# Applications of TRIZ to IT: Cases and Lessons Learned

**Kas Kasravi, HP Fellow, Hewlett-Packard**
***kas.kasravi@hp.com***

## Abstract

*The results of applying TRIZ to several information technology (IT) problems are summarized and reported. Two case studies demonstrate examples of applying the 9 Laws to predict the next state of technology in IT Services and Business Intelligence. Four case studies demonstrate examples of problem formulation and conflict resolution through the application of the 40 Principles to IT problems, including data security, CPU cycle costs, private clouds, and leased asset management. In each case, a summary of the business problem is presented, followed by an overview of the TRIZ analysis and the outcome. TRIZ helped uncover novel and innovative solutions in these cases. These applications offered an opportunity for assessing the feasibility of using TRIZ in IT. The lessons learned regarding the applications of TRIZ to IT are also presented. These cases demonstrated that TRIZ is a viable approach for systematically solving certain IT problems.*

## Introduction

TRIZ is an inventive problem solving methodology that has helped solve numerous engineering problems. Historically, the focus of TRIZ has been on addressing complex physical problems in the engineering domain. Literature search suggest that there have been a number of instances where TRIZ has been applied to problems in other domains, including IT.

The focus of this paper is to examine several application of TRIZ to Information Technology (IT), and in particular to IT Services.

IT Services is defined as the study, design, development, implementation, support or management of computer-based information systems, particularly software applications and computer hardware [1]. Examples of IT Services include providing the back-end systems for insurance claim processing, ATM transactions, airline reservations, computer facilities management, and operating help desks. A conservative estimate of the worldwide market size of IT Services is reported to be $131B by 2013, with other estimates as high as $450B/year. A steady growth of 3.8% CAGR is also estimated 2013 [2]. Therefore, IT Services is a large segment of the economy, with many opportunities for innovation and problem solving.

In this paper, we briefly review a number of applications of TRIZ to IT Services. Specifically, we address the following two distinct classes of problems:

1. **The Next Big Thing** – where, we investigate the future development of a system.
2. **A Specific Problem** – where, we investigate solutions for an existing problem.

In particular, the following problems are addressed:

**The Next Big Thing**

1. Future of IT Services
2. Future of Business Intelligence

**A Specific Problem**

3. Secure but Open
4. High CPU Cycle Costs
5. Private Cloud Conflict
6. Leased Asset Management

For confidentiality reasons, names and other proprietary information are excluded from the case reviews; however, the information relevant to the core of the problems and the application of the TRIZ methodology are described.

This paper assumes that the reader is already familiar with the basic concepts of TRIZ, and therefore no introduction to TRIZ is provided.

In the absence of mature and well-developed methodologies for applying TRIZ to IT problems, each case took a different approach to using different aspects of TRIZ, and more work remains to be done. Thus, the following analyses are considered to be "inspired" by TRIZ and experimental in nature.

## Case 1: Future of IT Services

IT Services, as defined earlier, is the study of approaches for levering the IT technology to meet business needs. IT Services have been evolving over the past 50 years, and affect virtually all businesses. Therefore, predicting the future of IT Services is deemed to be a useful exercise for both the providers and users of IT Services. Predicting the future is always a dangerous business; but, it's the journey that is worth it. Below, we reference a study that we recently conducted [3].

The study of the future of IT Services leveraged the Technology Evolution Vector to gain insights into where IT Services are heading, and then used the 9 Laws of TRIZ to develop specific predictions. This analysis was built on a brief historical trend in IT.

The Technology Evolution Vector suggests that systems evolve from a Rigid state, to a Modular state, then to a Programmable state, and finally reach an Autonomous state. For IT Services, we define these phases as shown in Table 1.

In order to position the current state of IT Services along the evolution vector, the history of IT Services was reviewed and summarized in Table 2.

The trends represented in Table 2 suggest that IT Services is currently in the early stages of the Programmable phase. Therefore, any further development, according to TRIZ, will be focused on characteristics defined by the Programmable and Autonomous phases.

To fine-tune the future state of IT Services, we next leverage the 9 Laws. Table 3 presents ideas about the future of IT as inspired by the 9 Laws. In practice, only one or two laws may typically be used, but in our exercise we leveraged all 9 laws for completeness and experimentation. We concluded:

- IT Services will be:
    - Transparent to the end user
    - Embedded in other products/services used by consumers
    - Substantially less expensive
    - Much more reliable
    - Much more flexible

- IT Services will be highly automated:
    - Artificial intelligence as the foundation of automation
    - Very few “people” will be involved
    - Value will be in the IP owned by highly specialized people and IT companies

- IT hardware will be:
    - Integrated
    - Plug-compatible
    - Energy efficient (often relying on ambient energy sources)
    - Small and inexpensive

Above recommendations suggest that common notions such as outsourcing, insourcing, and off-shoring will become irrelevant. Instead, we concluded a new model that we called eShoring™.

We further defined that eShoring™ as the delivery of IT Services using fully automated systems, and proposed that it requires an entirely new business model with the following characteristics:

- Much more complex than simply adding automation to existing delivery models
- Different sales, design, development, and delivery strategies
- Creating and maintaining back-end systems at the global level
- Cloud computing is a pre-requisite
- Simplicity will be essential
- Trust and liability will be significant factors – different risk models
- Different HR models

We also concluded that the eShoring™ model will require far fewer people, but these people are much more specialized. Value will be placed on the intellectual property for the automation that will drive the future automated IT systems.

