Building
Tomorrow's Enterprise

White Paper

Infosys®

Structured Fallout Management for Broadband Applications

Sivakumar Thiyagarajan and Ravi Kumar Kusuma

Broadband Internet services – both DSL and fiber – have grown rapidly along with consumer demand. As Communication Service Providers (CSPs) grapple with the complexities of rolling out services and bundles to new locations and geographies, they face three key challenges: 1) Increasing network flexibility, 2) Enabling rapid service development and 3) Achieving operational excellence.

To remain competitive, CSPs need to differentiate themselves through operational excellence. An efficient order management process offers competitive advantage and minimizes the order-to-cash interval. This process enables CSPs to compete effectively with carriers offering cable and satellite broadband services.

The order management system is a complex business process engine that coordinates the entire delivery process including ordering customer CPE, configuring and enabling network capabilities, establishing access and security configurations for new customers, and enabling billing processes. Multiple systems are involved in the DSL and fiber order management process with multiple interfaces in cases where problems can arise. Effective order fallout management is critical to handle orders falling out at and around these interfaces.

For more information, Contact askus@infosys.com

Oct 2007

Need for Structured Fallout Management

Having partnered with several telecom service providers on their projects, Infosys believes that there is scope for improving fallout management in OSS/BSS applications. In one global CSP, more than 40% fallouts were observed immediately after the OSS/BSS applications went live.

Typically, Operations Support and IT teams react by deploying fallout reduction methods on a trial and error basis. This unstructured fallout management strategy leads to revenue loss, increase in operations cost and customer dissatisfaction.

Infosys recommends a structured approach for handling fallouts across different system lifecycle stages. A structured approach avoids fallouts through better planning and helps fix fallouts by using analytical tools. This white paper addresses various issues of fallout management within the broadband service fulfillment process and presents innovative strategies to address these issues in both new and established broadband environments. These analytical strategies:

- Deliver results even if the earlier phases did not use structured fallout strategies
- Can be used without any dependencies on each other
- Are self-contained and modular

Order Fallout Management

OSS/BSS applications (henceforth referred as 'applications') have requests and responses flowing to and from different applications. The basic protocol followed by each application for handling requests is – Acknowledge, Accept, Error and Completion. Each application assigns a unique error code and error description for each error scenario. If a request, and hence an order, goes through an error state, it may either need a retry or manual intervention. In the ideal order management process, requests and responses happen with no errors and with minimum elapsed time between request and response. However, due to various reasons, a percentage of these requests fall out and hence remain unprocessed, needing a non-standard manual intervention.

Fallout management straddles different lifecycle phases:

- System Development
- System Maturity
- Steady State Operation

As any one approach may not suffice to identify all causes of fallouts, different fallout management approaches must be deployed in different lifecycle phases.

System Development

Invariably, fallout prevention design and analysis is not given much thought as Business and Systems Analysts focus on translating business requirements into system requirements. Multiple interfaces between applications and two-way messages across applications demand a thorough analysis to identify potential causes of fallouts.

Rigorous system development with an eye on fallout reduction helps achieve better operational results by significantly reducing explicit and implicit fallout expenses and customer complaints. Infosys recommends structured fallout management and analysis be given higher priority and adequate resources be marshaled in this phase. The payoff for the effort invested in this phase on structured fallout prevention and management is substantial.

In the structured approach, a small upfront investment is required for fallout prevention design and analysis. Systems designed without focusing on fallouts during the design phase require considerable effort for data collection and subsequent analysis of data to get to the root causes of fallouts. Consequently, a heavy price is paid in subsequent phases in the unstructured approach. A comparison of the two approaches in this phase is covered in the section on 'Economics of Structured Fallout Management'.

Analytical approaches A, B, C, D and E are ideal for the System Development phase.

A. Sequence Analysis

Out-of-sequence messages from different applications are frequently the root cause for fallouts as these messages violate business rules and clog order flow. They can be avoided by analyzing the processing sequence of orders of different applications and right-sequencing the processing steps. This makes the order flow smooth with reduced wait time. Parallelizing multiple processing steps reduces wait period and increases order flow velocity.

