Improving Live Sequence Chart to Automata Translation for Verification

Rahul Kumar & Eric Mercer

GT-VMT 2008, Budapest, Hungary

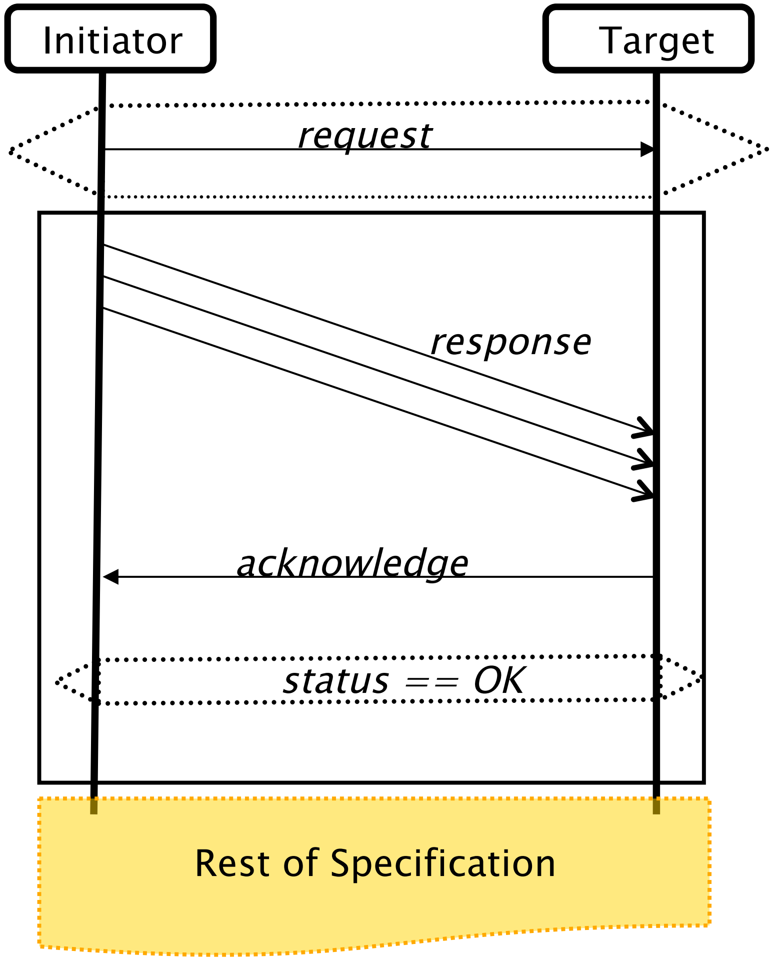
Specifications

- Bulky
- Hard to write
- Even harder to read
- Extracting correctness properties…

Protocol	Pages in Specification
HTTP	114
TCP	91
BVCI	60
SSH	38

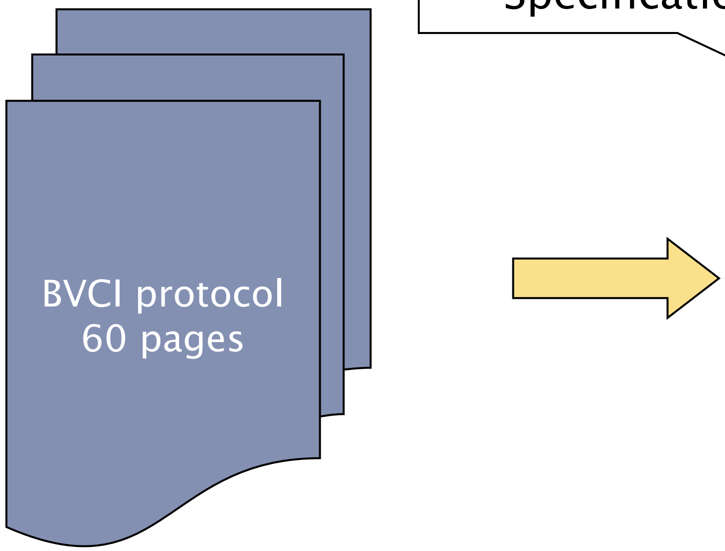
Alternative: Live Sequence Charts

- Intuitive
- Formal semantics
- Inter-process behavior
- Other:
  - Interaction diagrams
  - Message Sequence Chart
  - Timing Diagrams
  - Sequence Diagrams



