Provenance-Based Strategies to Develop Trust in Semantic Web Applications

Xian Li[1], Timothy Lebo[1,2], and Deborah L. McGuinness[1]

[1] Rensselaer Polytechnic Institute, Troy, NY USA
[2] Air Force Research Laboratory, Information Directorate, Rome, NY USA
{lix15,lebot}@rpi.edu,
dlm@cs.rpi.edu

Abstract. Linked data and Semantic Web technologies enable people to navigate across heterogeneous sources of data thus making it easier for them to explore and develop multiple perspectives for use in making decisions and solving problems. While the Semantic Web offers benefits for developers and users, several new challenges are emerging that may negatively impact users' trust in Web-based collaborative systems.

This paper describes several use cases to illustrate potential trust issues faced by Semantic Web applications, and provides a concrete example for each using a specific system we built to investigate United States Supreme Court decision making. Provenance-based solutions are proposed to develop trust and/or minimize the distrust that is provoked by the situation. While these use cases address distinct situations, they are all described in terms of how a contradiction can arise between the user's mental model and the statements presented in the display. This commonality may be used to develop additional classes of trust-threatening use cases, and the proposed provenance-based solutions can be applied to many other Semantic Web Applications.

1 Introduction

As the amount of data available on the Web quickly increases, applications based on Semantic Web technologies are being developed to utilize information from multiple data sources. With a variety of visualization technologies, the advantages from heterogeneous sources of data are indicated by providing comprehensive views on a given problem, and gaining understanding of problems from a larger picture where different sources of data are interlinked. However, one essential growing challenge is related to diverse quality of the data. As more diverse data is used, user's distrust may increase, particularly when their prior knowledge and expectations are different from what the system appears to be presenting.

To address these emerging challenges, we propose provenance-based solutions using the Proof Markup Language (PML) [7] and apply them to representative use cases in Semantic Web Applications. The organization of the paper is as follows: in Section 2, we review related works in the area of provenance in the

D.L. McGuinness, J.R. Michaelis, and L. Moreau (Eds.): IPAW 2010, LNCS 6378, pp. 182–197, 2010.

context of the Web. In Section 3, we give a brief introduction to the system with which we developed use cases to illustrate challenges and provenance solutions. Section 4 describes the principles used to generate use cases, the main cause behind different types of distrust risks, and the provenance used to develop trust. Section 5 concludes with a discussion of future work.

2 Related Work

In their multidisciplinary survey of trust research, Artz et al. [1] distinguish between policy- and reputation-based trust paradigms. While the former focuses on "hard security" that is determined through authentication, access control, and encryption, the latter combines personal experiences and experiences of others to determine a degree of trust. They identify the common theme that "trust is a willingness to be vulnerable to the actions of another." When making decisions based on information from the web, one can be vulnerable to distributed contributions that the system incorporates and displays. For example, Zhao et al. [14] describe risks to scientific users' trust in FlyWeb, a consolidation and alignment of three independent data sources related to the fruit fly. They note that asynchronous updates of component data sources can violate expectations of scientists working with older releases.

Vulnerability, however, is not necessary to gain trust, since trust may be imparted without vulnerability. To help qualify the notion of trust in an information system, it may be viewed as proportional to the characteristics of information quality. Naumann et al. [9] describe information quality using several components and incorporated them into query planning over multiple, distributed, and heterogeneous sources. Their approach strives to plan queries that select from good data sources based on their completeness, timeliness, uniqueness, availability, financial cost, and accuracy. Their approach uses rules at the granular level of attributes and queries instead of at the class or source level and relies upon a domain-specific global schema to which all contributing data sources are cast.

Data believability is similar to information quality. Prat and Madnick [10] reinforce its definition as "the extent to which data are accepted or regarded as true, real and credible." They propose the three dimensions of trustworthiness, reasonableness, and temporarily to measure data believability using provenance-based measurements and present computational approaches to combine the components of believability. They measure *trustworthiness of the data source* at a holistic level. *Reasonableness of data* measures the extent to which a data value is likely using possibility, consistency over sources, and consistency over time. *Temporality of data* measures the extent to which the query time overlaps with the data value's validity interval.

Provenance is an important aspect of information quality and believability. To track the lineage of changes between releases of FlyWeb, Zhao et al. proposed a provenance-based solution. Sillence et al. [11] describe several provenance factors important to users in a study of non-experts using the web to investigate health topics. Factors included the ability to cross-validate across information sources, recency of information updates, and citations to original sources. Buneman et al. [3] described primary issues for data provenance in the context of the web, including obtaining provenance information, citing components of a document in another context, and ensuring integrity of citations in situations where the cited items evolve. Hartig [5] proposes a general provenance model that incorporates both data creation and data access. Fox et al. [4] address the problem of identifying validity and origin of data on the web by modeling and maintaining information sources, information dependencies, and trust structures. Using four levels of provenance ranging from strong provenance (high certainty) to weak provenance (high uncertainty), they annotated web data to create islands of certainty among the wild uncertainty and incompleteness. Miles et al. [8] present use cases for process documentation in e-science experiments. For each use case presented, they present *provenance question* as an action that can be realized by processing recorded process documentation. In our use cases, we outline how trust can be developed by answering a provenance question.

