# The Politics of Information and Knowledge Sharing for Systems Management


## Geert de Haan

*Genovevalaan 180*
*5625 AM Eindhoven*
*The Netherlands*
gdehaan@worldonline.nl
http://home.worldonline.nl/~gdehaan/



### ABSTRACT
This paper reports on a case-study of introducing intranet facilities to improve co-operation by way of information support of a department that provides system management services.



### Keywords
System management, Internet, Intranet, Information System, Requirements analysis, Management.


### INTRODUCTION
This paper reports on a case-study of introducing intranet facilities to improve co-operation by way of information and knowledge management within the Distributed Systems (DS) department of a large international company that provides full IT services.

The main product of Distributed Systems is system management for distributed computer systems services built around VMS, UNIX and NT server platforms. In addition, DS provides expertise for special projects and it is responsible for the development of new services in its working area.

DS is part of the Technical Infrastructure Services (TIS) department which system management that may be characterised as: full services, around-the-globe, around-the-clock and high availability (well below .1% unplanned downtime). In practice, the main customers are large, internationally operating companies such as ICI and Philips who cannot do without and who are able to afford such services.

System management is a knowledge-based activity; to do this efficiently, you need a standing organisation of people to provide the knowledge where it is needed, procedures and standards to manage and possibly improve the process, and an extensive set of information sources to save the people from having to know everything.

Extensive growth caused problems in managing the organisation and the information, so a project "Design of an Information Infrastructure for Service Delivery" (DI2SD) was started intended to provide better information support.

### Business Processes
Regarding procedures, system management is set up according to a methodological framework called ITIL (IT Infrastructure Library), originally a set of reference books on IT management written under the auspices of the British government for its Central Computers and Telecommunications Agency (CCTA, 1989).

ITIL or SMART as the local adaptation is called divides systems management into a number of services and processes such as Customer support (the helpdesk), Database management, Contingency services (disaster management), and Problem- and Change Management, etc. that describe what is delivered and how it is delivered.

SMART forms the conceptual layer in an ISO-9000 process quality system (ISO, 1987). Because ISO-9000 is limited to description only, elements from methods like TQM (Total Quality Management; Roa et al., 1996) and CMM (Capability Maturity Model; Paulk et al, 1993) are used in the quality system to measure and improve the processes.

The conceptual layers are further effectuated by a set of process descriptions and work instructions which relate the processes to the role assignments and responsibilities of the organisation, and a set of standards on e.g. operating system and database configuration that enable systems be managed in a uniform way.

### Standing Organisation
Within the company there are four main business processes: Sales Acquisition, Portfolio Management, Contract Implementation and Service Delivery. DS is responsible for the Service Delivery in its working area, and actually provide the system management services. It is also involved in parts of Contract Implementation to prepare computer systems, organisationally and technically, for system management.

To provide services globally, DS is geographically distributed in one global, a few regional and several local sites. The local sites are responsible to the day-to-day activities, the regional sites add regular

systems management 'following the sun', and the global site adds support by specialists.

The organisation of the global site is split horizontally and vertically. Horizontally there are platform specific groups (UNIX, VMS, etc.) and vertically there are layer specific groups with respect to both support level (helpdesk, system management, specialists) groups and support type (SAP using an Oracle database on a UNIX platform).
A main problem for the organisation of DS are the growth figures, for some groups well over 100%. These show up in a constant struggle for resources between actual system management work, and the work required to address the growth by means of e.g. educating trainees, improved tooling and more usable information support.

### Information Sources

Information support consists of a large number of wordprocessor-, paper and intranet documents, local office- and Lotus Notes databases, a globally accessible mainframe tool/database 'INFOMAN'[1]. In addition, the systems themselves can be seen as information sources.

Ideally, a paper and wordprocessor document describes the configuration of a particular system, and all the problems, changes and contract data should be managed in Infoman.

The **documentation** became troublesome when different groups started to use their own, to address their specific responsibilities, levels of skill and their own favourite places to store (and hide) them. Due to the workload and the need to update the same information at different places, maintenance suffered and inconsistencies between information sources were born.

Although **Infoman** is a good tool for managing mainframes where it provides for both, monitoring, problem and change administration and process automation. Within DS Infoman is only used as an administration tool with two main disadvantages:

- Infoman is not very suitable for the dynamic nature of distributed systems. Plugging in an extra diskdrive overnight is *not done* in mainframeland but common for DS, and whereas mainframes are generally dedicated to only a few processes, DS systems are used for general purpose.
- Infoman has a user interface from the stone age. Infoman is what Thoresen (1997) calls "Simple but Cumbersome"; it has a steep learning curve and it is not appropriate for 'occasional' usage, which is further deterred by a lengthy login procedure.

