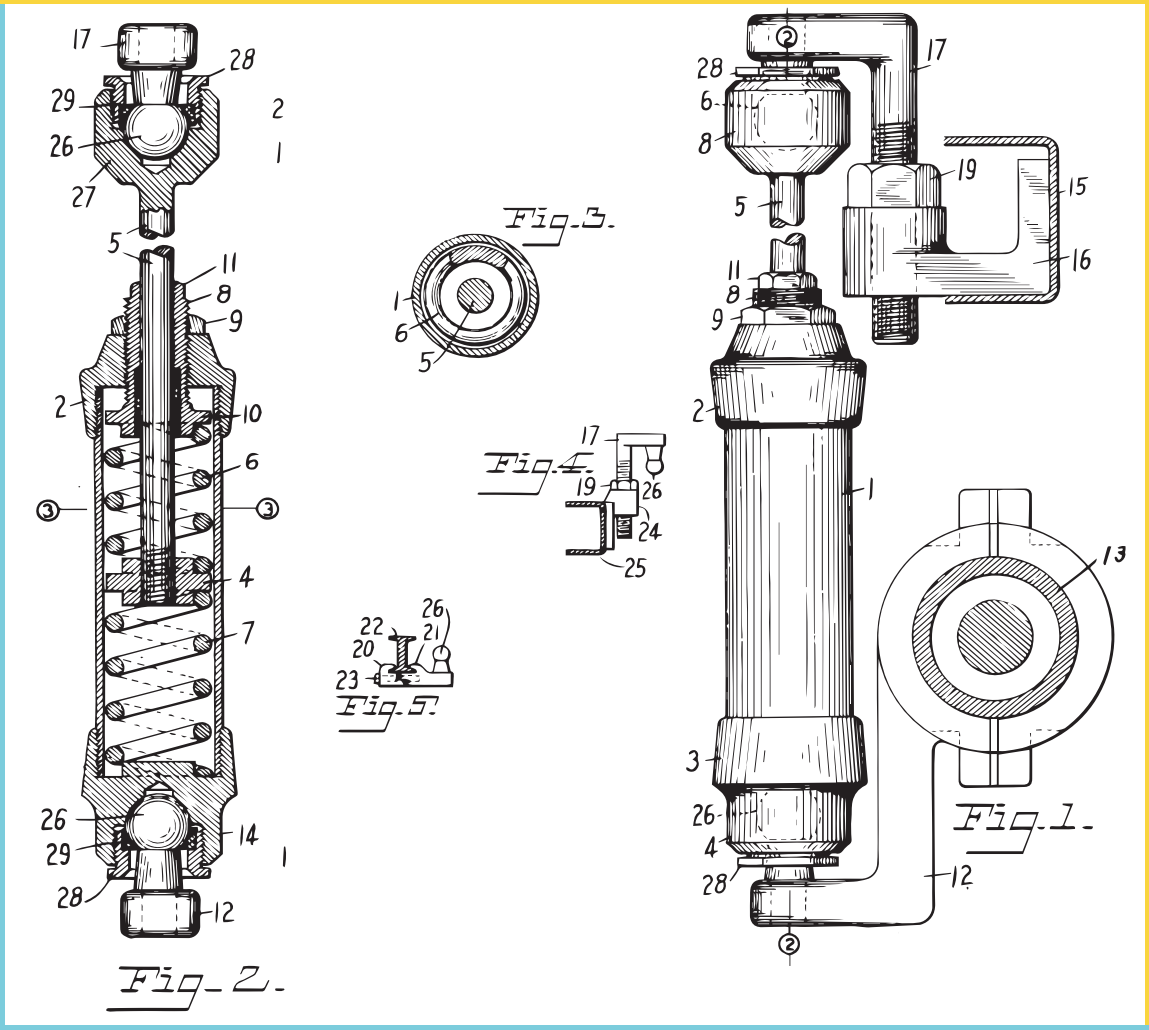
The Ultimate Shock Absorber



AMPS

a real-time streaming analytic database built on a high-throughput, low-latency, expandable replay engine