## Case 2: Future of Business Intelligence

Wikipedia defines Business Intelligence (BI) as "Computer-based techniques used in spotting, digging-out, and analyzing business data, such as sales revenue by products and/or departments or associated costs and incomes". BI typically involves ETL, OLAP, data mining, text analysis, data visualization and similar technologies required for making sense of large volumes of data.

BI supports decision making via fact-based analysis of data. BI has been steadily gaining popularity over the years.

Business strategists and BI tool developers are interested in knowing where BI is heading in the future. Therefore, we used the Technology Evolution Vector and the 9 Laws to gain insights into the future of BI [4]. This study also leverages the historical trends in BI, as well as relevant enablers and drivers.

As shown in Figure 1, the process of BI consists of five major stages: Data, Analysis, Information, Decision, and Action.

In the early days (BI-0 phase), human intelligence was responsible for performing all the tasks, from the initial observation (data), to analysis of the facts and digesting them into useful information, then making decisions based on that information, and finally taking an appropriate action. Over the years, BI technologies have been steadily taking over these tasks and performing them better and faster for the users (BI-1 and BI-2). Table 4 illustrates the historical trends in the development of BI solutions (BI-0 to BI-2), and maps the developments to the appropriate phase in the evolution vector.

The Technology Evolution Vector suggests that BI has been moving from left to right along the stages shown in Figure 1; thus, it's plausible to conclude that in the future BI will move into decision making and actually performing actions for the users, as shown in Figure 2.

To further investigate the characteristics of the future of BI (i.e., BI-3) we first consider the business and technical enablers as shown below:

**Drivers**

- The intense global competition, requiring enterprises to instantly tune to the rapidly changing market demands
- The need to make faster decisions and respond to the market changes
- The need to make more fact-based decisions to avoid errors and lost opportunities
- The economic recovery requires timely actions more than ever before
- Information explosion can be leveraged

**Enablers**

- Business Enablers
  - Increased recognition of the value of fact-based decision making
  - The recognition that BI is more effective when embedded within other business processes
- Technology Enablers
  - Web 2.0 & social media
  - Enterprise data warehouses
  - Mashup data
  - RFID, sensors, and location data
  - Cloud computing

We also consider the insights provided by the 9 Laws of TRIZ, as shown in Table 5.

To summarize the findings, we conclude that the future BI systems will have the following characteristics:

- More transparent, and embedded within other applications and processes
- More pervasive, and used more broadly by many more knowledge workers
- Less expensive
- Automated decision making and performing actions

Hence, we observe a convergence in BI, as shown in Figure 3. The trust and liability hurdles identified by this process are significant risk factors that need to be addressed by the future BI practitioners.

## Case 3: Secure but Open

A beverage manufacturing company's IT department wanted to improve their IT security system. In particular, they wanted their systems to be more secure, but they didn't want to burden the users with additional security measures. Another words, they wanted a system that was more secure and yet more open (a classic TRIZ trade-off). The IT department was initially looking for a new technology to address this problem.

A problem redefinition exercise (Figure 4) quickly demonstrated that the problem was not technical in nature, but it involved a lack of business case, management support, and funding.

A problem formulation exercise (Figure 5) showed that IT security is an auxiliary tool that does not support the main function of the company, i.e., sell beverage to the consumers. It was then clear that the original "secure but open" problem was the result of lack of funding that was not being provided to IT. So, the question became one of how to obtain funding from the management to acquire the needed security enhancements.

The TRIZ ideality tactics, among others, suggested the following:

- Eliminate the auxiliary tool (AT)
- Enhance the primary function (PF)
- Introduce auxiliary tool (AT) to eliminate harmful action
- Create new auxiliary function (AF)

These ideality tactics and the problem formulation in Figure 5 suggested that we need to create a link between IT security and the Consumer. This insight and a short brainstorming session led to the concept of creating an entertaining online virtual world for the consumers, where access and activities were based on codes printed on the bottles/cans of the beverage (Figure 6). So, the more beverages the consumer purchased, the more capabilities they had in the virtual world. Also, opportunities for additional sales and marketing presented themselves in the virtual world.

Because IT was now connected to the consumer, that the consumers' information had to be protected, and a new revenue stream was identified, IT was in a much better position to seek additional funding for enhancing the security systems. Meanwhile, the company discovered additional opportunities for revenues.

## Case 4: High CPU Cycle Costs

A financial institution had installed a specialized software product for processing financial transactions, and the software was generating excessive CPU cycles, leading to unexpectedly high IT charges. The software was being used in an unintended fashion, and the excessive CPU cycles could not have been predicted. This situation was very critical, because the cost of processing the financial transactions was greater than the revenue generated. Several months of investigation by the parties involved had not yielded satisfactory results. As the last resort, a TRIZ analysis was conducted.

During a 15-hour analysis, the environment was modeled as shown in Figure 7. The problem formulation exercise led to identifying several business conflicts, and specifically an imbalance between the contractual terms for providing IT services and the cost recovery model. Application of TRIZ ideality tactics immediately led to several issues for consideration:

1. **Enhance the Inadequate Actions**
    - Introduce high-margin value-add services to customers to offset higher operational costs
2. **Tool must match the Object**
    - Restructure fees
    - Higher fees for receiving transactions
    - Higher flat fees

Also, several technical issues, such as duplication in back-up and recovery actions were identified using the application of various TRIZ principles. More importantly, the application of ideality tactics led to an insightful outcome:

3. **Use the Environment to replace a Tool or Object**
    - Transaction processing to be performed locally by users
        - Minimize hardware costs
        - e.g., use a grid model (non-mainframe platform)

