



Workshops der wissenschaftlichen Konferenz  
Kommunikation in Verteilten Systemen 2011  
(WowKiVS 2011)

Kombination von Modellen zur Systemanalyse im  
Selbstmanagement-Kontext

Fabian Meyer

10 pages

# Kombination von Modellen zur Systemanalyse im Selbstmanagement-Kontext

Fabian Meyer

Hochschule RheinMain  
University of Applied Sciences  
Wiesbaden Rüsselsheim Geisenheim

**Abstract:** Für das Selbstmanagement von IT-Systemen spielt die MAPE Loop eine wichtige Rolle. Dabei werden in der Analysekomponente unterschiedlichste Verfahren zur Gewinnung neuer Informationen eingesetzt. Dieser Beitrag betrachtet die Kombination solcher IT-Management-Verfahren zur Gewinnung von höherwertigen Informationen aus Datenströmen. Konkret werden Ontologien und Bayes'sche Netze kombiniert. Dazu findet eine dynamische Ableitung eines Bayes'schen Netzes aus einem Systemmodell und ein Austausch von Werten zwischen den Modellarten statt. Für den Ableitungsvorgang werden spezielle Modellentitäten definiert, die dazu genutzt werden, Beziehungen von Komponenten des Systems zu beschreiben. Für die entwickelten Konzepte wird ein agentenbasiertes Softwaredesign entwickelt und auf Basis des OSGi-Frameworks umgesetzt. Abschließend wird der Ansatz in einer Fallstudie auf ein verteiltes Softwaresystem angewandt.

**Keywords:** Selbstmanagement, Bayes'sche Netze, Ontologien, MAPE Loop

## 1 Einleitung

Das Management von IT-Systemen stellt mit steigender Komplexität und Größe der Anwendungen eine fast unüberwindbare Aufgabe dar. Systeme müssen in einer Feedback-Schleife überwacht und analysiert werden, sodass eine automatisierte Anpassung der Laufzeitumgebung stattfinden kann. Dazu findet in der Regel eine Anwendung der *Monitor-Analyze-Plan-Execute (MAPE)* Loop [IBM06] statt. *Data Stream Mining* hat in diesem Kontext in den letzten Jahren immer stärker an Bedeutung gewonnen. Bei diesem Ansatz werden mit Hilfe von speziell entwickelten Methoden Datenströme untersucht und höherwertige Informationen aus den Messwerten abgeleitet. Diese Daten können in einem nächsten Schritt auf statistische Verfahren und mathematische Modelle angewandt werden. Das aus den Modellen gewonnene Wissen wird in einem Management-Schritt dazu genutzt, eine Bewertung des Systemzustands und gegebenenfalls eine Anpassungen am System vorzunehmen.

Im Bereich der Fehlervorhersage und Fehlertoleranz haben die *Bayes'schen Netze* eine starke Rolle bei der probabilistischen Analyse eingenommen. Sie ermöglichen eine Vorhersage von Fehlerwahrscheinlichkeiten unter der Voraussetzung eines beobachteten Systemzustands, sowie die Bestimmung der wahrscheinlichsten Ursache für einen Fehlerzustand. Dazu wird jede Zufallsvariable des Systems zu einem Knoten des Netzes. Jeder Knoten kann eine endliche Menge diskreter Zustände annehmen und kann über Kanten mit anderen Knoten verbunden werden, um

eine kausale Wirkung darzustellen. In Wahrscheinlichkeitsverteilungstabellen sind die Zustandswahrscheinlichkeiten der Knoten in Abhängigkeit der Zustände der Elternknoten festgelegt. Für unbeobachtbare Knoten kann daraus eine Wahrscheinlichkeit ihres Zustands berechnet werden. Bei der Nutzung eines probabilistischen Modells geht jedoch häufig vorhandenes Kontextwissen verloren, da es nicht abgebildet werden kann.

Exakte Zusammenhänge werden in der Regel in Systemmodellen abgebildet. Darin sind Komponenten und deren Beziehungen modelliert. Zur Darstellung von exaktem Wissen wurde im Kontext des *Semantic Web* die *Web Ontology Language (OWL)* [BM09] entwickelt. Sie basiert auf dem *Resource Description Framework (RDF)* [GK04] und ermöglicht es durch den Einsatz von Klassen, Instanzen und Beziehungen Systemstrukturen zu beschreiben. Über die reine Modellierung der Daten hinaus bietet sie einen Prozess zum Schluss neuer Fakten. Dabei wird die vorhandene Wissensbasis genutzt, um aus bekannten Fakten neue Fakten abzuleiten.

