PROJECT FINAL REPORT

Grant Agreement number: 243381

Project acronym: CERCO

Project title: Certified Complexity

Funding Scheme: STREP

Period covered: from 01 February 2010 to 31 March 2013

Project coordinator: Prof. Claudio Sacerdoti Coen
Alma Mater Studiorum – Università di Bologna
Tel: +39 051 2094973
Fax: +39 051 20 9 4510
E-mail: claudio.sacerdoticoen@unibo.it
Project website address: http://cerco.cs.unibo.it

4.1 Final publishable summary report

This section must be of suitable quality to enable direct publication by the Commission and should preferably not exceed 40 pages. This report should address a wide audience, including the general public.

The publishable summary has to include **5 distinct parts** described below:

- An executive summary (not exceeding 1 page).
- A summary description of project context and objectives (not exceeding 4 pages).
- A description of the main S&T results/foregrounds (not exceeding 25 pages),
- The potential impact (including the socio-economic impact and the wider societal implications of the project so far) and the main dissemination activities and exploitation of results (not exceeding 10 pages).
- The address of the project public website, if applicable as well as relevant contact details.

Furthermore, project logo, diagrams or photographs illustrating and promoting the work of the project (including videos, etc…), as well as the list of all beneficiaries with the corresponding contact names can be submitted without any restriction.

4.1.1 Execute summary

Problem statement: Programs can be specified using both functional constraints (what a program must do) and non functional constraints (how many resources – time, space, energy, etc – is the program allowed to use). Following the current state of the art, functional properties are verified on the high level source code combining user interaction (e.g. to state preconditions and invariants) with a multitude of interoperating techniques (invariant generators, type systems, abstract interpretation, theorem proving, etc.). Non functional properties are independently verified on the object code, but the verification requires knowledge on the functional behaviour, that needs to be reconstructed from the object code via abstract interpretation and user interaction. This situation presents several problems: 1) it may be hard for the user to provide knowledge on the execution of the object code in presence of complex optimizations, which are avoided; 2) techniques that work on the object code are not useful on the early development stages; problems detected by late cost analysis are more expensive to tackle; 3) parametric cost analysis is very hard: how can we reflect a cost that is parametric on the object code state (e.g. the value of a register or a carry bit) to a cost that the user can understand looking at the source code?; 4) the functional analysis performed on the object code using a fixed set of techniques and limited interaction yields results that are less precise than the ones obtained on the source code, affecting the precision of the estimated cost bounds.

CerCo vision and approach: We propose a reconciliation of functional and non functional analysis: both should be performed at the same time on the source code, sharing knowledge. What prevents this approach is the lack of a uniform and precise cost model for high level statements: 1) each statement occurrence is compiled in a different way and compiler optimizations may change the control flow; 2) the cost of an object code instructions depends on the state of execution dependent hardware components like pipelines and caches, which is not visible in the source code. To solve the issue, we envision a new generation of compilers able to keep track of the structure of programs during compilation and able to exploit that information to induce on the source code a non uniform, precise cost model for control blocks which is parametric on the execution history. Once the source level cost model is known, we can simply reduce non functional verification to the functional case and we can exploit any combination of techniques to automate the verification. In particular, the techniques previously used by WCET analyzers on the object code are still available on the source code, but can now be coupled and interact with more complex analysis. When the approach produces precise cost models too complex to reason about, safe approximations can be used to trade complexity with precision. Finally, the analysis on the source code can be performed even during early development stages when the components have been already specified, but not implemented yet: it is sufficient to axiomatize the non functional behaviour of all non implemented components.

Contributions: We have developed a technique, called labelling approach, to implement compilers that induce cost models on the source programs by keeping track of the code evolution in a very lightweight way. We have studied how to formally prove

the correctness of compilers that implement that technique. We have implemented and partially certified in an interactive theorem prover such a compiler from C to the 8051 object code. We have implemented a Frama-C plug-in that invokes the compiler on a source program, generates cost invariants from the cost model induced by compilation and finally certifies that the program respects the generated cost invariants by calling automated theorem provers, an innovative technique in the field of cost analysis. As a case study, we show how the plug-in can automatically compute and certify the reaction time of reactive C code obtained compiling Lustre programs to C.

