

# CENTER FOR QUALITY OF MANAGEMENT JOURNAL

Reprint No. RP12700

**About This Issue** Page 2  
*David Walden*

**Creativity from Adversity — Three Breakthroughs  
at Maeda Corporation** Page 3  
*Matabee Kenji Maeda*

**Case Study of Trantex** Page 19  
*Described by Kaija Pöysti to Shoji Shiba and David Walden*

**Assessment of Results of Applying the Language  
Processing Method to Reduce the Experiment Cycle  
Time for the NASA Microgravity Research Program** Page 29  
*Dr. Mark C. Lee and John F. Newcomb*

**Beyond Lean: Building Sustainable Business and  
People Success through New Ways of Thinking** Page 37  
*Jamie Flinchbaugh*

**Theoretical Foundations of The  
Seven Infrastructures Model** Page 51  
*Matthew W. Ford, James R. Evans, Charles H. Matthews,  
and Gary Burchill*

**Volume 10, Number 2**

**Winter 2001**

# Case Study of Trantex

Described by Kaija Pöysti to  
Shoji Shiba and David Walden

Kaija Pösti is a successful entrepreneur. She established the software localization company Trantex, grew it, and eventually sold it. She also established and developed Bitwit, a training and new media design company.

Since selling Trantex in 1997, Ms. Pösti has been involved in the Finnish high-tech VC world. In 1998, she became a partner in IT Venture Development, a private VC fund in Finland. She also has participated in various national programs on information society and served on the board of the Finnish Broadcasting Corporation (YLE).

Ms. Pöysti currently lives in Boston, where she is a partner in Blue White Venture, a consulting company that helps Finnish high-tech companies enter the U.S. Market.

...

For several years, Shoji Shiba has been researching methods of breakthrough.

...

David Walden is the editor of this journal.

*Kaija Pöysti, founder of Trantex, was interviewed in May 2001 by Shoji Shiba and David Walden. Walden wrote up the Trantex case study based on the interview. Over the next three months Walden and Shiba provided successive drafts to Pöysti who in each case corrected errors and provided further detail. Shiba and Walden then drafted the final section of this paper, to which Pöysti again reacted. Throughout this iterative process, Pöysti clarified her own thinking about the essence of the Trantex case.*

## Creation of Trantex

In 1983 Kaija Pöysti was a graduate student in physics at the Helsinki University of Technology in Finland. She saw a request on a bulletin board from a company importing computers for help translating a little user's manual from English into Finnish. Having an interest in languages and computers, she answered the advertisement.

At the time and for some years to come, translations were the province of people trained in languages, and translation agencies had a stable of freelance translators to whom the agencies sent material to be translated. The actual translation activities were essentially done as cottage industry.

Kaija translated the booklet, her customer was satisfied, and other translation jobs came her way. In 1983 she started Trantex to do more translation work with herself as the sole employee, residing on the premises of another company for which she was freelancing. She rented one room and office services from the company in which she resided.

Along the way, it occurred to Kaija that the translation process might be converted from essentially an individual art into more of an industrial process. Also along the way, Kaija began to have more translation opportunities than she could handle alone, and she involved a couple of other people from the University of Technology.

At first Kaija just did a couple of translations as a way of making some money while studying. Her first customer was a company importing basic computers, and they did not pay much attention to quality. Her second customer, however, was HP, which paid much more attention to quality. When she saw how pleased they were with the quality they received (she understood computers and could write fluent Finnish), she realized that there were not that many people doing what she did. At that time, Kaija states, translators had no clue about computers, and computer people wrote jargon incomprehensible to anyone else. That had been fine before the PC, because computers were used only by computer people who wanted to have everything in English anyway. However, with the PC, the number of end users grew rapidly, and they needed text that was

not jargon and was in their own language.

For HP, Kaija worked on marketing materials which were targeted for end users and decision-makers. The amounts of text were still rather small. Her next customer was a company that worked for IBM, and through them she started working for IBM. IBM also was very quality conscious, and for them quality meant more than just good translations — it included timely deliveries and reporting. They also had much more material to be translated: an entire piece of software — IBM DisplayWriter, a text-processing system targeted entirely for lay users — so the software also had to be translated into Finnish. That confirmed her thinking that more and more translations would be needed. Therefore, she started looking for a few other customers.