As out-of-sequence messages cannot be completely avoided, an event sequencing framework can be designed upfront to handle them. Out-of-sequence messages are parked under different statuses and are auto-retried at an appropriate time by the event sequencing framework. This framework eliminates manual intervention and avoids fallouts occurring due to messages being out of sequence.

B. Transaction Management Analysis

Occasionally, fallouts can be due to hardware (like database server) or software (message queue, for instance) failure. The frequency of these occurrences may be few, but the fallout volumes in these situations are typically high. When a failure occurs, the transaction gets disrupted and there is no way for the transaction to continue from the earlier state when the system is restarted.

In this scenario, manual intervention to reflow high volume of fallouts is not an effective approach from time, cost and SLA perspectives. A well designed and implemented transaction management framework is effective in handling these situations.

An exhaustive list of potential hardware and software failure scenarios is drawn and a transaction management framework is designed to handle fallouts from these issues. In the event of a hardware or software failure, the entire transaction is rolled back and the message is stored to be retried later. This framework ensures that no orders are missed for processing in the event of a failure. The transaction management framework makes an application more robust without depending on any upstream system or manual intervention to reflow the affected orders.

C. Robot Analysis

Certain fallout types are handled by the Operations team using the routine approach of correcting data and reflowing orders. Any routine process is an ideal candidate for automation to reduce manual intervention and squeeze elapsed time.

During the system development phase, repetitive reflow opportunities are identified and “robot” scripts are designed to automate order reflow. Robot automation scripts are a sequence of simple commands that are written by the IT team and executed by either the Operations or IT teams to correct data of known scenarios and reflow orders. Manual intervention can be completely avoided for these fallout types by implementing the robots.

D. Integrated Fallout Queues

Usually, fallout queues are tied to one system and have limited capabilities in fixing fallouts end-to-end. This results in silos as an order may have been pushed from one system, but may get stuck in another system down the line. If an order is not cleared from fallout in any of the systems, it will result in missing customer deadlines and a prolonged waiting period for the customer.

During the system development phase, an integrated fallout queue is designed, enabling loose integration with all the order management, provisioning and billing systems. Loose coupling of the queue ensures there is no direct impact from changes to these systems and makes them robust.

Orders that have missed due dates or are in jeopardy are assigned to the integrated fallout queue and worked by the Operations team. The integrated nature of this queue makes it feasible to correct fallout in any flow through the system. This reduces the time required to correct the fallout and hence reduces customer waiting period. An automated customer notification system can notify the customer (via IVR or e-mail) about the progress and revised timelines.

E. Alert Mechanism

It is possible for system issues such as hardware crash and out-of-limit capacity to cause orders to fall out in large numbers. In the absence of an alert mechanism, the Operations teams get to act only after considerable damage is done and any opportunity to reduce further fallouts is lost. With constant monitoring of such instances and a mechanism to alert the Operations team, fallouts can be limited and in-flight orders segregated.

Designing an effective alert mechanism requires identifying the queues where orders fall out, defining threshold limits for alerts and mapping the Operations team to be notified. An automated monitoring system constantly monitors the queues for the defined thresholds and alerts the owners (through pager or e-mail) with metrics and a brief description of the issue.

System Maturity

After the system is implemented within the service provider environment, a period of "maturing" is required to identify and address product operation issues. During this phase, the IT and Operations teams almost always have to grapple with high fallouts. It is caused by an unstructured fallout prevention effort in the system development phase, and at times, the absence of fallout prevention effort. Project teams try to remedy the situation with more unstructured efforts and knee-jerk fixes. As the teams wrestle with the issue, customer complaints pile up and loss of revenue attracts executive attention.

In the structured approach, fallouts are fewer due to fallout prevention investments made in the system development phase. In the unstructured approach, the cost of handling fallouts is higher and increases with time, while in the structured approach fallout cost is significantly less and gets lesser with time. This is the direct result of the small investment made in the previous phase and due to the approaches to handle fallouts effectively in this phase. A comparison of the two approaches in this phase is shown in the section on 'Economics of Structured Fallout Management'.

Analytical approaches F and G are suited for the System Maturity phase.

