How Customer Perceived Latency Measures Success In Voice Self-Service

Jeff Fried and Rob Edmondson

Don't hang up. Please stay on the page, and the next available performance-enhancing *BCR* article will be with you shortly.

Jeff Fried is CTO of Empirix (www.empirix.com), specializing in testing and monitoring of next-generation networks, contact centers and Web-based applications. Jeff can be reached at 781/266-3430 or jfried@empirix.com. Rob Edmondson is senior field engineer with Empirix, and has years of experience working with customers on testing and monitoring of contact center applications. Rob can be reached at 916/781-9873 or redmondson@empirix.com

Voice self-service applications can help to lower operating costs, increase efficiency and provide better customer service, but only if they work reliably and consistently. Unfortunately, this isn't always the case. In fact, 79 percent of respondents in a recent Empirix survey said technology issues in their implementations regularly impact agent productivity and customer experience.

Of course, measurement and monitoring can help ensure that any application performs properly, and most enterprises are swimming in performance data—yet few are able to translate these numbers into how well their applications perform from a business point of view. This is particularly true for voice applications.

At the same time, voice self-service applications handle the majority of customer-initiated contacts in almost every organization, so their performance is critical. All too often, however, the importance of measuring these applications is overlooked, or the results are lost in a mass of other performance measurement data.

Customer perceived latency (CPL) is a simple measure that captures the most important behavior of voice self-service applications. CPL is the length of time between the end of a caller's input and his/her perceiving the successful response from the system, as shown in Figure 1. Because CPL measures performance at the application

