TrustCoM

Deliverable

74

Lessons Learned

WP19 General Management

Ignacio Soler, Atos Origin

May 2007

V2.0

Trustcom

A trust and Contract Management framework enabling secure collaborative business processing in on-demand created, self-managed, scalable, and highly dynamic Virtual Organisations

SIXTH FRAMEWORK PROGRAMME

PRIORITY IST-2002-2.3.1.9

Prepared for the European Commission under FP6 Contract No. 01945 as a deliverable form

LEGAL NOTICE

The following organisations are members of the Trustcom Consortium:

Atos Origin Sociedad Anonima Espanola,
Council of the Central Laboratory of the Research Councils,
BAE Systems,
British Telecommunications PLC,
Universitaet Stuttgart,
SAP AktienGesellschaft Systeme Anwendungen Produkte in der Datenverarbeitung,
Swedish Institute of Computer Science AB,
Europaeisches Microsoft Innovations Center GMBH,
Eidgenoessische Technische Hoschschule Zuerich,
Imperial College of Science Technology and Medicine,
King's College London,
Universitetet I Oslo,
Stiftelsen for industriell og Teknisk Forskning ved Norges Tekniske Hoegskole,
Universita degli studi di Milano,
The University of Salford,
International Business Machines Belgium SA .

Neither the Trustcom Consortium, any member organisation nor any person acting on behalf of those organisations is responsible for the use that might be made of the following information.

The views expressed in this publication are the sole responsibility of the authors and do not necessarily reflect the views of the European Commission or the member organisations of the Trustcom Consortium.

All information provided in this document is provided 'as-is' with all faults without warranty of any kind, either expressed or implied. This publication is for general guidance only. All reasonable care and skill has been used in the compilation of this document. Although the authors have attempted to provide accurate information in this document, the Trustcom Consortium assumes no responsibility for the accuracy of the information.

Information is subject to change without notice.

Mention of products or services from vendors is for information purposes only and constitutes neither an endorsement nor a recommendation.

Deliverable datasheet

Project acronym: TrustCoM

Project full title: *A trust and Contract Management framework enabling secure collaborative business processing in on-demand created, self-managed, scalable, and highly dynamic Virtual Organisations*

Action Line: AL8 Project Management

Activity: Activity 8.3 Overall Coordination

Work Package: WP19 General Management

Document title: Lessons Learnt

Version: v2.0

Document reference:

Official delivery date: May 2005

Actual publication date: May 2005

File name: D74 Lessons_Learned.doc

Type of document: Report

Nature: Public

Authors: Ignacio Soler(ATOS ORIGIN) with contributions from all Trustcom partners

Reviewers: UoM, SICS

Approved by: Trustcom Consortium

Version	Date Date	Sections Affected
V.1	31/07/04	All
V2.0	31/05/2007	Introduction, 2.4 Finalising an Integrated Project

Table of Content

1 Introduction

This document is an update of the previous D26 document, some of the thoughts from the partners are reflected in this document, it pretend to give an overall vision of how the projects have to be managed.

The project TrustCoM is now finalising, and we are up to re-think our vision of what has been the project, which are the aspects that we have learnt, and how things could be better done in the future applying the know–how to next coming projects in FP7.

It is time to reflect on the good things and in the bad things that we have experienced and try to extract some lessons learnt that could help us improve our performance as a group, pursuing the planned objectives in a collaborative way and as individuals, enhancing our skills and our personal development.

The aim of this document is precisely to list these lessons learnt in various facets of the project (management level, technical level, communication among partners, etc).

1.1 TrustCoM at a glance

Trustcom will develop a framework for trust, security and contract management in dynamically-evolving virtual organisations. The framework will enable secure collaborative business processing within on-demand created and self-managed, dynamic collaborative networks of businesses and governments built on top of the emerging convergence of Web Services, agent and Grid technologies.