4.1.2 Project context and objectives

Formal methods for verification of functional properties of programs have reached a level of maturity and automation that is allowing a slow but increasing adoption rate in production environments. For safety critical code, it is getting usual to combine rigorous software engineering methodologies and testing with static analysis in order to benefits from the strong points of every approach and mitigate the weaknesses. Particularly interesting are open frameworks for the combination of different formal methods, where the programs can be progressively specified and are continuously enriched with new safety guarantess: every method contributes knowledge (e.g. new invariants) that becomes an assumption for later analysis.

The scenario for the verification of non functional properties (time spent, memory used, energy consumed) is more bleak and the future seems to be getting even worse. Most industries verify that real time systems meets their deadlines simply measuring the time spent in many runs of the systems, computing the maximum time and adding an empirical safety marging, claiming the result to be a bound for the Worst Case Execution Time of the program. Formal methods and softwares to statically analyse the WCET of programs exist, but they often produce bounds that are too pessimistic to be useful. Recent advancements in hardware architectures is all focused on the improvement of the average case performance, not the predictability of the worst case. Execution time is getting more and more dependent from the execution history, that determines the internal state of hardware components like pipelines and caches. Multicores and non uniform memory models are drastically reducing the possibility of performing static analysis in isolation, because programs are less and less time composable. Clock precise hardware models are necessary to static analysis, and getting them is becoming harder as a consequence of the increased hardware complexity.

Despite the latter scenario, the need for reliable real time systems and programs is increasing, and there is an increasing pressure from the research community towards the differentiation of hardware. The aim is the introduction of alternative hardware whose behaviour would be more predictable and more suitable to be statically analyzed, for example decoupling execution time from the execution history by introducing randomization.

In the CerCo project we do not try to address this problem, optimistically assuming that static analysis of non functional properties of programs will return to be feasible in the long term. The main objective of our work is instead to bring together static analysis of functional and non functional properties, which, according to the current state of the art, are completely independent activities with limited exchange of information: while the functional properties are verified on the source code of programs written in high level languages, the analysis of non functional properties is entirely performed on the object code to exploit clock precise hardware models.

There are two main reasons to currently perform the analysis on the object code. The first one is the obvious lack of a uniform, precise cost model for source code

instructions (or even basic blocks). During compilation, high level instructions are teared apart and reassembled in context specific ways so that there is no way to identify a fragment of object code with a single high level instruction. Even the control flow of the object and source code can be very different as a result of optimizations. For instance, loop optimizations reduce the number or the order of the iterations of loops, and may assign different object code, and thus different costs, to different iterations. Despite the lack of a uniform, compilation and program independent cost model on the source language, the litterature on the analysis of non asymptotic execution time on high level languages that assumes such a model is growing and getting momentum. Its practical usefulness is doomed to be minimal, unless we can provide a replacement for such cost models. Some hope has been provided by the EMBounded European Project that compositionally compiles high level code to a byte code that is executed by an emulator with guarantees on the maximal execution time spent for each byte code instruction. The approach indeed provides a uniform model, at the price of loosing precision of the model (each cost is a pessimistic upper bound) and performance of the executed code (because the byte code is emulated compositionally instead of performing a fully non compositional compilation).

The second reason to perform the analysis on the object code is that bounding the worst case execution time of small code fragments in isolation (e.g. loop bodies) and then adding up the bounds yields very poor estimations because no knowledge on the hardware state can be assumed when executing the fragment. By analysing longer runs (e.g. by full unrolling loops) the bound obtained becomes more precise because the lack of knowledge on the initial state has less effects on longer computations.

In CerCo we propose a radically new approach to the problem: we reject the idea of a uniform cost model and we propose that the compiler, which knows how the code is translated, must return the cost model for basic blocks of high level instructions. It must do so by keeping track of the control flow modifications to reverse them and by interfacing with static analyzers. By embracing compilation, instead of avoiding it like EMBounded did, a CerCo compiler can at the same time produce efficient code and return costs that are as precise as the static analysis can be. Moreover, we allow our costs to be parametric: the cost of a block can depend on actual program data or on a summary of the execution history or on an approximated representation of the hardware state. For example, loop optimizations assign to a loop body a cost that is a function of the number of iterations performed. For another example, the cost of a loop body may be a function of the vector of stalled pipeline states, which can be exposed in the source code and updated at each basic block exit. It is parametricity that allows to analyze small code fragments without loosing precision: in the analysis of the code fragment we do not have to be ignorant on the initial hardware state. On the contrary, we can assume to know exactly which state (or mode, as WCET literature calls it) we are in.

