Chapter 5
Monitoring in the Healthcare Setting

Federico Chesani, Catherine G. Enright, Marco Montali, and Michael G. Madden

5.1 Introduction

Monitoring is an activity in which a running system is observed, so as to become aware of its state. The fact that the system is *observed* makes monitoring complementary to approaches like formal verification and validation, which are tailored to assess the quality and trustworthiness of the system before the execution. While verification/validation works on a model of the system, as discussed in Chap. 19 monitoring observes real system executions. The fact that the system is typically observed while *running* makes monitoring complementary to approaches like data/process mining, which are applied post-mortem, i.e., on historical, logged information.

Typically, monitoring does not only work solely on a *stream of data* representing the evolving trace of an *actual* system behavior, but also considers a *model* capturing the *expected* system behavior. In this light, monitoring is concerned with continuously contrasting the actual and expected behavior, so as to correspondingly provide a meaningful feedback to the actors responsible for the system execution, not only to make them aware of its current state and of deviations with the expected state, but also to make them able to promptly react to exceptional situations. This is why it is considered to be one of the main pillars of what is termed *operational decision support* [1, 2].

5.2 Monitoring Tasks

Figure 5.1 depicts a generic monitoring framework, and the typical tasks involved. We briefly describe them.

Calibration involves adapting system parameters so that the model of expected behaviour is attuned to the individual patient being monitored, as opposed to an idealized patient. This is a key component of personalized medicine.

A. Hommersom and P.J.F. Lucas (eds.), *Biomedical Knowledge Representation*, LNAI 9521, DOI 10.1007/978-3-319-28007-3_5

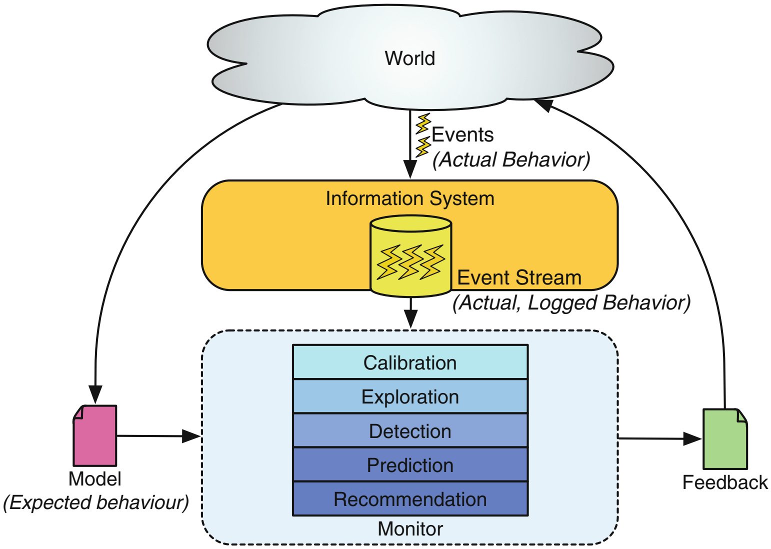

Fig. 5.1 A generic monitoring framework (inspired by [2])

Exploration deals with the proper reconstruction of the current system state (and possibly of the history) by using the events collected so far, and the model of the system. This typically includes a range of visualization techniques, tailored to provide an intuitive, end-user oriented representation of the system current state (and history).

Detection is meant to check the alignment between the actual behavior (reconstructed from the stream of events), and the expected behavior (obtained from the model). When the two behaviors are indeed aligned, we say that the actual behavior is *conformant* or *compliant* with the expected one.[1] If instead a deviation is detected, a warning is issued by the monitor, so as to make the responsible actors aware of the misalignment.

Prediction exploits the model, and typically also historical data about past executions of the system, so as to determine the likely future evolution(s) of the currently monitored behavior. When the resulting predicted behavior has undesired aspects, proper countermeasures can be taken in advance so as to properly redirect the system.

Recommendation refines prediction by automatically providing suggestions on what to do next. Obviously, the generated recommendations have to be continuously reconsidered in the light of new, incoming events.

[1]The latter acception is typically employed when the model of the system carries a normative meaning.

