Behavioral Analysis of Service Delivery Models

Gargi B. Dasgupta, Renuka Sindhgatta, and Shivali Agarwal

IBM Research India
{gdasgupt,renuka.sr,shivaaga}@in.ibm.com

Abstract. Enterprises and IT service providers are increasingly challenged with the goal of improving quality of service while reducing cost of delivery. Effective distribution of complex customer workloads among delivery teams served by diverse personnel under strict service agreements is a serious management challenge. Challenges become more pronounced when organizations adopt ad-hoc measures to reduce operational costs and mandate unscientific transformations. This paper simulates different delivery models in face of complex customer workload, stringent service contracts, and evolving skills, with the goal of scientifically deriving design principles of delivery organizations. Results show while Collaborative models are beneficial for highest priority work, Integrated models works best for volume-intensive work, through up-skilling the population with additional skills. In repetitive work environments where expertise can be gained, these training costs are compensated with higher throughput. This return-on-investment is highest when people have at most two skills. Decoupled models work well for simple workloads and relaxed service contracts.

1 Introduction

Service-based economies and business models have gained significant importance over the years. The clients and service providers exchange value through service interactions with the goal of achieving their desired outcomes. Given the focus on the individual customer's value and uniqueness of the customer's needs, the service providers need to meet a large variety of expectations set by the customers. This is the primary reason for the service delivery to be labor-intensive where human intervention and interaction is unavoidable.

Service providers aim to maintain the quality of service by structuring their service delivery (SD) operations as service systems (SS). A SS is an organization of resources and processes that support and drive the service interactions so that the outcomes meet customer expectations [22][19]. The size, complexity, and uniqueness of the technology installations require specialists at provider's end to support customer needs. In addition, customers require multiple business functions, applications and technologies to be supported. Hence their workload tends to be complex and dynamic. The specialized service workers (SW) or human resources of a SS are teamed together in order to serve the service requests (SR) or work of the customer.

We motivate the study by presenting the IT Service Management SS where a service provider maintains complex systems and infrastructure of the customer as described in detail by authors in [21]. When system interruption or degradation occurs

S. Basu et al. (Eds.): ICSOC 2013, LNCS 8274, pp. 652–666, 2013.

i.e., a server goes down or a network link is broken, the customers request for service to be restored in the form of tickets or service requests (SR). In a typical IT infrastructure set up, there are several dependencies between the supporting systems. Hence, a ticket resolution often requires multiple systems to be analyzed and rectified. Fig 1. depicts the dependencies between such systems. A single SR stating "Unresponsive and slow Web server" would require service workers to check the web server, the database server and the storage space to identify root cause and resolve the SR.

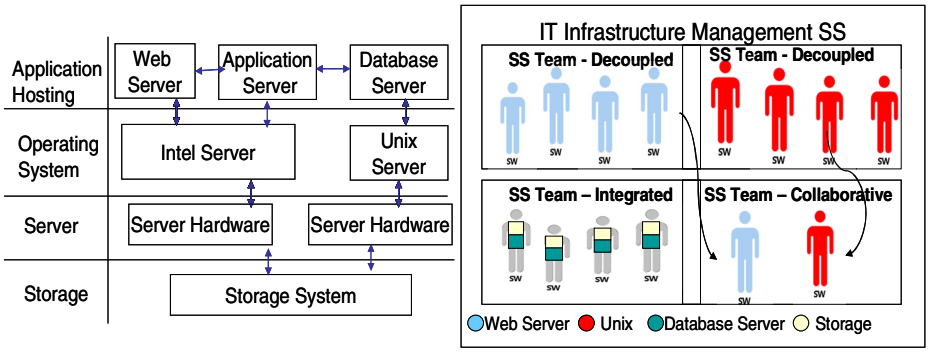


Fig. 1. Sample of IT Service Management Dependencies and Service Delivery Models

The SS team in a delivery organization can be arranged based on their skills and competencies. We use SS team and team interchangeably in the paper. Each team has skills in the domain of specialization. The customer work then gets delivered out of one or more SS teams. We classify customer work as complex when it requires support of more than one skill domain or technology for its resolution. As complex customer SR arrives at the delivery organization, it requires the different teams supporting it to work towards its resolution. Depending on how the SS teams have been organized, the following structures are imposed on the SR resolution workflows.

- Decoupled Workflow: When multiple teams work independently on a complex customer SR, with each team only responsible for partial resolution of the issue, it imposes a Decoupled structure on the SR resolution flow. No single team has ownership of the SR and the work is completed sequentially by teams working on parts of it. This often results in complex work taking longer to resolve as it traverses multiple teams. The structure is prevalent when complex SR is handled by different teams as shown in Figure 1.
- Collaborative Workflow: When the complex SR is handled by experts from multiple teams, working on the SR simultaneously, it imposes a Collaborative structure on the SR resolution flow. In this case effort of multiple people is locked in parallel but the quality of work improves. As indicated in Figure 1, experts from teams work together to complete the SR.
- Integrated Workflow: In cases where a team is composed of multiple skill specializations, the SR may be handled by multiple skills within the same team. Here the SS team owns the SR and one or more multi-skilled people work towards its resolution. This imposes an Integrated structure on the resolution flow. While customer satisfaction is highest in this model due to tightly synchronized

workflow, the cost to the provider is higher from perspective of supporting multiple skills. Figure 1 depicts an integrated workflow where the team has both database server skill and storage system skill required for resolving an SR.

