

Robot to Business: Informationstechnisch gestützte Dienstleistungen in mobilen und verteilten Umgebungen – ein ganzheitlicher Ansatz

Emanuel Georgiew, Johannes Pläßmann, Jens Wittrowski, Thilo Steckel

Siemens AG
Siemens IT Solutions and Services C-LAB
Fürstenallee 11
33102 Paderborn
{emanuel.georgiew, johannes.plassmann, jens.wittrowski}@siemens.com

Claas Selbstfahrende Erntemaschinen GmbH
Entwicklung Systeme und Dienstleistungen
Münsterstraße 33
33428 Harsewinkel
thilo.steckel@claas.com

Abstract: Die Anreicherung von Produkten um Dienstleistungen hin zu einer hybriden Wertschöpfung erfordert innovative informationstechnische Lösungen. Informationstechnisch gestützte Dienstleistungen in einem hoch dynamischen Umfeld bringen zusätzlich besondere Herausforderungen mit sich. Die Randbedingungen sind einem ständigem Wechsel unterzogen, die Teilnehmer mobil und nicht permanent in Kommunikationsnetze eingebunden. Anhand des mit Mitteln des Bundesministeriums für Wirtschaft und Technologie geförderten Projektes *Robot to Business* wird ein ganzheitlicher Lösungsansatz entworfen, der generisch anwendbar ist, um Dienstleistungen im mobilen und verteilten Umfeld zu definieren, einschließlich der Unterstützung für die automatisierte Ausführung.

1 Einleitung

Hersteller von Produkten mit langen Lebenszyklen sehen sich zunehmend mit der Herausforderung konfrontiert, mittels innovativer Dienstleistungen einen langen Lebenszyklus wertschöpfend abzudecken. Die Hersteller binden informationstechnische Lösungen an ein Produkt, um zusätzliche Dienstleistungen anzubieten. Ein so realisierter Zusatznutzen bleibt damit über die Lebenszeit des Produktes statisch fest gebunden. Diese enge Kopplung wird gelöst, indem das Geschäftsmodell von einem reinen Produktgeschäft hin zu einer hybriden Wertschöpfung – eine flexible und dynamische Anreicherung eines Produkts durch neue innovative Dienstleistungen - gewandelt wird.

Moderne Technologien aus der Informations- und Kommunikationstechnik, wie die Weiterentwicklungen im Bereich des Mobilfunks und eingebetteter Systeme, sowie neue Paradigmen, wie serviceorientierte Architekturen (SoA), ermöglichen die Entwicklung von flexiblen, lose gekoppelten und anpassungsfähigen Softwarediensten, welche für hybride Geschäftsmodelle geeignet sind. In diesem Fall ist also die Bereitstellung von Spezial-Software eine zusätzliche Dienstleistung, mit der ein Produkt angereichert werden kann. Gleichzeitig kann die Softwarelösung aber auch dazu dienen bereits bestehende Dienstleistungen zu optimieren oder diese mit weiteren Teilleistungen anzureichern. Besondere Anforderungen an eine informationstechnische Unterstützung ergeben sich für Dienstleistungen, die in einem hoch dynamischen Umfeld von mobilen Teilnehmern erbracht werden.