Because of the problems with Infoman several groups within DS developed their own solutions by creating local databases, document standards, etc. The general result is that information is not always accurate, available, easy to locate, and easily accessible. This problem is made more serious because system management often requires close co-operation between globally dispersed groups from DS and from other TIS departments.

## THE DI2SD PROJECT

At this point the project entitled "The Design of an Information Infrastructure for Service Delivery" started with the aim to improve the efficiency and quality of the service delivery processes by using the intranet to actively, and more-and-less interactively exchange information between the parties involved in this process. Note that the software to control and monitor computer systems was not part of the project; only the information support system management was.

The project was formulated in two steps, first a requirements analysis part, after which a decision would be made about a subsequent design part.

## REQUIREMENTS ANALYSIS

The aim or 'deliverable' of the requirements analysis should be a set of requirements for the structure and contents of an information system as follows:

- procedures, methods and tools to store, update and remove of information.
- an information structure to facilitates task-appropriate navigation.

Also, four technical requirements should be met:

- allow for global access to locally stored (distributed) information.
- facilities for user authorisation, access control and ownership
- document management facilities to prevent inconsistency and manual maintenance.
- facilities to search, retrieve and access information.

The approach followed in requirements analysis was threefold. First, following the idea that information should support formal procedures governing the service delivery process, an extensive analysis was made of the process quality documentation.
Secondly, to become familiar with the 'way of working' beyond the formal procedures -which indeed proved much more important- a series of interviews was held. From each group the manager and one of the system managers was interviewed to determine which information was used from where, how, and for which purposes.

---

[1] INFOMAN is the local name for IBM's "Information Management" tool that is now part of the TIVOLI framework of system management tools. See:http://booksrv2.raleigh.ibm.com/infoman/

Finally, time was allocated for less structured observations during so-called "Severity Ones", crisis situations that clearly show acute information needs.

## Some Observations

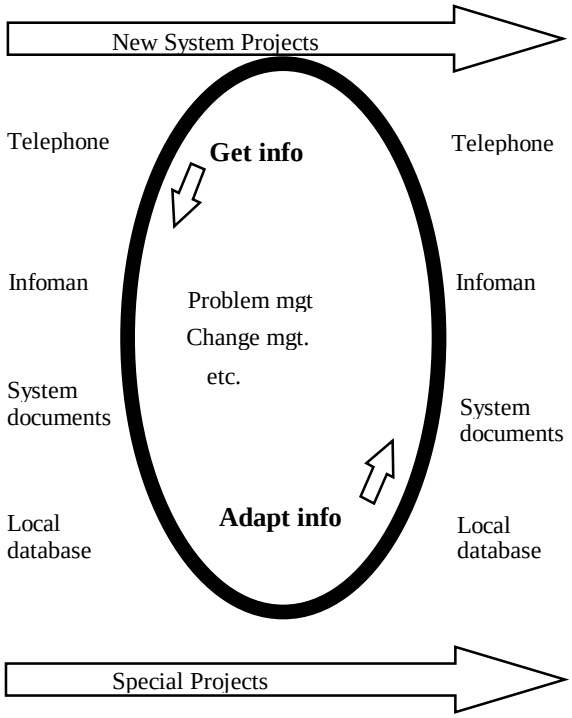
A number of general observations were made during the requirements analysis part of the project. These have been described elsewhere (de Haan, 1998) and will only be briefly described here:

