BI Spektrum
tdwi WISSEN
ONLINETHEMENSPECIAL

Fachartikel

Ad-hoc-Datenintegration mittels Entity-Augmentation-Systemen

von Maik Thiele

Der Begriff Big Data wird meist mit den Herausforderungen und Möglichkeiten des heutigen Wachstums von Datenvolumen und -geschwindigkeit assoziiert, ist aber auch durch die zunehmende Vielfalt von Daten gekennzeichnet. Das Spektrum an Datenquellen reicht von Sensornetzwerken, Protokollinformationen aus Industriemaschinen bis hin zu Log- und Click-Streams von immer komplexeren Softwarearchitekturen und Anwendungen. Darüber hinaus gibt es einen stetigen Zuwachs kommerzieller bzw. öffentlich verfügbarer Daten wie zum Beispiel Daten aus sozialen Netzwerken wie Twitter oder Open Data. Immer mehr Unternehmen sind bestrebt, alle diese vorhandenen Arten von Daten in ihren Analyseprojekten zu nutzen, um zusätzliche Erkenntnisse zu gewinnen oder neue Funktionen in ihren Produkten zu ermöglichen.

Dem Bedürfnis einer unternehmensweiten, integrierten Sicht aller relevanten Daten wurde klassischerweise mit Hilfe relationaler Data-Warehouse-Infrastrukturen entsprochen. Diese sind jedoch aufgrund der notwendigen Schemadefinitionen wie auch der starren und kontrollierten ETL-Prozesse, die wohldefinierte Eingabe- und Zielschemata voraussetzen, nicht flexibel genug, um situationsbezogen Daten unterschiedlichster Struktur aufzunehmen. Von den technischen Herausforderungen abgesehen ist es oftmals gar nicht wünschenswert, alle in einer Big-Data-Landschaft anfallenden Daten zu integrieren, da deren zukünftige Anwendungsfälle zumeist unbekannt sind.

Aufgrund dieser Entwicklung hin zu einer agilen und explorativen Datenanalyse entstanden neue Prinzipien des Informationsmanagements wie zum Beispiel Data-Lake-Architekturen [MBC15; OLe14] oder MAD [CDD09], die darauf abzielen, Daten jeden Formats in einfacher Weise aufzunehmen (engl. data ingest). Dies erleichtert zwar die Datenübernahme enorm, verschiebt den Integrationsaufwand jedoch nur auf einen späteren Zeitpunkt und macht diesen zum Teil des eigentlichen Analyseprozesses. Gleichzeitig ist der Aspekt der Integration von Daten auch meist der aufwendigste und teuerste Schritt in vielen Datenanalyseprojekten. Aktuellen Studien zufolge verwenden Informationsarbeiter und Data Scientists 50–80 Prozent ihrer Zeit mit der Suche und der Integration von Daten, bevor die eigentliche Analyse beginnen kann [Loh14]. Da die exakte Datenintegration als sogenanntes „AI-vollständiges–Problem“ bezeichnet wird [HRO06], das im Allgemeinen die Validierung durch Menschen erfordert, ist eine Automatisierung dieser Aufgabe nicht absehbar.

Aus diesem Grund sind in den letzten Jahren verschiedene Systeme entstanden, die im Kern auf die analytische Mächtigkeit relationaler Systeme setzen, diese jedoch um zusätzliche Fähigkeiten erweitern, um zur Anfragelaufzeit Daten aus verschiedensten Quellen nutzen zu können. Anstatt also wie bisher alle Daten in relationale Schemata zu integrieren, sobald sie verfügbar sind, werden hier fokussiert nur bestimmte Daten basierend auf konkreten Informationsbedürfnissen und analytischen Fragestellungen identifiziert, abgerufen und integriert. Aus diesem Grund spricht man auch von Ad-hoc-Datenintegration.

