Advanced Distributed Debrief for Joint and Coalition Training

Randy Pitz, Curtis Armstrong
The Boeing Company
St. Louis, MO
randy.pitz@boeing.com, curtis.a.armstrong3@boeing.com

ABSTRACT

As the need increases for mission training centers to support large-scale distributed, joint and coalition training exercises in addition to their traditional individual, team, and sometimes collective focus, a corresponding increase in system capability is needed. While current debriefing capabilities may support the traditional training center mission well; these systems often do not effectively support distributed joint and international participation. This paper will discuss the challenges of large-scale distributed debrief focusing on the problem of distributed record and playback. An approach is described that is capable of automatically keeping locally recorded data, time synchronized across a wide-area network. This approach provides distributed synchronization without the requirement to replay data across the distributed network, or through use of common tools. Such capability enables warfighters to use the same tools with which they are already familiar. Details are discussed, including how it achieves synchronization, extension possibilities, security considerations, and analysis of implementation options. Relevant research and experiences with distributed debrief will be discussed including some innovative ideas for advancing debrief state-of-the-art as follows. Highly accurate and automated synchronization, distributed network bandwidth reduction, multi-level security support, and easy legacy system integration to name a few. Recent experiences from the U.K. Mission Training through Distributed Simulation (MTDS) will be included to describe these benefits. The implications of this work point to the need for standards development for distributed debrief. Standardization will lead to improved interoperability for large-scale distributed debrief.

ABOUT THE AUTHORS

Randy Pitz joined Boeing Training Systems and Services in 1997. He is currently the Principle Investigator for Distributed Mission Operations (DMO) Technologies, currently focusing on transitioning technologies to various training programs within Boeing. He has also served as the Principle Investigator for Debrief Systems focused on developing debrief architectures including integration of situational awareness displays; digital video and audio; cockpit switch and display repeater; record and playback; and performance assessment technologies. He has worked on many other projects involving simulation networking technology and instructor operator stations for various training system programs. Mr. Pitz holds a Master's Degree in Computer Science from Washington University.

Curtis Armstrong joined Boeing in 2005. In addition to distributed debrief, his recent research includes virtual instructor technology and visual database management. He earned a Master's Degree in Computer Science from Brigham Young University in 2006. His thesis research involved methods to automatically convert raster images into vector images.

Advanced Distributed Debrief for Joint and Coalition Training

Randy Pitz, Curtis Armstrong
The Boeing Company
St. Louis, MO
randy.pitz@boeing.com, curtis.a.armstrong3@boeing.com

INTRODUCTION

Joint and Coalition training exercises are becoming more prevalent as the needs for these types of missions increase. The ability to perform live training for individuals and teams for Joint and Coalition training missions is challenging due to the costs, logistics and policies among other factors. Applying virtual simulation to this training need has huge potential to offset these challenges. Recent experiments including Red Skies, First Wave, Virtual Flag and U.K. Mission Training through Distributed Simulation (MTDS) are paving the way to this future. As the scale and complexity of the virtual simulations increase to support Joint and Coalition exercises, existing practices may become obsolete. The focus of this paper is about an approach to distributed debrief that provides synchronization across sites, is scalable to large scale exercises, and fosters interoperability.

The study of distributed debrief systems have been a part of many recent activities. In particular, past U.S. Air Force Coalition Mission Training Research (CMTR) experiments involving the Air Force Research Laboratory (AFRL) have successfully performed Coalition-like exercises involving debrief (Greschke, Mayo, Grant, 2002; Gehr, Schurig, Jacobs, van der Pal, Bennett Jr., Schreiber, 2005; Smith, McIntyre, Gehr, Schurig, Symons, Schreiber, Bennett Jr., 2005). These experiments have been expanded recently during the Battle Buzzard and Condor Capture exercises as part of the U.K. MTDS Capability Concept Demonstrator (CCD) program. Other approaches to distributed After Action Review (AAR) protocols have also been discussed (Travers, Ferguson, Langevin, 2006; Seeger, Jergens, Devol, Owens, 2002). These researchers and engineers have been at the forefront of developing concepts that help enhance distributed debrief.

Many virtual simulation training centers exist across the world to service the training needs of warfighters. These Mission Training Centers (MTCs) provide access to valued resources and operators for training in Joint and Coalition contexts. Leveraging these facilities is a cost effective way to support larger scale scenarios. The existing systems at these facilities are often designed for a specific training purpose and work well in support of that mission. As these training centers expand their mission to include mission rehearsals and Joint and Coalition training, their debrief systems will be put to task. While many existing system components work well to enhance debrief and enable the warfighter, the lessons learned from larger-scale training experiments need to be applied to guide future upgrades and development. Examples of systems that have worked well include off-the-shelf video teleconferencing systems (VTC), touch based whiteboards/SmartBoards and office automation tools like Microsoft Office. However, providing a reliable and robust technique for recorded event replay for distributed debrief remains a challenging endeavor.

Large-scale exercises such as Joint and Coalition training presents challenges for distributed debrief. The nature of Joint and Coalition training implies geographically separated sites, with equipment provided by a variety of suppliers. Keeping geographically separated debrief systems time synchronized and coordinated is critical to providing effective debrief. Standards such as Distributed Interactive Simulation (DIS) and High-Level Architecture (HLA) exist for making training devices interoperable, but there is no equivalent set of standards for creating interoperable debriefing systems.