Sowohl Ontologien als auch Bayes'sche Netze weisen in ihrem jeweiligen Teilbereich Stärken auf, was das Ziehen neuer Schlüsse aus modelliertem Wissen betrifft. Ziel dieser Arbeit ist es daher, die beiden Verfahren in einem gemeinsamen Modell zu kombinieren, aus dem ein Mehrwert an Informationen gegenüber den einzelnen Modellen gewonnen werden kann. Das daraus resultierende Modell wird auf einen Datenstrom angewandt, um neues Wissen über ein zu überwachendes System zu gewinnen. Dieser Analyseprozess ist in eine Feedback-Schleife eingebettet, sodass gewonnenes Wissen genutzt werden kann, um automatisierte Entscheidungen zur Anpassung des Systems zu treffen.

Im nächsten Abschnitt werden zunächst verwandte Arbeiten vorgestellt. Darauf aufbauend werden die Konzepte, die Architektur und die Umsetzung des entwickelten Analysesystems vorgestellt und auf eine Fallstudie angewandt. Abschließend wird eine Zusammenfassung gegeben und auf zukünftige Arbeiten eingegangen.

## 2 Verwandte Arbeiten

### 2.1 Ontologien im IT-Management

Der Einsatz von OWL Ontologien im IT-Management hat bisher keine große Verbreitung gefunden, einige Arbeiten betrachten jedoch bereits mögliche Anwendungsfälle. In [VVB04] wird die Abbildungsmöglichkeit anderer Modellierungssprachen auf OWL analysiert. Dabei werden Entitäten und Relationen aus SMIV2, GDMO und MOF/CIM auf ein gemeinsames OWL-Modell abgebildet, sodass eine übergreifende Repräsentation geschaffen wird.

In [VV06] wird OWL auf ein Netzwerk-Management-Problem angewandt. Dabei findet eine Beschreibung der Infrastruktur in einer Ontologie statt. Beim Ausfall einer Ressource findet durch den Einsatz der *Semantic Web Rule Language (SWRL)* [IH04] eine neue Ressourcenzuweisung statt.

Eine Möglichkeit zur hierarchischen Anordnung von Ontologien wird in [VGVB09] vorgestellt. Dabei entsteht ein pyramidenartiges Gebilde von Ontologien, bei dem Elemente auf unterster Ebene am allgemeinsten und auf oberster Ebene am konkretesten sind.

## 2.2 Bayes'sche Netze im IT-Management

Bayes'sche Netze werden im IT-Management meist zur Fehleranalyse und Fehlervorhersage genutzt. In [HE01] wurden die Netze dazu eingesetzt, anhand von Festplattencharakteristiken Ausfallwahrscheinlichkeiten zu bestimmen.

Eine Methode zur Voraussage von Software-Verlässigkeit wurde in [SCC<sup>+</sup>01] entwickelt. Dabei wurden auf Basis der *Unified Modeling Language (UML)* Annotationen an Modellen angebracht und in Use-Case-Diagrammen die Wahrscheinlichkeiten von Methodenaufrufen und in Sequenzdiagrammen die Zuverlässigkeiten von Komponenten definiert. Anschließend fand eine Überführung der Modelle in ein Bayes'sches Netz statt, in dem Aussagen über die Gesamtzuverlässigkeit des Systems beim Austausch einzelner Komponenten gemacht werden konnten.

In [Bai05] wurde ebenfalls ein Ansatz verfolgt, bei dem mit Hilfe von Bayes'schen Netzen Aussagen über die Zuverlässigkeit von Software gegeben werden kann. Dazu wird ein spezielles Markov-Bayes-Netz genutzt, mit dem es möglich ist, Aussagen über diskrete Zeit zu treffen. Der Ansatz nutzt die in [Mus93] definierten betrieblichen Profile als Eingabe und führt ein Training der Wahrscheinlichkeitsverteilungen anhand von Softwaremetriken durch.

## 2.3 Kombination von Ontologien und Bayes'schen Netzen im IT-Management

In [DP04] wurde ein Ansatz entwickelt, der die Einbettung von Probabilistik in OWL vorsieht. Dabei werden spezielle Klassen und Properties definiert, die zur Beschreibung von bedingten Abhängigkeiten und Wahrscheinlichkeitsverteilungstabellen dienen. Diese können von speziellen Reasonern ausgewertet werden, um die Wahrscheinlichkeit der Klassenzugehörigkeit einer Instanz zu bestimmen.

