

Monitoring von Applikationen mit Hilfe eines Monitoring-Schichtenmodells

Version 1.1

von

Michael Lübben
(michael_luebben@web.de)

© by Michael Lübben 2012

Inhaltsverzeichnis

1	Einleitung.....	1
2	Grundlegendes.....	2
2.1	Abgrenzung.....	2
2.1.1	Monitoring-Tool Icinga.....	2
2.1.2	Messpunkte.....	2
2.1.3	Schnittstellen.....	2
2.2	Die Sicht.....	2
2.2.1	Sicht des Kunden oder der Umwelt:.....	2
2.2.2	Sicht der Geschäftsführung:.....	3
2.2.3	Sicht des Administrator/en:.....	3
2.3	Die Metriktypen.....	3
3	Das Monitoring-Schichtenmodell.....	4
3.0.1	Serverschicht:.....	4
3.0.2	Komponentenschicht:.....	5
3.0.3	Applikationsschicht:.....	5
3.0.4	Präsentationsschicht:.....	5
3.1	Sicht auf das Monitoring-Schichtenmodell.....	5
3.2	Monitoring der einzelnen Schichten.....	6
4	Monitoring von SLAs.....	8
5	Schnittstellen.....	9
5.1	Schnittstellen im Monitoring-Schichtenmodell.....	9
6	Messen von SLA's.....	11
7	Messintervalle.....	12
8	Abgrenzung SLA-Überwachung.....	13
9	Fazit.....	14
9.1	Abschliessendes.....	14
9.2	Dank und Änderungen.....	14

1 Einleitung

In diversen Literaturen, die es auf dem Markt zum Thema Monitoring gibt, wird man genug Lösungen dazu finden, wie z.B. Datenbanken, Applikationen usw. überwacht werden können. Anders sieht es aber bei der Frage aus, wie überwache ich meine Applikation richtig und auf Verfügbarkeit? Es kann heute fast alles überwacht werden. Da sind den Möglichkeiten und Fantasien fast keine Grenzen gesetzt. Doch was sind wirklich relevante Informationen? Und welche können z.B. für einen Service Level Agreement (SLA) herangezogen werden? Gerade im Bereich des Outsourcings und Outtaskings, ist das Reporten bzw. der Nachweis einer Einhaltung eines SLAs sehr wichtig. Doch wie realisiert man soetwas?

Im Rahmen meiner Diplomarbeit habe ich dazu Ideen zu einem Monitoring-Schichtenmodell entwickelt, welche ich in diesem Beitrag vorstellen möchte. Die Aufteilung einer Applikation in die Elemente Server, Komponenten und Applikation ist in dem Unternehmen in dem ich tätig bin, gang und gebe und somit nicht neues. Allerdings wird es nicht direkt in Schichten betrachtet, sondern eher als eine Aufteilung in Elemente. In diesem Schichtenmodell finden sich desweiteren auch die Clients einer Applikation in einer optionalen Schicht wieder. In der Präsentationsschicht. Diese Schicht ist wichtig für die Messung von SLA relevanten Daten.

Die Fragen die sich natürlich stellt, wozu ein Monitoring-Schichtenmodell?

- Übersichtliche Darstellung einer Applikation für die Einrichtung eines Monitorings
- Abgrenzung des Monitorings auf Intressenträger und somit Vermeidung von überflüssigen Überwachungen und sammeln nicht benötigter Information bzw. Daten
- Eventuelle Reduzierung von Fehlalarmen
- Schaffen von Motivation und Vertrauen der Mitarbeiter gegenüber dem Monitoring
- Reduzierung von Kosten und Aufwenden eines Monitorings und effizientere Nutzung von Ressourcen eines Monitoring-Systems

2 Grundlegendes

Bevor ich näher auf das Monitoring-Schichtenmodell eingehe, möchte ich noch ein paar grundlegende Dinge erläutern.

2.1 Abgrenzung

Zum ersten möchte ich eine kurz Abgrenzungen treffen. In diesem Beitrag beziehe ich mich zur Erläuterung des Monitoring-Schichtenmodells konkret auf die Applikation SAP-Portal. Ich denke diese Applikation wird den meisten vertraut sein. Natürlich kann diese Monitoring-Schichtenmodell auch auf andere Systeme bzw. Applikationen angewendet werden.

2.1.1 Monitoring-Tool Icinga¹

In diesem Beitrag beziehe ich mich auf das Monitoring-Tool Icinga. Das hier vorgestellte Monitoring-Schichtenmodell lässt sich natürlich auf alle anderen Monitoring-Tools anwenden. Ich möchte hier nicht beschreiben, wie eine Applikation im genaueren überwacht werden könnte, sondern abstrakt anhand des Monitoring-Schichtenmodell erläutern, was bei einem Monitoring z.B. im Bezug auf SLA's beachtet werden sollte.

2.1.2 Messpunkte

Desweiteren gehe ich nicht genauer auf Messpunkte ein. Natürlich sei an dieser Stelle angemerkt, dass dieses ein sehr wichtiger Punkt ist, aber Hauptthema dieses Beitrages ist und soll das Monitoring-Schichtenmodell sein.