The above workflows form the basic building blocks of any complex delivery environment and define a Service Delivery Model (SDM) followed by the Service System to meet the customer expectations. The choice of a particular SDM influences SLA performance, costs, work completion times and learning. The teams, depending on how they handle work, in turn cater to a particular SDM. Henceforth we focus our analysis on the three SDMs. Since each has its pros and cons, a static one-time decision that is universally applied to all customers may not suffice. Especially with services business revenue being close to a billion USD for major providers, its success is strongly related to the trust and satisfaction of its existing customers. In the face of customers' unique expectations it is imperative to understand and weigh the design choices at hand. This necessitates a superior decision process regarding which customer workload, service contracts and skill distributions effectively map best to which SDM in terms of efficient, timely and cost effective delivery for the provider.

Current literature in services delivery [8, 6] focus on optimizing staffing for simple customer work following the Decoupled model, where work arrivals across technology teams have very little or no correlation. When the work is complex, authors [21] focus on improving SLA performance for higher priority work using the Collaborative model, but the throughput of the high volume work noticeably suffers.

Contributions: In this paper, we aim to analyze different SDMs from the perspective of performing complex work and focus on multiple performance parameters of SLA, throughput and utilization. Learning is modeled in workers as they perform repeated activities. The cost versus performance tradeoff for training on additional skills is analyzed to understand the optimal number of skills workers should be trained on. Different rework scenarios are studied that can lead to quality degradation. We define the best SDM as one that: (a) has the best SLA performance, throughput and resource utilization across all priorities of work (b) has least amount of degradation in the performance parameters in the event of high rework (c) has the least cost of delivery. The goal of this work is to establish insights into the best SDM under specific workload, SLA and learning environments and discern the improvements (if-any) that can be achieved by adopting a hybrid model. To the best of our knowledge this is the first work that addresses the above perspectives of service delivery design to this detail, and offers key insights.

This paper is organized as follows: Section 2 describes the different aspects of complex work and how they are affected by the SDMs. Section 3 introduces our simulation model and the various parameters of interest. Section 4 presents the experimental analysis and section 5 presents a review of the related work.

2 Complex Work in Service Systems

We now cover the background on the generalized service operations and present different aspects of delivery models for complex work resolution in service systems. Depending on the teaming principles in place, a customer's work could be supported

by one or multiple teams following different delivery models. An SS is typically characterized by:

- A finite set of customers, denoted by C, supported by the service system.
- A finite set of shifts, denoted by A, across which the W service workers (SW) are distributed.
- A finite set of skill domains, denoted by D, with L levels in each skill.
- A finite set of priority levels, denoted by the set P.
- A finite set of service requests (SR) raised by the customer that arrives as work into the SS

We next discuss the work arrivals, SLAs of the SR, and service times and skills of workers in context of supporting complex customer work.

2.1 Work Arrivals

According to existing body of literature in the area of Service Delivery systems [86, 8], work arrives into a SS at a finite set of time intervals, denoted by T, where during each interval the arrivals stay stationary. Arrival rates are specified by the mapping $\alpha : C \times T \to \Re$, assuming that each of the SR arrival processes from the various customers C_i are independent and Poisson distributed with $\alpha(C_i, T_j)$ specifying the rate parameter. When there is a correlation between the work arrivals across different teams supporting a customer, it denotes a complex SR from the customer that requires attention from multiple skill domains. In this case, the independence property still holds for the first team where work is performed.

2.2 Service Level Agreements

SLA constraints, given by the mapping $\gamma : C \times P \to (r_1, r_2), r_i \in \Re, i = 1,2$ is a map from each customer-priority pair to a pair of real numbers representing the SR resolution time deadline (time) and the percentage of all the SRs that must be resolved within this deadline in a month. For example, $\gamma(Customer_1, P_1) = \langle 4,95 \rangle$, denotes that 95% of all SRs from customer$_1$ with priority P_1 in a month must be resolved within 4 hours. Note that the SLAs are on the entire SR itself, which means for complex work the targets apply to resolution across multiple SS teams.

2.3 Skill

In a multi-skill environment, given D domains of skills, let the vector $Sr = (s_r(d_0), s_r(d_1), ..., s_r(d_i))$ denote the required skill levels required for a SR r, $d_i \in D, 0 \le i \le |D|$, where $s_r : D \to [0,1], r \in SR$ denotes the required skill function that returns the level of skill required in each of the domains to complete the service request r. Similarly, the possessed skill defined for each worker w is given by $Sw = (s_w(d_0), s_w(d_1), ..., s_w(d_i))$, where $s_w : D \to [0,1], w \in W$, returns a real number between 0 and 1 representing the level of skill that agent w possesses,

relative to each domain element. Further, 0 denotes no skill and 1 denotes perfect skill. Assuming at least two levels of expertise, $L: D \rightarrow N >= 2$ returns the number of discrete levels defined for each domain (minimum two levels). Work assignment via dispatching looks at the vectors Sr and Sw while deciding the best match between work and resources. Fig. 2 shows two existing skills in the domain $=\{D_1, D_2\}$, each with two levels of skill = {High, Low}.

