

Entwickler sich um solche technischen Details nicht kümmern muss. Der Zugriff auf EJBs kann von außerhalb des Containers und auch von außerhalb des Anwendungsservers vonstatten gehen, in dem passende Technologien, wie z.B. RMI, eingesetzt werden. Der Anwendungsserver beinhaltet nicht nur einen EJB-Container, sondern enthält weitere Container (z.B. Servlet-Container und Webserver-Container) und bietet verschiedene weitere Dienste an. Der Anwendungsserver stellt die Laufzeitumgebung für eine große Vielfalt von Softwareartefakten, die verwaltet werden müssen. Dabei fasst Java EE eine große Menge von APIs und Spezifikationen zusammen, die aber bezüglich der Implementierung beliebig kombiniert werden können. So nutzt WebSphere 7.0 als JPA-Persistence Provider beispielsweise standardmäßig Apache OpenJPA. Die Implementierung kann aber auch ausgetauscht werden, so wären auch Hibernate oder TopLink denkbar. Diesbezüglich gibt es von Seiten des Java-EE-Anwendungsservers keine Vorgaben. Die Anwendungsserver bieten außerdem eine große Vielfalt an unterstützten Kommunikationsprotokollen, wie HTTP(S), RMI, SOAP, etc. Der hier implementierte Webservice nutzt beispielsweise SOAP via HTTP.

Dementsprechend bietet die EJB-Technologie eine einfache Art und Weise für die Implementierung von service-orientierten, verteilten Business-Systemen, wie es auch bei dem hier entwickelten System der Fall ist.

2.4. Weiterführendes

Diese Arbeit gliedert sich in die momentane Forschung im Bereich der Metriken im *Lehr- und Forschungsgebiet Informatik 3 Softwarekonstruktion* ein. Deshalb folgt eine Beschreibung des sich in der Entwicklung befindenden Metrikwerkzeugs MeDIC [29]. Dies ist insbesondere deshalb wichtig, da in Zukunft eine Integration des in dieser Arbeit vorgestellten Systems in MeDIC vorgesehen ist. Es folgt ein kurzer Abschnitt zu *SOA (Service Oriented Architecture)*, da das System auch service-orientiert ist.

2.4.1. MeDIC

Um eine große Menge von Metriken zu definieren, zu pflegen, zu dokumentieren und weiterzuentwickeln, reicht es nicht aus, Metriken rein textuell zu beschreiben, da dies einige Nachteile mit sich bringt.

Erstens sind rein textuelle Beschreibungen der Messvorschriften selten präzise genug, um Fehler bei der Verwendung der Metriken zu vermeiden. Falsch durchgeführte Messungen aufgrund falsch verstandener Messvorschriften führen zu unbrauchbaren Ergebnissen. Des Weiteren können rein textuelle Beschreibungen leicht dazu führen, dass unterschiedliche Personen die Beschreibungen unterschiedlich interpretieren, so dass auch unterschiedliche Ergebnisse erzielt

werden, obwohl eigentlich die selbe Metrik angewandt wird.

Zweitens sind die Beschreibungen hinsichtlich der Einsatzgebiete häufig nicht präzise genug, was zur Verwendung von Metriken im falschen Kontext führen kann. Dies führt wieder zu möglicherweise falschen Messergebnissen.

Drittens ist die Wahrscheinlichkeit, dass die Beschreibungen inkonsistent sind, bei einer textuellen Beschreibung deutlich höher. Inkonsistent heißt in diesem Zusammenhang, dass innerhalb einer Metrikbeschreibung oder einer Menge von Metrikbeschreibungen widersprüchliche Definitionen vorhanden sind. Dementsprechend sind die Ergebnisse auch hier unbrauchbar.

Viertens fehlt in einer rein textuellen Beschreibung die Navigierbarkeit in den Dokumenten. In MeDIC kann man leicht über Links die verschiedenen Aspekte ansteuern. In einer Menge von Textdokumenten ist das leider nicht möglich.

Außerdem werden Metriken häufig nicht zentral verwaltet, so dass unter Umständen die gleiche Metrik mehrmals innerhalb einer Organisation definiert wird. Möglicherweise sind die Definitionen nicht identisch, so dass selbst Arbeitsplatzwechsel innerhalb einer Organisation zu Verwirrungen führen, da z.B. gleich lautende Metriken unterschiedlich benutzt werden. Eine zentrale Verwaltung der Metriken einer Organisation vermeidet den Aufwand, den neue Metrikentwicklungen verursachen, da vor der Entwicklung geprüft werden kann, ob bereits eine passende Metrik existiert. Diese Kommunikation der Metriken innerhalb der gesamten Organisation ist entscheidend um sich erheblichen Mehraufwand zu sparen.

Um diese Probleme zu beseitigen wurde vom Lehr- und Forschungsgebiet Informatik 3 Softwarekonstruktion die MeDIC-Werkzeugunterstützung entwickelt. Sie bietet eine Verwaltung für Metriken, einen Editor für selbige, einen Dokumentationsgenerator und ein prototypisches Berechnungswerkzeug. Der Editor für Metriken ermöglicht die Erstellung von Metriken auf Basis des zugrunde liegenden Metamodells. Eine Instanz des Metamodells wird im Folgenden als Metrikportfolio bezeichnet.