2.1.3 Schnittstellen

Ein weiterer wichtiger Punkt sind Schnittstellen zwischen den einzelnen Schichten. Diese stellen die Kommunikationswege der einzelnen Schichten dar.

2.2 Die Sicht

Doch zuerst sollte die Frage gestellt werden, wen interessieren überhaupt die Informationen die aus dem Monitoring entstehen? Ich möchte das Ganze hier auf drei Interessensträger abgrenzen:

2.2.1 Sicht des Kunden oder der Umwelt:

Zu der Umwelt gehören z.B. Benutzer oder Mitarbeiter einer Applikation. Für einen Kunden beispielsweise ist in der Regel von Interesse, ob seine Applikation verfügbar ist und eventuell die Performance seiner Applikation. Desweiteren möchte ein Kunde von seinem IT-Dienstleister bei einem Ausfall früh genug darüber informiert werden. Es sollte nicht sein, das ein Kunde den IT-Dienstleister über einen Ausfall seiner Applikation informiert.

1 <http://icinga.org>

2.2.2 Sicht der Geschäftsführung:

Für die Geschäftsführung oder dem Management eines IT-Dienstleister ist zwar auch die Verfügbarkeit von Interesse. Aber noch wichtiger ist, ob z.B. ein im Vertrag definierter SLA für eine Applikation mit dem Kunden eingehalten wurde oder nicht. Desweiteren sind eventuell noch abzusehende Trends, wie z.B. die Zunahme von Festplattennutzung wichtig für Verhandlungen bei Vertragsanpassungen oder Vertragsverlängerungen.

2.2.3 Sicht des Administrator/en:

Eine ganz andere Sicht hingegen haben Administratoren bzw. der Verantwortliche einer Applikation. In erster Linie ist hier von Interesse, wo das Problem bei einem Ausfall liegt bzw. bei einem drohenden Ausfall schon im Vorfeld informiert werden. Hier sollte das Monitoring bei drohenden Ausfällen oder bei der Fehlersuche und -analyse unterstützen, um z.B. die Ausfallzeit möglichst gering zu halten oder dem Geschäftsführer bzw. Kunden über notwendige Maßnahmen zu informieren.

2.3 Die Metriktypen

Betrachtet man diese Sichten genauer, wird man erkennen das sich hieraus zwei elementare Metriktypen von Daten ergeben. Die Status- und Performancedaten. Die Abbildung 2.1 zeigt einen Ausschnitt einer Festplattenüberwachung aus der Überwachung von Icinga. Hier sind zum einen die Statusdaten (Current Status) und zum anderen die Performancedaten (Performance Data) wiederzufinden.

Current Status:	OK (for 277d 2h 6m 3s)
Status Information:	DISK OK - free space: / 4239 MB (58% inode=69%):
Performance Data:	/=3037MB;6132;6899;0;7666

Abbildung 2.1: Ausschnitt aus der Icinga Statusübersicht

Statusdaten:

Statusdaten spiegeln, wie der Name schon sagt, lediglich einen Status wieder. Diese können z.B. Warning, Critical oder Ok sein. Diese geben also in der Regel nur Informationen darüber, ob ein vorher definierter Schwellwert im Monitoring erreicht bzw. überschritten wurde. Statusdaten geben aber keine Informationen darüber, wie z.B. die Gesamtkapazität einer Festplatte ist und eignen sich somit auch nicht für eine Trendanalyse. Diese Daten können aber zur Darstellung einer Verfügbarkeit in einem SLA herangezogen werden.

Performancedaten:

Die Performancedaten hingegen geben mehr Informationen über die Performance. Diese können beispielsweise sein, der Füllgrad und die Gesamtkapazität einer Festplatte und die dazu definiert Schwellwerte. Die Performancedaten lassen sich somit besser für z.B. eine Trendanalysen heranziehen.

3 Das Monitoring-Schichtenmodell

In dem Monitoring-Schichtenmodell werden zwischen vier Schichten unterschieden. Der Server-, Komponenten-, Applikationsschicht und Präsentationsschicht, wie in Abbildung 3.1 zu sehen.

