

# From content-driven to organisational competence management: the case of a software subcontractor

*Veikko Seppänen*  
*Dept. of Information Processing Science,*  
*University of Oulu,*  
*P.O. Box 3000,*  
*FIN-90014 Oulun yliopisto,*  
*FINLAND*  
*E-mail [Veikko.Seppanen@oulu.fi](mailto:Veikko.Seppanen@oulu.fi),*  
*tel. +358-8-553 1986*

*Sari Sallinen*  
*Faculty of Economic and Industrial Management,*  
*University of Oulu,*  
*P.O. Box 4600,*  
*FIN-90014 Oulun yliopisto,*  
*FINLAND*  
*E-mail [Sari.Sallinen@oulu.fi](mailto:Sari.Sallinen@oulu.fi), [jalajout@cc.oulu.fi](mailto:jalajout@cc.oulu.fi)*

*Kimmo Alajoutsijärvi<sup>2</sup>,*  
*University of Oulu,*  
*Finland*

## **Abstract**

The paper analyses the change of a software subcontractor's view to competence management from a content-driven to an organisational approach, in the context of project relationships. A model is proposed to describe the main patterns of the change.

## **1. Introduction**

In this paper we will analyse the evolution of a software supplier organisation's view to competence management during the past five years. The study involves a software department of a larger subcontractor firm, serving mostly electronics companies and employing about seventy software engineering experts at present. As a part of a professional engineering service firm, the department's main deliverables are software development projects carried out for individual customer companies. The projects require expert skills to develop computer software in the context of projects contracted by the customer companies. We are interested in understanding how the supplier views this competence and how the practices used to manage it have evolved. We have analysed the evolution during the past few years, and will suggest a model that makes explicit how the supplier's view to and management of competence has evolved, as well as analyse reasons for the change. The analysis is related to a more comprehensive study of

competence evolution in project-based collaborative networks, the results of which were reported in the 1999 IMP Conference (Seppänen et al. 1999), among others.

## **2. Content-driven competence management**

We consider the management of *technical competence* essential for a software supplier organisation. *Organisational competence* provides support for technical competence, but cannot replace it, since the services that are being offered are based heavily on technical skills needed to build software systems for professional industrial customers. However, the lack of organisational skills can seriously hinder the creation and utilisation of technical competence. From an organisational perspective, technical skills of individual software engineers should be effectively institutionalised. We will discuss this question in more details below, based on the evolution of the focal department's organisational structure. Organisational competence deals also with the management of contracted projects, i.e. inter-organisational relationships. This study of the focal supplier indicates, however, that it is not obvious how to manage portfolios of project-based relationships to create and maintain competence.

### **2.1 Identification of competence**

The original view to competence taken by the focal department revolved around the content of so called embedded software systems. We will therefore call the view *content-driven competence management*. In 1994, when the focal department was established as a part of the engineering service firm, a strategic plan was prepared. Several *technologies* (e.g., "application generator technologies") to be addressed were listed. They actually included also different engineering *techniques* (e.g., "real-word engineering, formal methods"). This view to competence can be regarded as an attempt to determine a uniform and convergent focus of the topics of the software development projects to be carried out by the department.

In 1996, competence screening was conducted by the management team of the department, based on three perspectives to the content of competence: the *problem solving concepts* followed, the *techniques* used to design and implement software systems, and the *applications* involved. As shown in Frame 1, three different competence areas were identified: embedded software platforms, software processes and software development practices. The last was a small area meant for customers, who were not professionals in the development of embedded software systems.

**SCREENING OF EMBEDDED SOFTWARE COMPETENCE***EMBEDDED SOFTWARE PLATFORMS (25 m.y.)*

Modular software architectures and components (10 m.y.)

- concept: embedded systems software platforms
- techniques: graphic-textual models of components
- applications: automation systems, electronic instruments

Data acquisition, monitoring and diagnosis (8 m.y.)

- concept: model-based real-time fault diagnosis
- techniques: adaptable diagnosis platform, electronic documents
- applications: automation and telecom systems

Intelligent, reliable real-time control (7 m.y.)

