How to Start an Environmental Software Project

Ioannis N. Athanasiadis

HAL Id: hal-01852630
https://inria.hal.science/hal-01852630

Submitted on 2 Aug 2018

How to start an environmental software project

Ioannis N. Athanasiadis

Wageningen University, Wageningen, The Netherlands
email: ioannis@athanasiadis.info

Abstract: How to lay the grounds for interdisciplinary teams to start communicating and collaborating effectively remains an obstacle for many environmental software efforts. In this work, a structured, participatory interactive method is introduced: The Inception Workshop aims to assist interdisciplinary teams at the start-up stage of an environmental software project, with the goal to explore the solution space and for early requirements analysis. It is an ice-breaker event to engage heterogeneous actors to open up, express their interests, and start working together to identify and solve common problems. Two installations of the workshop were conducted, and participant familiarity to the problem and technologies involved were captured with with pre- and post-workshop questionnaires. Participant responses proved statistical significance in increasing participant confidence with concepts across disciplines.

1   Introduction

Environmental software projects are intensive research collaboration projects, where research challenges go beyond a single discipline. Collaboration projects are typically characterised by heterogeneous actors with collective responsibilities and accountabilities, organised in geographically dispersed teams [6]. Those aspects imply challenges for conventional project management approaches, and involve uncertainties about working and collaboration methods. A clear *big picture* of the project, where heterogeneous actors identify their roles and communicate effectively is the key for successfully managing collaborative projects. To this end, it is essential that actors involved develop early in the project a common language to share their ideas, and shape a common vision.

Voinov and Shugart suggested that we need more creativity in integrated modelling, instead of mechanistically plugging modules together [13]. However, there has been little reported ever since, on how environmental software projects are set up in order to foster creativity. On the contrary, most interdisciplinary environmental modelling and software projects today are still organized in complex disciplinary hierarchies.

While this paper does not address the problem in its entirety, it introduces and evaluates a modus operandi for starting with an environmental software projects. It presents a single day workshop format that intends to lay the grounds for interdisciplinary teams to start communicating and collaborating effectively in integrated environmental software teams. The *Inception Workshop* is a participatory method to be applied at the start up stage of a project to explore the solution space or for early requirements analysis. The intention is to start a project with an event that allows for creativity, to charge

heterogeneous actors involved, to start communicating effectively about their disciplinary interests, and work together beyond disciplinary barriers to identify and solve common problems.

The rest of this paper is structured as follows. Section 2 debriefs common attitudes in interdisciplinary projects and introduces the *Inception Workshop* objectives and participant roles. Section 3 presents implementation guidelines and Section 4 details an experimental setup, involving the two workshop installations, along with lessons learned in practice. Section 5 presents the workshop evaluation results, and conclusions are drawn in Section 6.

2   Materials and Methods

2.1   Domain understanding and common attitudes in environmental software

Domain understanding is generally acknowledged as a prerequisite for successful projects that involves information technology innovations, however it is often neglected. Software developers seem not to appreciate domain modelling, nor to recognize its importance in effective requirements engineering and risk management [10]. A common attitude observed in several occasions in environmental software projects is that information engineers overestimate the capabilities of their solutions, while underplay the complexity of environmental systems. At the same time, environmental experts tend to underestimate the burden of technological solution deployment and system integration, they typically pass on the shoulders of the IT team. Both sides seem to spin a vicious cycle that creates great expectations on software solutions and drives projects to failure. In environmental software projects we observe the same, recurring motif: *Environmental modelling problems are convoluted for the IT experts, and information technology innovations are arcane for environmental scientists.* The result is the *illusion of the magic wand for environmental modelling*: Information technology experts believe that they have one, while environmental experts expect that is going to work!

Model integration and integrated modelling in environmental software projects is a challenging task for several reasons. Janssen (2009) viewed integration in interdisciplinary modelling projects as a multi-headed Hydra snake, due to the many different types of integration that have to be achieved in parallel, and to the important role of communication [9].

