

Toward Dynamic Expiration Dates: An Architectural Study

Åse Jevinger and Paul Davidsson

Abstract The durability of perishable food varies due to different storage and handling conditions during the supply chain as well as final consumer activities. If the durability of the individual products can be estimated, dynamic expiry dates may be developed and used to prevent food waste, ensure quality, and improve supply chain activities etc. Depending on the system architecture used for such a service, different qualities can be obtained in terms of usability, accuracy, security etc. This paper presents a novel approach for how to identify and select the most suitable system architectures of a dynamic expiry date service. The approach is illustrated by focusing on one of the potential user groups, the supply chain managers. The approach consists of three steps: (i) identify the potential architectures, (ii) filter out the least relevant candidates by applying a specified set of principles, and (iii) perform an analytic hierarchy process (AHP) based on a set of quality attributes.

Keywords Dynamic expiry date · Perishables · Architecture · AHP · Supply chain management

Introduction

Studies show that temperatures during supply chain activities often differ from the ones recommended by the producer, which might result in shorter durability of perishables, such as dairy products, fresh meat, and fish (Likar and Jevšnik 2006).

Å. Jevinger (✉) · P. Davidsson
Malmö University, Malmö, Sweden
e-mail: ase.jevinger@mah.se

P. Davidsson
e-mail: paul.davidsson@mah.se

Thereby the specified expiry dates¹ are no longer valid. On the other hand, food that has been well treated can often be safely consumed after the expiry date has passed. It has been estimated that one third of all food produced globally for human consumption is lost or wasted (Gustavsson et al. 2011). Jedermann et al. (2009) state that among environmental parameters during transport, temperature has the most significant influence on the quality of food products. Studies furthermore show significant temperature variations inside a vehicle as well as during storage (Moureh and Flick 2004).

To prevent people from discarding food still suitable for human consumption, as well as to ensure the quality of the expiry dates, information about the actual conditions during the supply chain activities are needed. By using local sensors to measure the environmental conditions, such as the temperature, the quality of each individual product can be estimated. These sensors must be present throughout the transport chain, and the closer to the products they are placed, the more accurate estimations can be made. Based on the estimated quality, the expiry dates can be updated continuously enabling actors within supply chain management as well as the final consumers, to make more informed decisions about how to handle the food and when to discard it. Furthermore, a dynamic expiry date service might be colligated with other services such as tracking and tracing, product information (for instance origin and handling instructions), carbon dioxide labeling, and dynamic pricing based on currently estimated expiry date (Bartels et al. 2010).

The aim of the paper is to present and apply a novel approach for how to select the most suitable system architecture for a service providing dynamic expiry dates of perishable food. The approach consists of three steps:

1. Identify all potential system architecture candidates based on a new system architecture representation model.
2. Filter out the least relevant candidates by applying a specified set of principles.
3. Perform an AHP based on a set of quality attributes to determine which architecture is the most suitable.

The approach is used to analyze all architectures satisfying the requirements of one particular user group, the supply chain managers, in order to select the most suitable architecture. The architectures differ in the intelligence required at product-level, vehicle/terminal level, mobile terminal level, and central level.

¹The producer generally guarantees good quality as long as the best-before date has not passed (if certain conditions have been respected). The producer generally guarantees food safety as long as the use-by/expiry date has not passed (if certain conditions have been respected). Shelf life is the length of time for which an item remains usable, fit for consumption, or saleable.

Related Work

Concepts implying a higher level of intelligence related to the product itself, in comparison to only possessing an associated ID, are often referred to “smart” or “intelligent,” such as “Intelligent Goods,” “Smart Goods,” or “Intelligent Products” (Meyer et al. 2009). Depending on the level of intelligence, these concepts can be applied for monitoring the local conditions during transportation (López et al. 2011), and maybe for calculating a dynamic expiry date. The intelligence related to a product may or may not be located on the product itself (Meyer et al. 2009). However, for the dynamic expiry date service, local intelligence, for instance implemented on the product, a pallet or in the vehicle, able to perform condition monitoring is required.