Ein ähnlicher Ansatz wird in [CCLL03] bei der Definition von PR-OWL verfolgt. Dabei findet jedoch eine Spracherweiterung statt, durch die es möglich ist, die Probabilistik in das Modell einzubetten. Die Kombination von Ontologien und Bayes'schen Netzen für den Einsatz im IT-Management wurde jedoch von keinem Ansatz betrachtet.

# 3 Architektur

Die im folgenden Abschnitt vorgestellte Architektur bezieht sich auf die Analysekomponente eines Systems für Selbstmanagement. Für diese Komponente wurde ein neues Konzept entwickelt, bei dem das Grundprinzip ein Modellentwurf des zu analysierenden Systems in OWL und eine anschließende dynamische Ableitung eines Bayes'schen Netzes ist. Dabei dient die Monitor-Komponente der Schleife als Eingabe in das Modell und die Planungskomponente als Ausgabe.

## 3.1 Systemmodellierung

Zur Abbildung von komplexen Beziehungen zwischen Variablen müssen einige Strukturen definiert werden. Die folgenden Typen von Beziehungen sollen dabei darstellbar sein:

- Korrelationen, zum Ausdruck von korrelierenden Variablen, deren exakter Zusammenhang nicht bekannt ist.

- Funktionale Beziehungen, zur Definition von mathematischen Zusammenhängen zwischen einer Menge von Variablen.
- Kausale Beziehungen, zur Darstellung von kausalen Abhängigkeiten zwischen Ursachen- und Wirkungsvariablen.

Um diese drei Beziehungstypen bei der Modellierung zu unterstützen, muss die Definition einiger grundlegender Elemente stattfinden. In OWL ist es nicht möglich, Verknüpfungen zwischen variablen Datenwerten (Properties) herzustellen. In [VVB04] wurde dies durch einen Wechsel auf die RDF-Ebene umgangen, dabei geht jedoch die Unterstützung der meisten OWL-Modellierungswerkzeuge verloren. Daher wird in dieser Arbeit eine Kapselung der Werte vorgenommen, bei der jede Variable des Systems durch eine Instanz einer Variablenklasse dargestellt ist. Dadurch wird die Möglichkeit gewonnen, Properties an der Variableninstanz anbringen zu können, in denen sowohl der eigentliche Datenwert, als auch zusätzliche Informationen gespeichert werden können. Dies ermöglicht die Anbringung weiterer Charakteristiken (z.B. die Maßeinheit oder Zugriffsrechte), die aus anderen Modellierungssprachen wie dem *Common Information Model (CIM)* bekannt sind.

Es wird zwischen drei Variablentypen unterschieden:

- Enumerations, zur Definition von Variablen mit einem endlichen diskreten Zustandsraum. Durch die Referenzierung einer OWL-Klasse wird definiert, welche Individuen Zustände der Variablen sein können.
- Diskrete Variablen, zur Definition von Variablen mit ganzzahligem Zustandsraum.
- Kontinuierliche Variablen, zur Definition von Variablen mit rationalem Zustandsraum.

Für die Darstellung der angeführten Beziehungstypen in OWL bieten sich mehrere Möglichkeiten. So könnten funktionale Beziehungen durch spezielle Strukturen von Operatoren, in Form von Klassen, modelliert werden. Da die Unterstützung von Regeln durch SWRL jedoch bereits in OWL integriert ist, können diese genutzt werden, um solche Beziehungen zu beschreiben.

Sowohl Korrelationen, als auch kausalen Abhängigkeiten liegt kein exaktes Wissen des Zusammenhangs zugrunde. Sie beschreiben den Zusammenhang von Variablen des Systems, bieten jedoch keine nutzbare Berechnungsgrundlage. Da es in OWL kein Sprachkonzept zur Beschreibung solcher Beziehungen gibt, könnte eine externe Repräsentation gewählt werden. Die Kapselung der Variablen durch Klassen ermöglicht jedoch den Einsatz von sprachinternen Strukturen. So können spezielle Object Properties definiert werden, um die Modellierung von Abhängigkeiten zu ermöglichen. Dabei ist zu beachten, dass Korrelationen bidirektional sind und es somit keine Rolle spielt, welche Variable als Domain und welche als Range des Properties gewählt wird. Im Gegensatz dazu sind kausale Abhängigkeiten gerichtet, sodass die Ursache grundsätzlich die Range und die Wirkung die Domain des Properties sind.