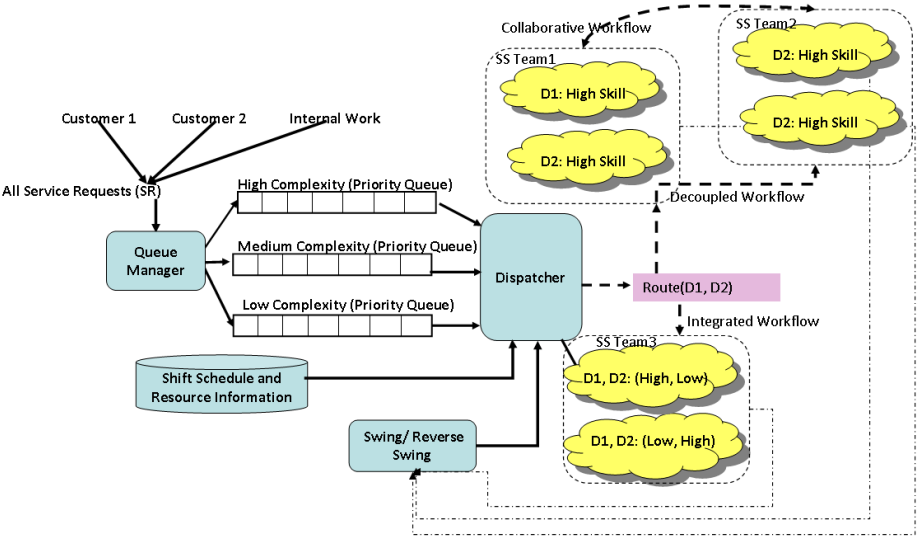


Fig. 2. An operational model of service systems (SS)

2.4 Cost

The cost of delivery is directly related to the cost of the resources working in the SS. Let C_{DC-l} be the base cost of the resource in Decoupled model with single skill expertise at level *l*. The base cost is assumed to be higher for higher skilled people (i.e., $C_{DC-l1} > C_{DC-l2}, \forall l1 > l2$). In contrast, the Integrated model has multi-skilled people who would need to be trained on each additional skill. Let l_H be the highest skill level of a resource in the Integrated model. We assume that the base cost of a multi-skilled resource is dominated by the base cost of her highest expertise. (S)he also has *N* additional skills, out of which n_i skills are at level l_i. Let δ_{l_i} the cost for training for each skill to level l_i. Assuming a linear cost model of skills, the cost incurred by the Integrated model for training a multi-skilled resource is then given by:

$$C_{INT} = C_{DC-l_H} + \sum_i n_i * \delta_{li}, where \sum_i n_i = N \quad (1)$$

Since $C_{DC-l} > \delta_l$, i.e., the base cost is higher than the training cost, at lower values of *N*, it makes sense to train the same resource on an additional skill rather than hire a new resource. For higher *N*, it becomes more beneficial to hire a new SW.

2.5 Service Time

The time taken by a SW to complete an SR is stochastic and follows a lognormal distribution for a single skill, where the parameters of the distribution are learned by conducting time and motion exercises described in [6]. Service time distributions are characterized by the mapping $\tau : P \times D \rightarrow \langle \mu_1, \sigma_1 \rangle$, where μ_1 and σ_1 are the mean and standard deviation parameters of the lognormal distribution and represent the time a worker usually takes to do this work. The distribution varies by the priority of a SR as well as the minimum skill-level required to service it. For complex work requiring multiple skills $(D_1, \ldots D_i)$ the total service time is an additive component of the individual work completions and follows a shifted lognormal distribution [16].

Since complex work takes more time to complete, for the sake of maintaining throughput, it becomes imperative to assign some work to people skilled below the minimum skill-level. When lower skilled people (s_w) do higher skilled work (s_r), where $s_r > s_w$, the service times become longer. This increase in service time is obtained from an adaptation of the LFCM algorithm (Narayanan et al. 2012), where the service time $\mu_n(s_w, s_r)$ to finish the n^{th} repetition of work requiring skill s_r by worker with skill level s_w is given by:

$$\mu_n(s_w, s_r) = \mu_1 n^{-\beta\left(1 - \frac{\log\left(1+\gamma/t_n\right)}{\log n}\right)} \tag{2}$$

where μ_1 is the mean service time to execute the higher skilled work for the *first* time, β is the learning factor, γ is the skill gap between levels s_w and s_r, t_n is the time spent by worker at level s_r. Higher the gap γ, and lower the time spent t_n, higher is μ_n. μ_1 represents the longest time to do this type of work, but with work repetitions, expertise is gained and μ_n decreases [13]. In practice we bound the minimum value of μ_n at $\mu_{\min}$, which is the lowest service time work s_r can take. The parameters $\langle \mu_1, \beta, \gamma, \mu_{\min} \rangle$ are learned by conducting time and motion studies [16] in real SS to measure the exclusive time spent by a SW on a SR. As given by Eqn. (2), slower learning rates and bigger gaps in the skill required of a SR and skill possessed by a SW, both contribute to longer service times.

