

**Carnegie Mellon University**

---

**From the Selected Works of Cécile Péraire**

---

2014

# State-Based Monitoring and Goal-Driven Project Steering: Field Study of the SEMAT Essence Framework

Cécile Péraire  
Todd Sedano



Available at: [https://works.bepress.com/cecile\\_peraire/1/](https://works.bepress.com/cecile_peraire/1/)

# State-based Monitoring and Goal-driven Project Steering: Field Study of the SEMAT Essence Framework

Cécile Péraire  
Carnegie Mellon University  
Silicon Valley Campus  
Moffett Field, CA 94035  
011-650-335-2851  
Cecile.Peraire@sv.cmu.edu

Todd Sedano  
Carnegie Mellon University  
Silicon Valley Campus  
Moffett Field, CA 94035  
011-650-335-2812  
Todd.Sedano@sv.cmu.edu

## ABSTRACT

At Carnegie Mellon University in Silicon Valley, the graduate master program ends with a practicum project during which students serve as software engineering consultants for an industry client. In this context, students are challenged to demonstrate their ability to work on self-managing and self-organizing teams. This paper presents a field study of the Software Engineering Method and Theory (SEMAT) Essence framework. The objective is to evaluate the effectiveness of the Essence's novel state-based monitoring and goal-driven steering approach provided by the Essence kernel alphas and their states. The researchers conducted the study on seven graduate master student teams applying the approach throughout their practicum projects. The research methodology involves weekly observation and recording of each team's state progression and collecting students' reflection on the application of the approach. The main result validates that the approach provides student teams with a holistic, lightweight, non-prescriptive and method-agnostic way to monitor progress and steer projects, as well as an effective structure for team reflection and risk management. The paper also validates that the Essence kernel provides an effective mechanism for monitoring and steering work common to most student software projects. This includes the work done during project initiation as well as the work done at the project or release level. Support for technical work should come from additional practices added on top of the kernel, or by extending or altering the kernel definition. The conclusion is that the approach enables students to learn to steer projects effectively by addressing the various dimensions of software engineering. Hence the approach could be leveraged in software engineering education.

## Categories and Subject Descriptors

D.2.9 [Software Engineering]: Management  
K.6.3 [Management of Computing and Information Systems]:  
Software Management - software development, software process  
K.3.3 [Computers and Education]: Computer and Information  
Science Education - curriculum

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ICSE'14, May 31 – June 7, 2014, Hyderabad, India.

Copyright 2014 ACM 978-1-4503-2768-8/14/05... \$15.00.

## General Terms

Management, Measurement, Experimentation.

## Keywords

Project monitoring and steering, software process improvement, software development methods, team development, field study.

## 1. INTRODUCTION

This paper presents the results of a field study of the Software Engineering Method and Theory (SEMAT) Essence framework [8, 9] investigating the Essence kernels' novel approach to monitoring and steering software development projects. One of the promises of the approach lies in its potential ability to monitor any type of project holistically based on universal project states and steer these projects based on universal goals, hence making the monitoring and steering mechanisms independent from the method (such as Scrum [13] and XP [3]) or set of practices adopted by the project team.

We are interested in understanding the strengths and weaknesses of Essence by gaining experience of using the approach on real projects. We conducted a field study involving seven co-located and distributed student teams working on industrial projects. These students finish their graduate program with a project course at Carnegie Mellon University in Silicon Valley in the context of the Master of Science in Software Engineering program. During the project each student has the opportunity to apply the software engineering skills acquired throughout the program to solve a real industry problem. Monitoring and steering projects with the Essence kernel allows the researchers to identify where value is added for an educational program. Student team projects serve as a possible metaphor for newly formed industry team projects; the value added for a student team could transfer to an industry team.

As faculty, we allow teams to be self-organizing and responsible for their decisions, yet at the same time expect them to incorporate generally accepted software engineering practices, and demonstrate the ability to effectively steer their project. Each team manages its own project with minimal faculty supervision. In the past, the faculty observed that with minimal supervision some teams revert to old habits [2]. As soon as the starting bell sounds, they can act as racetrack horses running towards the finish line with blinders, ignoring what they have learned in class and without concerns for software engineering discipline. The new freedom and the chance to write code for a client lead them to focus mostly on implementation and therefore to adopting a suboptimal way of working.

Our research hypothesis contends that Essence’s monitoring and steering approach provides a lightweight framework for students to look at their project holistically, helping them to address various project dimensions beyond implementation. We stipulate that the framework acts as a routine reminder about applicable software engineering practices covered in previous courses, and this without being prescriptive and while being method agnostic. For instance, by using this technique, we expect students to think about involving stakeholders, improving the team’s way of working, and demonstrating that the software system has the desired quality characteristics.

This paper reviews SEMAT’s Essence framework and kernel, describes the field study planning and execution, reports on the analysis and interpretation of the field study data, recommends effective practices for introduction of Essence to a software engineering curriculum, and summarizes the conclusions.

## 2. SEMAT ESSENCE OVERVIEW

In 2009, Ivar Jacobson, Bertrand Meyer, and Richard Soley started work on the “Software Engineering Method and Theory” (SEMAT) with the goal of “re-founding software engineering as a rigorous discipline” [7]. While many software engineering principles, techniques, practices and methods exist, the SEMAT founders’ ambition is to create a general theory of software engineering and a unifying process framework. Out of their initiative has emerged the SEMAT Essence language and kernel, which became an Object Management Group (OMG) beta standard in 2013.

The core idea of Essence is that software projects exhibit universal behavior and transition through identifiable states as they progress. Essence groups these states together by different software engineering aspects or dimensions called “alphas.” Essence identifies seven alphas as core to every software engineering project: **Stakeholders, Opportunity, Requirements, Software System, Team, Way of Working, and Work**. These seven alphas serve as the Essence kernel as illustrated in Figure 1.

Each alpha progresses through a number of states during the project lifecycle. For example, the **Stakeholders** alpha progresses through the states *Recognized, Represented, Involved, In Agreement, Satisfied for Deployment, and Satisfied in Use*. Each state includes a checklist to help determine if the project has achieved that state. Each checklist item provides a goal to be reached in order to progress to that state.

Despite the sequential definition of the states for each alpha, in practice, projects could fall back to previous states. Similarly, in some circumstances, it might make sense for a project to achieve a number of states simultaneously within a given alpha.