Entity-Augmentation-Systeme

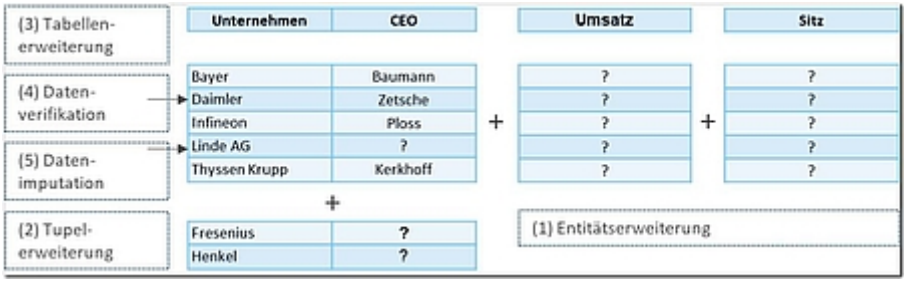


Abb. 1: Beispiel für verschiedene Erweiterungsanfragen

Die wichtigsten und bekanntesten Vertreter in diesem Bereich sind die sogenannten Entity-Augmentation-Systeme (EAS) wie zum Beispiel REA [ETB15], InfoGather [YGC12; ZaC13], WWT [PiS12; SaC14] und der Mannheim Search Join Engine [LRR15]. Diese ermöglichen die Verarbeitung von Entity-Augmentation-Anfragen (EAQ). Solch eine EAQ, die in Abbildung 1 beispielhaft dargestellt ist, benötigt als Eingabe eine Menge von Entitäten, zum Beispiel Unternehmen und ein per Schlüsselwort definiertes Attribut, wie „Umsatz“ oder „Sitz“, das für diese Entitäten bisher nicht definiert wurde und für das noch keine Daten vorliegen. Das Ergebnis der Abfrage sollte dann jede der vorgegebenen Entitäten mit einem Wert für das abgefragte Attribut assoziieren, indem automatisch die passenden Datenquellen identifiziert und ein Verbund mit dem lokalen Datensatz realisiert werden.

Abgrenzung zu Knowledge Bases

Im Allgemeinen ist jede Art von Datenquelle zur Beantwortung von Entity-Augmentation-Abfragen verwendbar. Wissensdatenbanken (engl. knowledge base) wie etwa YAGO [SKW07] könnten einige EAQ leicht beantworten, allerdings nur, wenn das gesuchte Attribut für die gegebenen Entitäten in dieser Wissensbasis definiert ist. Da Aufbau und Wartung von Wissensdatenbanken aber sehr aufwendig sind, bleibt deren Umfang per se beschränkt. Die meisten Entity-Augmentation-Systeme wie beispielsweise REA oder die Mannheim Search Engine verwenden daher große Webtabellenkorpora wie den Dresden Web Table Corpus (DTWC) [EBH15] oder den WDC [WDC15] als Datengrundlage.

Während für Wissensdatenbanken die gesuchten Informationen wesentlich einfacher zu identifizieren wären, haben Webtabellen den Vorteil, dass sie mehr Long-Tail-Informationen bieten und nicht darauf angewiesen sind, dass das abgefragte Attribut in einer zentralen Wissensbasis definiert wird. Natürlich sind diese Ansätze nicht nur auf Webtabellen als Datenquelle beschränkt, sondern können prinzipiell auch mit jeder anderen großen Sammlung heterogener Datenquellen, wie zum Beispiel Data Lakes, verwendet werden.

Beispielszenario

Eine der wichtigsten Aufgaben eines EAS besteht darin, aus einer Vielzahl von Kandidatenquellen ein konsistentes Ergebnis (engl. augmentation) zu ermitteln. Bei einer größeren Anzahl von Entitäten ist nicht davon auszugehen, dass eine einzige Webtabelle alle notwendigen Informationen, wie zum Beispiel alle Umsätze der angefragten Unternehmen, beinhaltet. Stattdessen muss das finale Ergebnis aus mehreren Webtabellen zusammengesetzt werden. Die dabei entstehenden Probleme sollen an dem in Abbildung 2 dargestellten Szenario besprochen werden, wobei die Anfrage oben als Tabelle und die darunterliegenden verfügbaren Datenquellen als Kandidaten dargestellt sind.