For an environmental scientist it is costly to be involved in an interdisciplinary project, as it entails extra effort, for which there is not always appropriate academic credit, and the development of new skills. Rhoten [11] notes that interdisciplinary research requires *sharing existing information through "collegial" interactions*, confirming that interdisciplinary science needs effective and active communication across disciplines.

Environmental software projects are also challenging for software experts. Software curricula cover a handful of application areas, and most textbooks have examples from the most common ones, let it be retail, finance, or entertainment. In this respect, IT experts are less prepared for science applications, and even less for interdisciplinary ones. With practicing the profession, software engineers become more experienced

with certain application areas, and switching to a new one can be a very stressful experience. Bjørner estimates that "*to establish a reasonably trustworthy and believable theory of a domain may take years, possibly 10-15!*" [5].

Environmental software projects are by definition interdisciplinary, and interdisciplinary science is founded on communication. This comes together with current practice in requirements engineering, as requirements gathering and agreement process has shifted from a documentation to a communication effort [15]. Traditional methods for requirement analysis as via detailed specification documents, questionnaires or interviews are not suited for environmental software projects. More agile approaches are better suited, as those using less formal means for team building, engaging with stakeholders, gathering requirements, and possibly involve rapid prototyping in small increments to verify them continuously with the end-users.

2.2   The Inception Workshop approach: purpose and objectives

Effective communication requires heterogeneous teams to speak the same language. In most environmental software projects this means that they need to invent it. To facilitate this goal, the *Inception Workshop* is proposed, an event that is expected to be the first meet-up of experts within an integrated project. At project starting phase, communication between team members is difficult. Each discipline has its own value systems, models of work, communication patterns, terminology and jargon. They look like knowledge islands with not much in common. Domain understanding is the bridge that will make these worlds collaborate effectively. Focusing early on building domain knowledge is essential for the success of an environmental software project, as it will help scientists to get out of their disciplinary safe house and foster collaboration. Interdisciplinary science can be achieved only by understanding better each other's scientific niche, while tackling together well-defined problems. This goes beyond superficial coexistence that is founded on short-lived meetups. Consilience is the key to interdisciplinary projects, as scientists not only need to interact with each other, but also have to engage in seeing the *big picture*.

In the case of environmental software projects, the more environmental scientists are exposed to innovative IT technologies, the easier it becomes to build confidence with IT processes and tools, and to engage actively in technical integration tasks. Similarly, there are benefits for IT scientists involved, as a dialogue with environmental scientists allows to identify early critical system aspects. In an environmental software project, IT experts need to get out of their cubicles and the tools they are locked-in with, and start working on interdisciplinary integration from the start of the project.

The *Inception Workshop* aims to establish communication bridges across heterogeneous team members in an environmental software project. It is intended for early project stages, for problem clarification and early requirements analysis. The goal of the workshop is to identify initial user stories, which will be further considered in the project. The main tool is to give the floor to scientists, members of a heterogeneous team to present their own expertise in an informal, open fashion. Team members interact intensively for a day, collaborate, and make their first achievements together.

When setting up the *Inception Workshop* format, it was assumed that time allotted is restricted. As integrated projects typically involve geographically distributed teams, we assume that there is available only one full day (or two half ones) which seems already too much for facetime in many collaboration environments. However, more time may be allotted, if conditions allow. Spontaneous, informal communications (that resemble standup meetings of agile programming) were intentionally preferred as opposed to structured formats preferred in academic fora.

In terms of preparation, team members are invited to come with no prepackaged content as presentation slideshows, or marketing pamphlets. All discussions are intentionally unpremeditated, unrehearsed. The invitation only sets the overall workshop goal in the form of a question, that does not impose any particular solution. It can be phrased as "*How to improve the domain-specific system X (with technology offering Y)?*". During the workshop, there is no speaker agenda, only a general structure for the discussion (see Table 1 below), with a single goal which is to identify one or more interesting *user stories*.

