Effort Instrumentation and Management in Service Delivery Environments

Melissa Buco, Daniela Rosu, Dikran Meliksetian, Frederick Wu, Nikos Anerousis,
IBM TJ Watson Research Center
Hawthorne, NY 10532 USA

Abstract— **Driving productivity transformations in services organizations must be a data-driven exercise. We develop a methodology that exploits effort-data analysis in order to drive productivity optimization initiatives with solid business cases and action plans. A services factory model is applied to a service provider organization to monitor where and how time is spent by the service staff. The results are used to identify and rank productivity optimizing initiatives. We illustrate how our methodology is applied to a real IT service delivery environment.**

Keywords-work effort; Servics Factory; Service Delivery Management;

I. Introduction

In many industries in the services sector, such as insurance, banking, transportation, or IT support, the diversity, quality, and efficiency of the offered services are continuously improving due to the advances in the underlying IT capabilities, such as the automated processing and storage of records, self-service over web and phone, and electronic collaboration. As a result, the productivity of the service delivery process depends on how efficient the IT capabilities are exploited. Service providers are interested to know where the inefficiencies lie and which improvement opportunities are worth pursuing. However, it is often difficult to identify improvement opportunities and assess expected solution benefits, especially for organizations with processes that might have been optimized in the past.

In this paper we propose a methodology for analyzing the inefficiencies of service delivery organizations and assess the business impact of potential improvement plans. The cornerstone of our approach is the methodical breakdown of how service practitioners spend their time (for example performing useful value-added work vs. attending non-productive meetings). This knowledge helps generate an opportunity map that guides productivity improvement actions. Therefore, the critical questions we address are: a) how to instrument a services organization, b) how to measurement activity effort, c) how to recognize inefficiencies, and d) how to assess the business impact of improvement plans.

Our approach is to apply the Time Volume Capture (TVC) technique to a generalized model of a service delivery organization that we refer to as "Services Factory". The services factory concept applies to organizations that rely on a careful orchestration of large number of human performers to deliver a set of services. Sample industries include insurance (claim processing), banking (back-office functions such as loan origination), and IT support (ticket-based resolution of incidents). TVC allows us to understand what activities are being performed, and with what effort. Then, a methodical analysis of the data identifies the improvement opportunities.

Compared to previous work on effort analysis for service delivery organizations, our contribution is the methodology that integrates effort instrumentation and analysis into the process of solving a significant business problem such as automation. This leverages on our approach to integrate effort instrumentation in the services factory model with its detailed activity taxonomy which greatly improves the *accuracy of the effort analysis.* In the following, we describe the models and methodologies we propose for effort instrumentation and analysis.

II. The Services Factory Model

A "services factory" is an environment where humans work collectively to handle an incoming flow of service requests. The term is an analogy to a manufacturing factory where work is performed on assembly line by workers with well-defined roles, governed by rules, processes, and methods for measurement of quality and productivity metrics. The services factory is organized around the following components:

- **service catalog** – the description of the types of service requests that can be handled by the factory;
- **work input system** – the entry point for service requests, performing activity decomposition and routing;
- **practitioner teams** – the group of people where service requests are eventually routed to be worked on;
- **governance system** – the monitoring and control of service delivery processes.

The service catalog is comprised of the *customer-facing (or external) catalog*, which describes the services as experienced by clients, and the *internal catalog,* which describes the activities performed in order to delivery the customer-facing services. For accurate effort instrumentation, the internal catalog must satisfy two requirements: 1) *Completeness* - all types of activities involved in service delivery must be referenced, including transformational (e.g., server reboot) and overhead work (e.g. organizational meeting, customer communication); 2) *Granularity* – activities must match the way work is managed, i.e. dispatched to be performed by a single practitioner of appropriate skill within a few hours.

978-3-901882-48-7 ©2012 IFIP

The work input system, automated or manual, uses the internal catalog to identify the next activity or workflow to be performed for a request and dispatches it based on service agreements, as well as performer skills and availability.

The organization of practitioner teams determines how work is managed in the service factory, and thus, it determines the composition of the internal catalog. Teams may be organized by customer-facing criteria, such as one team handle all of the requests from a customer, or by service-component criteria, where one team handles a group of activity types. An effective service factory model allows us to represent both team models.

Finally, the governance system manages the factory instrumentation and the calculation of metrics, and comprises processes for managing service-delivery parameters such as quality, cost, and capacity. Effort measurement and productivity improvement are managed as well.