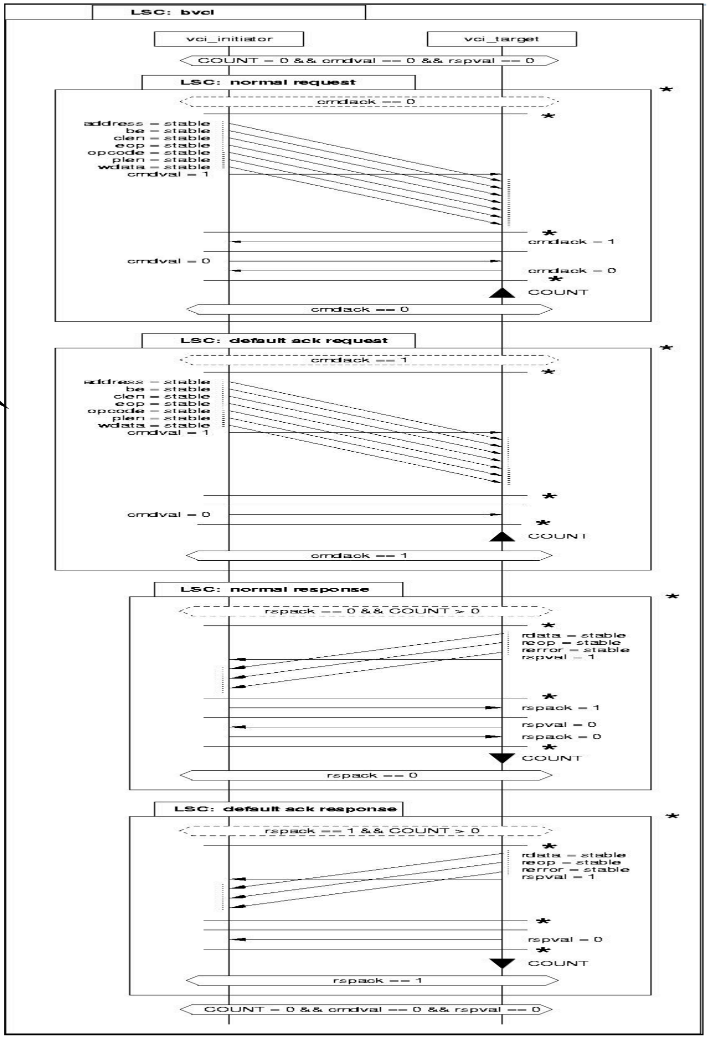
Damm et. al., Brill et. al., R.ITU-T. 120

Example



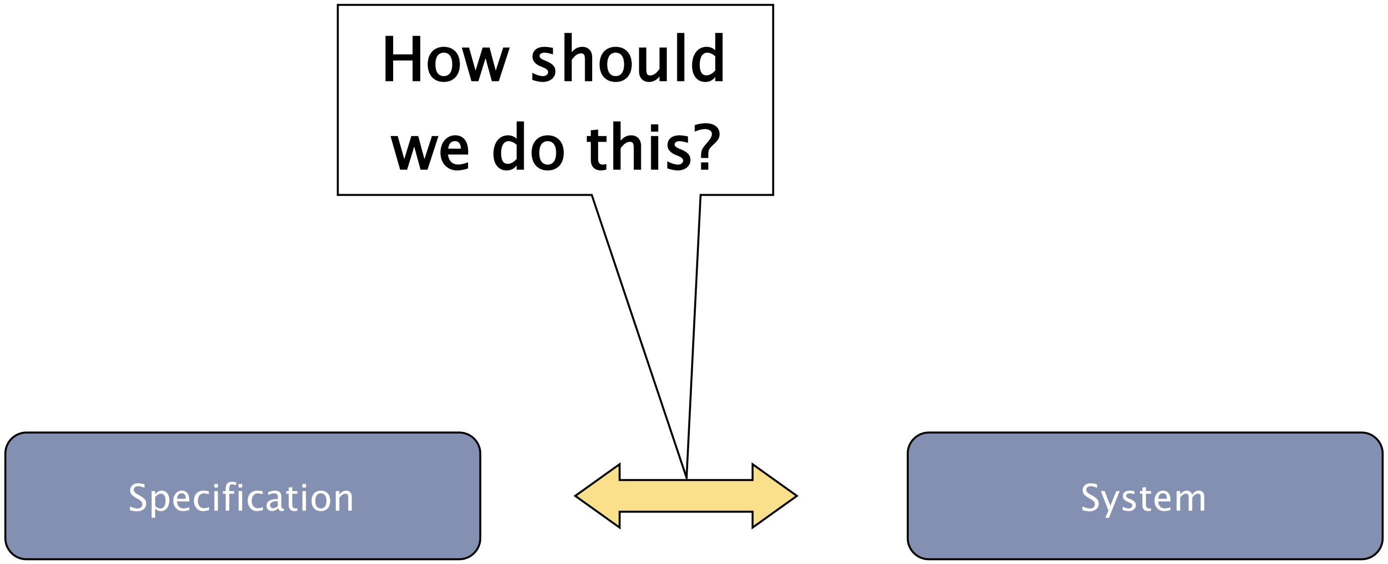
BVCI protocol
60 pages

One Page
Specification!

