

# Using SPIN to Formalise Accounting Regulations on the Semantic Web

Dennis Spohr<sup>†</sup>, Philipp Cimiano<sup>†</sup>, John McCrae<sup>†</sup> and Seán O’Riain<sup>\*</sup>

<sup>†</sup>Cognitive Interaction Technology Centre of Excellence  
Semantic Computing Group, University of Bielefeld, Germany  
{dspohr,cimiano,jmccrae}@cit-ec.uni-bielefeld.de

<sup>\*</sup>Digital Enterprise Research Institute  
NUI Galway, Ireland  
sean.oriain@deri.org

**Abstract.** The eXtensible Business Reporting Language (XBRL) has standardised financial reporting and provide a machine-interpretable format that makes financial and business reports easier to access and consume. Leveraging XBRL with Open Linked Data for purposes such as multi-dimensional regulatory querying and investigation requires XBRL formalisation as RDF. This paper investigates the use of off-the-shelf Semantic Web technologies to formalise accounting regulations specified in XBRL jurisdictional taxonomies. Specifically the use of the SPARQL Inferencing Notation (SPIN) with RDF to represent these accounting regulations as rule constraints, not catered for in the RDF abstract model is investigated. We move beyond previous RDF to XBRL transformations and investigate how SPIN enhanced formalisation enables inferencing of financial statement facts associated with financial reporting concepts and sophisticated consistency checks, which evaluate the correctness of reported financial data with respect to the calculation requirements imposed by accounting regulation. The approach illustrated through two use cases demonstrates the use of SPIN to meet central requirements for financial data and regulatory modelling.

## 1 Introduction

Despite the proliferation of financial information available from sources such as company websites, institutions and regulating bodies there remains a lack of transparency with regard to financial information. Two areas that can contribute to enhancing transparency are the adoption of a standard data representation formalism and greater levels of interoperability with and between different financial sources.

Multiple heterogeneous formats (e.g. HTML, PDF, CSV), ensure that data usage and interpretation typically has a dependency on manual intervention with a knock on effect for accurate and timely analysis of for example, financial reports [9]. Overcoming transparency and re-use issues associated with heterogeneous formats requires that financial information providers move towards data

provision in machine-interpretable and interoperable formats. XBRL<sup>1</sup>, has been adopted by regulatory agencies for consolidated financial filings and is gaining acceptance for general business reporting. Within the U.S. the Securities and Exchange Commission (SEC) has mandated XBRL use by all financial filers by 2014<sup>2</sup>.

XBRL, an XML-based format defines financial concepts and their relations based on jurisdictional Generally Accepted Accounting Practices (GAAP'S). Relations include calculation rules for monetary concepts – for example, the value for the financial element *Assets* is calculated from the sum of *Current assets* and *Non-current assets* – in addition to other more complex business rules expressed through XBRL formula.

XBRL offers automated processing of financial reports and increased interoperability between reporting instances. Even though XBRL provides a common interoperable format its document-centric nature has been identified as an inhibitor to integration of financial information from multiple sources [6] and [1]). XBRL's abstract model also remains unclear as to how instances and concepts can be linked with other data sources. Semantic Web formalisms such as RDF with a well defined and understood abstract model has been gaining popularity for multiple data source integration using the data mash-up approach. Attempts to make XBRL interoperable with other Web based information through its transformation to RDF or OWL have gained momentum in recent years (e.g. [6] and [1]). While the approaches adequately represent financial statements contained within XBRL formatted financial reports, they do not formalise the semantics inherent in the calculation rules. As the calculation rules represent country specific regulatory instruction for financial instrument calculation, inability to adequately express them will result in a lack of conformance and regulatory checking capability against the accounting standards from which they have been reported. Formalising regulatory information especially when transferring to an alternate representation is therefore important.

In this paper, we present an approach to implementing XBRL semantics using off-the-shelf Semantic Web technologies and specifically SPIN<sup>3</sup>, ) to semantically model regulatory requirements mentioned. The resulting representation can be used to infer values for monetary concepts, and consistency check reported values without the need for customised XBRL software. The SPIN vocabulary, developed for business rule representation, can capture the intended semantics of accounting regulations in a transparent and intuitive way. SPIN adoption is supported by tools such as TopBraid Composer<sup>4</sup>), in progress standardisation efforts and an open-source Java API<sup>5</sup>.

---

<sup>1</sup> XBRL V2.1 Taxonomy Specification Recommendations <http://www.xbrl.org/SpecRecommendations/>.

<sup>2</sup> See <http://xbrl.sec.gov/>.

<sup>3</sup> Specification at <http://www.spinrdf.org/>

<sup>4</sup> See [http://topquadrant.com/products/TB\\_Suite.html](http://topquadrant.com/products/TB_Suite.html)

<sup>5</sup> See <http://www.spinrdf.org/faq.html> for more details.