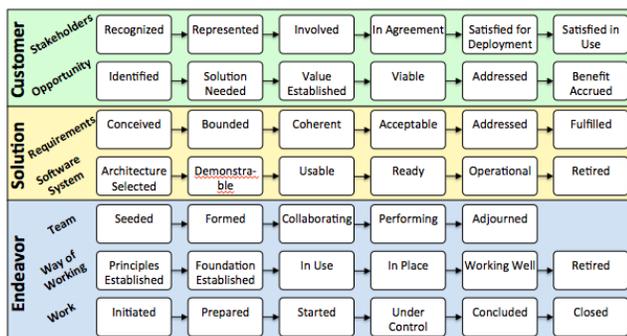


Figure 1: Essence kernel alphas and their states

The SEMAT vision is also to create a library of practices described using the Essence language and sitting on top of the Essence kernel. Practices would be composed to become specific methods addressing specific project or organization needs. Practices would help a team identify how to progress their project from one state to the next.

In The Essence of Software Engineering [9], the authors identify three separate applications of Essence: 1) project monitoring and steering, 2) determining when to green light projects, and 3) describing workflow through an organizational structure. This paper focuses mostly on the first application, *project monitoring and steering*, where a team assesses its current project state in each of the alphas and identifies possible actions to help transition from the current state to the next target state.

## 3. FIELD STUDY DESCRIPTION

The study focuses on understanding what value do project teams receive from following the SEMAT Essence’s monitoring and steering approach provided by the kernel alphas and their states. Essence was introduced to master students at the beginning of their practicum projects. The students had no prior knowledge of Essence.

Using the goal template from GQM [10], our research goal is to:

<b>Analyze</b>	SEMAT Essence’s monitoring and steering approach provided by the kernel alphas and their states
<b>for the purpose of</b>	evaluation
<b>with respect to its</b>	effectiveness
<b>from the point of view of the</b>	project team, educator, and researcher
<b>in the context of</b>	the software engineering practicum graduate course at Carnegie Mellon University.

This paper decomposes this goal into the following questions:

**Research Question 1:** Does the SEMAT Essence’s monitoring and steering approach provide value to the project team?

**Research Question 2:** How does the approach provide value to the project team?

**Research Question 3:** When in the project lifecycle does the approach add value?

**Research Question 4:** What are the limitations to the approach’s effectiveness?

The first section below describes the context of each project involved during the field study. The second section presents how Essence was introduced to the teams, either incrementally or using a workshop approach. The final section describes how the teams have been leveraging Essence during the remaining of their project.

### 3.1 Practicum Projects’ Context

The authors introduced Essence to seven student teams: three geographically distributed student teams, referred as Distributed-1 to Distributed-3 for the purpose of this paper, and four co-located student teams, referred as Co-located-1 to Co-located-4. As shown in Table 1, each team worked on creating or evolving a software solution for a different client, like an electric car fleet management system or a survivable social network. By design, the projects had a medium to high level of technical complexity, as they often involved multiple technologies or platforms or integrate with legacy systems.

**Table 1. Practicum projects' context**

Team Name	Industry Project	Team Size	Team Distribution	Average Work Experience	Technical Complexity
<b>First Pilot Projects</b>					
Distributed-1	Rendering of audio streams for accessibility purpose	3	Distributed within same time zone	10 years	Integration with legacy code on a single platform involving C, HTML5 and open-source technology.
Distributed-2	Access and preservation of electronic journals	4	Distributed across 2 time zones	6 years	Integration with legacy code on a single platform involving Java, MongoDB, Apache Jena and open-source technology.
Distributed-3	Survivable social network	4	Distributed across 2 time zones	8 years	Multi-platform involving Ruby on Rails, Javascript, jQuery Mobile. Embedded system constraints.
<b>Second Pilot Projects</b>					
Co-located-1	Electric vehicle fleet management	2	Co-located	3 years	Green-field development involving Query, Node.js and MongoDB. Integration with Rest APIs for two vehicle brands.
Co-located-2	Sonification of financial trading information	4	Co-located	3 years	Integrates with third party APIs (Yahoo! and Google Finance). Involves Objective C, Ruby on Rails, iOS, Google App Engine and Python. Requires financial knowledge. Has a special focus on user experience.
Co-located-3	Mobile performance testing	3	Co-located	4 years	Xcode, Instruments, Eclipse, iOS 6.0, Android 4.2, jQuery Mobile, HTML5, Benchmark.js, Appception, Pivotal tracker, Redmine, GitHub, RubyMine, Rails 3.2, Ruby 1.9.2. Heroku, Amazon EC2, HighCharts.js, CSS3, Cordova
Co-located-4	Virtual sensors definition and management	5	Co-located	5 years	Multi-platforms involving HTML5, Javascript and j2ee.

The geographically distributed, part time students were working professionals with an average of eight years of development experience. Their practicum projects ran for 15 weeks, during which each student dedicated about 20 hours per week to the project. They worked in small teams of three to four members distributed across one or two time zones.

The co-located, full time students were at the beginning of their career with an average of four years of development experience. Their practicum projects ran for 12 weeks, during which each student dedicated about 20 hours per week to the project. They worked in co-located teams of two to five members.

The course syllabus imposed a few milestones and deliverables (like roadmap, statement of work, reflection report, and working software), with potential additional requests coming from the client. Teams determined their own software development approach. Most students had a reasonable knowledge of a diverse set of generally accepted software engineering practices, and the ability to execute these practices somehow effectively. All projects adopted an iterative lifecycle.

Even though the student population was quite diverse in terms of origin and culture, by the time the students started their practicum project they were immersed in the North American culture for at least eight months.

Table 1 summarizes the context of each practicum project in terms of the following dimensions [1, 4, 12]: team size, team distribution, average work experience, and technical complexity.

Essence was introduced to the geographically distributed teams in the context of a first set of pilot projects, and to the co-located teams in the context of a second set of pilot projects.

## 3.2 Introducing Essence on Practicum Projects

### 3.2.1 First Pilot Projects – Incremental Approach

In the Spring 2013 semester, the authors introduced the SEMAT Essence framework to three geographically distributed student teams at the beginning of the project. Because first impressions are critical for adoption, and because the researchers were uncertain of the value provided by Essence, we made the decision of introducing the framework with minimum overhead to avoid adoption resistance and minimize potential waste.

We briefly introduced Essence to all students using a slide presentation of about 20 minutes. Since the teams were distributed, a set of physical Essence cards was sent to each student, and a digital Essence board was created using Google Drawing (see Figure 2). The digital board included one row for each of the seven alphas in the Kernel. Each row contained the sequence of states that the alpha progresses through during the project lifecycle.