FIGURE 1 CPL Measures Customer Perceived Latency

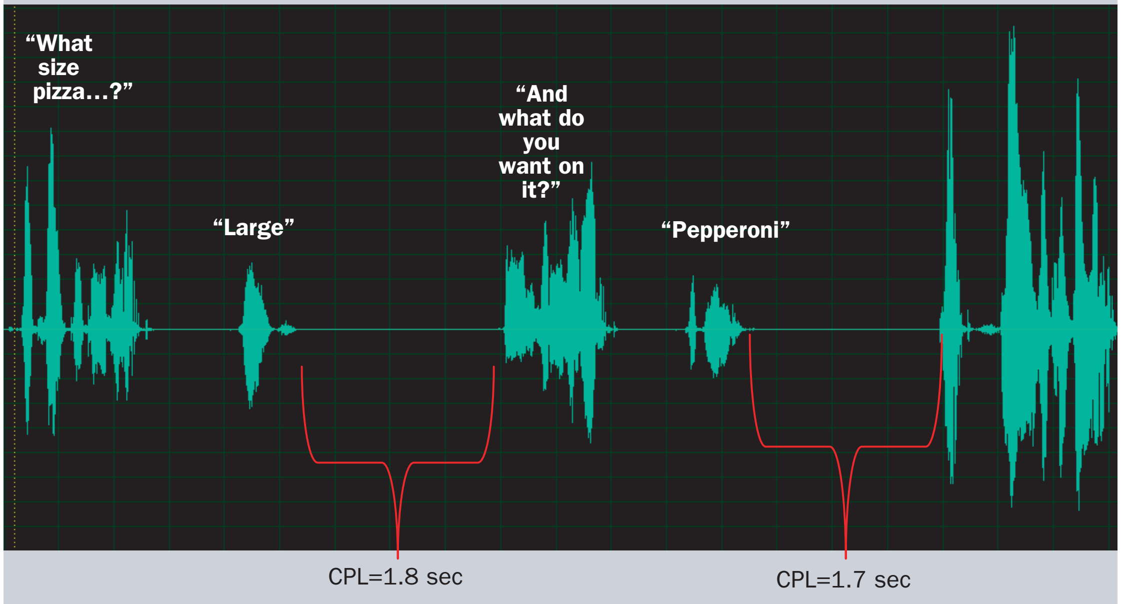
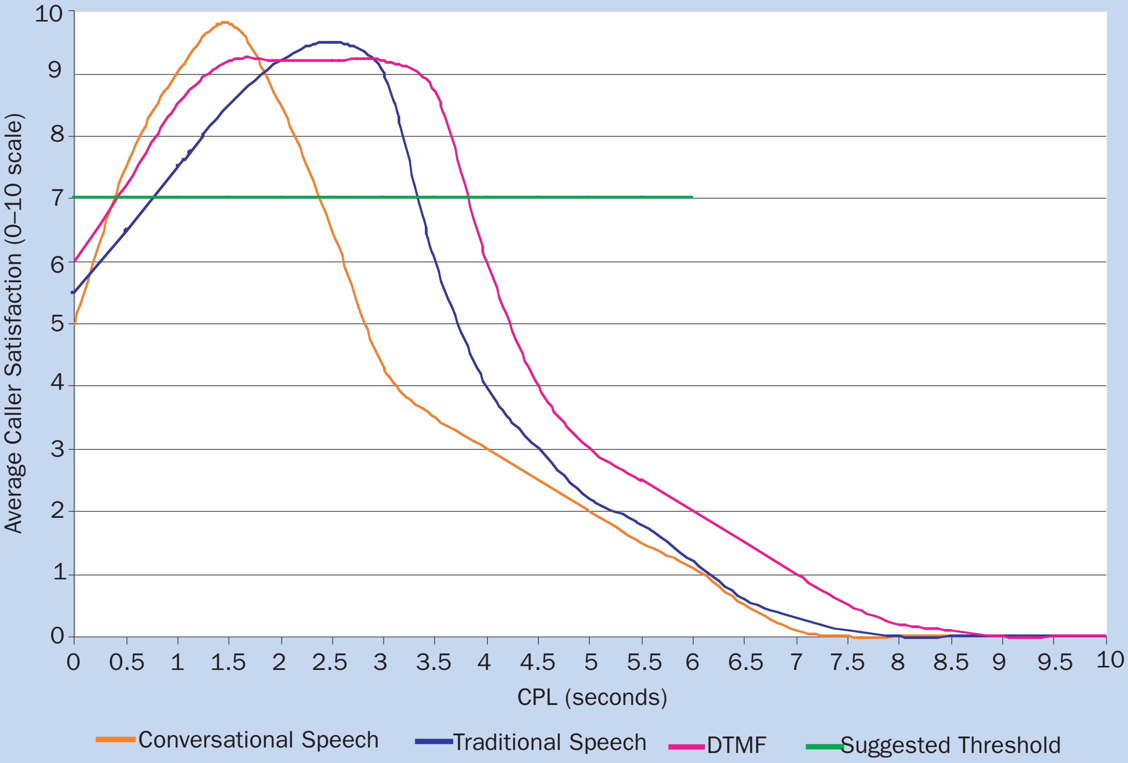


FIGURE 2 Caller Satisfaction vs. CPL



level, it includes not only the traditional transport-level round trip time (RTT), but also the time it takes for speech recognition, database access, etc.

Surprisingly, there appears to be no standard term for this simple concept. It has been called response time, user latency, dialog turn duration, and various other terms.

The direct relationship between response time and usability is common knowledge: Most people have first-hand experience with slow applications and the frustration they generate. The challenge comes in developing effective CPL targets, measuring CPL appropriately, and then using these measurements. Few organizations do these tasks effectively (or at all), but, with the right tools and knowledge, they can be put to straightforward and beneficial use.

Connecting CPL And Customer Satisfaction

Users of speech applications expect the timing of the application's responses to closely mimic that of human conversations. If they have to wait too long for a response, they can become impatient and may exit the system. In other cases, they may start to repeat inputs and cause the application to re-process or misrecognize responses. Either of these situations can kill the effectiveness of an otherwise well-designed application.