Bizer and Cyganiak [2] present WIQA, a named graph query system that permits users to select policies to qualify the sets of characteristics to which they attribute a certain equivalent level of trust. These policies are then used to filter aggregated information to a contextually-trusted subset, and explanations for why information fulfills a policy are provided. Filtering policies may also contain explanation templates, which can be used to generate natural language as well as RDF explanations about filtering decisions. Their use of templates to specify what to show in the application is analogous to our subject-centric template approach. They use the Semantic Web Publishing Language to describe the provenance and signing of the aggregated named graphs, while we are using the Proof Markup Language. They are permitting the user to specify the templates, while our application currently imposes a fixed selection policy. Their use of explanation templates are also a convenient solution for producing explanations, which is not an aspect that we have fully addressed in our current work.

Tummarello et al. [13] present Sig.ma, a semantic integration mashup API and user browser. Based on a textual search, it presents data aggregated from multiple traditional and semantic web data sources and offers a highly interactive browsing experience permitting users to inquire for the source of a particular data item, reject or hide certain sources or data values, organize the arrangement of information, and capture their current view for sharing using a variety of popular representations such as RDF, JSON, permanent URLs, and HTML snippets. Sig.ma develops trust by allowing users to express their opinions of reliability on different data sources. This very flexible infrastructure and interaction methodology is useful for free-formed, user-driven exploration of content. However, this paradigm is not amenable to application developers attempting to communicate a particular, well structured story – as with the application we are evaluating for the use cases we describe in this paper. Although the system permits re-publishing of "customized views", the view type is limited to a single subject with traditional attribute-value listings and does not extend to visual depictions that may more effectively communicate such as the maps, faceted browsers, scatter plots, bar charts, tables, and timelines that we find in the system we evaluate. Although Sig.ma preserves provenance information by tracking the URL sources of data, it is not clear that it uses a common representation that can be reused by external applications. Our provenance-based proposals for addressing distrust events reuses the Proof Markup Language that is already used across a variety of applications. Using PML also permits comprehensive capture of the data incorporation and permits easy elaboration.

3 Supreme Court: Justices and Decision Making

3.1 Application Domain

Supreme Court scholars utilize data from each case and vote to analyze judicial decision-making. Scholars have made independent efforts to collect needed information, format them for easy processing, check their accuracy, and publish and maintain them in a reusable state. The U.S. Supreme Court Database (SCDB) [12] is regarded as a core dataset that encodes many aspects of Justices' votes since 1953. The SCDB is periodically released in formats accepted by most statistical applications. However, focusing on statistical aspects of these isolated data may result in a limited set of views and analytical directions, with missed opportunities to gain different perspectives and insights on existing data.

Besides the SCDB datasets, studies on judicial decision-making have relied on additional variables such as birth, education, party identification, and appointing presidents. These personal attributes are readily available from biographical directories and data collections of other scholars in the field. Due to the large amounts of data involved, manual encoding methods are time consuming and their limited visibility minimizes the chances for others to correct mistakes. The isolation of each data source may also limit insight into aggregate relationships.

3.2 Tools and Techniques: Advantages and Challenges

Linked Data and other Semantic Web technologies were proposed to improve this situation, demonstrated by an application *Supreme Court: Justices and Decision Making* [6]. The SCDB dataset was transformed into Resource Description Framework (RDF) and connected to linked data available from DBpedia. RDF represents data in a directed graph where a single labeled edge, known as a triple, has components known as the subject, predicate, and object. The SCDB and DBPedia data sources were bridged using data from a Semantic Media Wiki to reconcile different naming of Justices. The approach has many advantages. First, linked versions of SCDB datasets can easily be connected to many other datasets, enabling multiple perspectives on Supreme Court and Justices. Second, Linked Open Data covers much more factual information about Justices' personal attributes and career histories. Third, linked data is readily accessible.

Finally, a large community maintains linked data, which can reduce the bias and errors in the information.

However, challenges rise as data from different sources are incorporated to gain understanding, make decisions, and solve problems. The quality of the data from heterogeneous sources determines to what extent the data could be trusted and utilized. For certain information shown in the system, it is important to identify its quality such as the source of this information, author of the data, the update time, reliability and trustworthiness. The representative use cases described below address the most common scenarios in which users of *Supreme Court: Justices and Decision Making* encountered distrust.

4 Developing Trust through Provenance

When the user, based on their background knowledge and current context, identies either a surprise or a direct contradiction with the content of an information system, any part of the system is susceptible to blame. We refer to this situation as a *distrust event*. To develop trust, the system's role is to respond by identifying the sources and the processing leading to a particular conclusion. This is done by accepting a description of the issue contradiction and providing a subset of provenance to show the cause of the concern. We assume supplemental user interface elements that accept the user's concern and casts it into the appropriate URIs of an RDF subject and property.