This paper is organized as follows. First, background about existing distributed debrief practice is presented, followed by methods for addressing the challenges of large scale distributed debrief event replay. Then an analysis of benefits and tradeoffs to the techniques introduced in this paper is presented. Lastly, the paper concludes with recommendations and future research possibilities.

BACKGROUND

In this discussion a debrief system is defined as a collection of hardware and software tools used to replay, analyze and evaluate the actions of trainees that occurred during a training session, thus, implying the existence of recording and playback systems. The types of data recorded for debrief can include simulation network protocols (i.e. DIS or HLA), video, audio, and other specialized data. A debrief is conducted to reinforce learning objectives and validate trainee actions against those objectives. The trainees can consist of a single student, crew, team, or possibly larger groups of individuals. An example debrief agenda may consist of a mission plan review, replay or review of actions, identification of lessons learned, evaluation and results reporting. A *distributed debrief* occurs when multiple geographically separated parties desire to conduct a debrief as if they are all at the same location.

Existing approaches used in debrief systems targeted for single site operation will not always be compatible or scalable with distributed operation. A survey of techniques used for event replay during distributed debrief indicates that several different approaches are being used. The most prevalent technique uses network retransmission of recorded data to drive the debrief systems. Another replay technique is manual synchronization of tools. A third technique employed is use of a common set of tools for all distributed participants.

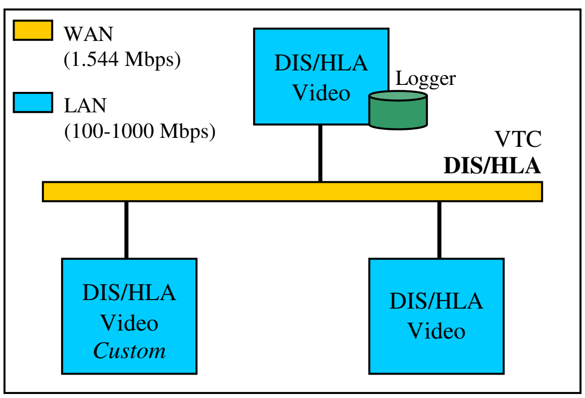

Figure 1 - Network Retransmission of Data

The first technique, network retransmission of data, involves having a remote recording device play the data back across the Wide Area Network (WAN), as shown in Figure 1. While this solution is easy to implement, it has several drawbacks. First, playing data back over the WAN uses a significant amount of bandwidth. Typically, the WAN is limited to a few training sessions at the same time, so replaying data on the WAN for debrief prevents that bandwidth from being used for concurrent live training sessions. Second, the transport latency is effectively doubled for data recorded on a remote site. This is because each packet is transmitted over the WAN twice: once for the original transmission, and once for the playback. On some WANs, the one-way latency is already significant. Similarly, for data sent over less reliable channels, such as UDP, the probability of packet loss is doubled since the packet is sent twice over the WAN. A third disadvantage of recording data at a remote site is that some of the local data might not be available at the remote recording site. The locally-recorded data may include greater detail than the data transmitted to remote sites in an effort to reduce bandwidth usage, or fulfill security requirements. This data would be available for local playback but not remote playback.

Rather then depending on network retransmission, the practice of manually coordinating debrief is sometimes used. Typically a human operator at each site will start playback, while using a telephone call or VTC to coordinate the start time, speed and playback offset. This approach has the advantage of being trivial to implement and doesn't require the development of an interoperable protocol. However, the disadvantage of this approach is a lack of automated synchronization, and operator error can affect the quality of debrief. For example, one operator could start slightly later than another, causing devices to be poorly synchronized. This is especially a problem if trying to synchronize multiple devices in the same room, since the tolerance for poor synchronization is quite small. Two separate devices in the same room could also produce noticeable clock drift over time due to a lack of automated synchronization. A lack of accurate synchronization can lead to great confusion by observers and lead to incorrect assessment during debrief.

Another approach to distributed debrief is use of homogenous tools across the distributed network. This type of solution, also referred to as "common tools," standardizes on tools rather than a specification. This kind of "one size fits all" tool standard may be tempting because some tools may be highly successful to solving a particular training system's needs. However, integrating foreign tools into existing systems tends to disrupt the target system, particularly when an existing tool is displaced. For example, users need to be trained on the new tool, and there are often missing features during early adoption phases. Forcing the warfighter to adapt and learn new tools that have replaced familiar tools can be a training detractor. Standardizing on interoperable protocols rather than common tools develops a stronger industry which provides innovative solutions for the warfighter.

An alternative to the techniques identified above is to synchronize device playback using a control protocol so that each computer knows when to start and stop playback. The protocol can also synchronize playback as it is occurring to account for clock drift. There is currently no widely-used or accepted protocol to synchronize and control event replay of data across multiple computers. However, U.K. MTDS experiments and trials have lead to some promising results.

