

**Unternehmen**

AXA/DBV-Winterthur

**Industrie**

Versicherungen und  
Finanzdienstleistungen

**Gebiet**

Global – Deutschland

**Herausforderung**

Suche nach einem Werkzeug für Lasttests und Diagnosen, das sich nahtlos in die dezentralen Java und JEE Umgebungen integrieren läßt:

- Fehlersuche und Stabilitätsprüfung bei der Entwicklung neuer Anwendungen
- Qualitätstests im laufenden Betrieb der Applikationen
- Nachverfolgen von Call-Chains über mehrere synchrone und asynchrone Java-Tiers hinweg

**Lösung**

**dynaTrace Diagnostics** zur Analyse einzelner Benutzertransaktionen in verteilten Java and J2EE Umgebungen bis hinunter auf Code-Ebene

**Ergebnisse**

- Verbesserung der kompletten Prozesskette (Betrieb – QA – Entwicklung) bei der Diagnose von Problemsituation
- Signifikante Beschleunigung der Ursachenanalyse bei Problemen in einzelnen Benutzertransaktionen
- Verwandlung von klassische Blackbox- Lasttest in technisch orientierte Whitebox-Test



**DBV-winterthur**

„Verantwortung aus Tradition“ – der Leitsatz des Wiesbadener Versicherers DBV Winterthur zieht sich wie ein roter Faden durch die

Unternehmensgeschichte. 1872 als „Deutsche Beamtenversicherung“ für die Berufsgruppe der Soldaten ins Leben gerufen, zählt die Assekuranz mit rund 4.600 Mitarbeitern, knapp 3,5 Millionen Mitgliedern und einem Beitragsvolumen von etwa 3,8 Milliarden Euro in 2006 spätestens seit der Fusion mit dem französischen AXA-Konzern zu Deutschlands größten Versicherern. Zu der Kundengruppe der DBV-Winterthur zählen sowohl Privatkunden als auch Beamte und Mitarbeiter im öffentlichen Dienst, kleine und mittlere Unternehmen und Ärzte. Versicherungsprodukte werden in den Sparten Lebens-, Kranken- und Schadenversicherung an, darüber hinaus auch Finanzierungen sowie Geldanlagen angeboten. <http://www.dbv-winterthur.de>

**Herausforderung:** Der hessische Versicherer DBV Winterthur scheut das Risiko: Ähnlich wie im urtypischen Geschäftsumfeld mit Lebens-, Kranken- oder Schadenversicherungen gehen die Wiesbadener auch in der IT keine unnötigen Wagnisse ein. Für sichere und zuverlässige Softwareprozesse und effiziente Applikationsentwicklungen in Java-Umgebungen sorgt ein modernes Diagnose- und Lasttestwerkzeug.

Verantwortung jenseits von Finanzierungen und Policen übernahmen vor rund einem Jahr indes auch die IT-Entscheider der hessischen Gesellschaft, als es im Rahmen des Projekts QSJ2EE um ein neues Engagement zugunsten der Qualitätssicherung für die interne Anwendungsentwicklung ging: „Wir suchten nach einem Werkzeug für Lasttests und Diagnosen, das sich nahtlos in unser Umfeld mit dezentralen Java- und J2EE-Umgebungen integrieren ließ“, erinnert sich Justus Wagner, verantwortlicher Projektleiter bei der DBV Winterthur. Wichtig dabei: Das neue Tool sollte sich sowohl für die Fehlersuche und Stabilitätsprüfung bei der Entwicklung neuer Anwendungen, als auch für Qualitätstests im laufenden Betrieb der Applikationen nutzen lassen. Darüber hinaus galt es, Call-Chains über mehrere synchrone und asynchrone Java-Tiers hinweg nachverfolgen zu können.

**Lösung:** Drei Lösungen standen schließlich auf dem Prüfstand der IT-Experten aus Wiesbaden: dynaTrace Diagnostics des Linzer Herstellers dynaTrace software GmbH, Introscope von Wily (jetzt CA Wily) sowie Performasure von Quest. Wagners Votum: „Wir haben uns schließlich für dynaTrace Diagnostics entschieden, weil es als einzige Lösung die Möglichkeit bot, Benutzeraktivitäten (unter anderem die Call-Chain) eines einzelnen Benutzers über mehrere Java Virtual Machines hinweg nachverfolgen und analysieren zu können“.

