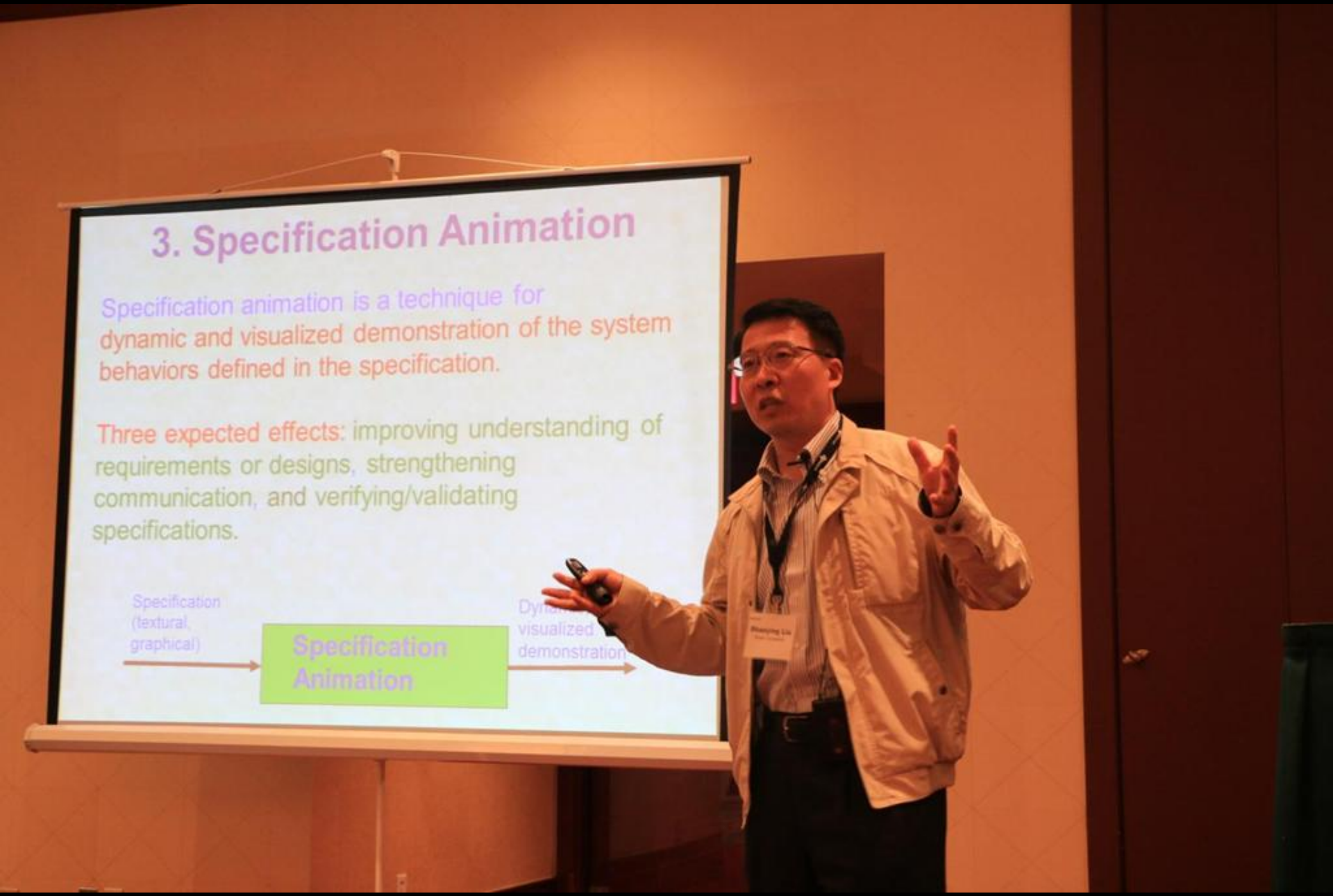

Overview
1. Challenges to Software Quality Assurance
2. Our Solution
3. Specification Animation
4. Specification-Based Program Testing and Inspection
5. Open Problems
6. Conclusions
7. Future Work


1. Challenges to Software Quality Assurance
The scale and complexity of software development projects
The scale of documentation
The complexity of documentation
The complexity of situations (e.g., requirements changing, people moving, client complaining, manager worrying, and developer fighting)
EXIT