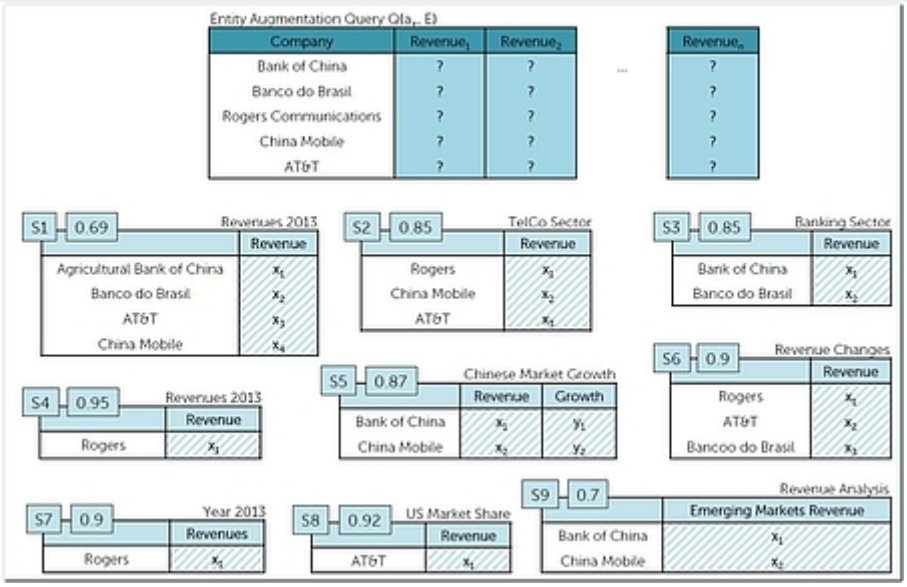


Abb. 2: Beispiel für Entitätserweiterung: Anfragetabelle und Datenquellen (die Zahlen bezeichnen hier den Relevanz-Score zur Entity-Augmentation-Anfrage)

Die Abfragetabelle besteht aus einer Menge von Unternehmen und dem Erweiterungsattribut „revenue“. Die Datenquellen unterscheiden sich in der Zahl der Entitäten, den bereitgestellten Attributen („revenue“ oder „emerging markets revenue“) und ihrem Kontext, zum Beispiel dem Titel der Tabelle. Des Weiteren ist jede Datenquelle mit einem Relevanz-Score versehen, der zum einen die Güte des Matching zum gesuchten Attribut „revenue“ bemisst, aber auch die Qualität der Datenquelle selbst. Für Webtabellen wären hierfür der PageRank oder der Alexa-Score, ein Amazon-Dienst der Statistiken zu Webseiten bereitstellt, denkbar. In Data-Lake-Szenarien sind die Aktualität beziehungsweise das Alter einer Datenquelle oder auch das Vertrauen in die Ersteller eines Datensatzes relevante Kriterien. Schaut man allein auf den Relevanz-Score, so würde ein naiver Algorithmus die Datenquelle S4 für „Rogers“, S8 für „AT&T“, S5 für „Bank of China“ und „China Mobile“ und schließlich S3 für „Banco do Brasil“ wählen. Somit würde für nahezu jede Entität eine andere Datenquelle verwendet werden.

Konsistenz und Minimalität

Aus Sicht der Konsistenz wäre das eine denkbar schlechte Lösung. In [DSS13] wird argumentiert, dass bei der Datenintegration und insbesondere bei der Auswahl der zu integrierenden Quellen dem Motto „weniger ist mehr“ gefolgt werden sollte. Die Auswahl zu vieler Quellen erhöht nicht nur die Integrationskosten, sondern kann auch die Qualität der Ergebnisse verschlechtern, wenn Quellen mit niedriger Qualität übernommen werden. Das Hinzufügen von S8 im Beispiel führt zu einem inkonsistenten Ergebnis, da hier der Umsatz in den USA statt, wie in den anderen Datenquellen, der Gesamtumsatz ausgewiesen wird.

Die Auswahl einer großen Anzahl von Quellen erhöht zusätzlich den Verifizierungsaufwand für die Benutzer, die mitunter einzelne Datensätze überprüfen möchten, bevor sie diese für weitere Analysen verwenden. Um die Zahl der Quellen zu verringern, muss der Ergebnisdefinition neben der Eigenschaft der Konsistenz noch die der Minimalität hinzufügt werden. Im laufenden Beispiel würde ein solches Ergebnis c1 (cover) aus S2 und S3, also zwei statt vier Quellen, bestehen. Die durchschnittliche Relevanz der Quellen ist dafür etwas gesunken.