**Figure 2. Digital Essence board**

After the initial presentation, the alphas were introduced incrementally over a number of 30 minute sessions, in order to make the experience as lightweight as possible. Each team met individually with a faculty and applied Essence on their practicum

project. As an example, here is what happened with one team over a one-month period:

**Session 1:** The two “customer” alphas, **Opportunity** and **Stakeholder**, were introduced. The main reason for introducing these two alphas first, was simply because it generally makes sense to have a discussion around the opportunity and stakeholders early in the project and before drilling down into the details of the solution and endeavor.

**Session 2:** After updating the progress made for the previously introduced alphas, one “solution” alpha, **Requirements**, was introduced. The rationale for introducing this alpha was based on a pain point, as the team was expressing concerns around the client’s expectations and hence needed to have a discussion around project scope and success criteria in relation with the **Requirements** alpha.

**Session 3:** After updating the progress made for previously introduced alphas, two “endeavor” alphas, **Team** and **Way of Working**, were introduced based on additional pain points. Indeed the team was having some communication issues and hence needed to have a discussion around team collaboration and way of working.

**Session 4:** After updating the progress made for previously introduced alphas, the remaining alphas were introduced for completeness: **Software System** and **Work**.

For each alpha, the team identified the current state, the target state, and any work items necessary to progress from the current to the target state. The identification of the current and target states was done through an informal discussion until the team reached an agreement.

To continue the example above, during the second session the team identified *Conceived* as the current state for the **Requirement** alpha and *Bounded* for the target state. Indeed all the items on the *Conceived* checklist were satisfied while a few items on the *Bounded* checklist were not satisfied. In order to achieve *Bounded*, the team first needed to define the project scope and clarify the success criteria with the client, as illustrated in Figure 3. The work items were dealt with right away or added to the team’s work item list or backlog.

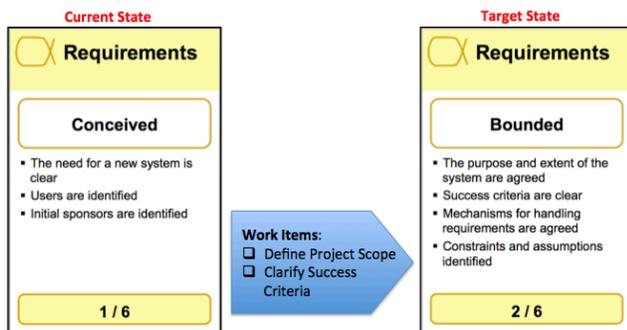


Figure 3. Work items to reach the *Bounded* state

With mentoring from faculty, Essence was introduced to the geographically distributed teams using the incremental approach described above. For each team, and following the initial presentation of the Essence framework, four sessions of about 30 minutes long were necessary in order to introduce all the seven alphas incrementally. During each session, elements relevant to our study were jointly logged by students and faculty as described in section 3.3 below.

### 3.2.2 Second Pilot Projects – Workshop Approach

In the Summer 2013 semester, Essence was introduced to four co-located teams at the beginning of the project. Based on our previous experience with the distributed students and armed with encouraging results (presented in Section 4 below), we decided to speed-up the introduction process using a workshop approach. Our goal was to help the teams benefit from Essence as early as possible in the lifecycle.

The workshop was time-boxed to 90 minutes and included all students. It consisted of a general introduction of the Essence framework, the motivation for adopting the framework, together with exercises teaching each team how to apply the Essence’s monitoring and steering approach on their own practicum project, one alpha at a time. Like in the first pilot projects and for the same reasons, the two green “customer” alphas, **Opportunity** and **Stakeholders**, were introduced first. Then, the remaining alphas (**Requirements**, **Software System**, **Team**, **Way of Working**, and **Work**) were introduced based on pain points when applicable, or in a random order otherwise. For each alpha, each team was tasked of identifying their project current state, target state, and any work items necessary to transition from the current to the target state. The identified work items were added to the team’s work item list or backlog.

A couple of changes were introduced compared to the first pilot projects:

- **Moving from physical cards to physical strips.** Handling a large set of cards could be a hassle. Therefore, we decided to create strips instead, as illustrated in Figure 4. For each alpha, one strip represents the typical sequence of states that the alpha progresses through during the project lifecycle. This way, we could easily provide the students with the information they needed to work on various alphas, one alpha at a time. Note that since the students were all physically present during the workshop, no digital boards were used.
- **Adoption of a poker game approach for state identification.** In order to remove anchoring bias, the informal way of identifying the current and target states was replaced with a “poker game” approach. In that context, each participant does a blind determination of their current state and all reveal their current state at the same time. In case of disagreement, the team discusses the different points of view until the participants reach an agreement. This technique is a simplification of Wideband Delphi [14] and agile estimation using Planning Poker [6], as the participants perform only one round of “estimation.” Just like Delphi and Planning Poker, the value is in the conversation, and allowing the team members to work through their different perspectives.



Figure 4. One strip per alpha

Using the workshop approach described above, and with mentoring from faculty, Essence was introduced to the co-located teams over a 90 minutes session. On average, 10 minutes were necessary for a team to cover one alpha, and most teams were able to cover the seven alphas during the workshop. Students and faculty jointly logged elements relevant to our study as described in the following section.

### 3.3 Using Essence on Practicum Projects

Once Essence was introduced, the teams were asked to continue leveraging the approach during the remainder of their project. Each team met on a regular basis (mostly weekly) for a 30 minutes session. During each session, the team reviewed most or all of the alphas. For each alpha, the team identified their project's current states following the poker game approach. The team would consider work items necessary to transition from the current state to the target state. Any new identified work items were added to the team's work item list or backlog. Teams were encouraged to act on their work items as soon as possible to accelerate their state progression.

A faculty member was present to facilitate each session. Faculty involvement was kept at a minimum to limit influencing the steering of the project. The faculty's role was constrained to guiding the team through the application of the Essence monitoring and steering approach, and to validating the objectivity of the team's self-assessment of their project state. By listening to the team's discussions, and asking clarification questions as needed, the faculty gauged the project state. At times, this helped reduce the tendency of some teams to be overly pessimistic or optimistic about the project state.

For the purpose of the field study, progress and work items were recorded jointly by the students and faculty using the teams' Essence progress log, as shown in Figure 5.

Alpha	Current State	Target State	Work Item / Notes
Stakeholders			
Opportunity			
Requirements			
Soft. System			
Team			
Way of Working			
Work			

Figure 5. Essence progress log

The distributed students mostly used their digital Essence board, while the co-located teams used both digital Essence board and strips. Some students kept their strips handy and used them throughout the project lifecycle while others preferred to rely on the digital board.