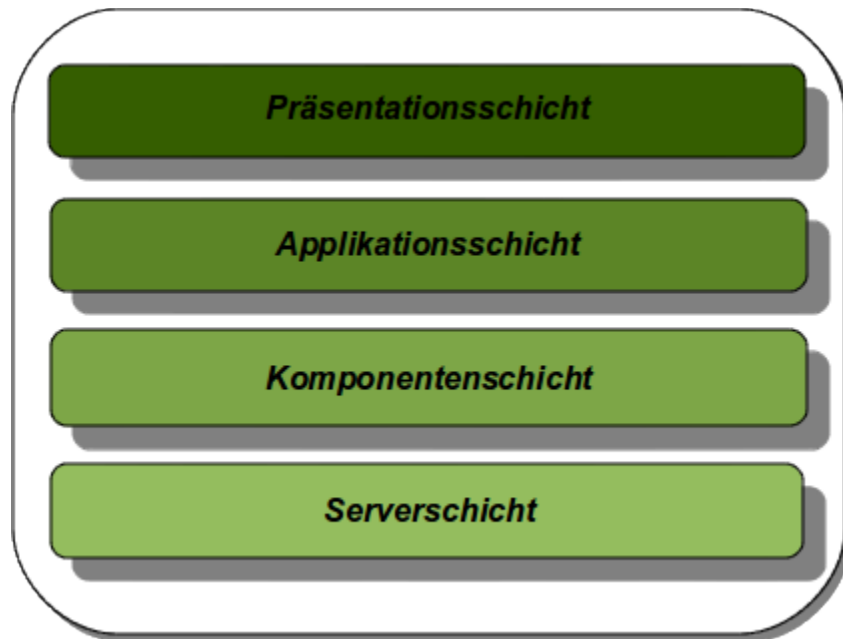


Abbildung 3.1: Monitoring-Schichtenmodell

3.0.1 Serverschicht:

Die Serverschicht ist die erste Schicht des Monitoring-Schichtenmodells und stellt die Basis für die Komponenten- und Applikationsschicht dar. Sie ist somit immer vorhanden und kann auch mehrfach in einer Darstellung auftauchen. Sie ist allerdings nach außen hin z.B. für einen Kunden nicht sichtbar. In dieser Schicht ist der Server mit seinem Betriebssystem zu finden. Hierbei kann es sich um einen physikalischen oder virtualisierten Server handeln. Abbildung 3.2 zeigt eine Verfeinerung der Serverschicht.

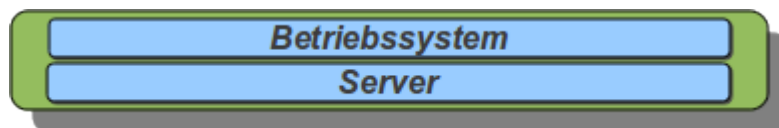


Abbildung 3.2: Verfeinerung der Serverschicht

Die Serverschicht kann aber auch weiter in ihre Umgebung hin unterteilt werden, wie die Abbildung 3.3 zeigt. So gehören dazu z.B. die Stromversorgung oder auch Klimatisierung. Je nachdem wie genau man ins Detail gehen möchte und was einem wichtig was Monitoring der Serverschicht erscheint. Hier sind den Möglichkeiten keine Grenzen gesetzt.



Abbildung 3.3: Verfeinerung der Serverschicht in ihre Umgebung

Abbildung 3.4 zeigt eine Darstellung eines Clusters. Hier ist es wichtig, dass die Applikationsadresse des Clusters als eigener Server dargestellt wird. Bei einer Überwachung ist z.B. nicht immer bekannt, auf welchem Node gerade die Applikation läuft. Desweiteren würde ein Ausfall eines Nodes nicht unbedingt zu einer Störung der Applikation führen. Daher ist es gerade bei der Überwachung einer Verfügbarkeit einer Applikation wichtig, die Applikationsadresse als eigenen Server darzustellen.

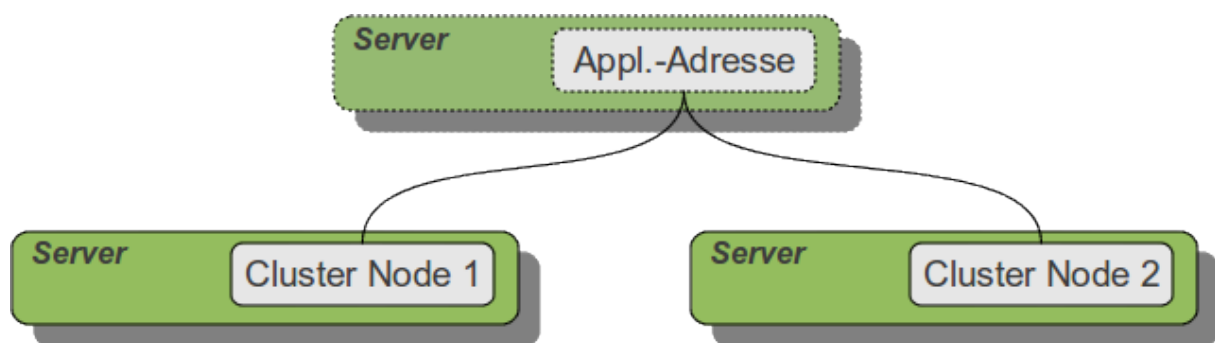


Abbildung 3.4: Darstellung eines Clusters auf der Serverschicht

3.0.2 Komponentenschicht:

Die Komponentenschicht ist die zweite Schicht des Monitoring-Schichtenmodells und liegt zwischen der Server- und Applikationsschicht. Auch diese Schicht kann mehrfach in einer Darstellung auftauchen. In der Komponentenschicht sind die Komponenten der Applikation zu finden. Diese Komponenten stellen einen Teil der gesamten Applikation dar und ist auch nach außen hin nicht sichtbar. Nicht jede Applikation muß zwingend eine Komponente haben.