Within the area of communicative packaging, potential means to monitor the condition of packaged contents by enhancing the intelligence of the package itself have been investigated. The solutions consist of a combination of packaging technologies and communicative signals (e.g., changing colors, diagrams, displays) (Dobon et al. 2011). Communicative packaging is seen as a specific type of smart packaging, which uses different technologies to add extra features to packaging (e.g. information about expiry date, identification about the type/origin of the product and protection against counterfeiting). A life-cycle assessment study on the use of a flexible best-before-date communicative device (FBBD, with a temperature logger and a display) on packaging consumer units, shows that the use of FBBD devices decreases environmental burdens associated to the production, packaging, and delivery to the point of sale, thanks to reduction in food losses (Dobon et al. 2011). Other case studies show potential benefits of radio-frequency identification (RFID)-based cold-chain monitoring in increased sales due to reduced out-of-stock, reduction in inventory due to lower safety stock, improvement of visibility and transparency in the supply chain etc. (Jol et al. 2006). Benefits from using time temperature indicators (TTIs) have also been investigated (Sahin et al. 2007; Bhushan and Gummaraju 2002). An alternative to enhancing the package of a product itself is to place the local intelligence, including temperature sensors, on a higher level, for instance on containers or in vehicles (Ruiz-Garcia et al. 2007). A discussion about different degrees of decision freedom that may be dedicated to products as well as commonly used applications on different hardware layers can be found in Jedermann and Lang (2008). Placing intelligence on different levels in the transport system results in different processing, information, and communication requirements as well as enables different service qualities and functionalities. Depending on the purpose of local condition monitoring and expiry date estimations, different solutions based on different system architectures might thereby be preferred—in particular by different target user groups. To the best of our knowledge, no previous study has identified these system architectures and investigated the differences between them, with respect to the issues above.

Methodology and Service Description

In order to be able to compare different system architectures of the dynamic expiry date service, they must first be identified and characterized. We identify the potential architectures by first specifying all activities (A) that must be performed, as well as the different locations (L) where each of these activities could be performed. By combining the activities with the locations we get the set of all potential architectures (i.e., the Cartesian product $L^{|A|}$). This set is then reduced by filtering out impossible and obviously impractical architectures. From the remaining set of architectures, the most suitable ones, with respect to the functional and quality requirements of each target user group, shall be found. Ideally, every identified architecture should be evaluated and compared. However, due to the amount of work this process would require, the least suitable ones are filtered out in two steps. First, all solutions possible for each user group are identified based on some prerequisites regarding how the expiry date should be presented. In this paper, we only focus on the architectures satisfying the requirements of the supply chain managers. Second, heuristics derived from general requirements on the architecture are applied to identify the most promising candidates. Finally, an AHP is performed on the resulting set of architectures, prioritizing the most suitable ones according to a set of quality-based criteria. AHP is a method for analyzing complex decision problems with multiple criteria (Saaty 1980).

Input to the dynamic expiry date service is Goods-ID (the unique ID of a product, e.g. SGTIN) and output is the expiry date. In this study, we assume that the calculation of expiry date is based on sensor data related to the condition of the product, including both current and historical data, for instance temperature values. Thus, a local sensor is needed to sense and record the local conditions according to some time interval, and the stored sensor data must be possible to read. Theoretically, the service could have been based on measuring the current bacteria level only, without regard to any historical data. However, given the current status of sensor technology, direct and possibly automated measurement of the current bacteria level in a product is not a viable approach. The Goods-ID is assumed to be stored at product level, making all products electronically identifiable.

The System Architecture Representation Model

The expiry date service requires the execution of four main activities: (a_1) the local conditions of the product have to be measured, (a_2) the recorded sensor data must be stored, (a_3) the expiry date must be calculated based on the stored sensor data, and (a_4) the result must be presented to the user of the service. These activities ($A = \{a_1, a_2, a_3, a_4\}$) can be located: at product level (P), on a fixed local device (N), on a mobile local device (M), in a remote device (R) or in the cloud (C), i.e. $L = \{P, N, M, R, C\}$. Below, a more detailed explanation of the different locations