At the end of the projects a survey was sent to the students, including the following questions:

- Survey Question 1:** What did you like the most about Essence?
- Survey Question 2:** What did you like the least about Essence?
- Survey Question 3:** In using Essence, did you adopt a practice that you wouldn't have? Or did you adopt it earlier than you would have without using Essence? (Please explain)
- Survey Question 4:** Was following the Essence approach worth your time? (Please explain why or why not)
- Survey Question 5:** Would you use Essence on your next project?

(Please explain why or why not)

**Survey Question 6:** Anything else that we should know?

## 4. FIELD STUDY ANALYSIS & INTERPRETATION

Our research questions relate to the value provided by the SEMAT Essence's monitoring and steering approach. The researchers measured the qualitative value based on students' feedback collected mostly during the weekly SEMAT sessions, course reflection reports, and final survey. The researchers measured the quantitative value based on alpha state progression as well as the number of work items generated during the weekly SEMAT sessions and allowing bringing the project to a higher state. Refer to Table 3 in the Appendix for the raw data collected on alpha state progression, and to Figure 10 under Research Question 3 for the number of work items generated by the approach per week.

**Research Question 1:** Does the SEMAT Essence's monitoring and steering approach provide value to the project team?

Our field study shows that students receive value from SEMAT Essence's monitoring and steering approach provided by the kernel alphas and their states. In our survey, 90% of the students said that following the approach was worth their time (80% of the students participated in the survey.) Similarly, 80% said that they will use the approach on their next project.

By following the approach, project teams monitored their progression through the Essence states over time, as illustrated in Figure 6 for team Co-located-4. Every week this team generated on average 6 new work items during their SEMAT session, and then acted on these work items in order to bring the project to a higher state.

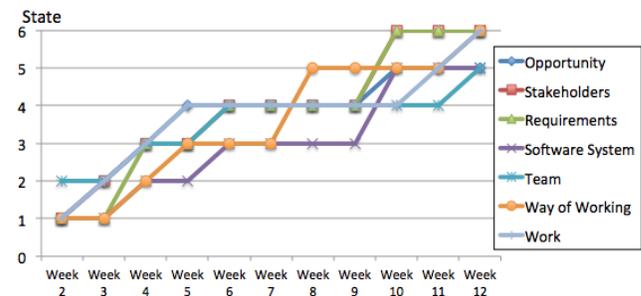


Figure 6. Alpha progression for team Co-located-4

In conclusion, most students have a positive perception of this approach as it helps them make decisions allowing to move their project forward.

**Research Question 2:** How does SEMAT Essence's monitoring and steering approach provide value to the project team?

The benefits that a project team receives from SEMAT Essence's monitoring and steering approach come primarily from the discussions that occur when the team is covering the various alphas. The discussions enable the team to pause and assess the situation:

- **The team steps back from its daily tasks and examines its project holistically.** One student noted: "Essence gives us a chance to step back and look at the project as a whole, from a bird's-eye view. There are aspects that we tend to ignore

when focused on the technology. Essence is very useful, because it makes me think about these particular aspects.” Similarly, another student noted: “I like the fact that Essence provides a structured way of thinking about critical aspects of the project at different stages of the project. Without Essence, our team could have overlooked some of these aspects.” Stepping back and looking at the project holistically is a healthy exercise providing the distance and perspectives needed to understand a situation, reflect, and make decisions. When asked about what they like the most about Essence, most students mentioned reflection or retrospective. For instance one student noted: “Though the team was holding retrospectives every week already, having Essence discussions be a part of it allowed the team to touch on important aspects of the project.”

- The team records progress accomplished and identifies what remains to be done.** When asked about what she liked the most about Essence, one student answered: “The choice of alphas: they seem to be exactly the right areas to monitor to promote the success of a software project.” The Essence mechanism for monitoring progress is illustrated in Figure 7. The chart shows the progress made by team Distributed-3 at week 5, compared to the practicum target state established by faculty. The team has been making good progress towards the target goal in most dimensions except **Software System** that lags at state 1 (“Architecture Selected”). This situation served as a red flag and a reminder that the team needed to focus its effort on taking their software system to the next level. The approach was used as a similar risk management mechanism in other instances. When asked if he would use Essence on his next project, one student answered: “Yes, it’s great for team reflection and risk management.”
- The team seeks guidance on what directions to take and identifies goals to transition to a higher project state.** Team Distributed-1 noted: “Essence gives us structure and direction.” Another student commented: “Essence is useful, as it gives you an agenda or checklist based on various dimensions (even though I was skeptical at first).” Essence provides a simple project steering mechanism. For each alpha, the identified target state provides the direction to take, and its checklist provides goals to reach. For instance and to continue the example above, during week 5 team Distributed-3 identified “Demonstrated” as its target state for **Software System**, with a number of goals including demonstrating the key architecture characteristics, as illustrated in Figure 7.

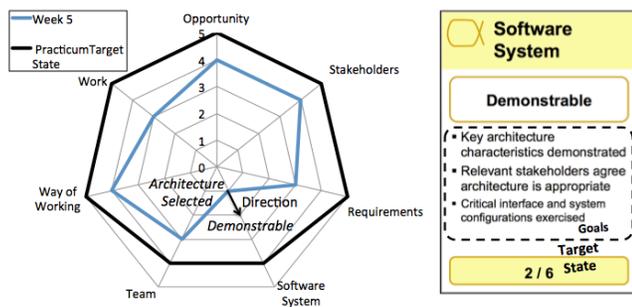


Figure 7. Project state for team Distributed-3 at week 5, with direction and goals for Software System

- The team decides what to do to reach the target goals.** Once the team identifies the goals, the team members rely on their software engineering knowledge and experience to decide how to reach these goals. Indeed, Essence does not prescribe the use of any existing method (like Scrum or XP) or set of practices or work items. Instead, the team has the flexibility to leverage any method or set of practices that best suit their needs, and decide what work items to perform to reach the goals set by the target state. As a consequence, our seven practicum teams were able to leverage Essence even though they all used a different set of software engineering practices. When asked if he would use Essence on his next project, one student answered: “Yes, especially with a project team that is not used to the same software engineering process. In that instance Essence is a backdrop at the basis of the communication about all the considerations for the success of the project.” Another student added: “It is simple, lightweight and useful.”

The team accelerates its state progression by acting on its work items as soon as possible and iterating through the steps described above. Figure 8 illustrates this iterative process.

