1 Executive Summary

The Reality Warp Software (RWS) team brings personal experience and lessons-learned from architecting and developing the DCGS-Army's current fusion framework and the Army's first cloud-enabled, mobile, biometric platform. This personal experience gained by Reality Warp Software's founder and President as the Chief Architect for these prior solutions provides a unique perspective and a promising initial direction to Reality Warp Software in providing distributed, scalable frameworks for Big Data analytics. These experiences have lead to the architecting and development of a different kind of technology in the Pirkolator that allows RWS to face the challenges of a complex, multi-modal enterprise such as the one that the Air Force represents with its many and varied sister agencies.

Reality Warp Software has developed several tools that provide us the capability to rapidly developing analytics. We have built the Pirkolator, which provides a distributed, scalable and extensible communication infrastructure that is the foundation for analytics. We have built in support for data ingest that supports many formats, such as JSON, compression techniques, and security protocols, such as PGP. We provide built-in support to ingest from a variety of different data source types from message queues using AMQP to cloud sources like Amazon's Web Services to persistent stores like relational databases and NoSQL stores. The infrastructure provides a transformation library that can automatically transform ingested data into other forms. We provide an abstracted data layer component that is used to provide access, search, query, and changes to specific data source implementations. Using this infrastructure, we built the Determinator to provide a computation engine for parallel processing targeted for use with association, correlation, and aggregation analytics. We built the Trendinator to provide an engine and tools for managing analytics work with data in series, such as over time. We have built a distributed management capability to load target data into these engines to prevent duplicate computations and enforce concurrent access when analytics work with dynamic, changing data. We use these tools to explore new ways to think about data.

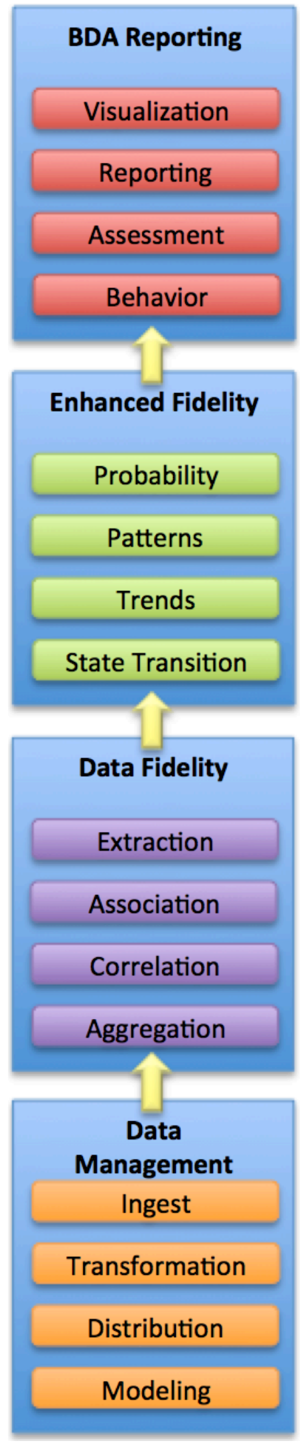

Figure 1: Steps to BDA Reporting

Reality Warp Software's approach to discovery and analytics is a 3-step solution. Our first step is data modeling and acquisition. We integrate the disparate systems, services, and data sources using our distributed, scalable architecture. Our next step is to provide BDA extraction, meta-data association during data ingests, and the Determinator engine to provide highly resolved, temporally and geospatially defined BDA elements. The final step in our approach is to use the highly resolved data to feed our Trendinator engine and other analytics to generate BDA products with data fidelity based on available resources and the operating environment. For this opportunity, RWS will leverage existing tools to allow us to focus on network forensics, generation, and delivery of BDA products to the EWO and required systems.

The Phase I opportunity under this topic will give the RWS team the chance to explore using our infrastructure, engines, and tool kits for Electronic Warfare Battle Damage Assessment. We will develop a data model for generation of both low- and high-fidelity BDA reports. We will develop this data model to support existing BDA information and modeling for digital network forensics. We will integrate existing data from current EW data stores, such as from the SKR and DDS, as needed for generation of higher fidelity DBA reports. As part of our Phase I delivery we will provide infrastructure and visualizations for delivering the generated BDA reports to the EWO, analysts, and required systems. Given the opportunity to continue under the Phase I Option, we will use this time to work with the customer to solidify the data sources needed, further develop the visualizations to support the EWO, and perform additional analysis and testing under various operating environments.