III. Measurement And Istrumentation of effort

In this paper, effort is the time expended to fulfill a request. We propose a methodology for effort instrumentation based on the internal service catalog. This allows us to model effort at both internal and customer-facing levels. By aligning with the model of dispatching work to practitioners, effort instrumentation is simplified, requiring time capture only at the beginning and end of a work assignment. The overhead of data collection incurred by practitioners is moderately low, independent of the procedural steps comprised in the action.

The measurement of effort in services factory must represent the specific nature of service delivery processes. First, work is often performed in non-contiguous blocks of time – activities may be interrupted due to process requirements (e.g. waiting for an approval or waiting for an installation to complete), or due to the collaborative nature of work (e.g., practitioners get involved in phone calls, meetings, higher priority requests, etc.) Second, effort depends on the specific request parameters, practitioner skill, and service context (e.g. the complexity of the customer's environment). And, third, effort depends on the (IT) tools used to fulfill the request (e.g., automated procedure vs. manual). Additionally, the execution of service activity often involves multi-tasking (i.e., performing two or more activities during a time period) and repetitive execution of low-effort activities.

Overall, the tools for instrumentation of effort must capture sufficient details to allow us to assess the actual practitioner's effort rather than elapsed time, to gather request and tools details that can explain the extent of effort, and provide effective ways to contain the overhead of data collection.

Consistent data collection across practitioners is required for good data quality and usefulness. Actions such as targeted education and periodic data-validation are necessary for ensuring consistency during a data collection exercise.

A. TvC – An instantiation of Effort Measurement

The Time Volume Capture (TVC) is the tool we use to implement our methodology for effort instrumentation and measurement in a large IT Service Delivery environment. TVC integrates several methods for effort measurement[1], such as the pre-determined motion time systems, activity sampling, and analytical estimation.

TVC is comprised of a client tool and a backend subsystem. The client tool is an integrated activity-monitoring agent installed on the practitioner's computer, which is regularly synchronizing with the backend. The client functions as a stopwatch that users keep open on their desktops, and use it to mark the start and finish of each of the actions they perform.

The TVC interaction model addresses all of the identified requirements, such as activity interruptions, multi-tasking, and repetitive actions. In order to accommodate for interruptions, the tool allows the practitioner to pause, and resume an activity. Eventually, an activity effort is described as a collection of contiguous work sessions.

When multi-tasking occurs, an adjusted activity effort is computed (offline) by distributing the multi-tasking intervals across overlapping activities. One solution is to split these intervals in equal shares. For instance, if for an hour, work is recorded for two activities, such as attending meeting and creating a change request, the adjusted effort for each activity is half hour. Although this may not be accurate, it avoids having to ask the user about the actual effort split, which most often would be an approximation as well.

For repetitive activity batches, such as "Handling Alerts" that arrive in a quasi-continuous stream due to an IT system event, the user can record a single TVC activity and specify a volume representing the count of actual activities of same type that he performed. This reduces the overhead of data collection relative to actual work and prevents delaying the action that must be performed under tight SLA-constraints.

In addition, TVC integrates features that facilitate the collection in special circumstances. For instance, backdating an activity action (start, pause, resume, or complete) facilitates collection when practitioners who may not having access to their desktops when activity must be recorded (e.g., performing work on a IT delivery center floor).

The client catalog is derived from the internal catalog of the service factory filtered for the role of the practitioner. The TVC data model enables us to collect activity details that describe the context of execution, such as customer, request id, complexity level, urgency, tools used. However, during collection exercises, practitioners might omit to provide some features in order to keep overhead low.

A typical TVC data collection for a practitioner team requires three weeks of good, consistent collection. While, in general, practitioners world-wide are highly sensitive to the added overhead of data collection, our experience in a global IT Service Delivery organization reveals some interesting cultural differences. In India and Brazil, we experienced very little resistance to capturing the data and a good deal of interest in leveraging the data beyond the scope of the automation effort. In Europe, the concern is about data anonymity and purpose of data collection. In the US, the primary concern is the added overhead; users request integration of TVC with existing service request management tools in order to minimize entry of duplicate data and likelihood of conflicting details.

IV. Effort Data Analysis

Effort data is a necessary ingredient for addressing many business requirements in a services factory-like organization, such as SLA compliance, and low-operational costs. Effort data is used for computing performance metrics, and to inform transformational efforts, such as (1) productivity improvement through automation of labor-intensive tasks; (2) assignment of resources across performer teams; and (3) work shift management and work assignment within performer teams.