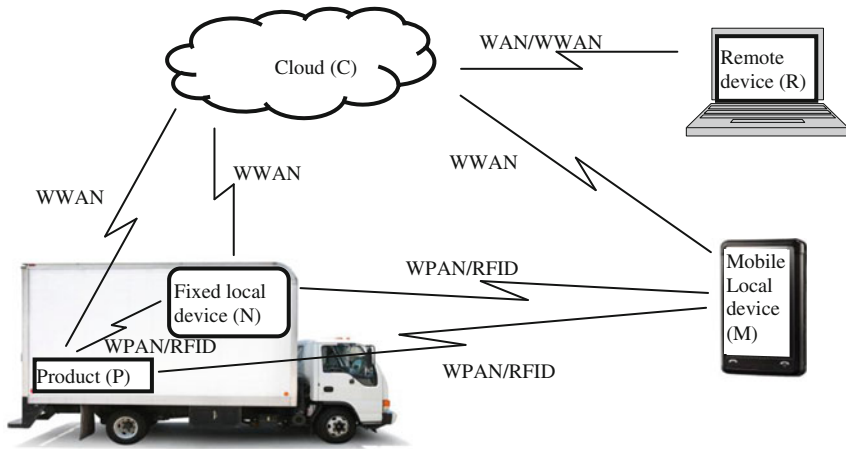


Fig. 1 Possible communication links between different parts of the service system, (W)WAN (wireless) wide area network, WPAN wireless personal area network

is given, whereas Fig. 1 illustrates the corresponding possible communication links. In this study, we assume that both R and M are connected with the cloud (when there is coverage) and that N is available whenever needed in accordance with the requirements of the architecture.

P: Entity at product level, e.g., attached to the primary package, secondary package or a pallet. This entity follows the product throughout the transport.

N: Fixed local device near the products, located for instance inside a vehicle, a terminal, or a refrigerator. The N used by the service on behalf of a product may change during transport, when the product is moved.

M: Mobile local device near the products, for instance a mobile phone provided with an app or an RFID reader. Consumers and local supply chain personnel (e.g. drivers and terminal workers) may employ such a device. M communicates with N and P directly, via RFID or WPAN (Wireless Personal Area Network). For instance, M may read the Goods-ID from the product and then retrieve the corresponding expiry date from C.

R: Remote device, for instance, a stationary computer running an ERP system. R is unable to communicate directly with N and P but communicates through the cloud instead.

C: Cloud-based entity, for instance a network server, accessible from any device with Internet access.

The set of potential architectures, $L^{|A|}$, consists of 625 elements of which some correspond to architectures that are impossible or obviously impractical. Measuring the local conditions in C or R is impossible, and to do it on M is impractical since M may be moved around. Also, to store the sensor data on M or N is impractical since the service should not be dependent on M being present (receiving sensor data via

WPAN/RFID) and since storing sensor data on N would require a data transfer whenever N changes and that might not be possible (or desirable for security reasons). Finally, to present the expiration date in C is not meaningful. To sum up, the activities may be executed in the following locations in the system:

1. Measurement of local conditions: P or N
2. Sensor data storage: P, C or R
3. Calculation of estimated expiry date: P, N, M, R or C
4. Presentation of expiry date: P, N, M, or R

The calculated expiry dates are assumed to be stored either where they are calculated or where they are to be presented in the system. The calculation of the expiry date is based on current and historical sensor data and this sensor data might, on the other hand, be stored somewhere apart from where it is collected or used. The reason for storing this data in a third place might be to avoid overloading the product level or to allow for the fixed local device to be changed during transport. Depending on where the sensor data is stored, a higher level of intelligence might be required by the involved entities, as well as additional communication.

The filtration above results in 120 different alternative architectures ($2 \times 5 \times 4 \times 3 = 120$). We have investigated each one of them with respect to individual characteristics, usefulness for different target user groups, communication paths and required capabilities (according to Jevinger et al. (2011)). Additionally, the set of architectures has been further extended to also incorporate different alternatives for long distance communication between P and C/R (resulting in 186 architectures), since this in some cases introduces additional entities and capability requirements. Long distance communication between P and C/R can be performed using a direct link, or by transmission via N or M. In this paper, we assume that P does not communicate directly with C/R, but only through N/M. The reason for this assumption is that the most reasonable solution considering today's technology, is to use short distance wireless communication to N/M and let N/M be responsible for the long distance communication.

The architectures can be grouped based on where measurement, storage, and calculation are located. All architectures with the same locations of these activities only differ in where the expiry date is presented, and they may therefore be combined to provide several means of accessing the expiry date. For instance, a solution showing the expiry date in both the R and M might be required (e.g. to satisfy requirements from drivers and terminal workers as well as centralized supply chain managers), based on architectures with the same locations of the other service activities (measurement, storage and calculation). In this study, we show an approach for how to prioritize and select the most suitable solutions for one of the target user groups; however, this approach can be used to prioritize the solutions for other user groups as well. If a combination is required as described, the topmost architectures with the same locations of measurement, storage, and calculation should be selected from the resulting prioritized lists of architectures.