2 Technical Approach

Current BDA by EWOs is remarkably limited due the absence of the Army from non-IED focused electronic warfare. EWOs are able to reach out to aviation assets as well as Joint assets to schedule EW operations. In doing so they can request BDA for each operation/mission. The return typically is a simple narrative of the relative success of the EW attack, such as effective and not effective. Within the IED venue, the simplest BDA is a negative attack report, but this is only valuable given the knowledge of and actual attempt to detonate a device. This collection occurs within the SIGINT arena and can provide the EWO with a direct Attack Failed BDA report.

This alludes to the overall spectrum knowledge capabilities of the EWO and the ability to search, use, and monitor the spectrum repository with respect to the actual, unfolding operation. The differences noted in both SIGINT collection (real from what was expected) and in electronic profiles within the given force structure, emitter characteristics and in spectrum profiles and usage can be a valuable tool to determining the relative effect of EW operations. It can further provide a tip as to emerging targets and variation win TTPs being used by the enemy. This knowledge will be useful for both EW and ES operations.

2.1 Data Fidelity

There are two fundamental types of data potentially available in the EW environment. Directly observable data is that which allows an EW agent to assess the RF spectrum and collect raw data. This raw data can then be analyzed to identify actors that are operating within the environment. Ancillary observable data requires being able to observe the data path from the actors being targeted and their communication destination. This is typically achieved through implants on devices or having control of the communications medium network that the targeted actors are leveraging. For Phase I, the RWS team will develop a model for representing both types of data. We will draw up our knowledge and provided knowledge of existing data sources, such as the Spectrum Knowledge Repository (SKR) and the DCGS-A Distributed Data Service (DDS).

There are a number of mechanisms that can be employed for real time BDA. These can be static analysis or dynamic analysis of impact on communications by EA. Static analysis will inherently apply only for a specific subset of devices (e.g. cellular and Wi-Fi.). The static analysis data sets can be referenced in the field with environment assessment of EW targets such as range, ambient noise, competing clients in a shared resource setting, etc. that creates a reference impact for specific communication systems. Static analysis is inherently low fidelity as it assumes the EW target adheres to previously tested communications equipment. Dynamic analysis can be applied and involves coordination of the EA and the BDA to identify cause-and-effect relationships. For example, jamming can be performed and then cease for a moment to gauge if clients attempt handshakes to reestablish their communication channel. Other dynamic analysis can include digital network testing to evaluate immediate impact of an action. Speed or bandwidth tests can be applied to a cellular tower to determine if its performance with clients is degraded or interrupted. These dynamic tests provide better fidelity than static analysis, but cannot compare to near-real time data. Near real time data involves being able to measure data over a length of time and analyzing it so that answers can be derived through correlation or directly by the data. This type of data may take longer to generate due to resource constraints or other environmental conditions. For Phase I, the RWS team will develop a model that can be used during our exploration of static and dynamic analysis. We anticipate this model to include existing elements of BDA for current EW data models and new modeling for network forensics.

2.1.1 Wi-Fi and Cellular Network Modeling

In a wireless communication networks, we can analyze the digital network traffic to generate a BDA. In analog you only really get transmit power, directionality, and ambient RF floor. You could generically assess the RF spectrum and identify clients trying to re-connect, more sync patterns being transmitted, and less dead transmit time. Digital links, though, are where interesting network analysis can take place. Modern cellular has digital links (GSM, CDMA, WiMAX, LTE), but a device is needed that can read those digital links. Unlike 802.11 Wi-Fi that uses an open standard, digital links use proprietary standards. We will explore the potential to bypass the issue with a modern phone baseband that does it all and provides the data link. There would be an anticipated lose in assessment using this technique that we will explore as well. I 'm sure some of the baseband offer signal strength negotiation and fun timing synchronization that could be messed with to gauge impact.