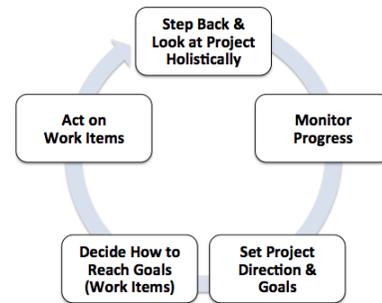


Figure 8. Essence monitoring and steering loop

In answer to our second research question, the approach adds value by providing the project team with a holistic view of the project, a mechanism for monitoring progress and steering projects, as well as an effective structure for team reflection and risk management. This is provided in a simple, lightweight, non-prescriptive and method-agnostic fashion.

**Research Question 3:** When in the project lifecycle does the SEMAT Essence’s monitoring and steering approach add value?

The value that the teams receive from SEMAT Essence’s monitoring and steering approach varies over the project lifecycle. Our study shows that most value is generated at the beginning of the project and that it decreases thereafter. Here are some illustrating quotes:

- When asked if following the Essence approach was worth their time, 20% of the students who answered yes to the question qualified their answers as follows: “Yes, it helped us at the beginning of the project”, or “Yes, it was worth my time in the first half of the project.”
- Among the students who mentioned that they would use Essence on their next project, one qualified her answer as follows: “Yes, but only at the beginning of the project.”
- When asked what he liked the least about Essence, one student answered: “Essence lost value once the project settled because we dead ended on a set of cards.” Another

student added: “Essence is useful in the planning stages. Later on its usefulness is dying down. If you spend multiple weeks on one card, then spending time looking at it is less helpful.” This opinion is shared by about 50% of the students.

- The faculty in charge of the practicum course, and who has taught the course for 10 years, noted: “Compared to previous years, I see a much better early project organization with lot less floundering. I hope that we keep using Essence in the future. We should definitely keep it at the beginning of the projects.”

The alpha progression of team Co-located-3 illustrates the decrease in value, as seen in Figure 9. The chart shows that the team progresses well during the first half of the project, then reaches a plateau or stable state during a few weeks, before progressing again at the end of the project. This picture represents most teams’ progressions.

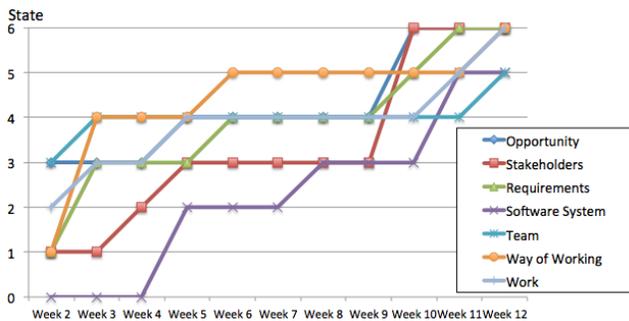


Figure 9. Alpha progression for team Co-located-3

By analyzing the work items generated by the team during the weekly SEMAT sessions, we find that the initial progression in Figure 9 is driven by those work items. Indeed these work items significantly contribute to bringing the project to a higher state. However, the final progression is done independently of Essence as the approach generates few work items at the end of the project. Most teams experienced this phenomenon, as illustrated in Figure 10 showing a consistent drop of work items over time.

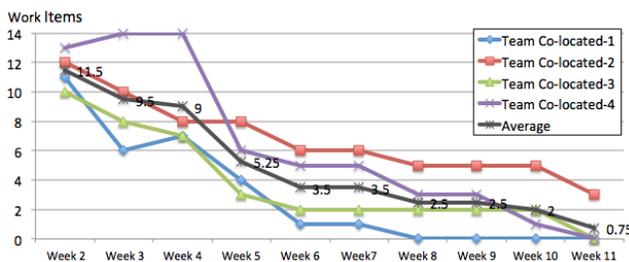


Figure 10. New work items generated per week

Despite the value decrease, the students’ perception of the approach remains positive by the end of the project according to the survey responses. Indeed a majority of students answered without qualification that they found Essence worth their time and will use the approach on their next project. A majority of students also mentioned reflection or retrospective as a strength of the approach. One student mentioned: “Even though we are not generating new tasks, the [SEMAT] meetings remain useful as they give us the opportunity to reflect upon our project.” Similarly, team Distributed-1 noted in its reflection report: “The team was pleased to see that Essence covered ‘The Way of Working’ as well as ‘The Team’. These two topics generated

useful team introspection at the beginning of the practicum and were nice reminders that the team does constant checkups for the overall condition of the members and the project.”

In conclusion, the SEMAT Essence’s monitoring and steering approach provided by the kernel alphas and their states is most effective at the beginning of the project. The effectiveness decreases over time. Indeed, the approach gradually loses its ability to enable the team to steer the project by generating new work items leading to a higher project state. The reasons behind the value decrease are explored in the next research question. However, most teams continue to perceive value throughout the lifecycle out of the approach reflection mechanism.

**Research Question 4:** What are the limitations to the approach’s effectiveness?

The previous section shows that the effectiveness of the approach decreases over time. This phenomenon could be explained by a number of factors. One factor relates to the fact that the need for the approach varies throughout the project lifecycle. For instance, as the project team matures and becomes a high-performing team producing high quality outcome, it becomes a self-steering team requiring less support from the approach. Similarly, once most project risks have been mitigated, the team requires less risk management support. Therefore, unless there is a disruptive event reverting the project to a lower state, the value decrease is to be expected.

In addition, some kernel characteristics influence the value decrease. Most Essence alphas have more states supporting the progression through the initial project phase compared to later phases. Figure 11 illustrates this phenomenon using the RUP [11] phases (Inception, Elaboration, Construction, and Transition) as an example. The chart shows that for most alphas, the number of states per phase is higher in Inception compared to the other phases. For five out of seven alphas, there is only one state in Construction. Most project teams spend the majority of their time in Construction. As a result, for these alphas the teams end-up remaining in the same state during their entire Construction phase. This lack of alpha states covering the Construction phase might be due to the fact that most of the technical work done in Construction is practice-specific, and therefore not supported by the universal kernel. Further investigations are necessary to identify potential ways to extend the kernel so it better supports the work done during Construction.