As shown in Figure 2, there is a strong connection between latency and caller satisfaction. These curves represent the subjective ratings of 153 users on a phone banking application, where a rating of 1 means "not satisfied" and 10 means "completely satisfied." Curves are shown for three different implementations of the same application: One requiring the user to make touchtone inputs (DTMF); one using traditional or structured speech ("say 'checking' or 'savings'"); and one using conversational speech ("how may I help you?"—"which account?").

The threshold satisfaction rating of 7 in Figure 2 is based on asking users whether they liked the application, then finding a satisfaction rating that divided most "yes" answers from most "no" answers. This is somewhat arbitrary, and the specific shape of these curves will vary with different applications and audiences, but determining a threshold is important and these responses are typical. A few observations:

- In general, faster is better; slow responses interfere with usability. Once latency exceeds a certain level, users are likely to either hang up or request a live agent.
- Responding too quickly can interrupt and irritate users, or cause them to pause in their response. This is indicated by the curves in Figure 2 rising from moderate satisfaction with a nearly immediate response to a peak with less than two or three seconds perceived latency. Immediate responses are unusual in practice due to network times (RTT), which are commonly in the 200–300 ms range, and application times for even the sim-

plest interactions, which are often 300–500 ms. Even so, very quick responses are far better than very slow responses.

- The more dialog-oriented the speech, the faster users expect the system to respond. Users will accept a wider range of wait times when using touchtone input systems (DTMF) than when using structured or conversational systems.
- Consistency is important. Although this is not reflected in Figure 2, applications that have a consistent CPL receive much higher satisfaction ratings than those that are erratic, even if the erratic application is sometimes faster.
- When a longer wait time is needed for the system to respond to the customer's request, it is important to advise the customer immediately. Issuing a spoken prompt like "one moment please," can reset a customer's expectations so that a longer wait time is less likely to be seen as unacceptable system latency.
- The best measurements for CPL are the type used in service level agreements, e.g., X percent of turns with CPL under Y seconds. Every turn matters, and a single call can include many turns, so a rating "per call" could be misleading.

