Reflecting on Evidence-Based Timelines

Elizabeth Bjarnason, Lund University

Anne Hess, Fraunhofer Institute for Experimental Software Engineering

Richard Berntsson Svensson, Chalmers | University of Gothenburg

Björn Regnell, Lund University

Joerg Doerr, Fraunhofer Institute for Experimental Software Engineering

Consider the latest project you worked on: did the project team take the time to collectively reflect on the experience once the project was completed? If so, was it a constructive group discussion or merely a feel-good event?

In fact, project retrospectives, also called post-mortem reviews or lessons learnt, can be a powerful tool for organisational learning and process improvement. But once a project is completed it is often swiftly replaced by another. The project participants rarely take the time to sit down together to discuss how it went, and what and how to do better in the future. And, as time passes the details and sequence of events are soon forgotten. This is one of the problems with reflecting after the fact. Our memory of an event is also affected by how we felt and what we experienced at that point in time. When this happens, a project retrospective can turn into an emotional venting session rather than a constructive discussion on how to improve on practice.

Also, when a retrospective relies solely on peoples’ experiences of events there is a risk of drawing incorrect conclusions since only part of the picture is considered. In order to support projects in remembering events and performing a fact-based retrospective we developed the EBTR (Evidence-Based Timeline Retrospective) method.

Evidence-Based Timelines

Project retrospectives can be improved by providing the participants with a timeline of project events. This visualisation of a project’s history provides a memory prompt and a focal point for participants to collectively discuss past events. In EBTR, timelines containing evidence of project events are planned and prepared in advance. The displayed objective information, or evidence, is gathered from available project sources including documents and repositories used for planning, requirements, issues reports etc. Team members then share perspectives around such timelines at a retrospective meeting. Through these group reflections, the participants can gain new insights into the entire project life-cycle. Project members can also identify good practices and improvements to practice.

An important factor when designing EBTR was to minimize the cost for participating teams. So the project members only need to attend and actively participate in the retrospective meeting. A facilitator performs all the necessary preparations including summing-up the outcome.
EBTR

Evidence-Based Timeline Retrospective (EBTR) is a method that can support software development organisations in improving their processes and practices through project retrospectives. Project teams can gain new insights into needed improvements by collectively reflecting on project history visualised by timelines. The method is based on the timeline exercise described by Norman Kerth. While Kerth’s exercise relies on people’s memory of events, the EBTR timelines are evidence-based and consist of project data. This objective data is used as evidence of a project’s history. (The term ‘evidence’ is here used differently than, e.g. within evidence-based medicine and software engineering where it refers to scientifically validated ‘truth’.)

EBTR is applied in four main steps: preparations, timeline construction, retrospective meeting and validation of the outcome, see Figure 1. These will now be described at the general level. The method also has six variation points that can be used to tailor the method to a specific case. For example, by leading semi-structured discussions a wider range of topics are covered and participants gain wider and more new insights. Details of how to apply EBTR are provided in the section Reflecting on Development Projects. More detailed guidelines are available online at http://serg.cs.lth.se/research/experiment_packages/ebtr/

Step 1: Preparations The person or function responsible for a process and/or its implementation can initiate an EBTR for a certain purpose and identify a facilitator to manage the work. Through discussions with this responsible person, the facilitator defines goals and aspects on which to focus the EBTR. The goals are a variation point that can be set to target a specific area or a wider set of issues. Based on the goals and aspects, a set of focus questions are derived and the type and source of evidence to display in the timelines are determined. The focus questions are later used to guide the retrospective meeting in step 3. The projects for which EBTRs are to be performed are also agreed in this initial step.