5.3 Monitoring Issues and Knowledge Representation

From Fig. 5.1, the main issues involved in a generic monitoring framework can be inferred. We briefly review such issues, highlighting in particular how they relate with Knowledge Representation. This is by no means an exhaustive discussion, and the literature lacks a proper systematization of the field (for a recent attempt along this direction, the reader can refer to [22]).

Fetching the events is the *conditio sine qua non* for monitoring. To make monitoring applicable, the information system on which the monitoring framework is applied must be able to collect (and log) the relevant events occurring in the system, so as to construct a representation of the actual system execution that is as accurate as possible. This is typically difficult for those settings, like healthcare, in which part of the work is carried out by human actors in the physical world, without a direct computerized support. While this means, in general, that the logged trace of the system is only an incomplete representation of the real one, the presence of a model of the system makes it possible to exploit automated reasoning techniques so as to (partially) reconstruct the missing information. An example of the usage of KR techniques in this respect is [5].

Represent and process the events: Orthogonally to how the events are collected by the information system, there is the issue of how these events are represented. In fact, real-worlds events are typically heterogeneous both for what concerns their attached data, and their level of abstraction. As a simple example, consider the problem of monitoring the everyday life of a patient. In this setting, an event type may refer to the measurement of a (continuous) patient's vital parameter (such as the blood pressure), whereas another event type could denote that the patient took some medicine at a given time. This heterogeneity is depicted in Fig. 5.2. How signals/sub-symbolic events, as well as abstract/high-level symbolic events, are represented and related to each other is a central, longstanding problem of KR. It is worth noting that these two types of events could be reduced to each other before being processed by the monitoring system. On the one hand, symbolic events may be *reduced* to signals (e.g., through serialization into a binary stream). On the other hand, sub-symbolic events may be analyzed by an *activity recognition* module, so as to extract symbolic events by suitably correlating subsymbolic information.

Construct and represent the model: Similar to the problem of representing and processing the events is the representation and exploitation of the system model. This requires identifying the relevant aspects of the system to be modeled, trying to balance between expressiveness and tractability (which, as KR suggests, cannot be separated from "how" this model is used for reasoning). Since monitoring focuses on the system dynamics, a central aspect is obviously constituted by time. This motivated the extensive research on monitoring and runtime verification using variants of temporal logics (see, e.g., the *Runtime Verification* conference series[2]).

[2] http://runtime-verification.org/.

How events are modeled (e.g., atomic vs non-atomic events), whether time must be considered in a qualitative or quantitative fashion, which kind of data must be modeled, how to represent provenance information and event originators, are important questions for the KR community. Finally, beside how to represent the model, it is central to understand how such model is constructed. In particular, in addition to standard, top-down modeling, data/process/specification mining techniques can be exploited to semi-automatically extract model-level information from historical data about the system. An alternative approach is to transform existing models of a different type into a form amenable for use in monitoring [13, 14].

Perform online reasoning: While it is obvious that the reasoning techniques embedded by the monitoring framework depend on how events/models are represented, and on which task(s) are of interest (cf. Sect. 5.2), there are some specific key requirements that monitoring poses, no matter the application domain. First of all, since monitoring is an online, continuous activity, efficiency and *reactivity* are a must: the monitoring feedbacks must be produced in a timely fashion, and continuously revised taking into account newly incoming events. This calls, in turn, for *incremental* reasoning techniques, which exploit the previously computed results instead of recalculating everything from scratch [17]. A second specific aspect of monitoring is that it must provide *continuous support*, In particular, since the monitored system is in general not under the control of the monitor, the monitor must be able to follow the system evolution also in exceptional, unforeseen situations.

Calibrate the feedback: A final important aspect is how the monitoring outcome is represented, communicated to the actors responsible for the system execution, and possibly automatically exploited. This ranges from the extreme case in which the monitor is only able to observe the system and provide "symbolic" feedbacks to humans, to the one in which the monitor can (automatically or semi-automatically) supervise and adapt the real system depending on the monitoring outcome. An example of a system that automatically adapts to the patient is described in Chap. 7.

