

Entwicklung eines Software-Prototyps zur automatischen Erstellung nutzerspezifischer ETL-Dokumentation: Ein detailliertes Fallbeispiel für gestaltungsorientierte, problemzentrierte Forschung

Marcus Hofmann, Robert Krawatzek, Frieder Jacobi

Technische Universität Chemnitz

Fakultät für Wirtschaftswissenschaften, Professur Wirtschaftsinformatik II

Abstract

Die Durchführung von rigoroser und relevanter gestaltungsorientierter Forschung (engl. Design Science Research, DSR) gewinnt in der Wirtschaftsinformatik international an Akzeptanz und Bedeutung. Der vorliegende Beitrag verfolgt das Ziel, das Verständnis von DSR in der Wirtschaftsinformatik zu schärfen und DSR-Erfahrungen mit der Community zu teilen. Dafür wird ein Fallbeispiel, das ein umfangreiches, mit DSR durchgeführtes Forschungsprojekt zum Gegenstand hat, detailliert beschrieben und anschließend kritisch diskutiert.

1 Einleitung

Die gestaltungsorientierte Herangehensweise zur Problemlösung findet in der deutschen Wirtschaftsinformatik seit jeher Anwendung (Österle et al., 2010) und wird zunehmend im angelsächsischen Raum befürwortet (Chen, 2011). Die von Hevner et al. (2004) vorgeschlagenen Leitsätze sowie der von Peffers et al. (2007) angegebene Prozess für die Durchführung von rigoroser und relevanter gestaltungsorientierter Forschung (engl. Design Science Research, DSR) verhalf dem gestaltungsorientierten Problemlösungsansatz zur internationalen Akzeptanz.

Sowohl Hevner et al. (2004) als auch Peffers et al. (2007) nutzen in ihren Veröffentlichungen abgeschlossene und veröffentlichte Forschungsprojekte, um die Umsetzbarkeit und die Anwendung ihrer entwickelten Ergebnisse zu demonstrieren. Hevner et al. geben hierfür drei Beispiele an und veranschaulichen jeweils rückwirkend, wie diese die sieben

vorgeschlagenen DSR-Leitsätze befolgt haben. Peffers et al. demonstrieren ebenfalls rückwirkend für die vier möglichen Forschungseinstiegspunkte („problemzentriert“, „zielstellungszentriert“, „entwurfs- und entwicklungszentriert“ sowie „initiiert durch Auftraggeber/Kontext“) die Anwendung der DSR-Forschungsaktivitäten mithilfe eines Fallbeispiels. Der Einsatz von Fallbeispielen¹ veranschaulicht die Anwendung von DSR und erhöht das DSR-Verständnis (Hevner et al., 2004; Peffers et al., 2007).

Die von Peffers et al. (2007) präsentierte DSR-Demonstration leistet einen Beitrag zur Steigerung des DSR-Verständnisses, geht aber aufgrund der nachträglich zugeordneten DSR-Forschungsaktivitäten sowie der kompakten Darstellung der Fallbeispiele (jedes Forschungsprojekt wird auf maximal drei Seiten erläutert) aus Sicht der Autoren nicht genügend in die Tiefe und öffnet somit einen Interpretationsspielraum.

Um das DSR-Verständnis weiter zu schärfen, präsentiert der vorliegende Beitrag daher zunächst ein detailliertes DSR-Fallbeispiel, in dem ein abgeschlossenes Forschungsprojekt präsentiert wird, welches den DSR-Prozess vollständig durchlaufen hat (Kapitel 2). Die Struktur des Fallbeispiels orientiert sich an dem Vorgehen von Peffers et al. (2007). Abschließend folgt eine kritische Diskussion, in der dargelegt wird, inwiefern das Vorgehen des präsentierten Forschungsprojektes anzupassen ist, um die Rigorosität des DSR-Forschungsergebnisses zu steigern (Kapitel 3).

2 DSR-Fallbeispiel „Computer-Aided Data Warehouse Engineering“

Das Forschungsprojekt „Computer-Aided Data Warehouse Engineering“ (CAWE) der Technischen Universität Chemnitz hatte mit einer Laufzeit von 3 Jahren (August 2010 bis Juli 2013) die Entwicklung eines modellgetriebenen Vorgehens zur Unterstützung des Lebenszyklus von Business-Intelligence-Systemen (BI-Systemen) zum Forschungsgegenstand. Ein wesentlicher Schwerpunkt der Forschungsarbeit lag in der Entwicklung eines Software-Prototyps zur automatischen Erzeugung von Dokumentation für Extraktions-, Transformations- und Ladeprozesse (ETL-Prozesse) innerhalb bestehender BI-Systeme (Re-Dokumentation). Ausschlaggebend für die Entwicklung dieses Prototyps war eine zu Projektbeginn durchgeführte Umfrage (Gluchowski et al., 2011; Hofmann et al., 2012), welche einen entsprechenden Bedarf identifizierte. Das CAWE-Projekt ist demnach ein Fallbeispiel für DSR-Forschung mit einem problemzentrierten Einstiegspunkt in die Forschungsaktivitäten.

