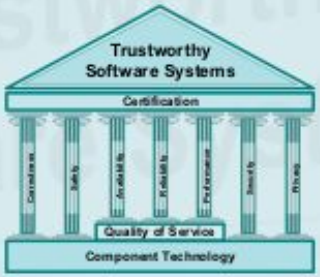
CARL VON OSSIETZKY universität OLDENBURG



graduate school
TRUSTSOFT
http://trustsoft.uni-oldenburg.de

Availability Simulation of Peer-to-Peer Architectural Styles

Simon Giesecke, Timo Warns, Wilhelm Hasselbring

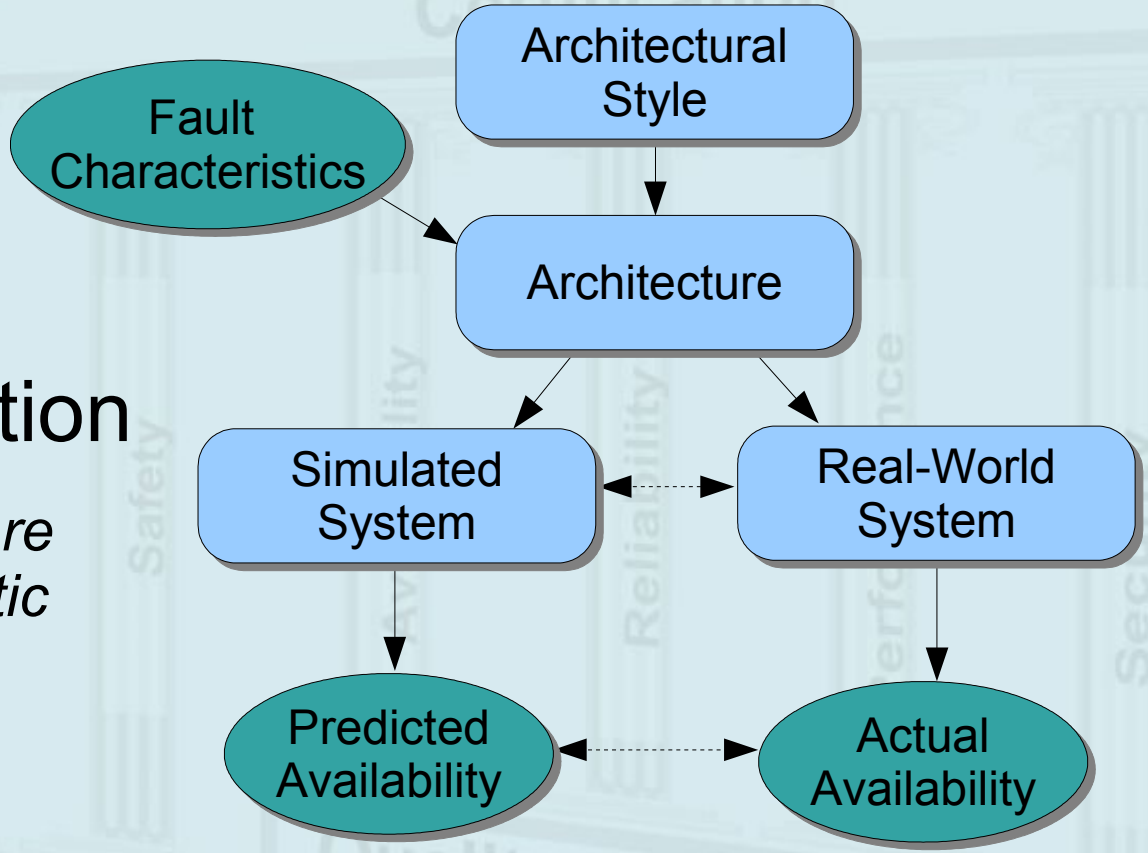
Referee: Timo Warns

Motivation

- Evaluation of availability of P2P services
- Specifics of P2P context impacting availability
  - Failure distribution of peers
  - Means of handling failures
  - Dynamic architecture / topology
- How to integrate these aspects?
  - Focus: Architectural Style