Recent years have seen an unprecedented acceleration in the evolution of the Internet as the technological vehicle underpinning the expansion of service provision and inter-/intra- enterprise integration in all market sectors. This brings about the prospect of ad hoc integration of systems across organisational boundaries to support collaborations that may last for a single transaction or evolve dynamically over many years. This sets new requirements for scalability, responsiveness and adaptability that necessitate the on-demand creation and self-management of dynamically evolving virtual organisations (VO) spanning national and enterprise borders, where the participating entities (enterprises or individuals) pool resources, information and knowledge in order to achieve common objectives.

The provision of cost effective trust and contract management solutions for VO ecosystems that enable secure collaborations in such VOs is the most demanding and timely research challenge in this field. In order to achieve this goal, TrustCoM will need to conduct multidisciplinary applied research into complex, adaptive and self-organising systems applied to VO trust and security. The TrustCoM consortium provides a balanced blend of academic and applied researchers, end-user organisations, and enterprises looking to utilise results in products and services. As such it is well-placed to define, conduct and exploit leading edge research that is relevant to the needs of European business, government and society.

1.2 Trustcom in numbers

Number of partners	16
Countries represented	9
Duration	3 years
Current Timeline	Month 40 (end of the project)
Total Person-Months	991
Total Costs	10.906.703,92
Total Funding	6.299.998,99

1.3 Structure of the document

This document gathers the feedback, opinion, thoughts and suggestions of the consortium members that consolidates a list of lessons learned during the full duration of the project and that, hopefully, will become a good guideline of improvement for other EC projects.

The method to gather the inputs from the partners was a simple questionnaire addressing 4 relevant topics in the project which can be seen as four different working levels (management, technical operations, conceptual and research operations and communication). The email was the channel to gather the requested feedback.

Following the structure of the questionnaire provided to the partners, this document has been structured in four sections that cover the different topics chosen.

These topics are:

- Management level: issues related to project management (financial statements, contract and legal issues, quality control, work plan, management reporting, IPR, WP structure, management decisions, etc)
- Technical level: basically lessons learned from the technical implementation in Trustcom (AL2 + AL3)
- Conceptual level: lessons learned from the non technical action lines (good comprehension of the project, framework and architecture, socio-economic and legal aspects, exploitation, etc)
- Communication level: (meetings, communication with other partners, collaborative tools, etc).

The way in which these general impressions are presented consist on listing the positive and negative aspects related to the topic and a final summary table highlighting the principal lessons learned.

2 Management Level

This section will report the problems and feedback gathered from the project partners at management level, i.e. financial statements, contract and legal issues, quality control, work plan, management reporting, IPR, WP structure, management decisions, etc).

2.1 Structure of the consortium

- ☺ Experienced partners (in big projects and in IST projects)
- ☺ Strong business partners that can take an important role in the future impact and commercialisation of the project outcomes
- ☺ Many countries participating. Cultural interchange is a positive experience.
- ☺ Many partners did already work in the past in others EC projects. It is very important for the trust and confidence for the future work.
- ☺ Partners changed their legal structure (M&A) during the project which surprisingly did not cause changes to their approach to the project.
- ☺

- ☹ Small experience with Integrated Projects.
- ☹ Administrative management is highly time consuming
- ☹ The size of the consortium can become not only an organisational problem (programme management) but also a logistic problem (meetings, etc).
- ☹ Project focus and alignment produced several face-to-face meetings and overspending of travel budget in some cases
- ☹ Many countries participating. Different cultures, different languages sometimes lead to misunderstandings
- ☹ Staff on certain partners change during the project. This issue cause misunderstandings, and cause repetition of discussions already done by the previous staff
- ☹ The consortium was structured with partners in technical roles, non-technial roles and some partners as the interface between the two. There was no motivation for partners to fulfil this interface role so it didn't happen.
- ☹ The balance between technical and non-technical partners as well as between research and commercial exploitation staff in the project resulted in technical issues and motivations dominating the operation of the IP over legal and socio-economic research or exploitation activities.

- ☹ The software evaluations would have benefited from more information on what the reviewers were expecting- the level of evaluation and so on. This is a customer-relationship issue that doesn't appear to have been managed very well.

Lessons Learned
There is never enough travel budget
Important that many partners already worked together in the past in other EC projects. This strengthen trust and commitment
Staff of partners changed to a long the project, and sometimes is difficult to have an overall vision of the project in some partners.