The cost of an execution is always the sum of the cost of basic blocks multiplied by the number of times they are executed, which is a functional property of the program. Therefore, in order to perform (parametric) time analysis of programs, it is necessary to combine a cost model with control and data flow analysis. Current state of the art WCET technology performs the analysis on the object code, where the logic of the

program is harder to reconstruct and most information available on the source code (e.g. types) has been lost. Imprecision in the analysis leads to useless bounds. To augment precision, the tools ask the user to provide constraints on the object code control flow, usually in the form of bounds on the number of iterations of loops or linear inequations on them. This requires the user to manually link the source and object code, translating his often wrong assumptions on the source code to object code constraints. The task is error prone and, in presence of complex optimizations, may be very hard if not impossible.

The CerCo approach has the potentiality to dramatically improve the state of the art. By performing control and data flow analysis on the source code, the error prone translation of invariants is completely avoided. It is in fact performed by the compiler itself when it induces the cost model on the source language. Moreover, any available technique for the verification of functional properties can be immediately reused and multiple techniques can collaborate together to infer and certify cost invariants for the program. Parametric cost analysis becomes the default one, with non parametric bounds used as last resorts when trading the complexity of the analysis with its precision. A priori, no technique previously used in traditional WCET is lost (e.g. full unrolling for non parametric costs): they can just be applied on the source code.

Traditional techniques for WCET that work on object code are also affected by another problem: they cannot be applied before the generation of the object code. Therefore analysis of functional properties of programs already starts in early development stages, while when analysis of non functional properties becomes possible the cost of changes to the program architecture can already be very high. Our approach already works in early development stages by axiomatically atttaching costs to components that are not implemented yet.

All software used to verify properties of programs must be as bug free as possible. The trusted code base for verification is made by the code that needs to be trusted to believe that the property holds. The trusted code base of state-of-the-art WCET tools is very large: one needs to trust the control flow analyzer and the linear programming libraries it uses and also the formal models of the hardware. In CerCo we are moving the control flow analysis to the source code and we are introducing a non standard compiler too. To reduce the trusted code base, we implemented a prototype and a static analyzer in an interactive theorem prover, which was used to certify that the cost computed on the source code is indeed the one actually spent by the hardware. Formal models of the hardware and of the high level source languages were also implemented in the interactive theorem prover. Control flow analysis on the source code has been obtained using invariant generators, tools to produce proof obligations from generated invariants and automatic theorem provers to verify the obligations. If the automatic provers are able to generate proof traces that can be independently checked, the only remaining component that enters the trusted code base is an off-the-shelf invariant generator which, in turn, can be proved correct using an interactive theorem prover. Therefore we achieve the double objective of allowing to use more off-the-shelf components (e.g. provers and invariant generators) while reducing the trusted code base at the same time.

Summary of the CerCo objectives. To summarize, the goal of CerCo is to reconcile functional with non functional analysis by performing them together on the source code, sharing common knowledge about execution invariants. We want to achieve the goal implementing a new generation of compilers that induce a parametric, precise cost model for basic blocks on the source code. The compiler should be certified using an interactive theorem prover to minimize the trusted code base of the analysis. Once the cost model is induced, off-the-shelf tools and techniques can be combined together to infer and prove parametric cost bounds.

The long term benefits of the CerCo vision are expected to be:

1. the possibility to perform static analysis during early development stages
2. parametric bounds made easier
3. the application of off-the-shelf techniques currently unused for the analysis of non functional properties, like automated proving and type systems
4. simpler and safer interaction with the user, that is still asked for knowledge, but on the source code, with the additional possibility of actually verifying the provided knowledge
5. a reduced trusted code base
6. the increased accuracy of the bounds themselves.

The long term success of the project is hindered by the increased complexity of the static prediction of the non functional behaviour of modern hardware. In the time frame of the European contribution we have focused on the general methodology and on the difficulties related to the development and certification of a cost model inducing compiler.

