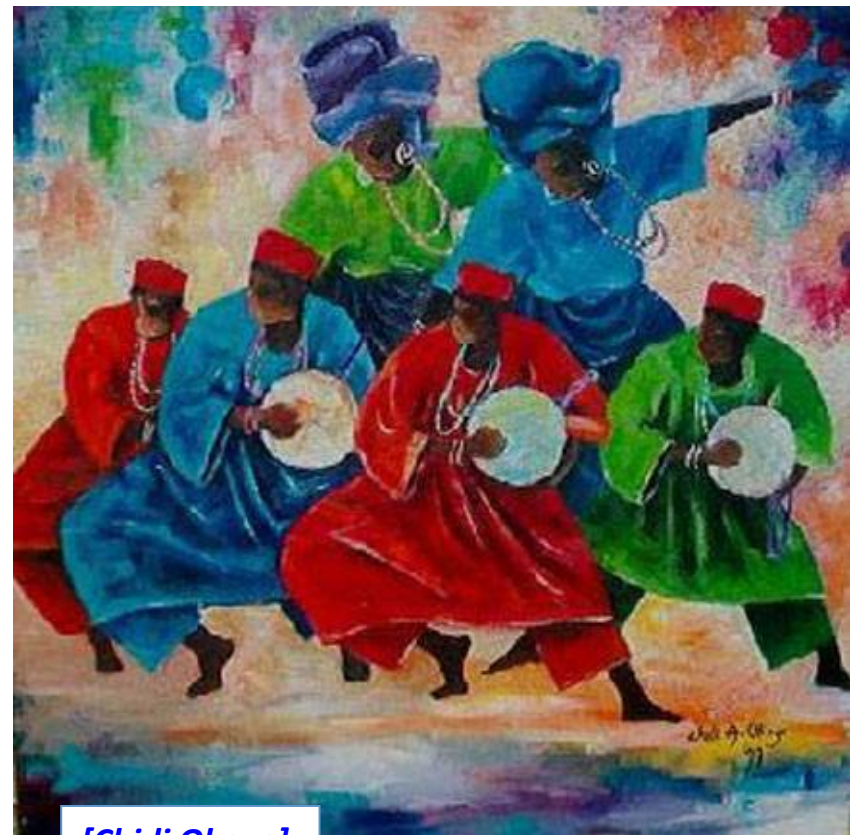


Engineering Adaptive Software Systems: A Requirements Engineering Perspective

John Mylopoulos
University of Trento

EASSy Workshop,
Shonan Village Centre,
September 9-12, 2013



[Chidi Okoye]

