EUROPEAN COMMISSION

Thematic Priority:
SIXTH FRAMEWORK PROGRAM

Priority 2.5.3
INFORMATION SOCIETY TECHNOLOGIES
Unit G3 Embedded Systems

Project Acronym:

SOCRADES

Project Full Title:

Service-Oriented Cross-layer infRAstructure for Distributed smart Embedded devices

Proposal/Contract No: EU FP6 IST-5-034116 IP SOCRADES

Deliverable D10.4a

(Technology) Roadmap of the SOCRADES paradigm

Status:	**Final**
Dissemination Level[1]:	**CONFIDENTIAL**
Date:	**11.04.2008**

Organization Name of the Lead Contractor for this Deliverable: Politecnico di Milano

[1] See Annex for explanation of Dissemination Levels, as defined in the DoW

Status Description:

Scheduled completion date[2]:	29.02.2008	Actual completion date[3]:	11.04.2008
Short document description:	This document presents the first release of SOCRADES Technology Roadmap. The roadmap presents not only a vision on how the results will be used, but also the identification of the limit of the developed technology. A long term view on expected feature in technology is shown.		
Author(s) deliverable:	Marco Taisch (POLIMI), Alessandro Cannata (POLIMI), Marco Gerosa (POLIMI)	**Report/deliverable classification:** [X] Deliverable [ ] Three-Monthly Activity Report [ ] Six-Monthly Activity Report	
[ ] [ ] Partner ↓ Contributions ↓ Peer reviews	[X] [ ] Schneider Electric [ ] [ ] ABB [ ] [ ] APS GmbH [ ] [ ] Boliden AB [ ] [ ] FlexLink Components AB [ ] [ ] Institut f. Automation und Kommunikation e.V. Magdeburg [X] [ ] Kungliga Tekniska Högskolan	[X] [ ] Loughborough University [ ] [ ] Luleå University of Technology [X] [ ] Politecnico di Milano [X] [ ] SAP AG [X] [ ] Siemens AG [X] [ ] Tampere University of Technology [ ] [ ] Jaguar Cars Ltd. [ ] [ ] ARM Ltd.	
Peer review approval :	[X] Approved [ ] Rejected (improve as specified hereunder)	Date:	31.03.2008
Suggested improvements:			

[2] As defined in the DoW

[3] Scheduled date for approval

Table of Contents:

List of Figures:

Executive Summary (POLIMI)

"It is very difficult to make an accurate prediction, especially about the future." Niels Bohr, Nobel laureate in Physics

"The best way to predict the future is to invent it". Alan C. Kay, Computer Scientist

As reported in the letter to EU Reviewers on the 26th September 2007 and accepted in the Review Meeting in Nuremberg (8-9 October 2007), Task 10.3 changed its aim from "Road mapping for the adoption of the SOCRADES paradigm" to "Road mapping of the SOCRADES paradigm"; hence the new objective is to develop a Technology Roadmap of the SOCRADES paradigm and the related technologies.

As a consequence, a characteristic of the roadmap is its long term view on future advancement in technology, being able to give to the project not only a vision on how the results will be used, but also the identification of the limit of the developed technology. Therefore, this document mainly proposes and suggests some expected feature that may overcome the limit of SOCRADES itself.

The document is structured in this way: first, in the introduction, a general overview is given concerning the roadmap definition and methodology adopted. Then some preliminary information is given concerning roadmap justification, targeted audience, partner involved and the main Technology Areas identified as boundaries for the road mapping exercise. In the third section a brief reference to the state-of art analysis in the main Technology Areas involved in SOCRADES is done. Then in the forth section for each Technology Area some related Expected Features have been identified; this represent the core part of the deliverable.

In the future releases of the (Technology) Roadmap for SOCRADES paradigm, some relations with high level future manufacturing objective are expected to be identified. This step is needed to make explicit how R&D in SOCRADES and the related technologies can help aim manufacturing competitiveness and face manufacturing challenges.

1. Introduction

1.1. Roadmap definition

Robert Galvin, former Motorola chairman and advocate of S&T roadmaps, offered this definition: "A 'roadmap' is an extended look at the future of a chosen field of inquiry composed from the collective knowledge and imagination of the brightest drivers of change in that field. Roadmaps communicate visions, attract resources from business and government, stimulate investigations, and monitor progress." [11]

Roadmapping can aid in forecasting the features of future environment and how technological research has to be used to achieve long term objectives. SOCRADES Technology Roadmap will in fact focus on the evolution of the SOCRADES themes beyond the project time horizon. This will help in the project carrying out by keeping the research trajectories as focused as possible on the future competitive environment. Once these technology enhancements are identified they can be developed coherently within SOCRADES objectives.

1.2. Overview of the Roadmapping process and methodology

The SOCRADES Roadmap is made of 3 parts: preliminary information, Roadmap and follow-up activities phase.

1. Preliminary information consists of the roadmap justification and its scope and boundaries. Such information consist a sort of prerequisite to the Roadmap itself.
2. Roadmap building consists of 3 steps (the third step will be carried out in the future releases):

   1) State of the Art of the SOCRADES main technologies definition to define the Current Technological Features that characterize such technologies.

   2) Identification of Expected Technological Features.

   3) Identification of the trajectories to be able to fulfil the gaps.
3. Follow-up activities: This is the moment when the roadmap must be critiqued, validated and hopefully accepted by the group that will be involved in any implementation.