What is a good "rule of thumb" benchmark for

FIGURE 3 "Before" Results: System Configured Using Standard Guidelines

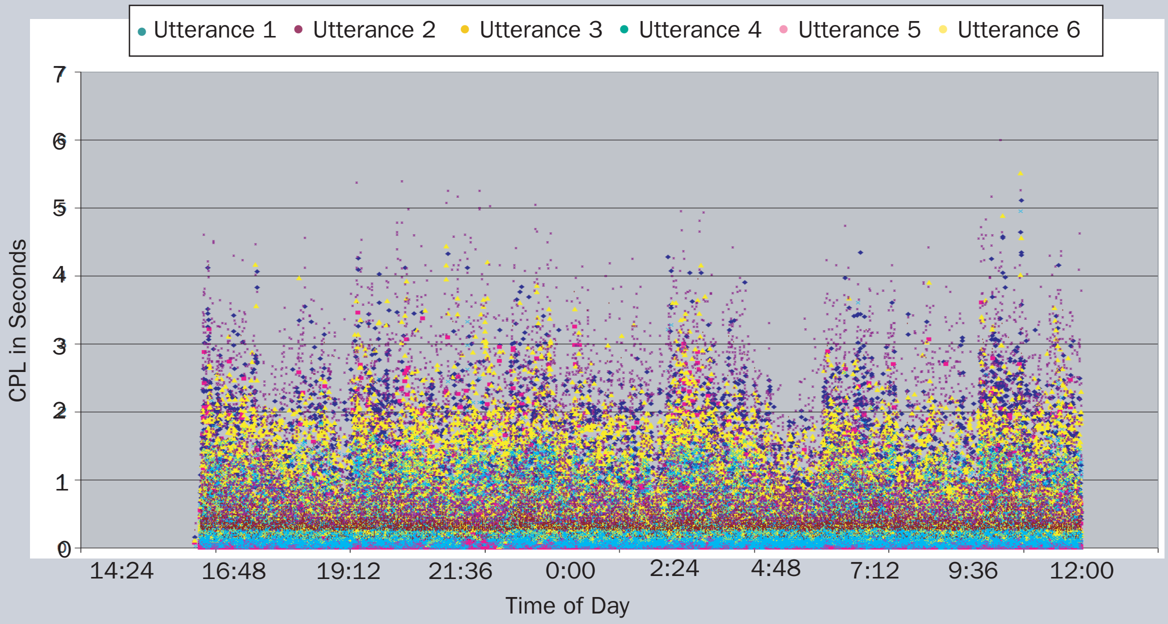
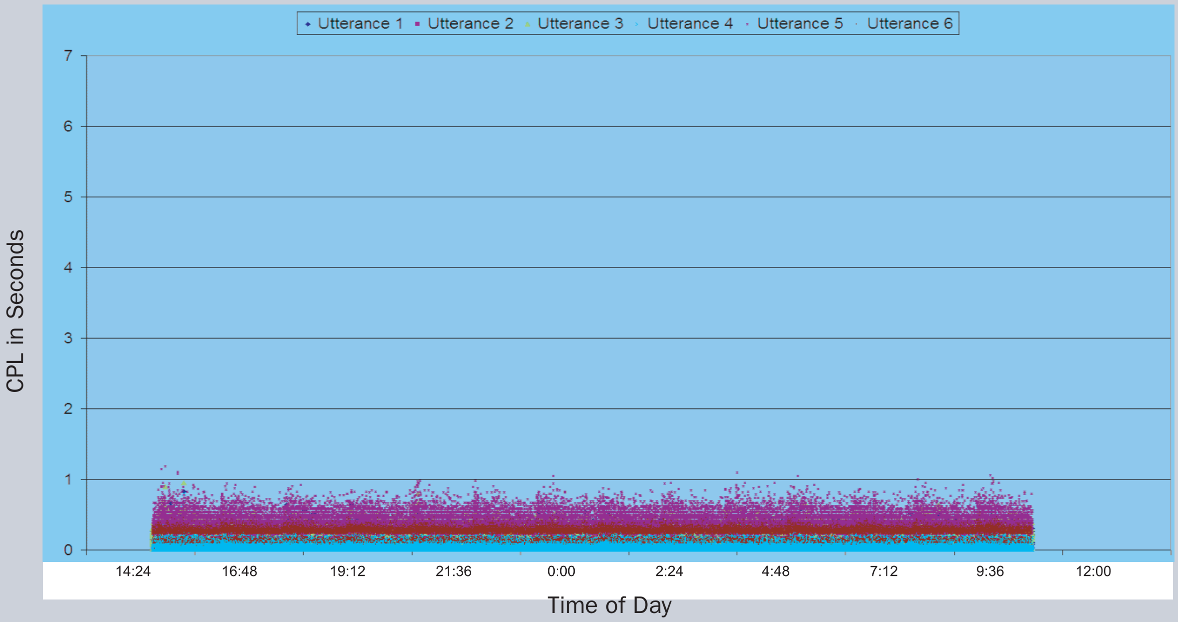


FIGURE 4 "After" Results: Same Hardware And Application With New Configuration Options



CPL? Each system and application has to determine a target. In our experience, many touch-tone implementations have CPL in the range of 1–3 seconds. Some applications aim for 95 percent of turns under 3 seconds, some for 95 percent under 2 seconds. However, we have found several applications, especially conversational ones, that have more stringent latency needs: in one system, for example, the CPL target was 98 percent under 1 second.

Microsoft's documentation about performance engineering for its Speech Server describes current industry solutions as having "80 percent of calls with user-perceived latency of less than 2.5 seconds". There is no substitute for understanding each application and audience, but our experience suggests an appropriate target for most voice self-service applications would be 95 percent of turns with CPL under 2 seconds.

