

# Software – what our everyday life is depending on

Zoltán Horváth  
hz@inf.elte.hu

Faculty of Informatics, Eötvös Loránd University, Budapest

Doctor Honoris Causa award ceremony, University Babeş-Bolyai  
May 22, 2014

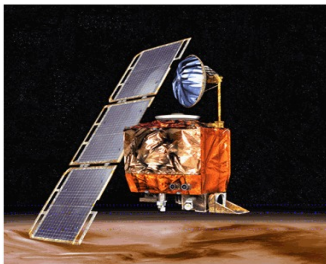
# Software – what our everyday life is depending on

- communication: phone, e-mail, social media
- navigation
- home banking
- healthcare administration, medical devices
- car – in-vehicle software: diagnostic, safety, driver assistance
- aircraft – autopilot control system ("Who's really flying the plane?")
- nuclear power plant control system safety software

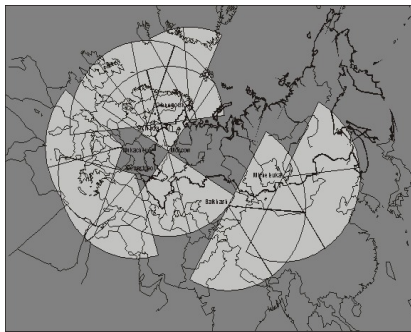


# Infamous software bugs

- The Mars Climate Orbiter doesn't orbit (1998, metric system, \$327.6 million)



- Call waiting ... and waiting ... and waiting (On Jan. 15, 1990, around 60,000 AT&T customers, congestion lead to cascade reset of 114 switches)
- Therac-25 Medical Accelerator disaster (1985, race condition, two modes)
- Soviet early-warning system (Sep. 23, 1983, Petrov)

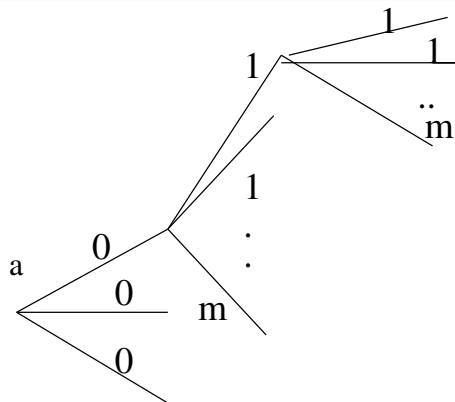


- Nuclear Power Plant shutdown (June 5, 2008, USA, software update in a distributed system)



- Ford in-vehicle software failure - hw. reset at Körösfő (2014)
- Samsung mobile network connection error (2014, flight mode off and on)

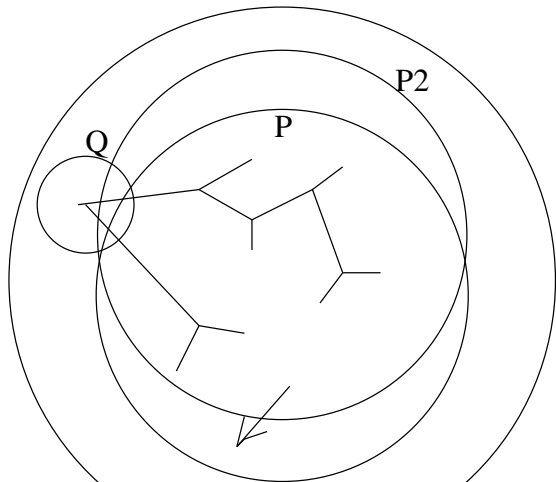
# Correctness of Distributed programs



- interleaving, branching time semantics of distributed and parallel programs
- testing of properties, reachable states
- specification of the problem to solve
- static verification, analysis, calculation of reachable states

# Always true is not always invariant

- P invariant – P holds initially and P is preserved (also in unreachable states)
- P2 always true – P2 holds in all reachable states



# Programmig languages support for software quality and correctness

## Functional programming languages

- strong type systems
- 5-10 times less Line Of Code – less errors
- parallel, multicore evaluation
- referential transparency, equational reasoning



# Parallelism and Functional Programming

- DClean - a coordination language for type safe distributed cooperation of Clean programs - with Viktória Zsók, Zoltán Hernyák
- Sparkle-T - proof tool for temporal properties (e.g. invariants) of Clean programs - with Máté Tejfel and Tamás Kozsik
- Static analysis - to support code comprehension at industrial level in Erlang - with Melinda Tóth, István Bozó and others
- Paraphrase - property preserving transformations of Erlang programs to enable parallel multicore execution - with Tamás Kozsik, Melinda Tóth, István Bozó, Judit Kőszegi, Dániel Horpácsi, Viktória Fördős and others

# Babes-Bolyai and ELTE

- SZÁMOKT, Illyefalva, May 27-29, 1994 : a paper on Teaching parallel programming,
- CEEPUS network, from 1998, Ministers' Prize of Excellence: 2004, 2007, visiting Babes-Bolyai every year, coordinators: Zoltán Kása, Judit Robu
- CEFP functional programming summer schools biannually from 2005, 4 LNCS volumes published by Springer, organized with Horia Pop, Anna Soós, Lehel Csató, Viktória Zsók
- Giving regular lectures, courses on Parallel and Functional programming at Babes-Bolyai since 2009,
- agreements on joint master and Phd programme in all fields of education, with Anna Soós,
- MACS - joint conference,
- future strong cooperation (e.g Mundus)