glassbeam™

Glassbeam White Paper

Best Practices In Call-Home Data

www.glassbeam.com

glassbeam™

Contents

Business Case for Call-Home Data

We are living in an era of connected devices and products. These devices are constantly producing valuable operational data (hereafter "Call-Home data") on configuration, performance, usage, and other important parameters that define the very life of the device in the field. To instrument a device to effectively collect and relay Call-Home data is fast becoming a necessity; and forward-thinking corporations are actually differentiating themselves by the pedigree and maturity of their Call-Home initiatives.

Call-Home data, when properly analyzed, is a veritable goldmine of information on product usage, performance, feature popularity and other important metrics. This information can be acted upon with relative ease to favorably alter both sides of a corporate balance sheet. A product manufacturer can reduce support costs by significantly reducing mean time to resolution (MTTR) by quickly identifying the root-cause for any product issue. Similarly, technology companies can identify up-sell and cross-sell opportunities, build completely new revenue streams, from their customer base by actionable intelligence provided by this data.

These operational efficiencies are achieved by capabilities provided by Call-Home analysis, including:

- The ability to drill down to the root cause of any issue with both speed and accuracy using germane data and powerful parametric search.
- The ability to use system alerts to get advanced notification of any impending problem and to address these by remedial measures before the problem actually occurs.
- The ability to understand product usage by filtering the data by operational parameters.
- The ability to offer, to end-customers, diagnostics of how a system is performing in the form of continuous 'Health Check' services, and u The overall ability to glean detailed insights into almost any operational parameter either directly from the data or by intelligent parsing and analysis of this data.

Call-Home Data Is the Raw Material in IOT

The act of instrumenting a device to collect and send back Call-Home data is essentially the genesis for Internet of Things (IOT) analysis in the B2B world. IOT-savvy technology

companies are already reaping numerous benefits from the 'single source of truth' that Call-Home data necessitates – that is getting timely, accurate, and granular information about their install base directly from the field – and using this data to make informed operational decisions that help them align significantly more closely with their customers.

Most complex technology products like storage servers, medical devices and networking end-points send data in log files and other formats on a regular basis. These log files contain important and comprehensive data about the configuration, state, performance and usage parameters for that product. When this data is sent back to the manufacturer through email, ftp, or some other means, it is often referred to as Call-Home data since it is being 'Called back Home' to the original manufacturer of the device. The data can be in structured, unstructured or semi-structured format. Call-Home data is one of the richest, cleanest and most consistent source of information about your products. As is evident, the practice of collecting and sending back Call-Home data can virtually transform the way technology manufacturers operate. Unfortunately, the promise of Call-Home data, and all the 'downstream' benefits are not fully understood yet, and consequently designing the Call-Home bundle is often just an afterthought. In the rest of this paper, we will cover how you can design your Call-Home data to deliver the optimal amount of information at the right time, frequency, and in the most secure manner.

The Anatomy of Call-Home Data

Call-Home data contains key information about a system and its operation. The various pieces of information in Call-Home data have different uses and are important to different audiences. Making sure all key information is captured and transmitted in the Call-Home data is essential to getting the most out of the data being processed.

Call-Home Data Categories

The following table gives an overview of the main categories of information that should be included in Call-Home data. The design considerations for each category are explained in detail in the sections following the table

Call-Home Data Categories		
Category	**Use**	**Design Considerations**
System Identification (ID)	• Link Call-Home data to a specific system	• ID should be static • ID should not be repeated between system • Should be obvious and deterministic • May need a separate Super-ID or multiple keys at a broader system level for clustered devices • Should match keys maintained in backend CRM databases
Configuration Information	• View physical and logical layout • Track current system settings • Track system state • Extract cumulative statistic	• All system components should be covered • Log all relevant configuration information including static, configurable, state and statistics information
Statistical Data	• Troubleshooting data • Trending and predictive analysis	• Identify key metrics • Determine appropriate polling interval • Consider space required to retain data before logging • Define log format

Call-Home Data Categories		
Category	**Use**	**Design Considerations**
System Messages	• Event tracking and alerting	• Include message identification strings • Consider having unique message numbers • Use severity designations

SYSTEM IDENTIFICATION

While processing Call-Home data, collecting and collating historical data is critical to diagnostic and proactive analysis. System identification information lets you track which system the Call-Home data belongs to and allows archived Call-Home data to be tied together.