1. Challenges to Software Quality Assurance


➢ Deficiencies of techniques available for use
❑ Formal proof of correctness: ideal but tedious, ineffective (for faulty programs), requiring skills (loop invariants), error-prone, and time consuming.
❑ Model checking: needs appropriate abstraction of a real system to a FSM model and faces the state explosion problem (two state space explosions for software: initial state space and program state space).
❑ Testing: can tell the existence of bugs, but cannot tell their absence in general. Nevertheless, it is a common practice in industry.
❑ Review and inspection: easy to carry out, but heavily depend on human judgment, ability, and experience


2. Our Solution
Specification
Animation
Software
defects
Specification
-Based
Testing


3. Specification Animation
Specification animation is a technique for dynamic and visualized demonstration of the system behaviors defined in the specification.
Three expected effects: improving understanding of requirements or designs, strengthening communication, and verifying/validating specifications.
Specification (textural, graphical)
Specification Animation
visualized
demonstration


. Specification Animation
cation animation is a technique for
c and visualized demonstration of the sy
rs defined in the specification.
xpected effects: improving understandin
ments or designs, strengthening
nication, and verifying/validating
ations.
cation
l,
al)
Specification
Animation


Overview
1. Challenges to Software Quality Assurance
2. Our Solution
3. Specification Animation
4. Specification-Based Program Testi
and Inspection
5. Open Problems
6. Conclusions
7. Future W


EXIT


EXIT


Potential Component Leaks in Android Apps
An Investigation into a new Feature Set for Malware Detection
LI Li
SnT, University of Luxembourg
Joint work with
• Kevin Allix, Daoyuan Li, Tegawendé Bissyandé, Jacques Klein, SnT
• Alexandre Bartel, TU Darmstadt
EXIT