2.6 Dispatching

The Dispatcher is responsible for diagnosis of the faulty component(s) as well as work assignment to a suitable worker. For fault diagnosis the dispatcher intercepts the complex SR to determine the most likely faulty component(s) and maps them to skill domains that must be consulted (in sequence) for resolution. The SR then traverses through the diagnosed list of teams. In Fig. 2, a SR dispatched with the tag

Route $\{D_1, D_2\}$, needs to traverse through teams that support D_1 and D_2. When multiple domains of customers are supported, solving the fault-diagnosis without ambiguity is non-trivial [24]. Dispatching errors are commonly termed as *misroutes* and may result in wasted time and cause customer dissatisfaction. With more number of supported components, the risk of misrouting is higher. This is exacerbated in the Decoupled model. The Integrated model avoids this to some extent with multi-skilled resources being able to handle complex issues within the team.

During work assignment, SWs are to SRs of the matching skill-level requirements. When matching skills are not available higher or lower skilled SW may be utilized for servicing a SR. This is referred to as *swing* and *reverse swing* respectively. Fig. 2 shows the Dispatcher routes complex customer work either through the Decoupled, Integrated or Collaborative models and decides to turn on swing/reverse-swing policies based on feedback information from the system.

3 Simulation Based Evaluation

There are many challenges in real-life Service delivery operations that make analytical modeling of a SS a cumbersome exercise [6]. These include the aggregate SLAs specified by customers, the inter-dependence of work queues, the variation of service time distributions with the skill level of the worker, the random breaks taken by resources and the complex preemption rules on the ground. Hence, we resort to simulation as a tool to model the operational characteristics of SS and estimate the performance of systems pertaining to the three SDM. There have been other comparisons of analytical and simulation-based models that corroborate our choice of simulation as a tool [11]. We propose a discrete-event simulation model for a SS according to its definitions in Section 2 and the parameters defined below. The model is similar to the one proposed in [6, 8] with the main that exception that both work and people can have multiple skills. All our experiments have been conducted with data from SS in server support area in the data-center management domain. The data is collected over multiple years using tools [6] defined for IT service management.

Simulation Parameters

The SS simulated extend the definitions in Section 2 with the following specializations.

- T contains one element for each hour of week. Hence, |T| = 168.
- P = {P1, P2, P3, P4}, where, P1 > P2 > P3 > P4.
- We assume $|D| = 3, L = 3$. The three different levels of expertise simulated are {Low, Medium, High}, where, High > Medium > Low. Each level of expertise has a least service time distribution $(\mu_{\min}, \sigma_{\min})$ associated with it (as in Table 1), which characterizes the minimum time this work type could take. The estimates are obtained from real life time and motion studies [6].

- *Swing:* Swing is invoked when Low queue length>10, where low skilled work is assigned to a high skilled resource. Service times remain same in this case.
- *Reverse Swing:* Reverse Swing is invoked when High and Medium queue length>10, where high skilled work is assigned to a low skilled resource. Service times become longer (Eqn. 2) in this case.
- *Preemption:* Preemption relation $\Rightarrow$ is the transitive closure of the tuples $P_1 \Rightarrow P_2$, $P_2 \Rightarrow P_3$, $P_2 \Rightarrow P_4$.
- *Transfer:* In case of work requiring multiple skills and a Decoupled work structure, the work gets handed over from one team to another. The teams could be geographically co-located (transfer time ~30min) or dispersed (> 30min). There are no transfer rates in Integrated since multiple skills can be found in single team. Collaborative has no transfer times.
- *Lead Time:* The time taken to synchronize the availability of multiple workers in Collaborative flow.
- *Rework:* A percentage of the work is re-opened due to bad quality fixes. Occurs when low skilled workers work on high skilled requests.
- *Dispatching:* Assigns work to resource with matching skill requirements if available. If not available, route to the worker with the lowest skill gap.
- *Learning Factor:* We assume a default moderate learning rate of $\beta = 0.1$ for each SW. Table 1 shows the mean and standard deviation (μ_1, σ_1) for the maximum service times taken by a resource at each skill level, when the work is executed the *first* time. These estimates are obtained from the time and motion studies reported in [6].
- *Misroutes:* Dispatching errors cause misroutes and are associated with wasted effort in addition to the transfer-times. Thus they map to longer completion times.
- *Utilization of SW:* This is related to the productive hours or busy time spent in work resolution.

The interplay of the above parameters and their combined effect is addressed in the experimental analysis.