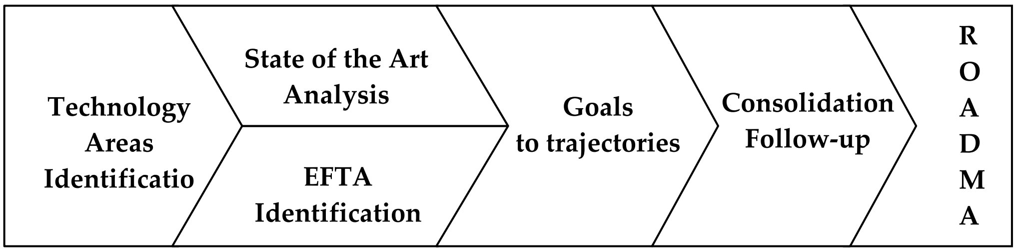

Figure 1: SOCRADES Technology Roadmap Process

The methodology used to carry out all the Road mapping process (especially in the second part of the roadmap) is composed of several main sources of information. In the SOCRADES Technology Roadmap the sources of information considered are:

1. State of the art: analysis of existing Roadmaps and White Papers available on the topics. A comprehensive list of such roadmaps is available in the references [1-11,13].
2. Workshops, carried out with experts coming from both inside and outside the SOCRADES Consortium.

3. Delphi study, still under development: the results will be included in the next releases of the roadmap.

2. Preliminary information

2.1. Justification of the roadmap

Manufacturing plays a vital role in the European economy and society, remaining fundamental to creating stable employment. The development of competitiveness in manufacturing industry is essential for European prosperity; therefore the adoption of manufacturing innovations, such as the SOCRADES paradigm, is needed in order to promote performance improvement of European manufacturing processes. However the present context is characterised by a wide supply of innovative technologies for manufacturing process improvement. Thus, it is essential to put in evidence the sound benefits of adopting the SOCRADES paradigm.

Beyond addressing SOCRADES paradigm by itself, there is a need for addressing more widely the research issues in the specific field. In the SOCRADES context there is a need for example for supporting and improving the overall comprehension of the Embedded Systems future features in manufacturing scenario.

To further support both spreading of this new technology and related R & D efforts, there is a need for road mapping exercise. Benefits of the SOCARDES paradigm and related technologies must be stressed especially linking them with future challenges in manufacturing.

Hence, an auspicated characteristic of the roadmap should be its long term view on future advancement in technology, being able to give to the project not only a vision on how the results will be used, but also the identification of the limit of the developed technology in order to advise further R&D trajectories.

2.2. Targeted Audience

Project Partners

To create credibility, the benefits of this new paradigm and related technologies have to be confirmed through active participation of the industrial partners in this project. Coordination between these actors is extremely important to achieve the task of diffusing a new paradigm (e.g. collaboration between the major player in the ICT and automation value chain is essential to create a successful standardisation body that could satisfy the end-users needs) and to support in designing industrial research trajectories towards the objective defined into the roadmap.

Policy Makers

Showing how SOCRADES tackles the future challenges posed to European manufacturing justifies the policy makers' commitment in the diffusion of this paradigm and in the continuous improving of the research field (i.e. funding further researches, dissemination projects, ad hoc policies etc).

2.3. Definition of the scope and boundaries for the roadmap

The goal of the SOCRADES project is to create new methodologies, technologies and tools for the modelling, design, implementation and operation of networked hardware/software systems embedded in smart physical objects. Starting from this main goal some specific and relevant Technology Areas (TAs) have been identified. The scope of this roadmap is the analysis of the main TAs especially referring to the present state of art and the expected future requirements. By setting out this comparison between present and expected future features, the impact of SOCRADES project should be stressed. In this deliverable first draft the roadmap will focus on four main technology areas of SOCRADES:

1) Ad-hoc networking services platform – Service-oriented Architectures (TUT)

2) Wireless sensor/actuator networking infrastructure (Siemens)

3) Service-centric infrastructure - Enterprise Integration (SAP)
4) System engineering & management (LOU)

It is expected that the final version of the deliverable will consolidate these Technology Areas and provide others, if necessary for comprehending all SOCRADES relevant aspects.

2.4. Partners involved in the Roadmap development

The partners actively involved in the Roadmap development are the ones responsible for the specific technologies on which the Roadmap is focused. However, also contributions from other partners have been useful in stimulating discussions and improving the quality of the roadmap, especially during the workshop, organized for the roadmap specific purpose.

3. State of the Art definition

The state of the art of the SOCRADES main Technology Areas is here mentioned in order to highlight the Current Features that characterize each Technology Area. However, to have more details about each specific state of art of TA, please refer to the respective deliverable.

3.1. Ad-hoc networking services platform - Service Oriented Architectures State of the Art

An analysis of the state of the art of SOA has been conducted and presented in D1.1.

A service-oriented architecture (SOA) is a set of architectural tenets for building autonomous yet interoperable systems. Autonomy and interoperability are somewhat contradictory properties. One of the challenges of SOA is, therefore, to reconcile these opposing principles.

A SOA is typically characterized by the following properties:

Logical view: The service is an abstracted, logical view of actual programs, databases, business processes, etc., defined in terms of what it does, typically carrying out a business-level operation.

Message orientation: The service is formally defined in terms of the messages exchanged between provider agents and requester agents, and not the properties of the agents themselves.

Description orientation: A service is described by machine-processable metadata.

Granularity: Services tend to use a small number of operations with relatively large and complex messages.

Network orientation: Services tend to be oriented toward use over a network, though this is not an absolute requirement.

Platform neutral: Messages are sent in a platform-neutral, standardized format delivered through the interfaces.

Service-oriented architectures rely on a set of standards and specifications (Web Services, WS-* specifications, Frameworks for building service-oriented systems, Orchestration and choreography standards). An analysis of all these specification and standards is presented in D1.1.

3.2. Wireless Sensor/actuator Networking Infrastructure State of the Art

An analysis of the state of the art of WSN has been conducted and presented in D3.1.

In this analysis the most important present communication standards (IEEE 802.11 (WLAN), IEEE 802.15.4, ZigBee, SP100.11a, Bluetooth, WirelessHART, 6LoWPAN, etc.), existing solutions and approaches (Crossbow, MeshScape, SmartMesh, ArchRock, Sun SPOT, WISA, WiBree, The Particle Computer) have been described. Moreover two example hardware platforms (Moteiv Tmotesky and IMTEK SWAM) and the most related European Research Activities (Angel, AWARE, CoBIs, Embedded WiSeNts, RUNES, SENSE, μSWN and WASP) have been considered.