U.K. MTDS Experiences

The UK MTDS CCD program has experimented with two techniques for distributed debrief event replay. During the Battle Buzzard and Condor Capture exercises, daily distributed debrief between Royal Air Force (RAF) base Waddington and AFRL in Mesa, Arizona were conducted. The sites were connected over a secure WAN with VTC, SmartBoard and event-replay system components. For these debrief sessions both the network retransmission and synchronized control protocol approach were used to control event replay. The synchronized control protocol used for these trials was developed over many years' experimentation and trials by QinetiQ Ltd. When using the network retransmission technique, the VTC quality between the sites became so poor it was rendered unusable. However, when using the synchronized control protocol, the VTC was stable and exhibited good quality. The source of the VTC problem was discovered to be the high bandwidth used when VTC and network retransmission of DIS packets were combined. Clearly, the distributed debrief would not have been as successful if the network transmission approach was the only solution available.

A DISTRIBUTED DEBRIEF SOLUTION

The following paragraphs will describe a protocol for controlling and synchronizing playback of devices across distributed sites while using a common representation of time. It leverages existing tools and practices as much as possible, while providing a viable alternative for scaling up to include many distributed sites.

Overview

A control protocol is currently under development called Distributed Debrief and Control Protocol (DDCP). It is designed to address many of the challenges of large-scale distributed debrief. First, it provides time synchronization of distributed systems and has provisions for issuing commands uniformly across all interested sites. Second, it limits use of bandwidth by eliminating the transfer of recorded data for event replay. Third, it organizes around groups to support multiple concurrent debrief sessions and hierarchical control structures. Fourth, it supports implementation over any network transmission layer. Lastly, protocol extensions are supported for "future-proofing" systems, i.e., a technique which provides the user with the ability to extend the protocol without the need for extensive data structure changes.

A control protocol for any general purpose playback system can be easily established. The protocol is manipulated using typical media playback features found in VCRs, DVD and DVR players. These controls consist of record, play, pause, fast-scan and seek. When issuing these controls as commands, provisions are made so that they happen uniformly across all sites.

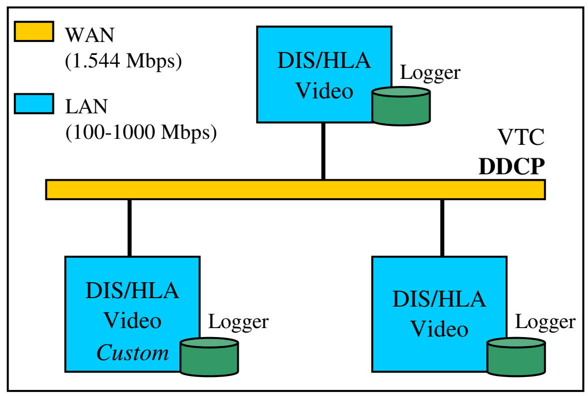

Figure 2 - Synchronized Debrief

To limit bandwidth usage, only control messages are sent over the network, not the data itself as Figure 2 shows. It is expected that each site records data independently. Since this protocol does not send recorded data over the network, any site that has not recorded data necessary for debrief must use some other transfer mechanism (such as file transfer) to retrieve the data from a site that did record it. Figure 2 implies that each site records multiple data types that can later be replayed during synchronized playback. Limiting the protocol to control messages also makes it easier to support the security policies in multi-level security systems, which is a topic that will be discussed later.

Devices are organized into groups, with each group coordinated to the same playback time by a *master*. The master device is predetermined prior to debrief and manages remote device membership. The group concept allows multiple debrief sessions to co-exist at the same time. Debrief groups typically never interact with each other. The protocol includes messages for managing group membership.

The protocol has been prototyped using UDP as the transport layer. However, the protocol is transport-layer agnostic and can just as easily be implemented using DIS, HLA or other simulation protocols. DDCP is not intended to be a replacement for any simulation network protocol, but rather complement them.

The protocol incorporates several features necessary for complete control of a distributed debrief session. These features include synchronization, recording, playback and group management. Each of these features will be discussed in turn, along with a discussion on implementation, reliability and extensibility.

Coordinated Time and Synchronization

The first challenge to achieve a distributed debrief is to establish a uniform playback time across all sites. Here the *playback time* is the virtual time within the debrief session; in other words, the wall-clock time when data was recorded. For example, suppose a particular training session was recorded from 0800 to 1042 UTC. Later, the data is replayed starting at 1600 UTC. As seen in Figure 3 during debrief, the playback time increases from 0800 to 1042, as the wall-clock increases from 1600 to 1842. An item of data recorded at 0820 would be played back at 1620, since the play clock is 0820 at wall-clock time 1620.

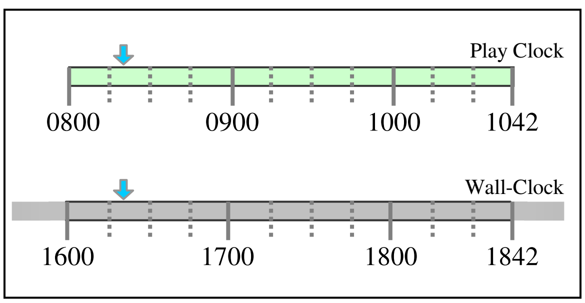

Figure 3 – Coordinated Time at 1x Playback Speed

For this to work, the devices are assumed to already have access to synchronized wall clock time; each device should already know the correct UTC time, so that even devices in separate time zones can be synchronized. The system clocks may be synchronized by Global Positioning System (GPS) receivers, atomic clocks, Network Time Protocol (NTP) servers, or other means. Because of this assumption, playback is not synchronized better than the system clocks.