Datenvielfalt

Erschwerend kommt hinzu, dass eine EAQ, ähnlich wie typische Suchanfragen im Web, nur durch eine Menge von Schlüsselwörtern beschrieben wird und daher unterspezifiziert ist. Dies wird der Komplexität und Vielfalt der Datenquellen kaum gerecht. Zum Beispiel wurde das Erweiterungsattribut in Abbildung 2 einfach mit „Revenue“ bezeichnet. Allerdings ist das Konzept tatsächlich komplexer und weist mit „US revenue“ (S8) oder „Emerging Markets Revenue“ (S9) und abgeleiteten Attributen wie „Revenue growth“ (S5 und S6) viele Varianten auf. Darüber hinaus sind viele Arten von Werten mit zeitlichen oder räumlichen Einschränkungen verbunden, beispielsweise mit unterschiedlichen Gültigkeitsjahren, oder sie wurden auf verschiedene Arten gemessen, beispielsweise unter Verwendung verschiedener Währungen.

Im REA-System wird diesem Problem dadurch Rechnung getragen, dass nicht nur ein Ergebnis, sondern eine gerankte Liste der Top-K-Ergebnisse erzeugt wird, aus denen der Nutzer eines für seine Analysezwecke wählen kann. Mit Hilfe komplexer Ähnlichkeitsmaße wird gewährleistet, dass innerhalb eines Ergebnisses die Werte konsistent sind, also zum Beispiel die gleiche Einheit besitzen. Im Beispiel in Abbildung 2 wäre neben dem bereits gezeigten Ergebnis c1 bestehend aus S2 und S3 auch ein alternatives Ergebnis c2 bestehend aus S1 und S4 denkbar. Es hat eine schlechtere durchschnittliche Relevanz als das erste Ergebnis, jedoch ist in beiden Quellen die gleiche Jahreszahl (hier 2013) ausgewiesen, was sich hinsichtlich der Konsistenz positiv auswirkt.

Ein weiterer, nicht zu vernachlässigender Aspekt ist der explorative Charakter von Entitätserweiterungsanfragen. Zum Beispiel möchten Benutzer, ähnlich wie bei der Verwendung einer Web-Suchmaschine, über neue Aspekte ihres Informationsbedürfnisses „stolpern“, die sie in die Lage versetzen, diese besser zu spezifizieren oder zu erweitern. Ein Beispiel wäre ein drittes Ergebnis c3 auf der Grundlage von S6 und der zweiten Spalte von S5, das beide Umsatzveränderungen, statt des absoluten Umsatzes, darstellt.

Diversität

Des Weiteren wird innerhalb der Top-k-Ergebnisliste eine hohe Diversität gefordert, um zu vermeiden, dass sich die Ergebnisse zu stark ähneln. Auch in diesem Punkt gibt es wieder Ähnlichkeiten zur Websuche, bei der ebenfalls die Interessantheit eines Dokuments bezüglich der anderen Dokumente der Ergebnisliste zu beachten ist. Angesichts der bereits ermittelten Lösungen c1, c2 und c3 wäre ein solches redundantes Ergebnis die Abdeckung bestehend aus S1 und S7. Da sich S7 nur oberflächlich von der Quelle S4 (Teil von c2) unterscheidet, wäre das Ergebnis c2 sehr ähnlich. Solche leichten Variationen bereits existierender Lösungen sollten daher vermieden werden, da diese dem Top-k-Ergebnis wenig neue Informationen hinzufügen würde. Die Forderung nach einer diversen Ergebnisliste ist insbesondere vor dem Hintergrund einer hohen Zahl an Duplikaten im Web [LDL13; DMP12] noch einmal zu betonen. Ähnliches gilt auch für Data-Lake-Szenarien, in denen die gleichen Informationen in verschiedensten Dokumentversionen mehrfach vorkommen können.

Zusammengefasst können für Entity-Augmentation-Systeme folgende Zielstellungen abgeleitet werden: Für eine EAQ, die aus einer Menge von Entitäten besteht, und ein gesuchtes Attribut soll das EAS eine diversifizierte Top-k-Liste alternativer Ergebnisse (engl. augmentations) liefern, die zum einen relevant, aber zum anderen auch in sich konsistent und minimal sind. Der interessierte Leser sei auf [ETB15] verwiesen, wo verschiedene Algorithmen (Greedy, evolutionäre Algorithmen) vorgestellt werden, die in der Lage sind, dieses Problem approximativ zu lösen.

