

Modellierung eines Anwendungssystems zur Behälterlokalisierung und Behälterreservierung auf Basis des Architekturstils REST

Thomas Barton, Harriet Bach

Fachbereich Informatik, Fachhochschule Worms

1 Einleitung

Der Zugriff auf betriebliche Anwendungen von beliebigen Standorten aus eröffnet neue und interessante Anwendungen für mobile Geräte. Eine solche Anwendung stellt die Lokalisierung und Reservierung von Behältern im Rahmen eines RFID-gestützten Behältermanagements in der Produktion dar. Bei der Realisierung eines Anwendungssystems sind eine gemeinsame Schnittstelle, die einfache Wiederverwendbarkeit von Funktionen und eine gute Skalierbarkeit wichtige Faktoren, die der Architekturstil REST (Representational State Transfer) als Teil von Web 2.0 aufweist. Dieser Beitrag stellt REST vor, listet die Anforderungen an ein Anwendungssystem zur Behälterlokalisierung und –reservierung mit Hilfe von Anwendungsfällen auf und beschreibt, wie dieses Anwendungssystem modelliert werden kann.

2 Behältermanagement mit RFID

Zur Bezeichnung des gezielten Einsatzes von Mehrwegtransportbehältern und die Verwaltung von deren Beständen bzw. zur Steuerung von Behälterkreisläufen in der Supply Chain hat sich der Begriff Behältermanagement etabliert (Lackner und Zsifkovits 2007). Mehrwegtransportbehälter oder kurz Behälter werden sehr häufig mit RFID-Transpondern versehen (Strüker, Gille und Faupel 2008). Dadurch kann die Datenerfassung berührungslos und ohne Sichtkontakt erfolgen. Verschiedene Ladeprozesse können gleichzeitig erfasst werden, so dass eine Zeitersparnis im Wareneingang und Warenausgang erzielt werden kann (Golabeck 2007). Der Einsatz von RFID-Technologie im Behältermanagement rechnet sich in ROI-Berechnungen in überwiegendermaßen (Fleisch, Christ und Dierkes 2005) und reduziert die Einkaufs- und Mietkosten (Günther, Kletti und Kubach 2008). Ebenso sind Behälter in Echtzeit lokalisierbar, und leere Behälter werden schneller einer erneuten Nutzung zugeführt (Franke und Dangelmaier 2006). In einer erst kürzlich

veröffentlichten Studie wurde der Nutzen von RFID-Technologie anhand von Modellrechnungen bestätigt: So steigt die Rückgaberrate der Behälter an, und neue Behälter können weniger häufig und in kleineren Einheiten bestellt werden (Thoro, Melski und Schumann 2009). Eine Einführung in RFID und einen Überblick über RFID-Anwendungen mit dem Fokus Logistik ist in folgenden Quellen verfügbar: Franke und Dangelmaier 2006; Gillert und Hansen 2007; Günther, Kletti und Kubach 2008; Wohlers und Breitner 2008.

3 Architekturstil REST

Das World Wide Web besitzt eine Architektur, die von Fielding im Jahre 2000 in Form des Architekturstils REST (Representational State Transfer) identifiziert wurde (Fielding 2000). Ein wichtiger Begriff für das Verständnis des Architekturstils REST ist die Ressource. Eine Ressource kann ein elektronisches Dokument sein, ein Bild, eine Informationsquelle, die einem bestimmten Zweck dient, wie z. B. der „Wetterbericht von Los Angeles“ oder auch eine Sammlung anderer Ressourcen (RFC3986). Ist eine Ressource über das Internet zugänglich, besitzt sie eine URI (Uniform Resource Identifier). REST nutzt die Standards des Web wie das Protokoll HTTP, die Identifizierung von Ressourcen mit Hilfe von URIs sowie die eXtensible Markup Language XML. Der Begriff Repräsentation in dem Namen REST steht dafür, dass Anwendungen auf Basis des Architekturstils REST verschiedene Formate zur Verfügung stellen wie z. B. XML, HTML, Text- oder Binärformate. Ist beispielsweise der Aufrufer einer Anwendung ein Browser, wird er ein HTML-Repräsentationsformat anfordern, während eine Desktop-Anwendung ein XML-Repräsentationsformat benötigt. Dabei erfolgt die Kommunikation mit einer Anwendung auf Basis des Architekturstils REST statuslos, d.h. die Anwendung erhält serverseitig einen Kontext nicht aufrecht. Ein Zustand wird entweder vom Client gehalten, oder der Server wandelt ihn in einen Status einer Ressource um. Ein Aufrufer einer Anwendung wird von einem Zustand in einen anderen Zustand überführt, indem der Aufrufer einen Link folgt, der von der Anwendung an den Aufrufer zuvor gesendet wurde (Tilkov 2009). Durch eine statuslose Kommunikation kann die Skalierbarkeit einer Anwendung deutlich verbessert werden, unterstützt eine Anwendung basierend auf dem Architekturstil REST beispielsweise Caching und Load Balancing (Pautasso, Zimmermann und Leymann 2008). Eine wichtige Eigenschaft des Architekturstils REST ist die Verwendung einer gemeinsamen Schnittstelle (Fielding und Taylor 2002). Durch die Existenz einer gemeinsamen Schnittstelle wird die Wiederverwendung vereinfacht: So bieten viele Provider von Web Services wie Amazon, Ebay und Google Schnittstellen auf Basis des Architekturstils REST an (Liu und Deters 2008). Die Standardmethoden der gemeinsamen Schnittstelle sind die acht existierenden HTTP-Methoden, von denen die vier wichtigsten Methoden GET, PUT, POST und DELETE sind (RFC2616). Die Methode GET dient dazu, eine Repräsentation einer Ressource