The general flow of the workshop is as follows. First, domain (environmental) experts get the floor to present their current (or intended) activity in a storytelling fashion, and altogether try to identify *hotspots* which are interesting or challenging, (or weak and tedious), and improvements seem needed. This identifies a first list of open problems that call for interdisciplinary intervention. Then, the team reflects on technologies and solutions that could potentially address some of the open problems. This is a spontaneous reaction in the form of sharing anecdotes or experience from previous projects from other fields. The final step is that the team tries to match *hotspots* with *tools available* to put together solutions and formulate them as *user stories*. Last but not least, priorities and risks are estimated for each user story.

Table 1: *Inception Workshop* typical agenda

Duration	Topic	Lead participant
30'	Welcome - The workshop process and introductions	Facilitator
90'	'*Storytelling*' (Domain understanding session)	Domain experts
30'	Short Break	
90'	'*Technology roadshow*' (Explore technology opportunities)	IT experts
60'	Long Break	
60'	*Synthesis* (Define first user stories)	All
30'	Short Break	
60'	*Synthesis* (*cont.*) (Set priorities and identify risks)	All

2.3 Participants and Roles

There are three types of participants involved. Team members may have the role of environmental expert, IT expert and a facilitator. As interdisciplinary projects involve big teams, it remains a challenge to select the right people to invite. Readers may be refer to literature for stakeholder identification [8,12].

The **facilitator** is responsible for running the workshop according to the general schedule and to keep discussions focused. Experts tend to get excited with their work and thus dive fast into details, or divert the discussion to irrelevant topics. The facilitator is the one to give pace to the discussion, ensure that everyone gets the opportunity to contribute with their ideas and encourage participants to stand up. She should not be the one that needs to deliver the final outcome of the workshop, in the sense that the stress should not be on her shoulders. Her role is rather to stimulate critical thinking in a Socratic way, than that of the interrogator, and she is not expected publicly interview the experts in a panel discussion. She is not required to have a very deep understanding of the domains or the technologies presented, rather than appreciating the various disciplines involved, interdisciplinarity and the process of integrated modelling. She should be very experienced in discussion facilitation.

Environmental experts participate in order to represent their disciplines, and present obstacles in their everyday work practice, which they want to solve with environmental software. When invited they are asked to bring to the workshop documents, tools, photographs or other artifacts from their practice they thing they would like to show, and share with the rest team. While this sound odd, in practice it was proved a nice way to start the discussion. In the start of the workshop most participants feel a bit odd, so "*What have you brought?*" is a nice line to start with, though certainly this may not be working with all disciplines or cases.

IT experts come to the workshop to better understand the problem at hand, and do their first step in requirements elicitation, together with the rest of the team. The composition of the IT team is not restricted to analysts or designers, and certainly developers are welcomed. Our assumption is that IT scientists not directly involved may contribute. IT team members have a lot to gain in terms of domain understanding, and from engaging with disciplinary scientists. They are expected to show creativity and forward thinking and try to exemplify their technologies to the rest team. This is a crucial step for building confidence among the team members. Note that IT experts from different backgrounds may be less open to change, as new technologies may be disruptive to their routine.

The participation model is founded on open, democratic principles. While each organization involved in an environmental software project has its own structure and hierarchy, for this workshop participants are asked to contribute freely, directly, and simply. The facilitator needs to clarify from the beginning that the workshop is exploratory without formalities, and encourage participants to contribute. The facilitator is to protect the group from insisting participants that may slow down the discussion. At the same time, he makes clear that everyone has the right to speak, and he needs to accommodate time for all, while decisions are taken by the majority.

3   Implementation

3.1   Preparation and agenda

The workshop is intended to run for one or two days in a room with a round table. The equipment needed is a white-board or a flip-chart, pens, post-it notes and *stuff* brought by the participants. Part of the preparation is to provide food and beverages for coffee and lunch breaks. Table 1 presents a typical agenda. The workshop starts with a welcome from the facilitator and a short introduction round of the participants. The facilitator presents the workshop agenda and main question, and may give a couple of examples of user stories, if asked by the participants.