Step 2: Timeline construction Time-stamped data, or evidence, to display in the timelines are extracted from its sources. The content and visualisation of the timelines is a variation point that is tailored for each case and that depends on the type and amount of available project information and data sources. Evidence can also be planned to be stored during the project execution for an EBTR. The effort and tools required to manage the extracted evidence will vary. Small amounts of data from, for example, status reports can be collected and managed manually. Larger amounts of data from, for example, issue management systems with hundreds of entries require a systematic approach and tool support. Even with relatively modest sets of data, drawing and updating the timelines manually

Figure 1. Overview of the 4 main steps of the EBTR method.
quickly becomes cumbersome and time consuming. For the timelines to be effective in supporting the retrospective meetings the visualisation of project history needs to be clear and uncluttered, and easily interpreted without excessive training in how to interpret them.

**Step 3: Retrospective meeting** The core EBTR activity is the retrospective meeting where the project team gathers to reflect on the project. All key roles involved throughout the project are invited by the facilitator. Throughout the meeting the facilitator leads and moderates the discussions to focus on the intended goals and aspects. If a vital role can’t attend the meeting the facilitator could have a separate meeting with this person although this is not ideal.

The meeting consists of an introduction, a warm-up exercise, the reflective discussion and summarising of the meeting outcome. The introductory part includes a walk-through of the timelines and the applied visualisation techniques. This part can be given prior to the retrospective meeting. The project members will then be prepared and time is saved at the retrospective.

To familiarise the participants with the timeline they can be asked to perform a warm-up exercise. An exercise that has worked well is for each participant to mark their active involvement on the timeline and to add other people that have worked on the project. This can reveal hand-over points and previous team members. The exercise also encourages the participants to relate to and interact with the timeline. This is an important factor in enabling the timeline to act as a focal point and a trigger for the discussions that follow.

The project members reflect on events together by sharing experiences from their different viewpoints. These discussions are guided by the focus questions and by the evidence provided in the timelines. The degree of structure of discussions can be varied. A more structured format can encourage all participants to actively participate. If the discussions dry up there are a set of prompting questions which can be used to revitalise and stimulate further reflections.

The final part of the meeting is spent on identifying the main findings including suggested improvements. The facilitator elicits these findings with a set of pre-defined categories such as what we learnt and what to do differently. The facilitator documents these findings and the discussions in an EBTR summary.

**Step 4: Validation** In this final step the retrospective outcome is validated with the project members. The extent of this validation can vary, but should at least include distributing updated timelines and the EBTR summary. The project members then confirm, correct or complement these. Additional follow-up meetings can also be held to further communicate the outcome to the project members and the initiating EBTR responsible. Decisions concerning improvements such as new practices or enhancing existing processes are ideally discussed at such follow-up meetings.

**Reflecting on development projects**

EBTR has facilitated productive project retrospectives for software projects in two development contexts and domains. We have applied the method to one large and one small-to-medium sized case. The projects in our two case studies develop telecommunications software vs. web-based business applications. The specific retrospective goals for each case were met by tailoring EBTR to match these.

**Telecom projects**

We designed and evaluated the EBTR method in close collaboration with a large company in the telecommunications domain. We performed EBTR retrospectives for three of their agile development projects that delivered to a large product project. The three projects consisted of 4 to 9 members and had a lead time of 13 to 28 months. The cross-functional team of each project was responsible for all aspects of development of new functionality. The teams frequently held sprint, or iteration retrospectives but rarely conducted these for the whole project. The teams then risked losing sight of the overall picture and making short-term decisions concerning their practices. By introducing EBTR, the goal was to increase the engineers’ insight into how requirements were defined and communicated in the overall context (step 1). The aspects included in the EBTR (step 1) covered...
people involved in the project, artefacts, decisions, planning and project state. The facilitator had long experience of
the company and its development process, but not of the specific projects.

The EBTR facilitator generated timelines (step 2) by extracting large amounts of information from several
sources including systems for issue management, scope management and planning. The extracted data was stored
and categories per aspect in MS Excel. In order to provide a good visual overview one timeline was used for each
aspect. The timelines were constructed in MS Visio and printed on large sheets of paper that were displayed on the
meeting room wall. An example is shown in Figure 2.