Table 1 illustrates a simple taxonomy for the use cases described in this paper. The first dimension distinguishes among situations where the user believes that the system is either incorrect or has omitted some content. The second dimension distinguishes among the primary causes of the distrust event, whether it be the linked data, the application incorporating the linked data, or the application user. The use cases are further distinguished by three types of provenance that can be used to develop trust when faced with the distrust event. Each type of provenance is represented using the Proof Markup Language and is described further in the following sections with the use cases that they address. The composition of the three types of provenance represents the entire data flow from initial gathering of subjects, through multi-source incorporation as well as third-party APIs providing user interface components.

- **Provenance of subject scope** describes the actions taken by the application to determine the subjects that should be investigated. This includes a query, the web service providing a query response, and the web service parameters. The subject scope query may exist fully-parameterized or may be parameterized using user input.
- **Provenance of subject-centric queries** describes the actions taken by the application to gather information about the subjects within the available data sources. This includes the Uniform Resource Identifier for the in-scope subject, a query template parameterized for the subject, the web service providing a query response, and the web service parameters.

Table 1. Classification of five use cases that provoke distrust events, based on a simple taxonomy of the user response to the system and the primary cause of the situation.

	Linked Data is primary cause	Application is primary cause	User is primary cause
User believes that the System is Incorrect	**1**	**4**	**2,3**
User believes that the System Omitted Content		**5**	

- **Provenance of user interface invocation** describes the actions taken by the application to provide user interface elements using third-party APIs. This includes the API used and the subset of query results transferred.

4.1 Provenance of Subject-Centric Queries

The provenance solution proposed here is to show how answers were obtained for queries about specific subjects. It develops trust by displaying the source of the data and the steps taken to construct the query. The majority of use cases are addressed by this provenance type.

Use Case 1: User Belives That the System Is Incorrect, Incorrect Data

How distrust is generated. User trust is at risk when the system exhibits objective content that the user believes to be incorrect. Although it is the system's action to incorporate the data, the incorrectness of linked data is the primary cause. If the system attempts to display sources for its content along with the content, it may be difficult to ensure complete, granular coverage. The third factor leading to this event is an appropriate level of user background knowledge. Although common knowledge may lead to this potential conflict, greater user expertise will increase the likelihood of this type of distrust event. A conclusive resolution can be achieved because the conflict involves objective information. A corollary to this use case is where the user is not certain about the incorrectness, but is merely questioning his own interpretation due to unfamilarity with the content or the display design. Another corollary occurs when the application selects values of properties from multiple sources to find that they contradict.

How trust can be developed. Because the system relies upon external linked data sources, the objectively incorrect information can be traced to the source to identify whether it is the source that is incorrect or the application reads the source incorrectly. When the user inquires about the incorrect data, the subject's URI and one of its properties are used to search the provenance for the queries that were constructed for the subject using query templates that contain the property. It is important to note that the subject and property characterizing the distrust event need not be of the same RDF triple; they need only to co-occur within the provenance of an instantiated subject-centric template. Web services that responded to these queries would then be listed for the user as the source of contention. This strategy insulates the non-offending sources from blame and localizes the distrust to the appropriate linked data components. Higher granularities of source tracing would allow refined localization for offending sources. In the case where the same properties are selected from multiple sources, multiple subject/property pairs would be used to identify the appropriate provenance fragments for each.

Linked data can be used in a variety of ways. Dereferencable URIs may be crawled, and they may also be aggregated, indexed, and queried using SPARQL endpoints. Each technique offers benefits and tradeoffs. For lightweight clients accessing small portions of a large, curated dataset, use of a query endpoint is a good alternative. The application we evaluated used this approach, and the subject-centered provenance was identified to address its needs. Since the subject-centered provenance describes query construction and execution, the more straight forward "query" of dereferencing a URI could also be modeled in this fashion. Further, these two variants of subject-centered provenance are not mutually exclusive. A single application could perform both types depending on its data incorporation objectives.

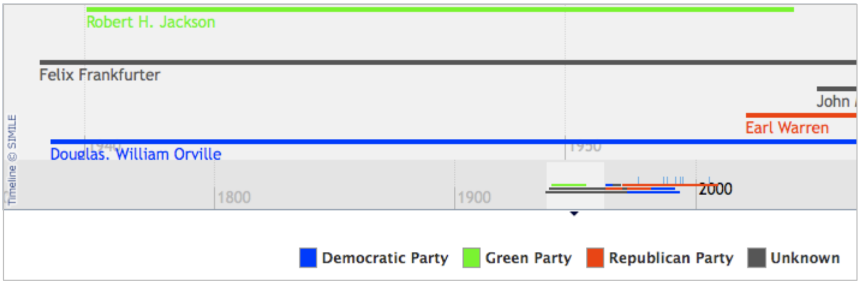


Fig. 1. User believes that no President was a member of the Green Party, and thus could not have nominated Robert Jackson to the Supreme Court of the United States

An example. As reproduced in Figure 1, the system reported that Robert H. Jackson was nominated by a President that was a member of the Green Party[1]. Only a moderate amount of common knowledge is needed to recognize that this nomination is impossible, since no President was a member of the Green Party. The commonality of this knowledge varies with the nationality and education level of the audience. Although the system is exhibiting a reasonable portrayal of the data by using the color green, this was done to handle the future possibility of the Green Party – the linked data is the primary cause of the distrust event.