¹ Im Rahmen dieses Beitrages wird der Begriff „Fallbeispiel“ für die Bezeichnung von beispielhaften Anwendungsfällen (engl. Use Cases) verwendet und ist vom Begriff „Fallstudie“ (engl. Case Study) abzugrenzen.

Dieses Kapitel beschreibt das Vorgehen und die Ergebnisse des Projektes anhand der von Peffers et al. (2007) bezüglich des DSR-Prozesses vorgeschlagenen Aktivitäten.

2.1 Aktivität 1: Problemidentifikation und Motivation

Problemidentifikation

Eine in 2011 von CAWE durchgeführte Umfrage unter 119 Unternehmen hat ergeben, dass BI-Systeme gar nicht oder nur unzureichend dokumentiert werden (Gluchowski et al., 2011; Hofmann et al., 2012). BI-Architekturkomponenten werden lediglich von 36,1% der Befragten dokumentiert. Für den Fall, dass diese dokumentiert werden, betrifft das zu 79,1% die Berichte und Reports. Das Schlusslicht bilden ETL-Prozesse mit einer Dokumentationshäufigkeit von 53,5%.

Ferner wurde gezeigt, dass die Dokumentation häufig nur einmalig in der Entwurfs- oder der Entwicklungsphase der Systeme erfolgt. Das Aufwand-Nutzen Verhältnis für die Erstellung von Dokumentation wurde von den befragten Unternehmen ungünstig bewertet. So schätzten lediglich 48% der Umfrageteilnehmer den durch Dokumentation gestifteten Nutzen höher ein als den dadurch verursachten Aufwand. Als Gründe dafür wurden z.B. (I) der hohe personelle Aufwand bei der Dokumentationserstellung, (II) die zu geringe Haltbarkeit (sinkende Aktualität vorhandener Dokumentation im Zeitverlauf) sowie (III) die fehlende strukturelle und inhaltliche Eignung der Dokumentation für unterschiedliche Nutzergruppen genannt.

Motivation

Mit Hilfe der Literatur motivierte CAWE die Relevanz von ETL-Dokumentationen, in dem es den möglichen Nutzen für verschiedene Adressatengruppen herleitete (Jacobi et al., 2012). Da die ETL-Prozessmodellierung bis zu 80% der Entwicklungszeit in Data-Warehouse-Projekten in Anspruch nimmt (Greenfield, 1996), ist die Dokumentation von ETL-Prozessen – welche zudem laut Umfrage am seltensten dokumentiert werden – von besonderem Interesse. Von einer ETL-Dokumentation profitieren *Entwickler* und *Unternehmen* wie folgt.

Entwickler erhalten Informationen über den aktuellen technischen Aufbau des Systems, wodurch das Umsetzen von Änderungen und Erweiterungen erleichtert wird.

Unternehmen profitieren von stets aktueller Dokumentation durch Kosteneinsparungen, die sich u.a. aus folgenden Punkten ergeben:

- durch eine verbesserte Qualität und daraus resultierender gewonnener Zeit, die bisher benötigt wurde, um mangelhaft dokumentierte Systeme zu verstehen (Chikofsky & Cross, 1990),
- neue Mitarbeiter bekommen die Möglichkeit, sich schneller in die Systeme einzuarbeiten, um diese produktiv nutzen zu können und
- Expertenwissen liegt in expliziter Form vor, wodurch beim Ausscheiden von Wissensträgern keine mit hohem Aufwand zu füllende Wissenslücke entsteht.

2.2 Aktivität 2: Beschreibung der Zielstellungen

Im Zuge der Ausarbeitung der Problemstellung identifizierte CAWE folgende Punkte, die dafür verantwortlich sind, dass Unternehmen ETL-Prozesse nicht bzw. nur unzureichend dokumentieren: (I) zu hoher personeller Aufwand, (II) zu geringe Haltbarkeit und (III) fehlende strukturelle und inhaltliche Eignung für verschiedene Nutzergruppen.

Aus diesen wurden die Zielstellungen abgeleitet, welche eine nutzenbringende Dokumentation zu erfüllen hat:

- (I) kostengünstige Erstellung,
- (II) hochwertige Qualität und
- (III) nutzerspezifische Struktur und Inhalt.

Bei einer Analyse des Softwaremarktes wurde kein Werkzeug gefunden, das eine umfassende Dokumentation von ETL-Prozessen unterstützt und dabei die genannten Zielstellungen abdeckt (Jacobi et al., 2012). Daher verfolgte CAWE das Forschungsziel, einen Software-Prototyp (im Sinne des DSR-Artefakttyps „Instanziierung“ (March & Smith, 1995; Hevner et al, 2004)) zu entwickeln, der die Dokumentation von ETL-Prozessen unter Berücksichtigung der hergeleiteten Zielstellungen unterstützt.