## Growth of the company

Once the new company had five customers secured, Kaija started hiring additional people, who in the beginning worked only on an hourly basis, to accommodate fluctuations in the volume of work and to keep fixed costs down. The first four employees were from the University of Technology, like Kaija herself, and she corrected their style and grammar. However, the fifth person was a Finnish language major, some 20 years older than Kaija and a very important person for the company, because she started training the “techies” how to write fluent text. She also was a “grown-up,” which meant that they were no longer just a bunch of students but were beginning to develop as a company.

Another important characteristic of the company was that Kaija immediately began to look for more customers, although IBM could have kept their hands full easily. However, Kaija never wanted to be dependent only on one customer: First, anything can happen, no matter how good you are. Second, adding customers leveled the fluctuations in IBM volumes. Third, they learned more than they would with only one customer. In a service business, you need to become “part of the customer” and at the same time keep your own identity and keep learning. With one customer, it is relatively easy to do everything the way that customer wants, but then you stop innovating. With more customers, you learn all the time. They learned a lot from their customers. They also were fortunate to have companies like IBM and HP as first customers, with their existing systems and high quality standards.

## The Software Localization Problem and a Process for Doing It

With IBM DisplayWriter, Trantex got into software “localization,” which required even more technical skills than translating computer and software manuals.

“Localization” of a piece of software, such as the IBM DisplayWriter (or Microsoft Windows or Word) includes:

- The core software itself — all the words the software routinely displays to the user such as in menus, on windows, in dialog boxes, exception messages, and so on. These translations must be tested so that the translated words fit within the allowed graphical context, correctly describe the options offered to the user by the

software, have a selection letter that corresponds to the translation (such as Yes becoming Si in Spanish), etc.

- The help files — databases that can be accessed (often via a “Help” menu) that provide on-line key word searches for descriptions of commands and functions, provide tutorials in using commands and functions, etc. These files contain lots of hypertext links and search keywords that also need to be translated.
- “Printed” documentation — tutorial, reference, etc., manuals that are printed and bound or available on CD ROM, typically in PDF format. The manuals often include screen shots<sup>1</sup> of the software that have to be regenerated from the localized software. Sometimes developing a screen shot takes much time.
- Sample documents — letters, spreadsheets, presentations etc., which the user can use for practice. These must often be changed to correspond to local conventions, e.g., business letters have different styles, checks are not used in Scandinavia requiring examples of how to balance a checkbook to be rewritten, clip art must use soccer balls rather than American football helmets, etc.
- Web sites — most software systems nowadays also have a related web site with a lot of dynamic information about the software. The site needs to be translated and built.

<sup>1</sup> Screen shots are images included in the documentation of what would be seen on a computer display.

In the mid-1980s, much of the text to be translated was embedded in code — in worst cases in hexadecimal fields. Translators did not want to touch the code; and programmers did not want to give the code to translators, because the translators could introduce bugs in the code. In addition, most translators worked on typewriters or used dictation, which was impossible for software translation. Software translations required direct interaction with the computer files. This sounds so obvious now; but, in the mid-1980s, times were very different. Then many translators had no computers, so they could work only on printed material. Trantex had some customers who could hardly believe that Trantex could really process files and even compile them, changing as necessary local conventions such as decimal points, date formats, etc.

The software localization business was a major opportunity for Trantex. Localizing software was not about translating the text; it was about taking the English software and converting it into another language, including resizing the buttons, changing selection letters, etc. It required understanding of local conventions and also the ability to compile the translated files and to test that the translated version still worked. Unfortunately, there were very few tools at that time for all that work; most of it had to be done manually.

The typical cycle for localizing a product such as Word for the U.S. from Microsoft<sup>2</sup> into Word for Finland was as follows. Some time after software development of the new version of Word was started at Microsoft, an alpha- or beta-test version of the software became available. At this point the software files were transmitted to Trantex. At Trantex, the software files for the new version were compared with the files for the previous version, first manually and later using a tool called a *translation memory*. Use of translation memory tools allowed building each translation effort on prior versions of the translation, consistency across translators and documents, and horizontal exchange of information.<sup>3</sup>

<sup>2</sup> Another Trantex customer.

<sup>3</sup> Initially translation memories were available only for printed documents, but in time they were made to work with both documents and software. Translation memory tools were developed by a few companies, including IBM and Trados.