- concept: fuzzy logic
- techniques: embedded fuzzy control, safety analysis
- applications: automation systems

*EMBEDDED SOFTWARE PROCESSES (15 m.y.)*

Software process improvement (7 m.y.)

- concept: metric-based application-driven process improvement
- techniques: GQM, Primer
- applications: electronic instruments, telecom equipment

Automated integration testing (8 m.y.)

- concept: simulation of target systems
- techniques: MOSIM, TTCN
- applications: telecom systems

*R&D PROCESSES AND ENVIRONMENTS (1 m.y.)*

Improvement of R&amp;D processes

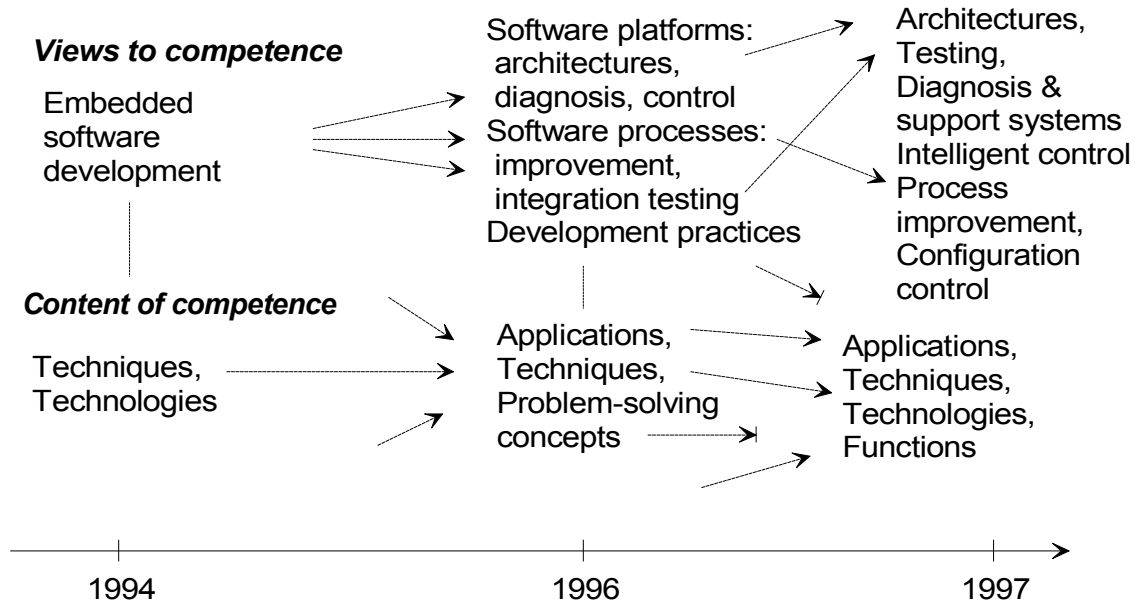
- concept: project-based development of electronic products
- techniques: ISO9000, TQM, Primer
- applications: industry that utilises electronics

*Frame 1. Competence areas of the focal department, as identified in 1996.*

Figure 1 illustrates the change of the view to and the content of the competence of the focal department from 1994 to 1996 and 1997: “embedded software“ has diverged in 1996 to three areas called “software platforms“, “software processes“ and “development practices“. The purely technical aspects of the content of the competence have evolved to applications, techniques and problem-solving concepts. Applications involve automation and telecommunication systems, techniques include certain engineering methods, and problem-solving concepts an approach to organising software development projects, among others.

The competence can be contrasted with its intra- and inter-organisational management, i.e. the internal structure and the external project-based relationships of the focal department. They involved projects on software process improvement and configuration management (Software production group, SPG), design and testing techniques and distributed software (Software architecture group, SAG), and fault diagnosis, diagnosis support systems, and intelligent control (Knowledge engineering group, KEG). The Software production group mastered such software engineering (SE) techniques as quality improvement methods, and worked for a large variety of customers ranging from telecommunication and automation to instrument companies. The

Software architecture group focused on telecommunication and automation applications, where it both developed new technologies and transferred in use software design and testing techniques. The Knowledge engineering group worked much for automation industry. It possessed expertise in knowledge engineering (KE) techniques, as well as in fault diagnosis and intelligent control of industrial applications, but did not focus on any specific software implementation technologies.