- procedure-based business processes makes task analysis difficult: because role-assignments are dynamic and seldom explicitly laid down, process descriptions are not as helpful as might be expected to understanding the organisation's work structure. For example, whereas the documentation describes service delivery in terms of separate processes, in practice, each group has only one work queue.
- work instructions and task analyses are clear and simple but say little: work instructions for e.g. Problem Management describe the administrative process but not any of the actual steps taken to solve the problem, and problem solving activities only seems to have a generic observe-act-test cycle as a commonality.
- techies don't document: system managers like to invent but not document their inventions, and they are even less likely to maintain the documentation, or adapt it to the needs of less experienced people.
- information analysis requires that information management is in-place: apart from the problem that some documents do not exist when they should, it took considerable work to determine what information is available, and how to get it.
- wheels are constantly reinvented and no one seems to keep track of them. This is a common problem of every sizeable organisation. This often became clear since DI2SD was one of only a few department wide projects.
- the tacit knowledge problem (Evans, 1988): much information is kept hidden in people's heads because knowledge is difficult to verbalise and because people, and particularly the gurus, tend to take it for granted that everyone has to the same background knowledge.
- system management is a highly creative task: a large part of system management is rather dull but still many problems and changes require creative problem solving.
- for managers it may pay to keep information to themselves: in presenting a thesis about knowledge sharing at Origin, Karssen (1997) noted that "some managers (especially within TIS) tend to sit on their knowledge". This happens for genuine reasons but also for 'political' purposes. A financial department, for example, would only grant DS perusal of contract data for security reasons, but also to keep contract management within their responsibility.
- the need for management backing is a common observation for project success (Markus, 1983). Although systems for project ownership and resource allocation are used, these proved insufficient to ensure sufficient co-operation. DI2SD became "Trapped in Obsolescence" (Martin, 1991) when a strong DS manager left and was replaced by a rather unsupportive acting manager.

## Some Results

The results of the analysis phase consisted of descriptions of the general information flow as regards the service delivery process and descriptions of the main information problems, both in general and in relation to the software used. A description of the information flow in full detail and a complete design specification could not be created because of large differences in the work process between groups, and because of lacking information.

**Figure 1: System management processes**




DS has three different processes; see figure 1. During a cyclic main process problems are solved, changes effectuated, etc. and a variety of information sources is consulted and subsequently updated. New systems are introduced in a linear process of planning, gathering information, documenting, installing software and handing everything over to the next group until the system is ready. A third process deals with all the projects not related to system management.

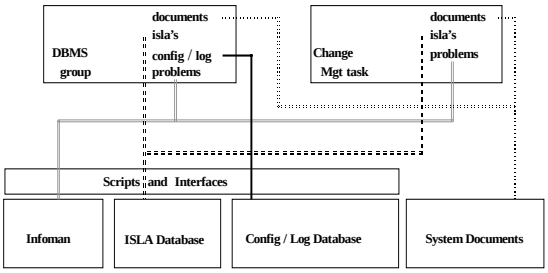
Requirements analysis did not go well. Although backed by the general manager of DS, not belonging to a group put me in an isolated position with insufficient means to get things done from the group

managers. It was much easier to gather information, especially in unscheduled sessions, from the system managers who were less restricted in talking about 'all the things that go wrong'.
Although the business procedures say little about how the work is actually done, they proved helpful as a framework. Informal information gathering during severity one's was useful, also because the subsequent 'root cause analyses' provided the opportunity to verify the observations.

## DESIGN

During the design phase, the top-down approach of the analysis phase was replaced by a bottom-up approach, driven by creating artefacts and demo's to actually solve the problems on the workfloor.
To make clear what the project was heading at, the results of the analysis phase were laid down in an information policy proposal and in a document explaining the possibilities of internet techniques to overcome any technical misunderstandings. In essence, the proposal suggested to use web-browsers to provide a uniform, group- or task specific interfaces to the different information sources, hiding information formats, storage places, and the particulars of databases and (legacy) systems such as Infoman. See figure 2.

**Figure 2: Uniform information access**




I will first discuss the main reasons to pursue a intranet-based approach, and continue with a description of the subprojects and their results.

### Intranet Technology

There was ample reason for DS to make more advanced use of internet technology:

- web browsers exist for all the platforms used and managed by DS. Using plugins, MIME encoding and translation tools the file format problems and most of the storage-place problems can be resolved.
- the web only allows for lightweight document management (Rein, et al., 1997) that would, however, suffice until the arrival of the much awaited real solution.
- with a combination of HTML and client-site scripting it is easy may to create uniform user interfaces (Flanagan, 1996).
- server-site scripting may be used to access local databases (Fraternali, 1998) and data in (legacy) applications (Barta and Hauswirth, 1995).
- internet technology has gained the status of proven technology or standard business practice. As such, it would fit the purchasing policies of DS. In addition, it would "seamlessly integrate" with the office environment provided by a main business partner.