Without some form of synchronization, it is possible that the various data sources could gradually lose synchronization during playback. To keep devices synchronized over time, the master periodically sends *Synchronize State* notifications. These notifications synchronize a clients current playback time (t_p) to its wall-clock time (t_c). The playback time is the time the data were recorded. In order for this to work, recording and playback devices all need to have synchronized clocks. Each synchronize state notification indicates what the masters playback time (t_s), or *Synchronize Time*, should be at a particular moment, called the *Effective Time* (t_e).

The synchronize state notification relates the current playback time to the current wall clock time according to Equation 1.

$$r_n(t_c - t_e) = r_d(t_p - t_s) \qquad (1)$$

This equation uses the *Effective Time* (t_e), the *Synchronize Time* (t_s), the *Speed Numerator* (r_n) and the *Speed Denominator* (r_d) from the most recent synchronize message.

Sending synchronize state notifications periodically overcomes any packet loss, without loss of state. If a device misses a state transition, it is able to enter the correct state after the next periodic synchronize state notification received from the master. Also, any device that joins the group after the group has started playing will know the correct state the next time it receives a synchronize state notification.

Recording

Recording is another critical step to achieving synchronized debrief. Without properly overlapping timelines, it becomes difficult to verify that two recordings occurred for the same exercise. One way to verify overlapping timelines is to use a coordinated record capability that is provided in the DDCP protocol. However, even uncoordinated recordings can be time correlated through the use of a synchronization event. which will be discussed later. For control purposes the debrief master creates a master timeline that consists of the union of individual timelines.

Playback

The synchronize state notification contains current playback state and time synchronization information for all members of a group. The playback state can be play, stop, record or seeking. Additionally there are fields to control playback speed, r_n and r_d as mentioned above, which can be negative for reverse or

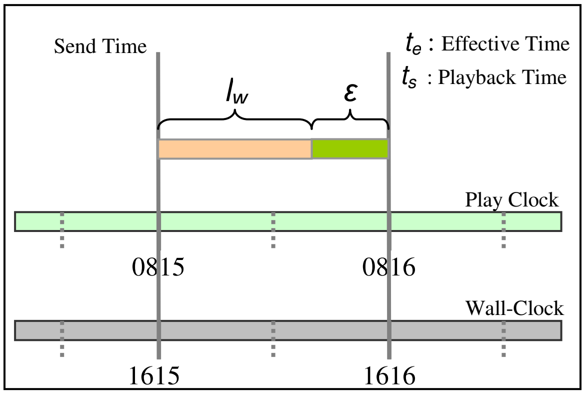

Figure 4 - Send, Effective and Playback Times

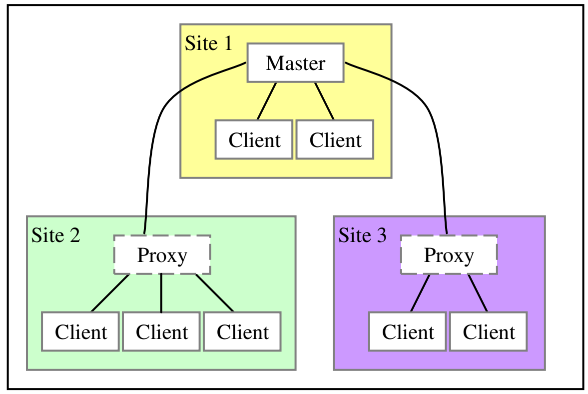

Figure 5 - Debrief Groups

positive for forward playback. The stop state is the default state and denotes no progression of playback time, t_p. The play state denotes playback time is progressing and is affected by the speed. The pause state can be achieved by expressing a zero playback speed. The record state can be used by recording devices for coordinated recording as discussed above.

When issuing a state change, the effective time is used to control when the state change is to occur. Typically, the master will set an effective time sufficiently long enough into the future so all devices will take action uniformly. Thus the effective time needs to accommodate the worst-case one way latency of the WAN (l_w) plus a constant (ε) to account for system processing overhead. The effective time is calculated as follows and is demonstrated in Figure 4.

$$t_e = t_c + l_w + \varepsilon \qquad (2)$$

Seek capability is provided and supports instantaneous as well as long duration seek operation. Some devices will be able to support seeking to a new playback time in a short or near-instantaneous amount of time. However, other devices may take much longer to seek and must be accommodated. This protocol supports a device responding to a seek command with a "seeking" state and an estimate to completion. This information allows the master to remain aware of what is occurring and prevents a premature start of the playback.

Group Management

A *debrief group* is created by a master device. The master manages group membership of local and remote devices by using join and leave commands. A remote device can request a change in membership by using request join and request leave commands appropriately. The group is typically established before debriefing begins so that the debrief master can monitor member status. A *proxy* device is used to control local devices that may not be able to communicate with a remote master. This situation can occur when firewalls or other security measures are used. The proxy device becomes a single point of control for all devices located at a site and acts on behalf of a master or protected devices. See Figure 5.