4. **Eliminate the Tool by having the Object act on itself**
    - Development of a new generation of financial transaction technology
        - Agent-based transactions
            - Each software agent processes itself on any available hardware
            - Software agents operate in a grid environment
            - Real-time processing

Above insights have led to a patent application that uses software agents to process transactions in real-time on existing hardware. The benefits of this invention over traditional batch processing are:

- Near real-time performance (vs. overnight)

- Leveraging existing IT infrastructures'' idle resources (vs. investing in high-end servers), saving money
- Creation of a virtual cloud environment for transaction processing

## Case 5: Private Cloud Conflict

Cloud computing is currently the hottest topic in IT. The cloud metaphor offers many advantages such as on-demand resource allocation, lack of capital investment, and lower operational costs. However, there are concerns about data security in the cloud, especially when private data is sent to off-site computers for storage and processing. Hence, many enterprises are looking into implementing private clouds, which are cloud environments installed within a company's firewall.

Private clouds offer many advantages, but they create an interesting conflict; where, on one hand, cloud computing promises reductions in IT costs, and on the other hand the organization has to invest in the required infrastructure. This conflict has caused many companies to adopt less than optimal strategies, e.g.:

- Stay with the current IT infrastructure (and lose the benefits of cloud computing)
- Wait and observe (and fall behind)
- Multi-year investigation and studies (expensive, and late to market)
- Leverage public cloud (and take certain security risks)
- Deploy a private cloud (and pay the infrastructure costs)

We leveraged TRIZ, with the goal of identifying strategies for realizing the benefits of private clouds in a conflict-free manner [5].

Figure 8 illustrates a cloud model, identifying the security risks when multiple tenants are involved. Figure 9 shows the typical conflict between public and private clouds. We reduced this conflict to:

**Security vs. Versatility**

This contradiction suggested the following TRIZ Principles [6]:

- Segmentation
- The Other Way Round
- Intermediary
- Another Dimension

These Principles suggested the following solutions:

| | |
|---|---|
| **Segmentation** | 1. Separate the private data from non-private data.<br>2. Separate critical applications from non-critical applications.<br>3. Use a smaller private cloud for private data and critical applications, and use public cloud for everything else. |
| **The Other Way Round** | 4. Calculate the cost of not implementing a private cloud, and justify the investment.<br>5. Find other trusted users for the private cloud, who could help cover some of the expenses (community cloud) |
| **Intermediary** | 6. Have a third-party finance and/or manage the private cloud, and charge for usage only. |
| **Another Dimension** | 7. Instead of using a private cloud or a public cloud, consider a combination of both, where each cloud provides different services (hybrid cloud).<br>8. Instead of a single-tenant model, consider a multi-tenant base of trusted users. |

There are multiple enterprise strategies for leveraging cloud solutions in view of IT Departments and Business Units, as shown below:

| | **IT Department** | **Business Unit** |
|---|---|---|
| **Add benefits** | *Augmentation*<br>(expand current capacity) | *Exploitation*<br>(create new revenue stream) |
| **Reduce Costs** | *Replacement*<br>(replace current IT systems) | *Integration*<br>(extend enterprise connections) |

Combining the TRIZ recommendations and enterprise cloud strategies, we now have a set of solutions for addressing the private cloud conflict as a function of enterprise objectives, as shown below:

| | **Strategy Recommendation** | **Enterprise Strategy** |
|---|---|---|
| 1 | Separate the private data from non-private data. | Replacement |
| 2 | Separate critical applications from non-critical applications. | Integration |
| 3 | Use a smaller private cloud for private data and critical applications, and use public cloud for everything else. | Augmentation |
| 4 | Calculate the cost of not implementing a private cloud, and justify the investment. | Replacement |

| | | |
|---|---|---|
| 5 | Find other trusted users for the private cloud, who could help cover some of the expenses (community cloud) | Integration |
| 6 | Have third-party finance and/or manage the private cloud, and charge for usage only. | Replacement |
| 7 | Instead of using a private cloud or a public cloud, consider a combination of both, where each cloud provides different services (hybrid cloud). | Augmentation |
| 8 | Instead of a single-tenant model, consider a multi-tenant base of trusted users. | Integration |

## Case 6: Leased Asset Management

IT service providers often lease computing assets to their clients. These assets are placed in service for specific periods of time, and then collected and disposed appropriately. Effective management of the leased assets requires them to be tracked at all times. Lack of asset tracking can be costly at the end of the lease.

In practice, tracking leased assets in large organizations is very difficult. Typically, there are thousands of assets involved, which are covered by numerous and independent service contracts, covering multiple locations. The assets are often moved, and then moved again to other locations. The users of the assets change over time; and, the new users may not be aware of the lease terms. Further, the assets may be upgraded or reconfigured during the life of the lease; some of the changes may not be approved. To further complicate the matter, multiple databases are typically used to track the leased assets, and these databases may be independent of each other. In some cases, even spreadsheets are used instead of databases. Despite the apparent chaos, every decision has a rational basis at the time it is made. It would appear that a common tracking database would solve the problem, but for various business, financial, and contractual reasons that is often not an option.

A limited TRIZ exercise was used to examine options available to the leasing managers. This exercise revealed that the required asset data exists, but in multiple formats and in multiple locations. It was also discovered that the business managers had a perception that creating a common database will require too much investment and cannot be justified.

Figure 10 is a sufield representation of the problem. In this model, Leased Assets (object) must be subjected to effective Data Management (field), so that an adequate leased asset Data Repository (object) could be created. The Data Repository could then be tracked (field) in order to fulfill contractual requirements (object) at lease end. In this model, the contractual requirements are harmed because an adequate Data Repository does not exist. Therefore, the focus needs to be placed on the fidelity of the Data Repository in order to eliminate the subsequent harm.