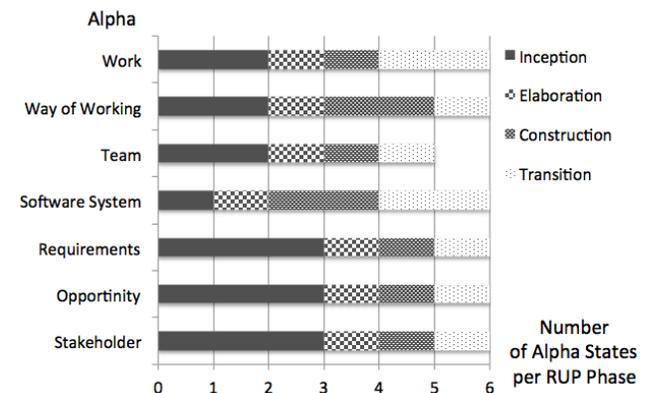


Figure 11. Number of alpha states per RUP phase

By definition, the Essence kernel is geared towards steering a project or release throughout its entire lifecycle in a fairly linear

fashion dictated by the alpha state progression. By repeating through the Essence monitoring and steering loop (see Figure 8), the team steers the project or release through the sequence of states for each alpha. This process is designed to support the work done at the release level and not to support the work done at the iteration level because of the following reasons:

- The kernel is lifecycle-independent and therefore does not provide specific support for iterative development.
- The kernel alpha states and their checklist items are specified at the project or release level, not at the iteration level.

Consequently, we were unable to effectively leverage the approach to help steer projects during Construction on iterative projects. The teams received some guidance on iterative development from whatever method they adopted, like Scrum and XP that are optimized for iterative development.

In conclusion, the observed value decrease could be explained by three factors. The first factor is a decrease in the need for the approach as the team matures and becomes self-steering. The second factor is a front-end focus of the alphas, making the approach most effective during project initiation. The third factor relates to the definition of the kernel alpha checklists, which are specified at the project or release level, not at the iteration level. This explains the value decrease during construction on iterative projects.

As a result, the monitoring and steering mechanisms are most effective during project initiation and for monitoring and steering the work done at the project or release level. This type of work could be qualified as “universal” as it is generally common across projects. This confirms that the kernel provides effective support for universal work. Support for non-universal technical work has to come from additional practices added on top of the kernel, or by extending or altering the kernel definition.

Finally, a limitation pointed out by about 40% of the students in the survey, is related to some ambiguity in the alpha checklists definition. For instance, one student mentioned: *“The checklist within an alpha can have cryptic language. Sometime, the points are ambiguous.”* Here are some examples of typical student reactions:

**Checklist Item:** Enough of the requirements are addressed ...

**Student:** *What do they mean by “enough”?*

**Checklist Item:** Constraints are identified and considered.

**Student:** *What kind of constraints are they talking about?*

**Checklist item:** Critical interfaces have been demonstrated.

**Student:** *What do they mean by “demonstrated”?*

**Checklist Item:** The key practices and tools that form the foundation of the way-of-working are selected.

**Student:** *Which tools form the foundation of the way-of-working?*

Thus, a limitation to the approach effectiveness comes from some ambiguities in the alpha checklist definitions, leading to situations where the team discusses the meaning of a checklist item instead of having a conversation about the project. At times, this disrupts the flow of otherwise valuable team discussions.

## 5. FROM THE EDUCATOR’S PERSPECTIVE

This section shares some of our findings related to introducing Essence to a project team having no prior experience with the approach, together with a few words of caution related to the use or misuse of Essence for evaluation purposes.

## 5.1 Incremental versus Workshop Introduction Approaches

Our experience with both the incremental and workshop approaches to introducing Essence indicates that both have value and that adoption resistance drives which approach is best for the situation.

### 5.1.1 Incremental

The incremental introduction approach brings immediate value to the teams, as noted by team Distributed-2 in its reflection report: *“The team found Essence valuable right from the start. [...] it helped manage the direction and organization of the team by examining overlooked aspects of the project.”* For instance, during the first session, this team identified the work items “Understand the risks and constraints” and “Identify all stakeholders” based on its discussion around the **Opportunity** alpha and **Stakeholder** alpha respectively. Some previous practicum teams overlooked these kinds of discussions and work items. However, this approach delays many important conversations until all the alphas are introduced.

This approach has minimum perceived overhead. In our case, introducing Essence incrementally required only a weekly session of 30 minutes, during a five weeks period.

When adoption resistance might be an issue, we recommend introducing the approach incrementally through a series of regular (e.g. weekly) and short (e.g. 30 minutes) sessions. We also recommend having one dedicated faculty or coach per team.

### 5.1.2 Workshop

The workshop introduction approach introduces all the alphas at once, hence enabling the team to look at its project holistically while having conversations covering the various project’s dimensions as early as possible.

This approach requires that each team invest in one upfront workshop during which the team starts applying Essence directly to its current project. During our 90 minutes introduction workshop, 10 minutes were needed to set the context and motivate the exercise. The remaining 80 minutes was enough time for most teams to have a substantial conversation about their perception of the project’s current state and to generate work items to make forward progress. On average, 10 minutes were necessary for a team to cover one alpha, and most teams were able to cover the seven alphas during the workshop. However, our largest team of five members took an average of 20 minutes per alpha and hence had to complete the exercise during a follow-up session. The size of the team as well as internal disagreements generated some longer conversations.

We recommend adjusting the workshop duration based on the team size and any other known parameters that might influence the length of the conversations, like team dynamic or project uncertainty. We also recommend having one dedicated faculty or coach per team.

Table 2 summarizes and compares the incremental and workshop introduction approaches. We recommend the workshop approach when introducing Essence, unless faced with initial adoption resistance that may require a slower incremental approach.

**Table 2. Comparison of introduction approaches**

Approach	Description	Benefits	Drawbacks	When to Apply
<b>Incremental</b>	Essence alphas are introduced incrementally over a number of short sessions	Brings immediate value with minimum overhead	Full benefit is delayed until complete introduction of alphas	In case of adoption resistance
<b>Workshop</b>	Essence alphas are introduced all at once during a single session	Brings immediate and optimum value (all alphas are covered)	Requires initial overhead	Always, except in case of adoption resistance

Given our recommendation of one faculty per team, in the future, we plan on introducing Essence and conducting a full assessment during a 90-minute team meeting. In a course with five teams, scheduling five separate team meetings is easier than trying to do five simultaneous team meetings with five instructors.

Following the introduction of Essence, we recommend that each project team continue to meet on a regular (e.g. weekly) basis for a short (e.g. 30 minutes) session throughout the initial project phase. Once a team reaches “construction” and becomes a high-performing team producing high quality outcome, the frequency of the sessions could decrease because the team needs less support from the approach.

### 5.2 Alpha Introduction Order

Participants can be overwhelmed when shown all the states for all the alphas at once. Introducing the alphas one at a time solves this problem, whether this is done incrementally over a series of meetings or during a single session or workshop.