*A translation memory is a database in which a phrase-by-phrase translation from English to Finnish was maintained. The translation memory supported the work of the human translators.*

*The prior version of the English was stored in the translation memory along with its phrase-by-phrase translation into Finnish, where each phrase had, in effect, been translated by a human translator. The translation memory also had an algorithm for comparing the words in English from the new translation with the previous translation and finding differences. (This was done as a left-to-right match of strings; it was not based on syntactic analysis.) When a 100 percent match was found for a phrase between the previous and new version, the translation memory accepted the translation from the previous version as the translation for the new version. When the match was less than 100 percent ("fuzzy" in localization nomenclature, although that had nothing to do with "fuzzy logic"), a human translator was shown the old and new versions of the English along with the old version of the Finnish for the phrase in question (of course, in the context of sentences and paragraphs). Then the human translator changed the old translation appropriately. Of course, sometimes an entirely new phrase was found in the translation and, for this, a new translation was provided by the human translator. Thus, the translation memory greatly improved the efficiency with which a human translator could work, by highlighting only those phrases that needed new translation work, thus saving unnecessary (and perhaps inconsistent) retranslating of already translated phrases.*

*The translation memory was also a central database that could be accessed by multiple human translators working on different parts of the same localization project. Thus, it enabled new or updated translations of phrases to be shared by all human translators working on the project. This provided increased efficiency and consistency from one translator to the next.<sup>4</sup>*

*Although translation memory tools could greatly improve the efficiency and consistency of translations, they could also spread incorrect translations very quickly into the memory, unless carefully administered. Most translation companies sent their translation work to separate freelance employees, each of whom worked on his or her piece of the job, using the translation memories separately. This meant that different translators entered different translations into their own separate translation memories in their own computers. Trantex, however, hired people who worked on the Trantex premises, using the same translation memory database online. In addition, Trantex had language experts who constantly monitored the translation memory databases. This ensured much better consistency of the databases, so that each phrase only had one approved translation in the memory. In some cases, two different translations were necessary for one original, depending on the context. In these cases, it was important that the person doing validation made sure that both occurrences could be found.*

Some time after the alpha software was available, updates to the software began arriving. There would usually be at least five rounds of updates and could be as many as 40 rounds before final software release. The English phrases in each software update were compared with the existing translations in the translation memory and appropriately revised or new translations were added to the translation memory. Thus, the translation memory provided efficiency and consistency from update to update as well as from translator to translator. All in all, the translation memory supported a remarkable productivity and quality increase over what had typically been possible with human translators working alone.

The localization process for the software doesn't end with translation down to the phrase level. The software has to be tested to make sure it still operates sensibly in its localized format. As mentioned

<sup>4</sup> This technology-aided translation is called Machined-Aided Translation, or MAT, in distinction to Machine Translation, or MT, where there the computer uses grammar rules to do a translation.

earlier, translated phrases must fit into the available dialog box, they must describe the required interaction between user and software in a clear way in the new language, etc. Thus, Trantex found it needed to develop a test function to augment its translation function.

In addition to the software, help files and print documentation were also part of the package that needed to be translated. Usually all of this material came from the manufacturer at same time and went through the same revision cycles that software did. In some cases, the manufacturer sent the help files and documentation later.

Sometimes Trantex would try to skip the early rounds of revision of the help files and documentation until the software had stabilized, to try to reduce the number of update rounds. One change in a software screen could mean 150 changed screen shots, so it was important to try to do the screen shots as late as possible. Again, by hiring people who worked in the premises and worked as a team, instead of using individual free-lancers working at home and requiring pre-booking of their time,<sup>5</sup> Trantex could be more flexible than most translation agencies and do the screen shots as late as possible, when the software was almost frozen. This both saved work *and* improved quality, as the software was almost stable and the screen shots needed to be done only once. Having a team work on the translation also enabled Trantex to quickly move more people to do the screen shots.

An important flexibility issue was the fact that, unlike most other agencies, Trantex expected all people to be able to handle several tasks in the translation process. Most agencies, for example, used graphics specialists to do the screen shots. This meant that they had a limited pool of resources for that activity and had to start early in the process. Trantex had several people who were able to do both translations and screen shots. This meant that Trantex, for example, could quickly have five additional people doing screen shots at the end of the process, instead of having only two graphics specialists who were overloaded. Trantex did have specialists, too, but they concentrated on the final inspections, fine-tuning, and very special tasks that were small in volume. In any case, the help file and print documentation went through a translation memory similar to that used for the software itself.