4 Experimental Analysis

In this section, we describe our experimental evaluation based on data from four real-life SS in the server support area. The four skill domains supported include Operating Systems, Storage, Database and Web Middleware. The Decoupled, Collaborative and Integrated models of delivery are simulated with work arrivals, priority distributions and SLA target times as shown in Table 1(left). In Decoupled and Collaborative models, we create 4 teams, each supporting one skill domain. The resources in these models predominantly possess medium and higher skill levels, as shown in Table 1(right), while in Integrated model the reverse is true. Every SR is dispatched with multiple skill requirements. In the Decoupled model, complex work starts at the first faulty component and transferred sequentially from one team to another. In the Collaborative model people from all teams work in parallel to solve the issue. In the Integrated model, one team is created containing all the 4 skills. The skills are distributed

among the workers based on whether they have (2, 3 or 4) skills each. We assume a SW in an Integrated team has only one skill in the highest level, and rest at low or medium levels. Service times follow lognormal distribution for each skill and the means get lower with repetitions according to Eqn. 2. Table 1(right) shows the minimum and maximum service times at each level.

We employ the AnyLogic Professional Discrete Event simulation toolkit [4] for the experiments. Up to 40 weeks of runs were simulated with measurements taken at end of each week. No measurements were recorded during the warm up period of first four weeks. In steady state the parameters measured include:

- SLA measurements at each priority level
- Completion times of work (includes queue waiting times, transfer times, and service times)
- Throughput of the SS (work completed/week)
- Resource utilization (captures the busy-time of a resource)

For all the above parameters the observation means and confidence intervals are reported. Whenever confidence intervals are wider, the number of weeks in simulation is increased and reported values in the paper are within 95% confidence intervals. We seed the simulation with a good initial staffing solution from the Optimizer kit [15] which returned the optimal number of staff to handle the work.

Table 1. Experimental Parameters (Workload, Skills, Service Times)

Workload	Arrival SR/week	1575
Priority	% Distribute	Target Time (hours)
P1	10	4
P2	20	6
P3	40	12
P4	30	24

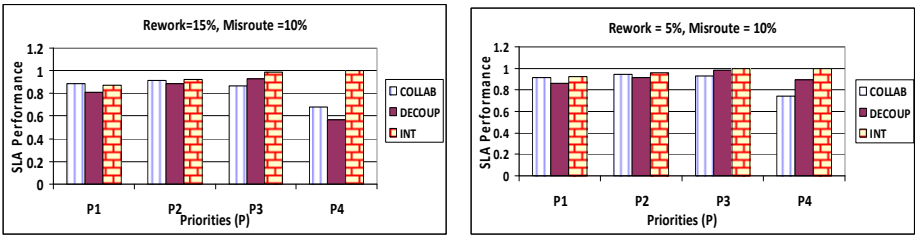
Number of resources = 38	SDM		Min, Max Service Times	
Distribution of Skill Level (L)	% Distribute (COLLAB, DECOUP)	% Distribute (INTEGRATED)	Min ST $<\mu_{min}, \sigma_{min}>$	Max ST $<\mu_1, \sigma_1>$
Low	40	70	30,20	60,20
Medium	40	20	15,15	40,15
High	20	10	10,10	30, 10

4.1 Complex Multi-skill Work

We investigate the scenario of work requiring multiple skills and correlated ticket arrivals across teams with experimental parameters given by Table1. First we handle complex work that requires 2 skills and resources are either single skilled (Decoupled, Collaborative) or have 2 skills (Integrated). The experiment assumes 10% misroutes and 5-10% rework in the environment. The results in Fig. 3 show that mean values for SLA performance with 95% confidence levels. The following observations stand-out: (a) Collaborative model works well in terms of SLA performance for higher priority work. This re-confirms the assumptions by authors [21]. The fact that experts simultaneously work on the multiple skills, and their service times are also the lowest reflects in the good performance of low volume, high priority work. However in case of high volume lower priority work, the Collaborative model does not do well.

As multiple people's effort is locked on higher priority work, lower priority work gets queued up and ultimately affects SLA performance. (b) Decoupled does better than Collaborative in case of P3 and P4 work. This is because Decoupled has the least synchronization overhead among multiple skills, which works well for high volume work. But this lack of tight integration in work resolution workflow, affects the P1, P2 SLA performance. (c) Integrated does the best across all severities, and clearly has the right balance between tight synchronization of critical work and decoupling of larger volume work. Interestingly, while people have lower skill levels in Integrated, and may initially take longer to service; overall SLA performance is not affected as long as there is some learning in the environment.

Fig. 4 presents performance of all the three models as the rework % increases. Only the P3 performance plot is presented as a representative case, but similar trends are observed at other priorities. The performance degradation in the Integrated model is the least, as rework increases. This shows that this model has the best appetite to absorb additional work in case of error situations, without affecting performance. In theory it can be argued that rework may be inherently higher in the Integrated model as people have lower level of expertise that may result in poor quality of work and higher rework % than the counterpart models. Fig. 4 shows that even if the rework in Integrated model is higher by as much as 10% than the other two models, the SLA performance of the highest volume P3 bucket is still better.



(a) Rework = 5%, Misroute=10% (b) Rework = 10%, Misroute=10%

Fig. 3. SLA Performance in different Service Delivery Models

In Fig. 5, we repeat the similar experimentation with misroutes, while keeping the rework at 10%. However as misroutes increase beyond 10%, the performance of all models degrade uniformly. Since misrouting is related to dispatching errors, we conclude that beyond 10% of misrouting in the environment, no SDM performs well and alternative methods for error diagnosis [24] are needed. Table2 presents the throughput and resource utilizations of the different models. At 5% rework and 10% misroute, the mean throughputs of Collaborative are the lowest, while Decoupled and Integrated are comparable. This is because in the Collaborative model multiple people's efforts are simultaneously blocked and the effort/SR is much higher.