2.2 Before the start

- ☺ A very positive feeling and commitment from the majority of partners from the proposal preparation phase. This is a good start!

- ☹ Very long legal discussions about Consortium Agreement took place before the start of the project. These discussions continued when the project already started.
- ☹ Consortium Agreement took a long time to agree; that put the project on hold while it was under negotiation, the resulting document is still not accepted by the EC since it includes statements about international affiliate organisations that they do not approve of. It is also unclear that it addresses the serious issues of the project enforcing work on partners they do not want, or taking funding away from partners for failing to complete work to the required standard.

Lessons Learned
Feelings about how the project can progress successfully are initially seen at the proposal preparation phase. In the case of Trustcom it was really positive.
The consortium really should agree the consortium agreement before the contract with the EC is agreed, because by then the lead partner looses control.

2.3 When starting

- ☺ A quality assurance plan has been helpful to define responsibilities in internal review of project deliverables (external and internal)
- ☺ From the beginning of the project there have been document templates for most of the papers produced (deliverables, letters, fax, etc)
- ☺ “To Do” List of all activities in the project shown in the Project Portal
- ☺ A List of agreed tools for operation was decided at the beginning (Office 2000, Project Portal, etc)
- ☺ Generation of project presentations to better explain and present the project goals and abstract to the world and to brief/induct new workers within the partners

---

- ☹ Several versions of the internal and external deliverable templates were made in order to capture most of the requirements given by the partners, part of them raised by problems using the template.
- ☹ Regarding work plan and work package structure, TrustCoM exhibited a need for restructuring very soon. Instead of targeting the issue immediately, it led to quite a bit of confusion and compensation mechanisms. For instance the number of face2face meetings and conference calls grew rapidly, but those measures were not able to cope with the problem. As a bottom-line, the ability to restructure a project's schedule is a good thing, PM should be able to detect the necessity to do so as early as possible and encourage partners. To avoid a planning "vacuum", PM needs to finalise the new schedule quickly, mainly in order to avoid confusion along which lines of planning partners should work and report.
- ☹ We should definitely not have major themes spread across different conceptual, architecture, design and implementation workpackages. Workpackages should be split thematically and not based on the "type of work" required. This cause divisions and imbalances in the project team, as some people were able to move towards development quicker than others, without having reached conceptual or integration agreement.

Lessons Learned
The coordinator must provide templates at the very beginning of the project
The start of the project is always costly for the coordinator when producing all the documentation that will become the basis for future work
Ensure deliverables fall on the review period boundaries and do not cross them
In future, presentations should be consistent with the documents presented for review so that they do not have to be updated.
Workpackages should be split thematically and not based on the "type of work"

required
For the second phase of the project management should develop more efficient methods to better integrate/discard the partners who are performing under the threshold. It is a challenge to work with these partners

2.4 Running the project

☺ A public web site available at the start of the project is also a good tool to develop the project identity. There was a public website available from the proposal phase which then became the official website of the project.

☺ A good kick-off meeting is important to establish a good basis of cooperation. The following actions became very helpful:

- All partners attended the kick-off meeting. This provided a good opportunity to know who is who, the role in the project and the expectations
- General rules for performance, communication and initial responsibilities
- A social event is necessary as ice-breaker

☺ WP structure along logical units more sensible than along "tasks".

☺ Implementation and design can be logically separated, but equal knowledge-distribution is mandatory

---

☹ A glossary of common terms would have been very useful from the beginning. It was discussed but never put in practice (a model for the glossary was presented for its inclusion as part of the portal but there were no appropriate resources available at that time for the task). This led to different interpretations of concepts in some occasions.

☹ The reporting cycle of deliverables, annual review and next 18 month plan approval, resulting in approved payment appears out of step.

☹ The reviewers found it confusing that they were given presentations at a review that advanced the position over that in the documents they had been sent 3 months earlier.