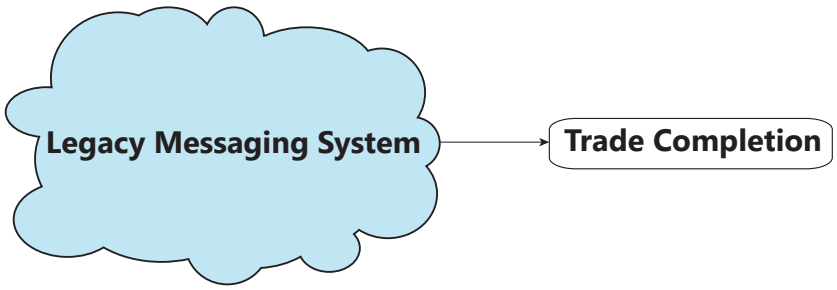
60East TECHNOLOGIES®

60East Technologies, Inc.
sales@crankuptheamps.com
http://www.crankuptheamps.com
(888) 206-1365

AMPS, the Advanced Message Processing System from 60East Technologies, is the fastest real-time streaming analytic database on the planet. Although AMPS is a general purpose middleware solution that applies to many problems, many customers also use AMPS to help buffer legacy systems that have been pushed to the breaking point. AMPS makes it easy to build applications that keep up with the most demanding 24x7 environments in the industry. Find out how in this paper!

The Perils of Success

It's a common problem. You needed a system to handle basic messaging needs. Nothing complicated, just a way to move data efficiently. You analyzed the problem, weighed your options, and you created a messaging solution that solved your problem perfectly.

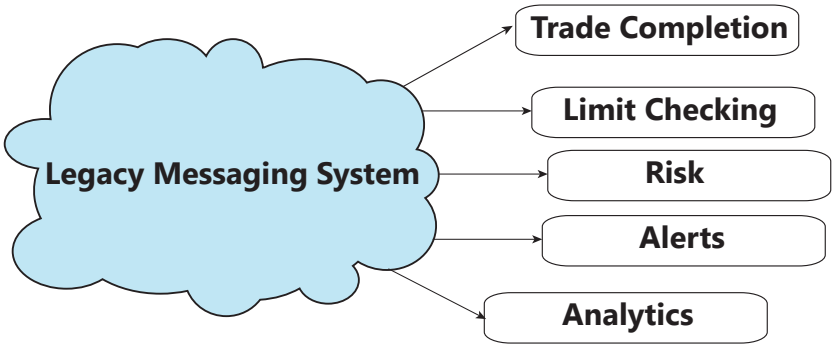


Success breeds demand. The system has been through trial by fire, and it's been updated as the needs of your business changed. More applications have come online, and you've found creative ways to get them connected to your system. The amount of data your system handles has grown. The number of types of data flowing through the system have exploded. You've learned more about messaging, and the solution that seemed perfect when you put it in now seems overly simplified for the problems you're facing.

The demand keeps growing. The system now has to do things it was never designed for, at higher volumes and faster speeds than you ever expected. Now customers want extra features, like persistence, value caching, and on-the-fly enrichment, and they want the system to run faster even with these new capabilities. The elegant system is growing into a tangle of dependencies and brittle assumptions. When something goes wrong, no one is quite sure exactly where to start looking for the problem, or how a fix will affect the rest of the system.

And the demand is still growing. At a certain point, typical applications breach their capacity, or see broadcast storms where the entire network grinds to a halt. Recovery is painful: not only do you have to deal with the technical aspects of recovery, but none of the application

users truly understand how their small, but vitally important, application could possibly be causing the problem.



Your system now needs to cope with applications that are disconnected for long periods of time, that request more data than they need (or can possibly consume), that expect to be able to get a perfect snapshot of current state on demand, or that need to replay a full day's worth of messages. The system may need to support reporting, backtesting, connectivity over a WAN, and mobile devices. None of these uses were part of your original design, but they are now critical for the business.

Eventually, people starting thinking of the solution as the problem. But that's not really true: the solution still works, it's just been overwhelmed by the demands of the business.

What you need is a way to protect the legacy messaging system from the ever-increasing demands of the applications. A way to provide the extra features that clients demand without compromising the existing system.