***Intra- and inter-organisational management of competence:***

Knowledge engineering group (KEG):

- projects: fault diagnosis and intelligent control of automation systems

Software architecture group (SAG):

- projects: design and testing of automation and telecom software

Software production group (SPG):

- projects: process improvement and configuration management

*Figure 1. Competence change from 1994 to 1996 and 1997.*

**2.2 Evolution of competence management**

In 1997, the concepts of applications, functions, techniques and technologies were started to be systematically used for the content of the competence of the focal department. Table 1 based on the strategic plan of 1997 shows seven views to the competence (called “areas” in the plan), the content of which consists of functions (called “problems”), techniques (called “methods”), technologies and applications (called “domains”). Existing competence elements were denoted by dashes “-”, new elements to be build in one to two years by arrowheads “>” and those that would require three to four years to build by double arrowheads “>>”.

Table 1. Embedded software competence in 1997.

Views to competence	Content of competence			
	Functions	Techniques	Technologies	Applications
1. (SPG) Software process improvement: 6,5 m.y.	- testing and SCM - measurement and assessment - improvement steps	- application of GQM and SPICE - Pr <sup>2</sup> imer - virtual processes	- support technologies for process assessment and improvement	- big electronics firms: 50% - small electronics firms: 25% - end user firms: 25%
2. (SPG) Configuration management: 3,5 m.y.	- storage and assembly of SW components - network-based delivery and maintenance	- SCM-Pr <sup>2</sup> imer - PDM-Pr <sup>2</sup> imer - software agents for delivery and maintenance	- SCM and PDM tools - application-specific SCM models, tools - Internet	- embedded software: 70% - electronic products: 25% - support software: 5%
3. (SAG) Testing: 6 m.y.	- integration testing > , >> testing of dynamically re-configurable software	- simulation, TTCN > visualisation >> 2nd gen. TTCN, self-testing, interfaces	- MOSIM, SDT, ITEX > GTP >> conformance testing technologies	- , > , >> tele-communication protocols
4. (SAG) Architectures (8,5 m.y.)	- flexible configuration of embedded SW >> distribution, environments, self-adjustment	- component-based software development >> real-time C/S solutions, software agents	- product feature models, software components and support tools >> RT-ORB	- , > automation and telecom equipment and systems >> distributed data acquisition and control
5. (KEG) Intelligent Control: 3,5 m.y.	- intelligent control, optimisation > configuration of reasoning systems >> multivariable environments	- fuzzy logic, CBR > neural networks, genetic algorithms >> statistical process control	- modelling toolbox > monitoring toolbox, e.g. CBR tools >> optimising toolbox	- intelligent control in automation applications > electronics manufacturing >> diagnostic applications
6. (KEG) Diagnosis: 7,5 m.y.	- fault diagnosis	- model-based methods	- monitoring through learning > , >> fuzzy logic, CBR	- automation, telecom networks > instruments >> telecom devices
7. (KEG) Diagnosis support: 5,5 m.y.	- , > , >> fault explanation, electronic documentation	- structured documents, explanation mechanisms	- WWW > SGML >> multimedia	- automation > , >> electronics manufacturing

The four types of concepts used for the content of competence had been derived from Abell's (1980) market model. The "techniques" dimension had been added to Abell's model to make a difference between generic scientific or technical methods and specific software implementation technologies. For example, certain computer systems may be designed based on so called object-oriented techniques, but implemented using many different kinds of technologies, e.g. the C, C++ or Java programming languages. It would be short-sighted for a software developer

organisation to mix the two dimensions, because they have different life cycles. Usually generic engineering techniques change much slower than specific software implementation technologies, which are provided by commercial vendors. Interestingly, the definition of the application domains shown in Table 1 is not uniform, but varies from such industrial segments as "automation" to logical clusters of customer needs, for example "distributed data acquisition and control". The same holds for the definition of techniques and technologies. The view to the technical content of competence has thus diverged. Customers' problems are characterised uniformly, as functional needs. However, the unifying "problem solving concepts" visible in 1996 have disappeared.