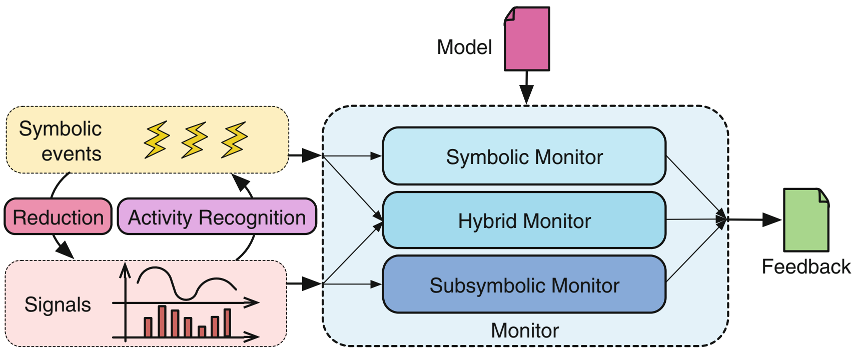

Fig. 5.2 Monitoring heterogeneous events

5.4 Monitoring in Healthcare

5.4.1 Where Monitoring Can Be Applied in Healthcare

The generic monitoring framework described in the previous section can be applied to a wide range on tasks in the healthcare setting.

- Monitoring patient signals
- Monitoring therapeutic interventions
- Monitoring disease progression
- Monitoring recovery
- Monitoring the execution of guidelines
- Monitoring alarms
- Combinations of the above

Traditionally, monitoring in the healthcare setting was carried out locally, but with the advent of telemedicine, remote patient monitoring is growing in popularity, mainly due to its convenience and effectiveness for both patients and clinicians.

It is important to emphasize that monitoring involves the observation of a system over time and reasoning with these observations. Much previous work on prediction and decision support in the healthcare sector was concerned with reasoning a particular point in time with a particular set of inputs. Examples include predicting morbidity, mortality and length of stay on admission [4, 8, 25]. The advantage of a continuous monitoring system is the monitor can learn about the individual patient and adapt to them in real-time.

5.4.2 Artificial Intelligence Techniques for Healthcare Monitoring

Artificial Intelligence techniques have been applied in the medical settings for many years. The ICU is particularly suited to the use of AI tools due to the wealth of available data and the opportunities for increased efficiencies [18]. The purpose of this section is not to present a complete history or ontology of AI applications in the ICU setting but rather to give a flavour of the variety of applications that have been proposed. The 2001 review by Hanson et al. [18] of artificial intelligence tools in the ICU provides a good overview. They conclude that neural networks and fuzzy systems are particularly useful for waveform analysis; fuzzy controllers can be integrated into bedside devices such as fluid and medication infusion devices, mechanical ventilators, and dialysis machines; Bayesian networks and neural networks can be used in the development of smart alarms; case-based reasoning, machine learning algorithms, and visualization tools can be used to analyse information from data warehouses describing the characteristic of an individual ICU.

Tools for predicting morbidity, mortality and length of stay have also been proposed. Barbini et al. [4] and Cevenini et al. [8] compared different models for predicting ICU morbidity following cardiac surgery. They found Bayesian classifiers and logistic regression models to be superior to an artificial neural network and the k-nearest neighbour classifier in terms of generalisation and calibration for this particular task. Ramon et al. [25] consider four different data mining algorithms for predicting 14 different tasks including probability of survival, length of stay in the ICU, probability of developing inflammation and the probability of developing kidney dysfunction. Their results for predicting probability of survival were better than the results obtained using the standard APACHE II score method [19] used by most ICUs. Using the APACHE II score the area under the ROC-curve (AUC) was 75 %. The best results were obtained using a Naive Bayes Classifier (AUC = 88 %). The AUC for Tree-Augmented Nave Bayesian networks was 86 % and for First Order Random Forests was 82 %. (First Order Random Forests are Random Forests in which the tests are first order logic queries [28].) Decision Trees provided the least promising results with AUC = 79 %. It is interesting to note that no one technique proved superior for all 14 tasks.

Other research includes Cismondi et al.'s fuzzy system for predicting the outcome of lab results [12], and the INTCARE system [23, 27] which combines data mining techniques and decision support systems to predict organ failure and suggest therapeutic treatment. The INTCARE research also addresses the need to distribute the application so it is available to doctors via mobile devices as well as in the ICU.