3.2   Part I: Storytelling

The first session focuses on *domain understanding*. The goal is to follow the current or future processes and increase the understanding of team members. Team share their knowledge across environmental domains involved in a **storytelling** fashion: They communicate their knowledge by sharing experiences and anecdotes. In a collaborative team environment, experts stand up to share their views of the system, while rest team members may intervene with clarifying questions. If needed, environmental experts may start drawing on flip charts to illustrate their claims. In other cases, they might show the stuff brought, that could be equipment, documents, or other exhibits of their practice. Each contributes with a partial view of the integrated system to be developed, undoubtedly biased by her own disciplinary background and intentions.

While the flip chart starts to fill up, and artifacts are on the table, participants are instructed to use post-it notes and identify "*hotspots*" that need intervention. These could be processes or activities that are popular, crucial or bottlenecks that are shared by team members. Discussions should not run into solutions at this stage, and participants are encouraged to flag hotspots, but reserve solution ideas for later. The facilitator needs to keep the storyline progressing, and prevent the team from diving into details.

3.3   Part II: Technology Roadshow

The second session, called **technology roadshow**, focus is on familiarizing the team with IT opportunities and identify possible solutions. This is achieved by working with examples and learning by analogy. This supports two objectives: First objective is to increase team awareness of IT tool capabilities, as potential options to adopt for certain problems. This will build interest in the beginning, and hopefully confidence at a later stage to (new) technologies for environmental software. Second objective is to identify via brainstorming possible solutions for the "*hotspots*" flagged in the previous session. Team members take turns and '*put guns on the table*', i.e they exemplify how technical solutions might be used for tackling the problems presented before. Existing or new solutions are presented informally, in the form of examples. Solutions do not need necessarily be related to the problem at hand, but may come from other disciplines as well. Simplifications are encouraged so that team members with different backgrounds get a grasp of how a solution works and which are the potential benefits. Participants are

encouraged to ask questions, and the facilitator is responsible for moderating jargon language and reminding technical experts to use simple examples from applications known to the rest team members.

3.4 Part III: Synthesis of User Stories and Risks

The final session aim is to produce a **synthesis** of the two previous sessions. In a collaborative team environment, participants are challenged to interpret *together* the identified hotspots and the tools presented in order to form *initial user stories*. Agile requirements methodology considers user stories as very high-level definition of requirements, containing just enough information so that the developers can produce a reasonable estimate of the effort to implement it [7]. In our case, initial user stories aim to produce high-level requirements of an environmental software system, often called *epic* in agile methodology.

The first part of this session is to identify user stories. Any workshop participant is free to summarize a solution of the equation *hotspot + tool = user story*. While a story is presented others may contribute to clarify or extend it. User stories at this stage are given short names and listed on the flipchart, so that participants may refer to them later on.

The second part of this session is to go through the user story list and set priorities in terms of effectiveness, risk and cost. While these are rough estimates, it is also interesting from a team building point of view, as it reveals the different priorities and perception of difficulty from heterogeneous team members.

4 Application

The *Inception Workshop* method presented above has been applied in two projects that the author has been involved in, and are shortly described below.

4.1 Workshop #1

The AITOLOS project aimed to identify smart ICT solutions to be incorporated into the everyday work of national forest protection services [1,3]. The project team consisted of forest service officers with several years of field work in forest protection, but from different functions, and software experts with varying expertise (business analysts, telecommunications, GIS, image processing, remote sensing, semantic web). In this team environment forest service officers were not proficient users of IT, and software experts had no prior experience in forestry applications. In order to identify potential solutions, a bidirectional knowledge shift between team members was needed, so that they develop a common language and achieve consensus.

The AITOLOS inception workshop took place in Thessaloniki, Greece, on January 25, 2013, with the participation of 12 environmental experts from the forest services of Kilkis and Goumenissa, in Greece, and 7 software experts from the Information Technologies Institute (CERTH), and the Surveying and Geomatics Department of TEI of Central Macedonia. The project was in the initial problem identification phase, and the main goal of the workshop was to answer the question: "*How smart ICT solutions may*

be applied in forest service practice to combat illegal logging and timber trade?". After a single day of intensive discussions, the team concluded with seven user stories, spanning the whole lifecycle of logging and timber trade. The workshop findings are detailed in separate reports [2,3].