The analysis showed that the following design issues are present in many WSNs, depending on the actual application: **Type of service, Fault tolerance, Limited energy supply, Small computational resources, Scalability, network density, Autonomy, Multi-functionality, Quality of service, Low priced**

Besides the choice of the most appropriate wireless technology to be applied, unresolved issues in this context are especially the following: **Multihop wireless communication, Energy efficient operation, Auto-configuration, Collaboration and in-network processing, Attribute based addressing, Locality, Trade-offs.**

From the state of art analysis, wireless technologies emerge to be already used and studied in industrial environments. Currently the focus is on monitoring and human machine interface applications.

To have more information on this issue please have a look to D1.1 and to D3.1.

3.3. Service-centric Infrastructure - Enterprise Integration State of the Art

Enterprise integration is principally concerned with the integration of business computer systems and workflows between separate organisations in the context of specific business partnerships. With the emergence of SOA / Web Services the field of Enterprise integration has experienced rapid technological evolution. Previously Enterprises were linked via fixed communication channels on rather rigid terms. Service centric infrastructures allow the links between enterprises to be integrated in a finer grained and flexible manner.

In order to achieve this SOA has become a key element of Enterprise Application Integration architectures and the SOA based middleware that links the Enterprises has largely supported the development of eBusiness techniques in the past decade. The middleware that achieved the Enterprise Integration supports characteristics that include negotiation between enterprises, security policy management and enforcement and service brokering in order to present a dynamic integration platform capable of supporting varieties of workflow and business models. To have more information on this issue please have a look to D6.1.

3.4. System engineering & management State of the Art

A number of system modelling and logic design environments are now emerging with the potential to not only automatically generate control logic software but to allow the support of multiple aspects of the complete systems engineering life cycle. Such tools promise to enable higher level design, improved reuse of control logic applications and the possibility to visualise applications in a virtual environment. They also have the potential of supporting distributed control system configurations. To have more information on this issue please have a look to D7.2.

4. Identification of Expected Features of Technology Areas

For each of the 4 Technology Areas (TAs) considered, the Expected Features (EFs) on a time horizon of about 10 years (referring to 2015-2020) are illustrated.

As result of the initial questionnaire on exploitation (refer to deliverable D10.1a "1st release of Exploitation Plan") figure 1 shows the expectations of the partners.

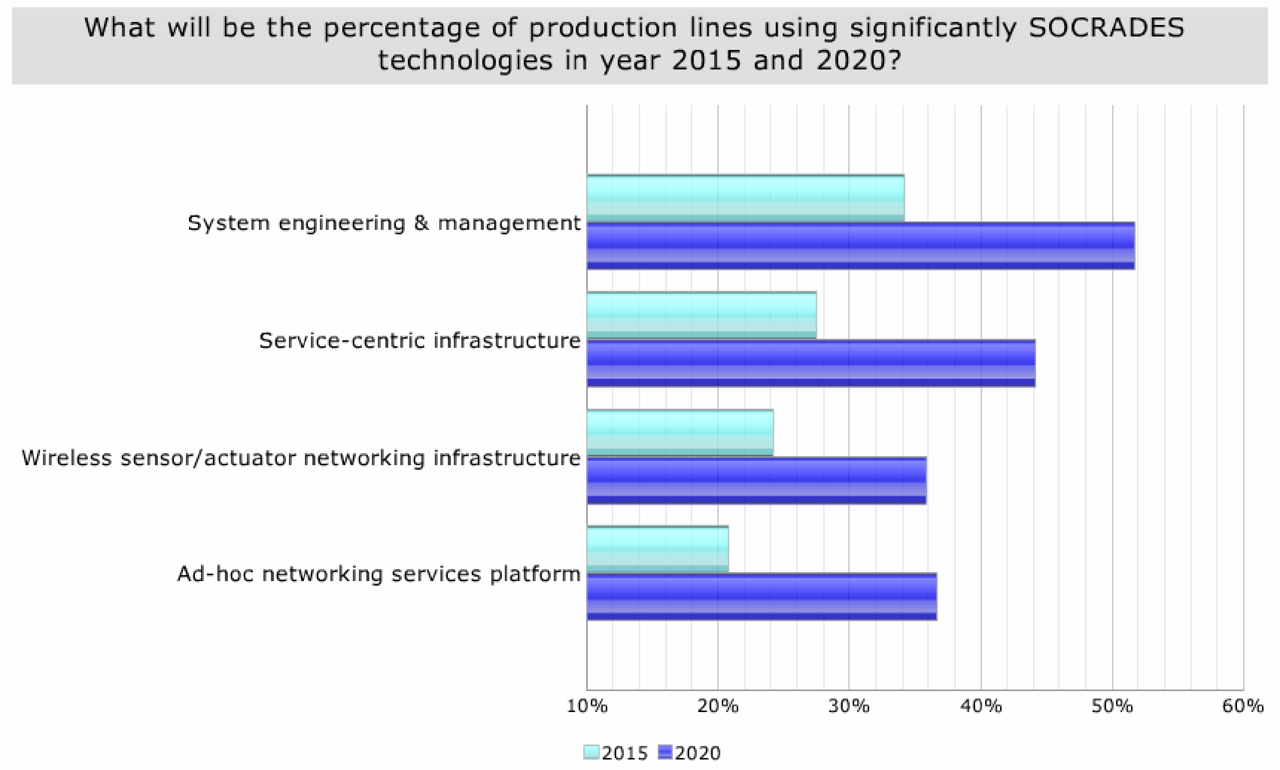

Figure 2: Expectations among long-term usage of SOCRADES technologies

4.1. Ad-hoc networking services platform - Service Oriented Architectures EFs

An important Feature of this TA is related to "**Loose Coupling":** a principal requirement for this infrastructure is that it should be loose-coupled (so without presence of "heavy" links, that means no need for re-programming), in order to accommodate to the highly dynamic environment created in RMS. Consequently this is a key feature in order to create a SOA. Nevertheless since "Loose Coupling" is an implicit and fundamental characteristic of SOAs, it is not considered to be a proper EFTA.