4.1.3 Main S&T results

4.1.4 Potential impact, main dissemination activities and exploitation of results

4.1.5. Further informations

Further informations can be obtained from the project Web site:

http://cerco.cs.unibo.it

or contacting the project coordinator:

Project coordinator: Prof. Claudio Sacerdoti Coen
Alma Mater Studiorum – Università di Bologna
Tel: +39 051 2094973
Fax: +39 051 20 9 4510
E-mail: claudio.sacerdoticoen@unibo.it

4.2 Use and dissemination of foreground

A plan for use and dissemination of foreground (including socio-economic impact and target groups for the results of the research) shall be established at the end of the project. It should, where appropriate, be an update of the initial plan in Annex I for use and dissemination of foreground and be consistent with the report on societal implications on the use and dissemination of foreground (section 4.3 – H).

The plan should consist of:

- Section A

This section should describe the dissemination measures, including any scientific publications relating to foreground. **Its content will be made available in the public domain** thus demonstrating the added-value and positive impact of the project on the European Union.

- Section B

This section should specify the exploitable foreground and provide the plans for exploitation. All these data can be public or confidential; the report must clearly mark non-publishable (confidential) parts that will be treated as such by the Commission. Information under Section B that is not marked as confidential **will be made available in the public domain** thus demonstrating the added-value and positive impact of the project on the European Union.

Section A (public)

This section includes two templates

- Template A1: List of all scientific (peer reviewed) publications relating to the foreground of the project.
- Template A2: List of all dissemination activities (publications, conferences, workshops, web sites/applications, press releases, flyers, articles published in the popular press, videos, media briefings, presentations, exhibitions, thesis, interviews, films, TV clips, posters).

These tables are cumulative, which means that they should always show all publications and activities from the beginning until after the end of the project. Updates are possible at any time.

TEMPLATE A1: LIST OF SCIENTIFIC (PEER REVIEWED) PUBLICATIONS, STARTING WITH THE MOST IMPORTANT ONES

NO.	Title	Main author	Title of the periodical or the series	Number, date or frequency	Publisher	Place of publication	Year of publication	Relevant pages	Permanent identifiers[1] (if available)	Is/Will open access[2] provided to this publication?
1	*Economic transformation in Hungary and Poland'*		*European Economy*	*No 43, March 1990*	*Office for Official Publications of the European Communities*	*Luxembourg*	*1990*	*pp. 151 - 167*		yes/no
2										
3										

[1] A permanent identifier should be a persistent link to the published version full text if open access or abstract if article is pay per view) or to the final manuscript accepted for publication (link to article in repository).

[2] Open Access is defined as free of charge access for anyone via Internet. Please answer "yes" if the open access to the publication is already established and also if the embargo period for open access is not yet over but you intend to establish open access afterwards.

TEMPLATE A2: LIST OF DISSEMINATION ACTIVITIES								
NO.	Type of activities[3]	Main leader	Title	Date/Period	Place	Type of audience[4]	Size of audience	Countries addressed
1	*Conference*		*European Conference on Nanotechnologies*	*26 February 2010*				
2								
3								

[3] A drop down list allows choosing the dissemination activity: publications, conferences, workshops, web, press releases, flyers, articles published in the popular press, videos, media briefings, presentations, exhibitions, thesis, interviews, films, TV clips, posters, Other.

[4] A drop down list allows choosing the type of public: Scientific Community (higher education, Research), Industry, Civil Society, Policy makers, Medias, Other ('multiple choices' is possible).

Section B (Confidential[5] or public: confidential information to be marked clearly)
Part B1

The applications for patents, trademarks, registered designs, etc. shall be listed according to the template B1 provided hereafter.

The list should, specify at least one unique identifier e.g. European Patent application reference. For patent applications, only if applicable, contributions to standards should be specified. This table is cumulative, which means that it should always show all applications from the beginning until after the end of the project.

TEMPLATE B1: LIST OF APPLICATIONS FOR PATENTS, TRADEMARKS, REGISTERED DESIGNS, ETC.					
Type of IP Rights[6]:	Confidential Click on YES/NO	Foreseen embargo date dd/mm/yyyy	Application reference(s) (e.g. EP123456)	Subject or title of application	Applicant (s) (as on the application)

[5] Note to be confused with the "EU CONFIDENTIAL" classification for some security research projects.

[6] A drop down list allows choosing the type of IP rights: Patents, Trademarks, Registered designs, Utility models, Others.