Using CPL In Design

The best way to ensure that any technology deployment delivers on its business objectives is to establish performance goals before the design process even begins. This provides clear benchmarks for success, and helps guide both the solution design and its deployment.

For voice self-service applications, several measurements are important, including CPL, resource utilization (CPU%) and success rate (call resolution rate, opt-out rate). CPL is the most critical, however, since it dramatically affects caller satisfaction with the automated system and the rate at which callers opt-out of it.

The minimum achievable CPL (e.g., the response time of a single call on an empty system) can be measured easily during early development. Once an acceptable baseline for application latency is achieved, capacity requirements can be determined by estimating current and future call workloads. With a repeatable load test, you can then test and tune CPL during later-stage development and system integration. With this kind of careful planning and ongoing testing, the resources needed to maintain the CPL in the production environment can be adjusted fairly easily.

System Configurations And Parameters Matter

It is remarkable how much CPL can vary across different implementations. Modern voice self-service applications (especially those based on VXML) are made of many "moving parts." Even systems and applications with established configuration guidelines need consistent measurement and tuning.

Figure 3 and Figure 4, for example, show real-life "before" and "after" measurements of a conversational system for billing inquiries that, even at a moderate size, required 9 servers to operate. Each dot on these graphs is a single measurement, with different colors for different turns in the application. The Y-axis is CPL in seconds, and the X-axis is the time of day during the test run (these tests ran for nearly 20 hours each).

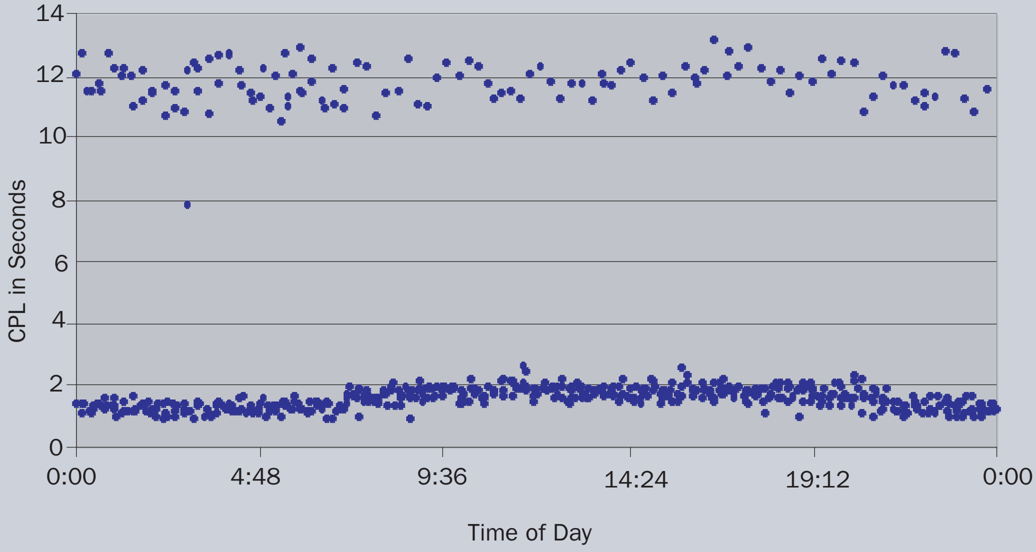
The CPL goal for this system was ambitious: 98 percent of dialog turns (customer perceived responses) taking under 1 second. (Measurements were taken in a lab, where the network delays are near 0, so users across a network would experience a CPL 100–300 ms longer.)

Using the system's standard configuration guidelines produced a CPL that was unacceptably slow; many dialog turns were taking as long as 6 or 7 seconds. The standard configuration guidelines failed in part because they did not address highly conversational applications, and in part due to plain old errors.