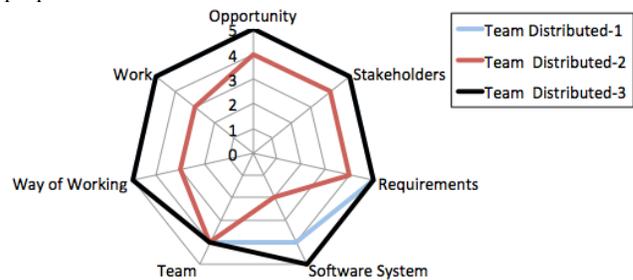
We recommend introducing the **Stakeholder** and **Opportunity** alphas (aka the customer dimension) first, since it generally makes sense to have a discussion around this dimension before drilling down into the details of the solution and endeavor. We introduce **Stakeholder** prior to **Opportunity** since many participants find it easier to relate to the **Stakeholder** alpha than the abstract **Opportunity** alpha. The other alphas could be introduced as needed based on pain points. When in doubt about which alpha to introduce next, one alpha could be selected randomly. Indeed, the value is in having a holistic view of the project, therefore looking at any dimension has potential benefits.

### 5.3 A Word of Caution

As noted by William Cameron [5], “not everything that can be counted counts, and not everything that counts can be counted.” This is especially true when it comes to individual and team performance evaluation, where using metrics could be dangerous. Inappropriate metrics could be used (like measuring a developer performance based on the number of lines of code produced), metrics could be gamed, and metrics do not give the full picture.

As an example, Figure 12 shows the final state of our distributed teams at the end of their project. From the chart, one might infer that Team Distributed-3 outperformed Team Distributed-1, which in turn outperformed Team Distributed-2 that lagged behind. In fact, despite having the highest project state, Team Distributed-3’s work was slightly disappointing according to both the clients and faculty. The team ended-up with a lower grade compared to the other teams. This was due to quality issues during various product demonstrations and a lack of architecture documentation that surfaced during the team final presentation. This makes the team self-assessment of the **Software System** alpha and therefore of its

overall project state questionable. Team Distributed-2 lagged only behind because the initial solution proposed by the client turned out to be infeasible, and the team did an excellent job redirecting the project into a constructive direction and hence earned an excellent grade. The routinely low **Software System** alpha state served as a red flag for risk management purposes, not evaluation purposes.



**Figure 12. Final project state of the distributed teams: Avoid this kind of comparison**

In conclusion, we recommend leveraging the approach to identify projects at risk rather than for performance evaluation purposes. Some teams might be only ahead because they are overly optimistic about their project state, while others might be behind because of circumstances beyond their control.

## 6. CONCLUSION

This paper presents the results of a field study aiming at understanding the value project teams receive from following the SEMAT Essence’s monitoring and steering approach provided by the Essence kernel alphas and their states. The study involved seven teams of master students with no prior knowledge of Essence.

The study validated our research hypothesis by showing that the approach provides student teams with a simple, lightweight, non-prescriptive and method-agnostic way to examine their projects holistically, structure team reflections, manage risks, monitor progress and steer their projects. Compared to the ten previous years, the faculty in charge of the practicum course noted that there was “*much better early project organization with a lot less floundering.*” Indeed, the approach enables students to learn to steer projects effectively by addressing the various dimensions of software engineering.

The study also highlighted that the monitoring and steering mechanisms are most effective during project initiation and for monitoring and steering the work done at the project or release level. This type of work could be qualified as “universal” as it is generally common across projects. This confirms that the kernel provides an effective support for universal work. Support for non-universal technical work should come from additional practices added on top of the kernel, or by extending or altering the kernel definition.

The results of this paper are limited to evaluating the effectiveness of the Essence approach when a facilitator is involved. Even though faculty involvement was kept at a minimum to limit influencing the students, a faculty served as facilitator during the SEMAT meetings throughout the practicum projects. More research is necessary to assess the effectiveness of the approach without facilitators.

The project context was fairly similar for all teams, except for their geographic distribution and average work experience. While the study has not revealed any influences of these contextual

factors on the approach's effectiveness, additional data is required to confirm or refute any impact. The Essence steering and monitoring approach appears equally applicable to both co-located and geographically distributed teams.

We are currently working with another set of practicum teams at Carnegie Mellon University in Silicon Valley to gather more data necessary to evaluate the SEMAT Essence's monitoring and steering approach, with a focus on both effectiveness of the approach and accuracy of the model.

## 7. ACKNOWLEDGMENTS

We would like to acknowledge the contributions of all the reviewers, and thank them for their insightful comments on early drafts of this paper. A special thanks to Hakan Erdogmus and Maria Nelson for their thorough reviews and invaluable suggestions, and to Ed Katz for recommending that we conduct our field study in the context of his practicum course and for his support throughout the study. Finally, we would like to warmly thank the students who participated in the study, for their openness and insights.

## 8. REFERENCES

[1] Ambler, S., & Lines, M. 2013. *Disciplined agile delivery: A practitioner's guide to agile software delivery in the enterprise*. IBM Press.

[2] Bareiss, R., Sedano, T., & Katz, E. 2012. Changes in Transferable Knowledge Resulting from Study in a Graduate Software Engineering Curriculum. *Software Engineering Education and Training (CSEE&T), 2012 IEEE 25th Conference*, 3-12. IEEE.

[3] Beck, K. 1999. *Extreme Programming Explained: Embrace Change*. Addison-Wesley. ISBN 978-0-321-27865-4.

[4] Boehm, B., & Turner, R. 2003. *Balancing Agility and Discipline: A Guide for the Perplexed*. Addison-Wesley

Longman Publishing Co., Inc. Boston, MA, ISBN:0321186125.

[5] Cameron, W. B. 1963. *Informal sociology: A casual introduction to sociological thinking*. Vol. 21, New York Random House.

[6] Grenning, J. 2002. Planning Poker or How to Avoid Analysis Paralysis while Release Planning. *Renaissance Software Consulting*.

[7] Jacobson, I., Meyer, B., & Soley, R. 2009. The SEMAT Initiative: A Call for Action. *Dr. Dobb's Journal*.

[8] Jacobson, I., Ng, P. W., McMahon, P., Spence, I., & Lidman, S. 2012. The Essence of Software Engineering: The SEMAT Kernel. *Queue*, vol. 10, no. 10.

[9] Jacobson, I., Ng, P. W., McMahon, P., Spence, I., & Lidman, S. 2013. *The Essence of Software Engineering: Applying the SEMAT Kernel*. Addison-Wesley.