After a brief introduction to XBRL, financial reporting using XBRL taxonomies and SPIN, we position our work with respect to recent efforts from García and Gil [6] and Bao et al. [1]. Section 3 presents the general translation of XBRL to RDF/OWL with particular focus on transforming calculation rules to SPIN (Section 3.2). Section 4 then illustrates how resulting representations are capable of inferring values for reporting concepts, and checking that the values of reported concepts conform to the rules as defined in the respective accounting standard. In Section 5, we relate this SPARQL-based approach to generally available rule-based approaches. Finally Section 6 outlines how emergent XBRL-related developments integrate with the approach presented here.

## 2 Background and related work

This section introduces the main aspects of XBRL and financial reporting as relevant to this work (Section 2.1), as well as the basic features of SPIN (Section 2.3). Section 2.2 discusses related work, firstly with respect to transforming XBRL data to its RDF equivalent, and secondly regarding the use of SPARQL for business-related modelling issues.

### 2.1 Financial reporting in XBRL

XBRL is an XML-based formalism which aims to replace dependency on proprietary formats usage in financial reports preparation and compilation [7]. XBRL targets increased interoperability across different companies, thereby reducing the manual effort required to create and consume financial information. At its core is the notion of *taxonomies* and *instance documents* (see Fig. 1). The instance document represents the financial report <sup>6</sup>, stating financial instrument data facts such as its monetary value and units. Each fact is linked to a reporting context, which additionally specifies an entity – commonly the company which issued the report – as well as the period to which the fact applies. In XBRL these are termed dimensions. The example (1), taken from the 2009 SAP annual report, specifies that SAP had *Cash and cash equivalents* of €1.88 billion on December 31, 2009.

```
(1) <context id="FYp0Qp0e">
  <entity>
    <identifier scheme="http://www.sec.gov/CIK">0000943042</identifier>
  </entity>
  <period>
    <instant>2009-12-31</instant>
  </period>
</context>
...
<ifrs:CashAndCashEquivalents contextRef="FYp0Qp0e" decimals="-6"
  unitRef="EUR">1884000000</ifrs:CashAndCashEquivalents>
```

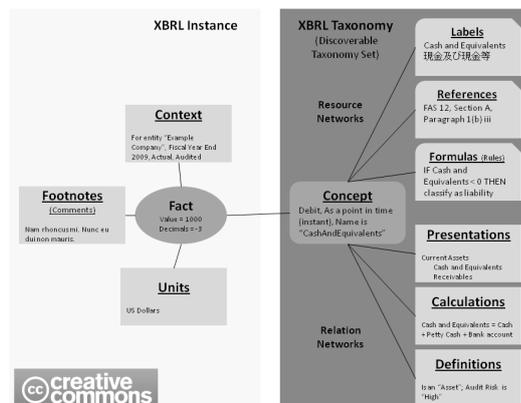


Fig. 1: High-level model of XBRL by Charles Hoffman<sup>7</sup>

As shown in (1), values in an instance document refer to concepts defined in XBRL taxonomies, which specify, for example, that *Cash and cash equivalents* is a monetary concept with balance debit, which is measured at a point in time – as opposed to over a duration of time. In addition to adopting internationally standardised taxonomies like the International Financial Reporting Standards (IFRS), companies can also provide their own taxonomy extensions, in instances where they need to report a value for a concept which is not covered by the standard. For example, the concept *Software revenue* is not found in the 2009 IFRS taxonomy, but is provided as a custom taxonomy extension by SAP, with the facts in the instance document. Each concept in a taxonomy is linked to a set of XLink *linkbases*<sup>8</sup> (called *resource* and *relation networks* in the figure). These specify labels for the concepts, as well as other information such as, how the values of the concepts should be displayed in different statement types. For example, the International Financial Accounting Standard<sup>9</sup> “*Statement of financial position, current/non-current*” specifies that the concept “*Assets*” should be displayed above “*Non-current assets*” and “*Current assets*” in a consolidated filing, whereas the financial instrument “*Statement of financial position, order of liquidity*” places “*Assets*” above “*Property, plant and equipment*”, “*Investment property*”. To express such relationships, XBRL uses XML Linking Language (XLink:<sup>10</sup>) arcs and extended links, which can be used to group any number of

<sup>6</sup> Note: XBRL has also been used by the Global Reporting Initiative <https://www.globalreporting.org/> to report on sustainability issues.

<sup>7</sup> <http://xbrl.squarespace.com>

<sup>8</sup> e.g. the U.S. 2009 GAAP taxonomy contains 450 linkbases.

<sup>9</sup> <http://www.ifrs.org/XBRL/IFRS+Taxonomy/IFRS+Taxonomy+2011/IFRS+Taxonomy+2011.htm>.