The timelines acted as a natural focal point and attracted people’s attention as soon as they entered the room.
The facilitator invited the project members to the 90-minute EBTR meetings (step 3). At the start of the EBTR
meeting they were given an overview of the timelines and the information visualised in them. The project team then
reflected on the project through an open discussion around the timelines.

The participants interacted with the timeline through pointing at events, identified connections between events
and added missing information. The facilitator used the defined focus questions as a checklist rather than as an
agenda. This resulted in covering a wide set of issues within the targeted area ranging from delays in deciding to
implement market-critical features to difficulties in planning technically complex features.

The IBIS project

We also applied EBTR in a research project called IBIS (www.ibis-projekt.de). This project developed the IBIS
method\(^8\) for supporting SMEs (small and medium sized enterprises) in designing intuitive-to-use software products.
Academic researchers worked in this project together with product managers and software engineers from two
SMEs. In total, 11 people participated in the joint project including requirements engineers, project managers,
developers and testers. We applied EBTR to a 7-month period during which the IBIS method was evaluated. One of
the participating researchers acted as the EBTR facilitator. Our goal (step 1) was to assess the IBIS method
application and the research-industry collaboration within the project concerning communication and workload.

The facilitator constructed the timeline (step 2) based on evidence for three aspects: activities, events and
artefacts including planned vs. actual delivery dates. The evidence was provided through questionnaires filled out
by the software engineers during the project. The facilitator visualised this evidence using one timeline drawn on
flip chart papers and cards that were colour-coded per aspect. The right-hand side of Figure 2 contains a snapshot of
the used timeline.

Figure 2. Examples of evidence-based timelines from one of the three telecom
projects (left) and from the IBIS project (right).
At the 4-hour retrospective meeting (step 3) the participants sat around the timeline that was placed on a large table. The facilitator opened the meeting with a short introduction of the goals, the agenda and the timeline including its visualisation scheme. The facilitator then led a structured, but open discussion, topic by topic using the focus questions. The participants considered these questions individually and noted their reflections on post-it notes. Each participant then presented their notes and added them to the timeline. The group commented on the presented notes and discussed reasons for the issues and possible relationships to other issues.

**Benefits of EBTR**

We found that there were several benefits of applying EBTR to the two cases and describe the most important ones here.

**Fuelling reflective discussions**

The evidence-based timelines fuelled discussions at the retrospective meetings. The timelines helped people to remember and triggered them to start discussing past events. One participant said: ‘It reminds you of what actually happened.’ Another one expressed that ‘It would have been difficult to just start talking based on nothing. It was a long time since we did this.’ Physical interaction with the timeline stimulated the participants and kept them active throughout the meeting. They pointed to discussed events in the timelines or added them, if they were missing. The participants also jointly visualised their degree of activity by each adding their activity line to the timeline. By keeping people alert and interested valuable reflections were made in a short and focused period of time.

**Providing a project overview**

The timelines provided a good overview of the entire project by summarizing activities, events, involved roles, etc. The participants could ‘see at a glance what we have achieved in the project.’ This is one of the strengths of the approach compared to retrospectives without evidence-based timelines. One participant said, when comparing to previous retrospectives: ‘you get a much better overview with this timeline.’ One manager even suggested that the timeline approach could be used to visualise and communicate progress throughout the project.

**Identifying connections**

The overview provided by the timelines helped to identify connections between events. Through pointing to an event in the timeline, potential causes or effects could be considered by looking at the previous or following events. For example, for the IBIS project some late scope changes were found to be caused by a lack of communication at the start of the project. By uncovering this connection the IBIS method was improved to avoid such problems.

**Seeing the big picture**

The team members gained new insights of the overall picture through participating in the retrospective. The timelines and the group reflections provided an extended perspective of their work as part of a larger context. For example, one participant became aware of the spent effort and achieved outcome of the project. The developers and testers in particular appreciated seeing the big picture including project activities in which they were not actively involved. ‘It is interesting to see a compilation of the big picture. As a developer, I don’t see even half of this.’ This insight can prevent communication gaps in the future and improve on the coordination of different activities.
Identifying improvements

The project members found it very useful to reflect on and discuss aspects of the project that went well and things that could be improved. The participants gained new insights into work practices and how things like company strategy could impact the scoping and detailed requirements for a project. The EBTR helped them to summarise their experiences and identify improvements. This could boost and motivate people and as one project manager said: ‘it is a good reminder that can help us when we start on the next bigger project.’ For the IBIS project, the EBTR summary even provided valuable feedback to the customers and was included in the final project report to the project funders.