Reliability

The protocol is designed to operate over unreliable networks and protocols, such as UDP, since such protocols are generally more efficient. Operating over UDP also allows the use of multicast. The protocol achieves reliability using acknowledgement messages, message retransmission and periodic transmission of some messages. The synchronize state notification mentioned above uses periodic transmission to achieve reliability. Group management messages rely on message acknowledgement and retransmission for reliability.

The master can at any time request that remote devices reply to a command. This will often be done when changing a critical state such as playback mode. The master can use received responses to determine whether any packets might have been dropped and re-issue the command. This is very similar to how reliable protocols such as TCP are implemented. Periodic message transmission helps to overcome packet loss for messages that don't require tighter handshaking between master and remote device. If a periodic message is dropped, a future message contains the complete state information needed by a device.

Extensibility

As any system developer knows, "future-proofing" is an important capability. DIS and HLA both have their extension mechanisms, and so it follows that a distributed debrief protocol should as well. DIS uses experimental PDUs, fixed and variable datum records, while HLA uses FOM composition. DDCP uses DIS-like datum records to extend messages. This can also be thought of as message bundling.

An illustrative example is used to help the reader understand the DDCP extension mechanisms. Travers, Ferguson and Langevin (2006) presented a debriefing concept of sharing common perspectives for cooperating teams within a training exercise. The shared perspective concept allows other teams to better understand one team's perspective using the tools they normally use rather than using a common tool. In this case, a few tools were modified with the ability to pass eye-point and related perception data to each other. During debrief, users would identify points of interest and have the other teams examine a situation using the perception of a particular team. Using DDCP to implement such a feature would be straight-forward. The eye-point and perception data that was being shared between applications constitutes an application-specific DDCP record, for example a Perception record. This record could be bundled with other records or sent alone. Any device that does not support this extension would just ignore the record, but still be capable to participate in the debrief session.

Solution Summary

This section introduced the DDCP synchronization concepts. The argument for a common time representation was made and details about the distributed protocol were presented. The next section will discuss these choices in more detail and compare and contrast the relative benefits.

ANALYSIS

The DDCP approach to conducting distributed debrief is now considered and compared against existing practices. In this analysis, the accuracy of the automated synchronization is considered, followed by an examination of bandwidth utilization. Then a look at approaches to supporting multi-level security requirements is considered.

Accuracy

The accuracy of a distributed debrief is controlled by the degree of clock synchronization of the involved devices. Without a distributed clock synchronization solution, debrief control would be ad-hoc and unsynchronized. There are many different solutions that can provide clock synchronization. For example, atomic clock receivers accept signals from Fort Collins, Colorado, Rugby U.K., or Mainflinger Germany. However, there is not one atomic clock radio that can service the whole globe. On the other hand, the Global Positioning System (GPS) constellation of satellites, can service the entire globe. The drawback with these systems is that they require specialized receiving equipment in order for devices to acquire the highly accurate signal. A more viable solution is the use of Network Time Protocol (NTP). NTP, configured to use either atomic clocks or GPS signals, provides slightly lower accuracy than using direct GPS signals, but is much more cost-effective to implement. All modern operating systems include NTP capability including Windows XP, Linux, and Solaris.

Since atomic clocks cannot provide global service and any two atomic clock providers are not guaranteed to be synchronized, it is not considered further in this analysis. However, GPS does provide global service and also is currently used in many simulation facilities for providing a source of absolute timestamps for DIS or HLA simulation network protocols. Thus, GPS has the distinction of being widely available, highly accurate and in some cases already in place. GPS time accuracy for Standard Positioning Service (SPS) is 340 nanoseconds, while Precise Positioning Service (PPS) provides accuracy of 200 nanoseconds (USNO GPS Time Transfer & Dana, 2000). This is why it is used as a common source of time accuracy for commercial telecommunication networks, as well as DIS and HLA networks. While the accuracy of GPS is very good, distributed debrief does not demand this accuracy.

The maximum accuracy needed of a distributed debrief is assumed to be 33 milliseconds, or 30 Hz. The source of this requirement comes from the video and motion picture industry. In that industry 29.97 frames per second is the standard for broadcast media. The derivation of that standard comes from the physiology of the human eye, the brain's processing power and a bit of clever engineering[1]. For debrief review purposes, more frequent frames per second would be wasteful as the human eye can't detect more frequent changes. So for distributed debrief this requirement is taken as a reliable source of engineering. Given this criteria the accuracy of GPS would meet the needs for providing

---

[1] Scientists and engineers at the time needed a way to build in predictable error so that audio and video interference was less noticeable.

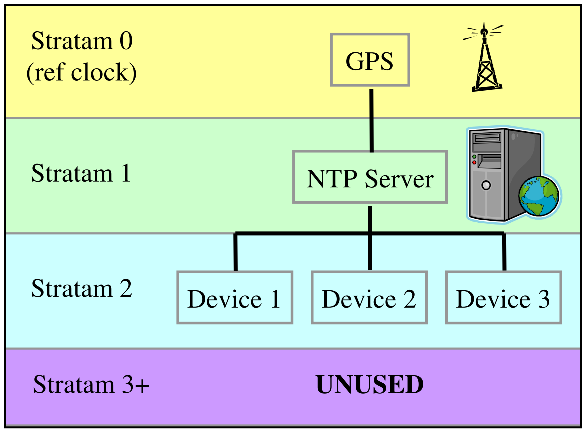

Figure 6 - NTP Strata

accurate 33 millisecond timestamps. A lower cost solution may be found in NTP.