Following, the Expected Features (EFs) for the Ad-hoc networking services platform – SOA Technology Area (TA) are described, with a focus on the current state and on the relevance of each specific EFTA.

1. **Orchestration:** the practice of sequencing and synchronizing the execution of services, which encapsulate business or manufacturing processes, is denominated orchestration. An orchestration engine implements the application logic necessary to orchestrate atomic services, and provides a high-level Web Service interface for the composed process. Even if this is more a technique than a feature it has to be considered a relevant element since it represents a key enabling technique, in order to implement and adopt an efficient and effective SOA.
The nature of orchestration makes it to be self-evolving: i.e. the orchestration can adapt itself to the requirements of the system in terms of number of services to be orchestrated and ways of

orchestrating them. The issues of scalability, dynamicity, etc. could be addressed with the current concept. However in the future optimization of the orchestration process may increase its complexity; this is due to the increasing number of dimensions of systems (so increasing number of orchestrations, dynamicity, scalability, etc.). Finally the increase of features to today orchestration is expected in the next future.

2. **Decision Support System:** since Orchestration by itself can not solve every problem arising during the operation of the system, a DSS is expected to increase, even more than now, its importance in the system. Some issues can be handled locally at the lower levels (through orchestration), but by implementing all functions in same level could neutralise the flexibility of the system, so external DSS implemented with MES, Expert Systems, MAS etc are expected to become necessary to address complexity and reconfigurability. A relevant sub-objective that is worth being mentioned is the following:
   - **Real-Time Scheduling:** as flexible scheduling becomes more prevalent, so the impact on the program structure will increase and the need to develop language abstractions which allow interactions with the scheduler will become a driving force (for example, alternative modules or optional components).
3. **Context-aware services:** this is the capacity of services being aware of devices, of factors such as service location or service state and in general of their environment and reacting to it by adapting their functionality e.g. customizes their offered services. Semantic description of services and ontology for equipment etc. should be the proper tools in order to describe context, however the present state of the art does not fully address how the implementation should be carried out, since the main limits are:
   - No language to describe context
   - No ability to make context composition
4. **Lean data generation and processing:** in order to obtain efficiency through the utilization of SOA, methods and tools are needed for data structuring and handling inside each machine (sensors, electrical signals, etc). This is a feature related to devices wrapped into services; machine developers have to decide how they want to expose their specific data in order to be used by DPWS and, therefore, to be exposed externally as particular service attributes. Moreover during machine composition, management and aggregation of devices' data will need to be addressed carefully. In order to dynamically extract useful knowledge only and master increasing complexity particular attention to this topic should be paid. This will enable the control of more complex and dynamic processes.
5. **Standardization of basic functionalities provided by services:** in the future standardization of basic functionalities should be developed. A further step is the evolution of standardization in different domains and cross-domain. Communication technologies are pretty much standardized (XML, SOAP, OWL-S, etc.) and every time there are new initiatives by interest groups that propose the evolution of these technologies. However this is not the same for standardization of basic functionalities provided by services. A separation between the standardization of the communication technologies and the standardization of basic functionalities provided by services is needed. From the latter group of standardization many improvements and supplements are expected in the future. UDDI repository can be considered a starting point, it represents service functionalities even if not to a great degree.
6. **Common Language:** since a principal requirement for this infrastructure is that it should be loose-coupling, in order to accommodate to the highly dynamic environment created in RMS, a common language is needed. The protocol should be at least XML-based (better DPWS-compliant) and support OWL mark-up in order to enhance it with explicit semantics. The aim is to have a common

basis for seamless operation of standard functions such as discovery, description, addressing, invocation etc.

7. **Choreography:** a complementary concept to service orchestration is that of service choreography. The orchestration level is concerned with the workflow-oriented execution and sequencing of atomic processes, but does not take into account the different types of conversation patterns required to invoke the services associated with those atomic processes. The choreography level considers the rules that define the messages and interaction sequences that must occur in order to execute a given process through a particular service interface. So choreography is related to Peer-2-Peer interaction, between two services according to a law. Examples of choreography include the preconditions set needed for executing a subsequent service: if a device recognizes only certain type of data as an input, then the incoming data must be conditioned before it is provided to such device. The evolution could be about the complexity due to the increasing dimension of systems (so increasing number of choreographies, dynamicity, scalability, etc.). And then the addition of features to today choreography.

8. **Chronology-aware Service Composition:** since time is not yet represented in the description of service before composing services, duration of the composed service could not be known. Methodology to infer composed times (capacity) of execution of two (or more as an expected evolution) services with time properties is expected to become relevant.

9. **Knowledge processing and reasoning:** when ontologies that contradict each other are used, there could be potentially catastrophic consequences for inference results. Suitable technologies for knowledge processing and reasoning for embedded systems need to be further researched. The issue is ontologies reconciliation for agent systems, especially at run-time (since Technology for ontologies reconciliation exists but not for the adoption at run-time). It is very complex to achieve reconciliation of ontologies since a suitable tool may require a considerable amount of computational resources, which are not available in embedded systems. An example of techniques for reconciliating ontologies is “sampling inference” but it does also require good computational resources. Since ontologies for agents and products must be created, a short time solution for avoiding contradictions between ontologies is to try to provide guidelines for their creation (i.e. limit the number of ontologies contradictions). However this doesn’t avoid the potential issue of a need for reconciliation technologies in the future.

10. **Large and complex:** increasing size involves increasing also the number of services and number of interactions. With a grater number of nodes and links the overall complexity of the system increases implying issues in managing the system.