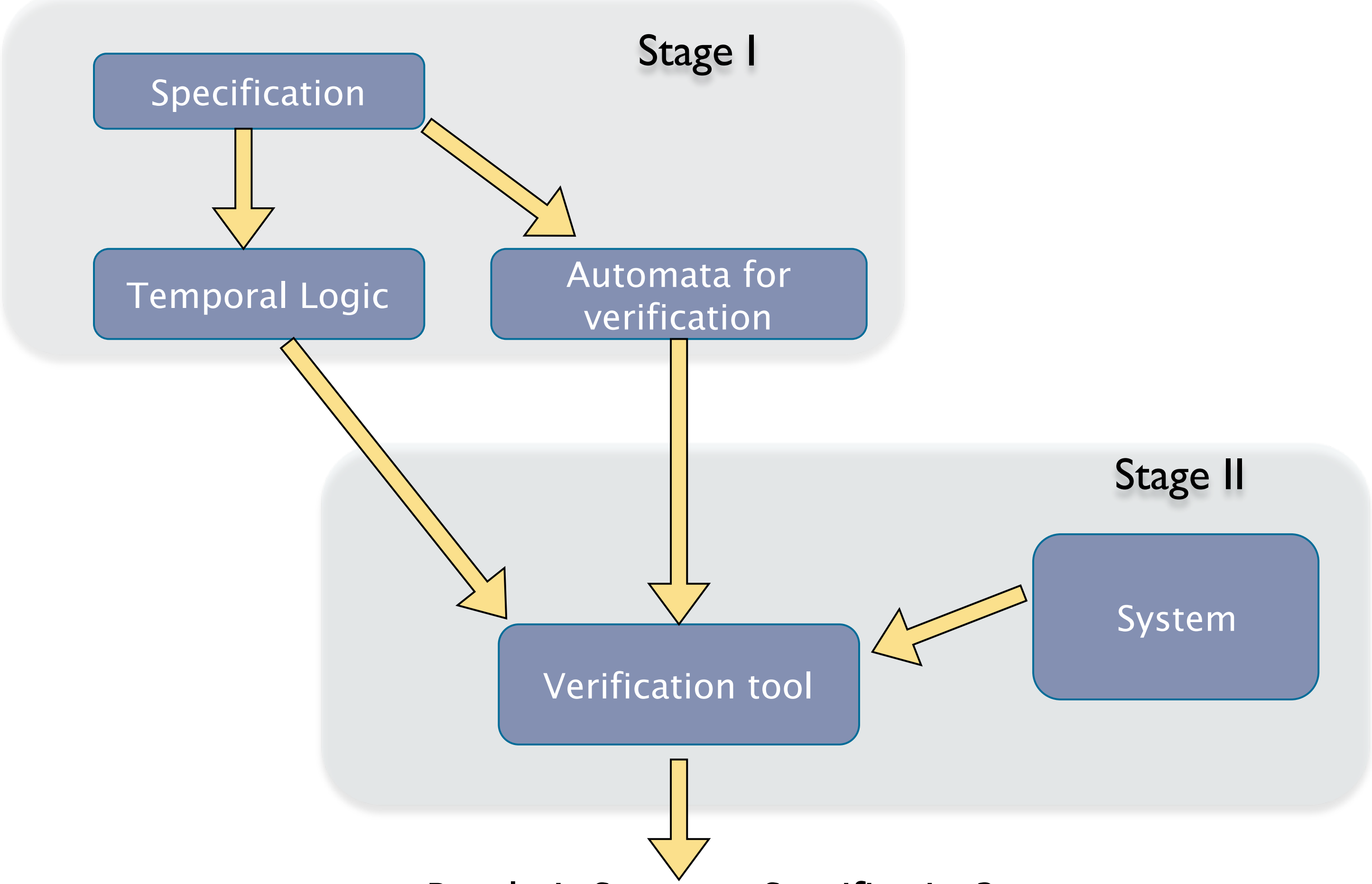


Bunker et. al.

How do we use them?