Aus allen definierten Modellentitäten entsteht die in Abbildung 1 dargestellte Ontologie, die in das eigene Systemmodell importiert und dort genutzt werden kann.

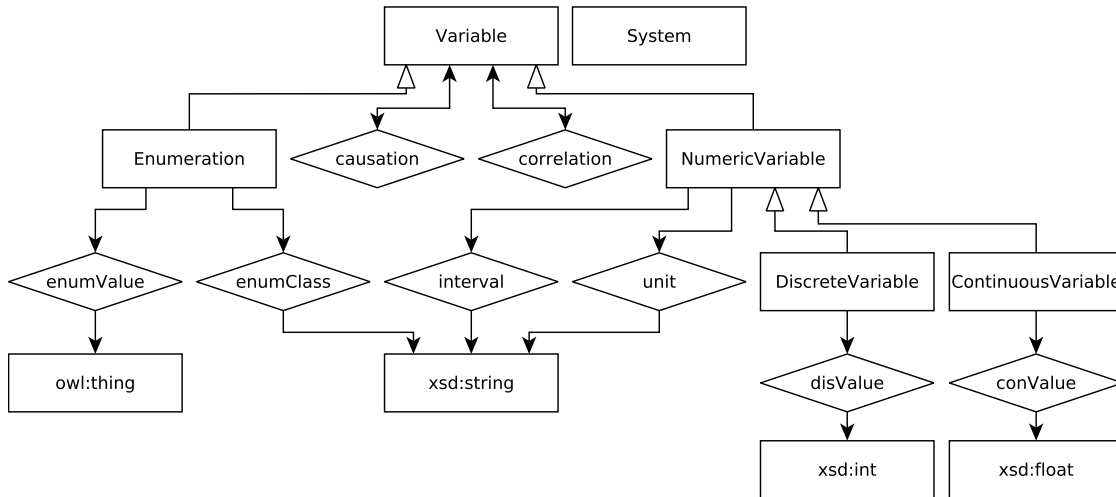


Abbildung 1: Struktur der bereitgestellten Modellentitäten.

### 3.2 Auswertung von Beziehungen

Zur Auswertung der drei Beziehungstypen werden unterschiedliche Verfahren eingesetzt. Da die funktionalen Beziehungen bereits in einer maschinenauswertbaren Sprache vorliegen, kann ein OWL Reasoner zu deren Analyse genutzt werden.

Für die Auswertung von kausalen Beziehungen wird das Bayes'sche Netz eingesetzt. Dazu findet eine Abbildung der Modellentitäten auf das Netz statt. Variablen im Systemmodell, die durch eine kausale Beziehung mit einer anderen Variablen verknüpft sind, werden zu Knoten im Bayes'schen Netz, die kausalen Beziehungen zu Kanten. Zur Bestimmung der Verteilungstabellen der Variablen wird ein Maximum-Likelihood-Verfahren eingesetzt, das auf historische Daten angewandt wird. Die Überführung von Werten aus der Ontologie in das Bayes'sche Netz gestaltet sich schwierig, da das Netz ein diskretes Modell ist. In modellierten Systemen gibt es jedoch in der Regel kontinuierliche Variablen, oder diskrete Variablen mit großem Wertebereich, wodurch eine Diskretisierungsmethode benötigt wird. Dazu werden den Variablen im Ontologiemodell Intervalle hinzugefügt, die eine Partitionierung vornehmen.

Nach der Überführung der Werte in das Netz findet eine Bestimmung der Wahrscheinlichkeiten der Zustände von unbeobachtbaren Variablen statt. Übersteigt die Wahrscheinlichkeit einen vordefinierten Schwellenwert, wird der geschlossene Wert in das Ontologiemodell eingefügt und die Variable mit einem Tag versehen, das sie als unscharf kennzeichnet. Dadurch ist es anderen Agenten, die die Ontologie als Wissensbasis nutzen, möglich zu erkennen, dass es sich nicht um einen realen Fakt handelt, sondern nur um eine Annahme.