Lessons Learned
Deliverables with multiple contributors must be also reviewed by an English native person
Empowering partners (WP leaders) enforces commitment in the project

2.5 Finalising an Integrated Project

- ☺ Creation of a new commercial website, that will ensure the dissemination of TrustCoM at least 1 year after the finalization of the project.
- ☺ Release of the software being hosted on the sam website previously mentioned.
- ☹ Deadlines for delivery of management reports (e.g. PAR/PMR) received on very short notice. Solution: send request at least 4 weeks before deadline to make people aware.
- ☹ Project portal is difficult to use and slow

3 Technical Level

This section summarises the feedback and thoughts gathered from the partners basically from the technical work (including technical management) in TrustCoM.

- 🙂 The overall architecture focused more on comprehensiveness than on UML compliance, which increased understanding significantly.
- 🙂 Collaboration between Action Line 1 (architecture) and Action Line 2 (design) was very good. The segmentation into the two action lines is very helpful, given a sufficient overlap of partners to increase cross action line communication.
- 🙂 The .NET community is constantly increasing and showing a growing interest in supporting a wide range of standards. .NET, as opposed to Java, showed high flexibility regarding interoperation between different framework versions which made updating more efficient.
- 🙂 Aligning the integration process on sub-"demos" increased the efficiency of the development process. This meant in particular that with each integration a diminishing amount of sub-demos were specified that pursued an increasing amount of component / functionality integration.
- 🙂

- 😐 The top-down / bottom-up distinction between AL1 and AL2 caused confusion in the beginning of the project. It bears the great risk of the two ends not meeting in the middle. This caused the work package reorganisation at the beginning of the project.

  In effect, the approach has to take a middle way which results naturally from the strong overlap in partners between action lines: the "pure" architecture work consisted in analysing the relationship between the technological areas (security, SLA etc.), and forwarding the interaction requirements accordingly and gather feedback from implementation. Within each technological area, the required functionalities and their relationships are first defined (by people involved in both action lines) and then verified & realised through action line 2.

  The respective results were then fed back to action line 1 to verify and / or update the according relationships.
- 😐

- ☹ Java is currently far less supported in the web services domain than .NET. Consequences for the project: If Java is less supported then industrial partners' exploitations will be based on other platforms. Results and Java artefacts from academic partners might not be easy to exploit at the "commercial" level.
- ☹ In such big projects as IPs it is difficult to align collaborative technical work. In TrustCoM, subsystems and testbed scenarios were identified and development started along those lines. Now, the problem becomes visible, how to integrate the work conducted in separate subsystems.

- ☹ On an even deeper technical level, it became apparent that even implementations based solely on comparably mature WS-Standards, such as WSDL, UDDI, and BPEL are not that easily interoperating. Even the specification of a complex message type in a WSDL, when transmitted in a SOAP message requires careful consideration of message encoding. In IP project, design must be done having interoperability in mind; developing for interoperability using incomplete, not compatible standards, may lead to the risk of either “wasting” resources in debugging a particular standard implementation or to not implement some feature simply because it is not supported by available standards.
- ☹ We should have been open and honest from a very early stage in the project about our software development resources and expectations. It seems as though the quality of software being produced and standards being affected is not that high, in comparison to the claims of the project. Rather, there is a lot of documentation that informally and verbally describes visions of Virtual Organizations. The move to more formal representations either came too late or was rushed, as we tried to get to a stage of producing software, XML documents and info-sets. We however managed to come up with some very convincing scenarios for how the vast amount of WS standards could be integrated and applied.
- ☹ Some of the more important AL2 decisions have not been clearly communicated. One example is at the last AL2 meeting when the Scenarios WP leaders were not told about the demo plans for the two industrial test beds until they began their sessions. This was very embarrassing.
- ☹ Integration has been a definite handicap, that had to be thought before hand. An integration environment would have been useful for that purpose.At the start there was no partner leading overall development. That resulted in a lot of ad-hoc solutions and erratic effort once such partner was finally located
- ☹ The individual development works were not communicated properly which led to confusion between different versions and interfaces. This led to high integration efforts. Future approaches should in particular try to use consistent interface declarations (i.e. with backward compatibility or no changes at all), or at least ensure that all according changes are communicated timely and thoroughly.
- ☹ Architecture and development existed “separately” until circa the Autumn of 2005, i.e. up to Month 19-20
- ☹ Very high integration efforts. Solution: Avoid changes of agreed interfaces.
- ☹ Mutual technical understanding difficult because of changing interfaces and missing documentation. Improvement: Intermediate software documentation should be produced
- ☹ The project did not start with even a basic architecture. Paul Kearney has made this point in the past and I agree with him here. A basic implementation architecture only began to emerge in October 2004 at the Aachen meeting and only became more mature in early 2005. This was a year after the project start!