Part B2

Please complete the table hereafter:

Type of Exploitable Foreground[7]	Description of exploitable foreground	Confidential Click on YES/NO	Foreseen embargo date dd/mm/yyyy	Exploitable product(s) or measure(s)	Sector(s) of application[8]	Timetable, commercial or any other use	Patents or other IPR exploitation (licences)	Owner & Other Beneficiary(s) involved
	Ex: New superconductive Nb-Ti alloy			*MRI equipment*	*1. Medical 2. Industrial inspection*	*2008 2010*	*A materials patent is planned for 2006*	*Beneficiary X (owner) Beneficiary Y, Beneficiary Z, Poss. licensing to equipment manuf. ABC*

In addition to the table, please provide a text to explain the exploitable foreground, in particular:

- Its purpose
- How the foreground might be exploited, when and by whom
- IPR exploitable measures taken or intended
- Further research necessary, if any
- Potential/expected impact (quantify where possible)

[719] A drop down list allows choosing the type of foreground: General advancement of knowledge, Commercial exploitation of R&D results, Exploitation of R&D results via standards, exploitation of results through EU policies, exploitation of results through (social) innovation.

[8] A drop down list allows choosing the type sector (NACE nomenclature) : http://ec.europa.eu/competition/mergers/cases/index/nace_all.html

4.3 Report on societal implications

Replies to the following questions will assist the Commission to obtain statistics and indicators on societal and socio-economic issues addressed by projects. The questions are arranged in a number of key themes. As well as producing certain statistics, the replies will also help identify those projects that have shown a real engagement with wider societal issues, and thereby identify interesting approaches to these issues and best practices. The replies for individual projects will not be made public.

A General Information *(completed automatically when **Grant Agreement number** is entered.*	
Grant Agreement Number:	243381
Title of Project:	Certified Complexity
Name and Title of Coordinator:	Prof. Claudio Sacerdoti Coen

B Ethics	
1. Did your project undergo an Ethics Review (and/or Screening)? • If Yes: have you described the progress of compliance with the relevant Ethics Review/Screening Requirements in the frame of the periodic/final project reports? Special Reminder: the progress of compliance with the Ethics Review/Screening Requirements should be described in the Period/Final Project Reports under the Section 3.2.2 *'Work Progress and Achievements'*	***No***
2. Please indicate whether your project involved any of the following issues (tick box) :	***YES***
RESEARCH ON HUMANS	
• Did the project involve children?	
• Did the project involve patients?	
• Did the project involve persons not able to give consent?	
• Did the project involve adult healthy volunteers?	
• Did the project involve Human genetic material?	
• Did the project involve Human biological samples?	
• Did the project involve Human data collection?	
RESEARCH ON HUMAN EMBRYO/FOETUS	
• Did the project involve Human Embryos?	
• Did the project involve Human Foetal Tissue / Cells?	
• Did the project involve Human Embryonic Stem Cells (hESCs)?	

• Did the project on human Embryonic Stem Cells involve cells in culture?	
• Did the project on human Embryonic Stem Cells involve the derivation of cells from Embryos?	
PRIVACY	
• Did the project involve processing of genetic information or personal data (eg. health, sexual lifestyle, ethnicity, political opinion, religious or philosophical conviction)?	
• Did the project involve tracking the location or observation of people?	
RESEARCH ON ANIMALS	
• Did the project involve research on animals?	
• Were those animals transgenic small laboratory animals?	
• Were those animals transgenic farm animals?	
• Were those animals cloned farm animals?	
• Were those animals non-human primates?	
RESEARCH INVOLVING DEVELOPING COUNTRIES	
• Did the project involve the use of local resources (genetic, animal, plant etc)?	
• Was the project of benefit to local community (capacity building, access to healthcare, education etc)?	
DUAL USE	
• Research having direct military use	No
• Research having the potential for terrorist abuse	

C Workforce Statistics

3. Workforce statistics for the project: Please indicate in the table below the number of people who worked on the project (on a headcount basis).

Type of Position	Number of Women	Number of Men
Scientific Coordinator	0	1
Work package leaders	0	5
Experienced researchers (i.e. PhD holders)	0	
PhD Students	0	
Other	2	0

4. How many additional researchers (in companies and universities) were recruited specifically for this project?	
Of which, indicate the number of men:	

D Gender Aspects

5.	**Did you carry out specific Gender Equality Actions under the project**?	✙	Yes No

6. Which of the following actions did you carry out and how effective were they?

	Not at all effective	**Very effective**
Design and implement an equal opportunity policy		○
Set targets to achieve a gender balance in the workforce		○
Organise conferences and workshops on gender		○
Actions to improve work-life balance		○

Other:

7. Was there a gender dimension associated with the research content – i.e. wherever people were the focus of the research as, for example, consumers, users, patients or in trials, was the issue of gender considered and addressed?