The Data Repository object is a detection/measurement sufield. A standard for this class of sufield is "If there is a problem requiring measurement or detection in a system, then the system should be changed so that the need to measure or detect is eliminated" [7]. In order to search for possible options to eliminate measurement (i.e., create a common database), or at least reduce the complexity of the problem, we reviewed several TRIZ principles for guidance:

- **Self -Service** – Provide users with self-reporting tools, or use web-crawlers.
- **Cheap Disposable** – Infer missing data based on historical data.
- **Fluid** – Use mash-up tools to collect data in disparate formats and create a common database.
- **Discard & Recover** – Use a dynamic data model to capture the incoming data from users, web-crawlers, or mash-up data sources.
- **Hurry** – Collect/Report/Discard data in a single action.

The solution inspired by these principles could be a virtual database created by pulling together the available data via user input, web-crawlers, and mash-up to create the necessary reports, and then discard the data. This way, the current environment can be used to meet the objectives of having a Data Repository for measurement, without actually having one.

Above solution requires changes to the current tools and business models, which may not be practical. Therefore, we looked into another standard that suggested "If internal fields are not available, use fields present in the system's environment" [7]. A principle that proved insightful in this case was the segmentation principle:

- **Segmentation** – Have a third-party create a common database outside of the current environment.

It is common practice in business to seek the help of third-parties who specialize in specific tasks or services. In our case, the field present in the environment turned out to be a third-party provider of services for managing and tracking asset data. Thus, the proposed solution was to outsource asset data tracking to a third-party who specializes in this art.

## Lessons Learned

We consider above cases to be of an experimental nature, where we attempted to assess the feasibility and effectiveness of using TRIZ to address IT problems. In our opinion, TRIZ worked well, in a sense that it helped create new and innovative ideas in each case. The exercises were rapid, typically requiring 3-6 hours. The longest exercise took 15 hours of actual analysis. However, for reasons beyond our control, not every solution could be implemented.

The following list provides a summary of lessons we have learned from applying TRIZ to IT:

1. TRIZ can address a broad set of problems in the IT Services space
2. Innovative solutions discovered in nearly every case
    - Several patent applications in progress
3. Technology Evolution Vector works well as is
4. 9 Laws work well as they are
5. Problem re-definition often leads to a better understanding of the root causes
6. Problem Formulation works well as is
    - The most challenging aspect of the analysis
7. 39 Constraints were not applicable
    - Not certain if they are even needed
8. 40 principles mostly work well as they are
9. Both TRIZ and the IT problems need to be fully internalized
10. "Selling" TRIZ to (non-engineering) IT has been a challenge

Further, these cases presented a need for additional work in the following areas:

- Clear definition of the Technology Evolution Vector as applies to IT
- Clear IT definitions/examples of the 9 Laws
- Investigate whether IT Constraints are needed
- Development of a simple problem Formulation tool (preferably in Visio)
- Clear IT definitions/examples of the 40 Principles
- Conduct a full-blown analysis using ARIZ

## References


1- http://www.ehow.com/about_4689779_what-definition-information-technology.html

2- Storm, T., March 2009, "Worldwide and US IS Outsourcing Services 2009-2013 Forecast", *IDC* #217341, vol. 1.

3- Gibson, J.D., Kasravi, K., "Predicting the Future of IT Services with a Structured Innovation Model", *Society for Design and Process Science*, June 10, 2010.

4- Kasravi, K., "A convergence in Business Intelligence" *Cutter IT Journal*, Vol. 23, No. 6, pp. 6-12, June 2010.
5- Kasravi, K., "Strategies for Conflict-Free Private Clouds", *HP Technology Forum*, June 2010.
6- CREAX Innovation Suite 3.1.
7- Fey, V., Rivin, E, "Innovation on Demand", Cambridge University Press, 2006.