anzufordern. Mit Hilfe der Methode PUT wird eine bestehende Ressource aktualisiert. Das Anlegen einer neuen Ressource erfolgt mit Hilfe der Methode POST. Die Methode DELETE findet beim Löschen einer Ressource Verwendung (Tilkov 2009).

4 Anforderungen an ein Anwendungssystem zur Behälterlokalisierung und Behälterreservierung

Zur Beschreibung der Anforderungen gehen wir von einem Behältermanagement in der Produktion aus, wobei ein geschlossenes System vorliegt. Dabei dienen die Behälter dazu, während des Produktionsprozesses Halbfertigprodukte und Produkte zu transportieren. Vereinfachend gehen wir von einer Produktionsstätte aus. Die Produktionsprozesse selbst sind verschiedenen Stationen zugeordnet (siehe Abbildung 1).

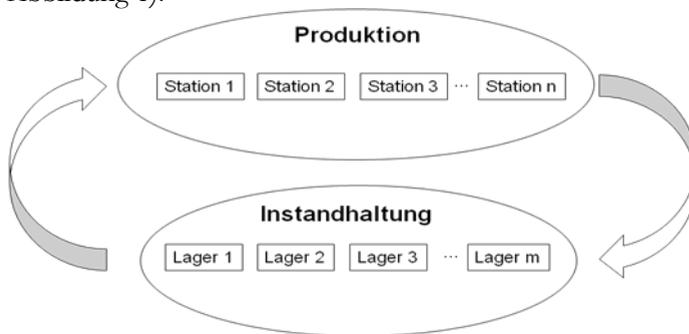


Abbildung 1: Behälter in einem Kreislauf aus Produktion und Instandhaltung

Befindet sich ein Behälter in einem bestimmten Produktionsprozess, soll die zu dem Produktionsprozess assoziierte Produktionsstation lokalisiert werden. Da die Behälter im Rahmen des Instandhaltungsmanagement zu reinigen und zu warten sind, gilt es, Behälter zu verfolgen, die sich in einem Reinigungs- oder Reparaturprozess befinden. Weitere Anforderungen beziehen sich auf das Bestands- und Umlaufmanagement: So gilt es, die Lagerung eines Behälters in den verschiedenen Lagern zu verfolgen und Aussagen über die Verfügbarkeit (in Reinigung, in Wartung, verfügbar, nicht verfügbar) zu treffen. Weiterhin soll ein Benutzer in die Lage versetzt werden, einen Behälter für einen gewissen Zeitraum zu reservieren.

Aus Sicht der Anwender sind die folgenden Anwendungsfälle in dem zu erstellenden Anwendungssystem umzusetzen:

- Informationen zu Behälter abfragen (siehe Abbildung 2):

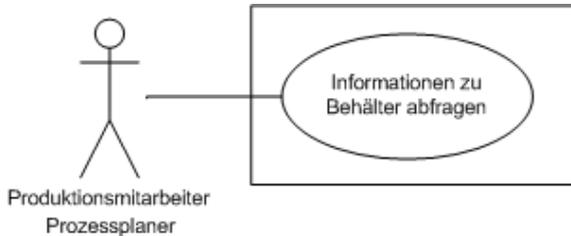


Abbildung 2: Anwendungsfall „Informationen zu Behälter abfragen“

Dieser Anwendungsfall dient dazu, u.a. folgende Informationen verfügbar zu machen:

- Wo befindet sich ein spezieller Behälter?
- Ist ein spezieller Behälter verfügbar oder wird er gerade gereinigt?
- Welche Behälter sind zu einem festen Zeitpunkt für einen gewissen Zeitraum verfügbar?