11. **Run-time behaviour of a SOA:** the model of concurrency is the core component of a real-time program. Processes facilitate parallel execution, the placement of timing specification and the handling of faults. Whilst the concurrency model (often called Ravenscar) of high-integrity systems is now well understood and has found representation in subsets of languages like Ada and Java, the model is conservative. There is a need for more expressive subsets. Choreography could help to solve the concurrency problem in a very primitive way by establishing preconditions; however, this cannot guarantee the proper behaviour of system in some situations. It should be taken into consideration that, as mentioned for scheduling problems, some issues can be solved locally (e.g. where goal is to handle concurrency), others at higher levels, where more knowledge of the system is located (e.g. where goal is to optimize performance). Present state of the art is not enough, further solutions for handling concurrency, embodying multiple levels of abstraction, are needed to be researched.
A relevant subtopic of this EFTA is:
    - **Prediction of specifications:** the ability of performing an analysis in order to predict how specification will evolve, during the run time behaviour of the system. This sub-EFTA is also

important since in the future changes in specification will be more unpredictable; complexity and dimension of systems of services may increase the number of changes and so the difficulty in predicting changes. Services can appear/disappear/change at run-time and processes using such services must be able to take notice and to reconfigure the process.

12. **Localization of functionalities:** this feature will be much important in the future. In a distributed system it is fundamental to have tools and methods for efficient service localization in highly distributed environment. Otherwise these functionalities could not be reached whenever required. These methods could be considered more an Expected Feature of the application controlling the system in the factory floor and of higher-level platforms than an EF of the SOA. However this EFTA is essential for a distributed system, as SOCRADES and it has a strong impact on the performances of a SOA-based system (a way to reach this EFTA should be to use Clustering technique for localization).

13. **Intellectual Protection:** Another concern for systems that facilitate interoperability by openly sharing knowledge is that of Intellectual Property (IP) protection and IP interfaces. Through SOCRADES architecture, IP should be guaranteed since only the service is seen, not how the service is furnished (the data measured not how the measure is provided), however improvements are expected.

14. **Dynamically Deployment of Services:** the dynamic deployment of a web service is an important issue to be solved. In the present state of art this can not be done. Following the dynamically deployment of web service should represent a key solution to reach and improve orchestration and choreography.

4.2. Wireless Sensor/actuator Networking Infrastructure EFs

In this paragraph the Expected Features (EFs) for the Wireless Sensor/Actuator Networking Infrastructure Technology Area (TA) are described, with a focus on the current state and on the relevance of each specific EFTA.

1. **Quality of Service:** This is a macro-EFTA which includes a list of relevant EFTAs

   - **Real-time service:** Discrete manufacturing requires data sampling rates in the millisecond range, motion control even in the sub-millisecond range, while process control applications are satisfied with hundreds of milliseconds. These are challenges to current WSANs as, firstly, wireless links by nature are less stable than wired ones and, secondly, such real-time requirements imply a higher degree of energy consumption than non-real-time - and energy is a very limited resource. WSANs currently don’t meet the hard real-time requirements of motion control and discrete manufacturing while those for Process Automation are within reach.

   - **Determinism:** this is related to the reliable delivery of data packets with a guaranteed (mainly maximum) delay within a WSAN. As the wireless medium is an open one and subject to inference by machines or other wireless networks, this requirement is difficult to fulfil especially for hard real-time applications in discrete manufacturing or motion control.

   - **Reliability of sensors:** in particular concerning the integration of support for dependability and real-time, continues to be an important research topic (both from a timing specification and scheduling perspective). Now that the low-level mechanisms for asynchronous transfer of control, budget timers and processing groups are understood, the time is ripe to re-consider the high-level abstractions that can provide a better integration for reliable real-time atomic actions and their introduction into high-integrity applications. High-level dependability mechanisms (including application-level ones) are complementing low-level mechanisms such that the distributed application running on the sensor network as a whole exhibits reliable/dependable behaviour even if parts of the HW/network fail. This can be for instance be achieved by reassignment/relocation of important tasks to other nodes.

- **Efficient communication:** this is important in order to avoid energy wasting. Energy use is linked to frequency of communication and to duration for each communication. So efficient communication can improve energy saving, helping also in reaching energy autarky (see EFTA No. 3 of this section).

2. **De-centralization:** this describes the trend in the shifting of intelligence and processing tasks towards the field level devices; examples are management, control and data processing tasks carried out by a the sensors and actuators instead of a central gateway, controller or network manager. Of course, this is not a dogma for all devices, some might still be communicating directly to the central unit. Generally decentralization can add robustness, flexibility and scalability to the system by reducing the number of centralized critical points. For factory Automation de-centralization at the moment is impossible due to the requirements for real-time and determinism. In Process automation applications with less stronger demands for real-time and (in some cases) determinism, first degrees of de-centralization and meshing are realizable already. One issue that should be solved in order to enable this EFTA is the problem of standards, in particular concerning the communication between different levels. In order to reach de-centralization, the elimination of the network coordinator, even within large networks, will be required in the future. Furthermore another issue concerning gathering (aggregating) information from many nodes has to be solved. Finally, information processing must be developed, especially concerning diagnosis in-network, where the development is still in an early stage and so it prevents from making radical improvements.
3. **Energy autarky (self-sufficient devices):** as a WSAN is thought to be communicating wirelessly, it makes no sense that there are still wires left for power supply. Correspondingly, another way of powering the single devices in a decentralized manner has to be found. This might be (very powerful) batteries, ways of energy harvesting, or a hybrid systems of both approaches. Still a very demanding research topic with only very limited solutions. Some "wireless" way that may enable energy harvesting and energy transmitting should be developed. Another way to reach energy autarky should be improving optimization of energy usage and power management: e.g., activating sensors only when useful.
4. **Self-X Features:** for reasons of easier engineering, easier maintenance, etc. features like self-organization, self-optimization, self-healing, self-configuration, self-stabilization, self-describing etc. are of highest importance. In their maximum extent this would mean a WSAN which does not need any engineering, maintenance, configuration, or human care at all. Even if simpler self-x features already exist, as e.g. self-description, higher levels of self-x, as e.g. self-optimization, are more or less only in their infancy. R&D should be done in order to understand how can a completely autonomic system be controlled/configured from outside, taking into consideration the different needs for degree of autonomicity in different automation scenarios.
5. **Enhanced Service oriented Features:** due to the limited amount of resources, SOA features nowadays can only be implemented on the gateway as this has its own power supply and much higher storage and computing capabilities than a sensor has. The latter in fact nowadays has to be hardly optimized for energy autarkic applications. As soon as the energy challenge is solved or at least in an advanced stage, it will also be possible to apply common IT technologies, e.g. web services, on the sensors and actuators. On the other hand, SOA-related protocols and software for WSAN has to be made more compact and energy efficient. Even if in the present state of art SOA/Web Services can be implemented reasonably only within gateways, first attempts to realize service-oriented middleware for sensor networks are already under way.