Ein Beispiel hierfür ist die Landwirtschaft. Der gesamte Ernteprozess – sofern dieser von einem Lohnunternehmer erbracht wird – stellt eine Dienstleistung dar, die sich aus mehreren Teilleistungen zusammensetzt. Einzelne Teilleistungen werden hierbei von mehreren mobilen Landmaschinen erbracht, etwa mähen, wenden oder schwaden. Die Planung der Leistungserbringung hingegen wird von zentraler Stelle durch einen Lohnunternehmer durchgeführt. Um den gesamten Ernteprozess von der Planung bis zur Abrechnung informationstechnisch zu unterstützen, müssen verschiedene Softwarekomponenten miteinander interagieren. Da eine exakte Planung (Disposition) in diesem dynamischen Umfeld nicht zielführend ist, wird eine allgemeine Beschreibung vorgezogen. Dies wird als Konfiguration bezeichnet. Somit kann die Zusammenstellung von verschiedenen Teilleistungen zu einer Gesamtleistung vorgenommen werden, ohne eine konkrete Disposition. Zum Zeitpunkt der Leistungserbringung wird der Arbeitsprozess von einer Workflow-Engine auf der Maschine gesteuert und überwacht. Informationen (z.B. Messwerte von Sensoren, Zeitmessungen, etc.) der Arbeitserledigung werden automatisiert festgehalten. Eine weiterführende Verarbeitung (z.B. Rechnungsstellung) ist nach Erledigung der Leistungserbringung ebenfalls automatisiert durchführbar. Diese informationstechnische Lösung führt zu einer Steigerung von Effizienz, Qualität und Durchsatz, sowie zu einer Reduktion von Kosten. Des Weiteren wird die Erbringung zusätzlicher Dienstleistungen durch den Lohnunternehmer ermöglicht, wie beispielsweise die Messung und Auswertung von Daten wie Pflanzenart, Bodenbeschaffenheit, Schädlingsbefall und vieles mehr.

Neben der Landwirtschaft gibt es noch zahlreiche weitere Anwendungsgebiete bei denen die Mobilität und Verteiltheit eine Rolle spielen, z.B. der Baustellenbetrieb oder die Wartung und der Service von IT-Infrastrukturen.

2 Robot to Business

Im vom Bundesministerium für Wirtschaft und Technologie (BMWi)[1] geförderten Projekt *Robot to Business - IT-gestützte Integration von semi-autonomen mobilen Maschinen und Prozessen in Geschäfts- und Service-Modelle (R2B)*[2] werden methodische Ansätze und technische Implementierungen entwickelt und umgesetzt, die von der Definition neuer Dienstleistungen bis zu deren Realisierung speziell für Szenarien von mobilen Geschäftsprozessen reichen.

R2B ist Teil des Förderprogramms „SimoBIT – sichere Anwendung der mobilen Informationstechnik zur Wertschöpfungssteigerung im Mittelstand und Verwaltung“[3]. Projektträger ist das Deutsche Zentrum für Luft- und Raumfahrt e.V (DLR). Der zu entwickelnde Lösungsansatz wird anhand von Anwendungsfällen aus unterschiedlichen Anwendungsdomänen validiert. Es werden ein landwirtschaftliches Anwendungsszenario (Grünfütterernte) und ein Szenario aus dem Bereich IT Service Management (flexible Wartung von IT-Infrastrukturen) betrachtet.

3 Architektur für den ganzheitlichen Ansatz

Die Grundarchitektur zur Umsetzung des ganzheitlichen Ansatzes besteht zunächst aus einer Unterteilung in zwei Ebenen: Management Layer und Member Layer (Abb.1). Im Management Layer sind statische Komponenten (z.B. Backend Systeme, Systeme externer Anbieter, etc.) angesiedelt, die zur Verwaltung und Steuerung des Gesamtsystems dienen. Im Member Layer sind die mobilen Komponenten angeordnet.