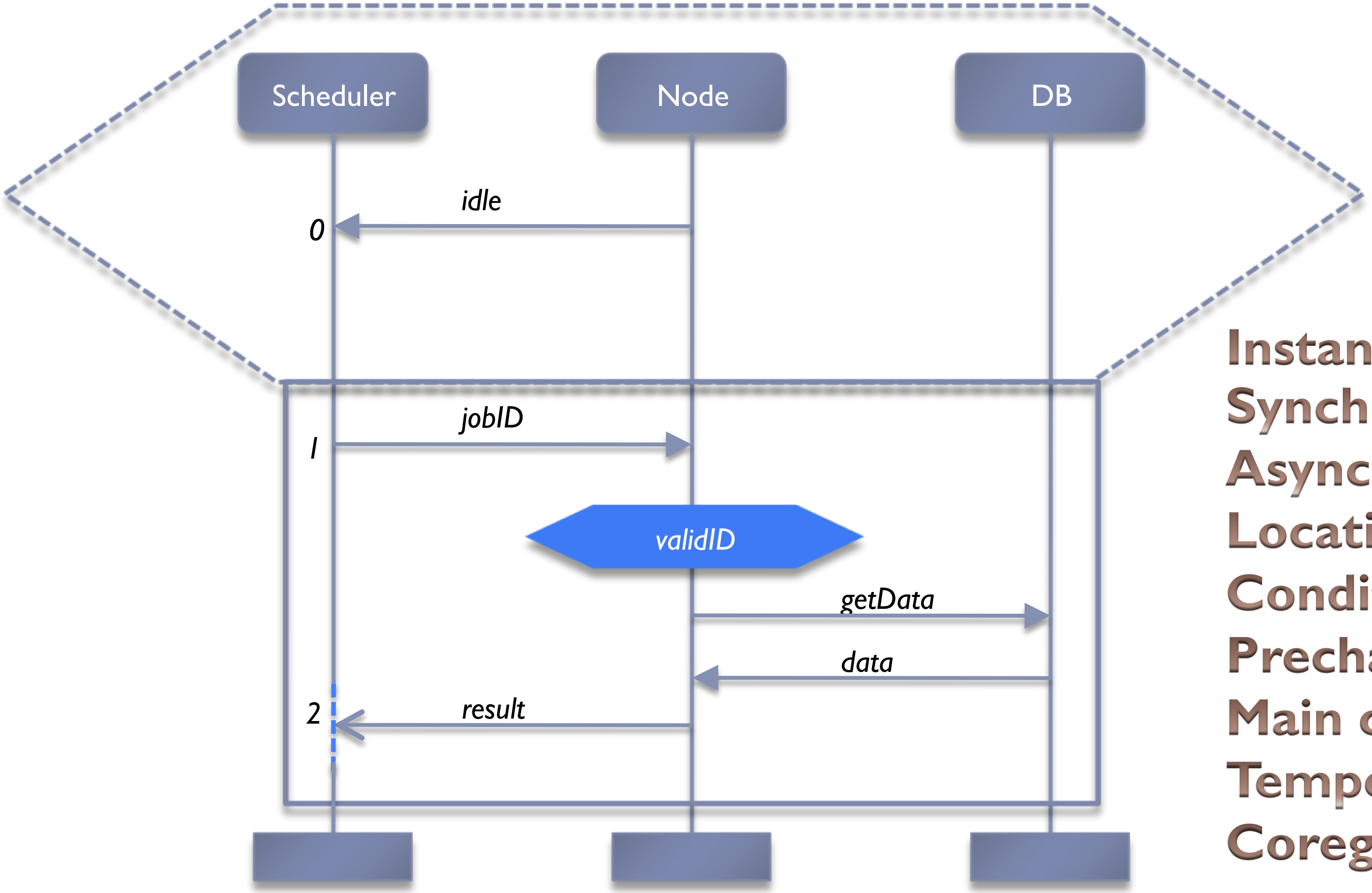


How do we use them?



Result: Is System = Specification?

