

# Modelling Provenance of Sensor Data for Food Safety Compliance Checking

Milan Markovic<sup>1</sup>, Peter Edwards<sup>1</sup>, Martin Kollingbaum<sup>1</sup>, and Alan Rowe<sup>2</sup>

<sup>1</sup> Computing Science, University of Aberdeen,  
Aberdeen, AB24 5UA

<sup>2</sup> Rowett Institute of Nutrition and Health, University of Aberdeen,  
Aberdeen, AB21 9SB

{milan.markovic,p.edwards,m.j.kollingbaum,a.rowe}@abdn.ac.uk \*

**Abstract.** The Internet of Things (IoT) is resulting in ever greater volumes of low level sensor data. However, such data is meaningless without higher level context that describes why such data is needed and what useful information can be derived from it. Provenance records should play a pivotal role in supporting a range of automated processes acting on the data streams emerging from an IoT-enabled infrastructure. In this paper we discuss how such provenance can be modelled by extending an existing suite of provenance ontologies. Furthermore, we demonstrate how provenance abstractions can be inferred from sensor data annotated using the SSN ontology. A real-world application from food-safety compliance monitoring will be used throughout to illustrate our achievements to date, and the challenges that remain.

## 1 Introduction

The Internet of Things (IoT) concept refers to the seamless integration of physical objects, sensors and mobile devices into the information network. The IoT encompasses numerous technologies, services and standards and is seen by many as the cornerstone of the emerging ICT market. Such devices are becoming ever cheaper and easier to deploy; for example, CAO Gadgets<sup>3</sup> market a range of low power plastic tags able to measure temperature, humidity and motion. Due to their low cost, there is now significant potential for technologies such as these to be used in a range of applications that require routine data capture, condition monitoring and behavioural tracking. One such application is monitoring of food safety compliance.

In its 2015-2020 strategic plan, the UK's Food Standards Agency observes that: "It is the responsibility of people producing and supplying food to ensure

---

\* The research described here was funded by an award made by the RCUK IT as a Utility Network+ (EP/K003569/1) and the UK Food Standards Agency. We thank the owner and staff of Rye & Soda restaurant, Aberdeen for their support throughout the project.

<sup>3</sup> <http://www.wirelesstag.net/>

it is safe and what it says it is ”. Non-compliance with food storage and handling guidelines presents a significant risk to individuals and society as a whole. As an illustration, *campylobacter* is the most common cause of bacterial food poisoning in the UK, and each year is estimated to be responsible for 280,000 cases of food poisoning - at a cost of around £900M to the economy. As a result there is now considerable interest in the use of technologies such as low-cost wireless meat probes as a means to monitor cooking processes. It is now perfectly possible to imagine a future restaurant kitchen in which a suite of sensors monitor food from the moment it arrives until it is served to a customer, with automated systems alerting staff to take appropriate action when necessary, and providing management information to aid staff training and reduce wastage.

Provenance has an important role to play in documenting entities representing real physical objects, and their relationship to activities as part of a food preparation workflow. Given descriptions of workflow plans (i.e. prospective provenance documenting expected behaviour) and records of actual events (i.e. retrospective provenance documenting what really happened), provenance can help support compliance analysis - by determining whether expected food safety protocols have been followed. For example, whether chilled food has been stored within the correct temperature limits (typically 1-5°C).

While the W3C recommendation for provenance capture PROV<sup>4</sup> is suitable for modelling the retrospective part of a provenance record (i.e. workflow execution) it does not support descriptions of workflow plans [MM12]. Approaches such as D-PROV[MDB<sup>+</sup>13], ProvOne[CVLM<sup>+</sup>14], and P-PLAN [GG12] have all proposed extensions to the PROV model, to enable more detailed descriptions of such plans. These extensions typically introduced new concepts to describe workflow structures in terms of expected workflow steps and corresponding inputs and outputs. As part of our work on the SC-PROV model [Mar16] [MEC13] we expanded on these earlier efforts, by providing a means to document constraints (e.g. preconditions) that might be associated with individual steps of a workflow plan. The ability to represent such constraints is especially relevant within the food safety domain, where frameworks such as HACCP (Hazard Analysis and Critical Control Point) define process workflows in terms of critical limits associated with the various workflow steps. Currently, monitoring of HACCP based workflows in commercial kitchens is predominantly a manual exercise and relevant records (e.g. temperature readings) are stored off-line.