- ☹ There was too much emphasis on the Review software demonstrations. There should have been more emphasis on integrated releases that one could identify as 'TrustCoM Version 1', 'TrustCoM Version 2' etc that had definite functionalities, however limited. In other words, intermediate *release* versions should have been made available not only to gather feedback from the community, but also to ensure rollback points and enhance backward compatibility (cf. above)
- ☹ Software documentation for each component was in some cases very poor. Although the written deliverables contained information on the software components at a technical level, there was not much documentation that described how to install it, test the installation and so on. The project would have benefited from documenting the installation and usage of the components from an early stage. It may have improved the communications between the partners as well.
- ☹ Although I can't directly prove it, there seemed to be a reluctance by some of the partners to install components from other partners. I got the impression that some of the partners were too focussed on developing their own components.

  It has to be stressed that this point relates to the issues of intermediate release versions and a common, “shareable” integration platform (such as Virtual PC). I understand that the partners did not so much focus *only* on their components, but that installation of other components was difficult and / or clashed with individual framework setups (cf. Java compatibility issues). Accordingly a stronger focus in future projects should *also* rest on easier and more flexbile installation routines, as well as commonly agreed infrastructure setups.
- ☹
- ☹ I now understand that there has been two approaches in this project: SAP with their 'top-down' approach and BT with their 'bottom-up' approach to implementing a VO. The biggest integration problems concerned matching the two systems. I think the two teams would have benefited from getting together in a single location, installing each others software, understanding their objectives and problems and sorting out a solution!
- ☹ There was no genuine TrustCoM release cycle. This made application of the software to the two test beds difficult. Evaluation was also compromised because there was no identifiable software to focus on and evaluate.
- ☹ One challenge was adopting two web service platforms- Java and .NET. Integrating certain components (STS, PEP in particular) became too wrapped up in low-level integration issues that had no research significance. Standardising on a single platform was perhaps unrealistic, but perhaps standardising on a single platform for certain critical parts, in particular with a strong interdependency, would have been preferable.
- ☹ Using a solution that is a mixture of two service platforms is a pain to the end-user as well! I'm thinking of the crypto store problems between Java and .NET... This might limit take-up of the TrustCoM RI.
- ☹ There was no analysis of the non-functional requirements defined for the CE Test Bed in 2004 to generate the functional requirements of the sub-

systems. This should have been done and documented in some detail. Perhaps communications would have been improved if this had been done. This is the biggest lesson for WP35.

- ☹ There was a reluctance by certain large commercial partners to employ a consensual approach to defining the exact technical profiles to be used. Their slightly bullying attitude seemed to be “we have done it this way, take it or leave it” or “we will not use standard X at all”.
- ☹ Some partners preferred to implement a whole system themselves and ignore the components produced by others rather than integrate those components, thereby making the work of the junior partner unnecessary to the project.
- ☹

☹ Lessons Learned
☹ The project should make the relationship between conceptual models and design more clear.
☹ We should try to get common UML representations of architectures and design details agreed earlier.
☹ Early "detection mechanisms" to avoid incompatible components later on are direly needed
Interoperability tests need a more timely plan
Partners must be open and honest from a very early stage about the software development resources and expectations
Avoid making important decisions that may affect other parties without previous discussion about intentions and plans.
Integration has to be thought through before hand.
The Lead development partner should be clearly identified beforehand. This partner should have strong experience in industrial software development and/or big and distributed software development project.
Partners should be more prepared to develop software in collaboration: adhere to once agreed interfaces, keep agreed development cycles and so on
Shareable and exchangeable common development / testing platforms, such as Virtual PC and VMWare should have been used more strongly.
All development processes need to be communicated timely. Intermediary “release” versions and sufficient backward compatibility, or at least restricting *interface* adaptations to a minimum, is a necessity
Regular “coding” sessions involving the development team are a necessary means to ensure integration

4 Conceptual and operational level

This section collects all the feedback around lessons learned from the non technical action lines (good comprehension of the project, framework and architecture, socio-economic and legal issues, exploitation, etc).