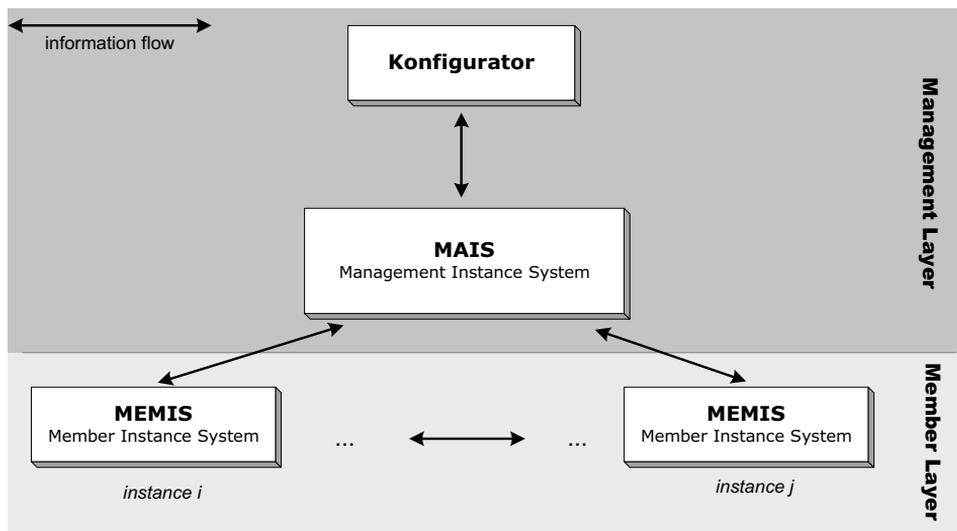


Abbildung 1: Grundkomponenten der R2B Architektur

3.1 Konfigurator

Die Komponente *Konfigurator* in Abb. 1 erfüllt zwei zentrale Aufgaben. Erstens dient er dazu bereits existierende Leistungen zu konfigurieren. Somit kann eine Leistung in unterschiedlichen Ausprägungen existieren, z.B. Abrechnung nach Zeit oder nach Pauschalen. Zweitens ermöglicht er die Schaffung von gänzlich neuen Leistungen. Dies wird dadurch ermöglicht, dass kleinere Bausteine, sog. Teilleistungen, Features und Profile, zusammengesetzt eine Leistung bilden.

Durch das Kombinieren bestehender Bausteine oder das Anlegen komplett neuer Bausteine ermöglicht der Konfigurator die Erstellung neuer Leistungen. Die Konfigurationen werden über wohldefinierte Kommunikationsschnittstellen an das in 3.2 näher dargestellte *Management Instance System* (MAIS) kommuniziert. Das MAIS steuert die konkrete Ausführung der Leistung auf Basis der Konfiguration. Zu beachten ist, dass die Konfiguration immer nur über die Zwischenschicht des MAIS geschieht. Es kann keine Konfiguration direkt auf tiefer liegende mobile Instanzen vorgenommen werden.

3.2 Management Instance System

Im MAIS findet eine Zuordnung der im Konfigurator festgelegten und konfigurierten Leistungen auf entsprechende ablauffähige Implementierungen statt. Diese Implementierungen müssen vorhanden sein und über ein Schema der entsprechenden Leistung entsprechen. Somit ist gewährleistet, dass für die mobilen ausführenden Einheiten die entsprechenden Aufgaben zugeordnet werden können. Das MAIS die oberste Kontroll-Instanz für die tatsächliche Ausführung einer Gesamtleistung. Anfallende Teilleistungen, die von mobilen ausführenden Komponenten zu erfüllen sind, dargestellt jeweils als *Member Instance System* (MEMIS), werden von hier aus angestoßen oder deren Ergebnisse zusammengeführt. Eine Kommunikation zwischen MAIS und MEMIS muss jedoch nicht permanent gegeben sein. Gerade ein asynchrones Muster kennzeichnet Flexibilität und gewisse Autonomie der mobilen Komponenten. Der Abschluss einer Leistung findet immer im MAIS statt, wie z.B. die Rechnungserstellung für eine Leistung.

3.3 Member Instance System

Eine MEMIS steht für eine in der Leistungserbringung involvierten mobilen Komponente. Das ist im landwirtschaftlichen Szenario z.B. eine Landmaschine. Eine MEMIS erbringt eine oder mehrere Teilleistungen für die vom Konfigurator festgelegte und dem von der MAIS gesteuerte Gesamtleistung. Dies wird durch die Kommunikationsverbindungen zwischen MEMIS und MAIS in Abbildung 1 dargestellt. Zu beachten ist der in Abbildung 1 dargestellte Informationsaustausch zwischen einzelnen MEMIS Instanzen. Die mobilen Komponenten können sich untereinander koordinieren und so auch kollaborativ – ohne Mitwirken des MAIS - über nahe und ferne Distanzen die Leistungserbringung realisieren.