In this paper, we argue that by enhancing IoT technology we can automate HACCP compliance monitoring, and facilitate other activities such as data exchange with appropriate government agencies. To support this, we describe an ontological model for recording prospective and retrospective provenance in the food safety domain. Furthermore, we demonstrate the utility of this model in the context of automated provenance generation for food safety compliance checking using a set of real sensor observations and sample inference rules.

The remainder of this paper is structured as follows: Section 2 discusses relevant related work in the provenance, semantic sensing and food safety arenas;

---

<sup>4</sup> <https://www.w3.org/TR/prov-dm/>

Section 3 describes the HACCP model in terms of its key elements, before Section 4 discusses a provenance model (FS-PROV) tailored to the food safety domain; Section 5 outlines an experimental deployment into a commercial kitchen. Using examples drawn from this real-world setting, we then discuss how provenance assertions are inferred from sensor data (Section 6), and how a range of queries can be used to check HACCP compliance (Section 7). The paper concludes with a discussion highlighting issues and future directions (Section 8).

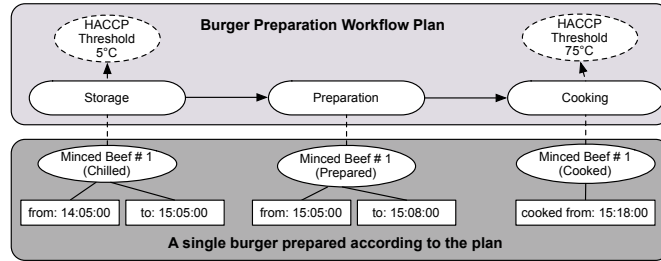
## 2 Related Work

Work in the provenance literature includes generic models for recording provenance (e.g. PROV [MM12]) and mechanisms for publishing plans and execution traces of scientific and social computation workflows (e.g. P-PLAN [GG12], D-PROV [MDB+13], ProvONE [CVLM+14], and SC-PROV [Mar16]). While the PROV specification could be used to record execution traces of food preparation workflows, the resulting provenance records would be limited in terms of their utility - due to the lack of information about the structure of the workflow plans, and configuration details of individual workflow tasks (e.g. HACCP constraints). Missier et al. [MDB+13] previously highlighted these limitations of PROV and proposed the D-PROV extension (which in turn later served as a starting point for the ProvONE extension). D-PROV and ProvONE provide a vocabulary for annotating execution traces of data-driven scientific workflows with descriptions of data-dependencies based on the planned data flow, but do not provide generic concepts for modeling constraints associated with workflow elements. Garijo and Gil [GG12] proposed a PROV extension called P-PLAN that focuses on describing abstract workflows in the form of *p-plan:Step(s)* and *p-plan:Variable(s)* to support modelling of diverse workflow structures. Steps represent the various planned activities that need to be executed, while variables represent the expected inputs and outputs of these activities. A step can refer to one or more activities recorded by PROV in the retrospective provenance record. This enables a provenance record to capture variant execution traces of the same plan. SC-PROV further extended P-PLAN with a vocabulary for describing various *sc-prov:Condition(s)* that might be associated with a step. In addition, it provides a means for capturing the parameters associated with such conditions, and the outcome of evaluation of these conditions during the workflow execution (i.e. a record of whether the condition was satisfied or not). This is modelled in a retrospective provenance record using *sc-prov:EvaluationContext*. This concept binds an *sc-prov:Condition* to a single instantiation of a *p-plan:Step*, and to the evaluation result represented as a *prov:Entity*. While the SC-PROV model supports modeling of constraints associated with individual steps, it is not able to associate constraints with variables. This is required to accommodate the HACCP view of constraints (e.g. cooked meat should have a core temperature of greater than 75°C).