Typically, the main identification (ID) number is a system serial number. This number should be relatively static and must exist for the life of the system. Sometimes this ID number is configured into non-volatile memory or another configurable media on the system. However, if a change in the media results in a change in the ID number (for example, a part swap due to a media failure), it could affect the continuity in the system logs during post processing.

Another potential problem with ID numbers is duplication, a problem which can confuse analysis at the backend and can be difficult to untangle. To ensure that ID numbers are never duplicated requires a robust manufacturing process.

A system ID number is required for each system, but what constitutes a system? Often there are multiple, independent units that work together as a system. Such clustered systems should have a separate ID number for the entire collection of units that creates the cluster.

CONFIGURATION INFORMATION

System configuration information should include the following types of information for each device (software, firmware, and hardware) in the system:

Configurable values: Values set by the administrator when configuring the system. This includes values set by command line or user interface input (for example, setting the netmask for a network interface) and values set as a result of some other administrative action like upgrading software (for example, changing the software version number) or adding new hardware (for example, adding new node information to a cluster).

Static values: Values that do not change during the life of the system or component. Examples of static values include serial numbers and model numbers.

State information: Values that represent the current operational state or characteristics of a system. These values are changed by the system itself. They may change as a result of a human action (like a command issued on the console), but the state change is acted upon by the system.

Statistical information: System configuration information often captures a point-in-time value on key statistics. For example, the number of packets received or the number of errors on a network interface is statistical information.

STATISTICAL DATA

Operational statistics for a system are logged as statistical data and are critical to problem diagnosis and predictive analysis. Designers of log file data should determine the key statistical data for a system. This key data usually includes:

- Metrics that demonstrate the value proposition for the system (for example: response time, throughput, and compression).
- Metrics that help identify problem sources (for example: disk I/O, media errors, and network errors).

Designers should determine a sampling interval and make sure the system can retain that volume of data until the data is transmitted or local log files are updated. They can then define an appropriate format for recording the statistical data. The amount of data collected in logs can be quite large. Because there can be so much data, it is not unusual to dump the statistical data in a condensed format without much (if any) annotation. Even though the data might be unintelligible to humans, it can, with proper documentation, be parsed to extract rich data and enable significant statistical analyses.

SYSTEM MESSAGES

The bulk of Call-Home data is usually in the form of system messages like events and alerts that are logged at regular intervals. Although a system developer has the flexibil-

ity to define the format of the log messages, there are some basic guidelines. All event logs should include these essential design elements:

- **Identification:** Each message should be tagged with identification strings. The two most common identification strings are the timestamp and the originating module of software or hardware that created the message. Timestamp should contain both the date and the time. The date should contain the year and the time should contain millisecond, so that events that are triggered very frequently have separate timestamps.
- **Message number:** An alphanumeric message identifier allows messages to be categorized and parsed.
- **Severity:** Messages typically convey information or an alert. Alerts can have different levels of severity that should be considered carefully depending on the expected reaction from the system administrator. For example, different severity designations can be designed into messages as shown here:

Emergency:	System crash
Alert:	Needs immediate attention
Critical:	Critical condition
Error:	Error condition
Warning:	Possible problem condition
Info:	Informational message

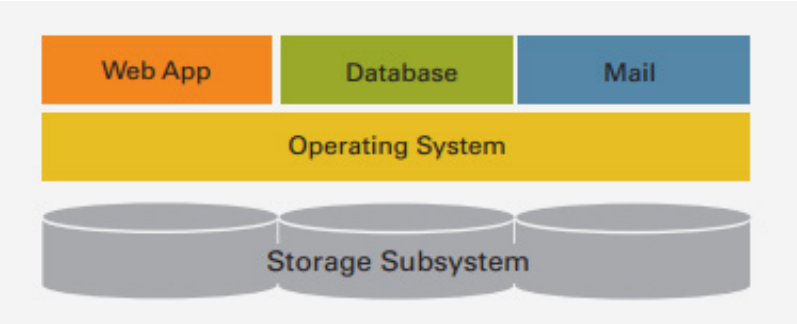
- **Frequency:** Messages should be logged as required. Some messages should be logged at predetermined intervals (for example, statistical data such as system uptime), while other messages are logged based on certain triggers. Triggers can be explicit failures or can be based on carefully defined thresholds.
- **Knowledge base:** While not directly related to the system messages, a knowledge base that maintains a catalog of message numbers and associated corrective actions and descriptions is very important.