- ☺ The initial legal work studied rather abstract questions and was a too detached from the rest of the project. To ensure that the legal work package was aligned with the rest of the project, a strong focus was needed on the TrustCoM test bed scenarios, which form a “glue” between the different work packages and Action Lines. The objectives and standards of achievement of those working on legal and socio-economic topics were different to those of the technical researchers, and unclear. It was not enough to take for granted that all staff had the same motivation across disciplines.
- ☺ The technical researchers wanted simple output as algorithms, parameters etc from the non-technical work which those working in socio-economic and legal issues did not see as appropriate outputs for them. The technical researchers were not interested in the issues which the socio-economic and legal researchers considered important.

- ☺ The legal risk analyses of the TrustCoM scenarios involved people with technical, socio-economic and legal expertise. The participants from other workpackages were motivated and helpful and provided very useful input to the legal work. This multi-disciplinary approach to legal risk analysis proved to be fruitful and facilitated an integrated analysis of security-, trust- and contract-related issues, leading to identification of risks and possible treatments which might not have been identified if the issues had been addressed separately.
- ☺ Common methodology and the corresponding templates with concrete recommendations and steps developed for the exploitation plan were very helpful to develop a consistent document and to include a lot of details into it.
- ☺ After initial difficulties with the conceptual work, the framework has shown strong improvements and wide acceptance. Also, the collaboration between implementation and design has greatly improved with realigning the project work package structure to reflect the technological areas.
- ☺ The integration of legal aspects into technical and design aspects of the project provided interesting and important input.

- ☹ The understanding of the benefits of the game theory work to the rest of the project have not been made clear, so much so that the project and reviewers agree that this line should be dropped in favour of business modelling for dynamic VOs.
- ☹ Through realigning the legal work package with the TrustCoM scenarios we were able to identify and focus on the legal issues that were of importance to the rest of the project.

- ☹ The use of methods and tools from model-based risk analysis, and in particular UML which is also a basis for much of the technical work in TrustCoM, facilitated communication and understanding between the different disciplines involved in the legal risk analyses.
- ☹ Partners had concerns about confidentiality with exploitation plans. Solution: 1) make exploitation plan confidential 2) partners can provide their individual exploitation plans individually to the commission.
- ☹ BAE should have involved internal contracts and legal specialists to comment in more detail on the CE application scenarios. This was done to a limited extent in early 2005 and into 2006. It would have been better to have done this earlier in the project in 2004.

Lessons Learned
Have a thematic view vs a “type of work” vision will engage more the technical issues with the socio-economic and legal issues
Increase effort in monitoring Action Lines, specially cross-action lines aspects to give a more coherent and consistent view
TrustCoM addresses a problem of a multi-disciplinary nature, however integration of these disciplines can be challenging. This requires not only motivated participants, but also good methods, languages and tools to facilitate collaboration across disciplines

5 Communication Level

This section gathers all the feedback from the project partners around the communication issue, i.e. meetings, communication with other partners, collaborative tools, etc.

- ☺ Communication level: Having a source code control system (subversion) for document sharing across project partners is a very helpful thing. People have copies of all necessary documents being pushed to their machines, all partners work on the same documents, all partners see the same documents, people don't have to go to web pages and pull new information and documents. Just like e-mail, documents arrive locally. Being able to link via a URL to the latest version of that document is very helpful, too.
- ☺ Document-sharing online meetings support conference call discussions. Using SubVersion for document sharing was a very good idea as well.
- ☺ Using tools like LiveMeeting for information sharing during conf calls was another very good thing
- ☺ Running development coordination through weekly conf calls proved to be effective and successful
- ☺ Regular Development workshops were quite effective. We could use more of them
- ☺ Good cooperation and working climate.
- ☺ External communication was driven by the researchers' desire to publish conference and journal papers which produced a large number of outputs to the technical community. Document-sharing online meetings support conference call discussions
- ☺

