KIT

Karlsruhe Institute of Technology

Optimization Approaches for Self-Adaptive Systems

Tim Engbrocks | July 01, 2021

Table of contents

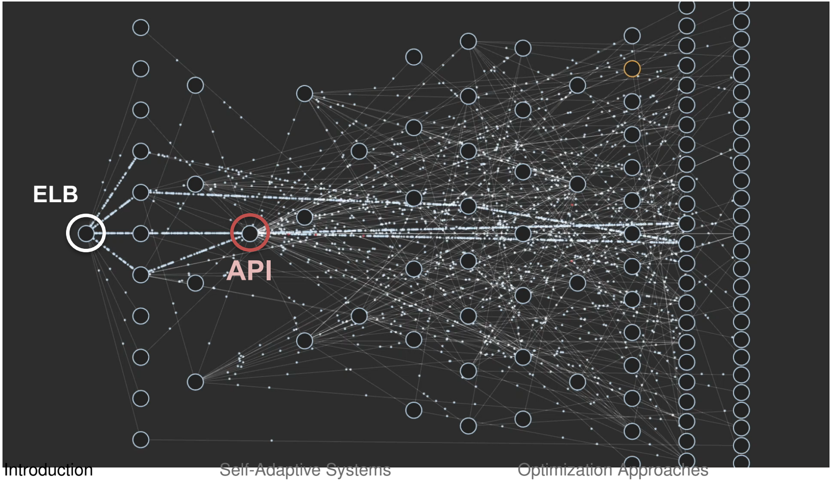
1. Introduction
2. Self-Adaptive Systems
3. Optimization Approaches
4. Classification Proposal
5. Conclusion

"The looming software crisis" - IBM, 2001

The complexity of software is constantly increasing.

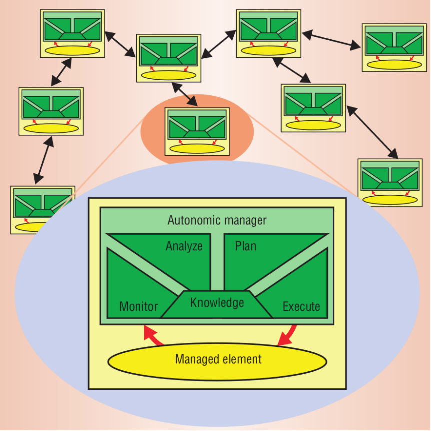
- Modern language models:
  - GPT-3 by OpenAI: 175 billion parameters 2020
  - Turing-NLG by Microsoft: 17 billion parameters 2020
- Tesla Autopilot AI predicts 10,000 parameters 2021

Netflix Microservice Architecture



Josh Evans, CTO @ Netflix, QCon 2016 2016

The Solution: Autonomous Management



Kephart and Chess 2003

Definition

Self-Adaptive System

A system which autonomously manages its self by:

- *Monitoring* its environment,
- *analyzing* changes
- *planning* adaptations and
- *executing* adaptations

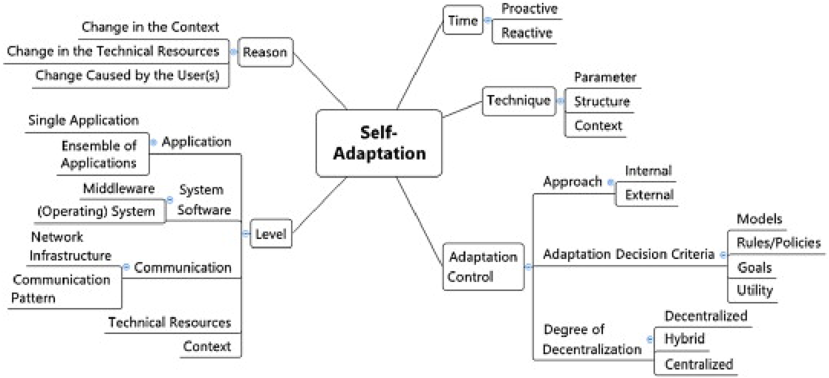
Example: Online store

- Scenario: System administrator of a commercial web service.
- Common task: Update system parameter X based on metric Y.

This can scenario can benefit from a Self-Adaptive System:

- Generalized adaptation rule: If metric Y crosses threshold Z: Update system parameter X.
- Example: If the server load gets too high: Start a new server instance.

Classifying Self-Adaptive Systems



Krupitzer et al. 2015

Limitations of Self-Adaptive Systems

Uncertainty

Environment changes ⇒ Unexpected adaptation results

Optimizations for Self-Adaptive Systems

There is one main approach for optimizing Self-Adaptive Systems that is being used:

- Dynamically update adaptation rules.

In most cases machine learning is used to generate new adaptation rules or to update them.