NTP consists of a hierarchical system called clock strata. Lower numbered strata provide input to higher numbered strata as Figure 6 illiustrates. Stratum 0 is where reference clocks are found - atomic clocks, GPS and the like. This level is the most critical for properly feeding the NTP servers at stratam 1. Stratum 2 is where the first instances of applications that use NTP are found. Stratam 2 is where distributed debrief applications would be implemented. The accuracy of NTP clock synchronization at stratam 2 is expected to be a few milliseconds or better (Deeths, D. Brunette, G, 2001). Any increase of network latency due to hubs, switches, routers and network traffic will reduce this accuracy. Therefore, the accuracy of NTP over a tightly controlled LAN is well within the bounds of what distributed debrief clock synchronization requires.

Every site would be responsible for maintaining an NTP configuration for clock synchronization. Since, some sites may not initially have GPS receivers, it is reasonable to expect that NTP could be served over the WAN. As long as security policies are accepting, serving NTP over the WAN is a viable solution. However, NTP accuracy across a WAN could range from 10-100 milliseconds (Deeths & Brunette, 2001). Thus, they may still be synchronized, but with much less accuracy. NTP clock accuracy over a WAN still requires further study to assess the effects of accuracy.

The use of NTP is being studied within the U.K. MTDS Royal Air Force (RAF) Waddington facility. It is being used to synchronize systems located in the Exercise Management room including the recording system and white force monitoring and control stations. The Mass Debrief room is also clock synchronized, so

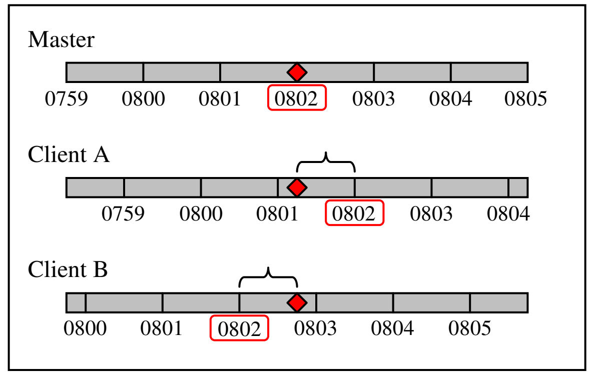

Figure 7 - Detecting Clock Synchronization

that playback of the multiple data recordings will have the high accuracies described above.

While techniques for keeping clock accuracy are important, so is the ability to validate clock accuracy. Detecting accuracy can be easily achieved by correlating the same event captured in multiple timelines. For instance, at the start of recording an exercise or beginning a distributed debrief an event can be sent to all participating recording devices. This event contains the known absolute time from the master and clients reply with their absolute time. Once this message interchange is completed all devices know how far apart their clock synchronization is with the master, and the master knows the global clock synchronization for the group. The difference in clock synchronization can be applied in many ways including a client side adjustment to obtain better synchronization. Additionally, user feedback could be provided to denote clients are out of synchronization. Figure 7 shows how such an event, denoted by the red diamond, can be used to detect client clock synchronization at runtime.

Bandwidth

DDCP uses a fraction of WAN bandwidth compared to any solution based on network retransmission of recorded data. The average bandwidth utilization, Figure 8, shows average bandwidth of increasing scenario size at various playback speeds using a network retransmission approach. The normal play speed, or 1x, is equivalent to what is seen during real-time training. Therefore, this is the amount of bandwidth that is also required for real-time simulation. As Joint and Coalition training scenarios become longer in length the need for utilizing faster play speeds becomes paramount in achieving a debrief. If the bandwidth represented by these curves need to be transmitted over the WAN, a large number of dropped packets will occur resulting in poorly reconstructed visual displays. A possible solution would be to increase the WAN bandwidth to accommodate debrief play speeds. However, incurring the additional cost of increasing WAN bandwidth solely for debrief purposes can not be justified when solutions exist to eliminate the need for such large bandwidth consumption. Instead, an alternate approach is provided by DDCP that does not require the dramatic increase of WAN bandwidth experienced with network retransmission. In fact, actual DDCP bandwidth use is much lower and linear with the number of interacting debrief devices.

VTC systems are used to keep distributed teams in virtual conference rooms together. As the number of participating sites increases, so does the VTC bandwidth requirement. Too often debrief systems rely on a certain amount of WAN bandwidth. The assumption of endless bandwidth is an alarming trend of systems that rely on retransmission of recorded data over the WAN during debrief. As the practical bandwidth limit of a WAN nears, the frequency of dropped packets increases. As packet loss increases, the quality of live video and audio systems decreases. This phenomenon can be overcome through the use of Quality of Service (QoS) solutions. QoS can be found in many off the shelf video and audio solutions,