Considerations for Complete Systems

A system is a set of components or subsystems sold by the vendor to function together as

what the end-user perceives as a complete product. At the lowest level, each component or subsystem may be able to generate Call-Home data. But more often than not, the CallHome data from the independent components is not as useful as the Call-Home data from the system as a whole.

Figure 1: Typical Data Center System



In a data center, a typical system is composed of several subsystems as shown above. When designing a Call-Home data package, it's important to consider which subsystem is the primary focus and what data needs to be sent along with the main subsystem information. You also need to determine when that information needs to be sent (see "Content of Call-Home Payload" later in this section for more information).

CLUSTERED SYSTEMS

Clustered systems are groups of independent systems that communicate and work with each other as a unit. In clustered systems it is important for each component to generate Call-Home data which can be correlated with data from other components in the cluster and which allows for the inference of all relevant relationships between cluster events.

Typically, clusters have higher order states, statistics, and configuration settings relevant to the cluster. This Call-Home data should be included in the data bundles sent from the cluster components. In addition, there should be configuration information—including component state data that represents the entire cluster—and a unique identifier for the cluster as a whole unit. Events and statistics pertaining to the operation of the cluster are also critical components of the Call-Home data for clusters.

Call-Home Data Parcel

When designing the payload and frequency of transmitting Call-Home data, the most important consideration is the intended use of the data. This lets you be sure that complete information is being sent at appropriate times.

For example, statistical data used in monthly trending reports does not have the same Call-Home urgency as a critical error event that may also need information about preceding events and up-to-date configuration information to resolve the issue. In the case of the critical error event, your system payload design should be complete enough so that a support engineer (or an automated process) can identify root causes and propose corrective actions without further data collection or customer interaction.

Effective system designs should consider the following:

- **Context of Call-Home data:** Understanding the state and configuration of the system for a particular observation.
- **Content of Call-Home payload:** Optimizing which operation data should be logged and transmitted.
- **Frequency of transmission:** Knowing when to send appropriate data to avoid swamping the Call-Home recipient.

Each of these considerations is discussed in detail in the sections that follow.

CONTEXT OF CALL-HOME DATA

In many cases, Call-Home data is only completely accurate or useful when observed in the context of the current system state and configuration. For example, a fatal error message on "Disk 2a.12" is useful to a support engineer only if the engineer is also able to determine the current model, slot, shelf, and state of the disk when the error was observed. Without this information, the engineer will need additional information before taking corrective action.

The design of the Call-Home data-packaging model must be able to ensure that an accurate view of both the state of the system and the system's configuration is available when needed.

To better understand the importance of the context of an observation, compare the two approaches to packaging Call-Home data in the following table.

Call-Home Data Packaging Models		
	Rich Data Parcel Model	**Event Log Streaming Model**
Content	• All elements of Call-Home data including events, configuration, state, and statistics up to the time of the Call-Home transmission	• Syslog events
Call-Home Frequency	• At the time of a critical event and/ or at predetermined frequencies (for example: daily or weekly)	• Periodically (for example, every few minutes, hours, or daily)
Pros	• A rich set of Call-Home data provides accurate contextual information at the time of an observation • Also receiving events at a predetermined frequency (e.g., daily or weekly) provides the opportunity for backend processing to handle events that were not deemed actionable at the time the system was designed • Call-Home Data such as configuration and state information is usually the output of system commands so no additional system design is required to gather this data	• Relatively simple to implement since only/all syslog events are sent • Since all events are sent, there is no need to predetermine actionable events. This can be done by the backend processing • Changes in system configuration and operation can be seen by the recipient of the Call-Home data almost immediately (if data is sent every few minutes)

Call-Home Data Packaging Models		
	Rich Data Parcel Model	**Event Log Streaming Model**
Cons	• Actionable events should be identified during system design and trigger a Call-Home notification	• All configuration, state and statistical information must be present in syslog format. This may not be true for many systems • More complex to reconstruct configuration/state from historical events for a given point in time. • Understanding the configuration may only be possible with the use of backend tools • Missing events prevent accurate reconstruction of context (for example, configuration)

The "Rich Data Parcel Model" in the first column contains elements of all Call-Home data categories as defined in "The Anatomy of System Call-Home Data". The second column shows a different approach, called the "Event Log Streaming Model," which simply sends syslog-type events at scheduled intervals.

Since the Event Log Streaming model doesn't provide the rich complement of Call-Home data needed for complex IoT analytics use cases, all further discussion of Call-Home design considerations in this paper will assume the Rich Data Parcel model.