Yes- please specify

No

E Synergies with Science Education

8. Did your project involve working with students and/or school pupils (e.g. open days, participation in science festivals and events, prizes/competitions or joint projects)?

Yes- please specify

No

9. Did the project generate any science education material (e.g. kits, websites, explanatory booklets, DVDs)?

Yes- please specify

No

F Interdisciplinarity

10. Which disciplines (see list below) are involved in your project?

Main discipline[9]:

Associated disciplineError: Reference source not found:	○	Associated disciplineError: Reference source not found:

G Engaging with Civil society and policy makers

11a	**Did your project engage with societal actors beyond the research community?** *(if 'No', go to Question 14)*	○ ○	Yes No

11b If yes, did you engage with citizens (citizens' panels / juries) or organised civil society (NGOs, patients' groups etc.)?

No

Yes- in determining what research should be performed

Yes - in implementing the research

Yes, in communicating /disseminating / using the results of the project

[9] Insert number from list below (Frascati Manual).

11c In doing so, did your project involve actors whose role is mainly to organise the dialogue with citizens and organised civil society (e.g. professional mediator; communication company, science museums)?	○ ○	Yes No

12. Did you engage with government / public bodies or policy makers (including international organisations)

No

Yes- in framing the research agenda

Yes - in implementing the research agenda

Yes, in communicating /disseminating / using the results of the project

13a Will the project generate outputs (expertise or scientific advice) which could be used by policy makers?

Yes – as a **primary** objective (please indicate areas below- multiple answers possible)

Yes – as a **secondary** objective (please indicate areas below - multiple answer possible)

No

13b If Yes, in which fields?

Agriculture		Energy		Human rights	
Audiovisual and Media		Enlargement		Information Society	
Budget		Enterprise		Institutional affairs	
Competition		Environment		Internal Market	
Consumers		External Relations		Justice, freedom and security	
Culture		External Trade		Public Health	
Customs		Fisheries and Maritime Affairs		Regional Policy	
Development Economic and Monetary Affairs		Food Safety		Research and Innovation	
		Foreign and Security Policy		Space	
Education, Training, Youth		Fraud		Taxation	
Employment and Social Affairs		Humanitarian aid		Transport	

13c If Yes, at which level?

Local / regional levels
National level
European level
International level

H Use and dissemination

14. How many Articles were published/accepted for publication in peer-reviewed journals?	
To how many of these is open access[10] provided?	
How many of these are published in open access journals?	
How many of these are published in open repositories?	
To how many of these is open access not provided?	
Please check all applicable reasons for not providing open access:	
❑ publisher's licensing agreement would not permit publishing in a repository ❑ no suitable repository available ❑ no suitable open access journal available ❑ no funds available to publish in an open access journal ❑ lack of time and resources ❑ lack of information on open access ❑ other[11]: ……	

15. How many new patent applications ('priority filings') have been made? *("Technologically unique": multiple applications for the same invention in different jurisdictions should be counted as just one application of grant).*		
16. Indicate how many of the following Intellectual Property Rights were applied for (give number in each box).	Trademark	
	Registered design	
	Other	
17. How many spin-off companies were created / are planned as a direct result of the project?		
Indicate the approximate number of additional jobs in these companies:		

18. Please indicate whether your project has a potential impact on employment, in comparison with the situation before your project:

Increase in employment, or	❑	In small & medium-sized enterprises
Safeguard employment, or	❑	In large companies
Decrease in employment,	❑	None of the above / not relevant to the project
Difficult to estimate / not possible to quantify		

19. For your project partnership please estimate the employment effect resulting directly from your participation in Full Time Equivalent (*FTE = one person working fulltime for a year*) jobs:	*Indicate figure:*

[10] Open Access is defined as free of charge access for anyone via Internet.
[11] For instance: classification for security project.

Difficult to estimate / not possible to quantify	❑

I Media and Communication to the general public

20. As part of the project, were any of the beneficiaries professionals in communication or media relations?

Yes No

21. As part of the project, have any beneficiaries received professional media / communication training / advice to improve communication with the general public?

Yes No

22 Which of the following have been used to communicate information about your project to the general public, or have resulted from your project?