The state of the clients on the network and our ability to communicate through networks enable us to create the BDA. Common wireless networks today establish digital network connectivity, usually through ATM- or IP-based connection. These digital networks provide an immense amount of information that can be leveraged for identifying actors actively leveraging them, but also the state of the actor on the network. Initial handshakes to establish a digital link occur between the base station and the station clients. This link is only directly observable. After the link is established, a network link is then established which usually requires registration on the network being created. Such registration can be as simple as communication with a DHCP server on a Wi-Fi network to a Radius or similar server on a Cellular network. Once registration completes, the actors are able to communicate within the network as their permissions allot. This typically allows unrestricted access to the Internet. At this point, the following targets allow network-based analysis:

- Base station access enables direct actor link connectivity status
- Registration server access enables actor registration status
- First hop or closer router access enables actor data flow state

If we have a data recording agent on one of the above network targets, we can inner the clients access our state in the network. If we can join the base station network, asking we are in the same class of service, we can test the network connectivity we are permitted and determine the impact for an example a denial of service (DOS) would have on the EW target communicating on the network. To clarify, if we use a performance monitoring tool, such as Iperf, to acquire our real time bandwidth with a base station, we can then determine how much bandwidth DOS would take from the EW target such that when the DOS is active, we can identify the amount if bandwidth remaining at their disposal. Since networks can leverage quality of service guarantees in the links, the fidelity of this specific analysis would be mid level. With near real time network analysis, we can track data usage over the base station by the client and acquire high fidelity BDA. Some attacks are plain hard to monitor, because it's degradation in quality of service, which would require knowing initial capacities to register a change. And what if an attack succeeds at destroying half of capacity but only 10% capacity ever really gets used. Having to identify actual impact to existing communications, not just what comms could be, is a difficult problem. The RWS team will explore in Phase I the varied ways of measuring assessment using the identified techniques. We will use this exploration to develop the model for network forensics and to discover existing tools and hardware that might be needed to provide the assessment.

2.1.2 Low-fidelity Modeling

Due to the inherent Electromagnetic (EM) issues with EW jamming, raw RF data is inherently going to be difficult to identify targeted actors and identify their communication state with their desired partners. For example, a suitable EW jammer may be able to cycle on for five-second periods to disrupt targeted actors sufficiently to require the actors to reestablish their communication channel. In the case of a cellular network, the handshake time and fact that clients are reestablishing communication would indicate that an EW attack succeeded in disrupting direct connectivity. However, this is a near-real time solution that provides higher fidelity as the EW jamming can be discontinued to get cleaner raw RF data. Approaching real time will require that the spectral analysis to identify actors and their communications be applied across a noisy RF signal, and will suffer the same issues as the radios of the actors themselves, that is difficulty in identifying signal from noise in the RF.

In constrained bandwidth environments, RWS proposes to explore 3 techniques – compression, optimized data modeling, and processing at the resource for low-fidelity BDA report generation. For high priority information that is large, we will explore automatic compression techniques such as ZIP and TAR processes to compact the data before sending. We will explore utilizing the technique of “analytic at the sensor” to process data before transmitting reports. This aggregation will allow for synthesis of data and streamline the potential volumes of duplicate or near duplicate information. Lastly, we will explore creation and use of a data model that only reports the absolute minimum elements needed for accurate BDA. We will also explore utilization of a dynamic data model that enable adjustment of what is delivered based on configurable elements. The configurable elements will allow the system to increase or decrease delivered elements of a high-fidelity or “complete” BDA report based on elements such as amount of bandwidth available, network connectivity, or CPU resources available. The same data model can also be used in generation of high-fidelity BDA reports.

2.1.3 High-fidelity Modeling

In more optimal, bandwidth- and resource-enabled environments, higher fidelity BDA reporting can be exploited to provide a more accurate operational assessment. This will allow for use of more granular, greater density data and more complex analytics for BDA generation.