Mögliche Akteure sind Mitarbeiter in der Produktion bzw. Prozessplaner oder Prozessingenieure.

- Behälter reservieren (siehe Abbildung 3):

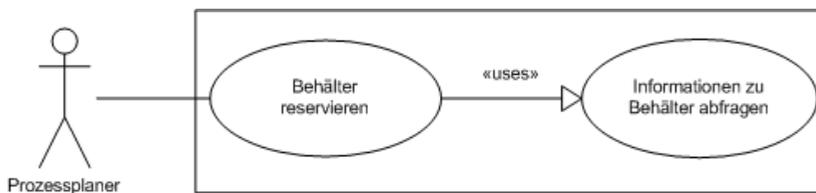


Abbildung 3: Anwendungsfall „Behälter reservieren“

Hier kann ein Behälter – unter Verwendung des Anwendungsfalles „Informationen zu Behälter abfragen“ – zu einem festen Zeitpunkt für einen gewissen Zeitraum reserviert werden. Dieser Anwendungsfall ist für einen Prozessplaner bzw. Prozessingenieur vorgesehen.

Die Interaktion zwischen Benutzer und Anwendungssystem soll über PDAs und WLAN erfolgen. Ein Einsatz von Smartphones und UMTS ist ebenso denkbar.

Weitere Anforderungen an das Anwendungssystem zur Behälterlokalisierung und Behälterreservierung beinhalten die Verfügbarkeit eines Web Services, der aus dem Anwendungssystem „Order Management“ aufgerufen werden kann. Mit Hilfe dieses Web Services soll ein Mitarbeiter aus dem Vertrieb oder dem Kundenservice

durch Lokalisierung eines Behälters herausfinden, in welchem Produktionsprozess sich ein zu fertigendes Produkt zu einer konkreten Bestellung befindet.

5 Modellierung

Folgende Ressourcen werden zur Entwicklung des Anwendungssystems benötigt:

- containers: Die Behälter („containers“) werden als eine Ressource angesehen.
- locations: Die Stationen im Produktionsprozess und die Läger in der Instandhaltung werden als Standorte („locations“) modelliert. An jedem Standort ist ein RFID-Gate vorhanden, das Behälter über ihren RFID-Transponder identifiziert.
- users: Zur Benutzerverwaltung, welche bei der Behälterreservierung verwendet wird, wird eine Ressource Benutzer („users“) benötigt.
- reservations: Die Reservierungen stellen eine Ressource „reservations“ dar.