*„Wir suchten nach einem Werkzeug für Lasttests und Diagnosen, das sich nahtlos in unser Umfeld mit dezentralen Java- und J2EE-Umgebungen integrieren ließ“*

*Wir entschieden uns schließlich für dynaTrace Diagnostics, weil es als einzige Lösung die Möglichkeit bot, Benutzeraktivitäten (unter anderem die Call-Chain) eines einzelnen Benutzers über mehrere Java Virtual Machines hinweg nachverfolgen und analysieren zu können.“*

*„Mit dynaTrace lässt sich nicht nur die Qualität der Software an sich, sondern auch die der kompletten Prozesskette verbessern.“*

*„Die KnowledgeSensors erfassen neben den Antwortzeiten auch systemnahe Informationen über performance-kritische Mechanismen wie Objektserialisation, Datenvolumen oder Speicherverbrauch.“*

*„Wenn wir heute bei einem Lasttest merken, dass die CPU zu 80 Prozent ausgelastet ist, wissen wir kurze Zeit später auch, welche Komponenten der Anwendung daran schuld sind. Wir müssen uns nicht mehr mit Hypothesen herumschlagen“*

**Ergebnisse: Root-Cause-Analyse.** Gelöst werden konnte diese Herausforderung mit Hilfe der PurePath-Diagnose des Werkzeugs: Die Technologie schlüsselt das Verhalten von Transaktionen auch über mehrere untereinander kommunizierende logische und physische Server hinweg auf. Bei der Root-Cause-Analyse lassen sich damit auch Zusammenhänge zwischen Tiers auf Basis von Requests eines einzelnen Users und der damit verbundenen Daten feststellen.

Damit lässt sich in vielen Fällen erst die eigentliche Ursache eines unerwarteten Verhaltens, wie etwa schlechte Antwortzeiten, ermitteln. In diesem Kontext ist es ein essenzieller Mehrwert, die durch einen Benutzer ausgelösten Aktivitäten in einer Applikation über mehrere physikalisch getrennte JVMs nachverfolgen und dabei die Auswirkung gegebenenfalls auf einen konkreten User zurückführen zu können.

**Die Qualität der Prozesskette verbessern.** Michele Arnese, Performance-Engineer beim von der DBV Winterthur mit der Einführung beauftragten Implementierungspartner C1 SetCon GmbH aus München, nennt die technischen Details: „Mit dynaTrace lässt sich nicht nur die Qualität der Software an sich, sondern auch die der kompletten Prozesskette verbessern“. Anders als bei alternativen Lösungen, mit denen „die zu analysierenden Daten erst manuell exportiert werden müssen“ sei das IT-Team des Versicherers mit dynaTrace in der Lage „Aufnahmen“ von Situationen aus dem laufenden Betrieb zu erstellen, die anschließend in der Entwicklungsabteilung unter die Lupe genommen werden können.

Sogenannte Knowledge-Sensoren erfassen dabei neben typischen Performance-Daten wie Antwortzeiten oder CPU-Usage auch Kontextinformationen – beispielsweise Methodenargumente und Exceptions - um eine punktgenaue Ursachenanalyse zu gewährleisten. Die gesammelten Messwerte werden anschließend bis hinunter zur Code-Ebene mit Methoden, Klassen und SQL-Statements korreliert: „Die KnowledgeSensors erfassen neben den Antwortzeiten auch systemnahe Informationen über performance-kritische Mechanismen wie Objektserialisation, Datenvolumen oder Speicherverbrauch“, so Arnese.

**Whitebox-Test statt Blackbox-Test.** DBV-Winterthur-Projektleiter Wagner bringt die Vorteile in einen praxisbezogenen Kontext: Anstatt Probleme anhand „eines Symptoms künstlich rekonstruieren zu müssen und zu hoffen, dass der Fehler erneut auftritt“, lassen sich Schwierigkeiten anhand „live aufgezeichneter“ Informationen identifizieren. Damit verwandelt sich der klassische Lasttest von einem Blackbox-Test in einen – technisch orientierten – Whitebox-Test.

Konkret sei das Team um Wagner nun in der Lage, eine Last auf mehreren Ebenen und Schnittstellen zu messen. Anders als die Mehrzahl der heute erhältlichen Werkzeuge ließen sich damit nicht nur typische IT-Komponenten wie Betriebssysteme oder Datenbanken monitoren – auch einzelne Bereiche von Anwendungen und Schnittstellen könnten so mit Messpunkten versehen und überprüft werden. So erhalten die Techniker der DBV Winterthur beispielsweise die Option, einen selektiven Dump einzelner JVMs zu erstellen.

Der Vorteil: Anstatt Daten mit mehreren Gigabyte durchforsten zu müssen, ist es jetzt möglich, einen Ausschnitt des JVM-Speichers zu erstellen und zu analysieren: „Wenn wir heute bei einem Lasttest merken, dass die CPU zu 80 Prozent ausgelastet ist, wissen wir kurze Zeit später auch, welche Komponenten der Anwendung daran schuld sind. Wir müssen uns nicht mehr mit Hypothesen herumschlagen“, erklärt Arnese.

Arbeiterleichterungen, die sich nach den Worten Wagners vor allem dann auswirken, wenn mehrere Systeme diagnostiziert werden müssen. Wagner: „Damit reduziert man die Analyse, Fehlersuche und Optimierung auf ein Anwendersystem beziehungsweise eine Komponente eines unter Umständen sehr komplexen Systems.“

Bitte kontaktieren Sie [takeaction@dynatrace.com](mailto:takeaction@dynatrace.com) oder besuchen Sie [www.dynatrace.com](http://www.dynatrace.com) für weitere Informationen.