Das Metamodell besteht u.a. aus Metriken, Messvorschriften, Skalen, vermessene Entitäten, Informationsbedürfnissen, Interpretationen der Ergebnisse, Ergebnisaufbereitungen und Auswertungen. Dem ISO-Standard 15939 [16] folgend werden bei den Metriken Basismetriken und abgeleitete Metriken unterschieden. Die Messvorschriften werden im Metamodell separat gehandhabt, wobei auch hier zwei Vorschriften unterschieden werden. Auf der einen Seite gibt es die nach ISO-Standard definierte Messmethode für Basismetriken und auf der anderen Seite gibt es die sogenannte Messfunktion, die die Messvorschrift für abgeleitete Metriken darstellt. Während die Messmethode natürlich-sprachlich oder in einer formalen Sprache definiert wird, stellt die Messfunktion meistens eine mathematische Berechnungsvorschrift für die Verknüpfung der zugrunde liegenden Basismetriken oder anderer abgeleiteter Metriken dar. Zudem sieht

das Metamodell die Definition von Skalen für die Metriken vor, die, wie bereits beschrieben, wichtig für die Auswertung von Messergebnissen sind. Zusätzlich enthält das Metamodell, auf dem MeDIC basiert, sogenannte vermessene Entitäten, mit deren Hilfe eine Verknüpfung zwischen Domänenmodell und Metrikportfolio geschaffen wird. Dieses Verknüpfung sorgt dafür, dass Metriken auch nur für Entitäten benutzt werden, für die sie auch vorgesehen sind.

Wie bereits angesprochen ist die Kommunikation der Ergebnisse und vorhandenen Metriken innerhalb der Organisation ein entscheidender Faktor für den Nutzen, der durch die Verwendung eines Werkzeugs entsteht. Denn dadurch können unter anderem Kosten für erneute Metrikentwicklungen gespart werden. Dies unterstützt MeDIC durch den Dokumentationsgenerator, der HTML-Dokumente für unterschiedliche Adressatengruppen generiert, die in der Organisation zentral auf einem Server gespeichert werden können, um überall in der Organisation verfügbar zu sein. Der baumbasierte Metrik-Editor, der mittels EMF realisiert wurde, ermöglicht es ein Metrikportfolio zu erstellen. Außerdem kann man mit MeDIC die Metriken validieren, d.h. einzelne Metrikdefinitionen werden auf Vollständigkeit und alle Metriken eines Portfolios auf Konsistenz untereinander geprüft. Dies ist ein besonderer Vorteil von MeDIC gegenüber der rein textuell basierten Metrikbeschreibung, wo eine ähnliche werkzeuggestützte Prüfung nicht möglich ist.

Das MeDIC sinnvoll ist, ergibt sich, wenn man sich die CMMI-Anforderungen für „Measurement and Analysis“ ansieht. MeDIC unterstützt bei der Erfüllung des spezifischen CMMI-Ziels *SG1: Mess- und Analysetätigkeiten in Übereinstimmung bringen* mit allen seinen spezifischen Praktiken [29]. Außerdem werden die spezifischen Praktiken *SP 2.2* und *SP 2.4* unterstützt.

2.4.2. SOA

Dieser Abschnitt bietet einen kleinen Einblick in *SOA*. Um eine möglichst hohe Flexibilität zu erreichen handelt es sich bei diesem Projekt um eine *Service-Oriented Architecture*. Laut Starke u. Tilkov [26] ist eine *SOA*

„eine Unternehmensarchitektur, deren zentrales Konstruktionsprinzip Services (Dienste) sind. Dienste sind klar gegeneinander abgegrenzte und aus betriebswirtschaftlicher Sicht sinnvolle Funktionen. Sie werden entweder von einer Unternehmenseinheit oder externe Partner erbracht.“

oder anders nach Starke u. Tilkov [26] ausgedrückt ist eine *SOA*

„eine unternehmensweite IT-Architektur, deren zentrales Konstruktionsprinzip lose gekoppelte Services (Dienste) sind. Services realisieren Geschäftsfunktionen, die sie über eine implementierungsunabhängige Schnittstelle kapseln. Zu jeder Schnittstelle gibt es einen

Servicevertrag, der die funktionalen und nicht-funktionalen Merkmale (Metadaten) der Schnittstelle beschreibt. Die Nutzung (und Wiederverwendung) von Services geschieht über (entfernte) Aufrufe (»Remote Invocation «).

Wie man anhand der Definition sehen kann, stehen die Dienstleistungen im Zentrum. Bei ihnen muss es sich nicht unbedingt um Dienste handeln, die durch Informationstechnologie realisiert werden. Dennoch liegt das Hauptaugenmerk der Informatik hauptsächlich auf diesen. Im Sinne der Definition müssen die Dienste auch einen betriebswirtschaftlichen Nutzen erbringen, was im vorliegenden System auch der Fall ist, da durch das System Zeit beim Metrikvorschlagsprozess gespart wird. Außerdem hilft es die Erstellung von Metriken voranzutreiben, die wiederum einen klaren betriebswirtschaftlichen Nutzen haben. Somit dient SOA der Standardisierung, Flexibilisierung und Wiederverwendung.