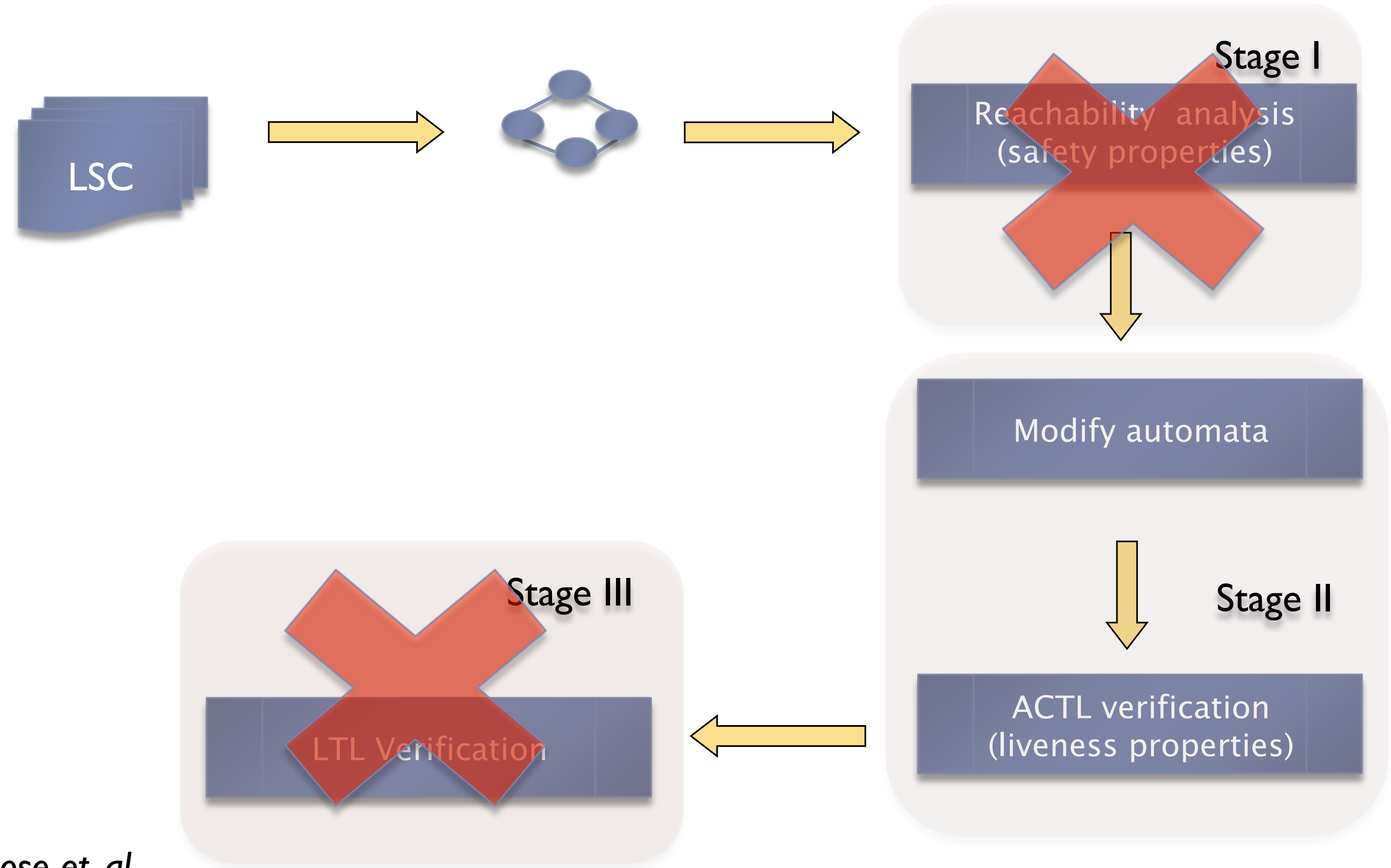
Live Sequence Charts



Instances/Processes
Synchronous Messages
Asynchronous Messages
Locations
Conditions
Prechart
Main chart
Temperatures
Coregions

Simultaneous regions

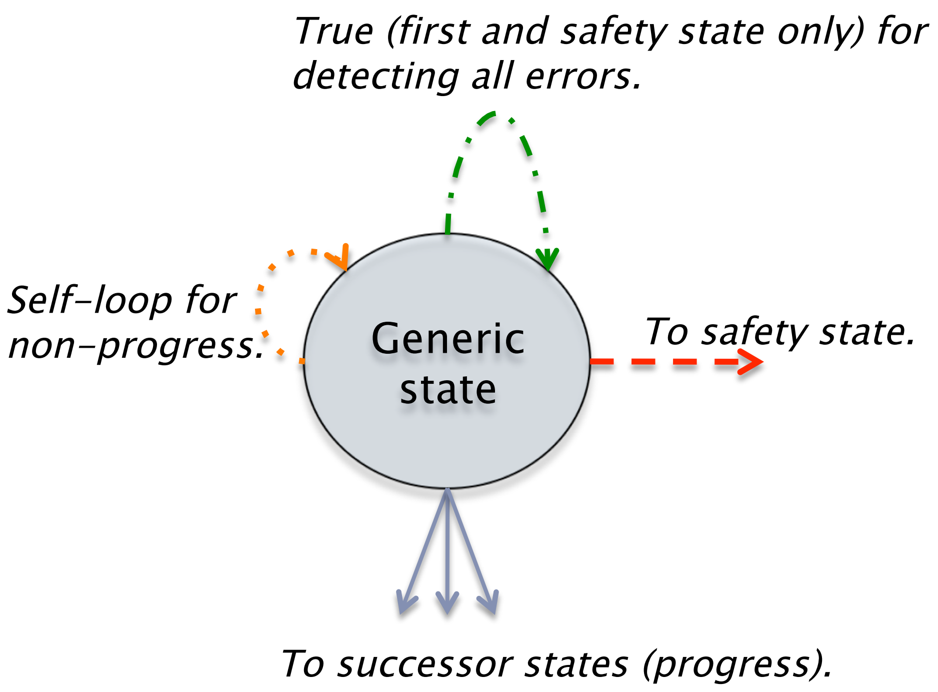
Previous Translation to Automata



Klose et. al.