Belgium
Germany
France
EXIT

Android

Leaks

Android browser flaw found to le

Dozens of Popular Android Apps Leak Sensitive User Data

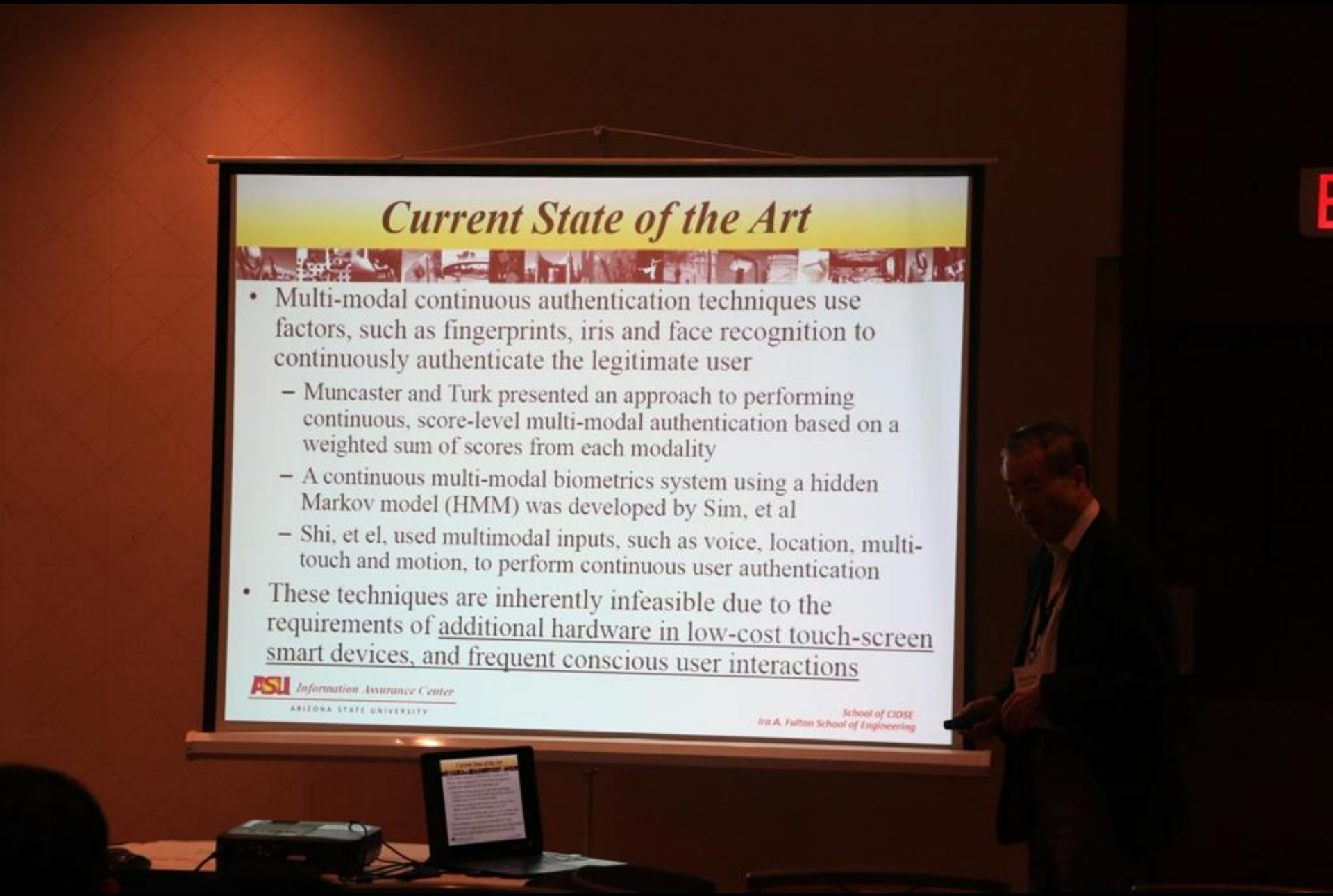
SnT, University of Luxembourg

EXIT


Bernhard
Garn


Bernhard
Garn


EXIT
An Effective Approach to Continuous User Authentication for Touch Screen Smart Devices
Arun Balaji Buduru and Stephen S. Yau
Information Assurance Center
School of Computing, Informatics, and Decision Systems Eng.
Arizona State University


Current State of the Art
• Multi-modal continuous authentication techniques use factors, such as fingerprints, iris and face recognition to continuously authenticate the legitimate user
– Muncaster and Turk presented an approach to performing continuous, score-level multi-modal authentication based on a weighted sum of scores from each modality
– A continuous multi-modal biometrics system using a hidden Markov model (HMM) was developed by Sim, et al
– Shi, et el, used multimodal inputs, such as voice, location, multi-touch and motion, to perform continuous user authentication
• These techniques are inherently infeasible due to the requirements of additional hardware in low-cost touch-screen smart devices, and frequent conscious user interactions
ASU Information Assurance Center
ARIZONA STATE UNIVERSITY
School of CIDSE
Ira A. Fulton School of Engineering


t (cont.)
aïve Bayes
ccurate
orithms, and
not user
ed as a trusted
EXIT


MAPPING MODEL COVERAGE TO CODE COVERAGE
A CEC contains Simulink blocks and conditionals, and
A Simulink CEC is a 4-tuple E={D,G,B,V},
D is a finite set of decisions determined by the Predicate Blocks of each conditional subsystem;
G is a finite set of decisions and conditions formed
B is the set of decisions and conditions that are formed by functional blocks;
V is the set of values associated with the conditions and decisions
Harriott
OUVER AIRPORT


VANCOUVER

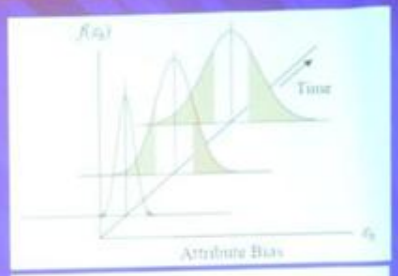

SVM Score distribution
Training Set Normal Model - SVM Score
malwares
benign
Training Set Malicious Model - SVM Score
malwares
benign
EXIT

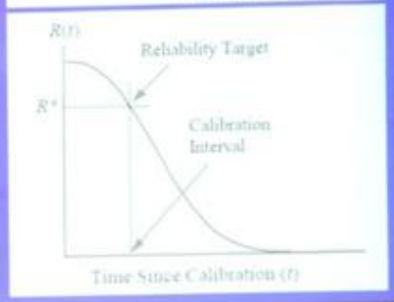

Detection Logic
algorithm
{
anomaly = normal_model_score
malicious = malicious_model_score
if anomaly < n1 && malicious < m1
predict benign
else if anomaly > n2 && malicious > m2
predict malicious
else
predict ambiguous
}

1. Introduction

Reliability of the real time system is depend on the distribution of uncertainty (attribute bias).

