TECHNISCHE UNIVERSITÄT DRESDEN

ResUbic

QualiTune

COOL Software

Faculty of Computer Science, Institute of Software- and Multimedia-Engineering, Software Technology Group

Runtime Variability Management for Energy-Efficient Software by Contract Negotiation

Sebastian Götz, **Claas Wilke**, Sebastian Cech, Uwe Aßmann

Models@runtime 2011, Wellington, NZ
October, 17th 2011

Europa fördert Sachsen. SACHSEN
ESF Europäischer Sozialfonds
Bundesministerium für Bildung und Forschung
DFG Deutsche Forschungsgemeinschaft

DRESDEN concept Exzellenz aus Wissenschaft und Kultur

Context

- **Target:** Run **software** w.r.t. **non-functional requirements** in an **optimal** way.

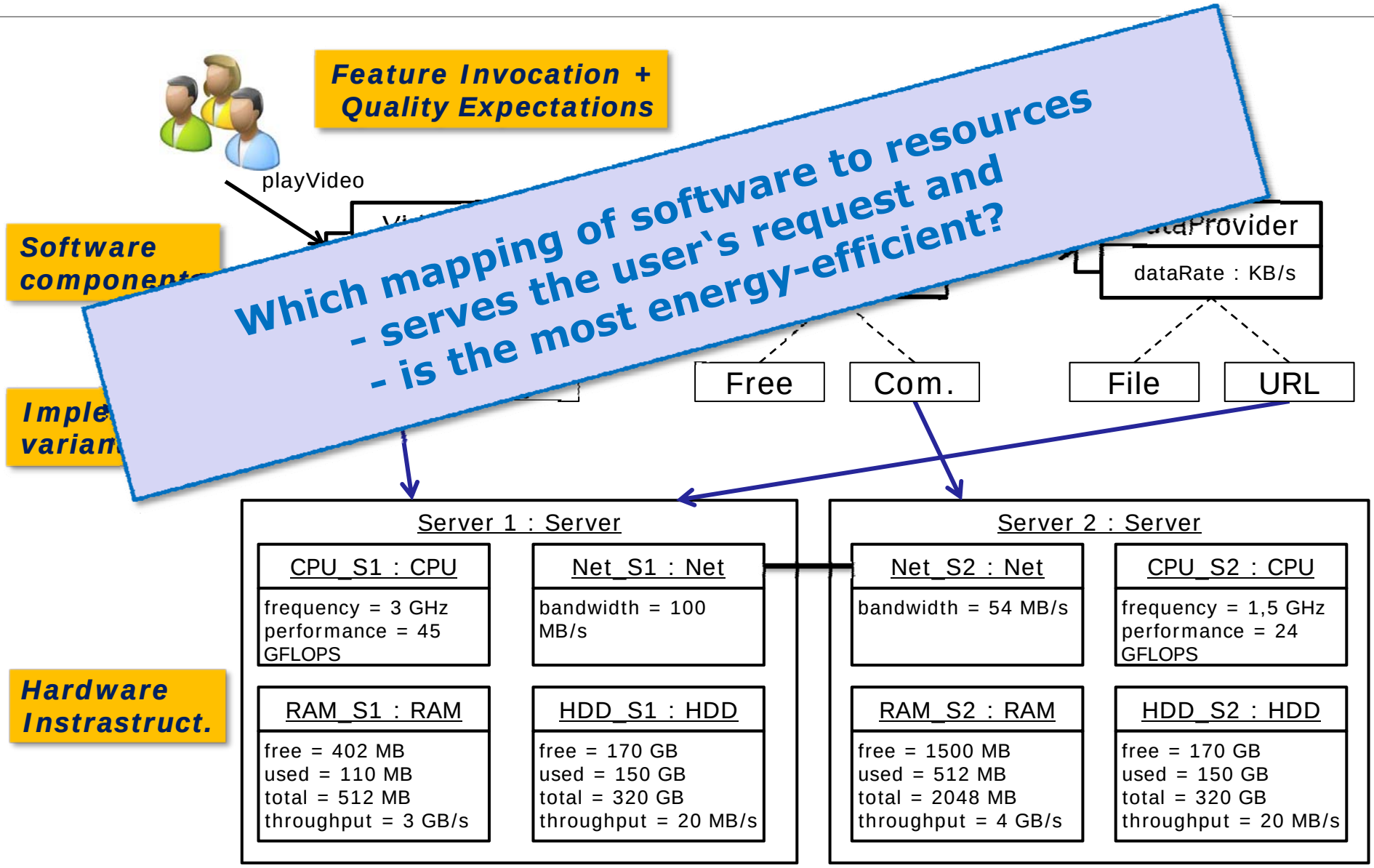
→ **Energy Auto-Tuning (EAT) [1,2]**