Figures 2 and 3 show the structure of the provenance for query creation and execution, respectively. These form the subject-centric provenance that describes

[1] This use case assumes that the user correctly interprets the displayed content. The following use case describes a distrust event where the user misinterprets the display.

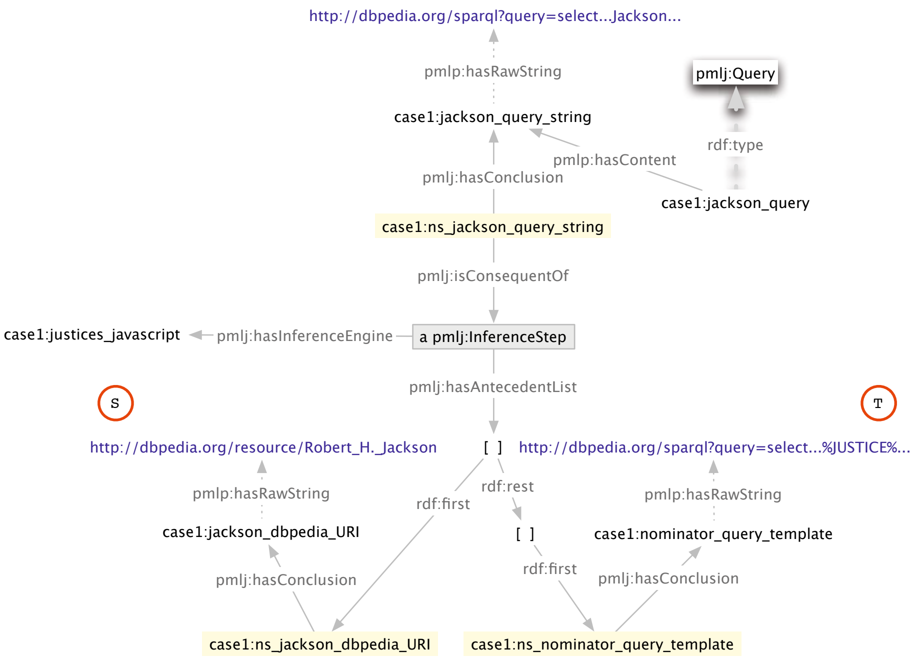

http://dbpedia.org/sparql?query=select...Jackson...
pmlp:hasRawString
pmlj:Query
case1:jackson_query_string
rdf:type
pmlp:hasContent
pmlj:hasConclusion
case1:jackson_query
case1:ns_jackson_query_string
pmlj:isConsequentOf
case1:justices_javascript
pmlj:hasInferenceEngine
a pmlj:InferenceStep
pmlj:hasAntecedentList
S
T
http://dbpedia.org/resource/Robert_H._Jackson
[ ]
http://dbpedia.org/sparql?query=select...%JUSTICE%...
pmlp:hasRawString
rdf:rest
pmlp:hasRawString
rdf:first
case1:jackson_dbpedia_URI
[ ]
case1:nominator_query_template
rdf:first
pmlj:hasConclusion
pmlj:hasConclusion
case1:ns_jackson_dbpedia_URI
case1:ns_nominator_query_template

Fig. 2. Query creation half of the subject-centric provenance for the Green Party claim. Literal values are in blue font, `pmlj:NodeSets` have a yellow background, and `pmlj:InferenceSteps` have a gray background. The red *S* highlights the subject for which descriptions were gathered using the template highlighted by the red *T*, which contains the property that characterizes the Green Party distrust event. These are connected to the data source through the query `case1:ns_jackson_sparql_results`.

the cause of the Green Party claim. The URI for Jackson and the involved property "party" are used to identify the data source http://dbpedia.org/sparql. A `pmlj:NodeSet` augments the constructed query string `pmlp:Information` with an `pmlj:InferenceStep` to justify how the query was constructed. The `pmlj:InferenceStep` cites the application code as the `pmlj:InferenceEngine` and the subject and query template it used as input. The query string is also the content of a `pmlj:Query`, which is connected (in Figure 3) to the query result "Green" by a `pmlj:NodeSet case1:ns_jackson_sparql_results`. The `pmlj:InferenceStep` identifies the SPARQL endpoint proxy and the parameters used as a cause for the query result. The data source of the "Green Party" is found at the granularity of SPARQL query endpoint[2].

[2] Note that while we provide a portion of the detailed PML encoding, this is not intended to be displayed to the end user. Instead the fine grained encoding provides enough information for GUI application developers to provide details of the sources, query formation, and other reasoning on demand to end users in an appropriate format and context.