Integration of information from other data sources for higher-fidelity assessment can also be used. Under both conditions of low- and high-fidelity BDA reporting, RWS will explore correlation and aggregation techniques to streamline BDA report in large volume environments. Additionally, we will explore a timeliness factor to allow for delivering data in intervals. There are some conditions where BDA reports will only need to be generated at some time increment, such as once every 30 seconds or one the hour. In an environment where external data can be leveraged, direct correlations can be drawn between EW actions and actor communications.

2.1.4 Lethality Estimation

Lethality Estimation needs a means of recording the state of the target after an EW action has occurred. Recommended levels should be simple and reflect the environment accurately.

- Degraded
  - Limited Range
  - Limited Transmission Speed
- Disrupted
  - Unavailable communication link
- Obliterated
  - Inoperable device

The RWS team will incorporate the definition of lethality estimation within the data models generated and delivery BDA reports that reflect this measure.

2.2 Autonomous Processing Platform

For this opportunity, Reality Warp Software will leverage open source applications and tools to provide a self-contained computation platform for generation of BDA reports. While being an autonomous platform, interactions with other systems and processes will be required to produce accurate BDA generation and reporting. Additionally, RWS recognizes that this platform will need to interact or be contained within existing EW tools and platforms, such as with EWPMT and MFEW systems. While Phase I does not require such a platform, we believe that implementation of such a platform from the start will increase the effectiveness and timeliness in our research and viability testing of digital and wireless BDA report processes. Additionally, RWS has already developed a platform that adheres to open, modular architecture and capability and can provide this infrastructure at no cost or development time for this effort. We have used our past experience to build an infrastructure capable of supporting the needs for the future EW processes and platforms and specifically for the EWPMT program.

To support the effort of researching and developing a state-of-the-art BDA solution, there are several supporting fundamentals that make this process much easier and faster. In the following sections, we will describe each supporting capability or tool as data moves from the

Benefits of Reality Warp Software's Architecture
• **Adaptability for mission modifications as threats change**
• **Interoperability to leverage all aspects of enterprise for developing mission strategies**
• **Extensibility to adapt to new sensors, data, and services**
• **Flexibility to build on past mission successes to formulate new missions plans with best options**
• **Ease of use for last minute modifications in dynamic, evolving scenarios**
• **Distributed to connect all available resources**
• **Scalable for optimized analytics and resource utilization**

originating source to distributed analytics and then being represented as views to analysts.

2.2.1 Infrastructure

Reality Warp Software has developed an open source, distributed analytic infrastructure, illustrated in Figure 2, that provides the system platform and key tools enabling distributed data processing. This infrastructure enables analytic developers by providing access to real-time and persistent data streams, efficient use of dynamic resources, and dissemination of results to an ever changing and evolving community of consumers. The infrastructure is architected around the publication/subscription pattern to establish a communication and data distribution platform that is tailored to processing real-time data streams, persistent SQL and NoSQL data stores. We recognize the ever-changing data landscape and provide analytic writers with a stable way to access data without worrying about the changes to the underlying data sources. The infrastructure addresses this by providing an abstracted data access layer that analytic developers use to access required data. The infrastructure inherently supports several architectural patterns for analytic execution. The distributed consumer pattern is used to provide the same data to analytics executing in parallel. The competing consumer pattern is used to execute many of the same types of analytics against streams of data in parallel. The event-driven consumer pattern is used to execute analytics when events and other stimuli occur. For this opportunity, we will leverage the benefits of this existing infrastructure for easier and timelier research and analysis of the efficiency of data processing and analytic results. Additionally, by leveraging this infrastructure, RWS will be able to more rapidly develop, prototype, and test analytics.

In building this infrastructure, we leveraged the experience gained from architecting and building the fusion infrastructure currently integrated with the DCGS-Army program. This insight and experience paved the way for the creation of an infrastructure that is tailored to running analytics under a variety of conditions. By leveraging open source and open standards, we have built-in compatibility with this and other standards-based infrastructures. We built a publication and subscription infrastructure that is both scalable and distributed. We utilize an open source application, Hazelcast, to provide one mechanism for distributed pub/sub, but the infrastructure is not limited to just Hazelcast. The Pirkolator is able to incorporate additional distributed pub/sub applications, such as Redis or Terracota, to provide the concept of a “Hub” for any pub/sub need. We incorporated the open source data transfer application, Apache Camel, to provide access to the most common network protocols, data formats, and translations. The benefit of this approach is more consistent, timely access to data and greater assurance of data integrity. For this effort, we will leverage the flexibility of this environment to incorporate existing open source analytics, tools, or other applications that will enhance our research and development of new techniques for digital network BDA.