3.0.3 Applikationsschicht:

Die Applikationsschicht ist die dritte und die oberste Schicht des Monitoring-Schichtenmodells. Diese Schicht ist nach außen hin immer sichtbar und immer nur einmal in einer Darstellung vorhanden. Diese Schicht ist z.B. relevant für die Überwachung eines SLAs, da diese die Sicht des Kunden repräsentiert. Die Applikationsschicht setzt entweder auf die Server- oder Komponentenschicht auf.

3.0.4 Präsentationsschicht:

Die Präsentationsschicht ist eine optionale Schicht und findet Anwendung in einer Client/Server-Architektur einer Applikation. In dieser Schicht befindet sich beispielsweise Clients wie ein Browser oder eine GUI mit Hilfe dieser auf einer Applikation gearbeitet werden kann.

3.1 Sicht auf das Monitoring-Schichtenmodell

Abbildung 3.5 zeigt eine gesamte Darstellung des Monitoring-Schichtenmodell und den

sichten der Intressensträger.

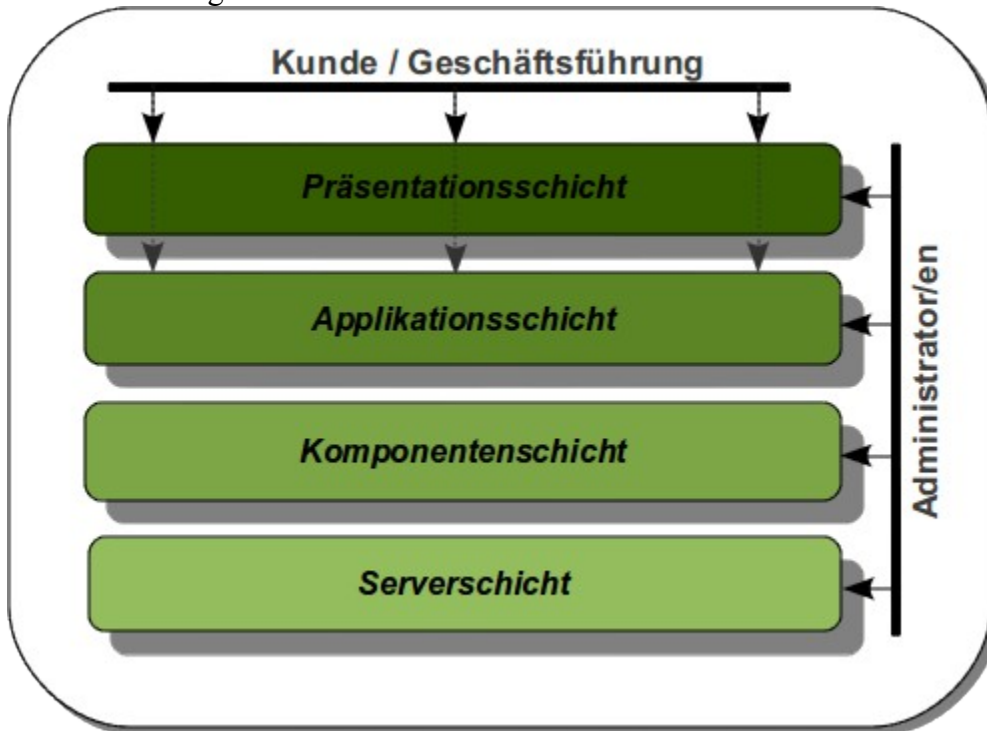


Abbildung 3.5: Monitoring-Schichtenmodell aus verschiedenen sichten

3.2 Monitoring der einzelnen Schichten

Die Abbildung 3.6 zeigt ein Beispiel, wie ein SAP-System im Monitoring-Schichtenmodell abgebildet werden kann. In diesem Beispiel wird davon ausgegangen, dass alles auf einem Server läuft.