Adaptive Software Systems

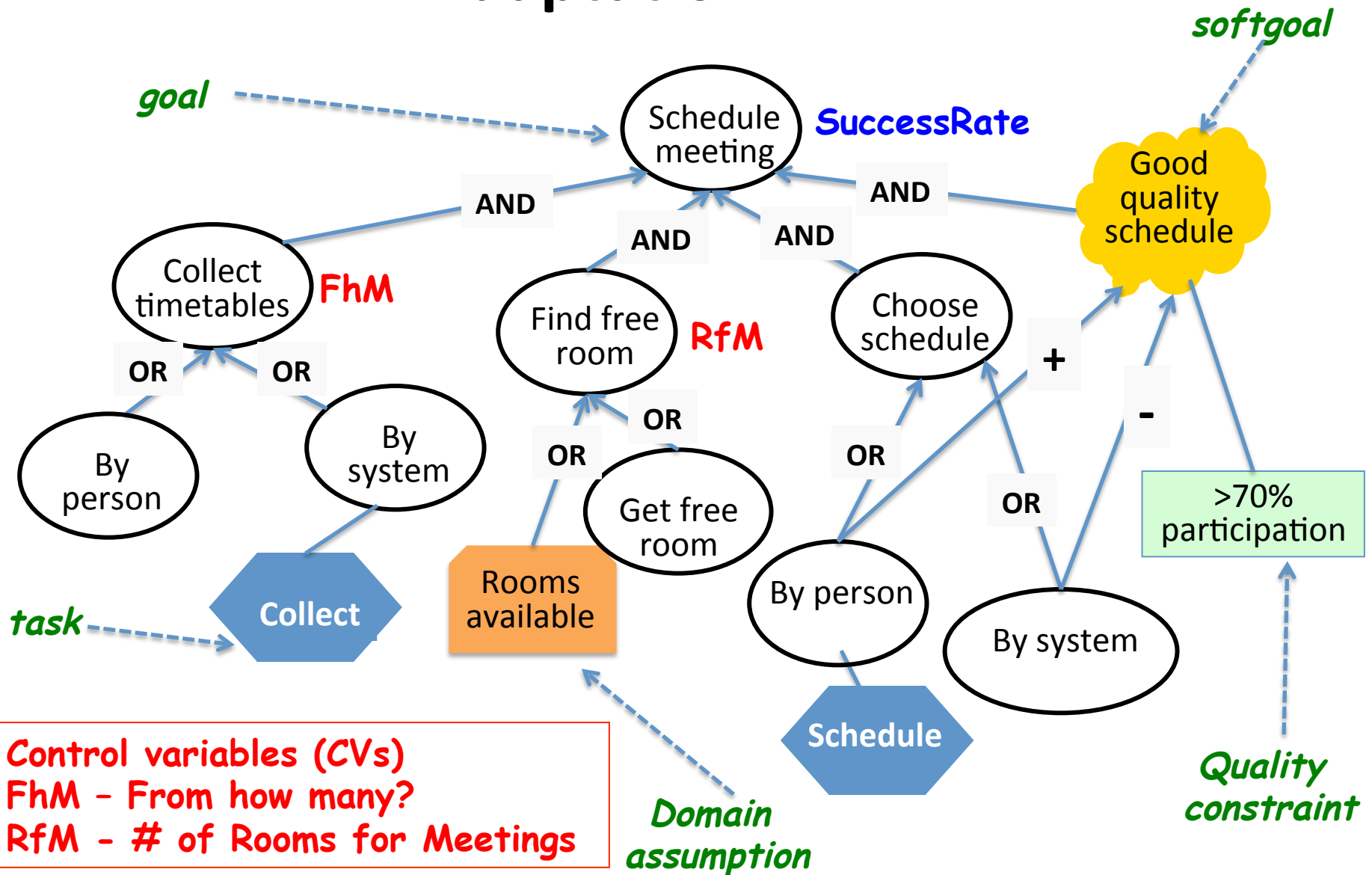
- Software lives and operates in uncertain environments, where system components may under-perform, and domain assumptions may not hold.
- How to cope? Make software *adaptive*.
- But, when to adapt, and how?