4.2   Workshop #2

The ALPINE project aimed to develop an intelligent, low-power sensor network architecture for environmental management. One of the two pilots is concerned with wildlife management, and specifically was concerned with how sensor networks can be used to improve the current situation with large carnivore monitoring and protection in the study area. Though the case study was different from the previous workshop, again the situation was similar: IT experts from industry and academia had very poor understanding of large carnivore protection domain, and wildlife scientists were not up to date with recent advances in IT. While the project was not in a very early stage, communication between the two sides was very slow and ineffective. Based on the first workshop experience, we decided to organize an event with the same structure, and monitor if it will help improving communications among the interdisciplinary team, and achieve some progress with the test case.

   The ALPINE/Wildlife Inception Workshop was held in Xanthi, Greece, on February 13, 2014. We invited two wildlife scientists, working with different species and six IT experts. The the main goal of the workshop was to answer the question: "*How intelligent sensor network technology and applications can be used for bear and wolf monitoring and management?*". After a single day of intensive discussions, the team concluded with four user stories, three related to bear and one related to wolf management. One of those epics has been actually implemented in the project [4].

5   Evaluation

5.1   Methodology

In order to evaluate the effectiveness of the proposed method, in both experimental workshops we conducted a survey. The survey was done in two stages, with pre- and post- workshop questionnaires. Both questionnaires included the same set of questions, in which the respondents were asked to evaluate their familiarity with environmental domain concepts and procedures, and IT tools and technologies. Half of the questions were related to the application domain and the rest related to technology. All participants were asked to respond to all questions independently of their expertise. The level of familiarity was measured using a five-point Likert scale. A portion of the two questionnaires is presented in Table 2**.** The post-workshop evaluation form included questions to evaluate workshop performance and organization, adapted from [14]. Workshop participants gave their oral consent to process their answers, given that no personal information are disclosed. In the first workshop, 14 out of 19 participants returned the questionnaire forms. In the second workshop, all seven participants returned the questionnaire forms.

Table 2: Questionnaires for the two workshops. The question was How familiar are you with...

	Workshop #1		Workshop #2
1	Forest value	1	Value of ecosystems
2	Forest office organization	2	Wildlife in Greece
3	Forest management plans	3	Large carnivore movement patterns
4	Forest products	4	Human-wildlife coexistance conflicts
5	Legal logging processes	5	Wolf damages on livestock
6	Forest product market	6	Human effect on wildlife habitat
7	Taxes on loggers permits	7	Spotting wildlife
8	Legal timber trade cycle	8	Trap and drug a bear
9	Illegal timber trade cycle	9	Bear identification collars
10	Cross-border illegal logging	10	GIS and maps
11	GPS applications	11	Ecosystem population models
12	GIS and maps	12	GPS applications
13	Remote sensing	13	Delay tolerant networks
14	Digital signatures	14	Knowledge modelling
15	RFID tags	15	RFID tags
16	Office applications	16	Barcode, QR code
17	Barcode, QR codes	17	Software development
18	Software development	18	Sensor networks
19	e-governance	19	Low power sensors
20	Social networks		

5.2 Ethics statement

Voluntary feedback evaluation forms are commonly distributed in workshops. Participants of the two pilot workshops evaluation were exposed to no risk; there were no vulnerable populations involved; and no sensitive data were collected. For these reasons, no approval from an ethics committee/IRB was asked. Participants were informed briefly before the start of the workshop about the goals of this research and were given the option to fill in the pre-workshop questionnaire that was included in their folder. The same happened with post- workshop questionnaires. The questionnaire forms included information about the researcher who conducted the study. The questionnaire forms included a written outline of the research purpose and contained a statement that completion and return of the questionnaire indicates consent to participate in the study.