The Semantic Sensor Network ontology (SSN) [CBB+12] represents the state-of-the-art in sensor metadata models and includes support for characterisation

of sensor hardware devices, sensor observations, and links between sensor capabilities and features of interest in the real world. In our view, SSN has been under-utilised in the IoT arena, where it could provide a useful platform for further standards development. Previous work [CCT14] defined alignments between the SSN and PROV-O ontologies, along with mechanisms for inferring provenance of sensor data. However, the richness of SSN descriptions for individual sensor readings can be seen as an obstacle to scalability and it is therefore necessary to consider how much of the ontology to use in any given setting. The volume of sensor observations likely in any application setting (e.g. food safety monitoring) also means that it is essential to find a way to identify and record abstractions, such as key events.

### 3 Food Safety & The HACCP Model



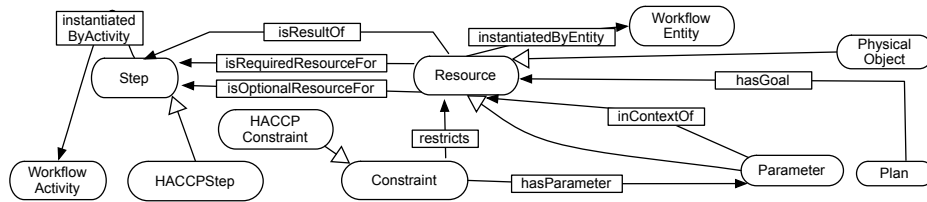
**Fig. 1.** A sample food preparation plan and corresponding instantiations of the plan concepts.

The HACCP model focuses attention on a set of critical food preparation factors. *Hazards* are anything that may introduce harm to customers, which can be microbiological, chemical or physical. *Control Measures* are ways to prevent or control hazards. For example, the survival of harmful bacteria in food, which may cause food poisoning, can be controlled by thorough cooking. *Control Measures* can be associated with a “Critical Limit”. For example, food is considered to be cooked properly if the core temperature reaches at least 75°C. Other aspects of the HACCP system encompass record keeping and verification to ensure that measures are being consistently applied. Businesses are expected to create and document their own house rules to reflect food safety working practices and articulate hazards, control measures, critical limits, etc. An example food preparation workflow is depicted in Figure 1. The example illustrates part of a typical food preparation workflow where steps (e.g. storage and cooking) are associated with relevant HACCP constraints.

To support compliance checking of HACCP-based food safety workflows, it is necessary to answer queries such as the following: **Q1:** *How long has this*

meat item been stored in compliance with HACCP guidelines for chilled storage? **Q2:** How long did this food item spend outside chilled storage before being cooked? and **Q3:** When was this food item first cooked in accordance with HACCP guidelines?. In the next section, we describe how a suite of existing provenance vocabularies can be extended to support such queries.

## 4 Modelling Provenance of a HACCP Workflow



**Fig. 2.** An illustration of the core FS-PROV concepts for modelling provenance of HACCP-based food preparation workflows and their execution.