- **System knows**
  1. Its **hardware** (resources)
  2. Its **software** (components)
  3. Their **energy behavior**
- **System adapts**
  1. The **deployment of software** components
  2. On available **hardware**
  3. W.r.t. their **energy consumption**
  4. And **non-functional properties**

[1] S. Götz, C. Wilke, M. Schmidt, S. Cech, and U. Aßmann. Towards energy auto tuning. In Proceedings of First Annual International Conference on Green Information Technology (GREEN IT), pages 122–129. GSTF, 2010.

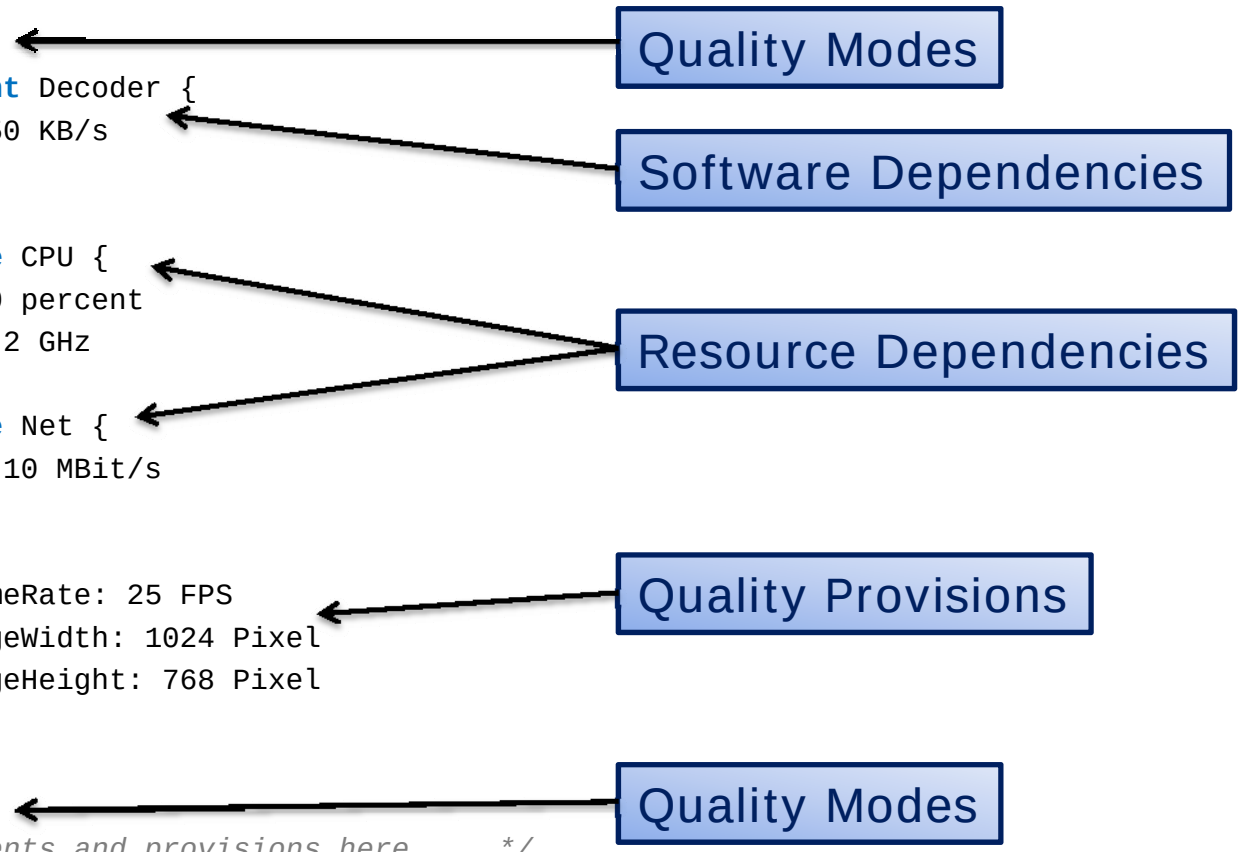
[2] S. Götz, C. Wilke, S. Cech, and U. Aßmann. Architecture and Mechanisms of Energy Auto-Tuning. To appear in: Sustainable Green Computing: Practices, Methodologies and Technologies, IGI Global, 2011