Key Lessons Learnt

We gained many insights into how to perform EBTRs through case studies including four different projects. Five of the most important lessons learnt are shared here.

Introduce the timeline prior to the retrospective

The participants from the telecom projects initially found it hard to understand and use the timelines. Even though they quickly caught on it did lead to a slow start of the retrospective meeting.

The retrospectives will be quicker off the ground if the participants are given a 10-minute introduction prior to the meeting. For example, the timeline concept can be presented at a project or team meeting. The actual timelines could also be distributed in advance. This has the added advantages of motivating the project members to participate and to introduce them to the facilitator. Both of these factors can support a more open and constructive retrospective discussion.

Organise the meeting room around the timeline

At the first retrospective meeting the timelines were posted in a corner of the room that wasn’t immediately visible to the participants. The project members also placed themselves around a table at the other end of the room. This physical distance and placement made it hard to focus the discussions on the timeline and the participants didn’t directly interact with the timeline.

At the following retrospective meetings we consciously organised the room to encourage the participants to look at and interact with the timelines. A central location was chosen for posting the timelines and chairs for the participants were placed around it. This resulted in more open and free flowing discussions. The participants also interacted with the timelines more frequently, both by pointing to events and adding new information to the timeline.

Keep the retrospective short and focused

We have varied the length of our retrospectives between 75 minutes and 3 hours. We found that a longer meeting doesn’t lead to more insights or findings. It is also hard to keep a constructive discussion going for more than 60-90 minutes. Everyone gets tired and finds it hard to concentrate. But, for the 75 minute slots we found that we had too little time to conclude and summarise the meeting.

The meeting time can be optimised by keeping the discussions focused towards the goals and aspects defined for the EBTR. The facilitator plays a key role in this and should have good knowledge of the domain and the applied development processes. The focus questions help to keep the discussions within the topic area. The time spent on each question should be monitored and kept within a rough time budget. This will ensure that there is time at the end to summarise the findings. The facilitator can also decrease the discussion time by selecting a sub-set of the topics or by presenting the topic area rather than the detailed focus questions.
Tailor EBTR to the purpose and project

One flavour of EBTR doesn’t necessarily suit all projects and organisations. Before performing the retrospectives, we needed to tailor the method to be a good match for each of the project contexts. This included adapting EBTR to the specific purpose of applying the method for each case.

In the preparation step, the purpose of applying the method affects the definition of the goals, aspects and evidence. A wide and general assessment goal is achieved by reflecting on a broader picture supported by presenting a wide range of evidence, or information. This was the case for the telecom projects where increased insight into requirements within the development process was the target. In contrast, reflecting on less evidence will enable a more focused discussion and enable meeting a more specific EBTR goal. The specific EBTR goal of evaluating the IBIS method is an example of this.

The discussion structure at the retrospective meeting can also be tailored to match the width of the EBTR goal. A more structured discussion around the focus questions will focus the reflections on the specific goal. In contrast, using the focus questions more as a check list will lead to discussions of a wider set of topics and more reflections of the big picture.

Finally, the facilitator’s relationship to the project is another variation point to consider. For the IBIS project, a project member acted as the EBTR facilitator. As such the facilitator had insight into the specific events and history of the project. This turned out to be well matched to the EBTR goal of assessing specific issues. The facilitator was well equipped to lead a detailed discussion of the targeted topics. In contrast, if the EBTR goal is a wider assessment of an area we recommend using a facilitator with no previous relationship to the project. This will encourage wider discussions and reflections of the bigger picture with the project team.