You need a shock absorber.

60East and AMPS are here to help.

Handling the Bumps

A shock absorber does exactly what the name suggests: the shock absorber is a buffer between two systems, protecting each system from the other and smoothing out the bumps in the road.

In messaging systems, a shock absorber acts as an intermediary between the legacy system and the applications. The shock absorber offloads the demands of the applications from the legacy system, and can also provide additional capabilities for those applications.

A perfect shock absorber must have the following characteristics:

Capacity. The capacity of the shock absorber needs to be higher than the capacity of the legacy system, and allow for future growth. Further, the shock absorber needs to be able to absorb the incoming message stream without impacting the current messaging environment.

Fairness. All consumers, regardless of their capacity, should be allowed to make progress. Slow consumers cannot be allowed to reduce the performance for fast consumers. Likewise, a fast consumer cannot be allowed to monopolize the resources of the system and prevent slow consumers from being served.

Depth. The shock absorber must be able to store messages for consumers that go offline or fall behind during peak load. There must be reasonable limits to prevent resource exhaustion, but it must always be possible for a consumer to catch up if necessary.

Availability. The shock absorber must be highly available to protect against additional risk or downtime.

Performance. Shock absorbers need top-notch performance. Adding an additional system between the consumers and the legacy system can only work if throughput and latency can meet the performance requirements of the system overall.

This is a demanding set of characteristics. But without these characteristics, a shock absorber can easily introduce more problems than it solves.

AMPS: The Perfect Shock Absorber

The Advanced Message Processing System (AMPS) is a world-class messaging system in its own right. The elements that make AMPS the go-to solution for some of the world's largest financial institutions also make it a perfect shock absorber. In fact, some of the largest AMPS deployments started out using AMPS as a shock absorber.

Capacity: Built to Scale

AMPS is designed for high-volume messaging applications. AMPS is NUMA-aware, and built from the ground up to take advantage of modern multicore processors. Even better, AMPS adapts to the environment, using the highest performance code available on the processor and system, and then tunes itself based on the current workload.

AMPS also allows you to minimize network traffic and avoid wasting client CPU through topic and content filtering. With content filtering, each consumer can provide any number of filters, similar to SQL WHERE clauses, that define the messages the consumer will take action on. AMPS sends only the messages that match a filter. There's no longer a need for a consumer to

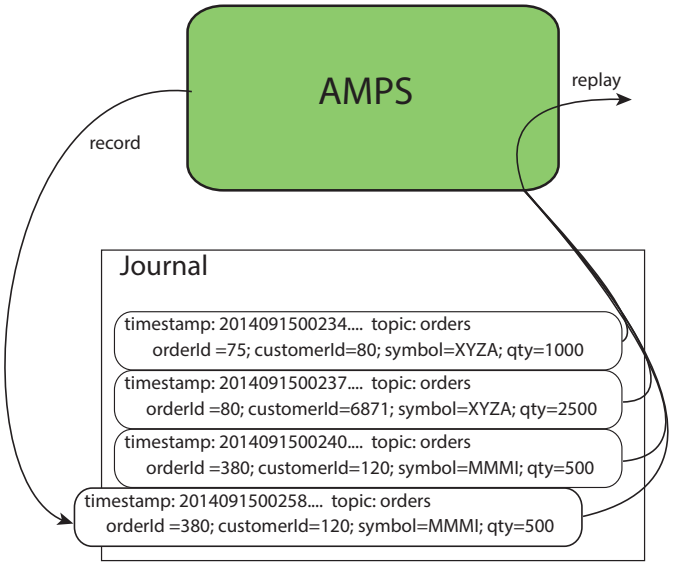
parse hundreds of messages just to find the next message of interest. With content filtering, AMPS delivers only the messages that a consumer needs. Content filtering reduces network traffic to each subscriber, which means that you can do more with your existing network.

AMPS allows publishers to run at full speed regardless of the speed of consumers. Even with the strongest durability guarantees, publishers don't have to wait for subscribers. AMPS, not the publisher, is responsible for final delivery to the subscribers and managing retransmission to slow consumers.