MIMIC II is a project undertaken by MIT, Philips Medical Systems and the Beth Israel Deaconess Medical Centre to develop and evaluate advanced ICU patient monitoring systems that will improve the efficiency, accuracy and timeliness of clinical decision making in intensive care. They aim to develop a research database from more than 30000 ICU patients. Their research includes estimating blood pressure and heart rate derived from the ABP waveform [20], using a Bayesian Network to estimate fluid requirements in the ICU [7], eliminating false alarms using classification trees and neural networks [29] and a decision tree to predict hypoglycaemia in intensive care patients [30].

A 2012 review by Bright et al. [6] shows the widespread interest across all medical fields in clinical decision-support systems. They examined 128 trials. From these they conclude that clinical decision support had a favourable effect on prescribing treatments, facilitating preventive care services, and ordering clinical studies across diverse venues and systems. They also stress the importance of delivering the right information to the right person in a timely manner.

5.4.3 Challenges in Healthcare Monitoring

Monitoring the physiological responses of a patient during recovery or in response to therapeutic interventions presents challenges. Substantial variability exists in the responses of different patients to medical interventions. This can be due to a variety of reasons including genetics, severity of illness and comorbidities. In addition an individual's response to events and interventions can fluctuate as their condition changes. These changes can be sudden or take place over a long time period. For a monitoring system to be truly useful it must be calibrated to the individual patient in real-time and adjust in real-time as the patient condition changes.

As mentioned in Sect. 5.4 human actors carry out events and/or manually record data in the Healthcare environment. These events may not be carried out at precisely the prescribed time, they may not be recorded in a timely manner nor with the precision that a monitoring system may require. Even automatically recorded information can be subject to measurement error. The monitoring system must be designed to deal with these uncertainties.

Healthcare is also an environment where disparate data may need to be integrated to provide the most complete picture. This is one area where monitoring systems can add huge value. A computational model can incorporate a much larger number of variables into its decision support process than a human can when making decisions unaided by a computer-based decision support system.

5.4.4 Probabilistic Monitoring Methods for Healthcare Monitoring

These challenges make healthcare monitoring particularly amenable to probabilistic approaches. Bayesian Networks (BNs) and Dynamic Bayesian Networks (DBNs) are used in the medical setting and here we note a few examples. An excellent introduction to the field is provided by Lucas et al. [21].

Aleks et al. [3] describe an application of DBNs to analysing ICU data. They demonstrated accurate detection and removal of artefacts in the arterial-line blood pressure sensor data. Charitos et al. [9] developed a DBN to successfully diagnose ventilator-associated pneumonia in ICU Patients. Research from the MIMIC II project mentioned in Sect. 5.4.2 includes applying a Kalman Filter to estimate blood pressure and heart rate derived from the ABP waveform in the presence of high levels of persistent noise and artefact [20], using a Bayesian Network to estimate fluid requirements in the ICU [7] and using a Bayesian network to model the cardiovascular system [26].

Techniques have been developed to learn both BN and DBN structure and parameters from data. More recently techniques for combining expert knowledge and automated learning for building BN structure [15, 24] have been published. However most models, including the ones noted here, are manually constructed using knowledge elicited from domain experts [11, 21]. This is a time consuming task [15, 21]. Knowledge elicitation therefore remains a bottleneck; this is clear in the paper of van Gerven et al. [16], for example, which provides an excellent demonstration of the steps required to build a DBN model for prognosis of carcinoid patients. In order to bypass this bottleneck, Enright et al. [13, 14] propose a methodology for automatically constructing DBNs from mathematical models since mathematical models can be considered to embody existing expert knowledge.

5.4.5 Characteristics of an Effective Monitoring System

In order for a monitoring system to be accepted and used in a healthcare setting it must be easily understood. A system that is understood by clinicians is more likely to be gain traction. This applies both to the internal workings of the monitoring system as well as the feedback provided to clinicians and patients. Feedback should be clear, concise and easily interpreted.

The monitoring system must be fully validated on realistic data. As mentioned in Sect. 5.4.3, data in healthcare is noisy. The monitoring system must therefore be designed to reason with incomplete and inaccurate data in a principled manner. It must also be thoroughly validated with this data.

The monitoring system must be suited to the task in hand. If monitoring a patient it must be individualised to that patient. That individualisation should be a continual process to adapt to ever changing patient conditions. If monitoring a process the system must be flexible to adapt to the variety of paths and outcomes.