4 Umsetzung

Die Implementierung basiert im Wesentlichen auf den Technologien OSGi[4] und Business Process Execution Language (BPEL)[5]. OSGi stellt einen auf JAVA basierenden Applikationscontainer bereit. In einem OSGi Container werden Module (sog. Bundles) installiert, die bestimmte Aufgaben übernehmen.

Das sind zum Beispiel Erfassung und Speicherung von Betriebsdaten wie GPS Positionsdaten, Arbeitsschritte oder Maschinenzustände. Auch die Steuerung der Kommunikation über verschiedene Kanäle, wie das Universal Mobile Telecommunications System (UMTS)[5] oder WLAN[6] wird mit spezialisierten OSGi Modulen realisiert. Der OSGi-Container erlaubt die Aktivierung, Deaktivierung und Installation von Modulen. Da dies auch zur Laufzeit möglich ist, kann der Container sehr flexibel und dynamisch um Funktionalitäten erweitert werden. Die entsprechende Maschine (z.B. Feldhäcksler oder Schlepper) ist in einem Arbeitsablauf eingebettet. Der Maschine werden im Rahmen dieses Ablaufes (Leistungserbringungsprozess) bestimmte Aufgaben zugewiesen, sogenannte Mikroprozesse. Diese Mikroprozesse werden auf der Maschine installiert und mittels einer Workflow-Engine auf Basis von BPEL[AA07] ausgeführt. Der gesamte Leistungserbringungsprozess als auch die darin enthaltenen Teilprozesse werden mit Hilfe eines BPEL-Designers modelliert und in einer BPEL-Engine auf einem Backend-System (für den gesamten Prozess) und der Maschine (für die Teilprozesse) zum Ablauf gebracht. Die angebotenen Leistungen werden in dem Konfigurator hinterlegt. In diesem wird auch definiert, welcher Prozess durch die BPEL-Engine zum Ablauf gebracht wird, wenn eine bestimmte Leistung (oder Geschäftsvorfall) abgearbeitet werden soll.