F. Pulse Analysis Model

Orders may travel slower in some applications compared to others. To improve the overall velocity of order flow, it is necessary to study the time taken by orders to flow through each application at different time periods – peak hours, lean hours and batch runs.

Orders flowing through applications analyzed are similar to the analysis of a pulse traveling in a water stream. Orders entering the system in a brief time span are tracked individually through various applications. Analyzing elapsed time for orders through each application reveals bottlenecks in each application.

The pulse analysis model is applied to modules within an application to drill down further and narrow down on the root causes for slow order flow.

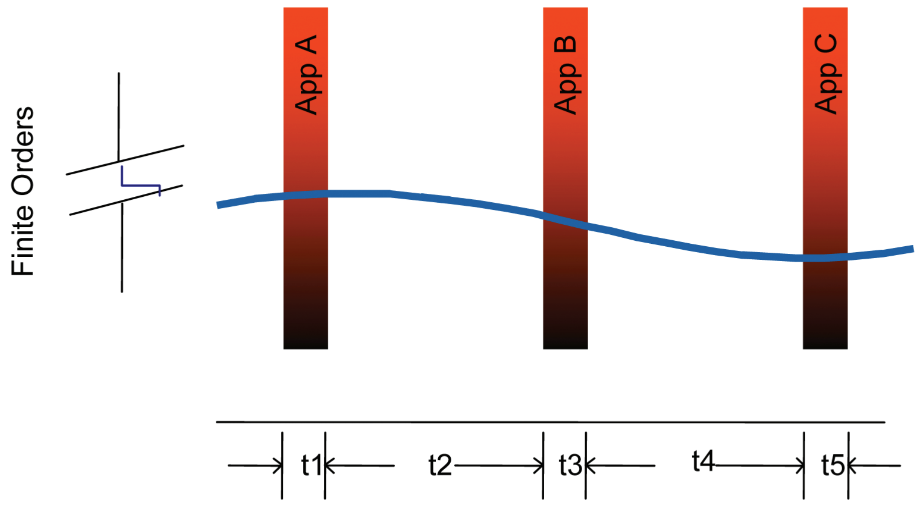

Figure 1: Pulse Stream Fallout Analysis

G. Out-Box Analysis

Orders falling out in an application are categorized under different buckets. The categorization throws light on probable root causes (changes to a product, changes to customer profile, etc.) that the Operations team can focus on. Multiple views can be applied to the categorization to identify the type of impact and take preventive steps to avoid additional fallouts until the issue is resolved. E.g. a view can be defined to know which product type for a given state is more impacted due to a fallout type.

- Maximum Fallout by System Count
- Maximum Fallout by Error Count
- Maximum Fallout by Order Type
- Maximum Fallout by State/ LATA
- Maximum Fallout by Functionality
- Maximum Fallout by Product Type
- Maximum Fallout by Customer profile

While more durable fixes are being worked on for the root causes, several approaches can be implemented for re-processing fallouts in the interim period:

- Scrubs to trap and clean in-flight orders
- Automated tools (developed upfront) for reflowing affected orders
- Work-rounds and alternatives for bypassing problem process flow

Steady State Operation

The transition of a system to "steady-state operations" is less about the stability of the platform and more about the manner in which the system is supported by the associated support and development organizations. In general, the operations organizations have developed "fallout business processes" to allow delivery of services within defined SLAs. But it is not uncommon to be plagued with high fallout and manual intervention rates.

The last mile analysis needed to eliminate these fallouts is important to rein in costs. There are no easy quick fixes and ad hoc handling of fallouts makes no impact in this phase. The strategies for dealing with fallouts in this state are different and more complex to introduce. It is critical that continued effort is made to identify the causes of fallouts and also to identify the means to either eliminate or implement automated fallout recovery scenarios to reduce the business and IT operations impact.

In the unstructured approach, the cost of handling fallouts stabilizes in this phase but remains high, while in the structured approach fallout cost is low and reduces further with the application of these strategies. A comparison of the two approaches in this phase is covered in the section on 'Economics of Structured Fallout Management'.

Analytical approaches H and I are appropriate for the Steady State Operation phase and help pinpoint areas that slow down order flow.