Finally, an official release date for the software could be set by the manufacturer, and at this point the final version of the documentation would have to be finished, typically on a tight schedule that would allow the translated documentation to get to the printer in time for the documentation to be available with the product on the release date. The major additional problem in translating the documentation was translating the myriad screen shots in the documentation, that is, running the software to produce the screens that needed to be shot.

Of course, the Trantex combination of human translators being coordinated and aided by translation memories also applied to the sample documents and web sites that are now a part of many software products.

Initially there was not much MAT software available other than what Trantex was developing itself. Later on big manufacturers developed proprietary tools to help with various tasks, and Trantex developed its own macros with which they could do large searches and replaces across a large number of files and do consistency checks, e.g., always a space after a period ending a sentence, etc. Also, for example, IBM began to use SGML for all their documentation.

<sup>5</sup> Translations companies depending primarily on freelance workers and anticipating a large translation job had to pre-book the more skilled freelance translators so the company had the capacity to do the job once it arrived (for instance, a month in advance for a four month period). However, after a number of freelance workers were pre-booked, the customer might slip the time it shipped the work to the translation company, leaving the company with freelance workers on the payroll but no project for them to work on. Thus, companies depending on freelance workers as a way to avoid fixed costs in the end often had a nearly equivalent form of overhead.

## Key Success Factors

Over Trantex's life, Kaija and her people gradually evolved from her initial approach of sitting down alone with a manual to be translated and doing it essentially manually as a cottage industry to the technology-based system described in the previous section.

At one point, Microsoft did a survey of its localization vendors. The results of the survey showed Trantex having highest quality but not the highest prices.

When asked what she thinks the key success factors were, Kaija lists the following, in priority order.

- Thinking of localization as a process rather than a cottage industry.
- Bringing the technology and translation people and methods together, including cross-training so the people from the two cultures had good insight into each other's methods and cultures.
- Getting deeply familiar with the customers' (e.g., Microsoft's and IBM's) release methods, so the Trantex process could be best integrated with the customer process.
- Having people on company premises, working as a multi-skilled team and understanding the entire process, instead of using separate free-lance employees who only specialized in one task each.

The conventional thinking in the translation industry had been that due to changes in work volume, companies had to use freelance employees to minimize fixed salary costs. Trantex took a completely different approach, hiring the people and having them work at the Trantex office. Although this meant fixed salary costs, it also brought much more flexibility: people could be trained to be multi-skilled, sudden changes in schedules and volume of work were easier to handle because people could work with two projects at a time, and no pre-booking of free-lance employees was necessary. Naturally, it also required high enough work volume to smooth peaks and valleys, but that came from maintaining high customer satisfaction.

Trantex put rigorous emphasis on quality — meaning overall quality — of translations, delivery times, reporting, communicating with customers, etc. Kaija believes they spent much more effort on finding out how to provide overall customer satisfaction than many of their competitors. Trantex worked more like a consulting company than a translation company, which typically would have a “black box” approach (text in, text out, and fairly little communication in between). The black box approach worked for traditional small translations consisting of a few pages or for books translated by one person. However, software, which is constantly updated, requires constant communication and reporting, and an organized process for communications

## Sale of the Company

By 1997, Trantex had expanded to 220 people. An industry consolidation was happening, with the “big four” of Bowne, L&H/Mendez, Lionbridge, and Berlitz buying up the smaller regional or national companies. The pressure for this consolidation came substantially from the desire of the big customers (e.g., Microsoft, IBM) to deal with only a few multilingual suppliers rather than single language suppliers in each country.

Trantex was courted by several companies. After some soul searching, Kaija et al. noted they had been working very hard for many years, they had an excellent financial offer on the table, if they didn't sell they might not be able to compete in the face of the industry consolidation, and the dot-com balloon was soaring and Trantex was facing increasing competition for its best people. Kaija and her husband (a long time key participant in Trantex) were the two major shareholders, owning over 60 percent of the shares, and there were eight other shareholders. Most of the others were turning 30, with young families and large mortgages, and to them a large windfall made a big difference. Overall, it made more sense to secure a major financial gain than to fly into an uncertain future. Thus, the Trantex employee/shareholders decided to sell.