2.2.2 Data Ingest and Extraction

Data streams and data ingest is provide through another open source application from Apache called Camel. This product provides us with an extensive range of data protocols and formats, as well as transformation services to position the data in an optimized way for analytics. We utilize Spring Data, an open-source platform, to provide an extensible, flexible, and abstracted data management technology. This technology allows us to access to out-of-the-box implementations for several data modeling standards, such as Relational Database Management Systems (RDBMS), Big Data, and Key-Value Stores. Custom repositories allow us to develop for any data source need. Using an abstract data access layer coupled with the open source Spring Data technology, allows analytics to focus on their job without worrying about changing data source environments. This means no code re-writes should a data source or even data format change.

2.2.3 Data Modeling

Data modeling provides for describing data in a digital form. From the personal experience of the Reality Warp Software team trying to provide a common model to describe everything is not feasible. We believe that a better, more efficient and extensible solution is to develop models that a tailored for how the data will be used. Analytics have different modeling needs than persistent storage and a signals processing model has different needs than textual processing models. A common data model is used in cases where different systems need to work together to share data in which case the model is developed around the common model attributes.

Using multiple data models, tailored to specific needs also increases the extensibility as new data types and models becomes available. This flexibility manifests through addition of new fields to an existing model and through addition of new data models working in concert. Some data models are not extensible in which case new data models must be developed to work in concert.

For RWS, data modeling's most important benefit is in positioning data for optimal utilization. By using tailored data models and architecture of extensibility, we are able to position data for processing even as the data model changes. To implement this strategy, though, data modeling is not the only function needed. The addition of transformation techniques combined with our thinking about data modeling allows us a novel way to position data that can be leveraged across systems and under a variety of analytic processing.

2.2.4 Data Transformation

Transformation services ensure that data is mapped and translated into the optimized format used by each analytic. Filtering methodologies built into the publication/subscription construct provides even further data refinement for analytics. Geospatial, temporal, and source-based filters provide analytics with data granularity, reducing the size of the data haystack. Custom filtering is provided to allow each analytic with data filtering tailored specifically to its needs. Our transformation services utilize the open source technology, Apache Camel, to provide network connectivity, data routing and common data format transformation support, such as for JSON and XML. We provide additional hooks into this data transfer framework to provide dynamic mapping and transformation of incoming and outgoing data to support any object model needed. The benefits of this approach are extensibility, reusability, and flexibility. As the data sources within the community change, extensibility within the infrastructure provides new transformation services that are automatically integrated without affecting analytics or other data-aware processes. We are able to reuse existing data mappings and transformation services where needed and the modular nature of new transformation services allow portability to other systems outside of the systems integration effort. The flexibility of the transformation services within the infrastructure allows for analytics to get the data in the format needed without having to understand the underlying ingest or originating format of the data.

2.2.5 Data Management

The data management component provides the necessary architecture for plugging in a variety of data stores by abstracting away the details of data management. We utilize open source technologies, such as Hibernate and iBatis, to provide data modeling, persistence, and query capabilities to structured RDBMS and Object-Relational Model (ORM) schemas. Unstructured data or large amounts of data are stored in a NoSQL data stores, such as Mongo DB, Accumulo and HDFS. It is important to allow for the integration of Big Data solutions and to provide the ability to run analytics over massive amounts of data in near real time. Multiple data source solutions are optimized for storing different types of data. With the ability to integrate these disparate data sources, analytics are optimized when performing complex operations specific to a data source, such as entity correlation and complex event detection.

Our infrastructure provides integrated metrics, pedigree, and data integrity. Metrics lend support to the systems administration command to maintain a healthy platform. Pedigree is integrated with the infrastructure to allow tracking of the processes that change or affect data. This builds upon the confidence, accuracy, and other data integrity definitions that help establish a reliable data tree of origination for each piece of original data, extracted entity, or transformed data. Having this tightly coupled relationship between pedigree and analytics promotes a strong chain of authentication when data verification is needed.