Monitoring systems are of most value when they run in real-time. In order to achieve practical run-time speeds models must often be simplified. However a delicate balance is needed to avoid over-simplification and to preserve a practical model of the underlying system.

This trade-off between complexity and practicality also emerges when deciding what model parameters should be allowed vary. Models with a minimal number of parameters to be individualised work better when dealing with noisy-data [10].

5.5 Conclusion

We have reviewed the role of monitoring in the healthcare setting, delineating the main forms of operational support that monitoring can provide to healthcare professionals, and the main corresponding challenges from the point of view of knowledge representation.

The following two chapters in this book provide two notable, and quite diverse, examples of effective monitoring systems for healthcare, built by taking into consideration all the key characteristics enunciated in Sect. 5.4.5.

Chapter 7 describes a Probabilistic Real-time Intelligent Monitor (PRIM) that can be used for *Exploration, Detection, Prediction* and *Recommendation*. The chapter describes a methodology for *constructing and representing* the monitoring model and an efficient algorithm for *online reasoning*. The framework is designed specifically to address the challenges of the healthcare environment i.e. inaccurate and incomplete data, inter-patient variability and patient instability. The issues described in Sect. 5.3 in relation to a generic monitoring framework are addressed in a principled manner. The probabilistic approach adopted enables a reasoned method to handle incomplete data and represent events. The proposed approach for model construction involves exploiting existing models thus expediting and simplifying the model construction phase. The Adaptive Time Particle Filtering algorithm presented is an efficient mechanism for online Reasoning which in the context of this probabilistic monitoring system meets the requirements for incremental reasoning and continuous support in exceptional and unforeseen circumstances.

Chapter 5 describes a very different application of monitoring in a clinical environment. The chapter deals with monitoring conformance to clinical guidelines. The challenge explored is how to build a generic framework to *Detect* deviations. In this case *knowledge representation* is a huge challenge because, as will be explained, clinical guidelines encompass many different types of knowledge. Human actors execute clinical guidelines and it is this behaviour that is monitored. In this chapter we again see how this leads to inaccurate and incomplete data.

References

1. van der Aalst, W.M.P., et al.: Auditing 2.0: using process mining to support tomorrow's auditor. IEEE Comput. **43**(3), 90–93 (2010)
2. van der Aalst, W.M.P.: Process Mining: Discovery, Conformance and Enhancement. Springer, Heidelberg (2011)
3. Aleks, N., et al.: Probabilistic detection of short events, with application to critical care monitoring. In: Proceedings of NIPS 2008: 22nd Annual Conference on Neural Information Processing Systems. Vancouver, Canada, pp. 49–56 (2008)
4. Barbini, E., et al.: A comparative analysis of predictive models of morbidity in intensive care unit after cardiac surgery part I: model planning. BMC Med. Inf. Decis. Making **7**, 35 (2007)
5. Bertoli, P., Dragoni, M., Ghidini, C., Martufi, E., Nori, M., Pistore, M., Di Francescomarino, C.: Modeling and monitoring business process execution. In: Basu, S., Pautasso, C., Zhang, L., Fu, X. (eds.) ICSOC 2013. LNCS, vol. 8274, pp. 683–687. Springer, Heidelberg (2013)
6. Bright, T.J., et al.: Effect of clinical decision-support systems a systematic review. Ann. Intern. Med. **157**, 29–43 (2012)
7. Celi, L.A., et al.: An artificial intelligence tool to predict fluid requirement in the intensive care unit: a proof-of-concept study. Crit. Care **12**(6), R151 (2008)
8. Cevenini, G., et al.: A comparative analysis of predictive models of morbidity in intensive care unit after cardiac surgery part II: an illustrative example. BMC Med. Inform. Decis. Making **7**, 36 (2007)
9. Charitos, T., et al.: A dynamic bayesian network for diagnosing ventilator-associated pneumonia in ICU patients. Expert Syst. Appl. **36**(2), 1249–1258 (2009)
10. Chase, J.G., et al.: Physiological modeling, tight glycemic control, and the ICU clinician: what are models and how can they affect practice? Ann. Intensive Care **1**, 11 (2011)
11. Chatterjee, S., Russell, S.: Why are DBNs sparse? In: International Conference on Artificial Intelligence and Statistics, Sardinia, pp. 81–88 (2010)
12. Cismondi, F.C., et al.: Reducing ICU blood draws with artificial intelligence. Crit. Care **16**, 436 (2012)
13. Enright, C.G., Madden, M.G., Madden, N.: Bayesian networks for mathematical models: techniques for automatic construction and efficient inference. Int. J. Approximate Reasoning **54**, 323–342 (2013)
14. Enright, C.G., Madden, M.G., Madden, N., Laffey, J.G.: Clinical time series data analysis using mathematical models and DBNs. In: Peleg, M., Lavrač, N., Combi, C. (eds.) AIME 2011. LNCS, vol. 6747, pp. 159–168. Springer, Heidelberg (2011)
15. Flores, J.M., et al.: Incorporating expert knowledge when learning bayesian network structure: a medical case study. Artif. Intell. Med. **53**(3), 181–204 (2011)