Fairness: Slow Consumer Protection

AMPS has protection to prevent slow consumers from compromising performance for fast consumers, or starving slow consumers to keep up with fast consumers. Each consumer is serviced individually, and AMPS keeps track of how fast the consumer is processing messages. When a consumer falls behind, AMPS can dynamically change which thread is handling the consumer connection to ensure that the slow consumer doesn't impact fast consumers. AMPS delivers messages to both slow and fast consumers at the fastest rate that the individual consumer can handle.

Depth: Message Recorder (Transaction Log)



Beyond buffering messages for slow consumers, AMPS also allows consumers to replay messages from any point in time. AMPS includes a message recorder (also known as the transaction log). The message recorder allows consumers to play back messages starting at any point in time. If a consumer loses connectivity, or cannot keep up with the live stream of messages, the consumer can easily reconnect and start replaying messages from the last message received. As messages arrive, AMPS adds an identifier to each message to eliminate gaps or duplication in the replay. Consumers replay messages from an AMPS-assigned identifier, or from a specific point in time. As with all other message delivery in AMPS, replay is fully filterable by topic and message content.

The message recorder in AMPS is fully configurable. AMPS allows you to record as many topics as you like. Built in file management lets you set a policy for compressing and archiving the log files. AMPS allows you to maintain more recent, active files on fast storage, and archive older files on less-expensive, slower storage -- while still allowing replay from archived and compressed files. This file management can also automatically delete files older than a certain

threshold on a regular schedule.

The message recorder gives you as much depth as your storage system allows, fully managed by AMPS, and online and available for any consumer that needs it.

Availability: Replication and Failover

A shock absorber can't compromise the availability of your system. AMPS provides integrated replication and failover to make sure that your application is always available.

Replication allows a message published to one instance of AMPS to be replicated out to any number of AMPS instances, with configurable durability guarantees. The client libraries support durable publication, so you can configure AMPS to enforce strong guarantees about message delivery. Each message has a defined state and is persisted in one or more places. The publisher knows when AMPS has persisted the message, so there's no guesswork or opportunity for message loss.

The client libraries for AMPS include automatic failover, with the ability to configure any number of equivalent failover servers. The interfaces for automatic failover are open, so you can write your own failover (or load-balancing) logic if you choose.