Within that same year, nearly twenty local translation companies were bought by the big companies. Curiously, this consolidation eventually backfired on the big customers because the big translation companies reduced quality to reduce costs. (They were publicly listed and under pressure to increase their bottom lines in an people-intensive industry, which often translates to cost reductions, which in this case resulted in reduction of quality.)

## Trantex and Models for Breakthrough

In an investigation of a number of instances of breakthrough in business over the past few years,<sup>6</sup> Shoji Shiba has observed:

- a cycle for breakthrough
- three stages of breakthrough
- an infrastructure for breakthrough

<sup>6</sup> Shoji Shiba and David Walden, *Four Practical Revolutions in Management* (Productivity Press: Portland, Oregon, 2001) chapter 27.

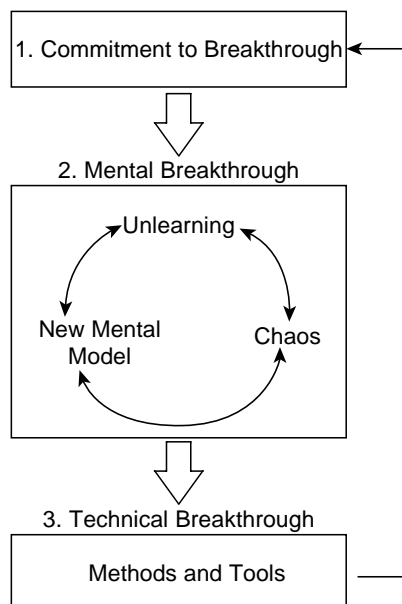


Figure 1: Breakthrough cycle

Figure 1 shows the cycle of breakthrough. First, a change leader discovers a source of motivation for commitment to breakthrough. Second, a mental breakthrough is needed; this typically begins with an unlearning step where previous conventional wisdom is discarded. Third, there is a realization step, where the necessary technical breakthroughs are achieved (these “technical” breakthroughs can involve technology, new



ways of looking at the market, process changes, etc.). These steps are repeated for successive breakthroughs.

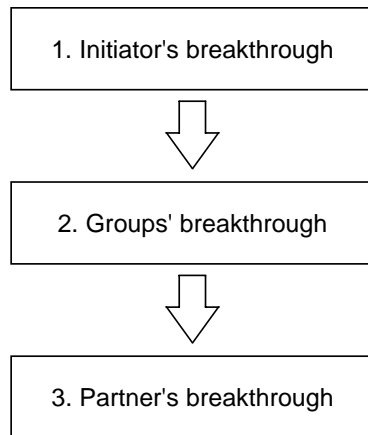


Figure 2: Breakthrough stages

Figure 2 shows the three phases of breakthrough. First the leader (the initiator) has his or her own breakthrough. Second the initiator works to collect an appropriate team and provide an environment in which team members can embrace the breakthrough and make it their own. Third, there is activity to involve an appropriate partner or partners (for example, potential customers) in refining and diffusing the breakthrough.