2.3 Aktivität 3: Entwurf und Entwicklung

Entwurf einer Lösung ausgehend von den Zielstellungen

Nachdem Gluchowski & Kurze (2010) bereits gezeigt haben, dass der Aufwand für die Erstellung von Dokumentation für multidimensionale Datenstrukturen (Spezifika der Datenhaltungskomponenten) durch den Einsatz von automatisierten Dokumentationsprozessen um bis zu 75% gesenkt werden kann, wurde auf eine automatisierte Lösung zur Erfüllung von Anforderung I des Forschungsziels gesetzt.

Anschließend blieb zu klären, durch welche Eigenschaften sich eine qualitativ hochwertige Dokumentation (Anforderung II) auszeichnet. Nach Wallmüller (Wallmüller, 2001) ist eine hochwertige Dokumentation durch folgende acht Merkmale gekennzeichnet:

- *„Änderbarkeit*: Eignung von Dokumenten zur Ermittlung aller von einer Änderung betroffenen Dokumententeile und zur Durchführung der Änderung.
- *Aktualität*: Übereinstimmung der Beschreibung des Programms in der Dokumentation mit dem jeweils geltenden Zustand des Programms.
- *Eindeutigkeit*: Eignung von Dokumenten zur unmissverständlichen Vermittlung von Information an jeden Leser.
- *Identifizierbarkeit*: Eindeutige Ansprechbarkeit der Teile von Dokumenten, die Angaben zu einem abgegrenzten Sachverhalt geben, die den Leser interessieren.
- *Normkonformität*: Erfüllung der für die Erstellung von Dokumenten geltenden Vorschriften und Normen.
- *Verständlichkeit*: Eignung von Dokumenten zur erfolgreichen Vermittlung der darin enthaltenen Informationen an einen sachkundigen Leser.
- *Vollständigkeit*: Vorhandensein der für den Zweck der Dokumentation notwendigen und hinreichenden Informationen [hinreichende Informationen liegen dann vor, wenn ein Dokument so wenig Informationen wie möglich, aber dennoch alle zur Erfüllung des Dokumentzwecks nötigen Informationen beinhaltet; Anm. d. A.].
- *Widerspruchsfreiheit*: Nichtvorhandensein von einander entgegenstehenden Aussagen im Dokument.“

Um Anforderung III des Forschungsziels im Sinne einer geeigneten Präsentation des Dokumentationsinhalts zu erfüllen, empfiehlt (Wallmüller, 2001) verschiedene, an den Zielgruppeninteressen ausgerichtete, Instanzen der Dokumentation zu erstellen.

Da nicht alle von Wallmüller vorgeschlagenen Dokumentationsmerkmale in Abhängigkeit von der Nutzergruppe verschieden ausgeprägt sein müssen, wurde in Forward & Lethbridge (2002) und Krawatzek et al. (2011) eine Unterteilung der Merkmale von hochwertiger Dokumentation in zielgruppenabhängige (nach Forward & Lethbridge (2002) als Dokumentattribute bezeichnet) und zielgruppenunabhängige Merkmale (so genannte Erstellungsprozessattribute (Krawatzek et al., 2011)) vorgenommen. Demnach zählen die Merkmale *Eindeutigkeit*, *Verständlichkeit* und *Vollständigkeit* in die Rubrik der zielgruppenabhängigen Attribute (Dokumentattribute), wohingegen die Qualitäts-

merkmale *Änderbarkeit*, *Aktualität*, *Identifizierbarkeit*, *Normkonformität* und *Widerspruchsfreiheit* unabhängig von individuellen Bedürfnissen sind.

Die zu dokumentierenden ETL-Prozesse beinhalten bereits einen großen Teil der für die Erstellung einer Dokumentation nötigen Informationen in Form von Quelltext (Forward & Lethbridge, 2002). Für den Bereich Data Warehouse Engineering angepasste modellgetriebene Softwareentwicklungsansätze (Kurze, 2011; Mazón & Trujillo, 2008) beinhalten zudem – über die im Quelltext enthaltenen Informationen hinaus – weitere für die Erstellung von nutzerspezifischer Dokumentation erforderliche Informationen, die aus Metadaten über die verschiedenen Anwendungen gewonnen und in maschinenlesbaren Modellen abgelegt werden. Durch die automatisierte Erzeugung von ETL-Dokumentation direkt aus den vorliegenden Modellen werden bis auf Normkonformität alle Erstellungsprozessattribute erfüllt (Krawatzek et al., 2011):

- *Änderbarkeit* und *Aktualität*: die Automatisierung erlaubt es, Systemänderungen jederzeit durch eine Neugenerierung der Dokumentation mit minimalem Personalaufwand zu veröffentlichen und damit aktuell zu halten,
- *Identifizierbarkeit*: durch die verfügbaren Metadaten ist es möglich, Zusammenhänge zwischen einzelnen Objekten, die im Quelltext u.U. nicht abbildbar sind, in der Dokumentation wiederzugeben und
- *Widerspruchsfreiheit*: durch die automatische Erzeugung von Dokumentation aus Modellen wird in der Dokumentation genau der Grad an Widerspruchsfreiheit abgebildet, der in den Modellen vorliegt.