The appropriate configuration was determined

A good target for most voice self-service apps would be 95 percent of turns in under 2 seconds

FIGURE 5 Time-Based CPL Analysis Of A Phone Banking Application



The root cause of CPL variability is often found in one or more system misconfigurations or routing delays

by trial and error and repeated load testing, in this case numerous experiments over the course of three weeks. By changing a few hardware configuration options and rearranging the processes that operated on each machine, the goal was achieved: 99 percent of dialog turns went to under 1 second. Since CPL measurements were taken at the start of the planned development process, the extra work and extra testing added only a week to the overall deployment schedule.

Post-production Monitoring Of CPL

Once established, CPL measurements should be used throughout the voice self-service system lifecycle: design, development, system test, production. Many issues only exhibit themselves in production, and some develop due to changes that are made long after an application is launched and running well.

CPL data collected from a production system can be analyzed with several different techniques to provide greater insight into system behavior. This data is typically collected using periodic measurements over 1–2 weeks to gain a true indication of production trends and issues.

Measuring CPL in a production environment from across the same phone networks that customers use, provides empirical and accurate measurements across all hours of operation. Calls that mimic one or two typical self-service transactions are placed every 10–15 minutes (except for planned system downtime). At each step of the transaction, the CPL is measured and recorded. These measurements can be analyzed in three different ways, as follows:

■ **Time-based CPL analysis**: Charting the CPL by time of day provides empirical data with which to fully understand the impact of internal procedures on callers and to help make better capacity and tuning decisions. An example is shown in Figure 5. Each dot represents a single CPL measurement taken by test equipment automatically calling the application over the phone network and measuring the response times.

Notice that most of the calls have a CPL around 2 seconds, but a fraction of calls actually have a CPL around 12 seconds. This is a relatively extreme example, but these types of discrepancies are fairly common in distributed and complex systems. The root cause is often a misconfigured system or extra delays incurred in routing to particular systems or sites. Since less than 10 percent of calls in the example incur this high delay, the cause is unlikely to be caught without systematic measurements.

Figure 5 also shows that CPL rises slightly beginning around 7 a.m. and throughout the business day. While the increase is not large, it does indicate that the CPL at this point of the transaction is dependent on call volume. Future increases in call volume may cause more significant latencies. By tracking this data periodically, the application designer can better anticipate required capacity increases.

In other cases, we have seen CPLs as high as 30 seconds in the wee hours, typically caused by nightly system maintenance and backups. These activities are required, but their impact on customer experience may be overlooked—who calls their own IVRs in the middle of the night? Once