Korrelationen lassen sich in keinem der beiden Ansätze auswerten. Sie stellen keinen echten funktionalen, sondern nur einen statistischen Zusammenhang dar und können daher nicht wie funktionale Beziehungen behandelt werden. Da Korrelationen gerichtet sind, ist es nicht möglich, sie in einem Bayes'schen Netz abzubilden. Deshalb werden Korrelationen anhand von historischen Daten auf ihren statistischen Zusammenhang untersucht und dieser in eine SWRL-Regel umgesetzt. Bei der Auswertung einer solchen Regel wird, wie bei kausalen Beziehungen

auch, ein Tag an der Variablen angebracht, um zu verdeutlichen, dass es sich bei dem geschlossenen Wert um keinen Fakt handelt.

Bei den eingesetzten Methoden werden neue Fakten anhand einer Menge von vorhandenen Fakten abgeleitet. In OWL existiert kein Mechanismus, um solche Abhängigkeiten zu verwalten und abgeleitete Fakten zu widerrufen, falls ein Ursachenfakt entfernt wurde. Daher ist im System ein gerichteter, azyklischer Graph vorgesehen, der alle Fakten und deren Ursachen beinhaltet. Bei der Löschung eines Fakts werden alle Kindknoten im Graphen gesucht und ebenfalls entfernt. Dadurch ist es nicht nötig, bei einer Änderung der Wissensbasis die gesamte Ableitungskette erneut zu durchlaufen.

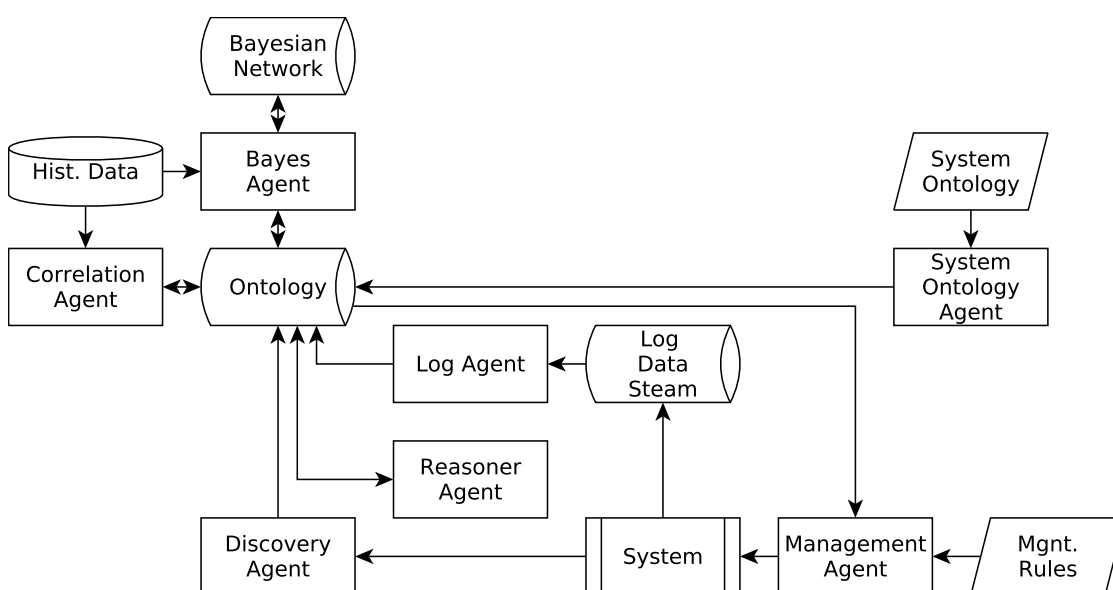


Abbildung 2: Datenfluss des agentenbasierten Systems

### 3.3 Softwaredesign

Die Umsetzung der Konzepte mündet in einem agentenbasierten Softwaredesign. Jeder Agent übernimmt dabei einen Teil der Funktionalität: So existiert je ein Agent für das OWL Reasoning mit Hilfe eines OWL Reasoners, zur Auswertung von Korrelationen durch die Analyse von historischen Daten, zur Verwaltung, Erzeugung und Auswertung des Bayes'schen Netzes, zur Verarbeitung des vom System erzeugten Log-Datenstroms, für das Discovery der Systemkomponenten des überwachten Systems, zum Einfügen des Grundmodells in die Ontologie und zum Management des Systems anhand des im Analyseprozess geschlossenen Wissens.