Des Weiteren erlaubt die Verwendung eines generischen Dokumentenformatstandards (bspw. DITA (Darwin Information Typing Architecture) (DITA, o.J.) oder DocBook (DocBook, o.J.)) als zentrales Zwischenformat bei der Erstellung der Zielausgabeformate die Einhaltung der Struktur- und Inhaltsvorgaben der ISO-IEC 26514 für Benutzerdokumentation (International Organization for Standardization, 2008), und erfüllt damit das in (Krawatzek et al., 2011) nicht betrachtete Erstellungsprozessattribut Normkonformität.

Um die Dokumentation nutzerspezifisch zu generieren, müssen die bereits vorhandenen Modelle um eine Konfigurationsmöglichkeit bei der Verknüpfung von Informationen erweitert werden. Diese Konfiguration ist in das modellgetriebene Vorgehen eingebettet, was im Ergebnis dazu führt, dass sie ebenfalls in Form von Modellen vorliegt. Über diesen Konfigurationsansatz wird es möglich, die bereitgestellten Inhalte (beeinflusst die Dokumentattribute Vollständigkeit und Eindeutigkeit), das Layout und das Ausgabeformat (beeinflussen Verständlichkeit) der Dokumentation nutzerspezifisch anzupassen.

Die dargestellten Überlegungen und Designentscheidungen (Automatisierung durch Anwendung der modellgetriebenen Softwareentwicklung, Verwendung eines Dokumentenformatstandards und Schaffen von Konfigurationsmöglichkeiten) führten zum Entwurf eines Architektur-Frameworks (ein Modell im Sinne der DSR (March & Smith, 1995; Hevner et al., 2004)), welches die Architektur der gesuchten Lösung beschreibt (vgl. Abbildung 1). Für eine detaillierte Beschreibung der einzelnen Komponenten des Architektur-Frameworks sei der interessierte Leser auf (Jacobi et al., 2012) verwiesen.