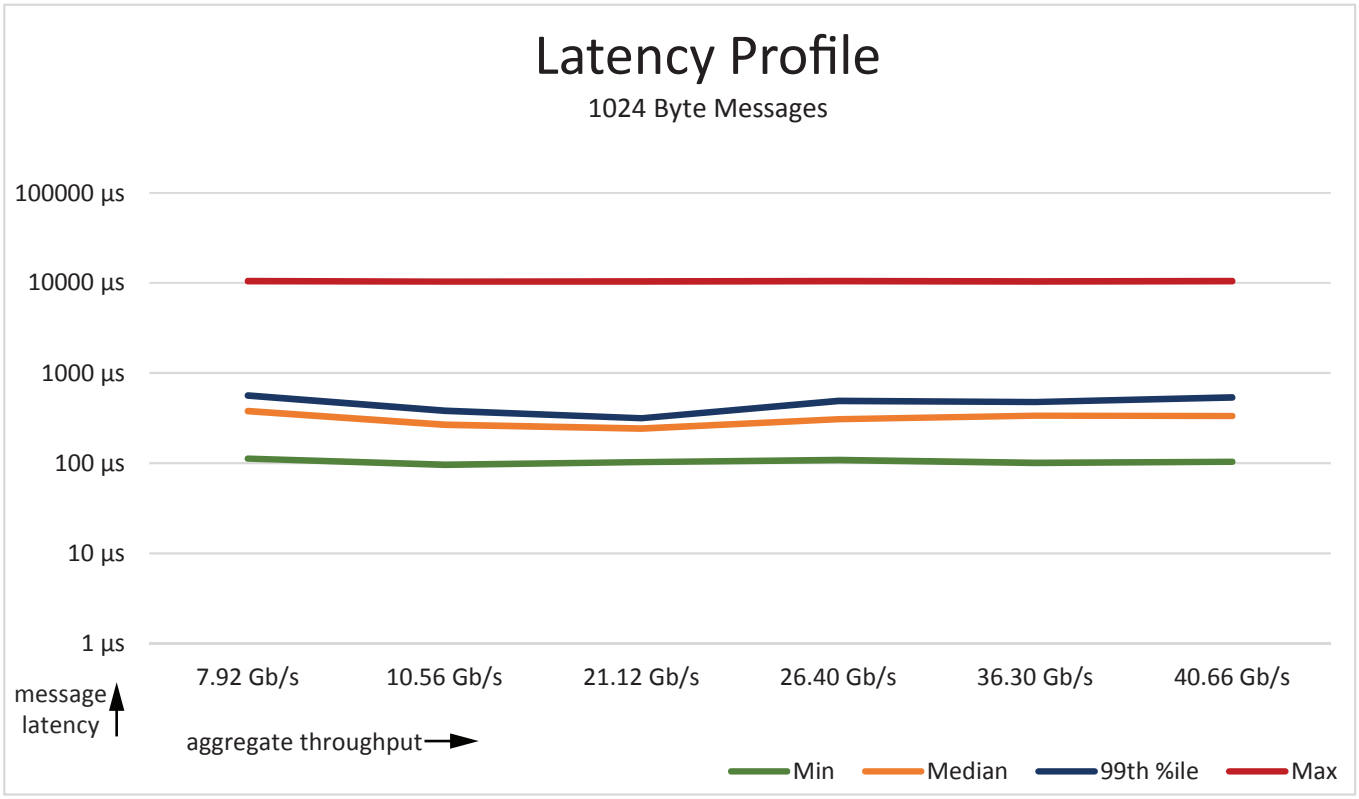
Performance: It All Comes Down to This

None of the other characteristics matter if the shock absorber doesn't give you blazing fast performance. AMPS delivers raw performance that easily exceeds popular hardware-based solutions.

To demonstrate the performance you can expect from AMPS, 60East Technologies created a test that mirrors one of the ways AMPS is used as a shock absorber in production. A publisher sends 1024-byte messages to AMPS, which fans out the messages to a variable number of subscribers. The tests were executed on an IBM 3750 with the following specifications:

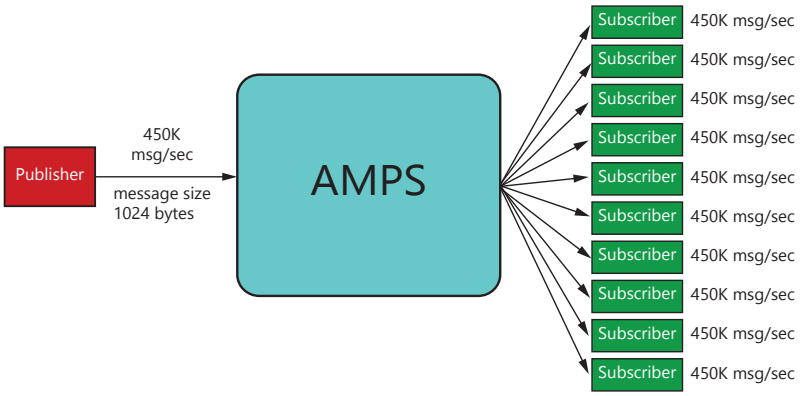
IBM 3750	
CPU	Intel(R) Xeon(R) CPU E5-4627 v2 @ 3.30GHz
Memory	128GB
OS	CentOS 6.5
Storage	Virident FlashMax II PCIe 4800GB
Network	Mellanox Technologies MT27500 Family [ConnectX-3] (40Gb)

The following chart shows the AMPS latency profile for publishing messages through AMPS. This test measures end-to-end message latency. A simple fan-out application was created where a publisher sends 1024 byte messages to subscribers. In this test, the publisher was set to constrain its throughput so as to avoid saturating the network. The publisher was set to ingress rates **ranging from 375,000 messages a second up to 450,000 messages per second**: with fan-out, AMPS was processing **4.6 million 1024-byte messages per second** at the point the network reached saturation. Most importantly, even at network saturation, there was no pushback from AMPS itself on the publisher.