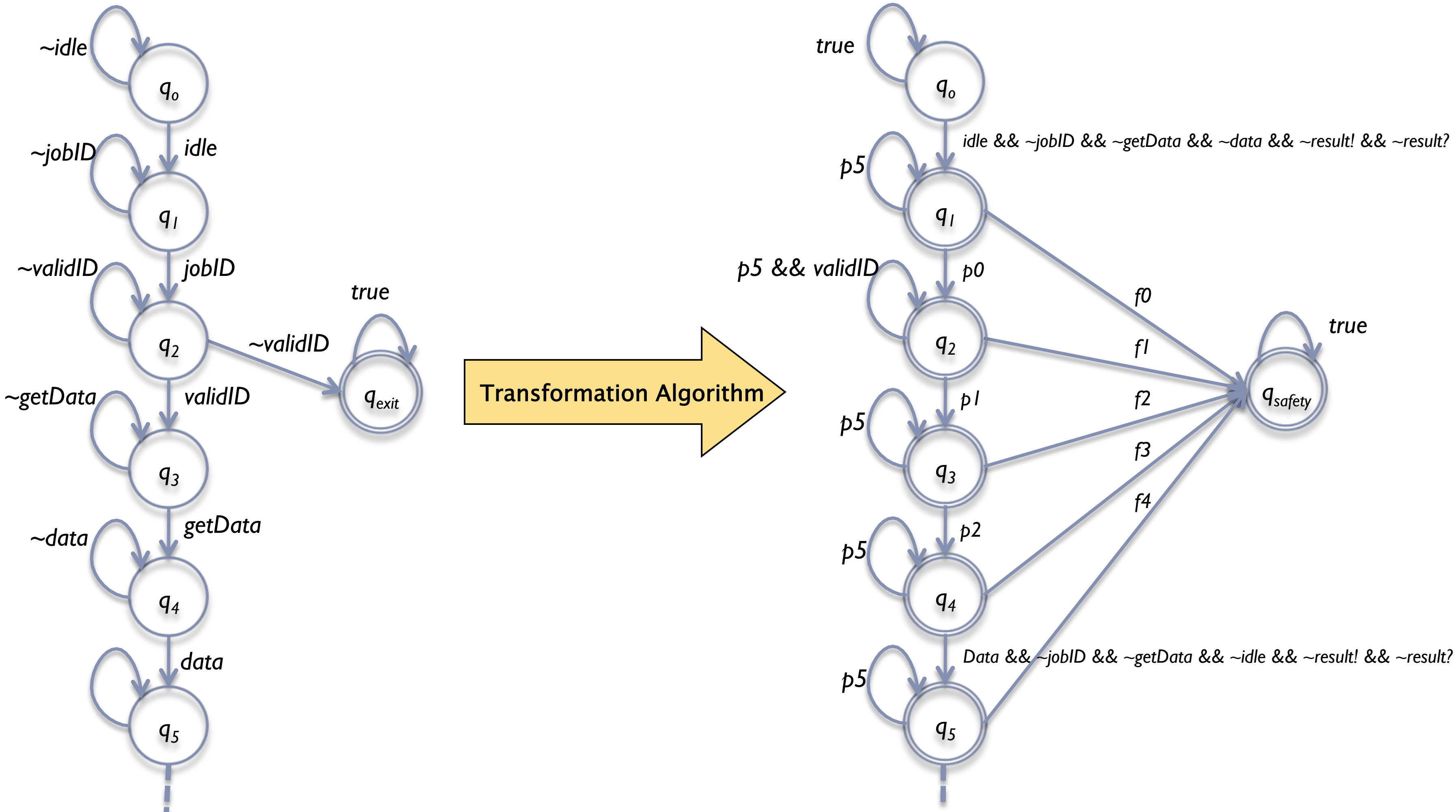
Transformation Algorithm

- Process each state of automaton using depth first traversal
- For each state:
  - Create deterministic transition relation
  - Create total transition relation
- Proof of correctness included in paper



True (first and safety state only) for detecting all errors.
Self-loop for non-progress.
Generic state
To safety state.
To successor states (progress).

Automata Transformation

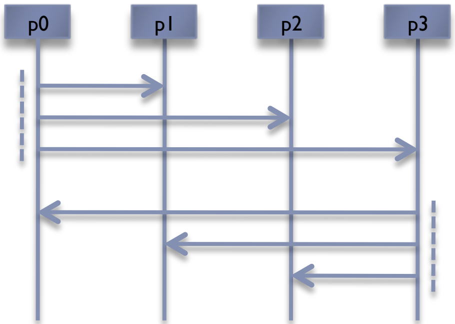


p0, p1, p2, p5, f0, f1, f2, f3, f4 : transition labels similar to transitions shown

Testing

- Test on symbolic model checking using NuSMV
  - Compare to previous automata approach (Klose *et. al.*, Toben *et. al.*)
- Test using SPIN
  - Compare to past LSC to LTL approach (Kumar *et. al.*)
- Highly concurrent specification (a worst case)
  - *Acxm*: Chart contains *c* co-region with *m* messages in each co-region
- Use puzzle solving models with messages

A2x3



Results: NuSMV

Specification	Traditional Verification						Improved Verification	
	Reachability		ACTL		Total			
	States	Time	States	Time	States	Time	States	Time
A3x5	1.02e06	34	1.47e07	35	1.57e07	69	1.42e06	34
A3x6	1.02e06	237	1.016e06	239	2e06	477	471552	251
A3x7	879048	1568	879048	1562	1.75e06	3130	521504	1550

Time in seconds.