Don’t overload the timeline

It was a challenge to visualise the project data in a clear and uncluttered way. This was experienced for both cases but in different ways. For the IBIS project, cards and post-it notes on flipchart papers were used to display the moderate amounts of data. Initially at the meeting this approach worked well. But as the meeting progressed the flipchart timelines became hard to read as the amount of cards and post-it notes increased. From this we learnt that the initial timeline is best prepared digitally and then printed. Additional events and information can then be added to this in the form of cards or post-it notes during the retrospective.

In contrast the huge amounts of data for the telecom projects posed a challenge already in the timeline construction. We choose a combination of MS Excel for storing and sorting the data and MS Visio for visualising it. We also found that it was necessary to separate between different types of evidence. This can be done by using visual symbols for different types of data and by placing the evidence for each aspect on individual timelines. For example, information of people involved in the project was displayed on one timeline and project planning information on another one.

Next steps

The response we have received from the retrospective participants encourages us to continue evolving and improving on EBTR. One extension to investigate is how to apply EBTRs to larger projects with more members than can attend a common retrospective meeting. This also requires more powerful tools for constructing timelines with huge amounts of data. Another avenue to explore is integrating the EBTR approach for agile and iterative development. For example, constructing the timeline for each sprint or iteration, and use it at the iteration retrospectives. The project members would then become more familiar with the timelines as the project progresses. The team could contemplate the wider picture for each iteration and ultimately reflect on the full project history.
We encourage practitioners to consider how EBTR can be tailored to their needs and projects. We believe that our reported experiences from applying the method can encourage more development projects to learn and improve by reflecting on evidence-based timelines.

**Acknowledgements**

We thank all EBTR participants. This work was partly funded by EASE (http://ease.cs.lth.se) and by the German Federal Ministry of Education and Research under project number 01IS11017.

**References**

Bibliographies

Dr Elizabeth Bjarnason is associate senior lecturer at Lund University’s Department of Computer Science. Her research interests include communication and coordination of requirements engineering within software development. She received her PhD in Software Engineering from Lund University. Contact her at Dept. of Computer Science, Lund Univ., Box 118, SE-221 00 Lund, Sweden; elizabeth@cs.lth.se

Anne Hess is a researcher at Fraunhofer Institute for Experimental Software Engineering. Her research interests lie in user-centered requirements engineering for complex and innovative information systems. She received a German Diplom degree in Computer Science from the Saarland University of Saarbruecken, Germany. Contact her at Fraunhofer IESE, Dept. of Information Systems Development, Fraunhofer Platz 1, DE-67663 Kaiserslautern, Germany; anne.hess@iese.fraunhofer.de

Dr. Richard Berntsson Svensson is senior lecturer in the Software Engineering division, Department of Computer Science and Engineering at Chalmers and University of Gothenburg. His research interests include quality requirements, software product management, empirical software engineering, and creativity and innovation. He received his PhD degree in Software Engineering from Lund University, Sweden. Contact him at Chalmers University of Technology, Department of Computer Science and Engineering, Software Engineering division, SE-412 96, Gothenburg, Sweden; richard@cse.gu.se

Dr. Björn Regnell is professor at Lund University’s Department of Computer Science and vice dean of Research at the Faculty of Engineering, LTH. His research interests include requirements engineering, software quality, software innovation, software product management, and empirical research methods. He received his PhD degree in Software Engineering from Lund University, Sweden. Contact him at Dept. of Computer Science, Lund Univ., Box 118, SE-221 00 Lund, Sweden; bjorn.regnell@cs.lth.se

Dr. Joerg Doerr is head of the Information Systems division at the Fraunhofer Institute for Experimental Software Engineering. His work areas include requirements engineering with a focus on non-functional requirements. He received his PhD in Computer Science from the University of Kaiserslautern, Germany. Contact him at Fraunhofer IESE, Fraunhofer-Platz 1, DE-67663 Kaiserslautern, Germany; joerg.doerr@iese.fraunhofer.de