### Organisational Issues

More important than the technical issues are the organisational advantages:

- an intranet, though mainly used as an elementary technical resource for e.g. Telnet, ftp and a variety of email and messaging systems, is readily available. New system monitoring software (CA TNG) had already replaced proprietary networking and hand-crafted scripts by a standard environment. As such, the logical next step would be to move information support and system activities themselves to the intranet (e.g. Pingleton and Fischer, 1994).
- the logically and physically distributed organisation of DS with many different parties involved in managing particular systems had already made it clear that the possibilities for co-operation by email, telephone and ftp were insufficient and counterproductive.
- DS is not meant to nor able to allocate resources to extensive development projects but it is able to provide the resources for the more incremental and piecemeal approach of internet development. The nature of internet development as low-key and anarchistic would also fit the user group and promote success.
- finally, the company's top management had recognised the need to use the intranet as a business advantage and supported this strongly. Apart from a slogan and a policy (see e.g. Telleen, 1996), staff and other resources had been allocated to provide expertise, tools, etc.

### Projects and Results

Three and a half projects were started to put the design phase and the proposal into effect:

- a webteam projects with members from each group to develop the department's web site and create involvement for interactive use of the intranet,
- a project management project to create a demo to help solve the problem that at the department level no one exactly knew what was going on and who was involved in different projects,
- a document management project to provide uniform and controlled access to documents,
- a small project to demo and provide connectivity with various local databases and Infoman.

Generally speaking, these projects worked well. The webteam suffered slightly because several groups were represented by their most junior employee but this was -fortunately- counterbalanced by a senior manager who also represented the team in the management meeting. The team was a support for

the web master in getting each group to produce the obligatory documents, and it worked well as a discussion platform and as a test forum.

The project management demo, a mere set of scripts running on a server, was a big success, also because it accidentally turned out useful as a tool to find expertise. The success of the projectbase and of a DS good-news page is best illustrated by the questions they raised to restrict access "*now that the whole company knows what we are doing*".

The document project was less successful because several group managers were reluctant to make the system documents accessible to all DS employees. A more advanced solution died during negotiations with the intranet management, who disallowed customer scripts. Earlier then expected, however, the document management software became available, and because DS was already working on it, the department was chosen for a pilot implementation.

The connectivity project had mixed success with respect to Infoman; the software was readily available but it required networking facilities far beyond the budget. As a workaround a script was used to dump essential data into a local database for read-only access. The project did deliver several working prototypes to access this and other local databases but before they could make it into the production environment, the DI2SD project was killed.

## CONCLUSIONS

It may be risky to draw conclusions from just one project in one department in one company, nevertheless, the following five conclusions seem justified:

- Business procedures say little about how which information should be available.
- The quality documentation gives a very precise description of the business processes that looks like a perfect task analysis model but was often rather different to find again in reality because of the dynamic and implicit role assignments and because work instructions only lay down the administrative side of working.
- Even the most primitive information sharing facilities can make a big difference in improving information support and co-operation. Improving overall information access would require big decisions such as replacing Infoman, but this is not within the possibilities of a small department. The majority of problems existed because most groups used different formats, systems, platforms and, often inaccessible, places to store their information. Solving these problems, at least in a technical sense is trivial and can be done with standard off-the-shelf technology.

The main problem is in the organisation and the culture and not in the information. A good deal of the information used in system management is sensitive: information about contracts, employees, and tools is better kept hidden from competitors, even though this may hamper business itself. It is, for example, not possible to search the personnel database on special expertise (Karssen, 1997).

Information support, at least in this case suffered most from organisational factors, such as reward schedules that favour allocating resources to customers, extensive growth and the scarcity of resources, and the tradition of co-operation based on trading. As a result, group managers are stimulated *not* to share resources because that would simply undermine the trading position of their group.

Group managers are neither stimulated to use information sharing facilities, even if these would save resources, because that would leave them with less if anything to trade with.

The most advanced document and information management systems won't solve the problems.
Sponsored by upper management, there are projects underway for document management and intranet development that may fail like a Lotus Notes project did before. In an organisation with powerful middle management, it seems that successful information sharing heavily depend on the proper incentives for this group, regardless of the importance of backing from the upper management (Markus, 1983) and perceived benefits among end-users (Grudin, 1988).

Information support should be approached bottom-up. The requirements analysis part of DI2SD was essentially a one-man show that began to look silly when management backing changed from support to indifference.

In this situation it was best to try to create a critical mass for change in a bottom-up fashion via rapid prototyping (Wilson and Rosenberg, 1988). Solving some of the most acute problems, if only by a demo, helped to create support among the system managers and to convince some group managers of the utility of the project.