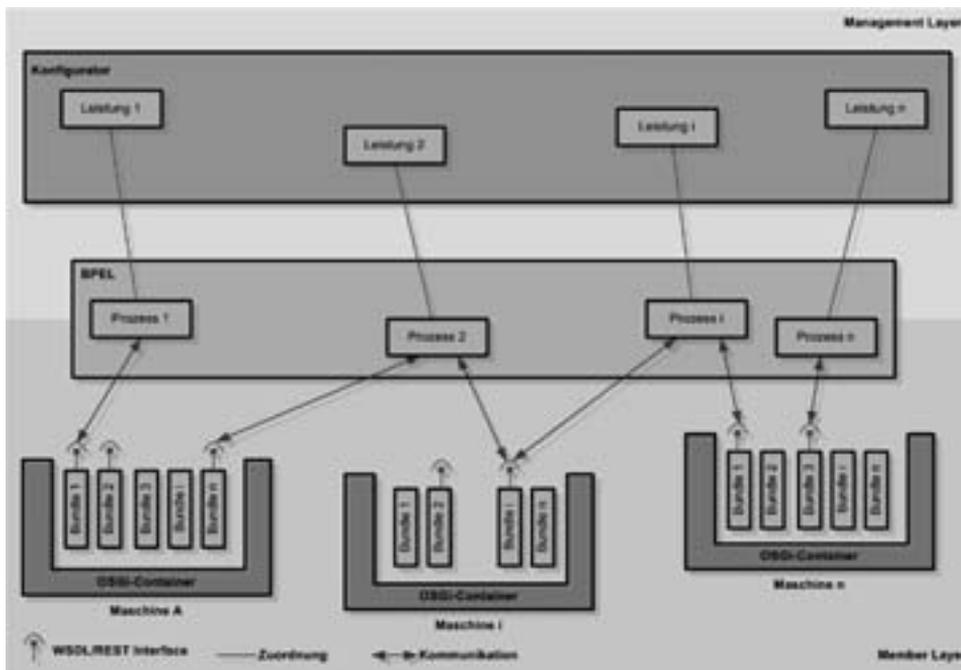


Abbildung 2: Schnappschuss einer aktuell laufenden Leistungserbringung

Eine Momentaufnahme einer laufenden Leistungserbringung zeigt Abbildung 2. Im Konfigurator sind Leistungen ausgewählt, deren Erbringung aktuell angestoßen ist. Der jeweiligen Leistung ist genau ein Prozess zugeordnet, welcher durch eine BPEL-Engine ausgeführt wird. Die BPEL-Engine überlagert den Management Layer und den Member Layer.

Damit ist angedeutet, dass die Prozessausführung auf beiden Ebenen statt findet. Die gestarteten Prozesse kommunizieren je nach Funktionalität bzw. Aufgabe mit einer oder mehreren Maschinen über definierte Schnittstellen, die von Komponenten (Bundles) des OSGi-Containers auf der jeweiligen Maschine zur Verfügung gestellt werden. Über diese Schnittstellen werden die zur Erbringung der Leistungen notwendigen Daten angefragt bzw. ausgetauscht.

Die prototypische Implementierung im Projekt R2B erfolgte mit einem Industrie-PC (Simatic Microbox), der mit einem Windows XP Embedded Betriebssystem ausgestattet ist. Als OSGi-Container kommt die Umsetzung von Eclipse-Equinox [7] zum Einsatz. Die Modellierung und Ablaufsteuerung von BPEL-Prozessen erfolgt mit Hilfe des Werkzeuges ActiveVOS von ActiveEndpoint. Das beschriebene Szenario wurde mit realen Maschinen (Schleppern und Feldhäckslern) unter Einbindung eines Backend-Systems und des Konfigurators erfolgreich zum Ablauf gebracht. Je nach Auswahl der Leistung wurden die beteiligten Maschinen mit entsprechenden Aufträgen versehen. Die für die Erbringung der Leistung notwendigen Maschinen-Daten wurden über einen ControllerAreaNetwork (CAN)-Bus[8] ausgelesen, verschiedenste Betriebszeiten erfasst und eine Fahreridentifikation durchgeführt. Diese Daten wurden dann je nach den Kommunikationsbedingungen vor Ort mittels WLAN oder UMTS zum Backend gesendet. Das Backend-System kumulierte diese Daten und stieß eine automatische Rechnungserstellung für den entsprechenden Geschäftsvorfall an.

Literaturverzeichnis

- [1] Bundesministerium für Wirtschaft und Technologie. <http://www.bmwi.de/>.
- [2] Robot to Business Projektseite. <http://www.r2b-online.de/>.
- [3] Förderprogramm SimoBIT– sichere Anwendung der mobilen Informationstechnik zur Wertschöpfungssteigerung im Mittelstand und Verwaltung. <http://www.simobit.de/>.
- [4] OSGi Alliance. <http://www.osgi.org/>.
- [AA07] Alves, A. et al. (2007), Web services business process execution language version 2.0. OASIS Standard.
- [5] 3rd Generation Partnership Project Specifications. <http://www.3gpp.org/specifications>.
- [6] Wireless Local Area Networks, IEEE 802.11. <http://grouper.ieee.org/groups/802/11/>.
- [7] Equinox OSGi-Framework. <http://www.eclipse.org/equinox>.
- [8] Controller Area Network. <http://www.can-cia.org/index.php?id=46>.