Example Application

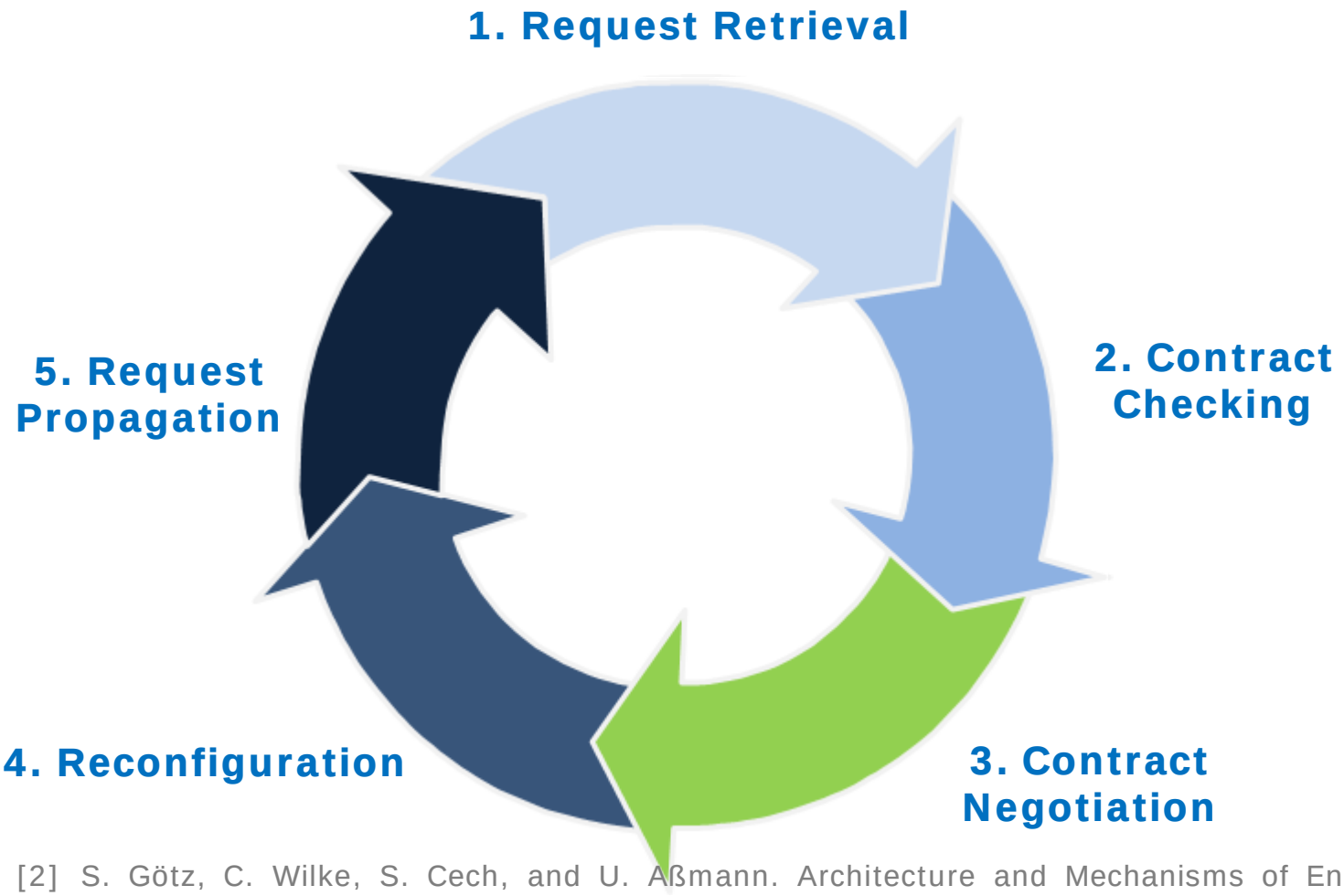


Quality Contracts for Components

```
1  contract VLC implements VideoPlayer {
2
3    mode highQuality {
4      requires component Decoder {
5        min dataRate: 50 KB/s
6      }
7
8      requires resource CPU {
9        max cpuLoad: 50 percent
10       min frequency: 2 GHz
11     }
12     requires resource Net {
13       min bandwidth: 10 MBit/s
14     }
15
16     provides min frameRate: 25 FPS
17     provides min imageWidth: 1024 Pixel
18     provides min imageHeight: 768 Pixel
19   }
20
21   mode lowQuality {
22     /* More requirements and provisions here ... */
23   }
24 }
```



Energy Auto-Tuning Process [2]



[2] S. Götz, C. Wilke, S. Cech, and U. Aßmann. Architecture and Mechanisms of Energy Auto-Tuning. To appear in: Sustainable Green Computing: Practices, Methodologies and Technologies, IGI Global, 2011

Solution at a Glance

- HW/SW Modeling using c**omponents** and **contracts**
- Application of **Contract Negotiation** to compute the **optimal system configuration** at runtime.

Definition: „System Configuration“

A selection of components and their mapping onto resources.

- **Transformation into an MILP**

  Definition: MILP (Mixed Integer Linear Program)

  Constraint System with **objective function** comprised of **linear constraints**, where some **variables** have to be **integers** (must not be reals).

- **MILP consists of**
  - Variables
  - Constraints
  - Objective Function