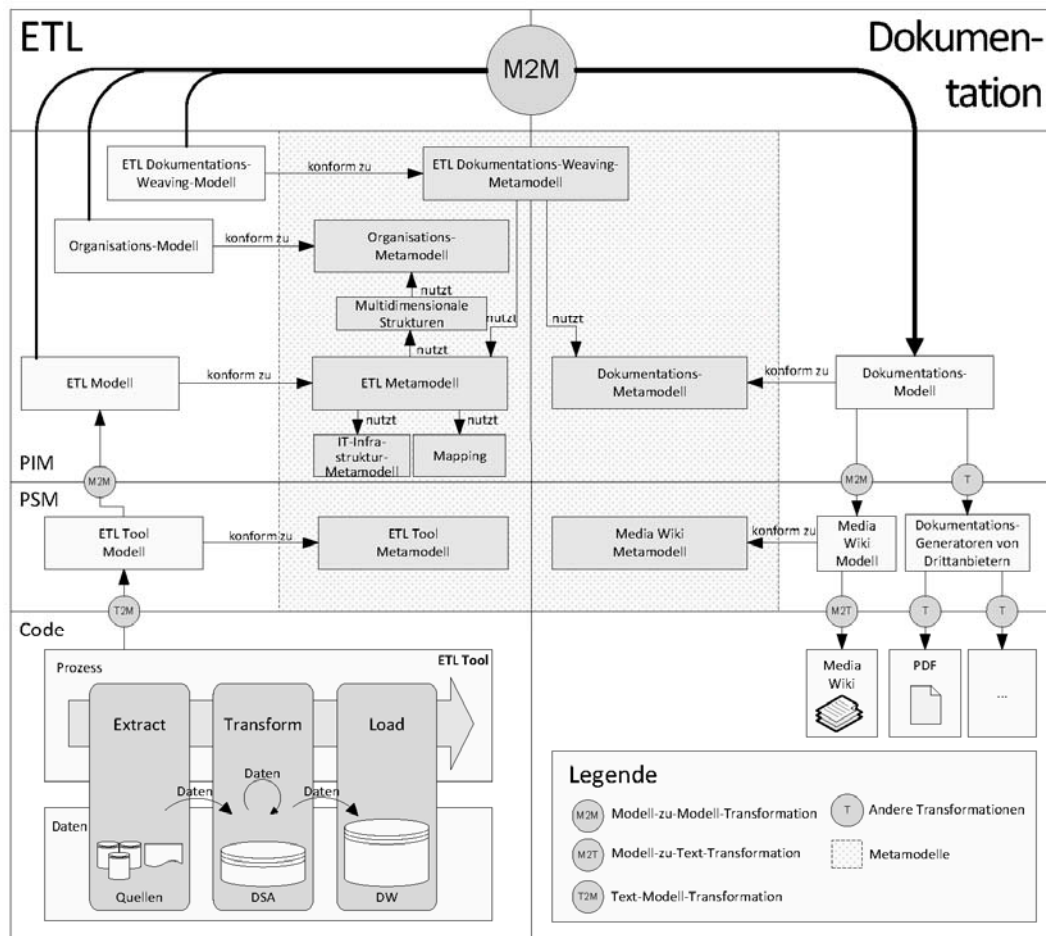


Abbildung 1: Architektur-Framework zur automatischen Erstellung nutzerspezifischer ETL-Dokumentation (Jacobi et al., 2012)

Entwicklung der Lösung

Das in Abbildung 1 dargestellte Framework wurde mit Hilfe des Eclipse Modeling Frameworks in Form des Software-Prototypen „DW Documenter“ implementiert.

2.4 Aktivität 4: Demonstration

Zur Demonstration des DW Documenters wurden für verschiedene Plattformen kleinere, von Praxispartnern bereitgestellte ETL-Prozesse dokumentiert. So konnte nachgewiesen werden, dass die entworfene Lösung in der Lage ist (Krawatzek et al., 2012):

- ETL-Prozesse automatisiert zu dokumentieren,
- detaillierte Informationen über Variablen sowie deren Lese- und Schreibzugriffe anzugeben (vgl. Abbildung 2, links),
- detaillierte Informationen über die bearbeiteten Felder einer Aktivität anzugeben,
- vollständige Lineage- & Impact Analyse pro Feld (vgl. Abbildung 2, rechts) durchzuführen und
- verschiedene Ausgabeformate wie Wiki-Seiten und PDF-Dokumente (vgl. Abbildung 2 und 3) zu erzeugen.

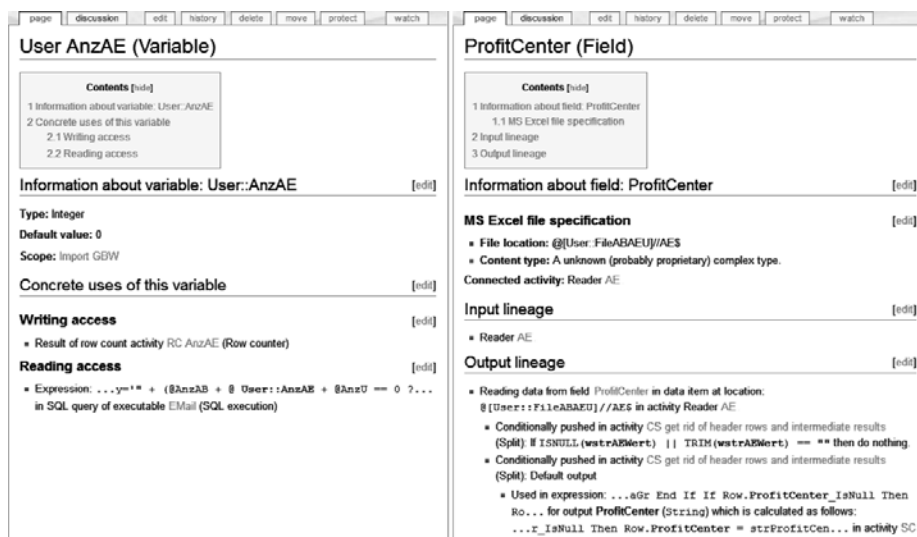


Abbildung 2: Mit DW Documenter erzeugte Dokumentation eines mit SSIS erstellten ETL-Prozesses im MediaWiki-Format (Krawatzek et al., 2012),
(Hyperlinks sind heller dargestellt).

2.5 Aktivität 5: Evaluation

Eine Möglichkeit zur Evaluation von Artefakten besteht nach Hevner et al. (2004) darin, den erzeugten Nutzen der Artefakte mit dem Nutzen von anderen Artefakten, die dasselbe Problem lösen, zu vergleichen. Eine vergleichende Evaluation des DW Documenters (genauer: des entworfenen Architektur-Frameworks) mit anderen Softwareprodukten, welche im Rahmen der Analyse des Softwaremarktes identifiziert wurden (vgl. Aktivität