Content of Call-Home Payload

Many systems provide detailed Call-Home data (such as system identification, configuration information, state information, and statistical information) in the form of user or administrator commands using the system's command line interface (CLI). Using the output of these commands, you can build a very effective rich data parcel with only a little additional software programming effort. Using these commands to provide information for the data parcel also makes it easy for the support group to use because most of the information is printed in well-formatted text.

The content of the Call-Home data parcel depends on the reason for initiating the Call-Home event. The following table summarizes three typical triggers for sending data Home and shows suggested content for each of the three Call-Home scenarios.

Rich Data Parcel Content for Three Call-Home Scenarios			
	System-Triggered	**Scheduled**	**User-Triggered**
Description	A predetermined condition occurs in the system that requires attention (for example, a power supply failure event)	A periodic Call-Home event to provide operation data for back-end processing such as proactive analysis, configuration change history, and data warehousing	An event where human intervention is required to transmit Call-Home data
System Identification	Required	Required	Required
Configuration and State	Required	Required	Required
Events Logs	All events since last one sent	All events since last one sent	All events since last one sent
Statistical Information	Not required	Required	Not required
Other Logs Such as Memory Dumps or Infrequently Used Files	Not required	Not required	Optional – depending on the type of trigger

The content of the Call-Home rich data parcel will generally include all elements of Call-Home data defined in "The Anatomy of System Call-Home Data". However, it may not always be necessary or desirable to send the full complement of information. For ex-

ample, cumulative statistics may be a sizable payload, but are usually required only for weekly or monthly trend analysis. Sending such a large data set with each data parcel would be unnecessary.

Many complex systems are built on general-purpose operating systems which may include services that also produce event logs. This is usually the case for email, web applications, databases, and other application servers. It is important to understand the role these logs play in the system as a whole so you can determine whether these logs will also need to be transmitted.

Less critical or rarely used data should be sent only when explicitly requested. Finally consistency is very important when creating data parcels, i.e., ensure that the file names of the log files are consistent, the content within those files are consistent and the way the files are packaged into a parcel is also consistent.

Formatting the Call-Home Data Parcel

When planning the internal format of the data parcel, designers should consider the following points.

- **Ease of parsing:** A well-designed data format speeds up the implementation of a machine data analytics initiative. Several techniques can facilitate parsing including the use of XML, name-value pairs, and aligned tables such as those that might be produced by well-formatted CLI output.
- **Use of compression**: The use of compression should be determined based on the processing cost to compress or un-compress data compared to the availability and cost of the bandwidth required to transmit the parcel within the appropriate timeframe. Generally speaking, compression is preferred.

FREQUENCY OF TRANSMISSION

How often a data parcel is sent is determined by the severity of the alert that triggered the transmission and the amount of information accumulated in the data parcel. For system-triggered alerts, the data parcel should generally be sent immediately. However, since many system error events can occur repeatedly over a period of time (for example, every second, every minute, every hour, and so on), it is impractical, annoying, and potentially overwhelming to send the data parcel every time the same error occurs over a short period of time. In this case, it is necessary to incorporate an "anti-spamming" methodology into the Call-Home processor.

For example, consider a system that checks for a critical over-temperature condition

once every five minutes and logs a critical event if the specified temperature is reached or exceeded. Now assume that the over-temperature condition lasts for one hour. If a system-triggered Call-Home data parcel is initiated with each over-temperature event, twelve Call-Home data parcels would be sent.

In this case, it is more practical to send one data parcel when the event is first logged and then transmit another data parcel only if the condition has not been cleared after some reasonable time has passed. If you are using a reliable transport mechanism, you may not need to retransmit the same alert at all. Another rule of thumb is that an alert notification should not be sent unless specific action is needed to correct the situation.

The frequency of scheduled Call-Home data parcels can also be determined by the amount of data accumulated and the maximum acceptable time between receiving log event updates. For example, if you do proactive problem analysis using event logs, the events should be sent on a time interval basis that is consistent with the urgency of the potential problems being analyzed.

It is easy for a prolific event-logging subsystem to overwhelm not only the Call-Home system, but the recipient of the data as well. However, it may be important to receive all such events. Understanding the growth rate of event logs will help determine the frequency to transmit scheduled data parcels. For example, rapid log growth may be the governing factor when deciding to send scheduled data parcels once per day rather than weekly.

Security Considerations

When designing log data and deciding how it should be transmitted back to a central repository, security and transport mechanism are the two important considerations. CallHome data should contain the content required for post processing, but should not contain information that a malicious agent could use to breach the data center's firewall or other company security policies or mechanisms.