MILP for Contract Checking

- **Four kinds of variables:**

1. **Base load**
   (e.g., baseload#Server1)
2. **Resource usage**
   (e.g., usage#Server1#RAM_[s1]#size)
3. **Implementation Mapping (Boolean variables, flags)**
   (e.g., b#FreeDecoder#fast#Server2)
4. **NFPs**
   (e.g., bitrate, throughput, framerate, …)

MILP for Contract Checking

- **Objective Function**:

$$\min\left(\sum\left(\text{weight}_{XYZ}\cdot\text{usage\#container}_X\#\text{resource}_Y\#\text{property}_Z\right)+\sum baseload\#server_i\right)$$

- **Selection criteria / mappings** → **Boolean** variables

```
  b#FileReader#file#Server2
+ b#FileReader#file#Server1
+ b#URLReader#url#Server2
+ b#URLReader#url#Server1
= 1.0;
```

Exactly one DataProvider on exactly one server

- 1 such constraint per software component type

MILP for Contract Checking

- **Three constraints per usage variable of HW component**
  - Upper bound (according to variant model)
    - `usage#Server1#RAM_[s1]#size <= 512.0;`
  - Lower bound (>= 0 or according to variant model)
    - `usage#Server1#RAM_[s1]#size >= 0.0;`
  - Requirements (from contracts)
    - **`usage#Server1#RAM_[s1]#size =`**

```
  512.0 * b#FreeDecoder#fast#Server1   0 or 1
+ 256.0 * b#FreeDecoder#slow#Server1
+ 128.0 * b#CommercialDecoder#slow#Server1
+ 512.0 * b#CommercialDecoder#fast#Server1
+ 1536.0 * b#CommercialDecoder#ultrafast#Server1;
```

- **Baseload constraints**
  - $\text{baseload\#server}_i = \text{b\#impl}_x\text{\#mode}_y\text{\#server}_i$