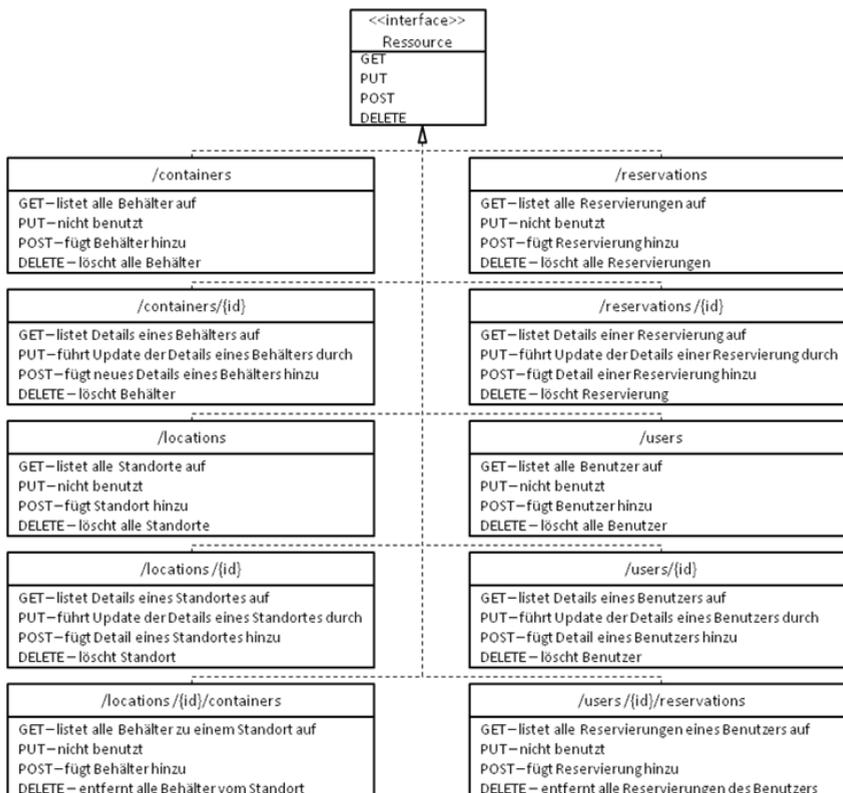


Abbildung 4: Ressourcenmodell

Alle Ressourcen verwenden das gleiche Interface mit den Methoden GET, PUT, POST und DELETE. Damit ergibt sich in einem Architekturstil REST folgende Modellierung der Ressourcen (Abbildung 4).

Die URIs für die Ressource Standorte lautet unter Vernachlässigung des Pfades (Hostname und optionaler Port): /locations. Eine HTTP-GET Anfrage auf die relative URI /locations unter Anforderung einer Repräsentation in einem XML-Format gemäß

```
GET /locations HTTP/1.1
Host: {Host}
Accept: application /xml
```

liefert als Antwort eine Liste der Standorte in einem XML-Dokument. Ein einzelner Standort mit ID 0815 wird dann unter der relativen URI /locations/0815 adressiert. Ein Zugriff auf alle Behälter, die am Standort mit ID 0815 vorgefunden werden, erfolgt über die relative URI /locations/0815/containers. Ein Benutzer mit ID 00001 ist über die relative URI /users/00001 adressierbar. Die Reservierungen eines Benutzers mit ID 00001 können über die relative URI /users/00001/reservations per HTTP-GET Request angefragt werden.

Für einen Behälter sind vier Zustände vorgesehen:

- available: Der Behälter ist (im Moment) verfügbar und kann reserviert werden.
- unavailable: Der Behälter ist nicht verfügbar.
- reparation: Der Behälter wird gewartet oder ist in Wartung.
- purgation: Der Behälter wird gereinigt bzw. ist in Reinigung.

Die möglichen Übergänge zwischen den vier Zuständen eines Behälters sind in Abbildung 5 dargestellt.

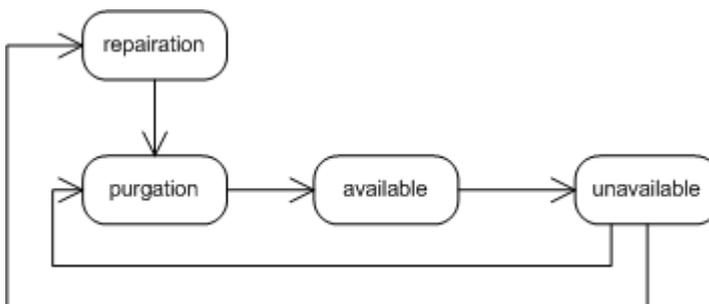


Abbildung 5: Zustände und deren Übergänge für Behälter

Eine Liste der verfügbaren Behälter ist über die relative URI /containers/?state=available adressierbar. Dabei unterteilt „?“ die URI in zwei

Teile, von denen „state=available“ als Query bezeichnet wird mit dem Query-Parameter state. Ein HTTP GET auf /containers/?state=available liefert somit eine Liste der (im Moment) verfügbaren Behälter. Für den Zustand „available“ sind zwei weitere Query-Parameter vorhanden:

- date: Tag, für den die Verfügbarkeit angefragt werden soll
- duration: Dauer (in Tagen) der Verfügbarkeit

Eine Anfrage per HTTP-GET nach der Verfügbarkeit eines Behälters ab 23.02.2010 für 3 Tage wird dann über die relative URI /containers/?state=available&date=23022010&duration=3 adressiert.

Entsprechend kann aus dem „Order Management“ – sofern eine auf REST basierender Web Service unterstützt wird – über die relative URI /containers/{id} per HTTP GET eine Repräsentation angefordert werden, die einen Link auf den Standort liefert, wo sich der Behälter gerade befindet. Da der Standort im Produktionsprozess mit der Produktionsstation assoziiert ist, kann aus der Antwort die Information über den Produktionsprozess eines im Behälter befindlichen Produktes abgeleitet werden.