*Adapt to requirements failures,
or when requirements say so ...*

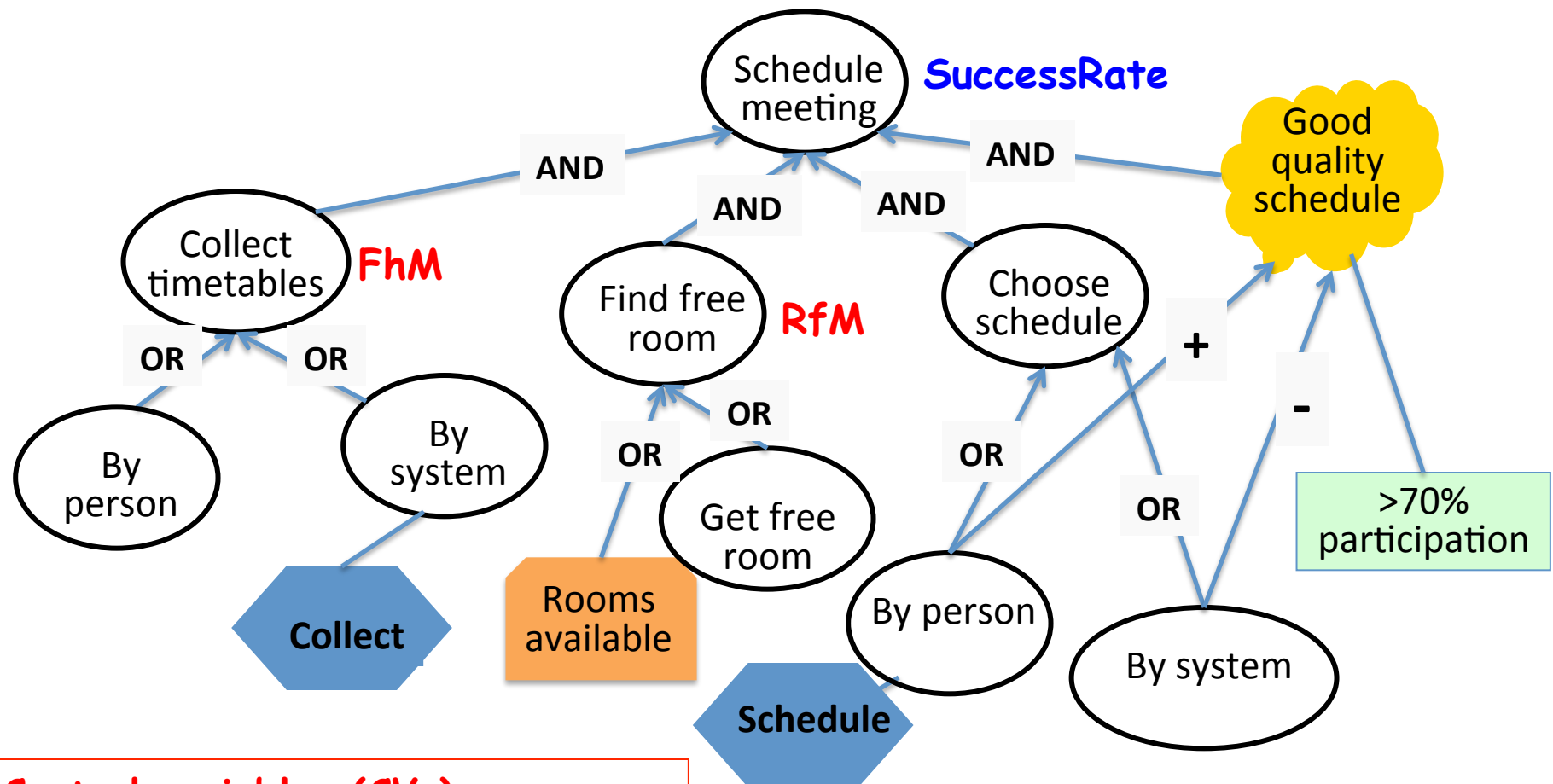
Requirements-driven Adaptation

- Requirements are (stakeholder) *goals* and generally have many solutions.
- A system can't be adaptive unless it supports several of these solutions.
- Adaptation takes place when one or more requirements fail.
- Adaptation amounts to having the system switch from one solution to another, with an expectation that the new solution will work better.
- Adaptation is driven by an *adaptation mechanism* that takes into account the failures-at-hand (divergence from requirements), and picks an alternative solution among those available by exploiting qualitative relations between alternative solutions and requirements.

Adaptation



Monitoring and Diagnosis



Control variables (CVs)
FhM - From how many?
RfM - # of Rooms for Meetings

Awareness and Evolution Requirements

- These are special classes of requirements that lead to adaptation.
- **Awareness requirements** define limits on the amount of runtime failure that can be tolerated.
- For example, if $R = \text{“Schedule meeting”}$, let $R' = \text{“}R \text{ will not fail more than 4\% of the time during any one month period”}$.
- **Evolution requirements** are requirements that define changes over time to other requirements.
- For instance, “If R fails, try R_- instead”, where $R_- = \text{“Schedule conference call”}$.
- Notice that evolution requirements specify what’s to change, awareness requirements do not.