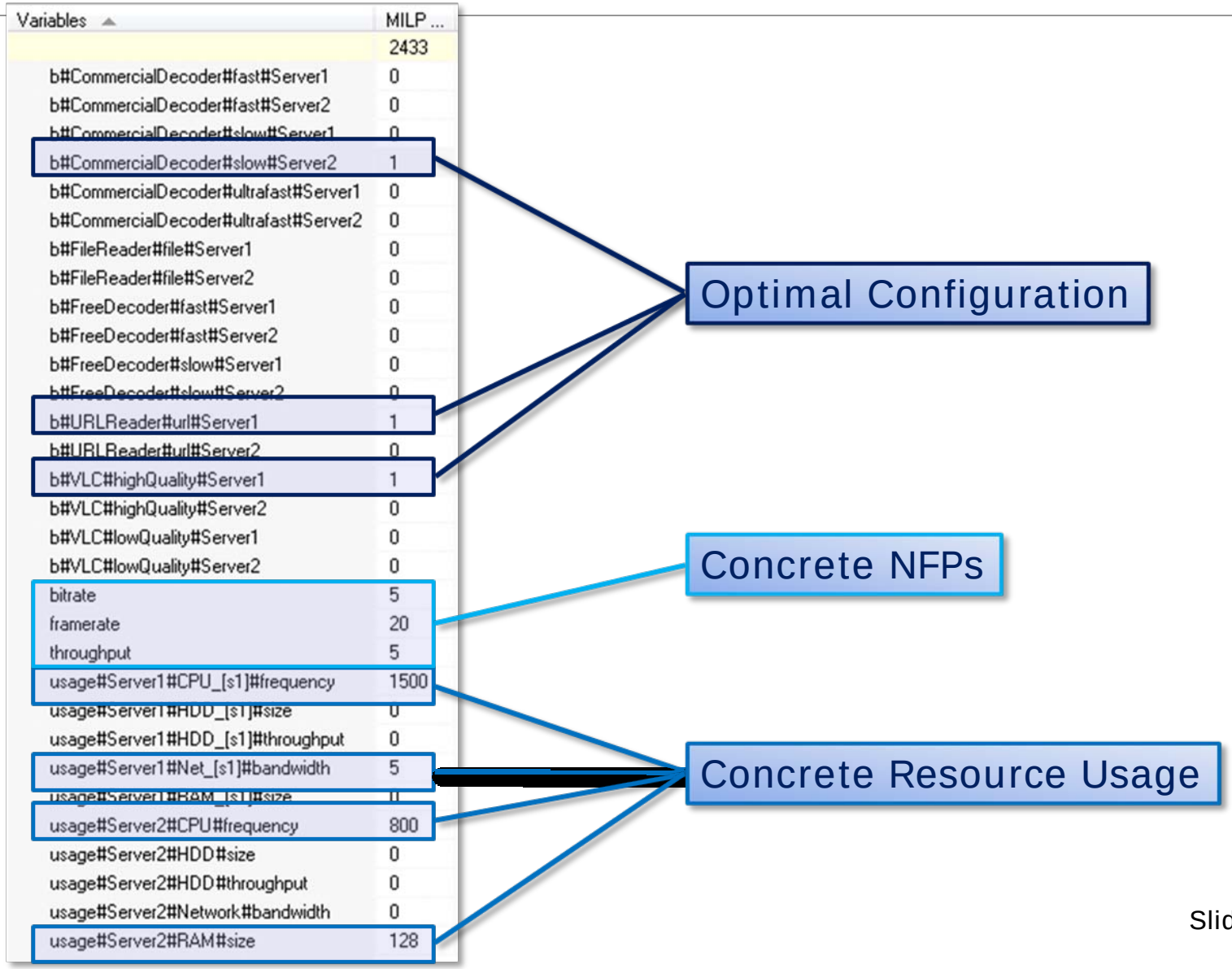
MILP for Contract Checking

- **Constraints for SW component NFPs**
  - **throughput =**

```
  5.0 * b#URLReader#url#Server1
+ 20.0 * b#FileReader#file#Server2
+ 5.0 * b#URLReader#url#Server2
+ 20.0 * b#FileReader#file#Server1;
```