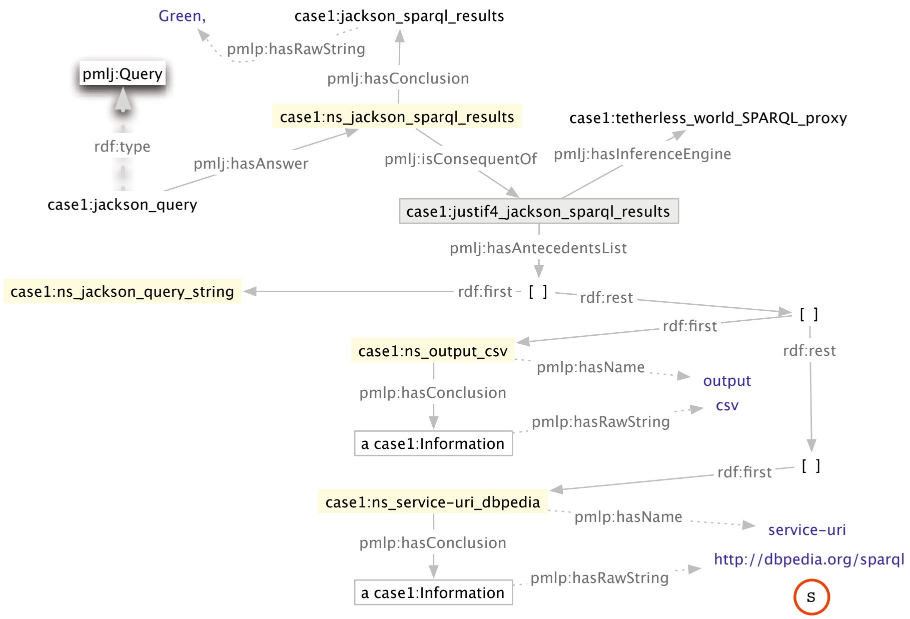


Fig. 3. Query execution half of the subject-centric provenance for the Green Party claim. Note that `case1:jackson_query` and `case1:ns_jackson_query_string` in this Figure are the same rdfs:Resources as shown in Figure 2. The red *S* highlights the data source responsible for descriptions involving the subject and property of the distrust event.

Use Case 2: User Doubts That the System Is Correct, User Misinterpretation

How distrust is generated. User trust is at risk when the user misinterprets content exhibited by the system. The system's role in this misinterpretation may contribute to this situation with varying degrees, depending upon the quality of its design. This situation occurs when either objective or subjective information is presented, where prior knowledge is correct, and linked data is correct. Both sides of this contradiction are based on factual information and the contradiction happens due to users misinterpretation of the data. This use case can apply in situations when the source data is correct or when it is incorrect.

How trust can be developed. Distinctions between objectivity and subjectivity and between incorrect or misinterpreted data rest with the user's perspective. In each case, the response to the distrust event is initiated by citing the subject and property of the data for which the user has a concern. In this way, the variety of distrust events are handled in a uniform manner. Because the system did not query for and did not receive this objectively incorrect information from external data sources, the system cannot blame the linked data. When the subject and

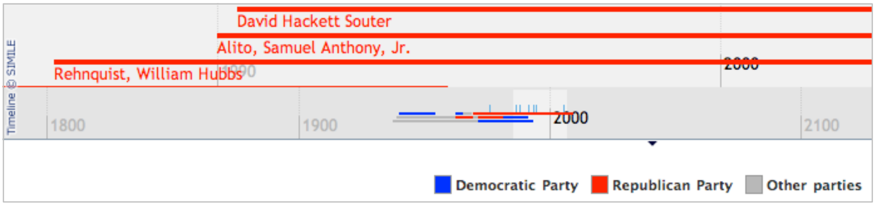


Fig. 4. The user misinterprets the content presented, thinking that the system is claiming David Hacket Souter was a Republican

property are questioned by the user, the provenance can be searched for the queries that were created using the subject's URI. Showing all of these using an interface appropriate for the user and context will inform the user that the property was not gathered by the application. Additionally, all query results may be searched for the property and value in question, and these can be shown to demonstrate that the subject is not included in these results. The graph patterns of SPARQL queries and the bindings structure of the results may present a challenge for a straight-forward solution.

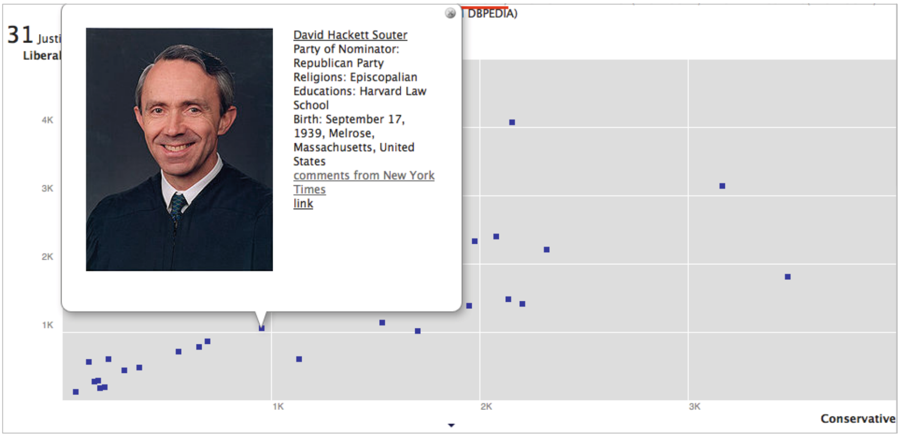
An example. As reproduced in Figure 4, a user knows Justice David Souter is a Democrat while what she perceives from the interface is that Souter is Republican. The application is referring to the party of Souter's nominator, not of Souter himself. The system can develop trust by showing all queries involving Souter, none of which will involve his political party. The system could also show all query results involving "Republican" and show that Souter is not involved in these results. The same subject-centric provenance components are used as described in Use Case 1. But because no specific property is indicated in this scenario, the provenance process will search for all queries given Souter's URI, identifying a variety of templates where the actual queries come from. Alternatively, the provenance process could be searched for all query results mentioning "Republican," showing the queries that went into the results.