2x faster!!

Results: SPIN

Specification	Model	Without Errors			With Errors		
		States	Memory	Time	States	Memory	Time
A7x6	soko	97500	17.2	125	89323	16.4	125
	plain	406	7.4	123	406	7.4	124
A8x6	soko	97500	18.5	214	89323	17.7	210
	plain	406	8.7	216	406	8.7	215
A9x6	soko	97500	20.1	325	89323	19.3	344
	plain	406	10.3	335	406	10.3	334

Memory in MB, Time in seconds.

5x bigger specifications!!

Conclusions

- New translation provides an automata
  - Better suited for verification
  - Performance improved
  - Eliminates need for special tools and algorithms
  - Does have to deal with standard synchronous composition
- Future work:
  - Extend translation to additional constructs of LSCs
  - Extend translation to knowledge based logics
  - Provide a tool for LSC to automata development

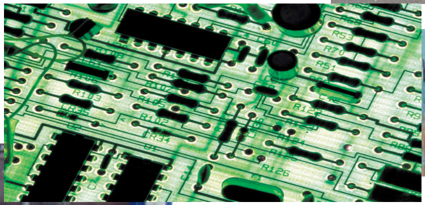
Questions?

Rahul Kumar (rahul@cs.byu.edu)

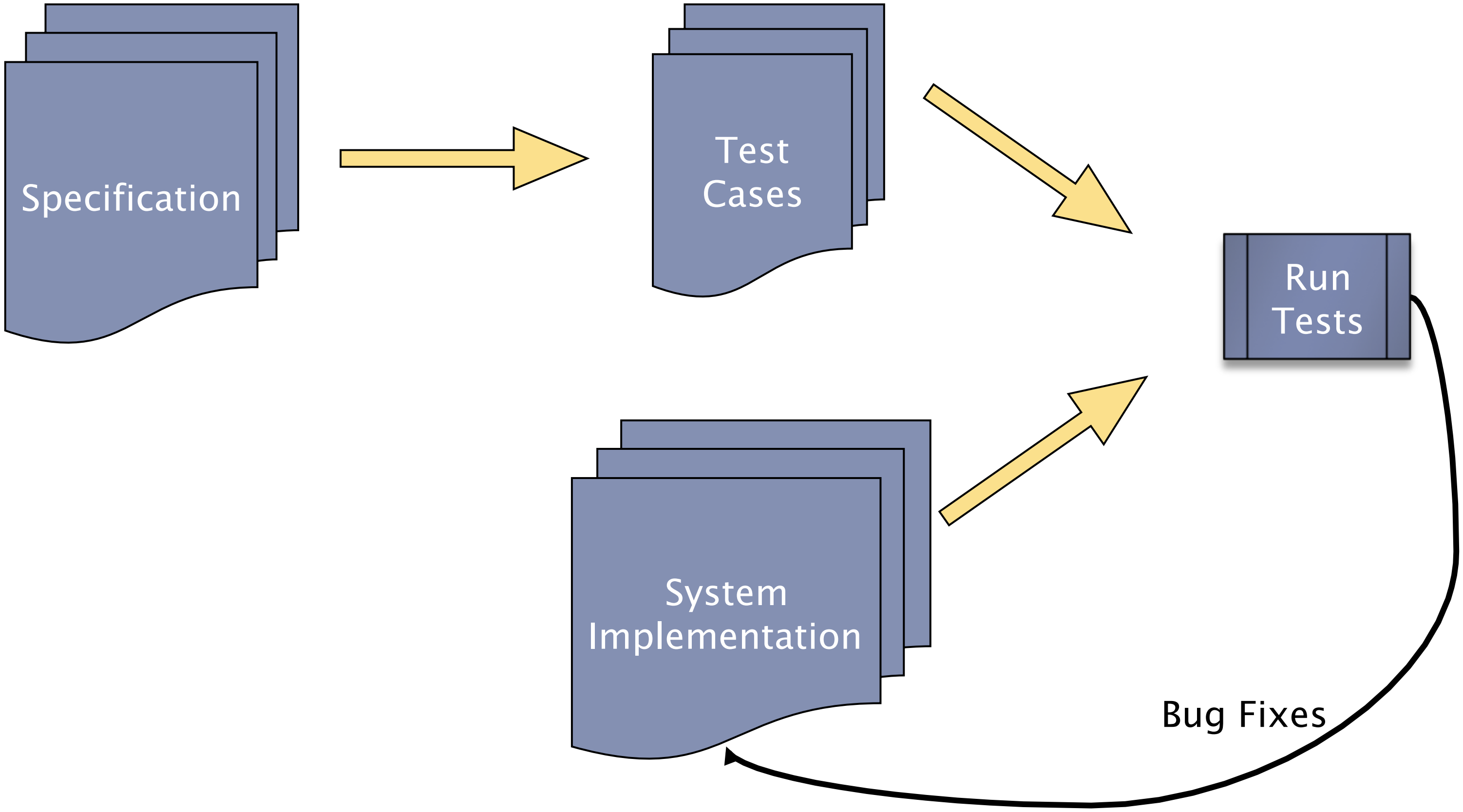
Eric Mercer (egm@cs.byu.edu)