- ☹ Meetings should be organised on Action Line level
- ☹ It was originally planned that the portal would be used as a means of entering exact financial data by administrative staff in partners. Such staff is not willing to do this, so it can only be used to enter approximate effort and cost figures to be used for management planning and not for reporting.
- ☹ It may be that the lack of project focus is due to the lack of full project meetings or their poor leadership. However, it is very hard to get the right agenda for all parties to join in with in order to generate focus.
- ☹ In the beginning of the TrustCoM project, none of the partners was experienced with IP work. This lead to a considerable amount of "experimenting" among the

partners, varying, but growing, numbers of meetings and conference calls were the result. The level of agenda, WP or AL varied as well. In the end, the result which proved most beneficial for project work was regular bi-weekly concalls on Action Line level and WP conference calls on demand in between. Meetings settled to a comfortable number every 1 or 2 month for bigger integration issues. Still, it is difficult to organise meetings with such a large number of participants.

- ☹ The Portal used for technical communication in the beginning was quite cumbersome to use and not very useful for collaborative document work with more than 2 or 3 editors. The introduction of subversion improved the situation a lot. Technically, the file repository can be accessed cross-domain from each partner and keeping track of edits is quite simple.
- ☹ Several issues could have been solved via email, had we been more "formal" in our discussions as opposed to too much free-form text and face-to-face meetings. The most effective meetings were those where status updates on different aspects were given. Attempts at doing conceptual or design work over the telephone failed. Face-to-face meetings had too many participants and should have been a bit more disciplined with the agenda and goals to be achieved at each sitting.
- ☹ With some notable exceptions the AL2 face-to-face meetings are not very well organised. A better defined agenda, better time-keeping and a stronger meeting management style should be adopted. We need to avoid meetings where there are only debates between a few people while everyone else watches.... Items tend to overrun very badly as well.
- ☹ Breakout sessions tend to be much better focussed and so these should be adopted as the norm. The October meeting at EMIC showed how things could be done.
- ☹ Developers Meetings sometimes tend to lose focus and to be stuck in theoretical discussion. Improvement: Clear focus on "running software" from the beginning of the meeting.
- ☹ We had great difficulty understanding all of the technical issues and how the components could have been applied. I would have had an easier time if the software was documented (installation, testing etc) and had the basic concept explained. We could then have installed it, tried it out and by using it understand better how to apply it.
- ☹ We think many of the AL2 meetings were un-productive. There seemed to be occasions where we all be watching two partners having an argument while everyone else watched... Perhaps having smaller, more focussed meetings (eg, between SAP and BT indicated above) may have improved matters.
- ☹ It could help to keep the agreements reached during Development Workshops (like message sequence, etc)
- ☹ It was hard to determine who should be the right audience for external communication when the objectives of the IP as a whole did not match the objectives of any partner organisation or individual in the project. The technical

community addressed by researchers was not the only audience required to establish TrustCoM as influential for standardisation and exploitation by partners.

☹ It was hard to communicate to non-research staff in project member organisations that TrustCoM was and R&D project and not a joint venture producing a single commercial product for exploitation. This resulted in the requirement for cumbersome legal statements on TrustCoM public outputs to overcome the misunderstanding.

Lessons Learned
Subversion has become a very useful tool for file sharing. Edit tracking functionality and the possibility to work with the same document by several users was one of the reasons to shift from ProjectPortal to Subversion.
ProjectPortal became a collaborative tool for keeping a common agenda, as a repository for final version of documents and as an address book with all the participants in the project rather than a repository of working documents. Subversion showed a better performance for this issue (edit tracking, version control, etc).
Face-to-face meetings should be clearly designed to have the correct number of people and a very specific agenda on goals to achieve.
Break out sessions help focus on the objectives at the meetings
Avoid meetings that turn into a discussion between only a few people