Use Case 3: User Is Certain That the System Is Incorrect, Subjective Content

How distrust is generated. This type of contradiction happens when the analytical results showed by the linked data contradicts with users subjective opinions instead of facts. The application is a source cause because it is aggregating data with subjective content. Linked data is a cause because the source is contributing data with subjective content. The user is a cause in this situation if their views disagree with the subjective content exhibited. Any subjective claims must be supported, but will not be conclusive as in the previous use cases that addressed objective content. This use case involves more complicated and ambiguous issues and can not be resolved conclusively because individuals disagree on interpretations of a set of facts.

How trust can be developed. A claim on its own is not a fact; it must be supported. The source of the claim can be provided using provenance of subject-centric queries, but instead of assigning blame of the source, the system is deferring the credibility of the claim. More elaborate solutions would incorporate how the analytical results were derived using source of data, computation mechanisms, and the people invoking the analysis.

An example. In the SCDB, each vote that a Justice makes is classified as reflecting a conservative or liberal position. The total conservative votes can be compared to the total liberal votes to quantify a Justice's stance. Figure 5 reflects the neutrality of David Hacket Souter because he voted approximately equally for conservative and liberal decisions. This contrasts with the general stereotype that he has a strong liberal stance[3]. Given Souter's URI and the property "decisionDirection," subject-centric provenance can be used to identify SCDB as the source of the vote tally for conservative and liberal votes.



David Hackett Souter
Party of Nominator: Republican Party
Religions: Episcopalian
Educations: Harvard Law School
Birth: September 17, 1939, Melrose, Massachusetts, United States
comments from New York Times
link

Fig. 5. A user with conservative political views considering David Souter to be a liberal disagrees with a claim that he served as a moderate Justice

4.2 Use Cases Addressed by Provenance of User Interface Invocation

Solutions for use cases in this category are based on an extension to the subject-centered query provenance described in the previous section. The content presented by a third-party user interface or visualization API is represented with an instance of `pmlp:Information`, and instances of `pmlj:NodeSet` provide justifications enumerating the content's antecedent query results. These query results

[3] http://topics.nytimes.com/top/reference/timestopics/people/s/david_h_souter/index.html

are part of the subject-centered provenance already described. The use case addressed by this provenance solution involves the Exhibit framework. The incorporating web application is insulated from distrust when illustrating that it obtained correct data and transferred requests to the third-party API.

Use Case 4: User Is Certain That the System Is Incorrect, Rendering Distortion

How distrust is generated. Contradictions at this level may be caused by the inconsistency between two content elements depicting the same data property. Distrust in the system is generated because of its self-contradictory exhibition. Some user background knowledge may be required to fulfill the contradiction. The objective nature of the data can lead to a conclusive resolution of this distrust event.

How trust can be developed. Instead of reporting the source of incorrect or missing data as in the previous solutions, trust is developed by illustrating that correct data was incorporated and provided to a supporting API. Any distortions of content become the responsibility of the API and not of the web application. If the source data elements for the two contradictory content elements are the same, it is clearly a rendering distortion. However, if the same data property was provided by separate sources and the content is correctly portrayed, then the conflicting data sources should be shown and the third-party user interface APIs can be absolved.

An example. As reproduced in Figure 7, the system exhibited self-contradictory nativity information for David Hackett Souter, since the map is pointing to Quebec and the information box lists Melrose, Massachusetts. Users background knowledge about the uniqueness of a birthplace led to a contradiction. It is possible that “Quebec” came from SCDB and “Melrose, Massachusetts” came from DBpedia, but SCDB does not describe nativity. In this case, the same value “Melrose, Massachusetts” was provided to both display APIs. One simply displayed the text, while the other obtained incorrect latitude and longitude values for the string. Given Souter’s URI and the property “Birthplace,” the appropriate subject-centered queries could be found. Unlike in the previous section, conclusions derived from these results are found to identify the `pmlj:InferenceEngine` that used the query results to produce the visual display.