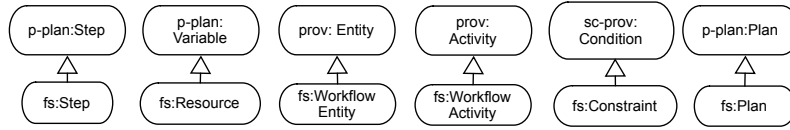
To enable modelling of HACCP-based food preparation workflows, we have extended three existing ontologies, namely PROV-O<sup>5</sup>, SC-PROV-O<sup>6</sup> and P-PLAN<sup>7</sup>. PROV-O was selected for its suitability as a means to model the retrospective workflow provenance. P-PLAN was used to model prospective provenance of a workflow, and these capabilities were further extended with concepts from SC-PROV-O in order to represent plan constraints and their evaluation results during a workflow execution. Figure 2 illustrates the core concepts of the FS-PROV ontology<sup>8</sup> (we will use the *fs* prefix when referring to these concepts in the text). FS-PROV extends the various ontologies through definition of subclasses of existing concepts with the alignments specified in Figure 3. The core concepts include definitions of planned food handling activities (*fs:Step*) and expected physical and virtual items (*fs:Resource*) that are required and produced by individual steps. In order to capture compliance requirements, we use the concept *fs:HACCPConstraint* together with the description of a physical property (*fs:Parameter*) of an item used and/or produced by the step of a workflow plan. In contrast with the SC-PROV model, *fs:Constraint* can also be associated with *fs:Resource* (e.g. the product of a cooking step) via a binding property *fs:restricts*. This property can link constraints directly to the representation of

<sup>5</sup> <https://www.w3.org/TR/prov-o/>

<sup>6</sup> <https://w3id.org/abdn/socialcomp/sc-prov>

<sup>7</sup> <http://vocab.linkeddata.es/p-plan/>

<sup>8</sup> <https://w3id.org/abdn/foodsafety/fs-prov>



**Fig. 3.** Alignment between the concepts of FS-PROV and concepts originating from PROV-O, SC-PROV-O, and P-PLAN.

food entities. As mentioned in Section 2, we argue that this is a more suitable approach to model HACCP constraints which are typically specified in the form of condition-parameter values to test some observable properties (e.g. a core meat temperature) against some threshold values. To capture the results of a condition evaluation in the retrospective provenance record, *fs:entity* (not shown in Figure 2) then binds *sc-prov:EvaluationContext* to the corresponding instantiation of an *fs:Resource*. In contrast to P-PLAN, we do not link the instantiations of planned concepts via a functional relationship to only one template description (i.e. *p-plan:Step* or *p-plan:Variable*). Instead, we define relationships in the opposite direction (see *fs:instantiatedByActivity* and *fs:instantiatedByEntity* in Figure 2). This enables independent modelling of various abstractions of workflow plans (e.g. a more detailed plan for the purposes of kitchen monitoring and less detailed for the food safety authority - without the inclusion of sensitive data) that can then be linked to the same execution trace. Furthermore, we defined the *fs:inContextOf* property to capture the relation between constraint parameters (e.g. surface temperature) and a particular resource. The *fs:hasGoal* was introduced to annotate the final output of a workflow plan (e.g. a cooked burger).

## 5 Experimental Deployment

As part of our experimental setup, we deployed 10 wireless tags from CAO Gadgets<sup>9</sup> and a wireless meat probe from Corintech<sup>10</sup> into a commercial kitchen in Aberdeen, UK. We focused on gathering temperature sensor data that related to three specific steps within a food preparation workflow: storage of raw burgers in their chilled state, preparation of the raw burgers, and cooking. Using the deployed sensors we collected data from two distinct experimental scenarios: *Scenario1* - kitchen staff complied with the HACCP temperature constraints for storage and cooking of minced beef; and *Scenario2* - staff deliberately violated these constraints.

Limitations on our experiments were caused by both hygiene and technological restrictions. The wireless tags could not be attached directly to the meat product or be used during cooking. As a result, continuous monitoring of the

<sup>9</sup> <http://wiresstag.net/>

<sup>10</sup> <http://www.corintech.com/>

transition of the raw meat product into its fully cooked state with one type of sensor was not possible. While in its raw state, burgers were contained within a plastic bag with a wireless tag attached on the outside (Figure 4 - left). A meat probe was then used to record the core meat temperature during cooking (Figure 4 - right). A common precondition for all scenarios explored in our ex-