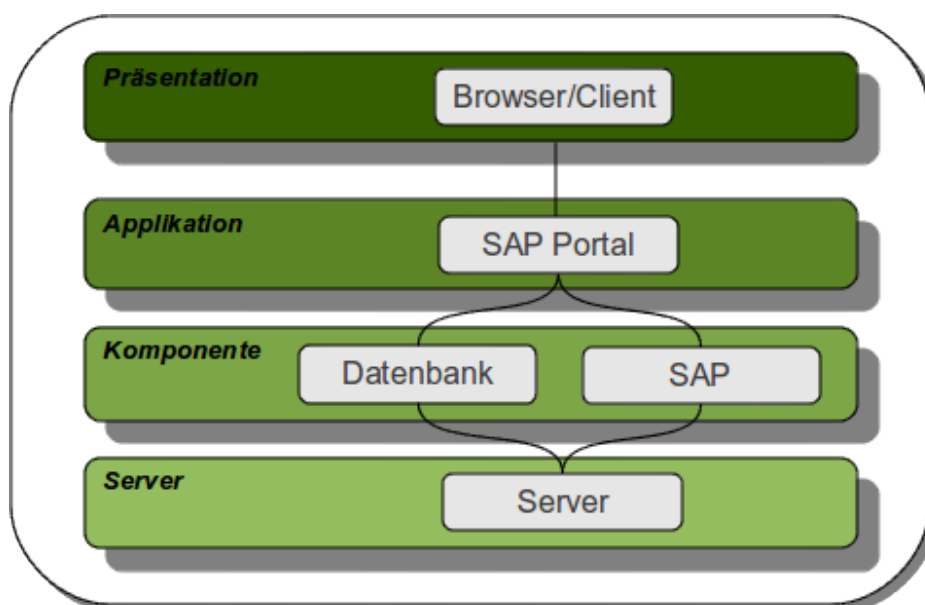


Abbildung 3.6: Monitoring-Schichtenmodell am Beispiel SAP

In der Serverschicht ist der Server mit seinem Betriebssystem zu finden. Die Komponenten Datenbank und SAP liegen in der Komponentenschicht und sind nach außen hin, wie die Serverschicht, für den Kunden nicht sichtbar. Das SAP-Portal stellt dabei die eigentliche Applikation dar, womit der Kunde arbeitet und welches nach außen hin für diesen sichtbar ist. In der Präsentationsschicht befinden sich der Client bzw. Browser, womit der Kunde auf die Applikation zugreift. Je nachdem was mit dem Kunden in einem SLA-Vertrag vereinbart wurde, findet ein SLA-Monitoring auf der Applikations- oder Präsentationsschicht statt.

Natürlich könnte auch die Komponente Datenbank z.B. auf einem anderen Server laufen. Klar sollte allerdings hierbei sein, dass die Serverschicht immer als Basis vorhanden ist. D.h. fällt ein Server aus irgendeinem Grund aus, ist somit auch immer eine Komponente oder Applikation davon betroffen die auf dem Server läuft. Ein Ausfall einer Komponente bedeutet dabei aber nicht zwingend einen Ausfall der Applikation.

In wie weit man eine Applikation in dem Monitoring-Schichtmodell abbildet, ist jedem selber überlassen. Dabei sei aber angemerkt, je genauer die Verfeinerung der Komponenten, wie z.B. das Einbeziehen von Netzwerkelementen (Switch, Router, etc.) ist, je schwieriger und unübersichtlicher kann es werden. Auch werden damit die Schnittstellen in einer solchen Darstellung in der Regel mehr. Es sollten daher also wirklich nur die relevanten Dinge einer Applikation dargestellt werden.

Die Abbildung 3.7 zeigt die Anwendung des Monitoring-Schichtenmodells auf eine grössere Umgebung mit mehreren Servern unter Einbeziehung der Sicht.

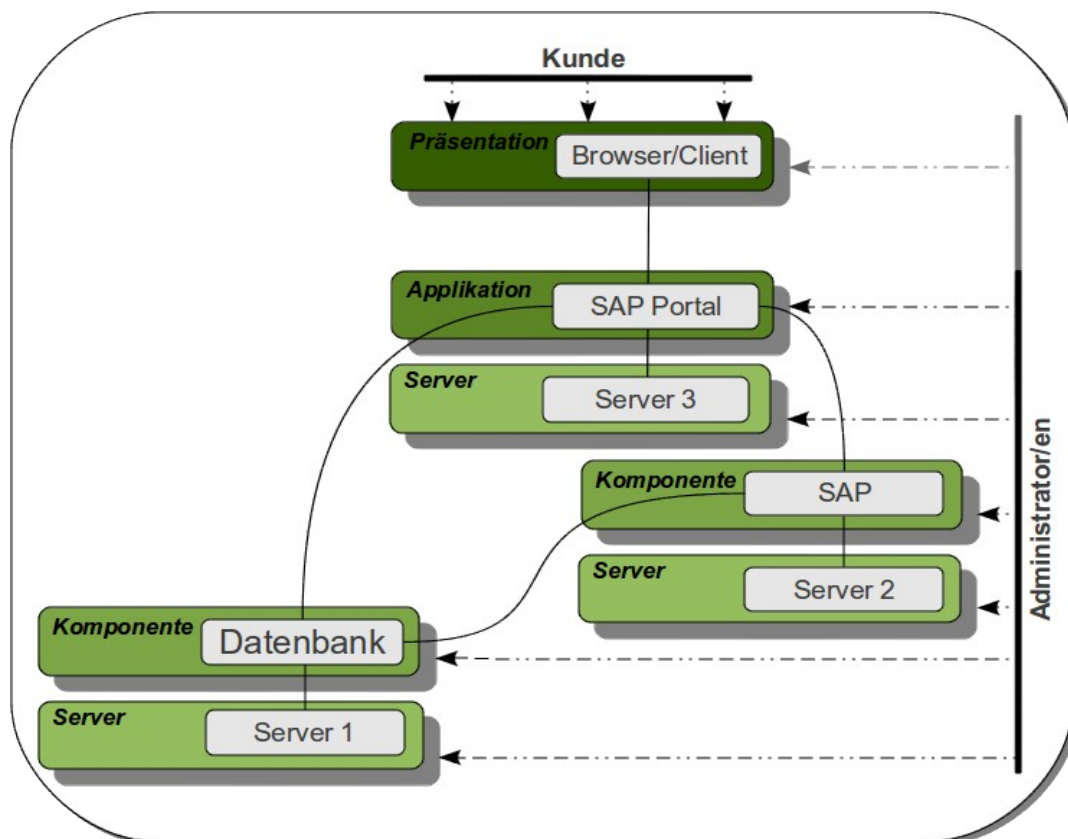


Abbildung 3.7: Anwendung des Monitoring-Schichtenmodells auf eine grössere Umgebungen