[10] Jedlitschka, A., & Pfahl, D. 2005. Reporting guidelines for controlled experiments in software engineering. *Empirical Software Engineering, 2005 International Symposium*. doi: 10.1109/ISESE.2005.1541818.

[11] Kroll, P., & Kruchten, P. 2003. *The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*. Addison Wesley.

[12] Kruchten, P. 2010. Contextualizing Agile Software Development. *Journal of Software: Evolution and Process, Special Issue: Selected Industrial Experience Papers of EuroSPI*.

[13] Schwaber, K. 2004. *Agile Project Management with Scrum*. Microsoft Press. ISBN 978-0-7356-1993-7.

[14] Stellman, A., & Greene, J. 2005. *Applied Software Project Management*. O'Reilly Media.

## 9. APPENDIX

The table below provides the reader with the field study row data on alpha state progression. Each number in **bold** (from 1 to 6) represents an alpha state. Each number in *italic* (from 2 to 15) represents the week during which the team reaches the state. For instance, team Distributed-1 reaches the **Stakeholder** state **1** at week 2. Two or more italic numbers in one cell reflects a state backtracking. For instance, team Co-located 2 reaches the **Team** state **3** at week 2. The team then progresses to a higher state, but backtracks to state **3** at week 10.

Table 3. Field study row data on alpha state progression

Alpha	Stakeholder						Opportunity						Requirements						Soft. System						Team						Way of Working						Work													
Team Distributed-1	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>								
	<i>2</i>	<i>3</i>	<i>3</i>	<i>3</i>	<i>14</i>			<i>2</i>	<i>3</i>	<i>14</i>			<i>4</i>	<i>5</i>	<i>7</i>	<i>14</i>			<i>5</i>	<i>7</i>	<i>14</i>	<i>14</i>				<i>4</i>	<i>14</i>	<i>15</i>			<i>3</i>	<i>5</i>	<i>14</i>	<i>14</i>	<i>15</i>			<i>5</i>	<i>7</i>	<i>14</i>	<i>15</i>									
Team Distributed-2	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>								
	<i>3</i>	<i>3</i>	<i>4</i>	<i>6</i>			<i>2</i>	<i>4</i>			<i>3</i>	<i>4</i>	<i>7</i>	<i>11</i>			<i>7</i>	<i>13</i>									<i>5</i>	<i>11</i>	<i>15</i>	<i>4</i>	<i>7</i>	<i>11</i>	<i>15</i>	<i>15</i>	<i>15</i>	<i>5</i>	<i>7</i>	<i>11</i>	<i>15</i>	<i>15</i>	<i>15</i>									
Team Distributed-3	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>								
	<i>3</i>	<i>4</i>	<i>4</i>	<i>4</i>	<i>13</i>	<i>2</i>	<i>3</i>	<i>3</i>	<i>4</i>	<i>13</i>	<i>3</i>	<i>4</i>	<i>4</i>	<i>6</i>	<i>13</i>	<i>5</i>	<i>6</i>	<i>13</i>	<i>13</i>	<i>13</i>			<i>5</i>	<i>13</i>	<i>15</i>			<i>2</i>	<i>4</i>	<i>10</i>	<i>15</i>			<i>5</i>	<i>10</i>	<i>13</i>	<i>15</i>													
Team Co-located-1	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>								
	<i>2</i>	<i>2</i>	<i>2</i>	<i>3</i>	<i>10</i>	<i>10</i>	<i>2</i>	<i>2</i>	<i>2</i>	<i>2</i>	<i>10</i>	<i>11</i>	<i>2</i>	<i>2</i>	<i>3</i>	<i>3</i>	<i>11</i>	<i>2</i>	<i>3</i>	<i>10</i>	<i>11</i>			<i>2</i>	<i>2</i>	<i>2</i>	<i>4</i>	<i>12</i>	<i>2</i>	<i>3</i>	<i>3</i>	<i>4</i>	<i>6</i>	<i>12</i>	<i>2</i>	<i>3</i>	<i>3</i>	<i>5</i>	<i>12</i>	<i>12</i>										
Team Co-located-2	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>								
	<i>3</i>	<i>9</i>	<i>9</i>	<i>9</i>	<i>11</i>	<i>11</i>	<i>2</i>	<i>3</i>	<i>6</i>	<i>6</i>	<i>11</i>	<i>2</i>	<i>3</i>	<i>5</i>	<i>6</i>	<i>11</i>	<i>11</i>	<i>3</i>	<i>5</i>	<i>6</i>	<i>11</i>	<i>11</i>			<i>2</i>	<i>2</i>	<i>2</i>	<i>10</i>	<i>11</i>	<i>2</i>	<i>2</i>	<i>2</i>	<i>10</i>	<i>11</i>	<i>2</i>	<i>2</i>	<i>2</i>	<i>3</i>	<i>12</i>	<i>2</i>	<i>3</i>	<i>3</i>	<i>3</i>	<i>6</i>	<i>12</i>	<i>2</i>	<i>3</i>	<i>5</i>	<i>9</i>	<i>11</i>
Team Co-located-3	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>								
	<i>2</i>	<i>4</i>	<i>5</i>	<i>10</i>	<i>10</i>	<i>10</i>	<i>2</i>	<i>2</i>	<i>2</i>	<i>5</i>	<i>10</i>	<i>10</i>	<i>2</i>	<i>3</i>	<i>3</i>	<i>6</i>	<i>10</i>	<i>11</i>	<i>5</i>	<i>5</i>	<i>8</i>	<i>11</i>	<i>11</i>			<i>2</i>	<i>2</i>	<i>2</i>	<i>3</i>	<i>12</i>	<i>2</i>	<i>3</i>	<i>3</i>	<i>3</i>	<i>6</i>	<i>12</i>	<i>2</i>	<i>2</i>	<i>3</i>	<i>5</i>	<i>11</i>	<i>12</i>								
Team Co-located-4	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>	<b>1</b>	<i>2</i>	<b>3</b>	<i>4</i>	<b>5</b>	<i>6</i>								
	<i>2</i>	<i>3</i>	<i>4</i>	<i>6</i>	<i>10</i>	<i>10</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>10</i>	<i>2</i>	<i>4</i>	<i>4</i>	<i>6</i>	<i>10</i>	<i>10</i>	<i>2</i>	<i>4</i>	<i>6</i>	<i>10</i>	<i>10</i>			<i>2</i>	<i>2</i>	<i>4</i>	<i>6</i>	<i>12</i>	<i>2</i>	<i>4</i>	<i>5</i>	<i>8</i>	<i>8</i>	<i>12</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>11</i>	<i>12</i>									