Press Release	❑	Coverage in specialist press
Media briefing	❑	Coverage in general (non-specialist) press
TV coverage / report	❑	Coverage in national press
Radio coverage / report	❑	Coverage in international press
Brochures /posters / flyers	❑	Website for the general public / internet
DVD /Film /Multimedia	❑	Event targeting general public (festival, conference, exhibition, science café)

23 In which languages are the information products for the general public produced?

Language of the coordinator	❑	English
Other language(s)		

***Question F-10*:** Classification of Scientific Disciplines according to the Frascati Manual 2002 (Proposed Standard Practice for Surveys on Research and Experimental Development, OECD 2002):

FIELDS OF SCIENCE AND TECHNOLOGY

1. NATURAL SCIENCES

1.1 Mathematics and computer sciences [mathematics and other allied fields: computer sciences and other allied subjects (software development only; hardware development should be classified in the engineering fields)]

1.2 Physical sciences (astronomy and space sciences, physics and other allied subjects)

1.3 Chemical sciences (chemistry, other allied subjects)

1.4 Earth and related environmental sciences (geology, geophysics, mineralogy, physical geography and other geosciences, meteorology and other atmospheric sciences including climatic research, oceanography, vulcanology, palaeoecology, other allied sciences)

1.5 Biological sciences (biology, botany, bacteriology, microbiology, zoology, entomology, genetics, biochemistry, biophysics, other allied sciences, excluding clinical and veterinary sciences)

2 ENGINEERING AND TECHNOLOGY

2.1 Civil engineering (architecture engineering, building science and engineering, construction engineering, municipal and structural engineering and other allied subjects)

2.2 Electrical engineering, electronics [electrical engineering, electronics, communication engineering and systems, computer engineering (hardware only) and other allied subjects]

2.3. Other engineering sciences (such as chemical, aeronautical and space, mechanical, metallurgical and materials engineering, and their specialised subdivisions; forest products; applied sciences such as geodesy, industrial chemistry, etc.; the science and technology of food production; specialised technologies of interdisciplinary fields, e.g. systems analysis, metallurgy, mining, textile technology and other applied subjects)

3. MEDICAL SCIENCES

3.1 Basic medicine (anatomy, cytology, physiology, genetics, pharmacy, pharmacology, toxicology, immunology and immunohaematology, clinical chemistry, clinical microbiology, pathology)

3.2 Clinical medicine (anaesthesiology, paediatrics, obstetrics and gynaecology, internal medicine, surgery, dentistry, neurology, psychiatry, radiology, therapeutics, otorhinolaryngology, ophthalmology)

3.3 Health sciences (public health services, social medicine, hygiene, nursing, epidemiology)

4. AGRICULTURAL SCIENCES

4.1 Agriculture, forestry, fisheries and allied sciences (agronomy, animal husbandry, fisheries, forestry, horticulture, other allied subjects)

4.2 Veterinary medicine

5. SOCIAL SCIENCES

5.1 Psychology

5.2 Economics

5.3 Educational sciences (education and training and other allied subjects)

5.4 Other social sciences [anthropology (social and cultural) and ethnology, demography, geography (human, economic and social), town and country planning, management, law, linguistics, political sciences, sociology, organisation and methods, miscellaneous social sciences and interdisciplinary , methodological and historical S1T activities relating to subjects in this group. Physical anthropology, physical geography and psychophysiology should normally be classified with the natural sciences].

6. HUMANITIES

6.1 History (history, prehistory and history, together with auxiliary historical disciplines such as archaeology, numismatics, palaeography, genealogy, etc.)

6.2 Languages and literature (ancient and modern)

6.3 Other humanities [philosophy (including the history of science and technology) arts, history of art, art criticism, painting, sculpture, musicology, dramatic art excluding artistic "research" of any kind, religion, theology, other fields and subjects pertaining to the humanities, methodological, historical and other S1T activities relating to the subjects in this group]

2. FINAL REPORT ON THE DISTRIBUTION OF THE EUROPEAN UNION FINANCIAL CONTRIBUTION

This report shall be submitted to the Commission within 30 days after receipt of the final payment of the European Union financial contribution.

Report on the distribution of the European Union financial contribution between beneficiaries

Name of beneficiary	Final amount of EU contribution per beneficiary in Euros
1.	
2.	
n	
Total	