4 Monitoring von SLAs

Wie bereits erwähnt, wird in dem Monitoring-Schichtenmodell ein SLA immer auf der Applikations- oder Präsentationsschicht überwacht und ist immer nur eine Überprüfung (Check). Wichtig bei einer solchen Überwachung ist allerdings, dass auch das Zusammenspiel der Komponenten mit der Applikation bzw. Client überprüft wird. Damit ist nicht gemeint, dass Komponente einzeln überprüft werden und diese dann alle zu einer Überprüfung zusammenfasst werden, sondern ob die Applikation zusammen mit den Komponenten arbeitet. Auch bringt es sehr wenig zu prüfen, ob z.B. ein TCP-Port vom SAP-Portal erreichbar ist. Denn fällt beispielsweise die Komponente Datenbank aus, so ist eventuell der überwachte TCP-Port des SAP-Portals immer noch erreichbar. Auch würde z.B. eine fehlerhafte Berechtigungen innerhalb der Datenbank nicht unbedingt eine Störung der Komponente bedeuten. Somit würde eine solche Überwachung keine ausreichende Aussage über die Verfügbarkeit geben und wäre somit beispielsweise für einen SLA unbrauchbar. Es muss also eine Überprüfung sein, welche die gesamte Funktionalität der Applikation zusammen mit den Komponenten überprüft. Dieses könnte z.B. ein Login in das SAP-Portal mit einem Browser sein. Wie soetwas im genaueren z.B. mit Sahi realisiert werden kann, beschreibt Gerhard Lauber in seinem Buch Nagios - Das Praxisbuch: Open Source-Monitoring im Unternehmen².

2 ISBN-13: 978-3827328007 (Addison-Wesley)

5 Schnittstellen

Eine Schnittstelle dient zur Kommunikation zwischen den Schichten. Dabei müssen die Schichten nichts voneinander wissen. Lediglich wie eine Schicht die andere ansprechen muss, also über welche Schnittstelle.

5.1 Schnittstellen im Monitoring-Schichtenmodell

In der Abbildung 5.1 sind die Schnittstelle anhand gelber Punkte gekennzeichnet. Hier sei angemerkt, dass es sich hier nur um ein Beispiel von möglichen Schnittstellen handelt. Jede Schicht kann auch mehrere Schnittstellen zu Verfügung stellen und über mehrere kommunizieren.