Written consent was not obtained as there was no risk for the participants; to ensure anonymity of the responses; and the involved procedures would not normally require written consent outside the purposes of this study. Completed questionnaire forms were returned in a box, while the researcher was not present. No private information has been disclosed. No penalty or consequences was implied by not filling in the questionnaire. No benefits or long term engagement was associated with participating to this study. Written responses were digitized, and original forms destroyed. Participants had the option to mention their name, a common practice for workshop feedback forms to build engagement and trust with a team. However, responses were treated anonymously. Anonymized responses of the questions relevant for reproducing the results presented in this manuscript have been archived on Zenodo [16].

5.3   Results

Answers to pre- and post-workshop questions suggest that most participants felt more familiar with the domain concepts and related technologies after the workshop end. This can be considered as a sign of effective communication towards the development of a common language in a heterogeneous team.

We performed the *Wilcoxon matched-pairs* signed-rank test, a nonparametric method to compare before-after (or matched) questionnaires. The test was conducted to each participant matched responses before and after the workshop, to determine whether there was a statistical difference in their reported *familiarity* with the domains and technologies involved. The results are reported in Table 3, and indicate that in the vast majority of the participants, in both workshops, there was a statistically significant difference in their responses. For *Workshop #1*, the Wilcoxon analysis yielded with $p < 0.05$ for 10 out of 14 participants. The difference in average response was not due to chance for the majority of both domain and IT experts. For *Workshop #2*, the Wilcoxon tests reveals that shift in respondents' *familiarity* is statistically significant for five out of seven participants with $p < 0.05$.

Table 3: Questionnaire statistics (Wilcoxon p-value) from the two workshops. The perception of each participant is evaluated with respect to her confidence to domain concepts and technology topics. Participants are identified with a letter and a number. The letter encodes the discipline of the participant - D for domain scientists and S for software experts.

	Workshop #1			Workshop #2	
ID	p value	p < 0.05	ID	p value	p < 0.05
D1	0.002	*	D1	1.000	
D2	0.009	*	S1	0.070	
D3	0.009	*	S2	0.001	*
D4	0.025	*	S3	0.001	*
D5	1.000		S4	0.033	*
D6	0.033	*	S5	0.000	*
D7	0.057		S6	0.009	*
S1	0.554				
S2	0.000	*			
S3	0.001	*			
S4	0.018	*			
S5	0.001	*			
S6	0.530				
S7	0.020	*			

5.4   Lessons learned

Both workshops concluded with valuable outcomes for each project, and participants in principle were satisfied with the workshop goals, structure, and process. However, the two workshops were executed in different ways.

In the first one, discussion went deep in operational detailed, which crunched the time available for risk estimation. However, this allowed us to identify many high-level user stories, and satisfied the project need for identification of alternative solutions. At the same time, the large number of participants made discussions last longer, and some

of the domain experts had no opportunity to participate actively, rather they participated as observers. This was considered as a significant organizational drawback as it is commonly acknowledged that engaging with experts is hard to achieve, and their time shouldn't be wasted. Thus, in the second *Inception Workshop*, the same process was repeated with less number of participants. However, due to an emergency we missed one of the two domain experts, to our disappointment.

In the second workshop we experienced a better balance of time allocation between 'storytelling' and 'technology roadshow' sessions. The four user stories we concluded with were more detailed, tailored to the needs of the participants. In this respect, the workshop was less exploratory. Risk analysis and prioritization was done in more detail, also taking under account the limitations of the project. The take-home message from this installation was to host the workshop in the vicinity of the customer headquarters, if possible.

In both cases, the user stories were epic, in the sense that they can be broken down to smaller, more targeted stories [7]. However, this was rather expected as the goal of these workshops was not to identify specific features to be implemented, rather it was to identify solutions to be subsequently specified in detail.

6   Conclusions

Domain knowledge is essential for heterogeneous teams that are challenged with integrated projects, as limited or no familiarity with a domain increases the risks for project failure. Domain knowledge is even more critical with increasing agility of the software development process. As interdisciplinary teams move from bureaucratic, plan-based processes to more iterative and less structured methods, requirements elicitation is transformed from an understanding and documentation activity to a learning and communication one.