<sup>10</sup> <http://www.w3.org/TR/xlink/>.

arcs and link them to other resources For the case just described, the XBRL presentation linkbase specifies how the concepts are linked using *parent-child* arcs, and that a set of presentation arcs are associated with a particular accounting standard using an extended link role.

For the purpose of our investigations we focus on the XBRL calculation linkbases, which defines how concept values are calculated as defined by specific accounting standards and general business rules (see sections 3.2 and 6. Example (2) shows the XBRL representation of a calculation arc taken from the calculation linkbase of the IFRS taxonomy. The XBRL formula linkbase is outside the research scope and left for future work considerations.

```
(2) <loc xlink:type="locator" xlink:label="ifrs_CashAndCashEquivalents"
      xlink:href="http://xbrl.iasb.org/taxonomy/2009-04-01/
                ifrs-cor_2009-04-01.xsd#ifrs_CashAndCashEquivalents" />
    <calculationArc xlink:type="arc"
      xlink:arcrole="http://www.xbrl.org/2003/arcrole/summation-item"
      xlink:from="ifrs_CurrentAssets"
      xlink:to="ifrs_CashAndCashEquivalents" order="1" weight="1" />
```

The arc states that the concept “*CurrentAssets*” is linked to “*CashAndCashEquivalents*” through a “*summation-item*” relation. The arc weight of 1 indicates that the concept values are added and a weight of -1 that the values are subtracted. Currently XBRL calculation links only allow for the summation of items. Clearly Semantic Web formalisms offer a far more compact and intuitive representation for expressing such statements. To that end we next discuss related approaches that convert XBRL data to its RDF equivalent.

## 2.2 Related work

*Transforming XBRL to Semantic Web formalisms.* Bao et al. [1] have presented the most recent approach to transforming XBRL data to a Semantic Web standard. They present an OWL-based model that “faithfully preserves the implicit semantics in XBRL” (see [1], p. 144). In fact, however, due to their focus on making the semantics of linkbase arcs explicit, their approach omits a considerable amount of the semantics described in XBRL documents. One of these concerns the use of extended link roles, which Bao et al. [1] refer to as “non-semantic”. These link roles (Section 2.1) limit the scope of assertions in an XBRL linkbase e.g. to a particular type of statement. Such information seems to be lost in the representation of Bao et al., making their arcs become globally applicable.

In addition to this, their strategy for representing linkbase arcs is based on the assumption that the intended interpretation of arcs holds between instances of the respective concepts linked by a particular arc. The XBRL Specification does not note this as being the intended interpretation, but instead states that arcs relate “one XBRL concept to one other XBRL concept”<sup>11</sup>. While the assumption seem reasonable for concrete-numeric-concepts, abstract concepts by

<sup>11</sup> See [http://www.xbrl.org/Specification/XBRL-RECOMMENDATION-2003-12-31+Corrected-Errata-2008-07-02.htm#\\_3.5.3.9](http://www.xbrl.org/Specification/XBRL-RECOMMENDATION-2003-12-31+Corrected-Errata-2008-07-02.htm#_3.5.3.9).

definition do not have instances. Irrespective they are still related by means of **parent-child** arcs, and it is not apparent how [1] caters for those cases.

Bao et al. are not “mechanical” in their preservation of the structural properties of XBRL while other approaches by Declerck and Krieger [3] and García and Gil [6] are. Adhering to the latter approaches we preserve relevant aspects of the structural information in XBRL, while adding further interpretation that address the shortcomings of Semantic Web vocabularies to semantically model mathematical relations contained in the XBRL calculation [8].

*Using SPARQL in the context of business information.* Fürber and Hepp [5] have presented an approach to using SPARQL in order to detect data quality problems. The use of SPIN to model consistency constraints in different scenarios is discussed and how detected inconsistencies are flagged, illustrated in TopBraid Composer. We note the overlap with our approach through the use of some of their SPIN constraints that can also be applied to XBRL data. We however further the use of SPIN in a more advanced scenario that requires performing constraint checking on data which have been inferred through iterative rule application.

### 2.3 SPARQL Inferencing Notation (SPIN)

According to its developers, the SPARQL Inferencing Notation (SPIN) “is the de-facto industry standard to represent SPARQL rules and constraints on Semantic Web models”<sup>12</sup>. SPIN has been developed out of the necessity to perform calculations on property values, a task which is largely unsupported in current Semantic Web formalisms, as well as the need for constraints checking with closed-world semantics (see Section 5).

SPIN provides the syntax to attach SPARQL queries to resources in an RDF-compliant way using RDF properties `spin:rule`, `spin:constraint`, and super-property `spin:query`. The `spin:rule` property accepts SPARQL CONSTRUCT queries as value and can be used to infer new triples on the basis of the statements in the query’s WHERE clause. A basic example is provided in (3), and the corresponding SPIN representation in Turtle syntax in (4).