## Figures and Tables

**Figure 1. The Stages of Business Intelligence**

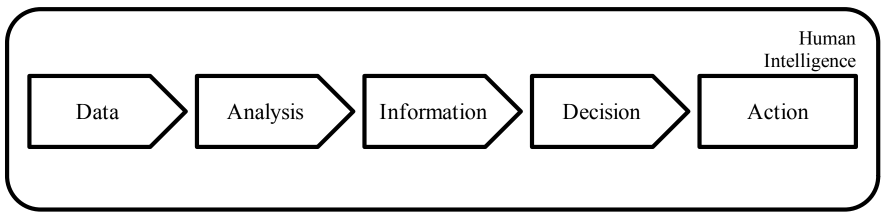



**Figure 2. Advances in Business Intelligence**

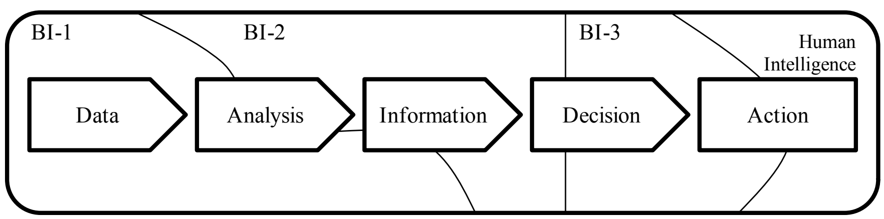
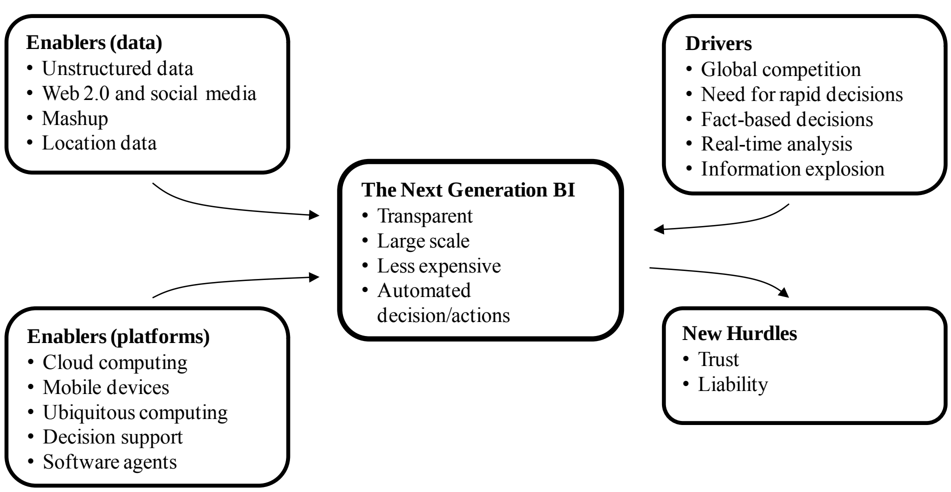
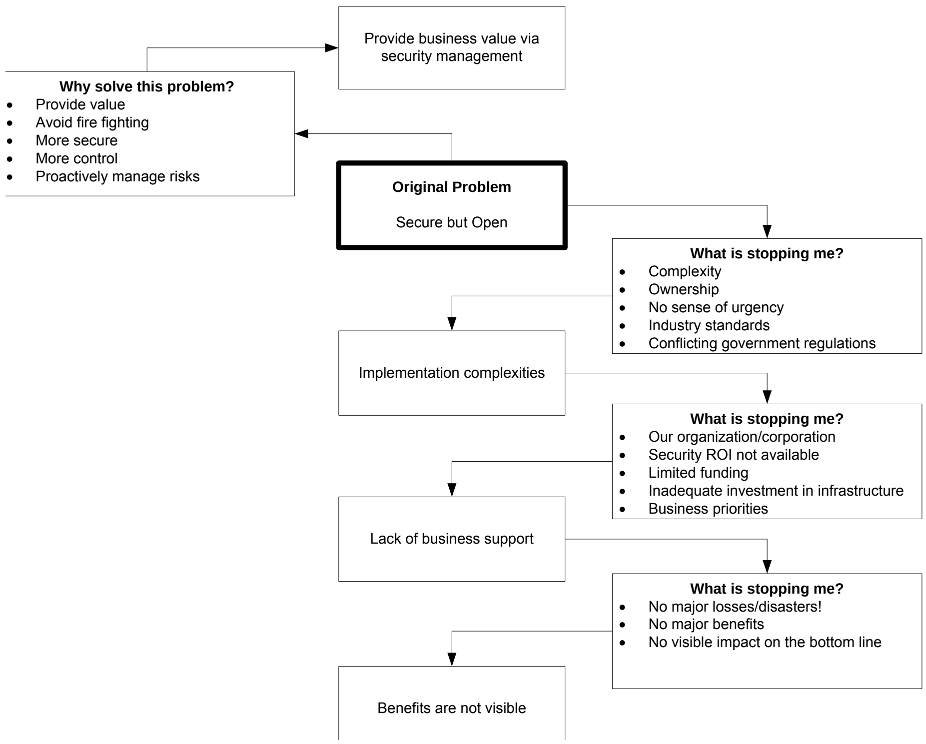


