Technical Challenges & Infrastructure – Evaluation

Alexander Nelson
February 6, 2019

University of Arkansas - Department of Computer Science and Computer Engineering
Evaluation
Multiple methods of evaluating Ubicomp systems

- Simulation
- Proof of Concept
- User Study
- Clinical Study
- Release & Maintenance
When designing your experiment you need to delineate:

- Objectives – What you seek out to prove
- Evaluation – How you seek to prove it
- Claims – What you actually proved

For publication, claims are extremely important!
Claim what your work proves, but no more
E.g. Simulation results ≠ Real-time systems
Simulation Studies

Not all Ubicomp research needs to be evaluated *in situ*

E.g. Mobile Ad-Hoc/Mesh Networks have extensive simulators
~ 75% of papers on MANETs are simulation based

Especially useful for evaluation of protocols and comparison
Need to scope claims to that which can be evaluated in simulation
Proof-of-Concept/Pilot Tests—

“The construction of working prototypes of the necessary infrastructure in sufficient quality to debug the viability of the system in daily use; ourselves and a few colleagues serving as guinea pigs.” – Mark Weiser
The PoC itself only demonstrates technical feasibility
Does not evaluate:

- Implementation metrics (efficiency, accuracy, etc...)
- User Interaction
- Scale limitations

These metrics must be evaluated with more rigor
Design experiments to test specific metrics
User study itself is a large field of research
Narrow the scope of a particular study
Concerns:

• Quantitative vs. Qualitative metrics
• Number of participants
• Data Management (PII?)
• Binary/continuous success metrics?
• Institutional Review Board?
Psychometric scale using questionnaires
Determine user impression of systems
System Usability Scale – John Brooke 1986
10-point Likert Scale to assess system usability

1. I think I would like to use this system frequently
2. I found the system unnecessarily complex
3. I thought the system was easy to use
4. I think that I would need the support of a technical person to be able to use the system
5. I found the various functions in this system were well integrated
6. I thought there was too much inconsistency in the system
7. I would imagine that most people would learn to use this system very quickly
8. I found the system to be very cumbersome to use
9. I felt very confident using the system
10. I needed to learn a lot of things before I could get going with this system

Score is tallied as sum of:
Score-1 for odd numbered questions
5-score for even numbered questions
Final score is sum $\times$ 2.5 (ranges between 0-100)
A strong evaluation is to release the system to others
Can be used & evaluated by others

Time consuming process; create standard platform for:

- Hardware platforms for test
- OS/Software for release
- Infrastructures
- Middleware
- APIs, Example Code, Documentation
Learning from systems research

Communicate your findings

Approaches:

- Make system available for others to try/test/critique/adapt
- Publish datasets collected from system
- Publish schematics/instructions/documentation (Wikis are great)
- Academic dissemination – papers, conferences, workshops, etc...
How to document your findings and system

• Explain the specific question/challenged addressed
• Explain any underlying assumptions
• Relate work to other works
• Divide documentation into technical and research contributions
• Describe your evaluation approach, methods, setup, and results
• Discuss the contribution – Claim what you can, but no more
Lessons & Takeaways

- Technical Challenges
  1. Resource Constraints – Challenges & Approaches
  2. Volatile Environments
  3. Heterogeneous Systems
  4. Fluctuating/Evolving Use Patterns
  5. Invisible Computing (non-intrusive)

- Designing a System
  1. Divide user/system responsibilities
  2. Handle Uncertainty
  3. User Mental Model
  4. Handling volatility
  5. Debugging
Lessons & Takeaways

• Implementing Systems
  1. Off-the-shelf solutions
  2. Custom/Bespoke solutions
  3. Agile approaches to system development
  4. Participatory Design
  5. Plan for Volatility

• Evaluating Systems
  1. Determine metrics
  2. Designing studies
  3. User study
  4. Communicating/Documenting your system