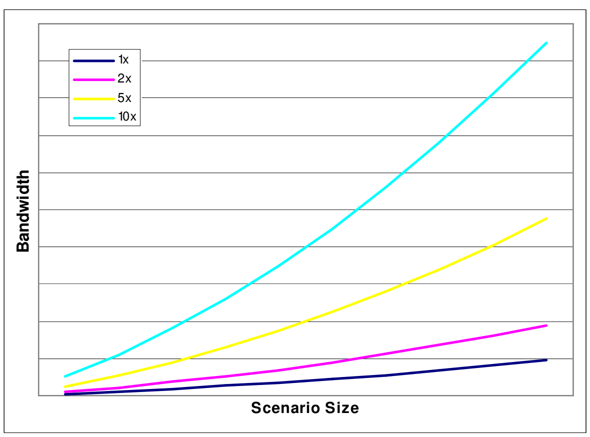

Figure 8 – Average Bandwidth as Playback Speed Increases

particularly those selected for VTC equipment. However, the cost of developing QoS into custom solutions can be very high, complex and potentially incompatible with other QoS solutions. Thus, the solution that is embodied by DDCP is to refrain from using WAN bandwidth for any form of recorded data retransmission. The WAN bandwidth is better utilized by VTC solutions and for keeping distributed sites synchronized while they replay data locally.

Lastly, a growth opportunity exists to maximize both training time and technology investment by sharing WAN bandwidth for concurrent exercises. Concurrent exercises exist when at least two virtual simulation spaces take place at the same real (non-virtual) time, but any two participants from different exercises cannot interact with each other within the virtual space. It is completely reasonable to expect that WAN bandwidths will increase while costs decrease over time. However, instead of filling that bandwidth with unnecessary network retransmission for debrief replay, it could easily be used more effectively for supporting larger scenarios and concurrent exercises. The DDCP approach can help provide the kind of technical solution needed to support future distributed simulation requirements.

Multi-Level Security

Multi-level security continues to be a topic of much discussion (Danner, Muckenhim, Valle, McElveen, Bragdon-Handfield, Colegrove, 2002). While DDCP is not a complete solution to providing multi-level security systems, it can be one component to a solution. In particular, the way in which recorded data is not replayed over the WAN for debrief is a critical basic step to achieving security. While the DDCP protocol still needs to be communicated between sites, it does not inherently transfer simulation data. During event replay, all locally recorded data can stay local and be completely isolated from the WAN. This is starkly different from the network retransmission approach where simulation data must be replayed across the WAN. Consider that each site will have different security policies that require the use of filtering rules during a training exercise. The intent is to prevent participants outside of a secure site from ever observing the classified data. Thus, a secure site may never be able to provide the network retransmission. Additionally, if multiple sites are involved, where each is at a different security level, then the best and possibly only solution is to have every site be responsible for recording data for later playback. It is beyond the scope of this paper to explore or describe how multi-level security systems can function during training exercises. Although, it is believed that one day such a solution may be achievable.

IMPLEMENTATION

This section explores issues related to implementing DDCP concepts in systems and solutions. First, legacy systems are considered and options for upgrading them are discussed. Next, more details on implementing multi-level security are examined. An example of how the U.K. MTDS CCD program is using these distributed debrief techniques is presented last.

Legacy Systems

DDCP does not require the use of common tools and can be integrated into existing tools that are familiar to the warfighter. Adding DDCP support to legacy recording and playback systems may require some effort. However, integrating DDCP into many existing systems is fairly straightforward. In some situations, a new recording and playback system may need to be created.

Consider a site that currently relies on the network retransmission technique for event replay. A solution for this site would involve the introduction of a record and playback system that understands the DDCP protocol. This record and playback system would be responsible for recording DIS or HLA and then later playing it back. This is probably the easiest solution to put in place and certainly the easiest legacy system to upgrade.

Systems that employ a manual synchronization or proprietary control protocol are a bit more difficult to enhance. Any tool that requires manual synchronization for distributed debrief in all likelihood does not have an externally available remote control interface. Thus, tools in that category would require direct upgrades, likely involving the original manufacturer of the system. On the other hand, any system that has an external remote control interface generally falls in the same category as systems that use proprietary control protocols for distributed debrief. It is conjectured that for each unique proprietary control protocol, some sort of bridging software could be developed as the shortest and most cost effective path for supporting DDCP.

Security

Implementing certain security requirements is easier with DDCP than with the network retransmission of data approach. In particular, the way local event replay is coordinated across distributed sites. Utilizing and extending existing simulation network standards with the DDCP concepts is an obvious first choice. For example, DDCP can be layered on top of DIS by placing it in Data PDUs. Although this option may be the simplest, it has a potential security flaw. By keeping DIS or HLA Federations active during debrief, a data spill could occur during debrief playback. The easiest way to prevent and verify local data replay will not accidentally spill onto the WAN is to shut off the data stream through firewall configuration. Recall that with DDCP every site is responsible for recording data for its local playback. That local recording may contain data at a higher classification than approved for WAN transmission. Using trusted firewall technology and configuration may be the most reliable data spill prevention technique, short of complete network isolation. Thus, the easiest way to avoid data spills occurring from retransmission of simulation protocols may be to avoid them entirely as the implementation basis for DDCP.

An alternative is to implement DDCP completely independent of simulation protocols, for instance, using UDP as the transport mechanism. The firewall would be configured to let through these UDP packets, while simultaneously blocking simulation protocols. Having the firewall block simulation protocols provides the option of local event-replay through network retransmission. Therefore, it is recommended that DDCP be implemented as a UDP protocol rather than be transferred over existing simulation network transports as flexibility is maximized.