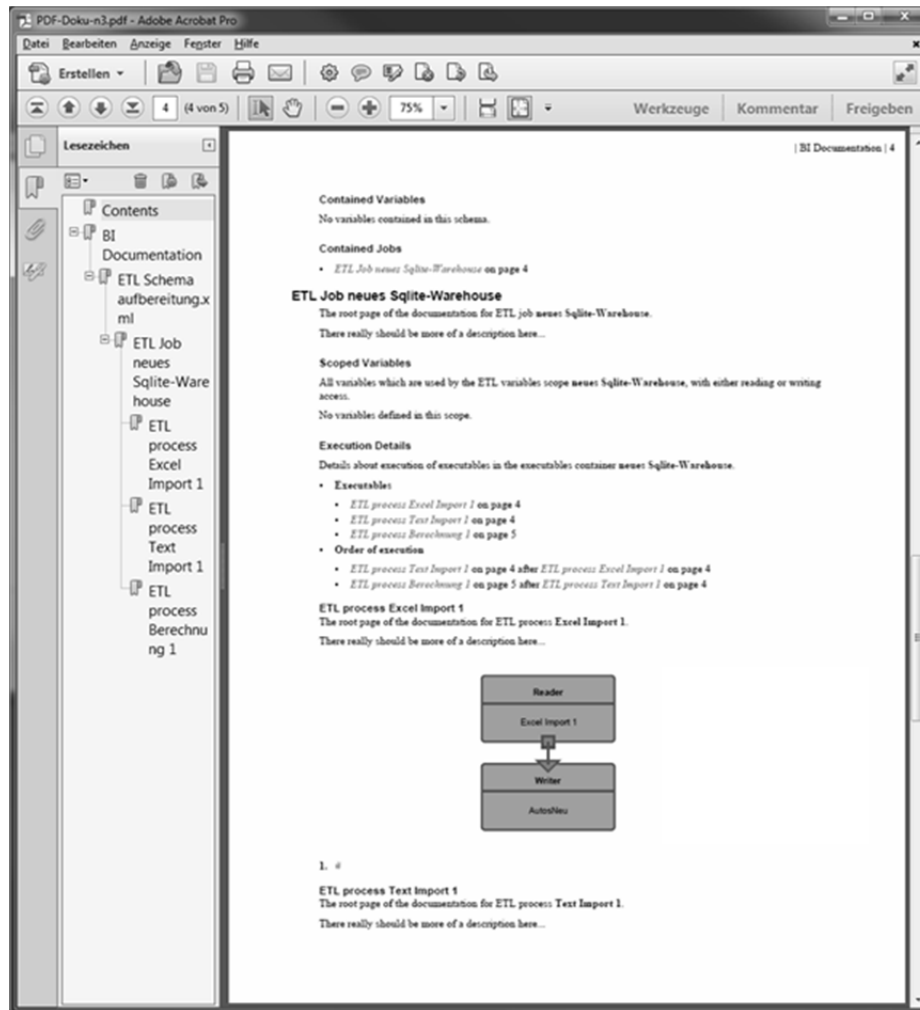


Abbildung 3: Mit DW Documenter erzeugte Dokumentation eines mit n^3 DataWarehouseBuilder erstellten ETL-Prozesses im PDF-Format inklusive grafischer Notation für die logische ETL-Prozessbeschreibung (Hyperlinks sind heller dargestellt).

2), war nicht möglich. Zielführend war hingegen ein Vergleich der durch die Softwarelösungen erzeugten Zielartefakte (ETL-Dokumentation). Die Grundlage für den Vergleich bilden die in Aktivität 2 definierten und in Aktivität 3 spezifizierten Zielstellungen:

- I. Erstellung von kostengünstiger Dokumentation,
- II. Einhaltung der acht Merkmale von qualitativ hochwertiger Dokumentation (Dokument- und Erstellungsprozessattribute) und
- III. Bereitstellung von Möglichkeiten zur nutzerspezifischen Konfiguration.

Durch die Einschränkung auf automatisierte Lösungsansätze zur Dokumentationserstellung unterstützen alle zur Untersuchung herangezogenen Softwarewerkzeuge – inkl. des

DW Documenters – eine kostengünstige Erstellung von Dokumentation (Zielstellung I) und erfüllen die Erstellungsprozessattribute (erster Aspekt von Zielstellung II).

Allerdings konnte der DW Documenter durch eine tiefgehende Metadatenanalyse signifikante Verbesserungen im Bereich der Dokumentattribute (zweiter Aspekt von Zielstellung II) erzielen.

Weiterhin konnte festgestellt werden, dass der Prototyp DW Documenter als einziges Werkzeug die Möglichkeit zur nutzerspezifischen Konfiguration unterstützt (Zielstellung III). Durch diese lassen sich nicht nur die Bedürfnisse unterschiedlicher Zielgruppen abdecken, sondern auch nutzerspezifische Verbesserungen im Bereich der Dokumentattribute erzielen:

- Konfiguration des Inhalts (beeinflusst *Vollständigkeit* und *Eindeutigkeit*),
- Konfiguration des Layouts (beeinflusst *Verständlichkeit*) und
- Konfiguration des Ausgabeformats (beeinflusst *Verständlichkeit*).

Die dargestellte Argumentation bezüglich der Zielattribute macht deutlich, dass der DW Documenter den untersuchten Dokumentationslösungen im Bereich ETL überlegen ist.