Heuristics

Different user groups require different alternatives for how the expiry date should be accessed. This paper focuses on supply chain managers and we assume they will read the date in a remote device (R). Ideally, all architectures fulfilling this requirement should be evaluated with respect to the functional and quality preferences of the supply chain managers. This would mean evaluating 48 architectures. For other user groups identified, the set of architectures to evaluate result in 138 architectures, and performing an AHP on all of these would require a huge amount of work. Moreover, many of the architectures that are possible from a theoretical perspective reflect solutions that do not seem reasonable from a practical perspective, such as calculating the expiry date in M and showing the result on P. Therefore, we have developed two principles for filtering out the least promising architectures, from the perspective of the supply chain managers. The list of principles may have to be extended for other user groups, though the principles below should be applied for those as well.

Principle 1: If sensor data is stored on C/R and the expiry date is presented on R, the calculation should be co-located with the storage or the presentation. The purpose of this principle is to prioritize the simplest solutions that do not involve sending information back and forth, in particular from the central level, to the local level, and back again. For instance, architectures storing sensor data in R, using P/N for calculation, and showing the resulting expiry date in R, are removed. This principle filters out the following architectures (Measurement-Storage-Calculation-Presentation): P-C-P-R, N-C-P-R, P-C-N-R, N-C-N-R, P-C-M-R, N-C-M-R, P-R-M-R, N-R-M-R, P-R-C-R, N-R-C-R, P-R-N-R, N-R-N-R, P-R-P-R, N-R-P-R.

Principle 2: Architectures showing the expiry date on R shall not use M for calculation. The supply chain managers should not be dependent on M for receiving the expiry date since M is mobile and is therefore not guaranteed to always be present. This principle filters out the following architectures: P-P-M-R, N-P-M-R.

As a result, 14 candidates have not been rejected by the principles and are thus subjects for AHP analysis. The choice of where to measure the local conditions (P or N) depends on the temperature variations between N and the product, but also on how sensitive the product is to measurement errors. Based on the properties and shelf life model of a specific product, architecture candidates measuring the conditions on N may in some cases be filtered out. In this paper, we assume relatively insensitive products in order not to narrow the analysis more than necessary.

AHP Analysis

AHP is an approach for selecting the best from a number of alternatives, which are evaluated with respect to several criteria. Pairwise comparisons are used to develop overall priorities reflecting the importance of each criterion, relative to the goal of the selection problem. The performance of each alternative on each criterion must also be determined, and together with the priorities of the criteria, a ranking of the alternatives can be produced. AHP has previously been successfully used for evaluating architecture candidates (Svahnberg et al. 2003). As a first step, the set of criteria must be selected. We have identified the quality attributes listed in Table 1, as relevant for the dynamic shelf expiration date service.

The second step of AHP involves prioritizing the criteria according the importance of each criterion, in relation to the goal of the selection problem. Each criterion is pairwise compared to all other criteria with respect to how desired they are to the service system. These comparisons are based on interviews with three actors within the food supply chain. We only use consistent priorities, and after the

Table 1 Criteria for AHP analysis

Criteria	Meaning
Response time	How long it takes to process a request (O'Brien et al. 2007). We define it as the time it takes from asking for the expiry date, until it is available for the user. Situations where the expiry date cannot be provided at all are covered by availability
Security	We consider integrity and confidentiality as the most relevant aspects of security. Integrity guarantees that information is not corrupted and confidentiality ensures that access to the information/service is granted only to authorized subjects (O'Brien et al. 2007). In our context, integrity corresponds to how difficult it is to manipulate the system into producing no or erroneous expiry dates, and confidentiality to how difficult it is for unauthorized users to access the expiry date
Availability	The proportion of time a system or component is operational and accessible (O'Brien et al. 2007). In our context, it is measured as the proportion of the time the service is operational during its lifetime. We do not address physical damage of equipment
Modifiability	The ability to make changes to a system quickly and cost-effectively (Clements et al. 2001). In our context, it is a measure of how easy and cost-effectively the software (date estimation algorithms, software functionality etc.) and the hardware components (tags, readers, displays etc.) of the service can be exchanged
Scalability	The ability of the system to function well when the system is changed in size or in volume in order to meet users' needs (O'Brien et al. 2007). In our context, it concerns how well the service system is able to handle a growing number of products or longer transport distances
Accuracy	The precision of computations and control (Larsson 2004). In our context, it concerns how well the estimated expiry date approximates the actual expiry date, including the reliability of the sensor data