In Abbildung 2 wird der Datenfluss zwischen Agenten und Wissensbasen dargestellt. Die zentrale Ontologie wird als gemeinsame Wissensbasis für alle Komponenten genutzt. Fakten können aus der Ontologie ausgelesen, neues Wissen kann hinzugefügt und altes Wissen entfernt werden.

Das Hinzufügen und Entfernen geschieht dabei über sogenannte Operationen, die auf der Wis-

sensbasis ausgeführt werden. Sie enthalten neben den Axiomen auch Informationen über den Ableitungsvorgang, sodass der Graph konstruiert werden kann. Das Bayes'sche Netz wird vom Bayes Agent genutzt, um kausale Beziehungen des Systemmodells abzubilden und auszuwerten. Dazu findet eine ständige Synchronisation zwischen Netz und Ontologie statt. Historische Daten stellen aufgezeichnete Log-Daten des Systems dar. Sie werden genutzt, um ein Training des Systems durchzuführen. So dienen sie zur Auswertung von Korrelationen durch Regressionsanalyse und zum Training des Bayes'schen Netzes mittels Maximum-Likelihood-Verfahren.

In der System-Ontologie ist das Modell des Systems verankert. Sie enthält die statische Struktur der Komponenten und dient als Grundgerüst der Wissensbasis. Die Management Rules sind Entscheidungsregeln, anhand derer der Systemzustand bewertet und angepasst wird.

Der Management Agent, der die Planungs- und Ausführungskomponente darstellt, schließt die Feedback-Schleife. Änderungen des Systems werden über den Log-Agenten in die Ontologie eingefügt (Monitor), die Agenten leiten neues Wissen ab (Analyze) und durch eine Menge von Management Rules wird eine Anpassung des Systems vorgenommen (Plan und Execute).

Für die Implementierung wurde OSGi als Service Framework genutzt. Jeder Agent stellt dabei ein Bundle dar, das eine Agentenschnittstelle implementiert und einen entsprechenden Service zur Verfügung stellt. Zudem wurden Schnittstellen eingeführt, die eine dynamische Erweiterung für die Verarbeitung von Log-Daten und das Discovery ermöglichen. Zum Zugriff auf die Ontologie wurde OWL-API, als OWL Reasoner Pellet und als Bayes'sches Netz WEKA genutzt.

## 4 Fallstudie

Das entwickelte System wurde in einer Fallstudie auf das IT-Management-System MIDAS der Firma Blue Elephant Systems angewandt. Das System basiert auf einem Enterprise Service Bus zur Kommunikation zwischen den verteilten Komponenten, ein entsprechendes Modell wurde gemeinsam mit einem Domänenexperten entwickelt (siehe Abbildung 3).

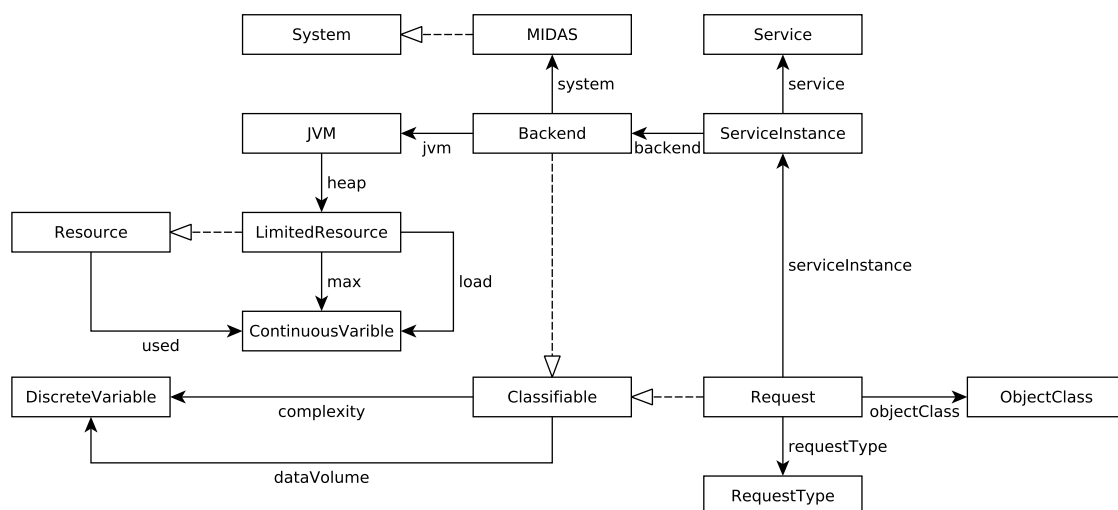


Abbildung 3: Ontologie des Systems der Fallstudie.