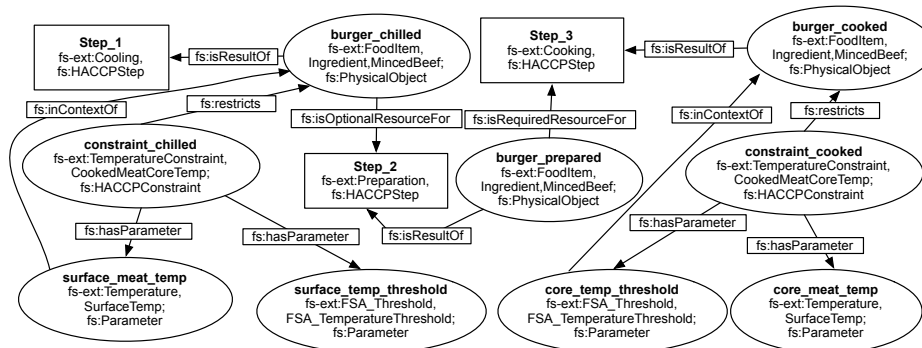
**Fig. 4.** Wireless tag attached to a pack containing a single raw burger (left). Wireless meat probe used to measure the core temperature of a burger during cooking (right).

periment was that the tracked burgers had been placed into the fridge at least two hours before the commencement of each experiment. The first part of the experiment focused on the collection of the “good” data. Chefs were asked to cook six burgers<sup>11</sup>. All burgers were kept at the correct temperature while in storage and they were also cooked according to the HACCP constraint requiring that the core meat temperature should exceed 75°C. In the second part of the experiment, we simulated non-compliance by asking the chef to under-cook four burgers. This provided us with sample sensor data from which HACCP compliance should not be inferred. In the next section we describe how the provenance records reporting compliance with HACCP constraints were generated.

## 6 Inferring Retrospective Provenance

The execution trace of our food preparation workflow was inferred using low-level sensor data and static descriptions of the workflow plan (e.g. HACCP constraint thresholds). Raw sensor data collected during the deployment were annotated using the SSN ontology. Each sensor reading (*ssn:Observation*) was associated with a specific *ssn:Sensor* (e.g. a temperature sensor) represented as an instance of an *ssn:SensingDevice*. We assumed that each food item (i.e. a burger) that was tracked within our IoT system was described by a unique URI. This was

<sup>11</sup> Four burgers were cooked separately and two burgers were cooked at the same time.



**Fig. 5.** An illustration of the plan concepts used in our burger tracking experiments.

used to represent an *ssn: FeatureOfInterest*<sup>12</sup> of an *ssn: Observation* produced by *ssn: Sensor*(s) (e.g. observations produced by wireless tag1 had a feature of interest <http://example.org/meatItem1>).

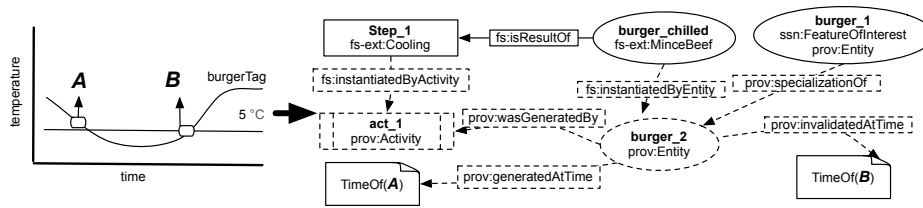
An extended FS-PROV ontology<sup>13</sup> (namespace *fs-ext*) was used to describe a domain-specific food preparation plan required for the experimental deployment. Figure 5 illustrates the manually populated ontology with instances for a three-step burger preparation plan and includes: three instances of *fs: HACCP-Step* (i.e. storage, preparation and cooking), two instances of *fs: HACCPConstraint* (i.e. constraint on chilled meat, and constraint on cooked meat), four *fs: Parameters* (i.e. observed surface and core temperatures, and thresholds) and three instances of *fs-ext: MincedBeef* (i.e. *fs: Resources* representing the changing states of burgers within the workflow from chilled to cooked).