Fig. 6 shows the drop in throughput for the different models as rework increases which shows that beyond 15%, the drop in Decoupled throughput is more pronounced. To understand the consistently lower throughput of Collaborative model, we look at how long the work took to complete in the different models. Recall that

completion time measure both the queue waiting times as well as the service times. Fig. 7 shows that at lower severities the completion times are comparable across all models, while higher severities see an exponential increase in the completion times for Collaborative. Completion times of Integrated are marginally higher than Decoupled at all priorities. This can be attributed to the higher service times for multi-skilled resources. But it does not translate to any obvious throughput disadvantages.

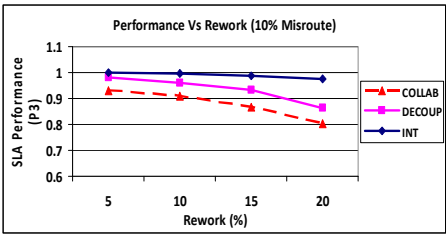


Fig. 4. Performance as Rework increases

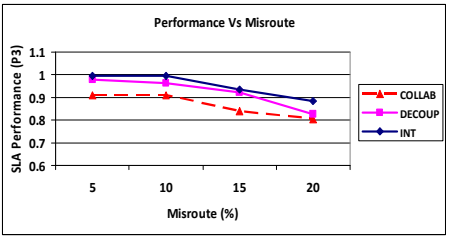


Fig. 5. Performance as Misroute increases

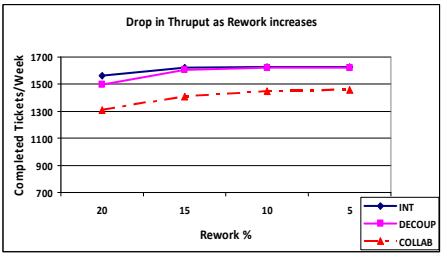


Fig. 6. Drop in Throughput as Rework increases Utilizations (Rework = 5%, Misroute=10%)

Table 2. Weekly Throughput and Resource

	Mean Throughput (SR Completed/ Week)	Resource Utilization
COLLABORATION	1457	62.8
DECOUPLED	1620	46.4
INTEGRATED	1624	53.0

Table 2 also shows higher resource utilizations for Collaborative for a lower net throughput that can be attributed to the longer completion times. Fig. 8 plots the change in utilization as rework in the SS increases. Decoupled has the best utilizations which is because of the lowest completion times. Integrated is slightly higher in terms of utilization at a comparable throughput. Overall, Fig. 8 shows that even across higher rework %, the utilizations of decoupled remains the best, with the Integrated model following closely. A hybrid model that is Collaborative for (P1, P2) work and Integrated for (P3, P4) achieves best of both and requires only partial upskilling of the population. Based on above results we summarize the following:

Observation 1: The Integrated model works well in terms of SLA performance, throughput and resource utilization across all reasonable rework scenarios. With some moderate learning in the environment, the higher service times in Integrated have a lower impact on SS performance than the transfers in Decoupled. If the higher priority work have tight SLAs and continues to be < 10% of the pool's work, having a hybrid model, can achieve best of both worlds: Collaborative for high priority complex work enables high SLA performance for critical issues. For high volume, low

priority work, an Integrated model that up-skills only 20% of the population with an additional low level skill can significantly improve performance and throughput of lower priorities.

4.2 Skills and Learning

Having established the Integrated model with the most uniform performance across the parameters of interest, we now experiment with some of the learning aspects of it.