Uncertainty analysis is used to investigate the reliability of RTS with uncertainty variables for decision-making program.





EXIT


Karthik
Pattabiraman


LUXURIOUS
METROPOLIS

XML Bomb/Billion Laughs (BIL)

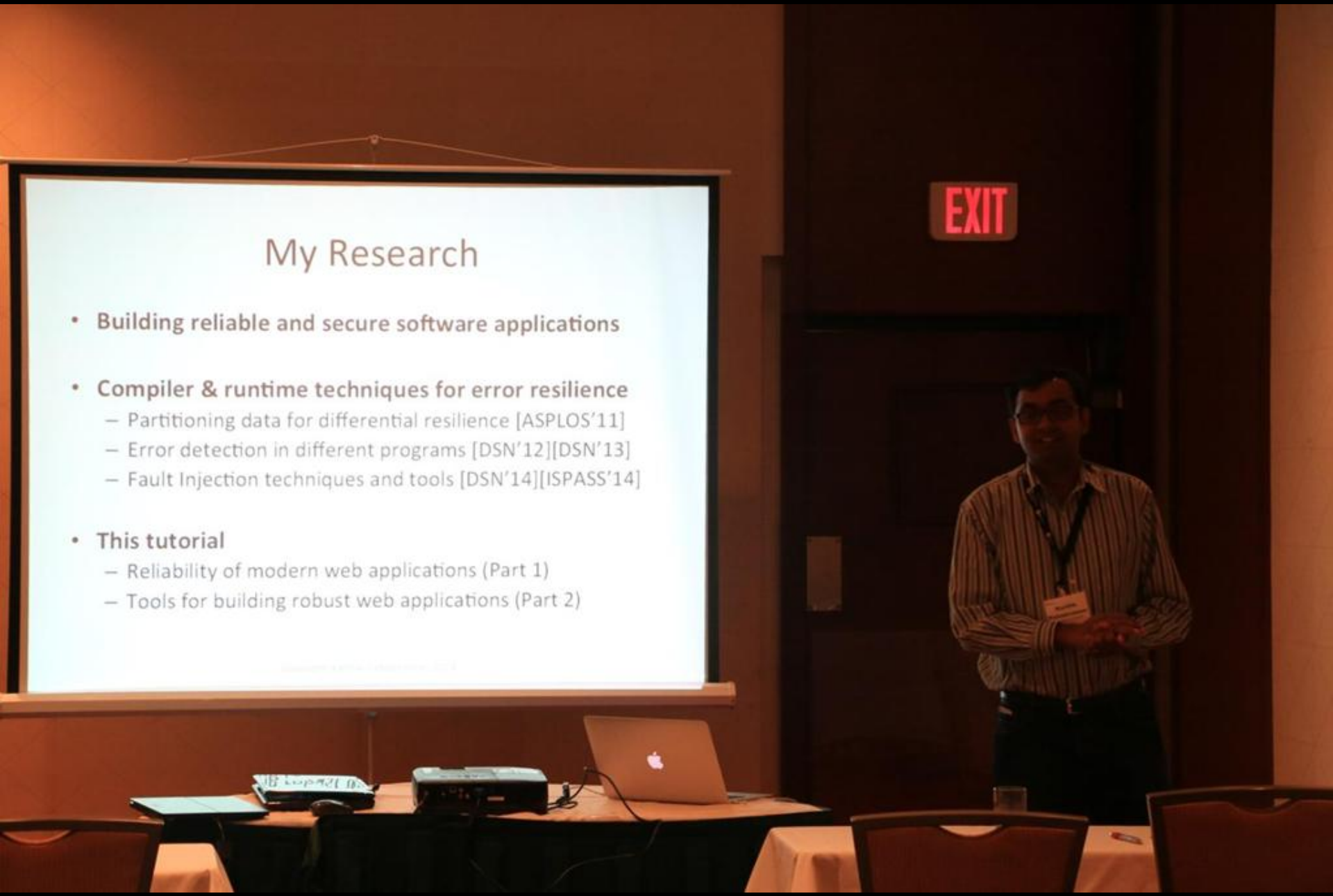
- DoS attack on XML parser using well-formed XML
  - Exponential entity expansion