A typical process for driving transformation involves three major stages in which effort data is a necessary input, namely (1) business case development; (2) solution deployment; and (3) business impact measurement. In the following, we present the methodology improving productivity through automation in a service factory for IT Service Delivery.

A. Methodology for Automation Analysis

While practitioners expect that automation solutions can improve productivity, it is often hard to pinpoint which activity will yield higher benefits. This is particularly difficult in large organizations, encompassing many teams and different levels of prior automation or process details. In the stage of business case development, analysts are using the effort data to identify the automation targets with highest expected impact. Since automation has both initial and ongoing costs, the time savings should be translated into monetary savings over time, and combined with the estimated automation costs for each activity. The target activities for automation are those expected to yield the most cost savings within the business case time horizon.

For solution deployment to the large-scale service factory, effort data analysis helps identify the teams that would benefit most from earlier deployment. By combining savings with availability of pre-requisite technology, change freeze periods, and other business constraints, the analysts can build schedules for staged deployment that allow the business to capitalize early from its investment in automation solution development.

For business impact measurement, effort data collected after deployment is used to quantify the actual effort reduction by comparison with previously collected data. This approach mitigates the limitations of typical business productivity metrics, such as "service hours/server", which often cannot accurately reflect the impact of the new productivity solutions due to confounding factors, such as major outages, or numerous software fix-pack releases.

Expanding on the identification of automation candidates, we submit that a systematic methodology is required to tackle the complexity, scale, and technological diversity of the delivery infrastructure while driving towards standard automation solutions. We take the following approach.

- First, effort data is collected for a set of representative teams. Managed technology, skill level, activity scope, activity volume are good criteria to consider. The collected effort records are expanded with context details based on other service request management sources, such as classification of incidents by cause, and type of solution.
- Second, effort data (with adjustment for multi-tasking) is analyzed to identify activity groups with similar technology and operations context that contribute to a relevant share of effort. Examples include patch management, health checking, and system and application alert management.
- Third, for each activity group we identify an automation solution framework that is best suited for its technology, operations and strategic context. This can be a tailored management application or the installation of off-the-shelf management products, a novel strategic tool for system management, or homegrown automation solution already deployed in some organizations.
- Fourth, for an activity group and its selected automation framework, we identify the specific processes, sub-activities, tasks that can be automated, and design related solutions. For instance, for the "health checking" group, one identifies the activities of periodic health check execution, periodic reporting of results, and remediation. Similarly, for the "alert-triggered incidents" group one identifies the activities of alert configuration and alert resolution for CPU, file system, connectivity, and other. For each solution, expert practitioners estimate the expected work effort. This is used to assess the overall expected work effort savings. A subsequent work effort collection with a limited scope may be warranted to acquire a more detailed activity context to help determine the use-cases to be handled by automation solutions.
- Fifth, we extrapolate the results of the activity group analysis to the entire scope of the automation effort. This involves extending the effort estimation to a time interval larger than the effort data collection, and to teams that did not perform effort data collection.

At the end of this process, we have identified a set of automation solutions that cover a relevantly high share of effort in the target organizations. For each solution we quantify the expected savings as monetary values using the share of a daily Full Time Employee (FTE), i.e., the average number of hours spent daily for an activity across all person-days in the scope divided by the number of hours in a daily FTE.

Extrapolation to longer time intervals and other teams is based on mapping the activity groups to the taxonomy of work-input systems, such as incident and change management tools. The mapping allows to collects stats about long-term request arrival rates. Combining long-term arrival rates with effort statistics allows us to estimate current effort and expected savings. For teams without effort data, SME input is required to quantify any differences in per-activity effort. Similarly, SME input is required to provide long-term request rates when work-input statistics are not available.

B. Instantiation of Effort Analysis for Automation Planning

We applied the above methodology to automation in Distributed System Management in a global IT Service Delivery provider. While we cannot share actual business details, we share insights on how the methodology was applied.

Effort data collection was applied to four teams with representative scope and low level of process automation. The teams made up only ~3% of the practitioner population in scope. Extrapolation methods described above are used to assess the overall extent of current effort and expected savings.