Identification of the competence areas has evolved towards making a difference between software process and solution related elements, the first three areas deal with the embedded software development process and the other four with embedded software solutions and applications. The competence has changed in terms of the areas as follows: the "Modular software architectures and components", "Intelligent, reliable real-time control", "Software process improvement" and "Automated integration testing" areas of 1996 (Frame 1) are viewed in 1997 as the "Architectures", "Intelligent control", "Software process improvement" and "Testing" areas, correspondingly. The "Data acquisition, monitoring and diagnosis" area (Frame 1) is now viewed as two distinct areas, "Diagnosis" and "Diagnosis support". The "Configuration management" area has been made separate from the process improvement area, and the small supporting area "Improvement of R&D processes" has disappeared.

If Frame 1 and Table 1 are compared, it can be seen that the line organisation has remained the same, i.e. it has not been affected by the changing competence. Considering the further evolution of the competence discussed in the next chapter it is worth pointing out that the organisational responsibilities regarding the competence areas are not distributed uniformly. As is shown in the table, the first two areas are managed by the Software production group (process focus), the third and fourth by the Software architecture group (process and solutions focus) and the last three by the Knowledge engineering group (solutions focus). In other words, one group may be responsible for an entire competence area or share a responsibility with some other groups regarding another area. Because of the unevenly distributed responsibilities, it remains unclear which persons, groups and projects are in practice involved in the creation and management of certain competence.

If the approximate volumes of the competence areas that illustrate roughly their importance in business terms are compared, it can be seen that there are now three rather small areas – less than or about five man-years (m.y.), compared to the 1996 situation, where all other but one area were close to ten man-years. In other words, the competence possessed by the three organisational groups (each consisting of at least fifteen persons), has diverged into several informal teams responsible for certain topics, i.e. clusters of related projects in which these topics are being addressed. Another way to put this would be that inter-organisational management of the competence (project teams) has affected the intra-organisational management of the competence (organisational groups).

### 3. Organisational competence management

In 1998, organisational changes were initiated to better make use of the department's competence. However, there was little intentional strategic evolution from content-driven competence management towards what could be called *organisational competence management*. The reason to tackle intra-organisational matters was that two of the three group managers left the focal department. It was, however, realised that there is an opportunity to change the whole organisation, to better match with the competence areas shown in Table 1.

#### 3.1 Matching of competence with the internal organisational structure

The internal organisation of the department was revised in 1999, by establishing six groups, each of the size of about ten man-years: embedded software engineering (focusing on software process improvement, based on the former SPG), embedded product data management (dealing with configuration management and diagnosis support, i.e. based on the former SPG and KEG), software design and testing (based on the former SAG), software architectures (based also on the former SAG), knowledge engineering (focusing e.g. on fault diagnosis, and based on the former KEG), and intelligent systems (also based on the former KEG). The former KEG was thus divided into three, SPG to two, and SAG to two new groups. Since the line organisation coordinates the usage of organisational resources, the new groups gained better managerial control over the technical competence to build and delivered. This kind of a role of middle management has been emphasised e.g. by Nonaka and Takeuchi (1995). Frame 2 illustrates the new group organisation.