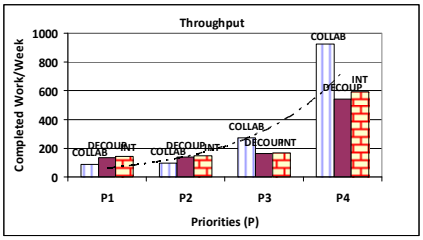


Fig. 7. Completion Time across Priorities

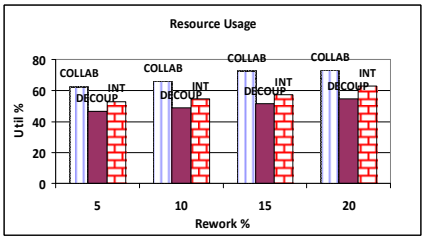


Fig. 8. Increase in Utilization with Rework

The biggest drawback of the Integrated scenario is the higher costs it entails, especially if the need to up-skill grows. Our next set of experiments investigates the benefits of up-skilling a resource beyond 2 skills. Table 3 presents the SS parameters as the work coming in becomes more complex (i.e. requires 3-4 skills and resources possess (2, 3 or 4) skills. Results are shown in Table3. Rows 1 and 3 show that for complex work, having SW with more skills does improve P1, P2 performance marginally. Also since work now takes longer to complete, we expect throughputs to drop uniformly in this scenario. Instead we notice that throughput drops are greater when skills per SW are more. This interesting observation can be explained by the fact that the people with more skills are now busier for longer, since the work resolution takes more time. With certain skills being more in-demand, it results in unique skills in the environment being tied up for too long, while incoming work in the queue waits for a suitable SW to become available. This is confirmed by the higher resource utilizations for lower throughput, when people have > 2 skills. Interestingly, the drop in throughput is lowest, when people have only 2 skills. It is therefore more beneficial to split highly complex work (requiring 3-4 skills) among multiple resources, than have one multi-skilled SW do all aspects of it. The cost implications of having multi-skilled SW are shown in the last column of Table3, computed as per Eqn. (1). A blended rate (across skill levels) of 80K₹ per SW per month and an up-skilling cost of 20K₹ per skill is assumed. We argue that since SLA performance at the higher priorities can be independently improved by having a Collaborative model for the critical work, the higher costs of multi-skilling a person beyond two skills has limited returns, especially since it comes with the risk of lowering throughput.

The sensitivity of the learning factor (β) on the performance of the Integrated model is shown in Table 4. Even with a small amount of learning in the system (0.07-0.1), the performance of the SDM is good. With close to no learning in the system (< 0.03), however the deterioration in service times is pronounced and this affects the SS parameters of throughput and utilization.

Observation 2: For complex work the return on investment for up-skilling is the maximum when resources have at most two skills. Up-skilling beyond that may result in little or no benefit in performance. An Integrated SDM works well in most cases, given that there is some amount of learning in the environment.

4.3 Workload and SLA variations

Our final set of results in Table 5 show that all previous observations hold for other workload variations and SLA ranges as well. We investigate both bursty traffic as well as flat arrivals, with work coming in only on weekdays as well as throughout the week. SLAs are varied in terms of stringency, by increasing the target times. Skill distributions are modified in the work as well as resources. The results are presented for the throughput and SLA performance parameter for a subset of the scenarios, but are seen to hold for the rest as well. When the arrivals are *bursty* Decoupled performance deteriorates. As the skill distributions change, the relative performance of the models remains same. When SLAs are relaxed, Decoupled works relatively well for P1 performance. However as seen from Table 5, Integrated continually performs better than its counterparts across variations in arrival patterns, skill distribution and SLA stringency.

Table 3. Complex Work requiring >=2 skills

(Skills Required, Skills Possessed)	SLA Perf (P1)	SLA Perf (P2)	Mean Thruput (SR/wk)	Util (%)	Cost ₹ / SW/Month
SR = 3, SW= 3	81%	89%	1173	70%	140K
SR = 3, SW= 2	77%	86%	1188	70%	120K
SR = 4, SW= 4	77%	86%	510	87%	160K
SR = 4, SW= 3	75%	81%	507	87%	140K
SR = 4, SW= 2	74%	80%	704	75%	120K

Table 4. Learning factor sensitivity

INTEGRATED	Mean Thruput (SR/week)	Completion Time (min)	Util (%)
With Learning Model (β=0.1)	1624	110.44	54.8%
With Learning Model(β=0.07)	1608	140.54	65.9%
With Learning Model(β=0.03)	1282	173.81	80.1%
With No Learning Model	1191	377.63	83.3%

Observation 3: The Integrated model consistently outperforms the others under a reasonable set of workload arrivals, SLA targets and skill distributions.

Table 5. Performance comparison, with workload, SLA and skill distribution variations

	Throughput (SR/week)				P1 SLA Attainment	
	Arrival (Flat)	Arrival (Bursty)	Skills Equal Dist	Skills Skewed to Higher Levels	SLA Stringent	SLA Relaxed
COLLABORATION	1418	1370	1360	1318	80	90.1
DECOUPLED	1635	1551	1400	1428	73.2	88.3
INTEGRATED	1638	1616	1505	1611	79.8	89.4

5 Related Work

The concept of shared service has existed for a long time, for e.g., multiple departments within an organization shared services like HR, finance, IT etc. However, its extension to shared delivery models for IT services has been gaining momentum from the last decade [5]. A recent study [23] of global service delivery centers revealed that shared services not only reduces costs, but also improves quality. A body of work exists on organizational design principles underlying an effective service delivery system [1] and resource hiring and training in such models [20]. However there is no work on generalizing the service delivery models and evaluating the pros and cons when presented with different kinds of workloads and work arrival patterns. This is the gap that this work addresses. Learning and forgetting curves in production and manufacturing industry [13] has received a lot of attention. The service delivery work, being repetitive in nature can benefit from these results in modeling the effect of learning and forgetting on service times. There is another line of work that studies the effects of task assignment on long term resource productivity. This is because the task assignment impacts mean learning rate, mean forgetting rate, mean prior expertise, variance of prior expertise etc and thus has a direct consequence on productivity. This paper incorporates some of the manufacturing domain results. The work in [18] presents a heuristic approach for assigning work by taking into account all these factors. How to staff, cross-train them and utilize multi-skill resources has also received adequate attention in the past in the context of call-centers [79]. The work in [12] advocates that a flexible worker should process a task s/he is uniquely qualified for before helping others in shared tasks. This is advocated in work-in-process constrained flow-lines staffed with partially cross-trained workers with hierarchical skill sets. Experimental results from our simulation are in agreement with many of the suggested best practices for multi-skilled resources. The effect of collaboration between teams has also been studied in work in [21] which proposes the concept of social compute unit. We have used this structure in the collaborative work flow model in this paper. The paper [10] theorizes how task/team familiarity interact with team coordination complexity to influence team performance.