2.6 Aktivität 6: Kommunikation der (Zwischen-)Ergebnisse

Um der Bedeutung der einzelnen DSR-Aktivitäten in Hinblick auf ein relevantes Gesamtergebnis Rechnung zu tragen (Gregor & Hevner, 2013), wurden die Zwischenergebnisse wie folgt in der wissenschaftlichen Gemeinschaft publiziert und frühzeitig zur Diskussion gestellt:

- Aktivität 1: Gluchowski et al. (2011), Hofmann et al. (2012),
- 2 und 3: Jacobi et al. (2012) und
- Aktivität 4: Krawatzek et al. (2012).

Das vorliegende Paper fasst die Ergebnisse der DSR-Aktivitäten als Fallbeispiel zusammen und ergänzt die bisherigen Veröffentlichungen um die Aktivität 5 „Evaluation“.

Neben der Veröffentlichung innerhalb der wissenschaftlichen Gemeinschaft wurden die Projektergebnisse in Form eines Fachvortrags auf einer praxisorientierten Konferenz (Hofmann et al, 2011) sowie im Rahmen eines Messebesuches (CeBIT‘2012) der Praxis kommuniziert.

3 Kritische Diskussion

Die detaillierte Beschreibung des DSR-Fallbeispiels CAWE demonstriert, dass der DSR-Prozess nach Peffers et al. (2007) erfolgreich umgesetzt werden kann. Trotz erfolgreichem Durchlauf des DSR-Prozesses inkl. Kommunikation der Ergebnisse beinhaltet das CAWE-Beispiel Verbesserungspotential, welches im Folgenden diskutiert und in zukünftigen DSR-Projekten berücksichtigt werden kann.

Im Rahmen der dritten Aktivität „Entwurf einer Lösung“ sollte zur Darstellung von Modellen (betrifft im CAWE-Beispiel das Architektur-Framework) auf standardisierte Sprachen zurückgegriffen werden. Durch die Verwendung von standardisierten Sprachen wird ein gemeinsames Grundverständnis sichergestellt und die eigentliche Lösung kann auf einer standardisierten Basis (beispw. mit Gutachtern) diskutiert werden. Ist zur Darstellung eines spezifischen Problems bzw. einer spezifischen Lösung keine standardisierte Sprache vorhanden, sollten zunächst die dafür notwendigen Konstrukte (DSR-Artefakttyp) eigenständig entwickelt werden (identifizierte Forschungslücke; Anwendung von DSR möglich).

Findet die Demonstration von DSR-Ergebnissen in Zusammenarbeit mit Praxispartnern statt, empfiehlt es sich, diese als Interviews zu führen und zu dokumentieren. Zeigt sich im Rahmen dieser Demonstration, dass Zielstellungen noch verfeinert werden müssen, dienen die dokumentierten Gespräche als Nachweis für die notwendigen Anpassungen. Ferner können diese nach dem erneuten Durchlauf eines „Generate/Test Cycle“ (vgl. Hevner et al., 2004) als Motivation für das Publizieren von verbesserten DSR-Ergebnissen genutzt werden. Im Rahmen der Demonstration des DW Documenters wurde beispielsweise der Bedarf an Metadatenanalysen für Lineage- und Impact-Angaben identifiziert (verbessert „Vollständigkeit“ einer ETL-Dokumentation) und im nächsten Iterationsschritt im Architektur-Framework berücksichtigt. Die Publikation des verbesserten Frameworks war aufgrund fehlender Transparenz nicht möglich, da die Praxisanforderungen nicht dokumentiert wurden und somit nicht als Motivation für eine erneute Iteration angegeben werden konnten.

Des Weiteren empfiehlt es sich, bereits im Rahmen der Aktivität 2 Metriken zu definieren, welche eine Messung der aufgestellten Zielstellungen erlauben. Diese Metriken ermöglichen es, eine quantitative und somit eindeutige Evaluation (Aktivität 5) durchzuführen. Ein Ausweichen auf die im CAWE-Beispiel angewendete argumentative Evaluation ist somit hinfällig.

Abschließend sei auf die Publikation von Gregor & Hevner (2013) verwiesen, welche für die erfolgreiche Kommunikation von DSR-Ergebnissen (Aktivität 6) ein „DSR know-

ledge contribution framework“ sowie ein „DSR communication schema“ bereitstellt. Diese vielversprechenden Kommunikationshinweise wurden erst nach Abschluss des CAWE-Projektes veröffentlicht und konnten daher nicht berücksichtigt werden.

4 Anmerkungen

Das „Computer-Aided Data Warehouse Engineering“ (CAWE) Projekt, in dessen Rahmen der vorliegende Beitrag entstanden ist, wird mit Mitteln des ESF und des Freistaates Sachsen gefördert.