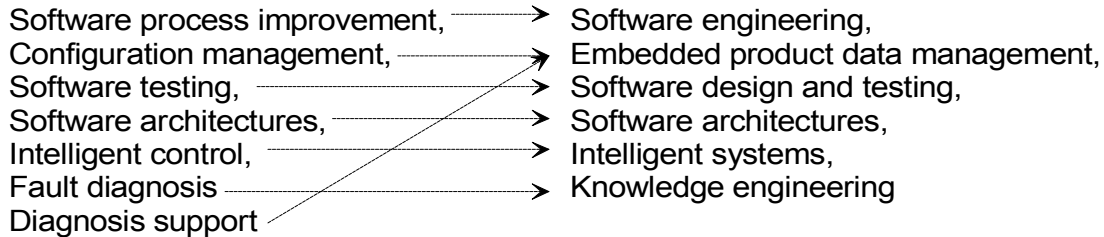
<p style="text-align: center;"><b>Embedded software engineering (12 m.y.)</b></p> <p>Improvement and measurement of industrial software processes; Pri<sup>2</sup>mer: Practical process improvement for embedded software; MetriFlame:GQM-based quality measurement environment.</p>
<p style="text-align: center;"><b>Embedded product data management (10 m.y.)</b></p> <p>Electronic product configuration data management; Management of digital information in electronic products; Electronic documentation technologies and applications; Data bases embedded in electronic products.</p>
<p style="text-align: center;"><b>Software design and testing (12 m.y.)</b></p> <p>Design and testing of embedded communication software;SDL, Uml and TTCN design and testing techniques and tools; NT-MOSIM simulation-based test environment.</p>
<p style="text-align: center;"><b>Software architectures (12 m.y.)</b></p> <p>Embedded middleware solutions, real-time software buses: Distributed and object oriented embedded software architectures; Feature-based and component software development.</p>
<p style="text-align: center;"><b>Knowledge engineering (10 m.y.)</b></p> <p>Model-based fault diagnosis of systems and products; Knowledge acquisition and engineering techniques; Ubiquitous computing technologies.</p>
<p style="text-align: center;"><b>Intelligent systems (10 m.y.)</b></p> <p>Intelligent control applications; Optimisation of industrial systems and processes; Intelligent monitoring; Ubiquitous computing applications; Soft computing techniques: CBR, fuzzy logic, genetic algorithms.</p>

*Frame 2. 1999 embedded software groups - alias competence areas.*

### 3.2 Evolution of competence management

Figure 2 demonstrates the changes that took place, compared to the 1997 situation, in the views to, content of and management of the focal department's competence – based on the new internal organisation of the department that was taken in use.

#### Views to competence



#### Content of competence

Applications: automation, telecom  
Techniques: KE, SE  
Technologies: embedded software  
Functions: product & process

Applications: development, exploitation  
Techniques: KE, SE, data management  
Technologies: embedded software  
Functions: product & process



#### Intra- and inter-organisational management of competence

##### KEG:

- projects: fault diagnosis, support systems and intelligent control of automation and telecom applications

##### Knowledge engineering group:

- projects: fault diagnosis, data mining of telecom systems and instruments

##### SAG:

- projects: design and testing of automation and telecom software

##### Intelligent systems group:

- projects: intelligent control, optimisation of automation systems

##### Software architecture group:

- projects: design of telecom and automation software

##### Software design and testing group:

- projects: design and testing methods and tools for telecom software

##### SPG:

- projects: process improvement, configuration management

##### Software engineering group:

- projects: process improvement, software quality metrics

##### Product data management group:

- projects: electronic documentation, product data management, configuration management

Figure 2. Competence change from 1997 to 1999.

One of the most interesting changes in the content of the competence visible in Figure 2 involves the “applications“ dimension. It is now considered to consist of the “development“ and “exploitation“ of embedded software systems, i.e. the customers are seen either as active embedded software developers or end-users. The Intelligent systems group is focusing much on the latter, which are typically industrial production sites. The software design and testing group, on the other hand, deals with customers that employ world-class industrial experts in the group’s competence area. The new Product data management group has no focus in this sense yet, its projects are being carried out both for active developers and end-users. Also the Software architecture and Software engineering groups have both developer and exploiter type of customers, but work mostly with the former.

In other words, considerable convergence has taken place in the internal competence management responsibilities, but the department’s new view to the customer segments based on two types of applications has not yet resulted in any changes in project planning and management. This could have required, for example, that those groups dealing with exploiter type of customers would have different project marketing and management practices than those dealing with active embedded software developers.