- **User Request** reflected by constraint, too
  - `framerate >= 20.0;`

MILP for Contract Checking



Future Work

1) Resource usage

- Improve resource utilization / energy relation
- Introduce time-dependant resource utilization models

2) Case studies

- VideoServer, StockTracking and further scientific case studies
- Industrial case study with a green enterprise application planning (EAP) system

Summary

- Energy Auto-Tuning **requires contract negotiation**
- Goal: select the **optimal variant** w.r.t. **user demands** **and** **energy consumption**
- Solution: **MILPs** can be used **for contract negotiation**
  - Are generated from models@runtime
  - Result includes additional information
    - Concrete resource usage of variant → resource control (e.g., setting the CPU frequency)
    - Concrete provided qualities

TECHNISCHE UNIVERSITÄT DRESDEN

Thank You!

Contact

http://st.inf.tu-dresden.de/

claas.wilke@tu-dresden.de

Software Technology Group

ResUbic
www.resubic.org

Qualitune
www.qualitune.org

COOL Software
www.cool-software.org

Europa fördert Sachsen. SACHSEN
ESF Europäischer Sozialfonds
Bundesministerium für Bildung und Forschung
DFG Deutsche Forschungsgemeinschaft

18.10.2011

BACKUP

Related Work

Dynamic Variability in Complex, Adaptive Systems (DiVA) [3, 4]

- Management of dynamic adaptive systems
- Special focus on exponential growth of potential configurations
- Automated adaptation at runtime with
  - Goal-based optimization of NFPs
  - Rule-based system reconfiguration

Major difference to our approach

- Level of abstraction for non-functional property values
- DIVA symbolizes non-functional properties
  (e.g., LOW, MEDIUM, HIGH memory)
- We consider subsymbolic information
  (e.g., the actual value of free size of memory in MB)

→ allows deriving **finer-grained configurations**

[3] F. Fleurey and A. Solberg. A domain specific modeling language supporting specification, simulation and execution of dynamic adaptive systems. In: Proceedings of MODELS '09, pp. 606–621, Springer, 2009.

[4] B. Morin, O. Barais, G. Nain, and J.-M. Jézéquel. Taming dynamically adaptive systems using models and aspects. In: Proceedings of the 31st ICSE '09, pp. 122–132, IEEE, 2009.

Related Work

MADAM / MUSIC [5]

- Management of **mobile** dynamic adaptive systems
- MUSIC aims to maximize **user utility** using manually written utility functions.
  - Goal is to maximize **user satisfaction**
  - Dependencies between qualities are not considered

Major difference to our approach

- We aim to maximize user utility **while minimizing resource usage** based on **quality contracts** between user, soft- and hardware
  - We focus on **efficient** user satisfaction
  - Dependencies between qualities are considered
    - Tradeoffs are negotiated

[5] http://ist-music.berlios.de/ - MUSIC project homepage.

Literature

[1] S. Götz, C. Wilke, M. Schmidt, S. Cech, and U. Aßmann. **Towards energy auto tuning**. In: Proceedings of First Annual International Conference on Green Information Technology (GREEN IT), pp. 122–129. GSTF, 2010.

[2] S. Götz, C. Wilke, S. Cech, and U. Aßmann. **Architecture and Mechanisms of Energy Auto-Tuning**. To appear in: Sustainable Green Computing: Practices, Methodologies and Technologies, IGI Global, 2011

[3] F. Fleurey and A. Solberg. **A domain specific modeling language supporting specification, simulation and execution of dynamic adaptive systems**. In: Proceedings of MODELS ’09, pp. 606–621, Springer, 2009.

[4] B. Morin, O. Barais, G. Nain, and J.-M. Jézéquel. **Taming dynamically adaptive systems using models and aspects**. In: Proceedings of the 31st ICSE ’09, pp. 122–132, IEEE, 2009.

[5] http://ist-music.berlios.de/ - **MUSIC project homepage**.