Finally, as a more personal conclusion regarding my first year in industry, I might support a conclusion from Landauer (1991) that cognitive psychology should not stick too much to significance levels and focus on ways to make things better in everyday life.

## ACKNOWLEDGEMENTS

I wish to thank Eva Vermeulen for her support and mints, and Hans Kappert for his -albeit unofficial- role as advisor and teacher.

## REFERENCES


Barta, R.A. and Hauswirth, M. (1995) Interface-parasite gateways. In: Proceedings of the 4th International WWW conference. O'Reilly, 1995. Also available from:
http://www.w3j.com/1/barta.273/paper/273.html.

CCTA (1989). Central Computers and Telecommunications Agency. *Information Technology Infrastructure Library*, 1989-1998. Available from: Examination Institute for Informatics (EXIN), www.exin.nl/itil/.

Evans, J.St.B.T. (1988). The knowledge elicitation problem: A Psychological Perspective. *Behaviour and Information Technology 7*, 2, 111-130.

Flanagan, D. (1996). User interfaces with HTML and Javascript. *World Wide Web Journal 2*, 1, 1996. Also available from:
http://www.w3j.com/5/s3.flanagan.html.

Fraternali, P. (1998). Web application development: tools and approaches. Proceedings of the Seventh International World Wide Web Conference, Brisbana, Australia. Elsevier. Also available from:
http://www.elet.polimi.it/~fraterna/www7/webtools.html.

Grudin, J. (1988). Why CSCW applications fail: problems in the design and evaluation of organisational interfaces. In: *Proceedings CSCW '88, Computer-Supported Cooperative Work*. ACM, New York, 85-93.

De Haan (1998). The design of an information infrastructure to support system managers and business procedures -or- How to catch a guru with quality. *Proc. Second Multidisciplinary Workshop on Cognitive Modelling and User Interface Development*. Freiburg, Germany, 1997.

ISO Standards 9000-9004 (1987). Quality Management and quality assurance standards - guidelines (9000), Quality Systems - models (9001-9003), Quality management and quality system elements - guidelines (9004).

Karssen (1997). Http://Knowledge_Sharing@Origin-IT.com. Ms. thesis. Erasmus Univ., Rotterdam.

Landauer, T.K. (1991). Let's get real: a position paper on the role of cognitive psychology in the design of humanly useful and usable systems. In: Carroll, J.M. (ed.) *Designing Interaction: Psychology at the Human-Computer Interface*. Cambridge University Press, Cambridge, UK.

Markus, M.L. (1983). Power, politics, and MIS implementation. *Communications of the ACM 26*, 6, 430-444.

Martin, J. (1991). *Rapid Application Development*. MacMillan Publishing Company, New York.

Paulk, M.C., Weber, C.V., Garcia, S.M., Chrissis, M. and Bush, M. (1993). Key practices of the capability maturity model, Version 1.1. *Report CMU/SEI-93-TR-25 [ESC-TR-93-178]* Software Engineering Institute, Carnegie Mellon University, February, 1993.

Pingleton, M. and Fischer, T. (1994) Utilizing mosaic and the WWW in an operations environment. *Proceedings of the Second International World Wide Web Conference '94: Mosaic and the Web*. Chicago, Illinois, USA. Available from:
http://www.ncsa.uiuc.edu/SDG/IT94/Proceedings/CorInfSys/fischer/fischer.html.

Rein, G.L., McCue, L. and Slein, J.A. (1997). A case for document management functions on the web. *Communication of the ACM 40*, 9.

Roa, A., Carr, L.P., Dambolena, I., Kopp, R.J., Martin, J., Rafii, F. and Schlesinger, P.F. (1996). *Total Quality Management: A Cross Functional Perspective*. Wiley, New York.

Telleen, S.L. (1996). *Intranet Organisation: Strategies for managing change*. Amdahl, iorg.com. Available from:
http://www.iorg.com./intranetorg/

Thoresen, K. (1995). Simple, but cumbersome. In: van der Veer, G.C., Henderson, A. and Coles, S. (eds.), *Proceedings DIS 97, Designing Interactive Systems.* ACM, New York, 385-394.

Wilson, J. and Rosenberg, D. (1988). Rapid prototyping for user interface design. In: Helander, M. (ed.) *Handbook of Human-Computer Interaction.* Elsevier, North-Holland.