16. van Gerven, M.A.J., Taal, B.G., Lucas, P.J.F.: Dynamic bayesian networks as prognostic models for clinical patient management. J. Biomed. Inform. **41**(4), 515–529 (2008)
17. Ghezzi, C.: Evolution, adaptation, and the quest for incrementality. In: Calinescu, R., Garlan, D. (eds.) Monterey Workshop 2012. LNCS, vol. 7539, pp. 369–379. Springer, Heidelberg (2012)
18. Hanson III, C.W., Marshall, B.E.: Artificial intelligence applications in the intensive care unit. Crit. Care Med. **29**, 427 (2001)
19. Knaus, W.A., et al.: APACHE II: a severity of disease classification system. Crit. Care Med. **13**, 818–829 (1985)
20. Li, Q., Mark, R.G., Clifford, G.D.: Artificial arterial blood pressure artifact models and an evaluation of a robust blood pressure and heart rate estimator. BioMed. Eng. Online **8**(1), 13 (2009)
21. Lucas, P.J.F., van der Gaag, L.C., Abu-Hanna, A.: Bayesian networks in biomedicine and health-care. Artif. Intell. Med. **30**(3), 201–214 (2004)
22. Ly, L.T., et al.: A framework for the systematic comparison and evaluation of compliance monitoring approaches. In: Gasevic, D., Hatala, M., Motahari Nezhad, H.R., Reichert, M. (eds.) Proceedings of the 17th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2013), pp. 7–16. IEEE (2013)
23. Portela, F., et al.: Knowledge discovery for pervasive and real-time intelligent decision support in intensive care medicine, 2011. Fundao para a Cincia e a Tecnologia (FCT) - PTDC/EIA/72819/ 2006, SFRH/BD/70156/2010
24. Radstake, N., Lucas, P.J.F., Velikova, M., Samulski, M.: Critiquing knowledge representation in medical image interpretation using structure learning. In: Riaño, D., ten Teije, A., Miksch, S., Peleg, M. (eds.) KR4HC 2010. LNCS, vol. 6512, pp. 56–69. Springer, Heidelberg (2011)
25. Ramon, J., et al.: Mining data from intensive care patients. Adv. Eng. Inform. **21**, 243–256 (2007)
26. Roberts, J.M., et al.: Bayesian networks for cardiovascular monitoring. In: Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 205–209. IEEE Engineering in Medicine and Biology Society (2006)
27. Santos, M.F., Portela, F., Vilas-Boas, M.: INTCARE: multi-agent approach for real-time intelligent decision support in intensive medicine. Fundao para a Cincia e a Tecnologia (FCT) (2011)
28. Vens, C., Van Assche, A., Blockeel, H., Džeroski, S.: First order random forests with complex aggregates. In: Camacho, R., King, R., Srinivasan, A. (eds.) ILP 2004. LNCS (LNAI), vol. 3194, pp. 323–340. Springer, Heidelberg (2004)
29. Zhang, Y., Szolovits, P.: Patient-specific learning in real time for adaptive monitoring in critical care. J. Biomed. Inform. **41**(3), 452–460 (2008)
30. Zhang, Y.: Predicting occurrences of acute hypoglycemia during insulin therapy in the intensive care unit. In: 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. EMBS 2008, pp. 3297–3300 (2008)