- *Impact*
  - *Higher memory & CPU Consumption*
  - *DoS on parsing system*

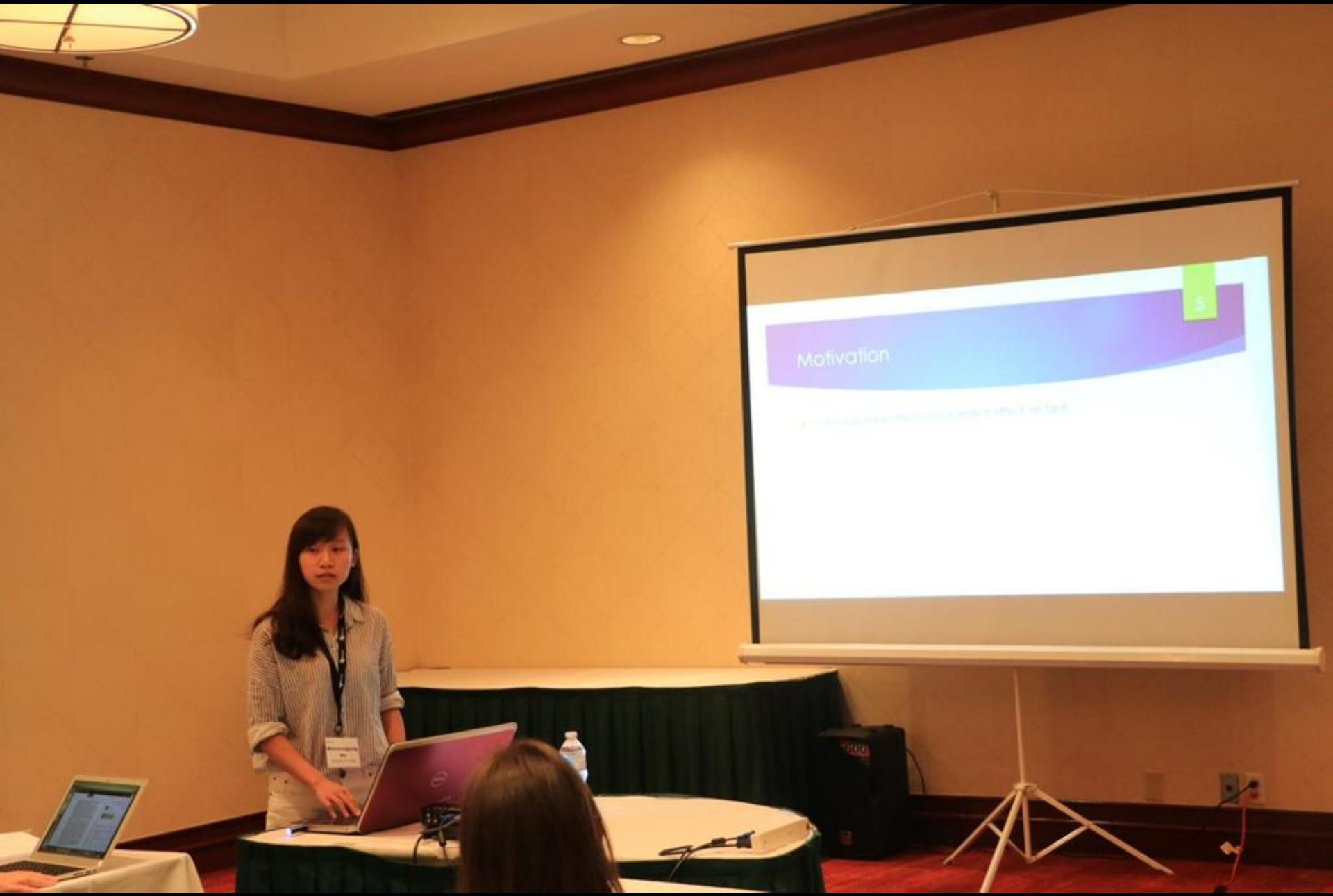
```
<?xml version="1.0"?>
<!DOCTYPE xmlBOMB [
    <!ENTITY a "Random Text">
    <!ENTITY b "&a; &a; &a; &a; &a;">
    <!ENTITY c "&b; &b; &b; &b; &b;">
    <!ENTITY d "&c; &c; &c; &c; &c;">
    ]>
<xmlBOMB>&d;</xmlBOMB>
```