FIGURE 6 Comparative CPL Analysis—DTMF versus Speech Inputs

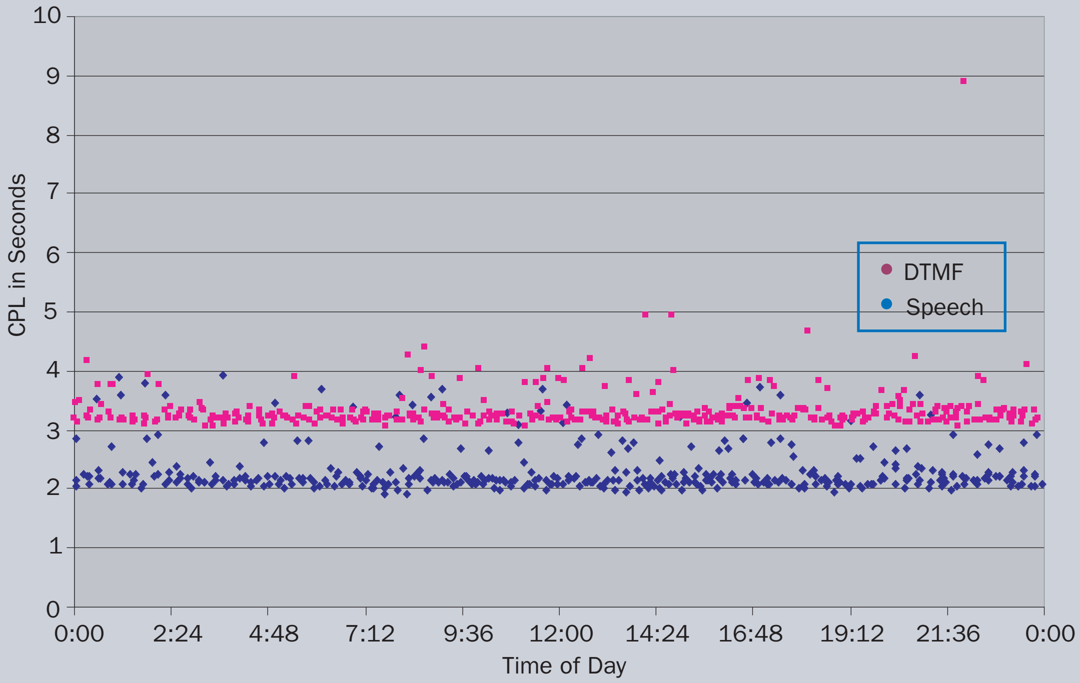
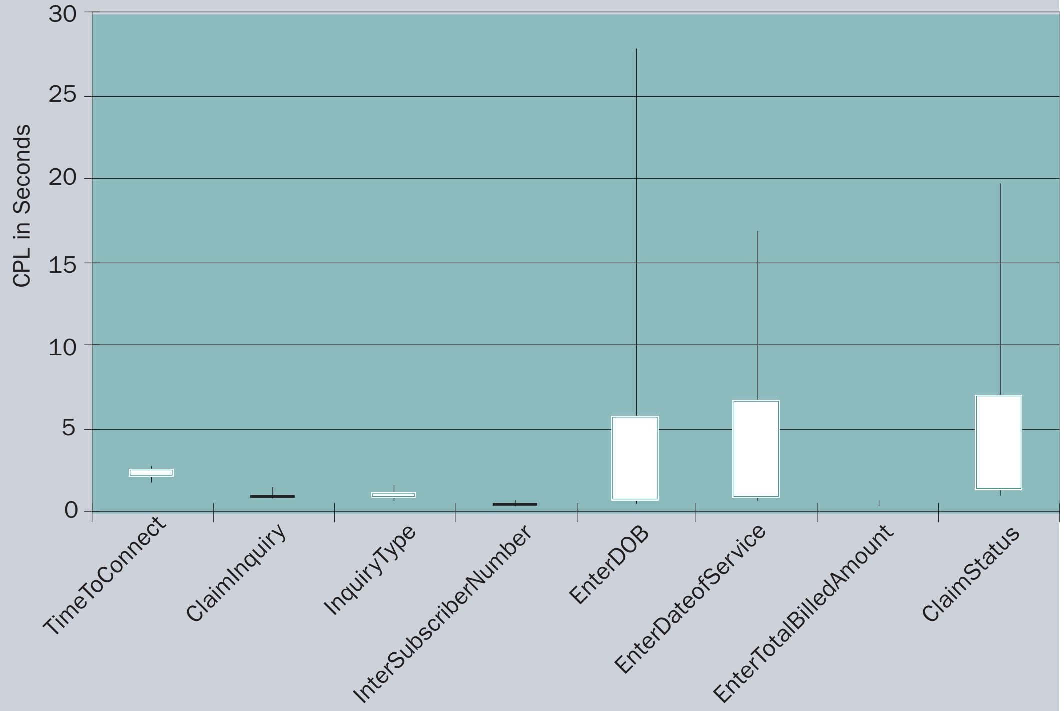


FIGURE 7 Transactional CPL Analysis Of Claims Application



Nightly system maintenance and backups can impact CPL in the wee hours

you know there's a problem, however, solving it is relatively easy: All that was needed in this case was to give the backup procedures on the affected systems a lower priority than the call processing on the target machine.