6. **Enhanced Efficient Data Processing (Quantity of Service + Quality of Service):** in order to improve efficient data processing and transmission these sub-EFTAs will be relevant
   - **Push-based information transmission:** information is transmitted when ready (push logic), not when required (pull logic) (e.g., a sensor moving between two checkpoints: if between those two checkpoints new information is generated, the information is transmitted, otherwise not). This helps in reducing waste of communication's resources (i.e. requests with related negative answers are avoided).
   - **Fusion/aggregation of data processed within the network:** information have to be propagated within the network by using rules of aggregation/fusion to avoid to overload some nodes of the network and to provide useful enriched information instead of raw data.
   - **Context-aware MAC and Routing:** a system that extracts, interprets and uses context information and adapts its functionalities to the current context will be required in the future. The challenge lies in the complexity of capturing, representing and processing contextual data. More than only gathering context-information, systems must also be able to process information and to infer new knowledge.
7. **Sensor calibration:** the issue of this EFTA is that calibration in WSAN is more frequent than in wired networks. It is important to understand when to do calibration. The problem is not only due to the technical characteristics of WSAN itself, but also to the increase of number of sensors enabled by the WSAN.
8. **Localization:** provisioning of localization information in distributed networks has been, and still is, an interesting research topic. It is important to find from where the data have been generated or taken. Localization can be both a service offered by the network to the applications (object tracking, mapping, etc.) and a technique for improvement of the network itself.
9. **Data storage and search:** collecting data it is crucial to minimize communication cost and delay for the application tasks. To address this, efficient storage and querying of sensor data are both critical and challenging issues in WSAN. This should involve, e.g., distributing to the network some information stored in a sensor to update a new sensor.
10. **Scalability:** number of nodes should vary depending on the need of the application. Existing approaches are still far away from the optimal desired scalability. This EFTA is influenced by self-x and decentralization.
11. **Robustness:** devices in a WSAN should be resistant to the potentially harsh environmental conditions (e.g., such as high and/or low temperature) and should integrate seamlessly in the environment. Robustness must be taken into consideration. Since the application trend requires more sensors and the operational environment gets more hostile, the importance of research on this EFTA increases.
12. **Mobility:** in a WSAN devices can be moving relative to each other. Mobility depends on the application. Mobility will be required and the issue of this EFTA will vary depending on the proprieties (speed, obstacles and radio propagation) of the mobility scenario, due to the application.
13. **Security (cross-layer):** WSAN have to be secured in many ways to ensure their reliable operation. So, security is one of the key points. Besides the security in WSAN, the focus will be in the security in switching from this layer to another, cross-layer security (e.g., from WSAN to SOA).
14. **Interoperability (heterogeneity):** today's WSAN technologies are mainly proprietary and do not provide product and/or vendor interoperability. This incurs higher costs and slows down market adoption. The emergence of WSAN standards (ZipBee, ISA SP100, WirelessHART) should facilitate the deployment of increasingly large industrial wireless networks at both plant and field levels. Since the integration of different smart embedded devices will enable a huge number of application

possibilities, this EFTA is extremely important. A key issue is the development of new models to deal with the complexity involved in such large and scalable systems.

15. **WSAN technologies for closed-loop control:** most applications today target open-loop control or simple monitoring and data collection tasks. If you combine wireless sensor networks with actuators you will get the possibilities of closed loop control between sensors and actuators. In the future it is expected that WSAN technologies should be able to also address closed-loop control needs through higher robustness, prioritization schemes, etc. The central components like PLC will become more and more unnecessary for controlling production processes. The control intelligence will migrate into the WSAN step by step (see also De-centralization).

16. **WSAN deployment tools:** planning and deployment tools for WSAN are today very basic and usually tied to a particular wireless technology. With the advent of interoperable WSAN technologies, new standard installation, deployment and maintenance tools will be available. Also, specific planning tools should be made available to cope with complex installations and provide pre-deployment simulation capabilities.

17. **Attribute based addressing:** data communication will change from data-centric point of view to address-centric. "Where is the temperature higher than 20°C?" NOT: "Which temperature do we have at node xyz?" The WSAN now operates in a data centric way.

4.3. Service-centric infrastructure - Enterprise Integration EFs

In this paragraph the Expected Features (EFs) for the Service-centric infrastructure – Enterprise Integration Technology Area (TA) are described, with a focus on the current state and on the relevance of each specific EFTA.

1. **Flexible Production with Enterprise Support:** We witness that the amount of information "content" in products is increasing, ranging from embedded information technology to simple but critical traceability data frequently included in today's consumer products. Also in the market, there are opportunities inherent in transforming "industrial workers" into "information workers" by empowering them with extended information and knowledge that enhances their ability to perform their jobs. The main characteristic of the future manufacturing plant will be its connectedness to all its vital components: workers, machines, and products. Recent statistics indicate that less than one percent of manufacturing data is automatically integrated into enterprise systems. This will change very soon. To accomplish the automatic integration of manufacturing data, widening the view of manufacturing in both vertical and horizontal dimensions is needed. Cross-Enterprise approaches considering the end-to-end process from the supplier via the plant to the customer need to be developed. Especially the vertical dimension where automation, manufacturing execution, and enterprise systems will be seamlessly integrated into each other is of key importance. Material flow along the value chain and Information flow along the product life cycle are considered significant future features to be supported. This implies also development of concepts and techniques for autonomous production, in which the issuing of production orders is automatically followed by self-preparation of the machines involved, automatic supply chain activation, production and self-qualification of produced parts (integrated Quality Check).