The FS-PROV-based provenance abstractions were created using SPARQL INSERT queries that implemented rules to recognise events from low-level sensor data (see Figures 6 and 7). To infer provenance entities indicating that a burger was in a chilled state (i.e. instantiations of the *burger\_chilled* resource in Figure 5), we used the first observation which reported meat surface temperature falling below the corresponding HACCP threshold. Similarly, an observation reporting meat surface temperature rising above the HACCP was used to infer a new provenance entity representing a burger in a preparation stage and at the same time the “chilled entity” was invalidated using the *prov: InvalidatedAtTime* assertion. Entities representing cooked burgers were generated at the point when an observation from a meat probe reported core temperature above the corre-

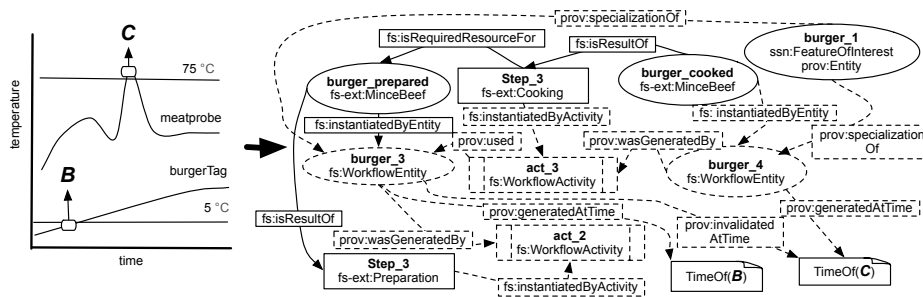
<sup>12</sup> The meat probe sensor data had to be manually annotated with the feature of interest (i.e. the meat item for which the core temperature was measured) as the current design of the probe does not support automatic recognition of probed items.

<sup>13</sup> [https://raw.githubusercontent.com/m-markovic/FoodSafety-Data/master/fso\\_extended.ttl](https://raw.githubusercontent.com/m-markovic/FoodSafety-Data/master/fso_extended.ttl)





**Fig. 6.** An illustration of observed temperature variations in relation to a HACCP limit (left) and their relationship to inferred provenance annotations (dashed lines in the graph on the right).



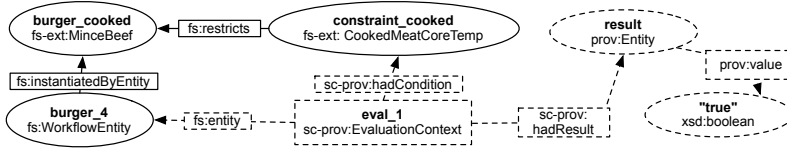
**Fig. 7.** An illustration of observed temperature variations in relation to HACCP limits (left) and their relationship to inferred provenance annotations (dashed lines in the graph on the right).

sponding HACCP threshold. At the same time the entities describing a burger in a preparation stage were updated with the *prov:InvalidatedAtTime* assertion.

As a result of our approach, entities representing burgers in their chilled and cooked state were only inferred if the corresponding HACCP constraints were satisfied. To record this, we generated additional annotations noting that a corresponding HACCP constraint had been satisfied (Figure 8).