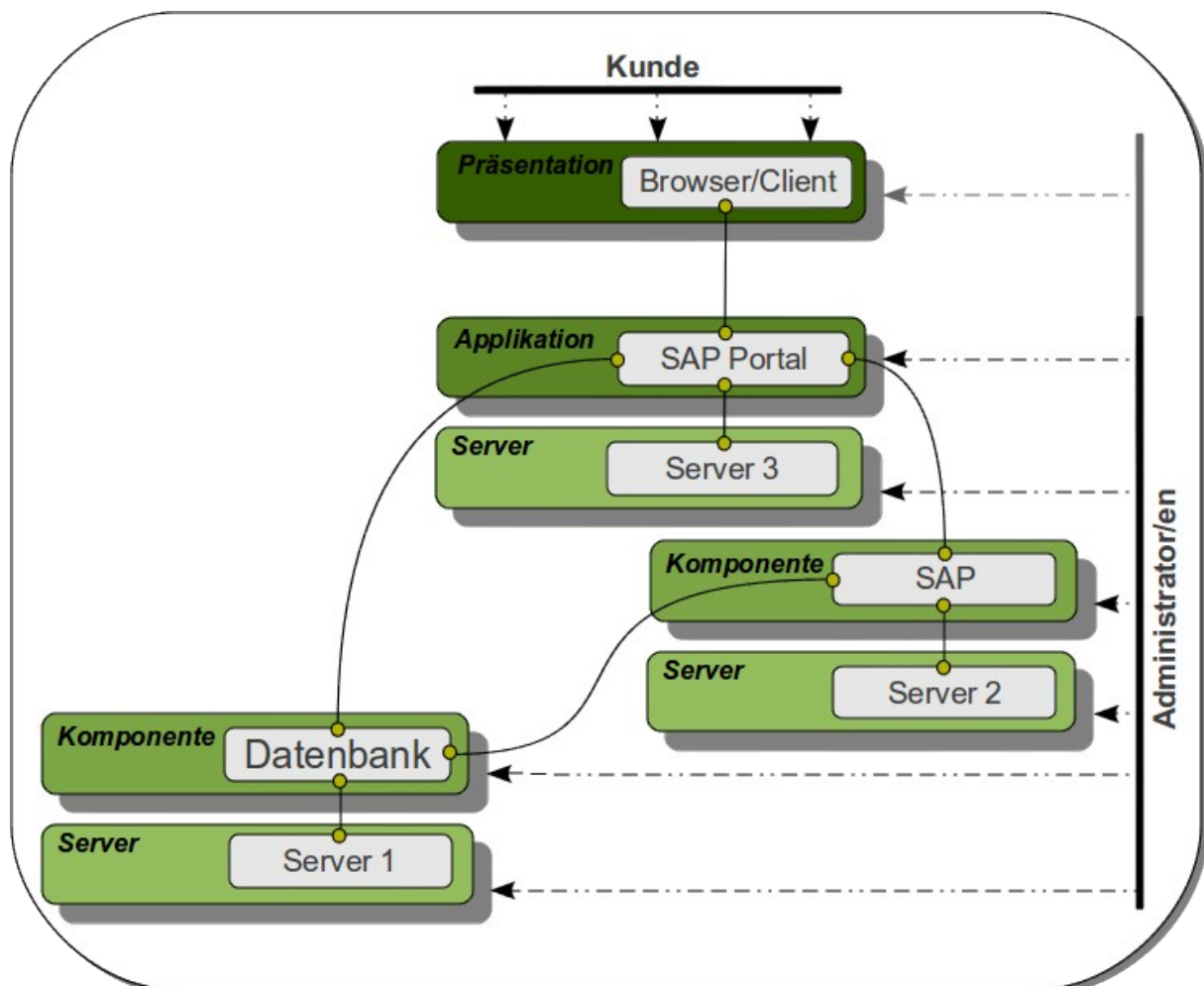


Abbildung 5.1: Schnittstellen im Monitoring-Schichtenmodell

In der Regel kommuniziert eine Schnittstelle über das Netzwerk. Das kann auch beispielsweise über das Internet sein. Natürlich gibt es auch andere Möglichkeiten mit einer Schnittstelle zu kommunizieren. Dieses kann z.B. der Austausch von Dateien sein. Schnittstellen können dabei auf verschiedenste Weise angesprochen werden. Entweder mit einem Client oder auch direkt.

Nicht jede Schnittstelle ist allerdings für ein SLA-Monitoring geeignet. Hier sollte genau hingeschaut werden. Auch das Zusammenfassen von verschiedenen Monitoringergebnissen von Schnittstellen, stellt in der Regel kein zufriedenstellendes Ergebnis dar. Es sollte, wie schon im ersten Teil dieses Beitrages (Punkt 3) erwähnt, darauf geachtet werden, daß das Monitoring einer Schnittstelle alle darunterliegenden Schnittstellen der Schichten mit einbezieht. Allerdings kann das Überwachen von einzelnen Schnittstellen beispielsweise für ein Business-Process Monitoring bzw. Geschäftsprozess-Monitoring wichtig sein oder auch unterstützen.

6 Messen von SLA's

Hier ein weiteres Beispiel für ein Schnittstellen-Monitoring bzw. SLA-Monitoring.

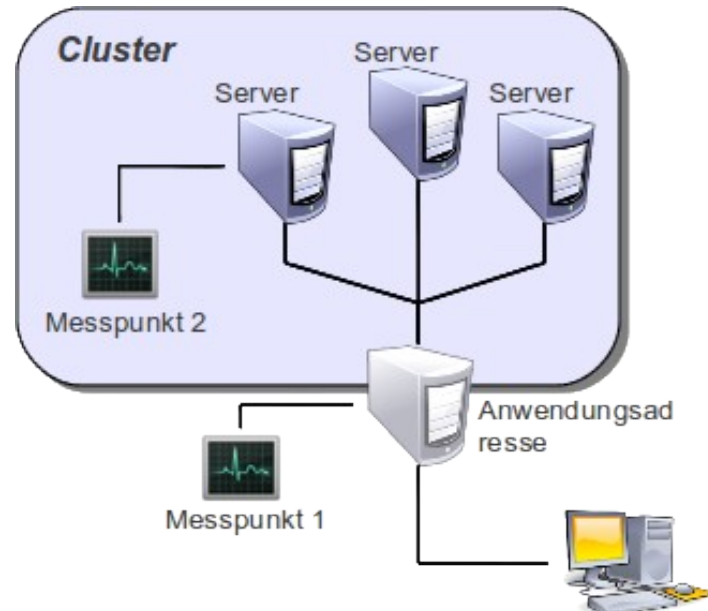


Abbildung 6.1: Messen von SLA's an der richtigen Schnittstelle

Abbildung 6.1 zeigt einen Cluster. Geht man nun davon aus, dass auf jedem Server ein Webserver als Komponente läuft und in diesen ein CMS als Applikation und diese Applikation ist über einen Loadbalancer erreichbar. So würde es wenig Sinn machen jede Schnittstelle (Messpunkt 2) der Webserver zu überwachen und diese als ein Ergebnis zusammenzufassen, da hier der Loadbalancer z.B. nicht mit einbezogen werden würde. Natürlich könnte noch eine weitere Überwachung des Loadbalancers hinzugefügt werden. Macht aber wenig Sinn. Viel wichtiger ist es hier das Zusammenspiel aller Schichten mit dessen Schnittstellen zu überwachen. In diesem Fall wäre es richtig einen Seitenaufruf von dem Messpunkt 1 zu machen und somit die Verfügbarkeit der Applikation CMS zu überprüfen. Fällt ein Webserver aus bedeutet dies nicht die Beeinträchtigung der Verfügbarkeit der Applikation CMS.