We augment the details related to operational context and refine the granularity of activity taxonomy based on the cause that triggered the actions, when applicable. For instance, for an incident of system not available, the cause could be network failure, OS malfunction, or application overload. This is because the actual action/process executed (and the effort expended) to solve the incident depends on the cause. This choice of offline augmentation of the catalog activity taxonomy helps us improve the accuracy of effort assessment without adding burden on practitioners at data collection time.

The result of effort analysis is a list of fine-grained activities in descending order of overall work effort. Figure 1 shows the breakdown of the observed/extrapolated effort by activity type, with anonymized activity names. The activity with largest effort share corresponds to "File system capacity and growth". It was estimated that 70% of the effort for this activity type could be saved through automation. The estimation of the implementation cost was complicated by fact that the automation of this activity type would be only a part of an overall automation framework, and thus only the incremental cost for this particular activity was relevant to the calculation.

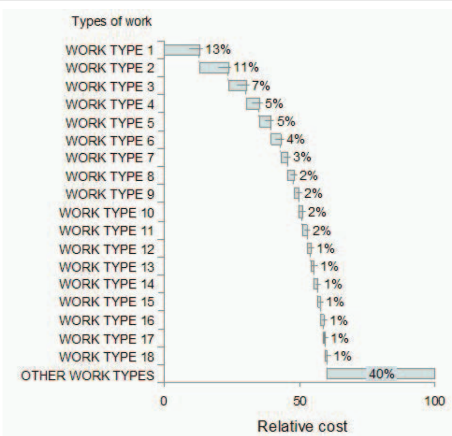

Figure 1 Activities in Decreasing Order of Effort

For other activity types on the list, the savings could be more significant. For instance, the handling of alerts triggered by high CPU utilization could be reduce almost completely with automation for collection and analysis of system vitals.

V. Related Work and Concluding Remarks

In previous work, the concept of "Service Factory" relates to methodology and tools for creation/deployment of new services [5], or to the concept of a factory that produces goods and delivers services related to these goods [4]. In our work, the concept of "Services Factory" is distinct from those above, relating to service operations management [6].

Our work is relevant for implementation of Lean [7], a production practice that targets the elimination of resource expenditures that do not create value for the end customer. Namely, the methodology for mapping how work effort is spent in service delivery processes addresses core elements of Lean implementation. Time Volume Capture tool and methodology for identification of automation opportunities demonstrate that a multi-level analysis approach is required for complex work procedures as occurring in IT Service Delivery. The SME input on current procedures and automation solutions resemble to phases of Value Stream Mapping [8].

In our work, we recognized the impact that time studies have on productivity and SLA compliance. Prolonged data collection or large-population coverage are not feasible. Different than existing products for Time-Motion studies, we mitigate these conflicts by exploiting service delivery-specific sources of information that details on the measured actions that allow us to extrapolate to larger time intervals or practitioner population.

In conclusion, this paper presents a methodology for effort measurement and driving productivity transformations in services organizations and demonstrates its applicability to a real IT service delivery environment. The methodology is based on the Service Factory model and provides answers to questions on how to instrument for effort-data collection and exploitation to solve real-business problems

One of the issues that we plan to address in the future is the non-invasive capture of effort data. Currently, the overhead incurred by the TvC tool is significant, about a 1 minute per action. We are investigating methods that will make the process completely transparent for the user, namely the continuous acquisition of user-generated data (from the keyboard, mouse, window content etc.) and the automatic interpretation of the corresponding activities (using rules and other heuristics). This will enable a continuous and seamless capture of activities.

References

[1] http://www.natprodcentre.com/eDemo/material/msc/CG_Work_Measurement_HTML/pag_02.htm

[2] Y.Diao, A. Heching, D.Northcutt, G.Stark, "Modeling a Complex Global Service Delivery System", Proc of 2011 Winter Simulation Conference, 2011

[3] M. Buco, C Ward, R. Chang, J. Wolf, L. Luan, P. S. Yu, "Utility Computing SLA Management Based upon Business Objectives", IBM Systems Journal, Volume 43, No 1, 2004

[4] R. Chase, D. Garvin, "The Service Factory", Harvard Business Review , July 1989

[5] Services Science Factory "http://www.servicesciencefactory.com/

[6] J.A., Fitzsimmons, R.S. Sullivan, "Service Operations Management", McGraw-Hill, New York 1982

[7] Lean Enterprise Institute,, "Principles of Lean" http://www.lean.org/WhatsLean/Principles.cfm

[8] M. Rother, J. Shook, "Learning to See value-stream mapping to create value and eliminate muda, ISBN 0966784308