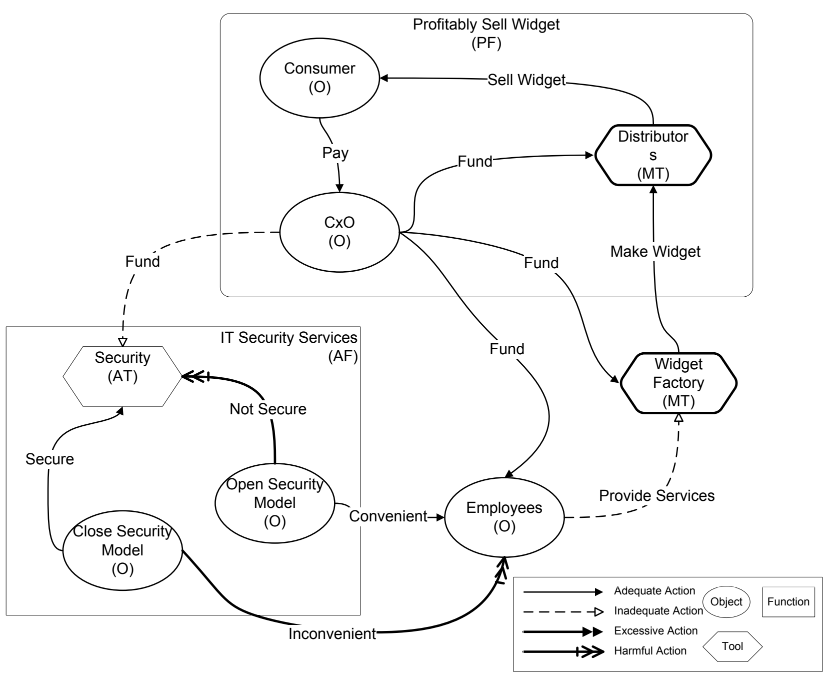

**Figure 3. A Convergence in BI**



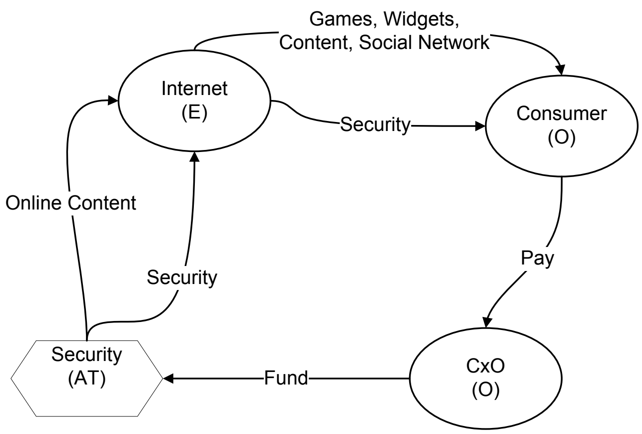
# Figure 4. Problem Redefinition



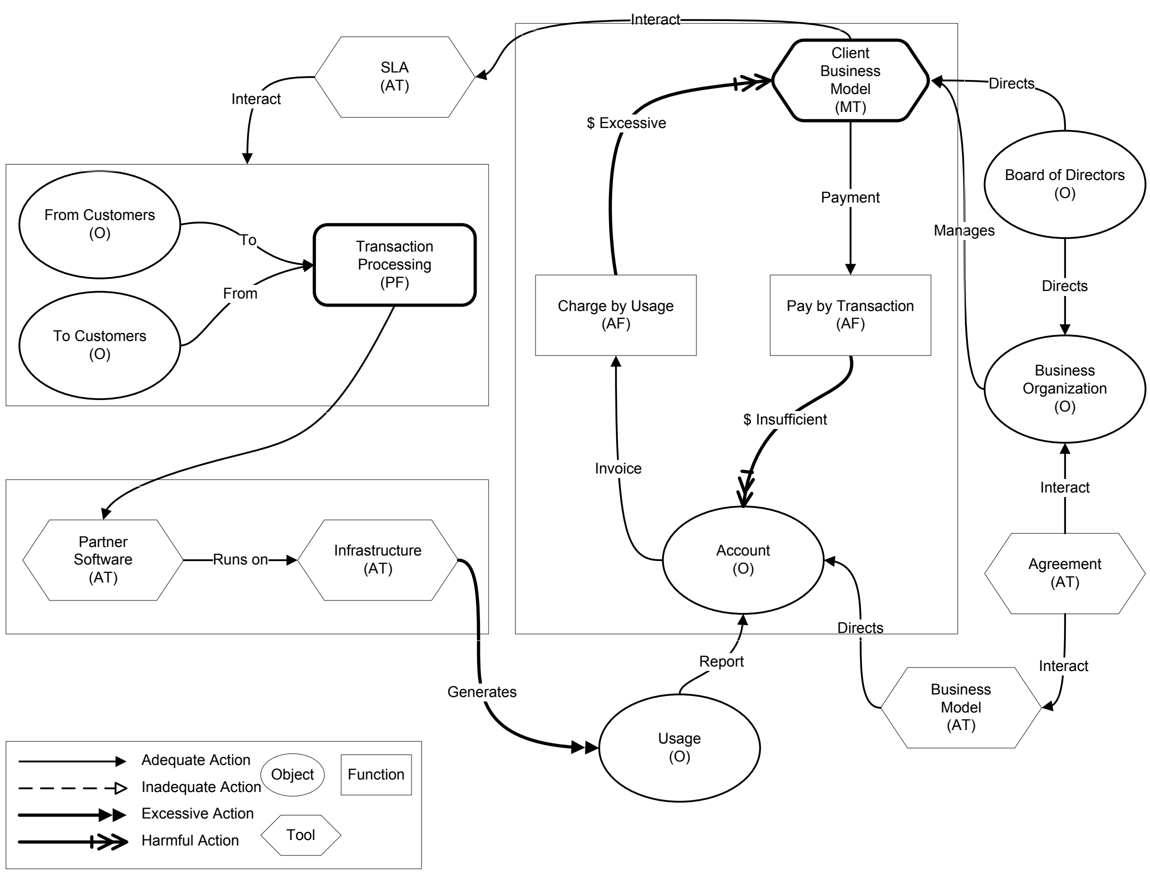

# Figure 5. Problem Formulation



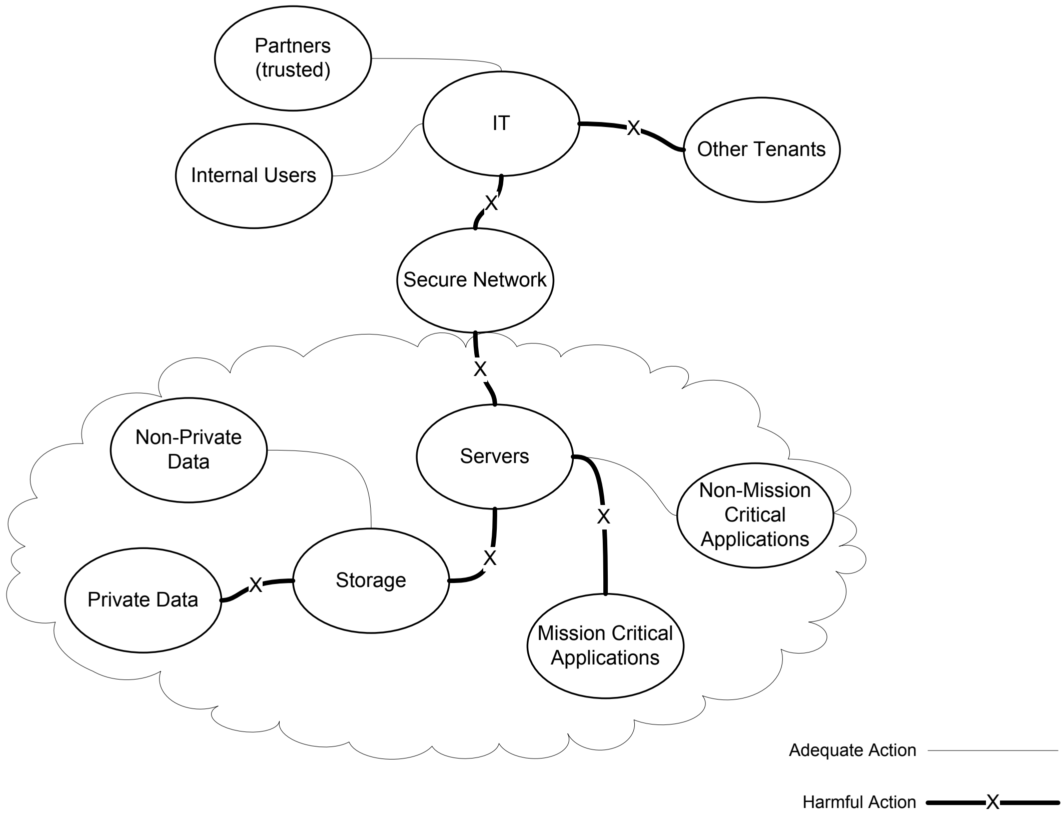
## Figure 6. Ideal Solution




## Figure 7. Problem Formulation



## Figure 8. Private Cloud Model




## Figure 9. The Private Cloud Conflict

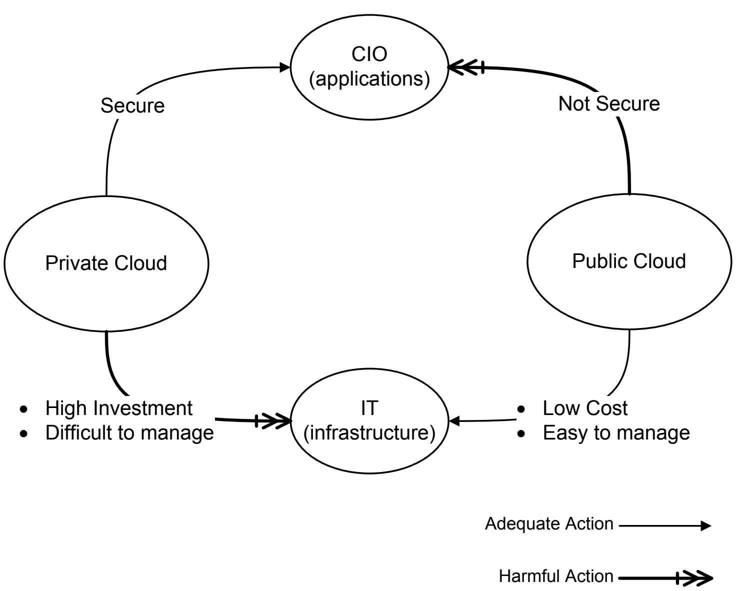


**Figure 10. Sufield Model**

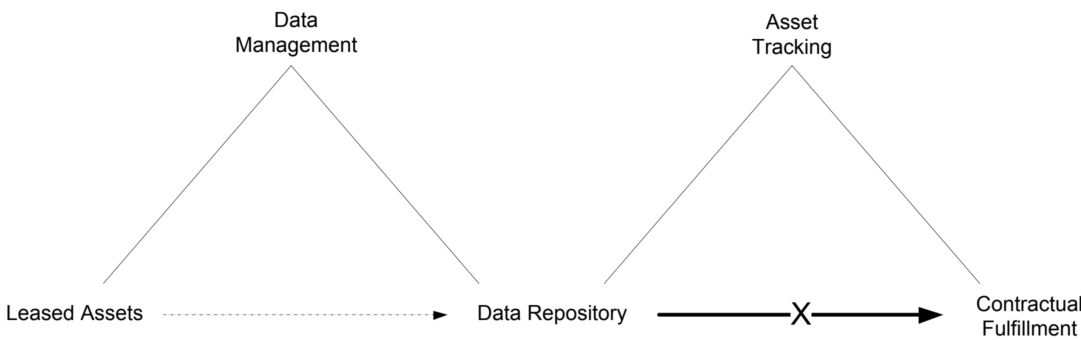



**Table 1. IT Technology Evolution Vector**

| **Phase:** | **Rigid** | **Modular** | **Programmable** | **Autonomous** |
|---|---|---|---|---|
| **Maturity:** | Initial state | Novelty | Readily available | Blends into the background |
| **Development:** | One-of-a-kind | Low production volumes | High production volumes | High production, |
| **Support:** | None, self supported | Minimal, highly specialized | Extensive | Transparent to the user |
| **Reliability:** | Very low | Moderate | High | Very high |
| **Cost per unit:** | Very high | High | Low, very affordable | The user is unaware of the cost |
| **User base:** | A few specialized users | Small | Very large | Very large |
| **Commercialization:** | Experimental, limited | Moderate | High | Embedded |
| ***Example:*** | *UNIVAC I* | *Early mainframes* | *Laptop computers* | *Vehicle engine control modules* |