If late-night CPL measurements are consistent, but measurements during peak load hours are erratic, this usually indicates different performance on different servers or platforms in a load-balancing configuration. Such variability should be identified and avoided by effective pre-production performance testing—but if the pre-production work wasn't done, at least proper testing in production can help identify the problem.

While time-based analysis is most often done based on time of day, similar analysis can be done using day of the week, day of the month or seasonal times over the year (e.g., holiday traffic). In many cases, this reveals interesting and useful trend data that are otherwise missed by traditional monitoring methods.

■ **Comparative CPL Analysis**: This technique measures 2 turns that vary in only one aspect, in order to provide further detail to aid in troubleshooting or in making an architecture decision. The variable may be the physical server or location from which the CPL is taken, or the modality (DTMF/speech) used in the transaction. For example, Figure 6 shows CPL measurements of the same transaction performed using DTMF input and speech input. Both inputs are producing consistent measurements, but the speech inputs are producing a 1-second higher CPL.

This information can aid troubleshooting and resolution. For example, extra latency in the speech input responses may be due to excessive delays when processing speech on a remote server. On a review of the system architecture, it may be advisable in this case to relocate that server.

Another important example of comparative CPL analysis is to look at applications as they change over several months. As self-service applications evolve, their CPL can be affected by rising call volumes, as well as application and system changes. A good practice is to compare measurements against a baseline on a regular basis; this can prevent creeping performance degradation.

■ **Transactional CPL Analysis**: Rather than viewing CPL as a standalone measurement, this technique displays each step of a typical customer transaction. This view gives an excellent perspective on customer experience, and quickly highlights performance bottlenecks. Transactional CPL analysis also can enable a more informed decision for on-premise self-service upgrades or changes.

Figure 7 shows a typical analysis for a health care transaction (Claim Status). For each step of the call flow, the thin black bar shows the minimum and maximum range of the aggregated data over the collection time period (in this case 2 weeks). The thick white bar shows the 5th to 95th percentile data range, indicating where the majority of values fell. A consistent and well performing system will have a short black bar, and a small white bar, in addition to low overall values.

In Figure 7, the transaction shows excellent performance up until the "Enter Date of Birth"

Customer satisfaction is critical to an organization's overall success

prompt. From that point forward, CPL is both longer and highly variable (long thin black bars). In the case at hand, at this point in the transaction, the call is transitioned from network prompting by the carrier to on-premise prompting by interactive voice response (IVR) systems. We can infer that the processing involved in verifying date of birth and date of service against a subscriber and the work in retrieving claim status all suffer from load- or data-dependent delays, perhaps indicating a need for database performance tuning.

Conclusion

CPL is a new term for a simple, established idea: measuring latency from the caller's perspective. It can be a very powerful tool in creating and maintaining voice self-service applications. Customer satisfaction and the customer quality of experience are critical to an organization's success, and latency is an important element of usability and of the customer experience.

CPL can and should be used throughout the application delivery process: from design, through system test and deployment, to production monitoring. Measurements are done from outside the system, ideally across the same networks customers use. Products and services are available to make CPL measurements simple, non-intrusive and cost-effective.

Remarkably little is published about appropriate CPL targets, but it can be straightforward to establish and maintain good CPL and customer experiences. While consistency is more important than the absolute time to respond, a faster response time is generally better; and the more dialog oriented the speech, the faster the system response should be.

CPL varies significantly from platform to platform, even with the same application. Organizations which use hosted facilities to run their voice self-service applications can benefit from monitoring their CPL. Measurement and optimization of CPL in development and system test can result in vast performance and usability improvements, and avoid customer-impacting problems in deployment.

In the end, the way customers perceive an application's performance is a key factor in its effectiveness, and Customer Perceived Latency is the most important metric for measuring a speech application's performance□