6 Zusammenfassung

Der Einsatz von RFID-Technologien im Behältermanagement reduziert die Kosten und erlaubt eine Lokalisation von Behältern in Echtzeit. Die Behälterlokalisierung und -reservierung von beliebigen Positionen aus ist somit eine interessante Anwendung für ein mobiles Informationssystem. Werden die Standards des Web wie die Nutzung des Protokolls HTTP, die Verwendung von URIs sowie XML auf die Entwicklung von mobilen Informationssystemen übertragen, bietet sich die Verwendung des Architekturstils REST an. REST bietet u.a. die Verwendung einer gemeinsamen Schnittstelle, eine einfache Wiederverwendbarkeit und eine gute Skalierbarkeit. Wesentlich für die Erstellung eines Anwendungssystems auf Basis von REST sind Ressourcen und ihre Adressierbarkeit über URIs. Demzufolge sind bei der Planung von Anwendungssystemen die Identifizierung von Ressourcen und ihre Modellierung wesentliche Tätigkeiten. Auf Basis einer Anforderungsanalyse zur Erstellung eines Anwendungssystems zur Behälterlokalisierung und Behälterreservierung mit Hilfe von Anwendungsfällen wird die Identifizierung von Ressourcen aufgezeigt und der Umgang mit Ressourcen dargestellt. Daraus resultiert ein Ressourcenmodell. Ressourcen wie die hier identifizierten „containers“ (Behälter) können Zustände aufweisen, die bei dem Design des Anwendungssystems mit ihren Übergängen zu betrachten sind.

Literatur

Fielding RT (2000) Architectural styles and the design of network-based software architectures. Ph.D. thesis. University of California. Irvine.

- Fielding RT, Taylor RN (2002) Principled Design of the Modern Web Architecture. *ACM Transactions on Internet Technology*, 2:2: 115–150.
- Fleisch E, Christ O, Dierkes M (2005) Die betriebswirtschaftliche Vision des Internets der Dinge. In: Fleisch E, Mattern F (Hrsg) *Das Internet der Dinge - Ubiquitous Computing und RFID in der Praxis: Visionen, Technologien, Anwendungen, Handlungsanleitungen*. Springer.
- Franke W, Dangelmaier W (2006) (Hrsg.) *RFID – Leitfaden für die Logistik*. Gabler.
- Gillert F, Hansen WR (2007) *RFID für die Optimierung von Geschäftsprozessen*. Hanser.
- Golabeck K (2007) RFID im Behältermanagement. In: Koschke R, Herzog O, Rödiger KH, Ronthaler M. *INFORMATIK 2007 - Informatik trifft Logistik (Band 1)*. volume P-109 of *Lecture Notes in Informatics*. 87-90. Gesellschaft für Informatik e.V..
- Günther O, Kletti W, Kubach U (2008) *RFID in Manufacturing*. Springer.
- Lackner E, Zsifkovits H (2007) Die Rolle von Behältern in der Supply Chain. In: Engelhardt-Nowitzki C, Lackner E. *Chargenverfolgung - Möglichkeiten, Grenzen und Anwendungsgebiete*. 241 – 254.
- Liu D, Deters R (2008) Management of service-oriented systems. *Service Oriented Computing and Applications*. 2:51–64.
- Pautasso C, Zimmermann O, Leymann F (2008) RESTful Web Services vs. “Big” Web Services: Making the Right Architectural Decision. In: *Proceedings of the 17th International World Wide Web Conference*. Beijing, China. 805–814.
- RFC2616 www.w3.org/Protocols/rfc2616/rfc2616.html.
- RFC3986 www.ietf.org/rfc/rfc3986.txt.
- Strüker J, Gille D, Faupel T (2008) *RFID Report 2008 - Optimierung von Geschäftsprozessen in Deutschland*. IIG-Telematik. Albert-Ludwigs-Universität Freiburg. VDI nachrichten. Düsseldorf.
- Tilkov S (2009) *REST und HTTP – Einsatz der Architektur des Web für Integrationsszenarien*. dpunkt. Heidelberg.
- Thoree L, Melski A, Schumann M (2009) The impact of RFID on management of returnable containers. *Electron Markets*. 19:115–124.
- Wohlers G, Breitner MH (2008) (Hrsg.) *RFID-Anwendungen Einführung, Fallbeispiele und Szenarien in der Praxis*. Shaker.