7 Messintervalle

Eine wichtige Rolle spielt auch der Messintervall. Eine Messung soll regelmäßig und in gleich bleibenden Zeitabständen durchgeführt werden. Wie oft und in welchen Zeitabständen eine Messung durchgeführt werden muss, ist von Fall zu Fall verschieden. So kann z.B. eine Messung der Festplattenkapazität im Intervall von 5 Minuten wichtig sein, weil viele Daten geschrieben bzw. abgelegt werden. Bei einer Festplatte hingegen, auf der sehr wenig Daten abgelegt werden, ist eine Messung im 5 Minuten Intervall nur wenig sinnvoll. Hier könnte je nach Daten auf kommen z.B. einmal täglich gemessen werden.

Zu bedenken ist, je häufiger gemessen wird, desto höher ist das Datenaufkommen und die durch die Messung verursachten System- und Netzwerklast. Dieses kostet System- und Netzwerkperformance, Plattenplatz und erzeugt letztendlich Kosten in monetärer Form.

Bei einer Verfügbarkeitsmessung hingegen muss häufiger gemessen werden. In der Regel ist ein Intervall von 5 Minuten ein guter Richtwert. Eine häufige Messung ist deshalb notwendig, um bei einem Ausfall oder einer Störung, so genau wie möglich die Dauer bestimmen zu können. Hier können unter Umständen schon einzelne Minuten über die Zahlung einer im SLA vereinbarten Strafe entscheiden.

8 Abgrenzung SLA-Überwachung

Auch wenn zahlreiche Parameter gemessen werden können, muss für SLA relevante Messungen das Augenmerk immer auf die Verfügbarkeit eines IT-Services gelegt werden. Ein IT-Dienstleister wird dennoch in der Regel mehr Parameter messen als nur die Verfügbarkeit, denn zusätzliches proaktives Monitoring liegt in seinem eigenen Interesse. Außer der Messung der Verfügbarkeit von Komponenten werden standardmäßig noch Messungen von Parametern wie Festplattenkapazität, Speicher- und CPU-Auslastung etc., hinzukommen. Diese Messwerte dienen dann eher der Trendanalyse oder als proaktives Monitoring, um Störungen schon im Vorfeld zu erkennen und ab zu wenden.

9 Fazit

Wie mit Hilfe des Monitoring-Schichtenmodell zu sehen ist, lässt sich übersichtlich Abbilden, wie eine Applikation mit seinen Komponenten, Server und eventuell Client aufgeteilt und überwacht werden kann. Genau wie in der Regel ein Konzept zur Installation, Einrichtung und Einführung einer neuen Applikation erstellt wird, sollte dieses auch mit dem Monitoring für eine Applikation geschehen. Das Monitoring-Schichtenmodell bietet dabei eine Unterstützung zur Erstellung eines Monitoring-Konzeptes unter Berücksichtigung aller Interessensträger. Zum anderen wird wirklich auch nur das überwacht, was relevant ist. Dieses führt dazu, dass unnötige Daten bzw. Informationen erst gar nicht gesammelt und somit Kosten und Ressourcen eines Monitoring-Systems geschont werden. Auch werden durch die nicht Überwachung nicht benötigter Dinge, die Fehlalarme reduziert.

Auch kann es oft sehr schwierig sein, eine Applikation auf ihr Verfügbarkeit hin zu überwachen. Schon bei der definition der einzelnen Schnittstellen wird es meist schwierig und unübersichtlich. Denn oft hat jeder ein anderes Verständnis für eine Schnittstelle und deren Messpunkte. Problematischer wird es dann noch, wenn die Applikation stetig wächst und nicht dokumentiert wird. Dieses macht eine Erweiterung des Monitoring nur schwierig.

9.1 Abschliessendes

Anregungen, Kritik, Verbesserungsvorschläge können jederzeit als Kommentar in meinem Blog³ hinterlassen werden oder in meinen Portal⁴. Auch wenn ich diesen Beitrag schon 100 mal Korrektur gelesen, sieht mal als letztes die Fehler einfach nicht mehr. Also falls noch Fehler entdeckt werden, bitte bscheid geben.

9.2 Dank und Änderungen

Datum	Änderungen	Dank
27.01.2011	Die Präsentationsschicht in Kapitel 2 als eine weitere Schicht hinzugügt.	Dank hier an <i>Lefty</i> aus dem Monitoring-Portal für diese Anregung
23.02.2011	Erweiterungen des Serverschicht in Kapitel 2 um die Umwelt des Servers	Dank hier an <i>Daniel Mahrenholz</i> für diese Anregung.

3 <http://luebben-home.de/Blog/Monitoring/LayerModel>

4 <http://monitoring-portal.de>