- (3) `CONSTRUCT { ?this a ?c2 . }`  
`WHERE { ?c1 rdfs:subClassOf ?c2 .`  
`?this a ?c1 . }`
- (4) `[ a sp:Construct ; sp:templates ([ sp:subject spin:_this ;`  
`sp:predicate rdf:type ;`  
`sp:object _:b1 ])`  
`sp:where ([ sp:subject _:b3 ;`  
`sp:predicate rdfs:subClassOf ;`  
`sp:object _:b1`  
`] [ sp:subject spin:_this ;`  
`sp:predicate rdf:type ;`  
`sp:object _:b3 ] ) ]`

<sup>12</sup> See <http://www.spinrdf.org>.

The example, based on that available from TopBraid’s SPIN website<sup>13</sup>, formalises the semantics of `rdfs:subClassOf` and illustrated variable use. While variables in a SPARQL query are generally mapped to blank nodes, in SPIN RDF notation, the variable `?this` refers to the resource `spin:_this`. This variable, like the corresponding keyword in object-oriented programming languages, refers to an instance of the class to which the respective rule has been attached. As a result, if the rule in Example (3) was attached to `owl:Thing`, it would be applied to every instance of `owl:Thing` satisfying the statements in the WHERE clause.

The `spin:constraint` property can be used to model consistency constraints, using the SPARQL ASK queries. Where an ASK query evaluates to true, the respective instance is indicated as violating the constraint. Finally, the general property `spin:query` can be used to generally attach SPARQL queries to RDF resources, i.e. also SELECT queries. As will be shown in Section 3, we make use primarily of CONSTRUCT and ASK queries in order to capture the intended semantics of accounting regulations.

In addition to standard SPARQL operators like UNION, OPTIONAL and FILTER, SPIN supports SPARQL extensions such as the ARQ keyword LET<sup>14</sup>, which allows for value assignment to variables, as well as the possibility to define custom functions.

With SPIN having recently started standardisation activities as a W3C member submission and the fact that SPARQL is already the query language of choice in numerous Semantic Web applications – there is a solid basis for its wide-spread adoption by the Semantic Web community.

### 3 Transforming XBRL to RDF

This section discusses the conversion of XBRL to RDF, with focus on the SPIN-based representation of accounting regulations. After a brief introduction to the general underlying ideas in Section 3.1, we discuss the representation of calculation rules (3.2) and consistency constraints used (3.3) to transform the accounting regulations from XBRL to RDF.

#### 3.1 General strategy

Our approach adheres to what Bao et al. [1] refers to as a representation of the “logical model” of XBRL, which preserves structural information from the original data. The motivation for doing so is to have a representation of XBRL that is interoperable with other data on the Semantic Web and also enables inferences and consistency checking, while at the same time allowing users to query the structure itself (e.g. “what is the *Assets*” concept hierarchy in the *Statement of financial position*)?

<sup>13</sup> <http://topbraid.org/spin/owlr1-all.html>

<sup>14</sup> See e.g. <http://jena.sourceforge.net/ARQ/assignment.html>.

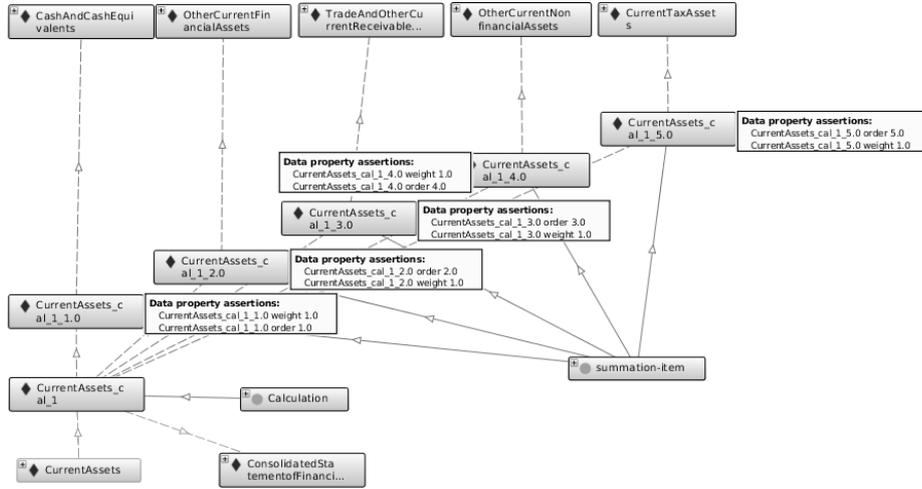


Fig. 2: Structural representation of the calculation of `ifrs:CurrentAssets` (generated with the OntoGraf Protégé 4 plug-in)

Figure 2 shows the structural representation of the calculation of `ifrs:CurrentAssets`.