6 Conclusion

We perform behavioral analysis of the different SDMs and present insights on their performance for changing workload patterns, SLAs, learning and skill distribution parameters. These insights have critical implications on optimized service delivery and can be used to transform service providers' work organization by helping determine which customer(s) work fits best into which SDM. In future we plan to create a platform where these insights are used for automated transformation.

References

1. Agarwal, S., Reddy, V.K., Sengupta, B., Bagheri, S., Ratakonda, K.: Organizing Shared Delivery Systems. In: Proc. of 2nd International Conference on Services in Emerging Markets, India (2011)

2. Agarwal, S., Sindhgatta, R., Sengupta, B.: SmartDispatch: enabling efficient ticket dispatch in an IT service environment. In: Proc. of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2012 (2012)
3. Alter, S.: Service System Fundamentals: Work System, Value Chain, and Life Cycle. IBM Systems Journal 47(1), 71–85 (2008)
4. Anylogic Tutorial 2008, How to build a combined agent based/system dynamics model in Anylogic. System Dynamics Conference (2008), http://www.xjtek.com/anylogic/articles/13/
5. Assembly Optimization: A Distinct Approach to Global Delivery, IBM White Paper (2010)
6. Banerjee, D., Dasgupta, G.B., Desai, N.: Simulation-based evaluation of dispatching policies in service systems. In: Winter Simulation Conference 2011 (2011)
7. Cezik, M.T., L'Ecuyer, P.: Staffing multi-skill call centers via linear programming and simulation. Management Science Journal (2006)
8. Diao, Y., Heching, A., Northcutt, D., Stark, G.: Modeling a complex global service delivery system. In: Winter Simulation Conference 2011 (2011)
9. Easton, F.F.: Staffing, Cross-training, and Scheduling with Cross-trained Workers in Extended-hour Service Operations. Manuscript, Robert H. Brethen Operations Management Institute (2011)
10. Espinosa, J.A., Slaughter, S.A., Kraut, R.E., Herbsleb, J.D.: Familiarity, Complexity, and Team Performance in Geographically Distributed Software Development. Organization Science 18(4), 613–630 (2007)
11. Franzese, L.A., Fioroni, M.M., de Freitas Filho, P.J., Botter, R.C.: Comparison of Call Center Models. In: Proc. of the Conference on Winter Simulation Conf. (2009)
12. Gel, E.S., Hopp, W.J., Van Oyen, M.P.: Hierarchical cross-training in work-in-process-constrained systems. IIE Transactions 39 (2007)
13. Jaber, M.Y., Bonney, M.: A comparative study of learning curves with forgetting. In: Applied Mathematical Modelling, vol. 21, pp. 523–531 (1997)
14. Kleiner, M.M., Nickelsburg, J., Pilarski, A.: Organizational and Individual Learning and Forgetting. Industrial and Labour Relations Review 65(1) (2011)
15. Laguna, M.: Optimization of complex systems with optquest. OptQuest for Crystal Ball User Manual, Decisioneering (1998)
16. Lo, C.F.: The Sum and Difference of Two Lognormal Random Variables. Journal of Applied Mathematics 2012, Article ID 838397, 13 pages (2012)
17. Narayanan, C.L., Dasgupta, G., Desai, N.: Learning to impart skills to service workers via challenging task assignments. IBM Technical Report. Under Review (2012)
18. Nembhard, D.A.: Heuristic approach for assigning workers to tasks based on individual learning rates. Int. Journal. Prod. Res. 39(9) (2001)
19. Ramaswamy, L., Banavar, G.: A Formal Model of Service Delivery. In: Proc. of the 2008 IEEE International Conference on Service Computing (2008)
20. Subramanian, D., An, L.: Optimal Resource Action Planning Analytics for Services Delivery Using Hiring, Contracting & Cross-Training of Various Skills. In: Proc. of IEEE SCC (2008)
21. Sengupta, B., Jain, A., Bhattacharya, K., Truong, H.-L., Dustdar, S.: Who do you call? Problem Resolution through Social Compute Units. In: Liu, C., Ludwig, H., Toumani, F., Yu, Q. (eds.) ICSOC 2012. LNCS, vol. 7636, pp. 48–62. Springer, Heidelberg (2012)
22. Spohrer, J., Maglio, P.P., Bailey, J., Gruhl, D.: Steps Toward a Science of Service Systems. IEEE Computer 40(1), 71–77 (2007)
23. Shared Services & Outsourcing Network (SSON) and The Hackett Group, "Global service center benchmark study" (2009)
24. Verma, A., Desai, N., Bhamidipaty, A., Jain, A.N., Barnes, S., Nallacherry, J., Roy, S.: Automated Optimal Dispatching of Service Requests. In: Proc. of the SRII (2011)