Bei der Verarbeitung eines Requests fällt eine gewisse Menge an Datenvolumen und Komplexität auf dem Knoten an. Dabei kann es vorkommen, dass der Heap-Speicher der Java Virtual Machine, auf der das Backend läuft, zur Neige geht und der Service nicht mehr erreichbar ist. Daher soll vor Ausführung von besonders komplexen oder datenintensiven Requests auf Modell-ebene geprüft werden, mit welchem Verhalten des Systems zu rechnen ist. Dazu wurde ein Modell entwickelt, bei dem sich die Gesamtkomplexität eines Backends kausal auf den verbrauchten Heap-Speicher der Java Virtual Machine auswirkt. Über diesen Verbrauch wird funktional die Auslastung bestimmt.

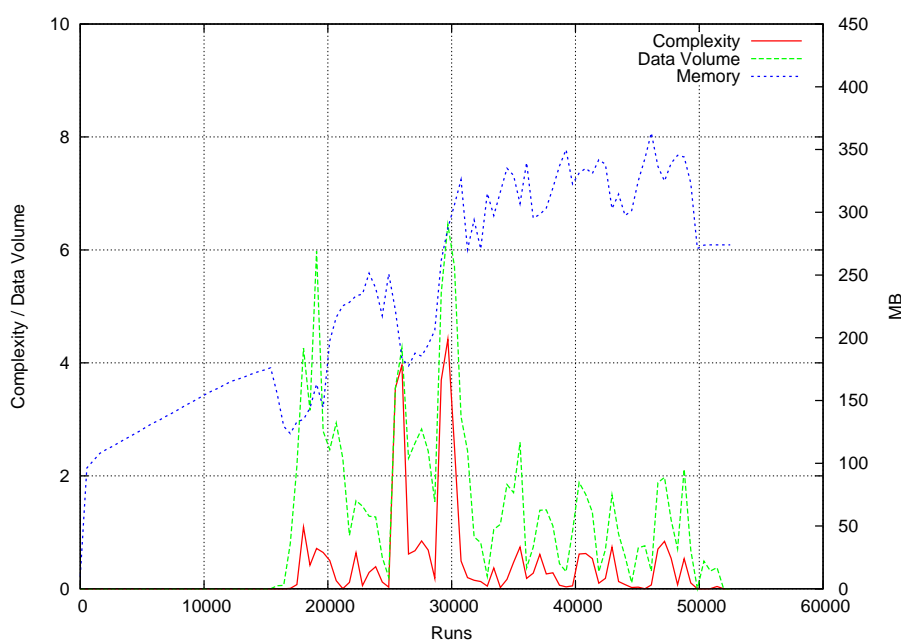


Abbildung 4: Kennwerte des betrachteten Systems.

Sowohl für das Training als auch für die Durchführung wurden echte Log-Daten des Systems verwendet. Dabei zeigte sich, dass eine Vorhersage des zu erwartenden Speicherverbrauchs zwar grundsätzlich möglich ist, die Qualität der Aussagen jedoch stark schwankt. Bei einer genaueren Analyse der verwendeten Daten (siehe Abbildung 4) zeigte sich, dass die Messwerte in den Spitzen zwar korrelieren, durch nicht modellierte Seiteneffekte jedoch zuviel Rauschen in den Daten entsteht.

## 5 Zusammenfassung

Beim Selbstmanagement von IT-Systemen müssen aus beobachtbaren Zuständen unbeobachtbare Zustände abgeleitet werden. Dazu muss eine Überwachung des Systems und eine anschließende Analyse der Daten stattfinden. Die so gewonnenen Informationen können anschließend genutzt werden, um Managemententscheidungen zu treffen.

Für den Analyseprozess wurde in dieser Arbeit ein Konzept entwickelt, um Ontologien und

Bayes'sche Netze zu kombinieren, um einen Mehrwert an Informationen aus Datenströmen gewinnen zu können. Für die Auswertung von kausalen Beziehungen wurde eine dynamische Ableitung eines Bayes'schen Netzes aus einer Ontologie genutzt, die auf speziellen Entitäten basiert.