The bottom left-hand corner of the figure shows an instance representing the XBRL concept which is linked to an instance of the class `Calculation`, as well as to several instances linking the individual concepts, of which all are part of the calculation. Each of these instances represents a reification of a calculation arc from the original XBRL format, and carries the weight and order attributes as values of datatype property statements.

In order to add to this structural representation and capture the intended semantics of XBRL, we make use of the meta-modelling facilities of OWL2 DL, namely *punning*. Punning allows us to refer to an entity, both a class and an individual, allowing them to be treated as distinct on the level of OWL and remain within DL. For SPARQL querying however, resources with the same URIs are interpreted as referring to the same entity. For XBRL this allows concepts defined in an XBRL taxonomy to be referred to as OWL individuals in order to express the relations that hold among them. They can also be modelled as classes, enabling values reported in XBRL instance documents to be used to *instantiate* the concepts. OWL restrictions can then be added to the concepts (e.g. that monetary concepts have integer or double values only), which are then inherited by their instances.

### 3.2 SPIN rules for calculations

In order to model the regulatory calculation of monetary concepts in a Semantic Web compliant way, SPIN rules based on the data contained in XBRL calculation linkbases were generated. Specifically calculation arcs such as those shown

in example (2), are converted into their SPIN representation (below), which represents the calculation of `ifrs:CurrentAssets`.

```
(5) CONSTRUCT { ?this xbrlrdf:calculatedValue ?cvalue . }
WHERE { ?this xbrli:contextRef ?context; xbrli:unitRef ?unit .
  ?x0 a ifrs:CurrentTaxAssets ;
      xbrli:contextRef ?context; xbrli:unitRef ?unit;
      xbrlrdf:calculatedValue ?cv0 .
  ?x1 a ifrs:OtherCurrentNonfinancialAssets ;
      xbrli:contextRef ?context; xbrli:unitRef ?unit;
      xbrlrdf:calculatedValue ?cv1 .
  ?x2 a ifrs:TradeAndOtherCurrentReceivables ;
      xbrli:contextRef ?context; xbrli:unitRef ?unit;
      xbrlrdf:calculatedValue ?cv2 .
  ?x3 a ifrs:OtherCurrentFinancialAssets ;
      xbrli:contextRef ?context; xbrli:unitRef ?unit;
      xbrlrdf:calculatedValue ?cv3 .
  ?x4 a ifrs:CashAndCashEquivalents ;
      xbrli:contextRef ?context; xbrli:unitRef ?unit;
      xbrlrdf:calculatedValue ?cv4 .
LET (?cvalue := 1.0 * ?cv0 + 1.0 * ?cv1 + 1.0 * ?cv2 +
      1.0 * ?cv3 + 1.0 * ?cv4 ) . }
```

Each of the graph patterns in the query represents one calculation arc. References to URIs for the context and units, ensure that the values of relevant instances are only taken into account. This excludes cases where a particular value refers to different entities or different segments of the same entity, as well as cases in which values are reported for different time periods. This is a normal occurrence as financial statements generally contain figures for both the current and preceding reporting periods. Finally, the LET clause specifies how the values of the individual concept instances should be combined. For accounting rules, this is limited to summation and subtraction. XBRL provides a single arc role `summation-item` for this purpose, and uses the value of the *weight* attribute – either 1 or -1 (example (2) above) – to indicate whether the value of a particular concept should be added or subtracted. For further more complex calculations possible using SPIN we refer the reader to the SPIN vocabulary specification <sup>15</sup>.

The example illustrates how rules can make use of previously calculated values to calculate further values. This allows value calculation for composite monetary concepts (i.e. those whose values are calculated on the basis of the values of other concepts) by specifying values for atomic concepts and then applying the rules iteratively (see Section 4.1 for more details).

Moreover, it should be noted that in our RDF representation, rules are not modelled as blank nodes, but instead carry a URI. This has the benefit of enabling other instances to dereference the rules, allowing a particular calculation rule be reused across different financial reports of the same accounting standard. It also further allows attachment of additional properties to a rule and a reference to the type of financial statement to which the rule applies. Therefore, in

<sup>15</sup> <http://www.spinrdf.org>.

addition to the actual rule, the instance representing the rule in (5) is the subject of a triple relating it to the URI representing SAP's *Consolidated Statement of Financial Position* by means of `xbrrdf:roleRef`.