Software Model Checking Laboratory
3325 TMCB
Brigham Young University
Provo, UT 84606
USA

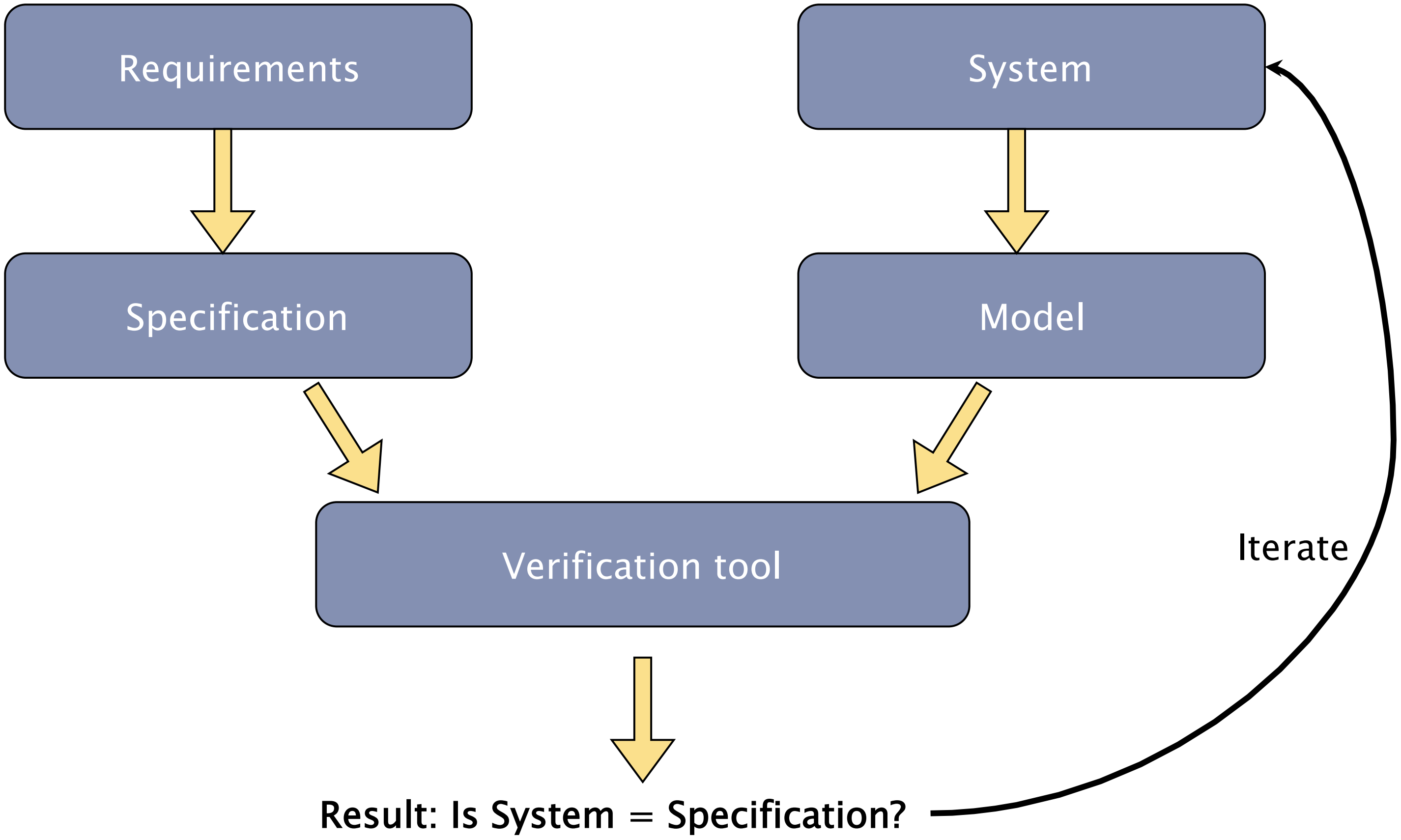
Trends



Software Testing Today



Formal Verification


Requirements
System
Specification
Model
Verification tool
Iterate
Result: Is System = Specification?