2. **Device to Business Integration (D2B Integration):** Device manufacturers have been dramatically increasing the amount of embedded software in their products, and as a result not only they are able to handle a wide range of computing and communication tasks, but are able to also cooperate and provide their functionality as a service. Therefore device can participate in real-world business applications by providing information coming from their domain and also consuming services available at enterprise level. As an example, devices can now directly trigger an event in the business process and have an effect on its execution. In the future it is expected to have Open Architectures for realizing Device2Business integration.

3. **Cross-layer Adaptive Modelling:** As software development is becoming more complex model-driven engineering technologies offer a promising approach to address the inability of third-generation languages to alleviate the complexity of platforms and express domain concepts effectively. This is becoming more critical in modern enterprise systems, as processes need to be modelled as flexible as possible, while taking into account very dynamic information coming from the shop-floor that affect their runtime behaviour. Development of approaches that effectively handle service modelling and management of intelligent distributed business processes in highly populated web service enabled device infrastructures is needed. Business logic traditionally resided at high-level systems (e.g. ERP) but in the future it will be distributed in several layers (e.g. ERP, middleware, network, device level). Future distributed processes will depend on complex composite distributed services that spawn several companies. As such an effective way of modelling such processes, services, resolving of their dependencies etc needs to be addressed.
4. **Security / Service Policy Compliance:** The future foresees an open infrastructure where rapidly changing business processes and collaboration among companies at several layers is occurring. As a result, business applications are moving from stand-alone systems to enterprise service-oriented architectures (enterprise SOA). The openness and heterogeneity of such systems, though necessary for their operation, is requiring a security approach different from that of traditional systems and architectures. Ensuring security and reliability of the information and application infrastructure is, therefore, vital to the success of these systems. These security architectures and solutions must be tailored to application-specific security requirements and individual risk assessment, comply and adapt to laws and regulations and to be seamlessly integrated with security environments and system dynamics and mobility. Issues that need to be addressed include: regulatory compliance, business-process and workflow security, authorization and trust management, Context-aware security, adaptive security, and mobility, secure services and compositional security, secure deployment and run-time monitoring, security engineering etc
5. **Industrialization of software development:** When it comes to complexity, precision, and quality, the software industry is similar to other sectors such as the mechanical engineering and construction. Much (but not all) of the software built for the automation domain is developed usually from the scratch (or a very limited basis). However as the heterogeneity and applicable domains increase it will be impossible to keep up tackling all needs at high quality without adopting some radical changes in the way we conceptualize, design and implement software. What is needed is to "industrialize" at a fine grained level the process of creating software that will give us the ability to rapidly configure, adapt and assemble independently developed, self describing components to produce families of similar but distinct systems, applicable in different domains that share common core functionality. As an example, the enterprise service-oriented architecture (enterprise SOA) delivered with the business process platform is itself an industrialized software "product." And composite applications are the primary means to either extend existing products built on top of the business process platform or to implement potentially new solutions based on the business process platform via reuse and composition. To further assist such processes coming from top-down, similar concepts must be developed at device level /shop-floor (down-top) and be effectively coupled with the top-down approach. During industrialization, balancing the major business constraints such as time, costs, and quality will get closer to a completely deliberate strategic decision based upon quantitative data gathered from both the market as well as the internal processes (which get data in a cross-layer way ranging from ERP down to device level). The goal: proof of being more valuable than others from an economic perspective.

4.4. System engineering & management EFs

In this paragraph the Expected Features (EFs) for the System engineering & management Technology Area (TA) are described, with a focus on the current state and on the relevance of each specific EFTA.

1. **Efficient/effective (re-)configuration:** ability to configure systems (machines, lines, etc.) built from SOA-enabled modules both statically and dynamically (e.g. add/remove devices) in a standardized manner, e.g., with predictable system performance. Flexible use of systems components coupled with a high-level system configuration capability (e.g. ERPs). For this EFTA partial solutions exist, primarily from research groups, but practical reuse remains poor and system performance is seldom predictable. Whilst individual system components may be of relatively low complexity, effective overall system configuration and change management are of key importance to the end-user.

2. **High level process definition:** the ability to describe the overall behavior of systems composed of many distributed devices in a high level process description language, which directly relates to the specific process the user is concerned with. In some domains user process information (e.g., timing diagrams) is handled by engineering systems however only with limited functionality and the mapping of this information to distributed systems remains problematic and proprietary. This EFTA is important since the mapping of desired process behavior to control systems is very time consuming and error-prone. Besides, distributed systems make this task even more challenging. In the future the standardization of process definition and standardization of definition tools are expected in order to make the process more open and standards based.

3. **Collaborative, integrated, distributed business-driven engineering:** systems capable of being configured and managed in a global business context. Support for globally distributed engineering teams. At the present state the integration of current business and embedded systems is generally very poor, e.g., in the manufacturing domain. In future Web-services offer the promise of better, more open enterprise integration. The added pervasiveness of WS enabled manufacturing devices and the effective coupling of embedded systems to business drivers is a key to greater agility.

   - **SOA-enabled Cooperative Lifecycle Management Design:** tools and methods to improve the design phase in a global scenario (e.g. secured engineering networks between project partners).

   - **Maintenance**: how to maintain the system will be an important topic to be explored, in order to let the system be available as much as possible. Live run monitoring tools and also tools to improve the design phase of maintenance will be also necessary.