Korrelationen wurden auf ihren statistischen Zusammenhang untersucht und durch SWRL-Regeln ausgedrückt. Funktionale Beziehungen wurden in SWRL beschrieben und von einem OWL Reasoner ausgewertet.

Die Umsetzung des Analysesystems fand mit einem lose gekoppelten Agentensystem statt, bei dem jeder Agent einen Teil der Funktionalität übernimmt und auf eine gemeinsame Wissensbasis zugreift. Darauf aufbauend wurde ein plattformspezifisches Softwaredesign auf Basis des OSGi Frameworks entwickelt.

Das entstandene Analysesystem wurde abschließend in einer Fallstudie auf ein ESB-basiertes Softwaresystem angewandt. Dabei konnte gezeigt werden, dass das entwickelte Konzept umsetzbar ist, die Qualität der Aussagen jedoch stark vom entwickelten Modell und den genutzten Trainingsdaten abhängt.

## 6 Ausblick

In einem nächsten Schritt soll eine Anwendung des Systems auf den Ambient-Assisted-Living-Kontext stattfinden. Dabei werden keine Datenströme analysiert, sondern Sensordaten einer instrumentierten Wohnung genutzt, um Schlüsse über den Kontext einer Person ziehen. Dieser Kontext soll genutzt werden um

- Anpassungen der Umgebung vorzunehmen (z.B. Steuerung von Geräten) und
- Gefahrensituationen zu erkennen und Hilfestellungen anzubieten (z.B. Verständigung eines Nachbarn).

Dazu findet eine Kombination des entwickelten Analysesystems mit anderen Verfahren aus dem DataStreamMiner- sowie aus dem Ambient-Assisted-Living-Projekt statt.

## Literatur

- [Bai05] C.-G. Bai. Bayesian network based software reliability prediction with an operational profile. *J. Syst. Softw.* 77(2):103–112, 2005.
- [BM09] A.-L. B. P. Boris Motik, Peter F. Patel-Schneider. OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax. October 2009.
- [CCLL03] P. Cesar, G. Costa, K. B. Laskey, K. J. Laskey. PR-OWL: A Bayesian Ontology Language for the Semantic Web. In *Center for Technology-Enhanced Learning, University of Missouri-Rolla*. 2003.
- [DP04] Z. Ding, Y. Peng. A probabilistic extension to ontology language owl. In *Proceedings of the 37th Hawaii International Conference On System Sciences (HICSS-37), Big Island*. 2004.

- [GK04] J. J. C. Graham Klyne. Resource Description Framework (RDF): Concepts and Abstract Syntax. February 2004.
- [HE01] G. Hamerly, C. Elkan. Bayesian Approaches to Failure Prediction for Disk Drives. In *In Proceedings of the eighteenth international conference on machine learning*. Pp. 202–209. Morgan Kaufmann, 2001.
- [IH04] H. B. S. T. B. G. M. D. Ian Horrocks, Peter F. Patel-Schneider. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. May 2004.
- [IBM06] IBM Corporation. An Architectural Blueprint for Autonomic Computing, Technical Whitepaper (Fourth Edition). June 2006.
- [Mus93] J. D. Musa. Operational Profiles in Software-Reliability Engineering. *IEEE Software* 10:14–32, 1993.
- [SCC<sup>+</sup>01] H. Singh, V. Cortellessa, B. Cukic, E. Gunel, V. Bharadwaj. A Bayesian Approach to Reliability Prediction and Assessment of Component Based Systems. In *ISSRE '01: Proceedings of the 12th International Symposium on Software Reliability Engineering*. P. 12. IEEE Computer Society, Washington, DC, USA, 2001.
- [VGVB09] J. E. L. de Vergara, A. Guerrero, V. A. Villagra, J. Berrocal. Ontology-Based Network Management: Study Cases and Lessons Learned. *J. Network Syst. Manage.* 17(3):234–254, 2009.
- [VV06] V. A. Villagra, J. E. L. D. Vergara. Ontologybased Policy Refinement Using SWRL Rules for Management Information Definitions. In *in OWL. In: Proc. 17th IFIP/IEEE International Workshop on Distributed Systems, Operations and Management (DSOM)*. Pp. 227–232. 2006.
- [VVB04] J. E. L. D. Vergara, V. A. Villagra, J. Berrocal. Applying the Web ontology language to management information definitions. *IEEE Communications Magazine* 42:68–74, 2004.