## Table 2. Historical Trends

| Years | State | Phase |
|---|---|---|
| 1950s to early 1960s | Operated by highly specialized in-house experts or researchers; Early adopters; Users doubled as service providers; In-house; Complex; Slow; Expensive; Often Undependable; No clear demarcation between developer, user and service provider. | Rigid |
| 1960s to mid 1990s | Rigor; Standards; Process; Higher predictability; Lower outages; Custom one-off applications; Long-term fixed priced contracts; Co-existence of outsource and insource; Service Level Agreements (SLAs). EDS founded by H. Ross Perot in 1962. | Modular |
| 1990s to Present | Outsource; Insource; Offshore; CMM/SEI; Multiple vendors providing services; Standards and consortiums; Virtualization; Multi-Tenancy; Cloud; XaaS (SaaS, IaaS, PaaS). | Early Programmable |
| Future | Unknown | Late Programmable & Autonomous |

## Table 3. Application of TRIZ Laws to IT Services

| TRIZ Law | Application of Law |
|---|---|
| 1. Increasing degree of ideality | Cost reduction via labor reduction, less re-work, re-use, assembly, and automation. |
| 2. Non-uniform evolution of sub-systems | Incremental developments in hardware and software in support of services. |
| 3. Increasing dynamism (flexibility) | Interchangeable services, interchangeable hardware/software components, multi-purpose hardware (e.g., memrister). |
| 4. Transition to higher level systems | Auto-configuration, auto-provisioning, inter-cloud, multi-function hardware components. |
| 5. Transition to micro level | Small and targeted services, MEMS and nano-scale computing devices, low energy signatures, smaller but inter-operable software/service components. |
| 6. Completeness | The end-user won't be aware of the existence of IT services, requirements automatically generate applications. |
| 7. Shortening of energy flow path | Low-resource services, ambient energy sources, real-time integration, hardware/software/network merge into a single component. |
| 8. Increasing controllability | Self-managed services, self-diagnostics, self-healing, self-tuning, and self-throttling. |
| 9. Harmonization of rhythms | Predictive service demand, predictive provisioning, predictive configuration and architecture. |

## Table 4. BI Historical Trends

| Years | State | Phase |
|---|---|---|
| Pre-1990 | BI-0<br>• A manual process | Rigid |
| 1990s | BI-1<br>• Data warehouses – a repository of organizational data<br>• Reporting – usually after the fact<br>• OLAP - slicing and dicing of data | Modular |
| 2000s | BI-2<br>• Enterprise data warehouses – data repositories contain information at the enterprise level<br>• Unstructured data – accessing the facts buried in textual data<br>• Real-time reports – more actionable information than was available in BI-1<br>• Advanced analytics - correlations, forecasts, predictions, optimization<br>• Dashboards – pulling together the results in an easy to digest form | Programmable |
| Future | BI-3<br>• Unknown | Autonomous |

## Table 5. Application of TRIZ Laws to BI

| TRIZ Law | Application of Law |
|---|---|
| 1. Increasing degree of ideality | The future BI system will offer a broader range of analytical features, easier to use by non-technical knowledge workers, costless, and will have a broad range of business applications. |
| 2. Non-uniform evolution of sub-systems | Different elements of BI solutions (e.g., data management, analysis, visualization, and application interfaces) will experience independent development by companies specializing the respective fields. There will be continued start-up and acquisition cycles in the BI market. |
| 3. Increasing dynamism | The future BI systems will grow from point solutions to multi-discipline solutions, and to multi-industry solutions, spanning extended enterprises. |
| 4. Transition to higher level systems | The future Bi systems will evolve from single applications, and begin to work in pairs, then in groups with other BI applications for more complex analysis. |
| 5. Transition to micro level | BI solutions' footprints will shrink in size, requiring less computing resources, and accessible via mobile devices. |
| 6. Completeness | The future BI systems will comprehend mashup data, include a full suite of analytical techniques, integration with all enterprise and personal applications, and driven by the users' context for decisions and actions. |
| 7. Shortening of energy flow path | BI systems will be personalized – they will automatically locate the relevant data, eliminate data latency, analyze it in real-time, and serve the needs of the user via automated actions. |
| 8. Increasing controllability | The users' context will control the operations of the BI system and the actions it will take. |
| 9. Harmonization of rhythms | BI systems will be event-driven, based on a coordination among data availability, temporal and location factors, the characteristics and the needs the user. BI systems will collaborate with each other. |

### Kas Kasravi

Kas Kasravi is an HP Fellow and a Patent Strategist at Hewlett-Packard. He has been with HP and EDS since 1985, where his efforts have been focused on developing and leveraging advanced technologies. His areas of interest include artificial intelligence, data mining, computational linguistics, predictive analysis, and simulation. He has developed innovative solutions for clients in diverse industries, including engineering, manufacturing, pharmaceutical, financial, securities, supply chain, insurance, defense, and government.

Kas has been the recipient of two General Motors trade secrets, eight patents, and has a dozen patents pending. He has published a number of papers in various topics including artificial intelligence, business intelligence, data analysis, and innovation methodologies. Kas has BS and MS degrees in engineering and a law degree with emphasis on intellectual property and patents. He is a Certified Manufacturing Engineer.

In addition to his work at HP, Kas has been involved with academia for many years, and he has taught engineering classes at the University of Detroit-Mercy, Highland Park Community College, and Central Michigan University.

Mr. Kasravi can be reached at kas.kasravi@hp.com