Transport security is important for protecting the Call-Home system from being the target of external data tampering (spoofing) or denial of service attacks (spamming).

Masking Sensitive Information

Information that can be used by intruders to access other resources or networks in the data center should be identified carefully. Examples of such information include IP addresses, personal identification like Social Security Numbers, and passwords.

If appropriate, logs should be designed to mask such information. Depending on the accessibility of the Call-Home data, the information can be masked before writing the

logs or before transmitting to a remote location. Of course, if the local copies are fully secure, and there is no risk of unauthorized access to the data, then the best approach is to implement a filter to mask sensitive information before transmitting it externally. The back-end data interpretation systems are responsible for dealing with anonymous data.

Protection from Spoofing and Spamming

A "spoofing attack" is a manual or programmatic interception, manipulation, and modification of data. Spoofing is possible in non-secure IP protocols such as SMTP-based email or HTTP. Data can be modified to misrepresent information or to create denial of service (DOS) attacks by replicating mail. Data that has been altered can corrupt the analysis of the data, and DOS attacks can overwhelm the Call-Home receiver system.

Secure IP-based connections such as HTTPS or secure email that use Secure Sockets Layer (SSL) are usually a better way to send logs Home without being exposed to the risk of spoofing.

Preventing Eavesdropping

Plain text data can be read and changed in transit. A "man-in-the-middle" (MITM) attack is a cryptographic term used to describe situations where an attacker reads, and then inserts and modifies information in messages between two parties without either party knowing that the link was compromised. Of course this compromises the integrity and value of any analysis done on the Call-Home data.

Using HTTPS or other implementations involving SSL is an effective way to minimize this risk. HTTPS uses TCP/IP port 443 and an additional encryption and authentication layer between HTTP and TCP/IP by establishing an HTTP connection over an encrypted SSL connection. The Call-Home server must implement the mechanism that creates and authenticates public key certificates and then accepts the connection from the transmitting systems.

Call-Home Transport Mechanisms

Call-Home transport mechanisms can include one or more of the following protocols: Email, HTTP, HTTPS and dial-up (over phone lines). The following table summarizes the pros and cons of each of these protocols.

Call-Home Transport Protocols		
Transport	**Pros**	**Cons**
Email	• Pervasive • Simple configuration End-user auditable • Simple to test after setup	• No guaranteed delivery May not be encrypted • Can be spoofed and/or spammed
HTTP	• Pervasive (port 80 open) • Simple Configuration • Reliable/guaranteed delivery	• Unencrypted
HTTPS	• Secure/encrypted Reliable/guaranteed delivery	• May require firewall configuration • Not auditable
Dial-up	• Secure • Reliable phone networks	• Costly infrastructure required • Less throughput

Each of these transport mechanisms has advantages and disadvantages that can make it difficult to choose just one. For example, the HTTPS solution appears appealing due to its secure nature. However, for customers who must audit all outgoing data transfers, HTTPS may not be a viable solution in their environment.

The Impact of Transport on Call-Home Usage

To achieve the maximum use of the Call-Home facility from the installed product base, you should consider the following when designing the Call-Home mechanisms.

Necessary Security

As you design the security for your specific environment and for the data being sent, make sure not to under- or over-engineer the security requirements. For example, Call-Home data that does not contain customer information (for example, Social Security Numbers) may need less-stringent security requirements than a web site that does allow access to customer information.

You must also understand the security requirements of the end-user environment

where the product is to be installed. Financial institutions and government sites have requirements that are different from those of a small business operator or home consumer. Some customers, for example, may allow only unencrypted data to leave their premises (so they can monitor outgoing communications), while others may not have defined rules one way or another. Still others may never allow Call-Home data to leave their premises under any circumstance, or they may allow data to leave only after it has been manually screened.

Usability

It is also important to remember that configuring the Call-Home facility for security may be the responsibility of the product's administrator, not the person who is familiar with the overall IT networking environment. Configuring the Call-Home facility must be obvious, discoverable, simple, and easy to test for correctness.

Since customers will have varied requirements, a single solution may limit the adoption of the Call-Home facility. It may make sense to consider implementing more than one transport mechanism. For example, an implementation that has both HTTPS and email may satisfy a broader spectrum of requirements.

Contact us at sales@glassbeam.com

Glassbeam, Inc.

6001 America Center Drive, Suite 250 • San Jose, CA 95002
Phone: 408-740-4600 • www.glassbeam.com