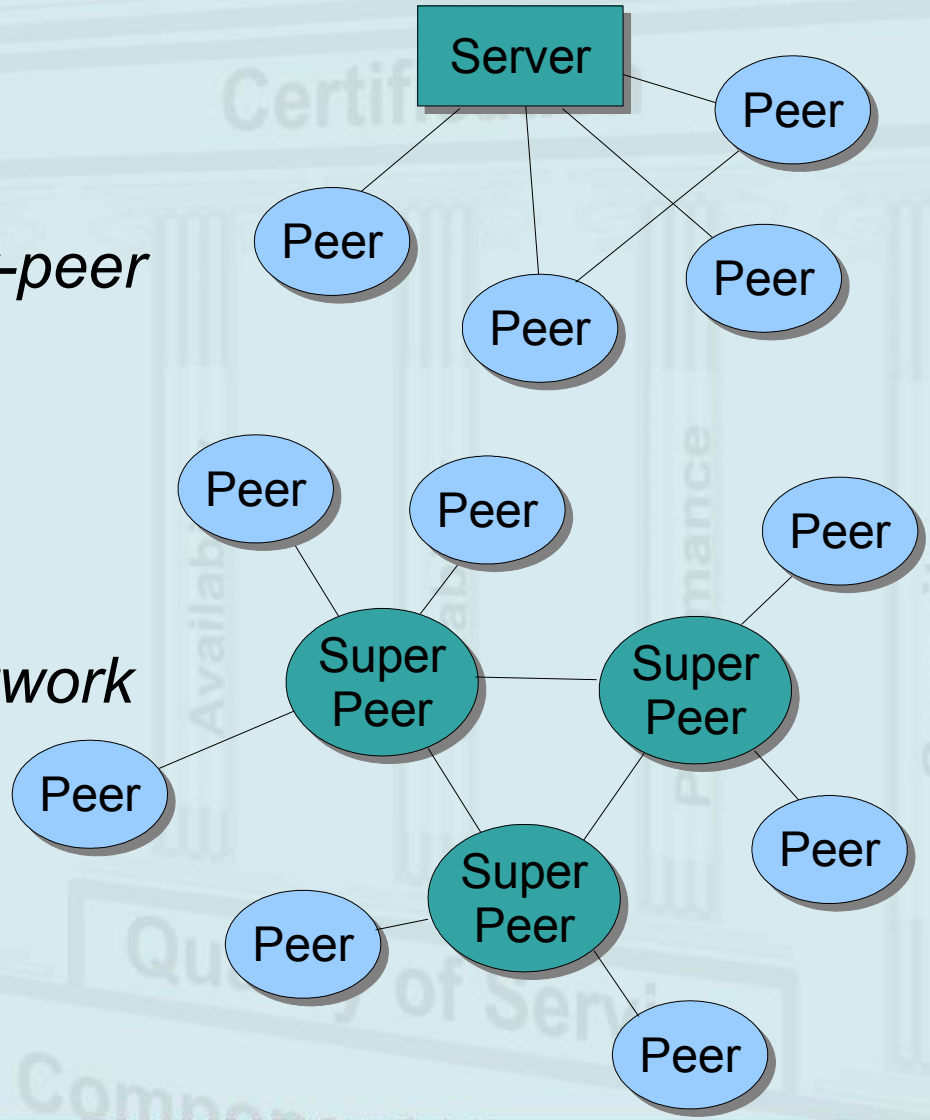
Approach

- Conceptual framework
  - P2P styles
  - P2P architectures
  - P2P systems
- Evaluation by simulation
  - *"most real-world systems are too complex to allow realistic models to be evaluated analytically"*
    Law and Kelton, 2000
  - Flexible



Peer-to-Peer Styles

- Classification scheme
  - Type of decentralization
    - *Decentralized, hybrid, super-peer*
  - Type of communication
    - *Direct, Indirect, Mediated*
  - Structural Characteristics
    - *Ring, Tree, Small-World Network*
- Rules for evolution
  - Joining / leaving of peers
- No formalisation yet



Architecture Description Model

- **Graph-based formalism** $A = (N, C, v, \lambda, \tau)$
  - N, C – Sets of nodes and connections
  - $v: C \rightarrow \{\{n_1, n_2\} \mid n_1 \neq n_2$ and n_1, n_2 in $N\}$ – Node function
  - $\lambda: N \rightarrow L$ – Labelling function
    - L is a set of node labels (e.g., “Peer”, “Server”, ...)
  - $\tau: T \rightarrow NC_T$ – Time mapping
- **τ describes evolution over time**
  - E.g., peer p participates at system from t_n to t_m
    => p is in image of τ for t in $[t_n, t_m[$

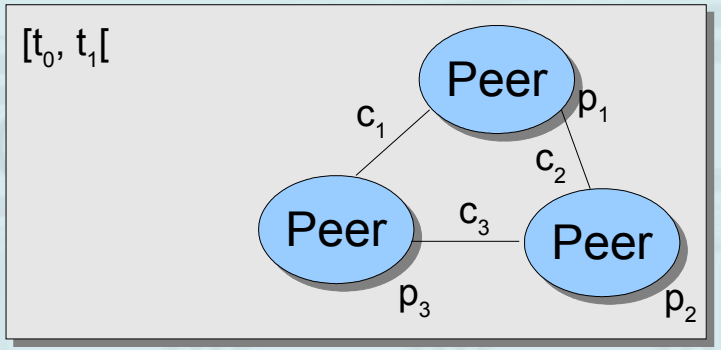
Example Description Model

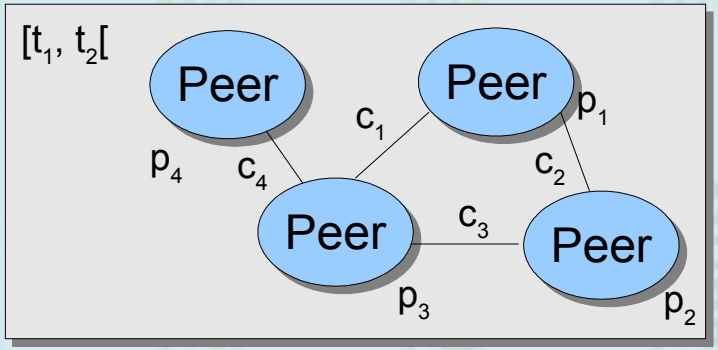
- $N = \{p_1, \ldots, p_4\}$
- $C = \{c_1, \ldots, c_5\}$
- $\lambda(n) = Peer$ for all n in N
- v:

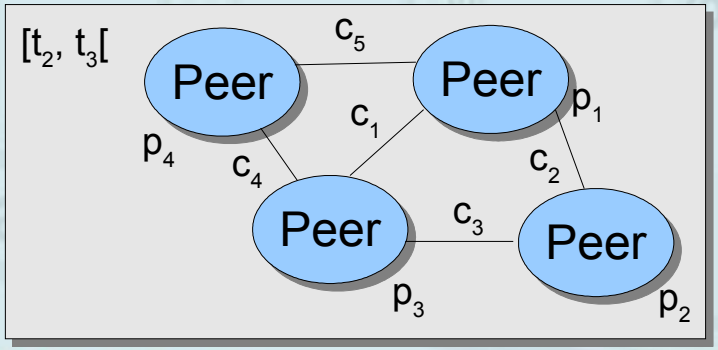
c	$v(c)$
c_1	$\{p_1, p_3\}$
c_2	$\{p_1, p_2\}$
c_3	$\{p_2, p_3\}$
c_4	$\{p_3, p_4\}$
c_5	$\{p_1, p_4\}$

- T:

T	NC_T
$[t_0, t_1[$	$p_1, \ldots, p_3, c_1, \ldots, c_3$
$[t_1, t_2[$	$p_1, \ldots, p_4, c_1, \ldots, c_4$
$[t_2, t_3[$	$p_1, \ldots, p_4, c_1, \ldots, c_5$

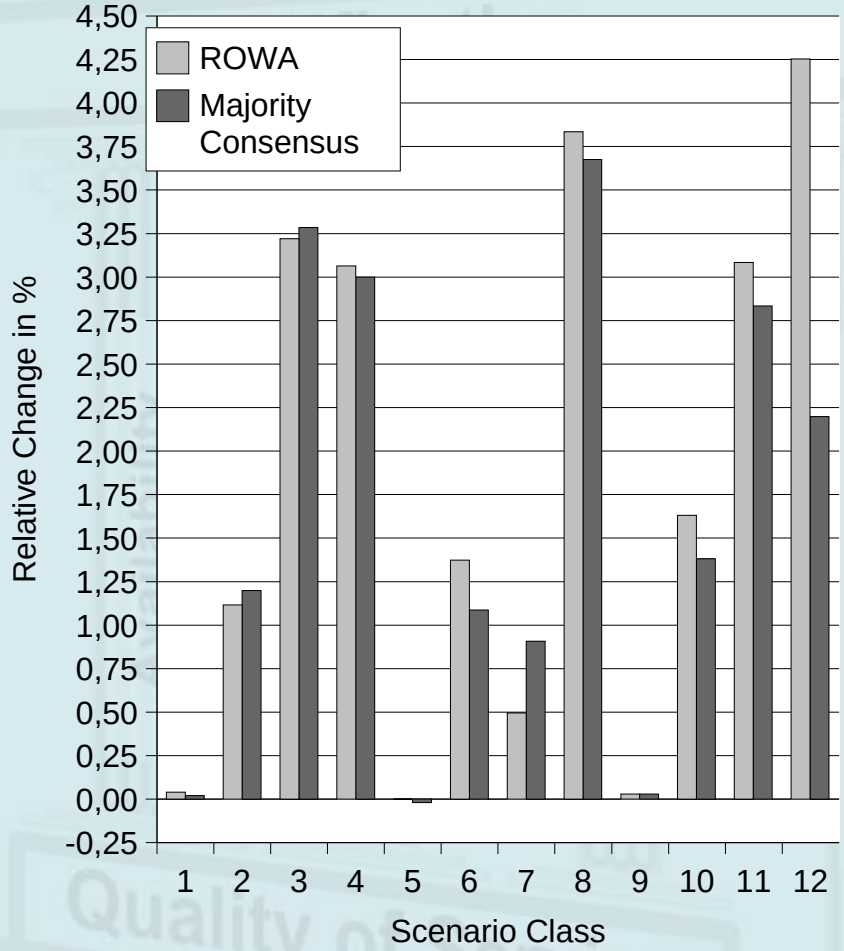






Simulation

- Prototype of simulator
  - Based on graph formalism
- Peer model
  - Derived from real-world system
  - Enhanced by classic replication strategies
- Evaluation of availability of replicated resources



Conclusions

- Conceptual framework
  - Evaluation of availability of P2P services
  - Architectural styles, architectures, systems
- Classification scheme for architectural styles
- Description model for P2P architectures
- Simulator prototype

Future Work

- Formalisation of architectural styles
  - Graph grammars?
  - Benefit: Automated creation of architectures
- Formalisation of peer model
  - Add peer model to input for simulation
  - UML?
- Development of improved simulator
  - Prototype used manually created architectures and one fixed peer model