## 7 Querying Food Provenance

The query presented in Figure 9 can be used to retrieve the times when a meat item (identified by a specific URI) was observed to comply with the HACCP temperature constraint for chilled storage. By using this approach, we were able to construct additional queries and successfully retrieve evidence which recorded whether a food item was in compliance with relevant HACCP constraints throughout the storage, preparation and cooking stage. When no evidence of compliance was recorded (e.g. a burger was under-cooked) the corresponding query returned no results. To evaluate our provenance queries we



**Fig. 8.** An example documenting that a *fs:WorkflowEntity* satisfies a planned constraint.

```

SELECT ?item ?start ?finish
WHERE {
  ?storageStep a fs-ex:Storage.
  ?resultResource fs:isResultOf ?storageStep;fs:instantiatedByEntity ?result.
  fs-ex:HACCPTempConstraintChilledFood fs:restricts ?resultResource.
  ?result a fs:WorkflowEntity;prov:specializationOf ?item; prov:generatedAtTime ?start.
  ?result prov:invalidatedAtTime ?finish.
  ?evaluationContext fs:entity ?result.
  ?evaluationContext sc-prov:hadCondition fs-ex:HACCPTempConstraintChilledFood.
  ?evaluationContext sc-prov:hadResult ?evaluationResult.
  ?evaluationResult a fs:WorkflowEntity; prov:hasValue "true".
}
VALUES (?item) {(<http://example.org/meatItem1>)}

```

**Fig. 9.** An example SPARQL query to retrieve start and end time for the entity representing a food item in a chilled state.

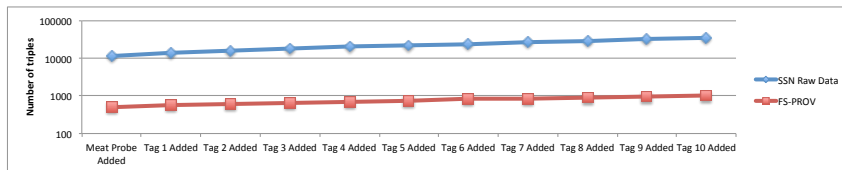
generated a gold standard data set based upon a researcher’s observations of the kitchen activities; this was then used to cross-check the results provided by the sample provenance queries.

It is important to recognise that from the HACCP-based provenance perspective, the activities recorded in the provenance record do not necessarily mirror the events observed in the physical kitchen environment. To illustrate this point, consider a situation when a food item is removed from chilled storage. Staff would immediately consider this item as no longer being stored in a chilled state. However, the item might still maintain a temperature below the HACCP threshold for some time after leaving the fridge. In the provenance record the item would therefore remain in the chilled storage state for some time (until its temperature rose beyond the HACCP limit).

To illustrate the utility of the FS-PROV model, we have compared the number of triples required to describe the provenance abstractions to the number of triples required to describe the raw sensor data using SSN (see Figure 10). We used JENA’s<sup>14</sup> *OntModel* to store annotated sensor data using the SSN concepts described earlier in this paper (see Section 6).

The provenance model (with model specification set to *OWL\_MEM\_RDFS\_INF*) was firstly loaded with an extended FS-PROV ontology (i.e. the workflow plan)

<sup>14</sup> <https://jena.apache.org>



**Fig. 10.** A comparison of the number of SSN triples required to characterise the sensor data vs. the corresponding FS-PROV provenance assertions.

and this was followed by addition of inferred retrospective provenance for each of the observed meat items. From our results, it is clear that FS-PROV based provenance abstractions can significantly reduce the number of triples required to capture compliance of HACCP-based workflows. However, this approach also forces us to consider the trade-off between storing abstractions vs. the original sensor data, which would enable re-evaluation of compliance. In addition, it raises additional questions regarding the reliability of tools that generate such abstractions. FS-PROV could potentially re-use other SC-PROV concepts (e.g. *sc-prov:ParameterCollection*) to record instantiations of parameter values (e.g. temperature readings). However, it may be necessary to introduce new mechanisms to decide what parameter values should be recorded. For example, we might have recorded three sensor readings (i.e. HACCPConstraint parameter instantiations) that prove that a food item was cooked (e.g. readings from a meat probe over a period of 30 seconds). However, we might have recorded thousands of observations that prove the compliance of a food item being in its chilled state (e.g. readings from a wireless tag over a period of 2 days). If we recorded all the parameter instantiations that correspond to the compliance of an entity with the HACCPConstraint for chilled storage, we would be negating the benefits in terms of storage requirements. Alternatively, if only a subset of readings (e.g. the readings just before and after an entity entered the chilled state) were recorded, new classes or properties would be required to record that these were only a sample of the observed sensor data. During our experiments we encountered various issues with sensor accuracy and sampling rates. While information about sensor calibration and measurement errors can be recorded as part of the SSN descriptions of raw data, we did not consider these in our work, and they remain challenges for the future.