Our strategy to data within the infrastructure provides several benefits. First, it allows for dynamic data sources. As data sources come and go online, analytics don't have to change. Transformation services ensure that data is in the format needed for an analytic. Pedigree built into the fusion infrastructure further ensures data integrity, insight, confidence, accuracy, and traceability. With cloud technology and cheap compute cycles, transformation services allow data to take many forms and support many models. If you need to map data from one data model to another, transformation services provide an easy mechanism to achieve this goal. Every analytic can have its own optimized data model while at the same time support a common data model that is used across enterprises.

2.2.6 Distributed Analytics

On top of these open source applications, RWS provides easy to use architectural patterns that allow other processes, such as trending analytics or map/reduce functions, to get the data, events, or manual stimuli needed for a solution. We employ new techniques and technologies to achieve the near real-time analytics over Big Data sources that are the first step in developing a real-time, latent relationship discovery capability. We employ an in-memory data grid that consists of Java Virtual Machines (JVMs) running on any number of servers in the data center. An in-memory data grid is a distributed memory based data store that allows the data to be distributed across many servers. This is backed by disk storage with a write behind scheme for permanent storage. The benefit of this technique is to allow fast access to large amounts of data, even under systems with load constraints, while still providing the needed streams for real-time analytics. The in-memory data grid can be implemented with off-heap storage to utilize larger amounts of Random Access Memory than is allocated for the JVM, and to avoid the penalty of the garbage collection when items are deleted. This new technique to in-memory, distributed data affords the capability to explore computational analytics against real-time data streams for producing real-time insights.

New technologies allow us to provide parallel, concurrent processing, in-memory and distributed. This can be much faster than traditional disk-based Hadoop/HDFS infrastructures. Additionally, concurrent technologies will promote new thinking. From past experience, a new way of thinking about an analytic was identified as an economy of scale for a CERDEC analytic. The Communication Effects Simulator (CES) was identified as an important analytic, but was slow due to being a Windows command-line tool that must be manually run. To achieve benefits, it was possible to run the analytic in parallel and then combine the results. This allowed for a 40% improvement in working with the analytic.

For legacy analytics that are Hadoop-based, we can leverage the results in the same way we distribute data by taking the output of a map/reduce job and sending it to where it needs to go. We use a persistent cache within our in-memory infrastructure to provide fail-safe capability in the case of system failures. Additionally, we are able to leverage large disk-based systems, just as if they were in-memory sources when the amount of data is too large to keep in memory. This type of in-memory, distributed processing is at the heart of where we think future processing platforms are heading. We are integrated with another open-source application called Spring XD. This platform provides a "unified, distributed, and extensible system for data ingestion, real time analytics, batch processing, and data export". It provides out-of-the-box access to Hadoop operations, such as Map/Reduce and HDFS. This is just another example of a tool that is easily integrated into the infrastructure that keeps us at the cutting edge of analytic development.

Trending analytics provide one of the two bases for exploration in this proposal. The infrastructure, data modeling, and other analytics support this type of analytic. Trending analytics allows for collecting data to allow inference discovery across groups of data. Aggregation and correlation are fundamental properties of trending analytics. We propose to use our existing Determinator engine, made up of a combination of aggregation and correlation analytics, to develop trends that will be useful in BDA reports. These trends will contribute to the fidelity of the data being reported in the context of a BDA report for digital networks being explored in Phase I of this effort.

2.2.7 Analytic Engines

Reality Warp Software's Determinator engine is a capability we developed to allow algorithms for Entity, Event and Relationship (EER) extractions to combine with Association, Correlation, and Aggregation derivations to provide more resolved entities. Our engine utilizes the distributed and scalable nature of the Pirkolator's infrastructure to provide a real-time, dynamic scoring engine. Algorithms are developed and added to the engine using the same publication and subscription filtering in the Pirkolator to provide the most granular data available. An extensible scoring construct is used to provide the results that can then be used for automatic threshold alerting or other post-calculation analytics. Additionally, results are automatically distributed to any interested analytic that is attached to the infrastructure. Lastly, results can be automatically persisted to any attached, disk-based data store to provide further analysis using functions such as Map/Reduce. The Determinator engine provides a fast, extensible, and scalable resource that gives developers a fast and easy start to providing EER extractions, associations, correlations, and aggregations.