But other aspects can be optimized as well:

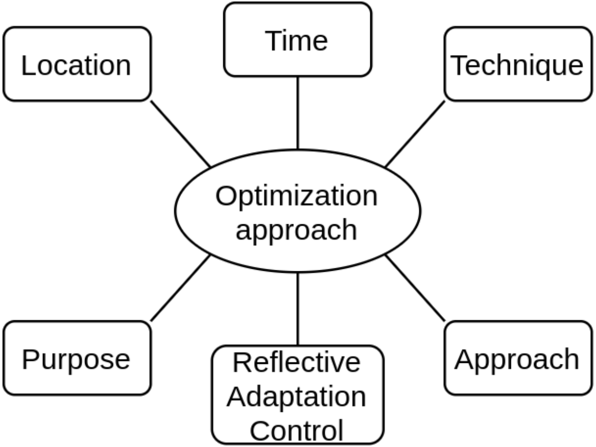
- Adaptation Control: How
- Level: Where
- Technique: What

Developing a classification

The classification is based on three concepts:

- Reflective computation (Danny Weyns and Andersson 2012)
- The 5W+1H questions (Salehie and Tahvildari 2009)
  - Who, What, Where, When, Why and How
- The taxonomy for Self-Adaptive Systems (Krupitzer et al. 2015)

Classification proposal



Classification proposal - Location

Location

- Adaptation Control
- Level
- Technique

Classification proposal - Time

Time

- Design time
- Run time / Online phase
- Offline phase

Classification proposal - Technique

Technique

- Rules / Policies
- Knowledge
- Adaptation Technique
- Level

Classification proposal - Purpose

Purpose

- Update Knowledge
- Apply Rules / Policies
- Goal Satisfaction
- Utility max-/minimization

Classification proposal - Approach

Approach

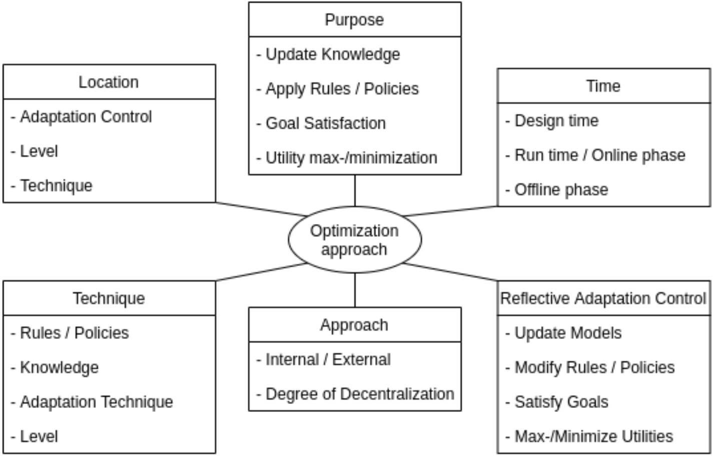
- Internal / External
- Degree of decentralization:
  - Fully centralized
  - Fully decentralized
  - Hybrid

Classification proposal - Reflective Adaptation Control

Reflective Adaptation Control

- Update Models
- Modify Rules / Policies
- Satisfy Goals
- Max-/Minimize Utilities

Classification proposal



Conclusion

This field still requires further research:

- The proposed classification has to be applied to existing approaches. This could verify the classification and give a structured overview of the field.
- Different approaches need to be tested. For example: Optimization Approaches which focus on the Level or Technique.
- Most approaches that are currently used are highly domain specific. This applies to Self-Adaptive Systems and their Optimization Approaches.

Literatur

Sam Malek Danny Weyns and Jesper Andersson. “FORMS: Unifying Reference Model for Formal Specification of Distributed Self-Adaptive Systems”. In: *ACM Trans. Auton. Adapt. Syst.* 7.1 (May 2012). ISSN: 1556-4665. DOI: 10.1145/2168260.2168268.

Josh Evans. *Josh Evans, QCon 2016*. Accessed: 2021-07-01. 2016. URL: https://www.infoq.com/presentations/netflix-chaos-microservices/.

J.O. Kephart and D.M. Chess. “The vision of autonomic computing”. In: *Computer* 36.1 (2003), pp. 41–50. DOI: 10.1109/MC.2003.1160055.

Christian Krupitzer et al. “A survey on engineering approaches for self-adaptive systems”. In: *Pervasive and Mobile Computing* 17 (2015). 10 years of Pervasive Computing’ In Honor of Chatschik Bisdikian, pp. 184–206. ISSN: 1574-1192. DOI: 10.1016/j.pmcj.2014.09.009.

Microsoft. *Turing-NLG by Microsoft*. Accessed: 2021-07-01. 2020. URL: https://www.microsoft.com/en-us/research/blog/turing-nlg-a-17-billion-parameter-language-model-by-microsoft/.

Nvidia. *OpenAI Presents GPT-3, a 175 Billion Parameters Language Model*. Accessed: 2021-07-01. 2020. URL: https://developer.nvidia.com/blog/openai-presents-gpt-3-a-175-billion-parameters-language-model/.

Mazeiar Salehie and Ladan Tahvildari. “Self-Adaptive Software: Landscape and Research Challenges”.