4.3 Use Case Addressed by Provenance of Subject Scope

Use Case 5: User Believes That the System Omitted Content, System Scoped

How distrust is generated. If the system is showing instances of a certain type, the user may reasonably expect all instances to be shown. The system is the primary cause of this distrust event, since it is scoped to show only certain data

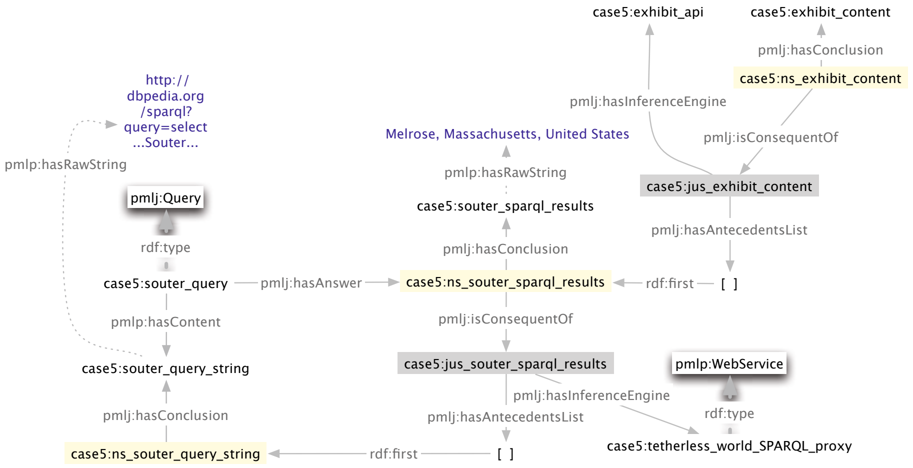
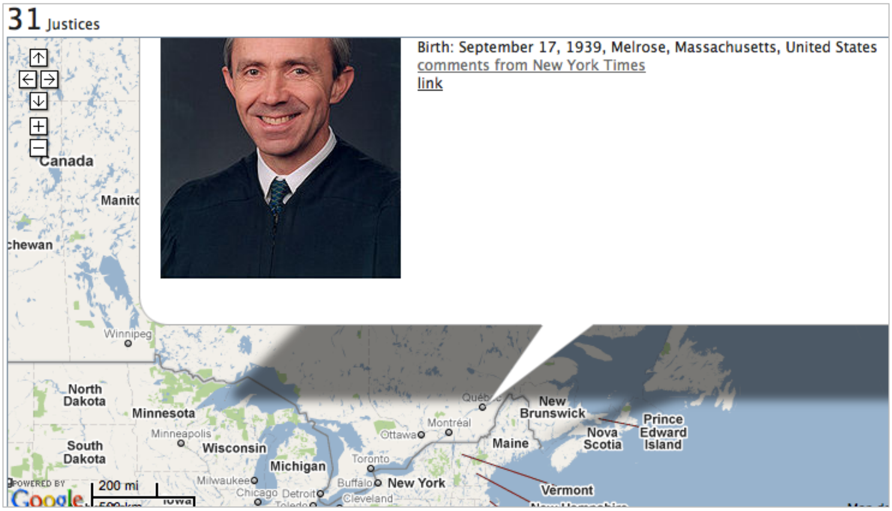


Fig. 6. With *provenance of user interface invocation*, subject-centered query provenance is extended to capture subsequent uses of the query result. An instance of `pmlp:Information` represents all content displayed by a third-party user interface API, and a `pmlj:NodeSet` provides justification for the information by citing that Exhibit was given the query results `case5:souter_sparql_results`


31 Justices

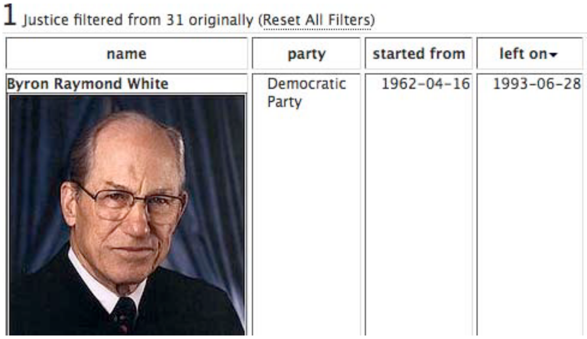
Birth: September 17, 1939, Melrose, Massachusetts, United States
comments from New York Times
link

Fig. 7. User is certain that a person could not be born in two places. This incorrect content could be caused by several situations, each with varying degrees of blame for linked data and the application.

from certain sources. The linked data is not a cause because the system did not request the data. The user is a cause because of reasonable expectations for a comprehensive view. This use case differs from the previous because the system is subject-scoping, and thus not intending to show omitted data. A corollary to this use case is where the user not only knows an entity exists, but also knows it is described in the cited sources. An additional corollary occurs when the primary cause shifts from the applications subject-scoping to the lack of requested linked data. This would be classified into the first empty box[4] of the taxonomy shown in Table 1.

How trust can be developed. While principle data sources provide the entities that will be displayed, augmenting data sources provide supplemental properties. Searching the subject-centered query provenance for a URI from an augmenting source will not succeed because queries were only performed for URIs from primary sources. Without an appropriate subset of provenance, showing the overall flow of queries can distinguish among primary and secondary sources. Multiple subject-centered query provenance segments can be composed to capture the application's chaining of queries.