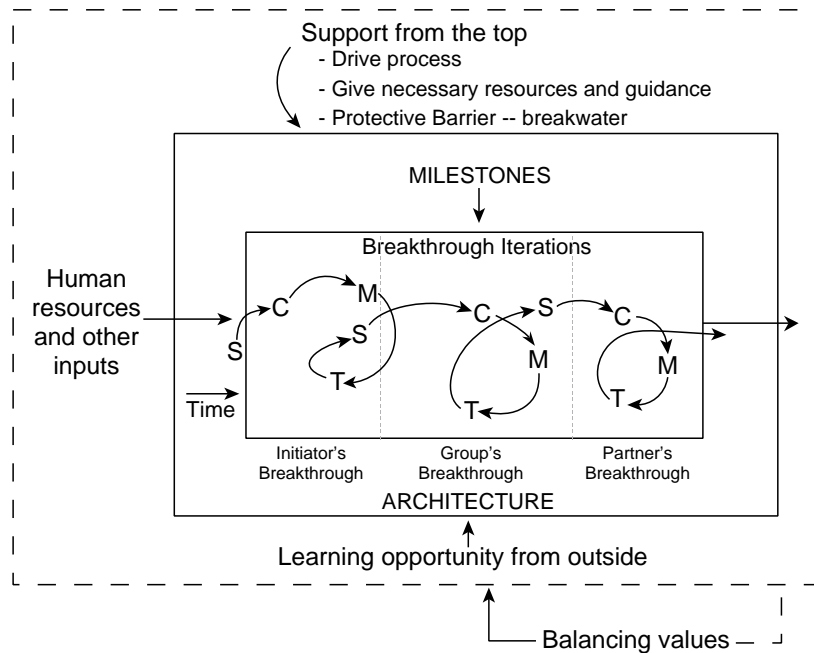


Figure 3: Breakthrough infrastructure

Figure 3 shows the infrastructure for breakthrough. The three stages of breakthrough are shown as three breakthrough iterations across the middle of the figure. Each breakthrough iteration is an instance of the breakthrough cycle shown in Figure 1 — Commitment, Mental breakthrough, Technical breakthrough — although in the Figure 3 each cycle includes an additional step of Sensing that a change is needed. The total process involves appropriate milestones, a supportive architecture (structure, processes), human, financial, technological and other resources, time, on-going learning from the outside world, and especially support

from the top to provide the necessary resources, to drive the process and, often, to protect the breakthrough activity from the rest of the organization which does not yet understand the need for the breakthrough activity.

As hinted in Figure 3 and its description of the infrastructure, a *lot* of force has to be brought together to create breakthrough. However, with so much force for breakthrough, there must also be a moderating influence to prevent reckless efforts, such as efforts too unconnected to the core capabilities of the organization. In Figure 3, these are called balancing values and, of course, to some extent these are part of a feedback loop reacting to the results of the force for breakthrough. However, waiting for feedback may be too slow. Thus, to a considerable extent these balancing values must modulate the forces for breakthrough as the latter are being applied.

The Trantex case is more or less consistent with the cycle, stages and infrastructure just sketched.

With regard for the cycle of breakthrough, Kaija initially saw a need for an alternative to the conventional approach of giving one translation job to one translator who was a language specialist. Her commitment to breakthrough grew as she struggled to do translations herself. Kaija's mental breakthrough was to see that technology and technology people could be used in conjunction with language specialists. The technical breakthrough involved creating a new business model of a translation organization (versus freelance translators) and applying computer technology (and people).

With regard to the stages of breakthrough, Kaija began with her own (initiator's) breakthrough. She then gathered a group of people and they built a new kind of company applying computer technology in a new way (group's breakthrough).

Whether the Trantex case involved a genuine partner's breakthrough stage is debatable. On the one hand, Trantex clearly began to work with its customers (IBM, Microsoft) in a new way — always striving to make Trantex's effort more compatible with the customer's larger process rather than only striving for quality translations. Ultimately, however, rather than continuing to expand as they diffused their new approach with many partners, languages and geographic areas, Trantex remained essentially a one language, one country provider. When faced with the requirement to expand or possibly succumb to overwhelming competition, they chose to merge themselves with a larger company. In some sense, this was another form of partnership — trying to bring the approach their customers embraced to their new partner/parent.<sup>7</sup> In another sense, this was just cashing in and giving up the effort to diffuse.

<sup>7</sup> Unfortunately, they could not, for a complex set of reasons.

Finally, the Trantex case definitely involved elements of the infrastructure for breakthrough.

**Inputs.** Trantex brought technology people and computer technology together with language people.

**Architecture.** An architecture was created for people working together in incrementally improving centrally stored translations and for integrating with customers' processes.

**Breakthrough iterations.** The initiator's breakthrough, group's breakthrough, and to some extent partner's breakthrough were described above.

**Breakthrough cycles.** The initiator's breakthrough cycle was already described. The group's breakthrough involved Kaija gathering a com-

mitted team (particularly involving technology people), getting them to think about translation in a new way (language people to embrace technology, computer people to learn about quality prose), and developing the appropriate computer systems and customer interaction processes.

**Learning opportunity from outside.** During each stage, there was ongoing learning from outside. Kaija first learned about how translation was done and how it could be done better. Group members with different backgrounds then learned from each other. The entire group learned from customers.

**Time and milestones.** As described, this case took place over a period of years, and there were constant milestones, driven by business pressures.