H. Modified Waterfall Model

Orders flowing through applications are analyzed using a modified waterfall model with feedback loop. Orders entering the system in a predefined daily window (these orders together form an 'order packet') are tracked through different applications. Aged order packets are tracked for a certain number of days through different systems. Analyzing aged order packets in different systems reveals issues in them.

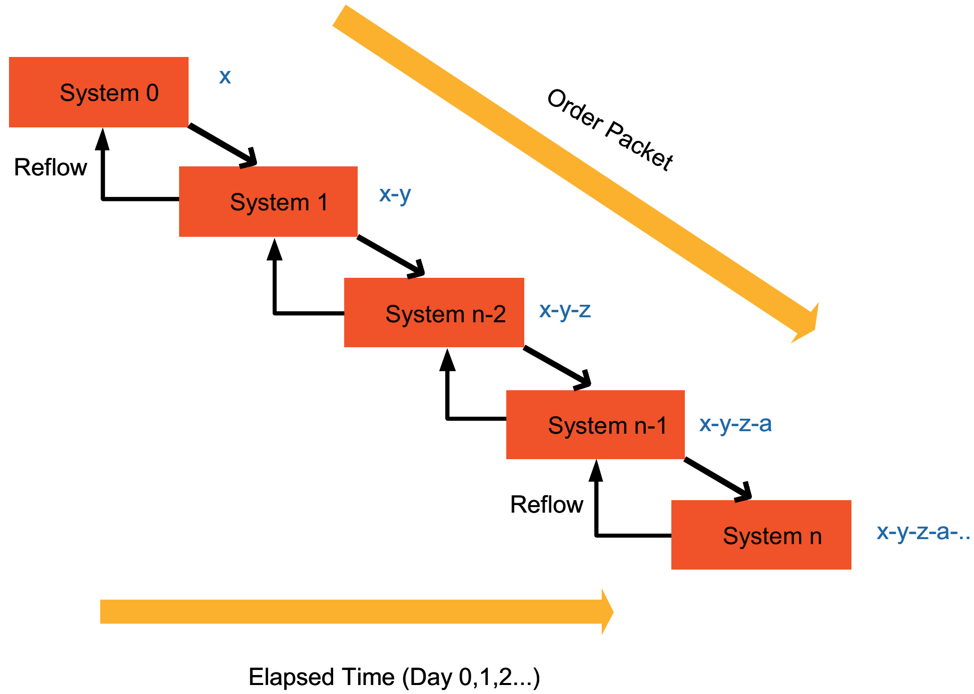

Figure 2: Waterfall Fallout Analysis

I. Leaky Pipe Model

To get an end-to-end view of major system "leakages" of orders, order flow data must be captured at regular intervals for orders of a predefined time window. This serves as the baseline for measuring the effectiveness of fixes done to applications over a period of time.

Orders flowing through applications are considered analogous to water flowing through a leaky pipe. Orders entering and leaving each application are tracked everyday. The difference between the count of orders entering the first application and the count of orders leaving the last application in a day is the order leakage in the system, similar to water leakage in a pipe.

Applications with large leaks are taken up for detailed analysis and drilled down to identify specific root causes.

Economics Of Structured Fallout Management

Based on data from executed projects, a comparison of operational expenses between structured fallout handling using an unstructured approach and the tools and models discussed above are presented here.

In the unstructured approach, fallout handling is not a planned activity. IT and Operations teams are in a reactive mode and usually act after fallouts happen. A comparison of expenses is depicted in Figure 3.