Sadeeq Jan
University of
Luxembourg


Background
Dynamic Atomicity Violation Detectors
Examples
CTrigger [ASPLOS 2009, TSE 2012]
AtomFuzzer [FSE 2008]
AssetFuzzer [ICSE 2010]
Maple [OOPSLA 2012]
Two-phase strategy is often used
Prediction phase
Confirmation phase


My Research
• Building reliable and secure software applications
• Compiler & runtime techniques for error resilience
– Partitioning data for differential resilience [ASPLOS'11]
– Error detection in different programs [DSN'12][DSN'13]
– Fault Injection techniques and tools [DSN'14][ISPASS'14]
• This tutorial
– Reliability of modern web applications (Part 1)
– Tools for building robust web applications (Part 2)
EXIT


XIT
Karthik
Pattabiraman


5
Motivation


Wanwangying


Running Example
Cell
G4
H4
J2
J3
E7
E8
G2
G3
G4
H2
H3
H4
J2
J3
Classification d
true
false
true
false
8


Running Example
Cell G4 H4 J2 J3 SC
E7 1.000
E8 0
G2 0.707
G3 0.816
G4 0
H2 0.500
H3 1.000
H4 0.707
J2 0
J3 0.707
Classification d true false true false
8


Birgit Hofer


Fei Zhao


Highly-impactful vs. Other bugs
9
Highly-impactful bugs
Other bugs
reporter experience
>
comment length
>
closed %
<
EXIT


Sheraton


Problem
• Detection of information flow bugs during software design and coding time


EXIT


EXIT


Boolean Differentiation for Formalizing Myers' Cause-Effect Graph Testing Technique
IZMIR INSTITUTE OF TECHNOLOGY
IZTECH
Tolga Ayav, Fevzi Belli
Izmir Institute of Technology
Dept. of Computer Engineering
5th IEEE International Workshop on Model-Based Verification & Validation
Vancouver, August 3-5, 2015

Cause-Effect Graphs

- Cause-Effect Graphs (CEG) assists deriving tests from a given specification given in natural language.
- CEG is constructed by an experienced test engineer.
- Test cases are derived from the graph.
- My… generation method from CEG is intuition-based.


Introduction
In multilevel hybrid cloud, most of the access control policies often deal with user's single access behavior only
Multiple access between malicious users may make data flow violating security policy
malicious behavior must be prevented in the high cloud systems


Evaluations
EXIT


EXIT


LUXURIOUS
METROPOLIS


T. Y. Chen
Swinburne University
of Technology


EXIT


EXIT
QRS 2016
The 2016 IEEE International Conference on
Software Quality, Reliability & Security
Vienna, Austria, August 1-3, 2016


QRS 2016
The 2016 IEEE International Conference on
Software Quality, Reliability & Security
Vienna, Austria, August 1-3, 2016
IEEE
TU
secure


QRS 2016
The 2016 IEEE International Conference on
Software Quality, Reliability & Security
TU
secure


QRS 2016
The 2016 IEEE International Conference on
August 1-3, 2016
IEEE
TU
secure


QRS 2016
Sam Keene
IEEE Reliability Society


QRS 2016
The 2016 IEEE International Conference on
Software Quality, Reliability & Security
Vienna, Austria
-3, 2016
IEEE
TU
Graz
secure


QRS 2016
The 2016 IEEE International Conference on
Software Quality, Reliability & Security
Vienna, Austria, August 1-3, 2016
IEEE
TU
secure


Vienna, Austria
August 1-3, 2016


Vienna, Austria
August 1-3, 2016


tional Conference on
iability & Security
gust 1-3, 2016
las.edu/qrs16
secure


IEEE
Reliability Society
TU Graz
secure
Best Paper Award
Takashi Kitagawa


QRS 2016
Vienna, Austria,


QRS 2016
The 2016 IEEE International Conference on
Software Quality, Reliability & Security
Vienna, Austria, August 1-3, 2016
secure


QRS 2016


QRS 2016


Eric Wong


Prof. Dr. Franz Wotawa / TU Graz
General Chair QRS 2016, Vienna, Austria


Bernhard
Garn
SBA Research
Birgit Hofer
Graz University of
Technology


STOP
STOP