Table 2 Priorities derived from interviews and comparisons between the architecture candidates with respect to the criteria as well as the final results of the AHP analysis

	Response time	Security	Availability	Modifiability	Scalability	Accuracy	AHP results
<i>Priorities</i>	<i>0.1467</i>	<i>0.1333</i>	<i>0.1533</i>	<i>0.1200</i>	<i>0.1133</i>	<i>0.1667</i>	
P-P-P-R	0.0719	0.0727	0.0727	0.1017	0.0455	0.1172	0.0819
N-P-P-R	0.0240	0.0242	0.0242	0.0508	0.0455	0.0391	0.0339
P-P-N-R	0.0719	0.0727	0.0727	0.0508	0.0227	0.1172	0.0715
N-P-N-R	0.0240	0.0242	0.0242	0.0339	0.0227	0.0391	0.0283
P-P-C-R	0.0360	0.0364	0.0364	0.1017	0.0455	0.1172	0.0631
P-C-C-R	0.0360	0.0727	0.0364	0.1017	0.0909	0.0781	0.0673
N-P-C-R	0.0288	0.0182	0.0182	0.0508	0.0455	0.0391	0.0326
N-C-C-R	0.1439	0.0727	0.1455	0.0508	0.1364	0.0469	0.0990
P-C-R-R	0.0360	0.0727	0.0364	0.1017	0.0909	0.0781	0.0673
N-C-R-R	0.1439	0.0727	0.1455	0.0508	0.1364	0.0469	0.0990
P-P-R-R	0.0719	0.0727	0.0727	0.1017	0.0455	0.1172	0.0819
N-P-R-R	0.0240	0.0242	0.0242	0.0508	0.0455	0.0391	0.0339
P-R-R-R	0.0719	0.1455	0.0727	0.1017	0.0909	0.0781	0.0919
N-R-R-R	0.2158	0.2182	0.2182	0.0508	0.1364	0.0469	0.1483

pairwise comparisons, the priorities are normalized to sum up to one. The resulting priorities from the interviews are listed in the first row of Table 2.

In the third step of AHP, the architecture candidates are compared with all other candidates, with respect to the criteria. These comparisons are based on pairwise subjective assessments and as before, we use consistent priorities, normalized to sum up to one. The resulting sets are shown in Table 2.

In the fourth step, the relative importance of each criterion is combined with how well each architecture candidate satisfy the criteria, by multiplying the normalized priorities related to the architecture candidates with the normalized priorities of the criteria. The resulting products are thereafter summed for each architecture candidate and the values of these sums reflect the suitability of each candidate, in relation to the other candidates. The final values for each architecture candidate are presented in the last column of Table 2.

Table 2 shows that, with this approach, architecture N-R-R-R receives the highest priority for the supply chain managers, with respect to quality and functionality requirements. Architectures N-C-R-R and N-C-C-R are similar and regarded as second best. Taking into consideration the costs of each solution as well naturally would change the order of priorities. For instance, architecture P-R-R-R most probably entails a higher cost than architecture N-P-R-R, since architecture P-R-R-R requires sensor capability on P whereas architecture N-P-R-R may be implemented using a simple read/write passive RFID tag on P. Implementing a high level of intelligence in product-level devices is usually relatively costly since the number of products is high in relation to the other potential locations, for instance vehicles. Thereby, the approach presented in this paper shall primarily be used for

prioritizing the architectures from a functional and quality perspective, and based on these results, in combination with the costs of each architecture candidate, the most suitable solution can be selected.

Conclusions and Future Work

We have shown a novel approach for how to identify the most suitable system architectures of an expiry date service. The approach is illustrated by focusing on one user group, the supply chain managers. Apart from the analytical desktop work, the selection process also incorporates interviews with three actors within the food supply chain, affecting how different qualities should be valued.

In order to be able to list all architectures we have developed a representation model, which is used to characterize the architectures based on where the different activities involved in the service are located. The primary advantages with this approach are that all possibilities can be identified and evaluated. Thereafter, we have used user group requirements and heuristics to filter out architecture candidates that reflect the general requirements on the final solution. The candidates have been evaluated in an AHP based on a number of quality criteria, and the results are presented as a list of priorities of the candidates.