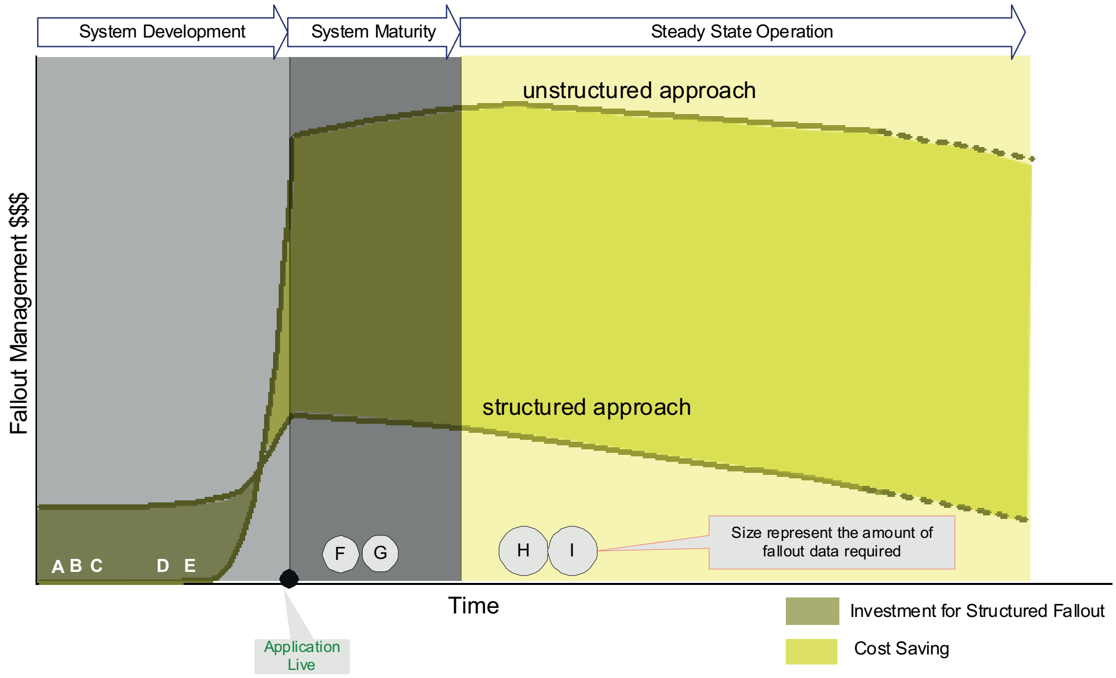

Figure 3. Comparison of Structured and Unstructured Fallout Management Approaches

Fallout management costs for structured and unstructured approaches are depicted for different phases of the application. Suitability of the fallout management approaches and the cost savings in each of these phases is discussed in the figure 3 above.

A structured approach compares favorably throughout the lifecycle of the system because:

- Robust design reduces fallout counts right from the day the application goes live
- Well planned analytical approaches in the tool kit reduce cycle time
- Alert mechanisms help take quick action before the fallout situation gets out of hand
- Models to measure efficacy of the fallout management techniques help course correction

Conclusion

Compared to an unstructured approach, the structured fallout management approach delivers better results, leading to satisfied customers. Infosys recommends that CSPs follow a structured fallout management strategy by deploying appropriate design and analysis tools for different lifecycle phases. These analytical approaches are self-contained, can be executed in parallel without dependencies and any one approach or set of approaches can be implemented on a stand alone basis.

About the Authors

Sivakumar Thiyagarajan is a Group Project Manager with the Communication Service Providers (CSP) Business Unit at Infosys. He has 11 years of IT experience in delivering IT services to global communication, retail and transportation corporations. Sivakumar currently manages a portfolio of broadband applications for a leading US telco.

Ravi Kumar Kusuma is a Project Manager at Infosys. He has over eight years of IT experience, including five years in telecommunications. He has managed several IT projects in the broadband space for leading telecommunications operators. His experience in the broadband domain covers systems development and operations management of various applications including Order Management and Order Workflow processing. Ravi's extensive experience offers him insights into the fallout handling aspect of operations. He has played a significant role in the launch of various products and initiatives in broadband.

Infosys® | Building Tomorrow's Enterprise

For more information, contact askus@infosys.com

About Infosys

Many of the world's most successful organizations rely on Infosys to deliver measurable business value. Infosys provides business consulting, technology, engineering and outsourcing services to help clients in over 30 countries build tomorrow's enterprise.

For more information about Infosys (NASDAQ:INFY), visit www.infosys.com.

© 2011 Infosys Limited, Bangalore, India. Infosys believes the information in this publication is accurate as of its publication date; suchinformation is subject to change without notice. Infosys acknowledges the proprietary rights of the trademarks and product names of other companies mentioned in this document.