As mentioned, rules can be executed iteratively, making use of previously inferred values. In order to make sure that a particular calculation rule with atomic concepts (i.e. those concepts which lack regulatory calculation rules attached) can also be applied, we add the default calculation shown in (6) to atomic monetary concepts. This rule then assigns the reported value of the respective instance to `xbrrdf:calculatedValue`.

```
(6) CONSTRUCT { ?this xbrlrdf:calculatedValue ?value . }
      WHERE { ?this xbrlrdf:value ?value } .
```

### 3.3 SPIN constraints for consistency checking

On the basis of the SPIN rules presented above, each monetary concept which participates in some calculation and has reported values in the respective context is assigned a calculated value. The next step in modelling the accounting regulation is to specify that calculated values need to match reported values. As SPIN rules and constraints are applied to all instances of the class to which they have been attached, as well as to the instances of its subclasses, this can be achieved by attaching a single SPIN constraint to the top monetary concept:

```
(7) ASK WHERE { ?this xbrlrdf:value ?value ;
                 xbrlrdf:calculatedValue ?cvalue .
                FILTER (?value != ?cvalue) }
```

Additionally, SPIN constraints can be used to formalise more general constraints imposed by the XBRL specification. Below, we illustrate this using a constraint (restriction) which states that if two concepts have the same balance type (i.e. credit or debit), they can only be added to one another, not subtracted. In other words, the value of the XBRL weight attribute, which is preserved in our structural representation of XBRL, has to be positive:

```
(8) ASK WHERE { ?this xlink:from ?from ; xlink:to ?to ;
                 xbrlrdf:weight ?weight .
                 ?from a ?balance . ?to a ?balance .
                 ?balance rdfs:subClassOf xbrlrdf:BalanceType .
                 FILTER (?weight < 0) }
```

## 4 Specific use cases

The SPIN rules and constraints given in Section 3 above can be used to infer values for instances of the respective reporting concepts, as well as check their consistency. We next illustrate how these representations integrate with TopBraid Composer, taking as example the SAP 2009 annual report reported against its custom extension of the IFRS 2009 taxonomy.

#### 4.1 Inferring values of monetary reporting concepts

In order to test whether the rules explained in Section 3 actually behave as desired, we have first generated a modified version of the report such that it contained reported values for *atomic* monetary concepts only. All composite monetary concepts were assigned their values through iterative application of the SPIN calculation rules. This allowed evaluation as to whether the available information triggered the application of all rules necessary to calculate the missing information and whether the calculated figures corresponded to those reported in the original filing. Table 1 summarises the results, with the figures from the original report recorded in parentheses.

	absolute	relative
Concepts in IFRS 2009 taxonomy and SAP extension	3,021	100.00%
Regulatory calculation rules	492	100.00%
Reported monetary values	351 (482)	72.82%
Monetary concepts with reported values	129 (171)	75.44%
Inferred monetary values	458	95.02%
Monetary concepts with inferred values	167	97.66%
Monetary values inferred by default rule	343	74.89%
Monetary values inferred by regulatory rules	115	25.11%
Regulatory rules applied	42	8.54%
Total number of monetary values	458 (482)	95.02%
Total number of correct monetary values	458	100.00%

Table 1: Reported and inferred values in the modified annual report 2009 of SAP

Tuples 3 and 4 of the table detail that the modified report contains 351 reported values for 129 monetary concepts, compared to 482 and 171, respectively, from the original report. After applying the calculation rules, values are inferred for 97.66% of these 171 concepts, indicating that the modified report contains 458 values, as opposed to 482 original report values. 25.11% of the 458 inferred values are due to regulatory rules, outlining that the remaining 343 values have been inferred for atomic concepts by means of the default rule. Over all 8.54% of all the regulatory rules available in the IFRS 2009 taxonomy and the SAP 2009 extension have been applied.

The figures illustrate that for 4 of the monetary concepts no value could be inferred. Analysis revealed that this is due to values for `ifrs:BasicEarningsLossPerShare` and `ifrs:DilutedEarningsLossPerShare`, being missing from the XBRL report instance, despite being part of a composite concept. As a result, the corresponding rules could not be applied (see Section 5 for a discussion regarding the optionality of calculation arcs).