4. **Open, lifecycle support:** engineering systems capable of being used at all phases of the machine lifecycle, e.g., initial design, debugging and reconfiguration functionality all provided in a consistent manner and in a vendor independent way. Some current engineering tools cover more than one lifecycle phase but they are generally vendor specific and difficult to use in an unplanned manner. In future, as the lifecycle of systems continually gets shorter the provision of lifecycle support rather than a collection of separate support tools is necessary. SOA will present a standards based and neutral platform on which these lifecycle systems can be both built and linked.

5. **Service-oriented engineering:** engineering support provided primarily through the provision of services rather than on-site engineering activities, e.g., diagnostics, configuration, expert assistance with problems, process optimization with all be remotely provided. Limited remote support is currently available and several research projects are addressing this. Typically it is provided on a single vendor, or single product basis. Security and safety issue currently limit the approach. In a competitive globalised market the effective utilization of remote services will offer a key competitive advantage. SOA will aid in the standards based development of this type of remote support by linking various systems both physically and in conceptual terms.

6. **SOA Marketplace:** SOA could be used to expose services to the systems. SOA enables IP (Intellectual Protection). IP enables marketplace for tools to be developed (maintenance, diagnosis, etc.), creating new business models. This marketplace model will present the users of services and systems in the domain to new pricing models that may be more suitable to companies such as Small to Medium Enterprises.

7. **SOA-enabled Digital Factory:** Planning and modeling. Improving cooperation between the enterprises, and internally- the base for outward cooperation.
   - **Seamless integrated digital engineering:** the ability to mix the engineering of digital, (i.e., 3D dynamic models) and real system components in a seamless manner. Systems will be prototyped in a digital environment and progressively migrated to physical systems under the control of the control of the end-user. Digital models will be utilized at all lifecycle phases. Today there is a still limited use of digital engineering in factory automation although this is becoming an area of heavy investment. Some digital-real systems available in the research domain. Something is carried out in SOCRADES but this feature will be much more developed in the future. Digital engineering is a relevant EF since it is key to the compression of the time taken to engineer, implement and subsequently change distributed automation systems.
   - **Fully digital mock-up of machines:** this feature includes overall digital representation of machines. So including mechanical structure simulation, process simulation, prediction and validation of final pieces' optimal results in design-time. Moreover it involves global dynamic simulation of machines, smart configuration and validation of the suitable fitting between control approach and mechanical structure.
   - **PLM:** tools to design, analyze and manage machine tool products from the stage of initial conception to the retirement stage (Product Lifecycle Management).
8. **Synthetic environments integration:** nowadays development of synthetic environments has reached a high degree of performance. Synthetic environments integrate the last developments in the area of virtual reality which allow representing environments close the reality. Future developments in the synthetic environments integration goes through the practical extension of the synthetic environments in the manufacturing domain.

5. Conclusions

As mentioned, road mapping exercise is fundamental for technological research orientation towards long-term manufacturing objective. This is especially true when technologies integrate as in SOCRADES (e.g. embedded systems, wireless and SOA, etc). An integrated (but detailed) view is needed to catch all the opportunities emerged from the combination of these technologies.

The main result of the first-phase road-mapping exercise carried out within SOCRADES is the identification of the Expected Features for each Technology Area. EFTAs represent the basic building blocks through which it is possible to motivate the importance of R&D in SOCRADES-related topics for future manufacturing aims.

Further improvements will be carried out to completing SOCRADES Technology Roadmap. First step is the analysis of the impact of EFTAs on competitiveness of future manufacturing. Hence, general long-term objectives or challenges for manufacturing should be identified and the links between these aims and the EFTAs should be made explicit. Second step is the allocation of priorities to each EFTA for achieving the objectives identified. Priority allocation is important in order to identify and suggest trajectories towards future competitiveness in manufacturing.

References

[1] ARTEMIS Strategic Research Agenda Working Group. ARTEMIS Strategic Research Agenda. 2006.

[2] ARTIST Advanced Real-Time Systems. Selected topics in Embedded Systems Design: Roadmaps for Research

[3] Bourgine P, Johnson J (editors). Living Roadmap for Complex Systems Science (2006). http://www.once-cs.net.

[4] De Bosschere K. High-Performance Embedded Architecture and Compilation Roadmap. On P. Stenstrom (Ed.): Transactions on HiPEAC Springer-Verlag. 2007. pp. 5–29.

[5] FuTMaN: the Future of Manufacturing in Europe 2015-2020. The Challenge for Sustainability. 2003

[6] ITEA Technology Roadmap for Software-Intensive Systems 2nd edition, (2004)

[7] Mantys Next Generation Machine Tools Technological Roadmap 2005.

[8] Marron PJ, Minder D and Embedded WiSeNts Consortium. Embedded WiSeNts Research Roadmap. (2006).

[9] Manufuture 2020. Assuring the Future of Manufacturing in Europe. Report of the High Level Group 2004.

[10] Richardson CE, et al. Sensor Technology Roadmapping Efforts at iNEMI. IEEE Transactions on Components and Packaging Technologies, Vol. 28, No. 2, June 2005, pp. 372-375.

[11] Galvin R., “Science roadmaps,” *Science*, vol. 280, No. 5365, May 1998, p. 803.

[12] United States Department of Commerce. The NIST 2010 strategic plan. Version 1B. June 2004.

Terms used

Abbreviation	*Explanation*
DPWS	**Devices Profile for Web Services**
EFTA	**Expected Feature of Technology Area**
SOA	**Service-Oriented Architecture**
RMS	**Reconfigurable Manufacturing System**
TA	**Technology Area**
UDDI	**Universal Description Discovery and Integration**
WS	**Web Service**
WSAN	**Wireless Sensor/Actuator Network**

Annex A – Dissemination Levels

PU	Public
PP	Restricted to other programme participants (including the Commission Services)
RE	Restricted to a group specified by the consortium (including the Commission Services)
CO	Confidential, only for members of the consortium (including the Commission Services)

Table 1: Dissemination levels for a document