In this study, we have assumed N is available whenever needed. In reality, this might not always be the case. Architectures in which N is only available during certain parts of the transport, sensor data might only be possible to transmit at arrival to the terminals, produce other values of the quality criteria. A more concrete situation would reveal the alternative locations of N and taking these locations into consideration as well would expand the set of architecture alternatives. This paper is focused on the fundamental solutions and if other varieties need to be evaluated, the approach presented in this paper may be used for these as well.

Once the solutions satisfying the functional and quality requirements have been identified, they should be analyzed with respect to their costs, in order to determine the ideal solution. We plan to consider the costs in a second paper, by performing a cost–benefit analysis on the set of architecture candidates resulting from this paper. The environmental impacts of each architecture will also be considered in the AHP criteria. Furthermore, the second paper will describe a pilot study implementing a number of architecture candidates, and the measured values from this study will be used in an additional AHP.

Acknowledgment The work presented in this paper is a part of an interdisciplinary project called “Minimized food waste with dynamic expiry dates,” funded by the TvärLivs programme managed by VINNOVA (Swedish Governmental Agency for Innovation Systems).

References

- Bartels PV, Tromp SO, Rijgersberg H, Kreft F (2010) Improvement of sustainability in the perishable food supply chain by using communicative packaging devices. In: *Towards effective food chains: models and applications*. Academic Publishers, Wageningen
- Bhushan N, Gummaraju K (2002) A petri net based simulation approach for evaluating benefits of time temperature indicator and wireless technologies in perishable goods retail management. In: *FOODSIM'2002 the second international conference on simulation and modeling in the food and bio-industry*
- Clements P, Kazman R, Klein M (2001) *Evaluating software architectures*. Addison-Wesley, Reading
- Dobon A, Cordero P, Kreft F, Østergaard SR, Robertsson M, Smolander M, Hortal M (2011) The sustainability of communicative packaging concepts in the food supply chain. A case study: part 1. Life cycle assessment. *Int J Life Cycle Assess* 16:168–177
- Gustavsson J, Cederberg C, Sonesson U, van Otterdijk R, Meybeck A (2011) *Global food losses and food waste*. Food and Agriculture Organization of the United Nations, Rome
- Jedermann R, Lang W (2008) The benefits of embedded intelligence -tasks and applications for ubiquitous computing in logistics. *The internet of things*. Springer, Berlin, pp 105–122
- Jedermann R, Ruiz-Garcia L, Lang W (2009) Spatial temperature profiling by semi-passive RFID loggers for perishable food transportation. *Comput Electron Agric* 65:145–154
- Jevinger Å, Davidsson P, Persson JA (2011) A framework for agent-based modeling of intelligent goods. *Agents in principle, agents in practice*. Springer, Berlin, Heidelberg, pp 97–112
- Jol S, Kassianenko A, Wszol K, Oggel J (2006) Issues in time and temperature abuse of refrigerated foods. *Food Saf* 11(6):30–32
- Larsson M (2004) *Predicting quality attributes in component-based software systems*. Dissertation, Mälardalen University, Västerås
- Likar K, Jevšnik M (2006) Cold chain maintaining in food trade. *Food Control* 17:108–113
- López TS, Ranasinghe DC, Patkai B, McFarlane D (2011) Taxonomy, technology and applications of smart objects. *Inf Syst Front* 13:281–300
- Meyer GG, Främling K, Holmström J (2009) Intelligent products—a survey. *Comput Ind* 60:137–148
- Moureh J, Flick D (2004) Airflow pattern and temperature distribution in a typical refrigerated truck configuration loaded with pallets. *Int J Refrig* 27:464–474
- O'Brien L, Merson P, Bass L (2007) Quality attributes for service-oriented architectures. In: *International workshop on systems development in SOA environments*, IEEE Computer Society
- Ruiz-Garcia L, Barreiro P, Rodríguez-Bermejo J, Robla J (2007) Review monitoring the intermodal, refrigerated transport of fruit using sensor networks. *Span J Agric Res* 5:142–156
- Saaty TL (1980) *The analytic hierarchy process*. McGraw Hill, New York
- Sahin E, Babai MZ, Dallery Y, Vaillant R (2007) Ensuring supply chain safety through time temperature integrators. *Int J Logist Manag* 18:102–124
- Svahnberg M, Wohlin C, Lundberg L, Mattsson M (2003) A quality-driven decision-support method for identifying software architecture candidates. *Int J Softw Eng Knowl Eng* 13:547–573