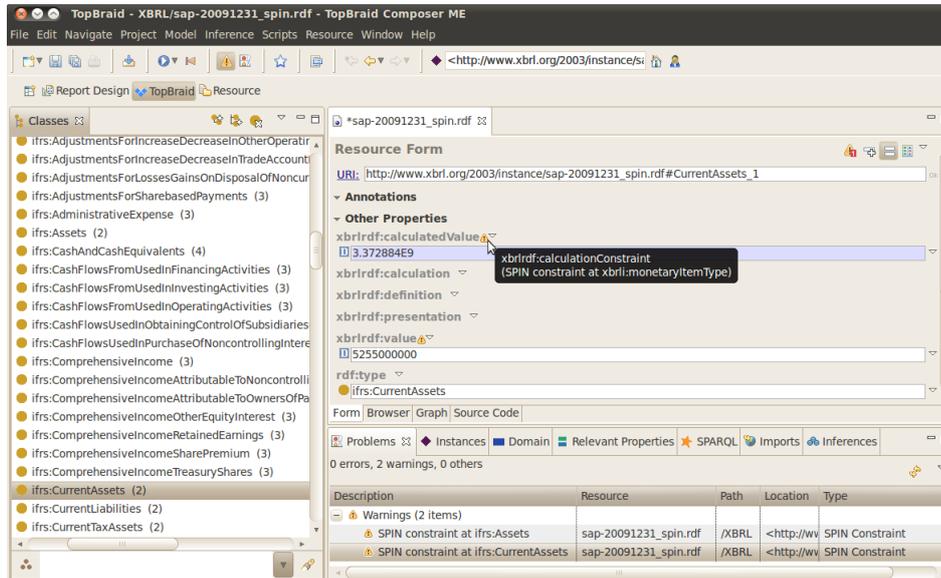


Fig. 3: View of an instance in TopBraid Composer after introducing an incorrect value

## 4.2 Consistency control of reported values

Table 1 results detail that all inferred values were correct, which in turn reflects XBRL’s contribution to overall financial reporting consistency. The results also indicate that the functionality of the SPIN constraint were not evaluated by this method. To address this situation the reported value of the atomic concept `ifrs:CashAndCashEquivalents` in the original report was changed, and the change tracked to see whether the change would be propagated along the calculation “hierarchy”, and yield inconsistent composite concepts. Result displaying an instance of `ifrs:CurrentAssets` after rule application and constraint checking in TopBraid Composer are reported in Figure 3.

The figure demonstrates that TopBraid Composer correctly flags the instance where the calculated value differs from the reported one and where inconsistency have arisen at concept that has `ifrs:CurrentAssets` as calculation component (i.e. `ifrs:Assets`).

## 5 Discussion

The previous sections detail how the SPARQL CONSTRUCT rules and ASK queries capture the semantics of the XBRL data in an intuitive and transparent way. However the XBRL does allow for calculation arcs and if so to have an assumed value of zero. This usage of default values, cannot be naturally handled

with monotonic logics such as OWL and would normally require the use of a formalism such as Reiter’s default logic [12]. A naive approach would be to develop a SPIN rules by the use of the OPTIONAL keyword available in SPARQL, for example:

```
(9) CONSTRUCT { ?this xbrlrdf:calculatedValue ?cvalue . }
  WHERE { ?x0 a ifrs:CurrentTaxAssets ;
          xbrlrdf:calculatedValue ?cv0 .
          OPTIONAL {
            ?x1 a ifrs:CashAndCashEquivalents ;
            xbrlrdf:calculatedValue ?cv1 . }
          LET(?cvalue := 1.0 * ?cv0 + 1.0 * ?cv1) }
```

If it is assumed that that rules can be applied in any order, this rule would be applied before the rule that calculates ?cv1, and a different value produced. We therefore take the position that every value should be defined or explicitly stated as not undefined. While OWL has no specific vocabulary to state that a value has not been defined we can achieve the same result using an OWL class axiom.

For example, in order to state that SAP did not report a value for `ifrs:BasicEarningsLossPerShareFromDiscontinuedOperations` in units `iso4217:EUR` year ending December 2009<sup>16</sup>, the following would need to be asserted.

```
(10) ifrs:BasicEarningsLossPerShareFromDiscontinuedOperations  $\sqsubseteq$ 
       $\neg((\exists \text{xbrli:contextRef}\{FYp0YTD\}) \sqcap (\exists \text{xbrli:unitRef}\{iso4217:EUR\}))$ 
```

With the non-existence of such an instance explicitly asserted it can be combined with the SPIN rules by providing a default rule that specifies the value of the property if it is known not to exist as follows:

```
(11) OPTIONAL { ?x4 a ifrs:CashAndCashEquivalents;
                xbrli:contextRef ?context; xbrli:unitRef ?unit;
                xbrlrdf:calculatedValue ?cv4 } .
      OPTIONAL { NOT EXISTS { ?x4 a ifrs:CashAndCashEquivalents;
                                xbrli:contextRef ?context;
                                xbrli:unitRef ?unit; } .
                LET (?cv4 := 0) . }
```

Here the assumption is made that NOT EXISTS predicate evaluates to true where it is provable that the dataset does not contain the predicate. Interesting future work could look to determine whether a default logic rule language such as RIF-SILK<sup>17</sup> could be used.

<sup>16</sup> In SAP’s XBRL instance document, the period context represented by FYp0YTD.

<sup>17</sup> See <http://silk.semwebcentral.org/RIF-SILK.html>.

## 6 Conclusion and Future Work

The paper outlines SPARQL's capability to meet complex query requirements, central to modelling financial data, and specifically their accounting regulations on the Semantic Web. The approach transforms financial reports represented in XBRL to RDF, and uses the RDF-compatible SPARQL Inferencing Notation to capture the regulatory rules expressed by the XBRL calculation linkbase. The resulting representation was evaluated against XBRL financial data, both with respect to inferring values for instances of monetary concepts and checking their consistency. Additionally the use of the representation to formalise additional constraints to address the well-formedness and high quality of the data was discussed.