There are tradeoffs to using UDP for DDCP transport. Specifically, the reliance on trusted firewall technology, validation of proper firewall configuration, and a change in firewall configuration between training and debrief modes become issues. The first two items would presumably be solved through commercial product evaluation, development and certification. The latter point, where a change in firewall configuration is needed, may require additional effort to make truly foolproof. In spite of the technology choices, developing policies and procedures for security accreditation is a pre-requisite for distributed debrief in a multi-level security context.

U.K. MTDS

An intriguing debrief solution can be found in the UK MTDS RAF Waddington facility. The Mass Debrief for most events consists of data playback using 2D plan view and 3D stealth view displays, power-point slides via smart-board, VTC and an analysis display. The analysis tool, the Boeing Analysis for Simulation Environments (BASE) tool, is used to explore objective measures of human and system performance assessment. The interesting capability here is that the assessment is able to be time synchronized to the current playback time. Thus, the current assessment metric values can be viewed for the same instant of time that is being observed in the plan view and stealth displays. What should be noted is the fact that the synchronization protocol was integrated into the tool in a matter of weeks. Thus, it was not difficult to achieve this higher level of functionality because of the simplicity of the protocol.

CONCLUSION

The discussion in this paper has focused on a distributed debrief event replay technique that coordinates multiple site replays through a synchronization protocol. The capabilities of distributed debrief can be greatly enhanced by the techniques described in this paper, including automated distributed synchronization of tools, reducing WAN bandwidth requirements and providing secure event replay. The quality of debrief is similarly enhanced providing the warfighter a familiar, informative, accurate and interoperable debrief environment. Interoperability does not come easily. An open standard for distributed debrief is needed before interoperability can be truly realized.

To meet the needs for future Joint and Coalition training exercises, existing MTCs and other facilities will need enhancement. In order to make wise investments the programs and platforms these facilities service need direction. While existing simulation networking standards like DIS and HLA continue to receive plenty of attention in standards groups, these existing standards are not enough to service the needs of distributed debrief. Thus, in order to provide higher quality and interoperable debrief solutions, a distributed debrief standard must be established.

ACKNOWLEDGEMENTS

The authors would like to thank the U.K. MoD and QinetiQ for their previous work in distributed synchronization protocols. In particular, Paul Henry of Qinetiq was instrumental in this regard. The experiences from the U.K. MTDS CCD program were beneficial in the development of the DDCP protocol.

REFERENCES

E.J. Seeger, C. "Jugs" Jergens, D. "Devo" Devol, and C. "Buck" Owens (2002). Distributed Mission Briefing/Debriefing for Simulated Mission Training. *Proceedings of 2002 Interservice/Industry Training, Simulation, and Education Conference*.

P. Crane, B. Tomlinson, J. Bell (2002). Similarities and Differences in the Implementation of Distributed Mission Training. *Proceedings of 2002 Interservice/Industry Training, Simulation, and Education Conference*.

D. Greschke, E. Mayo, S. Grant (2002). A Complex Synthetic Environment for Real-Time, Distributed Aircrew Training Research. *Proceedings of 2002 Interservice/Industry Training, Simulation, and Education Conference.*

B. Danner, C. Muckenhim, T. Valle, C. McElveen, J. Bragdon-Handfield, A. Colegrove (2002). Multilevel Security Feasibility in the M&S Training Environment. *Proceedings of 2002 Interservice/Industry Training, Simulation, and Education Conference.*

V. Travers, W. Ferguson, T. Langevin (2006). A Federating Protocol for Distributed After Action Review. *Proceedings of 2006 Interservice/Industry Training, Simulation, and Education Conference*.

USNO GPS Time Transfer. Retrieved June 10, 2007, from http://tycho.usno.navy.mil/gpstt.html

P. Dana, (2000). *Global Positioning System Overview.* Retrieved June 10, 2007, from http://www.colorado.edu/geography/gcraft/notes/gps/gps_f.html

D. Deeths, G. Brunette, (2001). *Using NTP to Control and Synchronize System Clocks Part I: Introduction to NTP*. Retrieved June 10, 2007, from http://www.sun.com/blueprints/0701/NTP.pdf

Gehr, S.E.; Schurig, M.; Jacobs, L.; van der Pal, J.; Bennett Jr., W.; Schreiber, B. (2005) Assessing the Training Potential of MTDS in Exercise First Wave. *In The Effectiveness of Modelling and Simulation – From Anecdotal to Substantive Evidence* (pp. 11-1 – 11-16). Meeting Proceedings RTO-MP-MSG-035, Paper 11. Warsaw, Poland: RTO. 2005

Smith, E.; McIntyre, H.; Gehr, S.E.; Schurig, M.; Symons, S.; Schreiber, B.; Bennett Jr., W. (2005) Evaluating the Impacts of Mission Training via Distributed Simulation on Live Exercise Performance: Results from the US/UK "Red Skies" Study. In *The Effectiveness of Modelling and Simulation – From Anecdotal to Substantive Evidence* (pp. 12-1 – 12-10). Meeting Proceedings RTO-MP-MSG-035, Paper 12.