As the number of subscribers increases, the bandwidth consumed increases until AMPS effectively saturates a 40Gbit network, yet end-to-end latency remains consistent.

What this means is that AMPS received 450,000 messages per second from a publisher, each message containing 1024 bytes of data. AMPS found the matching subscriptions for those messages, routed the messages, and distributed all of those messages to 10 subscribers, for 4,500,000 million messages a second egress. Despite the number of messages, the end to end latency was less than a millisecond for 99% of all messages, and approximately 10 milliseconds in the worst case.



AMPS consistently posts this type of predictable performance, which remains roughly linear until a component of the system is saturated. Although there is no way to add a shock absorber to a system without adding some latency, AMPS adds minimal latency and performs predictably, even at the most demanding throughput rates.

In fact, when applications use content filtering and advanced messaging features, introducing AMPS as a shock absorber can save time in the consumer, and reduce the time from message publication to the point at which a consumer can take action on the message.

Beyond the Shock Absorber

As described above, the capabilities of AMPS make it a perfect shock absorber. Those capabilities also make AMPS a great standalone messaging system. But that's not all there is to AMPS.

Scale Up or Down

AMPS scales seamlessly. From virtual machines running on a single-socket developer machine to the largest enterprise-class servers available, from WLAN connections to 100+Gbps networks, from spinning disks in a laptop to the most advanced flash storage solutions, AMPS scales both up and down.

AMPS is a pure software solution that runs on any modern, x64 Linux distribution. That means that AMPS can adapt to the capabilities of the system and get every bit of available performance. Deploying AMPS is as simple as installing an executable and editing a configuration file. There is no need for a large hardware investment to get started with AMPS, and as more hardware capability is added, AMPS pays you back with more raw performance. Since AMPS is built to scale as hardware scales, new generations of hardware provide immediate performance upgrades to AMPS, which means that an existing hardware refresh schedule also provides capacity and performance boosts.

Because AMPS doesn't require exotic hardware, it's easy to create development and test environments. And that means you can test thoroughly, without artificial constraints, to ensure that deployments go smoothly.

Next-Generation Messaging Capabilities

AMPS pushes the state of the art for messaging systems, making it easy to build high performance applications.

AMPS supports any message type you need. Message types are plugins to the AMPS server, with

plugins for FIX, NVFIX, XML, JSON and BSON included. The AMPS Message Type API allows you to write server plugins to handle any other message type required.

AMPS offers state-of-the art client libraries in Java, Python, C#, C++ and C (with others under development). These libraries are designed to be easy to use and highly performant. This means that your applications don't have to rely on a particular language or development environment.

The State of the World (SOW) is a current-value cache for a given topic. AMPS maintains the current state of messages and allows clients to easily query the current state with the same SQL-92 expressions used for content filtering. Consumers can also retrieve the current state of the world and subscribe to updates in a single atomic operation, without message loss or duplication.

With a State-of-the-World database, AMPS supports incremental update of messages, called delta messaging. Delta messages also allow publishers to send only the parts of a message that need to be updated, and for subscribers to receive only the fields that have changed. This saves bandwidth, and also helps ensure that consumers only receive actionable data.

AMPS allows you to aggregate and perform analytics over messages. AMPS views provide the most commonly used analytic functions, and work with the SOW capability to provide analytics that are always up to date with the latest messages.

Views aren't limited to a single topic or message type. Combining customer data published in XML with orders published in JSON and pricing data published in FIX in real time as messages are published is easy to do with AMPS.

The short descriptions here just scratch the surface of the capabilities of AMPS. Each of these capabilities is designed to make it easier to build and deploy performance-critical messaging applications.

Try It Yourself

The technologists at 60East believe that the best way for you to understand AMPS is to try it out yourself. This paper provides an overview of how AMPS works as a perfect shock absorber for legacy systems, and introduces some of the unique next-generation capabilities that make it easy to build messaging applications.

You've read about what AMPS can do. But don't take our word for it. Give AMPS a try. Put it through its paces. The engineering in AMPS speaks for itself.

You can download a free evaluation version from **http:/crankuptheamps.com/evaluate**, or get in touch with 60East at **info@crankuptheamps.com**.