### **3. Analysis of the developments**

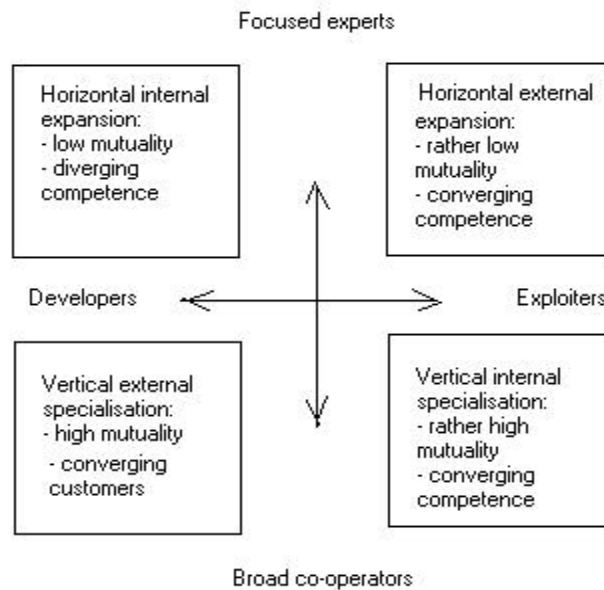
There are several interesting developments in the focal department’s view to and management of its competence. The changes illustrated above involve the evolving content of competence on one hand, and the changing internal organisation of the focal department, on the other hand. It is striking that inter-organisational relationships were behind important changes of the content of competence. It was evolving much due to successive projects dealing with certain topics. For example, projects related to intelligent systems development resulted in the identification and management of the competence area as a formal organisational group. Configuration management related projects did not evolve to the same extent, not to speak of the competence area “R&D processes and environments“ shown Frame 1. The internal organisation of the department changed only after a major – actually a devastating – change took place. The apparent inconsistency between the emerging content of competence and its organisational responsibilities did not result in any change from 1994 to 1998, although the department dealt with a rapidly changing high-technology service business.

In order to make the major patterns of change explicit, we will use the two key concepts proposed by van de Ven et al. (1999) as one result of their famous Minnesota Innovation Studies – *divergence* and *convergence*. They are applied here to the views to, content of, and management of competence, so that the model assumes simultaneous divergence and convergence of competence and its management.

The developments illustrated above indicate that the *view* to the competence diverged at first considerably due to the emerging project portfolio, but converged then to remain much the same until 1999, when embedded product data management was put forward as an entirely new competence area. The *content of competence* has evolved little by little, without much divergence. The *intra-organisational management* of competence diverged gradually inside the former three groups, but converged only after a radical organisational change had taken place. One of the main changes in the *inter-organisational management* of competence, i.e. the project

portfolio, is related to the convergence of the application dimension of the content of competence, resulting in a view that there are only two types of customers to serve.

Figure 3 summarises the developments in the form of a competence change model, which contrasts two types of focal actors to each other, inter-organisational (developers vs. exploiters), and intra-organisational (*focused experts* vs. *broad co-operators*). Focused experts are individuals or organisational units interested in special competence, e.g. in certain software testing techniques. For example TTCN shown in Frame 2 is one such technique. Broad co-operators deal with wider topics, cf. “Electronic documentation technologies and applications“ in Frame 2. The views to and content of competence are affected by the changing *mutuality* of the actors involved in intra-organisational activities and inter-organisational relationships. Mutuality is one of the well-known attributes of industrial relationships proposed e.g. in (Ford et al. 1986). It can be understood as a measure of the degree to which an actor is prepared to collaborate with other actors internally or externally.



*Figure 3. Model for competence change.*

Mutuality is often very low, when the supplier’s focused experts are associated with developer type customers. The supplier is selling diverging competence (“*Horizontal internal expansion*“) to customers, who are or will later be capable of developing and making use of the competence by themselves. From the intra-organisational viewpoint this means that the diverging competence may gradually or through some conflicts result in internal re-organisation. The focal department’s new structure established in 1999 is an excellent example of the process. The intended evolution (foundation of one new group, Embedded product data management) needed to be reconsidered after a radical change. The divergence of competence resulted in six groups, most of which can however be traced back to several years, when divergence was gradually proceeding.