Weitere Augmentation-Systeme

Neben den vorgestellten Entity-Augmentation-Anfragen wurde noch eine Vielzahl von Erweiterungen vorgeschlagen: Die Mannheim SearchJoin Engine [LRR15] etwa unterstützt auch uneingeschränkte Anfragen, für die zu einer Entitätsmenge nicht ein bestimmtes Attribut, sondern alle vorhandenen Attribute samt ihren Werten ermittelt werden. Neben der horizontalen Erweiterung können Tabellen auch vertikal ergänzt werden, indem auf Grundlage einer Tabelle mit Beispielausprägungen weitere passende Tupel identifiziert werden. Hier spricht man von einer Tupelerweiterung [PiS12], die in Abbildung 1 beispielhaft für die Unternehmen Fresenius und Henkel dargestellt ist.

Treibt man dieses Prinzip auf die Spitze, so können auch lediglich die Spaltenbezeichner definiert werden und die Tabelle wird automatisch mittels Tabellenerweiterung [PiS12] befüllt. Auch hier ist zur finalen Bewertung des Ergebnisses der Nutzer zu involvieren. Des Weiteren lassen sich diese Techniken verwenden, um zum einen fehlende Werte zu imputieren [ATE15], das heißt mit Werten aus externen Quellen zu ergänzen, und zum anderen zu validieren, das heißt auf ihre Gültigkeit beziehungsweise Aktualität zu überprüfen.

In [100] wurden zusätzlich noch die Operationen Sideways und Downwards entwickelt, um die Exploration von IS-A- beziehungsweise HAS-A-Beziehungen zu ermöglichen. Mittels Sideways werden für eine Eingabeentität weitere Entitäten ermittelt, die vom gleichen Typ sind. Für die Eingabeentität „Daimler“ wären das zum Beispiel alle weiteren deutschen Autobauer.

Durch die Anwendung der Operation Downwards können Eigenschaften von Entitäten ermittelt werden, zum Beispiel die Vorstandsmitglieder eines Unternehmens [CLK16].

Zum jetzigen Zeitpunkt sind diese Techniken nicht in kommerziellen Produkten zu finden. Jedoch existieren einige öffentlich verfügbare Softwareprodukte, in denen Teile dieser Ansätze integriert sind. So bietet etwa Google Tables [Tab18] eine Suchmaschine für Webtabellen, die darüber hinaus mit einer Exportfunktion in Google Fusion Tables [FuT18] integriert werden können. Letzteres stellt eine Vielzahl von Funktionalitäten bereit, um verschiedene Datensätze halbautomatisch miteinander abzugleichen.

Auch wenn wohl noch einige Jahre bis zur Marktreife dieser Prototypen vergehen werden, lässt sich feststellen, dass die hier dargestellten Techniken das Potenzial besitzen, bisher aufwendige Schritte wie die Informationsbeschaffung und -integration wesentlich zu vereinfachen. Gerade im Zuge der stetig wachsenden Datenvielfalt und -menge sind diese Systeme daher die perfekte Antwort, um die kontinuierlich steigenden Kosten von Datenintegrationsprojekten zu beherrschen.

Literatur & Links

[ATE15] Ahmadov, A. / Thiele, M. / Eberius, J. / Lehner, W. / Wrembel, R.: Towards a Hybrid Imputation Approach Using Web Tables. BDC 2015, S. 21–30

[CDD09] Cohen, J. / Dolan, B. / Dunlap, M. / Hellerstein, J. M. / Welton, C.: MAD skills: new analysis practices for big data. Proceedings VLDB Endowment 2, August 2009, S. 1481–1492

[CLK16] Fernando Chirigati, F. / Liu, J. / Korn, F. / Wu, Y. / Yu, C. / Zhang H.: Knowledge Exploration using Tables on the Web. PVLDB 10, 2016

[DMP12] Dalvi, N. / Machanavajjhala, A. / Pang, B.: An analysis of structured data on the web. Proceedings VLDB Endowment 5(7), März 2012, S. 680–691

[DSS13] Dong, X. L. / Saha, B. / Srivastava, D.: Less is more: selecting sources wisely for integration. In: Proceedings of the 39th international conference on Very Large Data Bases, VLDB Endowment 2013, S. 37–48