**Support from the top.** Obviously, Kaija provided the initial drive to start and grow Trantex. As a relatively small company, Kaija and a few others on the top management team continued to drive an ongoing change/breakthrough process, acquiring needed resources (new hires, the necessary financial backing). Because Trantex was initially a startup company, it didn't have to worry about adverse reactions from other divisions and corporate offices that so many breakthrough activities have to struggle against.

**Balancing values.** It seems clear that Kaija and Trantex had a core ideology that allowed and encouraged change but that prevented them from reckless change. This core ideology involved the mixing of linguistic skills and technology and with computer skills and technology in a way that embedded Trantex in the customers' processes.

At the end of the day, however, the infrastructure developed at Trantex seems to have been unable to undertake the next stage of breakthrough, (e.g., to becoming more than a national company) without being deemed more risky to the Trantex people than selling the company was, at least in the face of the circumstances Trantex then faced. Kaija elaborates as follows on the situation in which they found themselves:

*The environment was changing very rapidly: localization companies were being bought in all countries very quickly, meaning that Trantex would have lost its partners (Trantex did run multilingual projects for its customers, buying the translations from partners who were similar companies to Trantex in their respective countries), and there would have been too little time for Trantex to develop new opportunities on its own. Also, the dot-com boom was heating up, and people were enticed to change companies rapidly by offering them higher salaries and stock options. In many cases, even the customers of Trantex tried to hire people from Trantex. An attempt to take lots of outside capital and join the buying spree would only have driven prices for the national companies higher, to a level which would have become far too high, especially in hindsight, as the economy is now slowing down. So although the infrastructure was providing good results, it was going against the tide at that time (1997).*

*In the environment of today (2001), things are quite different again, and it is very probable that a similar infrastructure will emerge from one or more companies. Taking such an infrastructure multinational will require finding like-minded people in different countries and perhaps "franchising" the concept to them.*



The Center for Quality of Management Journal is a forum for disseminating the experience of organizations learning to implement modern management practices. It seeks to capture experiences and ideas that may be useful to others working to create customer-driven, continuously improving organizations.

The CQM Journal is refereed. However, it is not an academic publication. Experiences and ideas will be published if they seem likely to be useful to others seeking to improve their organizations.

**Send to:**

The Center for Quality of Management Journal  
 Editorial Department  
 One Alewife Center, Suite 450  
 Cambridge, MA 02140  
 Tel. 617-873-8950 Fax 617-873-8980  
 E-mail: [publications@cqm.org](mailto:publications@cqm.org)

If you have thoughts for a paper and you would like to discuss it with us, please write, call or submit an outline. We welcome your ideas.

**Production Team**

**Eric Bergemann**  
 Publisher

**David Walden**  
 Editor

**Janice Hall**  
**Kevin M. Young**  
 Design & Production

**CQM Officers**

**Ray Stata**  
 Chairman

**Gary Burchill**  
 President

**Paul van der Wansem**  
 Treasurer

**William Wise**  
 Clerk

**Final Manuscript Requirements:**

Entire manuscript should be double-spaced, including footnotes, references, etc. Text should include all the elements listed below. Generally, The CQM Journal follows the editorial principles of The Chicago Manual of Style. We strongly prefer submissions in electronic format for all text and as many of the figures as possible. IBM-based software (particularly Microsoft Word for Windows) is preferable to Macintosh-based software if you have a choice, but is by no means a requirement.

**Please include:**

1. Title page, stating the type of article (e.g., 7-Step case study, research paper, short communication, letter to the editor, etc.), main title, subtitle, and authors' full name(s), affiliation(s), and the address/phone/fax of the submitting author;
2. All needed figures, tables, and photographs (see below);
3. Footnotes (if appropriate), numbered consecutively from the beginning to the end of the article;
4. Reference list, if appropriate.

**Figures, Tables and Photographs:**

If you can, insert each figure or table into the text where you would like it to fall. Figures should be composed to conform to one of two widths: 3 1/8 or 6 1/2 inches. The maximum height for any figure is 9 3/8 inches. Text within figures should not be smaller than 5 points and lines not less than 1/4 point at the figure's final size. Figures should be labeled with the figure number underneath and title on top. Be sure that the text mentions each figure or table.

Please retain separate PICT or TIFF files of figures generated in drawing programs and a file with the text only for final submission.