Based on this case, mutuality between collaborating actors may be seen as high, although it has decreased due to internal horizontal expansion. The obvious reason for the expansion in the case organisation is the large number of applications, functions, techniques and technologies involved in software engineering subcontracting services offered to active developers of embedded software systems. Although this is the most common situation in many professional engineering service firms, it may include one of the biggest pitfalls for the supplier, from the viewpoint of competence management.

In the case of exploiter type focused buyers (*“Horizontal external expansion”*) the supplier sells similar competence to many different customers. They may, for example, need external software engineering experts to carry out some investments. The emergence of the Intelligent systems group illustrates this kind of a process. The so called fuzzy systems (Frame 1) in which industrial manufacturers were interested, launched the process some time in the early nineties. At first there was only one person in the focal department knowledgeable of this technique, in less than ten years a group of several experts. Compared to the horizontal internal expansion, external expansion requires that the view to and content of competence do not diverge too much, and that there are customers willing to make use of the gradually emerging competence. As an example, the Intelligent systems group now possesses skills on other techniques than only fuzzy logic (cf. Frame 2), but they are all related to the implementation of functions used in *“intelligent”* industrial software applications. The R&D process improvement services provided by the focal department to customers in 1996, who were not experienced product developers (cf. Figure 1), did not meet these criteria.

Long-term collaborative relationships are possible between broad co-operators and exploiter type of customers, provided that the supplier can deliver converging competence (*“Vertical internal specialisation”*) that becomes an integrated part of the customer’s own competence. This requires rather high mutuality between the parties and among the supplier’s internal organisational structures. The fault diagnosis related projects carried out by the Knowledge engineering group during the mid-nineties represent this situation. A majority of the group’s activities was devoted to fault diagnosis, and the group manager was an expert in this competence area. However, the later developments indicate that the competence was, after all, diverging too much. Broad collaboration required rather extensive work on other related topics (cf. the 5,5 man-years spent on *“diagnosis support”* in 1997, according to Table 1).

Moreover, some of the former key exploiter type customers evolved towards active developers of fault diagnosis systems, which changed the situation to horizontal internal expansion. As an example, the original automation related fault diagnosis systems are not mentioned in Figure 2, but new applications developed for telecommunication devices and measurement instruments. The arrows included in Figure 3 are meant to illustrate the fact that the four cells are not clear cut, and that the situation may indeed change considerably over time. For example, not only an exploiter type customer may change to a developer, but a reversed change may also occur. The former is, however, more typical in the case of the focal organisation. As indicated by the fault diagnosis competence, it is related to formerly new techniques and technologies becoming routinely used in the implementation of some functionality of certain applications.

With developer type customers, long-term relationships would require strategic alliances (“*Vertical external specialisation*“), where competence can be diverging, but the relationship is converging. This is an extensively studied process among the researchers of industrial relationships, such as the members of the IMP group. From the viewpoint of competence management the situation is not without any problems, though. It would require the supplier not only to acquire application-specific skills, but also to establish internal organisational structures to serve the key collaborators. In the focal organisation this has not happened. No particular customer applications are mentioned in Frame 2, and the new view to customers is a step away from the former application-oriented customer segmentation principles. Instead, evolution towards the horizontal internal expansion seems to be continuing.

#### **4. References**

Abell, D.A. (1980). *Defining the business*. Englewood Cliffs, NJ: Prentice-Hall.

Ford, D., Håkansson, H. & Johansson, J. (1986). “How do companies interact?“ *Industrial Marketing and Purchasing*, vol. 1, no. 1, pp. 26 - 41.

Nonaka, I. & Takeuchi, H. (1995). *The knowledge-creating company: How Japanese companies create the dynamics of innovation*. New York: Oxford University Press.

Seppänen, V., Alajoutsijärvi, K. & Eriksson, P. (1999) “Projects or products – seeking for the business logic of contract R&D“. Proc. of the 15<sup>th</sup> Annual IMP Conference, Dublin, Ireland, September 2-4, 1999. 8 p.

Van de Ven, A.H., Polley, D.E., Aarud, R. & Venkataram, S. (1999). *The innovation journey*. New York: Oxford University Press.