5 Literaturverzeichnis

- Chen, H.: Editorial: Design science, grand challenges, and societal impacts. *Journal ACM Transactions on Management Information Systems*. 2, 1 (2011).
- Chikofsky, E.J., Cross, J. H., I.: Reverse engineering and design recovery: a taxonomy. *IEEE Software*. 7, 1, pp. 13–17 (1990).
- DITA, <http://www.oasis-open.org/committees/dita>.
- DocBook, <http://www.oasis-open.org/docbook>.
- Forward, A., Lethbridge, T.C.: The Relevance of Software Documentation, Tools and Technologies: a Survey. In: *Proceedings of the 2002 ACM symposium on Document engineering - DocEng'02*. pp. 26–33, ACM Press, New York, USA (2002).
- Gluchowski, P., Hofmann, M., Jacobi, F., Krawatzek, R., Müller, A.: *Business-Intelligence-Umfrage 2011: Softwaregestütztes Lebenszyklusmanagement und aktuelles Dokumentationsgeschehen für Business-Intelligence-Systeme*. Technische Universität Chemnitz, Chemnitz, Deutschland (2011).
- Gluchowski, P., Kurze, C.: Modellierung und Dokumentation von BI-Systemen. *CONTROLLING - Zeitschrift für erfolgsorientierte Unternehmensplanung*. 22, pp. 676–682 (2010).
- Greenfield, L.: Don't let data warehousing gotchas getcha. *Datamation*, 42(5), pp. 76–77 (1996), zitiert nach: Inmon, B.: *The Data Warehouse Budget*. *DM Review Magazine*. p. 2 (1997).
- Gregor, S., Hevner, A.R.: Positioning and Presenting Design Science Research for Maximum Impact. *Management Information Systems Quarterly*. 37, 2, pp. 337–355, (2013).

- Hevner, A.R., March, S. T., Park, J., Ram, S.: Design Science in Information Systems Research. *Management Information Systems Quarterly*. 28, 1, pp. 75–105 (2004).
- Hofmann, M., Gluchowski, P., Jacobi, F., Kurze, C.: Computer-Aided Warehouse Engineering: Dokumentation und Modellierung komplexer Data-Warehouse-Systeme. Vortrag. 11. Europäische TDWI Konferenz, München, Deutschland (2011).
- Hofmann, M., Müller, A., Jacobi, F., Krawatzeck, R.: Umfrage 2011: „Dokumentation von Business-Intelligence-Systemen“ - Ergebnisse und Auswertung. In: Tagungsband der Multikonferenz Wirtschaftsinformatik 2012. pp. 1091–1104, GITO Verlag, Berlin (2012).
- International Organization for Standardization: Systems and software engineering - Requirements for designers and developers of user documentation (ISO-IEC 26514). (2008).
- Jacobi, F., Krawatzeck, R., Hofmann, M.: Meeting the Need for ETL Documentation: A Model-driven Framework for Customizable Documentation Generation. In: Proceedings of the Americas Conference on Information Systems. (AMCIS'12). Paper 23, Seattle, USA (2012).
- Krawatzeck, R, Jacobi, F., Müller, A., Hofmann, M.: Konzeption eines Frameworks zur automatisierten Erstellung nutzerspezifischer IT-Systemdokumentationen. In: Workshop Business Intelligence 2011 (WSBI'11) der GI-Fachgruppe Business Intelligence, Business Intelligence - Impulse für die Forschung oder Impulse durch die Forschung. pp. 15–26, CEUR Workshop Proceedings, Stuttgart (2011).
- Krawatzeck, R., Jacobi, F., Hofmann M.: CAWE DW Documenter: A Model-driven Tool for Customizable ETL Documentation Generation. In: Proceedings of the 31st International Conference on Conceptual Modeling (ER'2012). Florence, Italy (2012).
- Kurze, C.: Computer-Aided Warehouse Engineering: Anwendung modellgetriebener Entwicklungsparadigmen auf Data-Warehouse-Systeme. Verlag Dr. Kovač, Hamburg (2011).
- March, S. T., & Smith, G. F.: Design and natural science research on information technology. *Decision Support Systems*. 15, 4, 251–266 (1995).
- Mazón, J.-N., Trujillo, J.: An MDA approach for the development of data warehouses. *Decision Support Systems*. 45, 1, pp. 41–58 (2008).
- Österle, H., Becker, J., Frank, U., Hess, T.; Karagiannis, D., Krcmar, H., Loos, P., Mertens, P., Oberweis, A., Sinz, E. J.: Memorandum on design-oriented information systems research. *European Journal of Information Systems*, 20, 1, pp. 7–10 (2010).

Peppers, K., Tuunanen, T., Rothenberger, M., Chatterjee, S.: A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*. 24, 3, pp. 45–77 (2007).

Wallmüller, E.: Die Rolle der Dokumentation in Software-Projekten. *Software-Qualitätsmanagement in der Praxis: Software-Qualität durch Führung und Verbesserung von Software-Prozessen*. pp. 149–156 Hanser Fachbuch (2001).