Adaptation vs Evolution

- 🌍 In biology, individuals adapt and species evolve, both to survive. Adaptation involves switching to an alternative behaviour, evolution involves changing the genetic makeup/blueprint and physical structure of a species.
- 🌍 By analogy, we say that a particular software system (e.g., the MacOS running on my machine) *adapts* if it switches at run-time to an alternative behaviour. A software system *evolves* if its requirements change and the system architecture, implementation, etc. change in accordance.
- 🌍 Software individuals adapt, software species evolve.
- 🌍 Evolution entails requirements changes, adaptation entails no changes to requirements.

Notes

- The concepts presented constitute the basis of three theses:
 - ✓ Vitor Souza (2012, UTrento)
 - ✓ Yiqiao Wang (2009, UToronto)
 - ✓ Fabiano Dalpiaz (2011, UTrento)
- Two other theses have explore the notion of variability in Requirements Engineering:
 - ✓ Sotiris Liaskos (2008, UToronto)
 - ✓ Alexei Lapouchnian (2010, UToronto)
- Finally, the notion of requirements evolution is explored in the thesis of:
 - ✓ Neil Ernst (2011, UToronto)

Ongoing Work

- Runtime requirements models [Dalpiaz13]
- Adaptation mechanisms with multiple control variables and indicators [Angelopoulos]
- Agile requirements engineering [Ernst13]
- Adaptation and evolution for socio-technical systems [Aydemir]



*“Like Darwin, we don’t focus anymore
on how software species came to be.
Rather, we are interested in ways
they can adapt and evolve in order to
survive!”*

References

- 🌈 [Dalpiaz13] Dalpiaz F., Borgida A., Horkoff J., Mylopoulos J., “Runtime Goal Models”, 7th IEEE Int. Conference on Research Challenges in Information Science (RCIS’13), Paris, May 2013.
- 🌈 [Jureta10] Jureta, I., Borgida, A., Ernst, N., Mylopoulos, J., “Techne: Towards a New Generation of Requirements Modeling Languages with Goals, Preferences and Inconsistency Handling”, 19th Int. IEEE Requirements Engineering Conf. (RE’10), Sydney, September 2010.
- 🌈 [Souza11] Silva Souza V., Lapouchnian A., Robinson W., Mylopoulos J., “Awareness Requirements for Adaptive Systems”, 6th ICSE Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS’11), Waikiki Honolulu, May 2011.

References (cont'd)

- 🌈 [Souza11a] Silva Souza V., Lapouchnian A., Mylopoulos J., “System Identification for Adaptive Software Systems: A Requirements Engineering Perspective”, 30th International Conference on Conceptual Modelling (ER’11), Brussels, November 2011, 346-361.
- 🌈 [Souza12] Silva Souza, V., Lapouchnian A., Angelopoulos K., Mylopoulos J., “Requirements-Driven Software Evolution”, *Computer Science – Research and Development (CSRD)*, Springer-Verlag, October 2012 (online version).
- 🌈 [Souza12a] Souza V., Lapouchnian A., Mylopoulos J., “Requirements-Driven Qualitative Adaptation”, 20th International Conference on Cooperative Information Systems (CoopIS’12), Rome, September 2012, Springer-Verlag LNCS 7566, 342-361.
- 🌈 [Wang07] Wang, Y., McIlraith, S., Yu, Y., Mylopoulos, J., “An Automated Approach for Monitoring and Diagnosing Requirements”, 22nd IEEE/ACM International Conference on Automated Software Engineering (ASE’07), Atlanta, November 2007, 293-302.
- 🌈 [Yu11] Yu, E., Giorgini, P., Maiden, N., Mylopoulos, J. (eds.) *Social Modeling for Requirements Engineering*, MIT Press, ISBN: 978-0-262-24055-0, January 2011.