## 8 Conclusions & Future Work

In this paper we have outlined a promising approach that can be used to generate provenance abstractions of food safety sensor data. We have demonstrated that provenance records could play a significant role in facilitating scalable IoT infrastructures in the food safety domain. Our initial experiments were performed on static (archival) datasets. In our continuing work we aim to evaluate the use

of stream-based infrastructures for managing food safety sensor data. We will investigate the feasibility of on-the fly inference of provenance abstractions to support real-time food safety monitoring systems. In addition, we will explore other potential provenance queries such as **Q4**: *Who performed the activity that influenced this food item?* **Q5**: *Why was the activity that influenced this food item performed?* and **Q6**: *Where were the food preparation activities performed?*. To answer Q4, the sensors would have to be able to identify the agent (e.g. chef) who performed a particular activity involving the tracked food item. Such information could then be captured within a provenance record by associating an agent with a relevant activity such as an instantiation of the cooking step. To answer Q5 and Q6, a provenance record would have to include descriptions of activities that triggered the creation of entities which represent changing states of a food item. For example, an activity representing a customer order would trigger the activity representing the instantiations of the individual planned steps such as preparation and cooking. The activities could then be linked using *prov:atLocation* to a location where they were executed, for example, to a specific restaurant.

## References

- CBB<sup>+</sup>12. M. Compton, P. Barnaghi, L. Bermudez, R. García-Castro, O. Corcho, S Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog, V. Huang, K. Janowicz, W. D. Kelsey, D. Le Phuoc, L. Lefort, M. Leggieri, H. Neuhaus, A. Nikolov, K. Page, A. Passant, A. Sheth, and K. Taylor. The {SSN} ontology of the {W3C} semantic sensor network incubator group. *Web Semantics: Science, Services and Agents on the World Wide Web*, 17:25–32, 2012.
- CCT14. M. Compton, D. Corsar, and K. Taylor. Sensor data provenance: Ssno and prov-o together at last. In *Terra Cognita and Semantic Sensor Networks*, pages 67–82, 2014.
- CVLM<sup>+</sup>14. V. Cuevas-Vicenttín, B. Ludäscher, P. Missier, K. Belhajjame, F. Chirigati, Y. Wei, S. Dey, P. Kianmajd, D. Koop, S. Bowers, and I. Altintas. Provone: A prov extension data model for scientific workflow provenance. URL: <http://vcvcomputing.com/provone/provone.html>, 2014.
- GG12. D. Garijo and Y. Gil. Augmenting prov with plans in p-plan: scientific processes as linked data. In *Proceedings of the Second International Workshop on Linked Science 2012 - Tackling Big Data*. CEUR, 2012.
- Mar16. Milan Markovic. *Utilising Provenance to Enhance Social Computation*. PhD thesis, University of Aberdeen, 2016.
- MDB<sup>+</sup>13. P. Missier, S. Dey, K. Belhajjame, V. Cuevas-Vicenttin, and B. Ludäscher. D-prov: extending the prov provenance model with workflow structure. Technical report, School of Computing Science, Newcastle University, 2013.
- MEC13. M. Markovic, P. Edwards, and D. Corsar. Utilising provenance to enhance social computation. In *The Semantic Web-ISWC 2013*, pages 440–447. Springer, 2013.
- MM12. L. Moreau and P. Missier. Prov-dm: The prov data model. W3C Recommendation (April 2012), <http://www.w3.org/TR/prov-dm/>, 2012.