This paper argues that at the beginning of environmental software projects, more effective communication is required. While agile methods are well suited for ill-defined needs and ever-changing requirements, they assume effective communication among heterogeneous teams. This is certainly not the case in environmental software. As a remedy to this problem we presented the *Inception Workshop* to establish effective communication between team members, early in the project. With the *Inception Workshop*, a team of heterogeneous actors is challenged to collaborate for identifying user stories and associated risks for an interdisciplinary environmental software project. The method has been applied twice, and lessons were reported about the number of the participants, time allocation, specificity of the user stories. In both installations participant found the process stimulating and helped them to engage with each other.

Participants of both inception workshops were satisfied; and this is attributed to the fact that all had gains from the process. Some familiarized themselves with new domains, others with certain technologies. All participants learned new things, and had their say in the project start. A set of user stories to further develop in the project is a tangible result for everyone satisfaction. The workshops were also helpful in building team spirit due to the participatory sessions, where interactions brought participants

closer. Questionnaire analysis of the two pilot workshops suggests statistical significance of the participant responses after the workshop. Though, participant enthusiasm and long-standing engagement to their project success is the stronger evidence of success.

Acknowledgments

References

[1] AITOLOS Consortium, Project website, (2012).

[2] D. Anastasiadou, K. Koulinas, F. Kiourtsis, I.N. Athanasiadis, Complementary software solutions for efficient timber logging and trade management, in: J. Marx Gomez, others (Eds.), Proc. 28th Int'l Conf. Informatics for Environmental Protection (Enviroinfo 2014), Univ of Oldenburg BIS Verlag, Oldenburg, Germany, 2014: pp. 783–788.

[3] I.N. Athanasiadis, D. Anastasiadou, K. Koulinas, F. Kiourtsis, Identifying Smart Solutions for Fighting Illegal Logging and Timber Trade, in: J. Hrebicek, others (Eds.), Environmental Software Systems. Fostering Information Sharing (ISESS 2013), Springer, Neusiedl am See, Austria, 2013: pp. 143–153.

[4] I.N. Athanasiadis, F. Villa, G. Examiliotou, Y. Iliopoulos, Y. Mertzanis, Towards a semantic framework for wildlife modeling, in: J. Marx Gomez, others (Eds.), Proc. 28th Int'l Conf. Informatics for Environmental Protection (Enviroinfo 2014), Univ of Oldenburg BIS Verlag, Oldenburg, Germany, 2014: pp. 287–292.

[5] D. Bjørner, Domain engineering, In: Formal Methods: State of the Art and New Directions, Springer, 2010: pp. 1–41.

[6] J. vom Brocke, S. Lippe, Managing collaborative research projects: A synthesis of project management literature and directives for future research, International Journal of Project Management. 33 (2015) 1022–1039.

[7] C.G. Cobb, Making Sense of Agile Project Management, John Wiley; Sons, 2011.

[8] M. Glinz, R.J. Wieringa, Guest Editors' Introduction: Stakeholders in Requirements Engineering, IEEE Software. 24 (2007) 18–20.

[9] S.J. Janssen, Managing the hydra in integration: Developing an integrated assessment tool for agricultural systems, Wageningen University, 2009.

[10] R. Offen, Domain understanding is the key to successful system development, Requirements Engineering. 7 (2002) 172–175.

[11] D. Rhoten, Interdisciplinary research: Trend or transition, Items and Issues. 5 (2004): 6-11

[12] H. Sharp, A. Finkelstein, G. Galal, Stakeholder identification in the requirements engineering process, (1999) 387–391.

[13] A. Voinov, H.H. Shugart, 'Integronsters', integral and integrated modeling, Environmental Modelling & Software. 39 (2013) 149–158.

[14] WGBH, Enhancing Education: A Producer's Guide, (2004).

[15] Z. Zhang, M. Arvela, E. Berki, M. Muhonen, J. Nummenmaa, T. Poranen, Towards Lightweight Requirements Documentation, Journal of Software Engineering & Applications. (2010) 882–889.

[16] I.N. Athanasiadis, (2017). Anonymized survey responses from two Inception Workshops [Data set]. Zenodo. http://doi.org/10.5281/zenodo.1041743