An example. As reproduced in Figure 8, search results for "white" do not show Edward Douglas White, whom the user knows was a Justice. The application is using SCDB as its primary data source and augmenting these descriptions using DBPedia. Since Edward Douglas White served before the period that SCDB describes (1953-2009), his descriptions are not available. Showing that the initial queries were constructed from entities in SCDB and not DBpedia will provide this explanation. DBpedia can be highlighted as containing descriptions of Edward Douglas White to indicate that if the system incorporated DBpedia as a primary source, it would have included it as content.



1 Justice filtered from 31 originally (Reset All Filters)

name	party	started from	left on▾
Byron Raymond White	Democratic Party	1962-04-16	1993-06-28

Fig. 8. The user knows Edward Douglas White was a Justice of the Supreme Court, but search results for "white" showing only Byron Raymond White leads to a contradiction

[4] The remaining empty box in the taxonomy would contain use cases where the user was the cause of the system's omission of content, where search terms or other data-filtering user elements are employed.

5 Conclusions

Our accumulation and analysis of use cases for an existing linked data application has established provenance of the subject-centric query as a primary type that can be used to address a variety of distrust events and help develop user trust in applications incorporating linked data by insulating non-offending sources from blame and localizing the distrust to the appropriate components. The two remaining provenance types reinforce its importance by demonstrating its basis for extension to address still other types of distrust events. We identified two types of user response that provoke a distrust event, three types of their primary cause, and a contradiction-based technique for identifying and developing distrust trust use cases. We propose these factors be considered when developing linked data applications and services.

We plan to use these use cases to guide implementation of provenance within the current implementation of *Supreme Court: Justices and Decision Making*. This will enable an evaluation of our proposed methodology, which should include the characteristics for information quality and believability. We plan to identify application-independent functionality that should be part of a provenance-enabled web application framework. Saving intermediate results that can be retrieved in response to distrust events will be an important aspect. Development of additional uses cases for the application may lead to a more sophisticated taxonomy and general understanding of distrust events, and approaches for accepting the user's concern to initiate provenance search and explanation will also be needed.

References

1. Artz, D., Gil, Y.: A survey of trust in computer science and the semantic web. Web Semantics: Science, Services and Agents on the World Wide Web 5(2), 58–71 (2007)
2. Bizer, C., Cyganiak, R.: Quality-driven information filtering using the wiqa policy framework. Web Semantics: Science, Services and Agents on the World Wide Web, The Semantic Web and Policy 7(1), 1–10 (2009)
3. Buneman, P., Khanna, S., Tan, W.-C.: Data provenance: Some basic issues. In: Kapoor, S., Prasad, S. (eds.) FST TCS 2000. LNCS, vol. 1974, pp. 87–93. Springer, Heidelberg (2000)
4. Fox, M., Huang, J.: Knowledge provenance in enterprise information. International Journal of Production Research 43(20), 4471–4492 (2005)
5. Hartig, O.: Provenance information in the web of data. In: Proceedings of the 2nd Workshop on Linked Data on the Web, LDOW 2009 (2009)
6. Li, X., Ding, L., Hendler, J.A.: Study supreme court justice decision making with linked data. Technical report, Rensselaer Polytechnic Institute (2010)
7. McGuinness, D., Ding, L., da Silva, P., Chang, C.: Pml 2: A modular explanation interlingua. In: Proceedings of AAAI, vol. 7 (2007)
8. Miles, S., Groth, P., Branco, M., Moreau, L.: The requirements of using provenance in e-science experiments. Journal of Grid Computing 5(1), 1–25 (2007)
9. Naumann, F., Leser, U., Freytag, J.-C.: Quality-driven integration of heterogeneous information systems. In: VLDB Conference, pp. 447–458 (1999)

10. Prat, N., Madnick, S.: Measuring data believability: A provenance approach. In: HICSS 2008: Proceedings of the Proceedings of the 41st Annual Hawaii International Conference on System Sciences, Washington, DC, USA, p. 393. IEEE Computer Society, Los Alamitos (2008)
11. Sillence, E., Briggs, P., Fishwick, L., Harris, P.: Trust and mistrust of online health sites. In: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 663–670. ACM, New York (2004)
12. Spaeth, H., Segal, J.: US Supreme Court Judicial Data Base: Providing New Insights into the Court, The. Judicature 83, 228 (1999)
13. Tummarello, G., Cyganiak, R., Catasta, M., Danielczyk, S., Delbru, R., Decker, S.: Sig.ma: live views on the web of data. In: WWW 2010: Proceedings of the 19th international conference on World wide web, pp. 1301–1304. ACM, New York (2010)
14. Zhao, J., Klyne, G., Shotton, D.: Provenance and linked data in biological data webs. In: Proceedings of the 17th International World Wide Web Conference WWW2008 (Workshop: Linked Data on the Web LDOW 2008), vol. 22 (April 2008) Citeseer