The approach taken can be extended to cater for the more complex mathematical operations of the XBRL formula specification<sup>18</sup>. For example the formula specification defines that value calculations apply to instances that refer to *identical* contexts, and more generally to concept instances which are *p(arent)-equal*, *c(ontext)-equal* and *u(nit)-equal*. p-equality and u-equality have been previously shown through rule attachment to the composite class and including the reference to the unit in the rule. Alternatively c-equality could be inferred beforehand, by specifying that two contexts which share the same entity and period are linked by `owl:sameAs`. When applying the rules iteratively to a repository that is OWL-aware, the rules shown above can be applied as is.

Arguments for financial information integration include the ability to conduct financial metrics comparison [2] and querying of heterogeneous data sets to gain wider holistic insight [4]. For Linked Data driven information systems, data abstraction presents challenges for financial integration [11] and financial values comparison. Semantic Web offers a level of interoperability between data sources that would assist comparability based on the representational transformation of financial data, such as XBRL to RDF. The approach is not new, having been previously applied to areas such as investment funds analysis [8] or more recently promoted for wider financial ecosystems evolution [10]. Financial standards interoperability also faces additional challenges from different jurisdictional and regulatory rules. Use of an ontology architecture to accommodate multiple XBRL formats from the business community have only been proposed [13]. Within this context the ability of Semantic Web vocabularies to model regulatory relations contained within XBRL reporting formats will become increasingly important.

We have demonstrated SPIN's viability but others rule languages such as the Semantic Web Rule Language<sup>19</sup> suggested by [13], could also be investigated as part of a best practises recommendation for Semantic Web rule format representation.

<sup>18</sup> <http://www.xbrl.org/Specification/formula/REC-2009-06-22/formula-REC-2009-06-22.html>.

<sup>19</sup> <http://www.w3.org/Submission/SWRL/>

## 7 Acknowledgements

The work presented in this paper has been funded by the Multilingual Ontologies for Networked Knowledge (MONNET) project under EU Grant No. 248458, and Science Foundation Ireland under Grant No. SFI/08/CE/I1380 (Lion-2).

## References

1. Bao, J., Rong, G., Li, X., Ding, L.: Representing Financial Reports on the Semantic Web: A Faithful Translation from XBRL to OWL. In: *Semantic Web Rules, Lecture Notes in Computer Science*, vol. 6403, pp. 144–152. Springer (2010)
2. Debreceny, R.: *Feeding the information value chain : Deriving analytical ratios from xbrl filings to the sec* (2010)
3. Declerck, T., Krieger, H.U.: Translating XBRL into description logic. An approach using Protégé, Sesame & OWL. In: *Business Information Systems (BIS)*. pp. 455–467. Klagenfurt, Germany (2006)
4. Freitas, A., Curry, E., Oliveira, J.a.G., O’Riain, S.: Querying heterogeneous datasets on the linked data web: Challenges, approaches, and trends. *IEEE Internet Computing* 16(1), 24–33 (2012)
5. Fürber, C., Hepp, M.: Using SPARQL and SPIN for Data Quality Management on the Semantic Web. In: *Business Information Systems, Lecture Notes in Business Information Processing*, vol. 47, pp. 35–46. Springer (2010)
6. García, R., Gil, R.: Publishing XBRL as Linked Open Data. In: *Proceedings of the WWW2009 Workshop on Linked Data on the Web (LDOW)* (2009)
7. Hoffman, C., Strand, C.: *XBRL Essentials*. American Institute of Certified Public Accountants, New York (2001)
8. Lara, R., Cantador, I., Castells, P.: Xbrl taxonomies and owl ontologies for investment funds. In: *1st International Workshop on Ontologizing Industrial Standards at the 25th International Conference on Conceptual Modelling*. pp. 6–9 (2006)
9. Mueller, D., Raggett, D.: Report for the W3C Workshop on Improving Access to Financial Data on the Web (2009), <http://www.w3.org/2009/03/xbrl/report.html>
10. O’Riain, S., Curry, E., Harth, A.: Xbrl and open data for global financial ecosystems: A linked data approach. *Int J Account Inf Syst* (2012), doi:10.1016/j.accinf.2012.02.002
11. O’Riain, S., Harth, A., Curry, E.: Linked data driven information systems as an enabler for integrating financial data. *Information Systems for Global Financial Markets, Emerging Developments and Effects* pp. 239–270 (2011)
12. Reiter, R.: A logic for default reasoning. *Artificial Intelligence* 13(1), 81–132 (1980)
13. Wenger, M., Thomas, M., Jr, J.B.: An ontological approach to XBRL financial statement reporting. In: *17th Americas Conference on Information Systems*. Michigan, USA (2011)