[EBH15] Eberius, J. / Braunschweig, K. / Hentsch, M. / Thiele, M. / Ahmadov, A. / Lehner, W.: Building the Dresden Web Table Corpus: A Classification Approach. In: 2nd IEEE/ACM International Symposium on Big Data Computing BDC, 2015

[ETB15] Eberius, J. / Thiele, M. / Braunschweig, K. / Lehner, W.: Top-k entity augmentation using consistent set covering, Proceedings of the 27th International Conference on Scientific and Statistical Database Management, La Jolla, California 2015

[FuT18]https://support.google.com/fusiontables

[HRO06] Halevy, A. / Rajaraman, A. / Ordille, J.: Data integration: The teenage years. In: Proceedings of the 32nd International Conference on Very Large Data Bases, VLDB Endowment 2006, S. 9–1**6**

[Kar72] Karp, R.M.: Reducibility among Combinatorial Problems. Complexity of Computer Computations, 1972

[LDL13] Li, X. / Dong, X. L. / Lyons, K. / Meng, W. / Srivastava, D.: Truth finding on the deep web: is the problem solved? In: Proceedings of the 39th international conference on Very Large Data Bases, VLDB Endowment 2013, S. 97–108

[Loh14] Lohr, S.: For Big-Data Scientists, 'Janitor Work' Is Key Hurdle to Insights. In: The New York Time 2014

[LRR15] Lehmberg, O. / Ritze, D. / Ristoski, P. / Meusel, R. / Paulheim, H. / Bizer, C.: The mannheim search join engine. Web

Semantics: Science, Services and Agents on the World Wide Web, 20**15**

[MBC15] Mohanty, H. / Bhuyan, P. / Chenthati, D.: Big Data: A Primer. Springer India 2015

[OLe14] O'Leary, D. E.: Embedding AI and crowdsourcing in the big data lake. In: IEEE Intelligent Systems 29(5), 2014, S. 70–73

[PiS12] Pimplikar, R. / Sarawagi, S.: Answering table queries on the web using column keywords. In: Proceedings of the 36th International Conference on Very Large Databases (VLDB), 2012

[SaC14] Sarawagi, S. / Chakrabarti, S.: Open-domain quantity queries on web tables: Annotation, response, and consensus models. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York 2014, S. 711–720

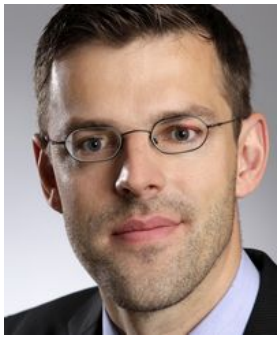
[SKW07] Suchanek, F. M. / Kasneci, G. / Weikum, G.: Yago: A core of semantic knowledge. In: Proceedings of the 16th International Conference on World Wide Web, New York 2007, S. 697–706

[Tab18] https://research.google.com/tables

[WDC15]http://webdatacommons.org/webtables/2015/EnglishStatistics.html

[YGC12] Yakout, M. / Ganjam, K. / Chakrabarti, K. / Chaudhuri, S.: Infogather: entity augmentation and attribute discovery by holistic matching with web tables. In: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, New York 2012, S. 97–108

[ZaC13] Zhang, M. / Chakrabarti, K.: Infogather+: semantic matching and annotation of numeric and time-varying attributes in web tables. In: Proceedings of the 2013 international conference on Management of data, New York 2013, S. 145–156



Dr.-Ing. Maik Thiele

ist wissenschaftlicher Mitarbeiter und Gruppenleiter an der Professur Datenbanken der Technischen Universität Dresden, wo er im Mai 2010 seine Promotion mit Auszeichnung abschloss. Er war Gastwissenschaftler bei der UBS in Zürich, der GfK in Nürnberg und HP Labs Palo Alto. Seine Forschungsinteressen umfassen unter anderem Data Warehousing, Maschinelles Lernen, Informationsextraktion und Datenintegration. Als Mitarbeiter im Big-Data-Kompetenzzentrum „ScaDS Dresden/Leipzig" verantwortet er das Themengebiet „Data Lifecycle Management".

E-Mail: maik.thiele(at)tu-dresden.de

Bildnachweis:

© Maik Thiele