Reality Warp Software's Trendinator engine is another capability we developed in conjunction with our infrastructure and Determinator engine to provide the grouping and threshold triggering useful during trend identification. Behaviorally, the Trendinator engine functions much the same as the Determinator engine with algorithms being added to the engine to provide scoring against any input data provided through the Pirkolator. The scoring results are run against threshold constraints and post-calculation analytics to provide automatic alerting. The Trendinator engine provides an additional function to the analytic developer through a grouping construct that can be used to build and establish related trend data. The grouping construct provides temporal and geo-spatial groups coupled with other custom, trend-specific constraints, such as area of interest or time range limits, to automatically keep only the data elements that fit the trend grouping. The grouping constructs utilize the in-memory, distributed data grid of the Pirkolator for fast and efficient storage and retrieval of the data elements. The

Trendinator engine provides developers with an immediate solution to quickly providing trend results against real-time data streams utilizing the available compute resources.

2.2.8 Analysis Visualization

Reality Warp Software leverages the Model-View-Controller architecture pattern to provide visualizations to the analyst. With our Transformation Services we are able to deliver representations of data in a format that can be easily viewed in any visualization platform, such as in browsers or standalone GUI applications, such as NASA World Wind. As data is produced within the Pirkolator, the Transformation library provides automatic conversion of the data into any view that may be in use. The Transformation library can also be used to manually transform produced data into view representations as needed. This strategy allows an extensible mechanism to provide data to multiple, different visualization platforms where required. As new ways to think about and visualize data are developed, transform modules can be added to support the new perspective.

2.2.9 Client Agents

In some situations, due to resource allocation or other system constraints, a lightweight, probable mechanism is needed to allow connectivity in constrained environments. Reality Warp Software uses the Client Agent architectural pattern to provide access and connectivity to data within systems that are unable to run a full instance of the Pirkolator. While not able to provide all of the benefits of a full installation, client agents can be tailored and optimized specifically for constrained environments.

3 Phase I Work Plan

3.1 Scope

For Phase I the RWS team will develop a data model for BDA that includes current EW assessment modeling. The RWS team will integrate existing data stores from any available EW system. For cases where access is not allowed, but still needed, the RWS team will provide a sample data store for testing and prototyping purposes. The RWS team will explore and develop modeling for digital network forensics. To build this model, the RWS team will explore available methods for digital network assessment. During this discovery, where existing tools are not available, the RWS team will generated technical reports that will be used during future phases to developed the required software or hardware for currently, non-existent assessment methods. The RWS team will provide an existing infrastructure and tools that will be used to facilitate BDA report generation, provide dynamic resource allocation, and adjust analytic processing under constrained resources. Lastly, RWS will explore and develop prototype visualizes of the results of our exploration with digital network forensic BDA generation.

Given the opportunity to pursue the Phase I option, we will solidify data source access and the BDA model to incorporate the new aspects of network forensics with existing BDA capability.

3.2 Task Outline

3.2.1 Phase I Tasks

3.2.1.1 Task #1 – Deploy Infrastructure

This is a relatively short task and involves setting up the project and tools to start research.

Estimate: 40 hours

3.2.1.2 Task #2 – Develop Data Sources
This task develops integration with any required, existing data sources. This task involves developing any prototype or test data sources. This task involves developing a mechanism to archive data for play back that is needed for accurate, repeated testing and analysis.
Estimate: 40 hours

3.2.1.3 Task #3 – Develop Data Model
This task will run in parallel with Task #2 to define the models needed for exploration.
Estimate: 40 hours

3.2.1.4 Task #4 – Research and Generate Digital Network Forensics for BDA
This task represents exploration into the tools and techniques for assessing a digital network.
Estimate: 480 hours

3.2.1.5 Task #5 – Develop Resource Allocation
This task defines the scope of effort for developing the management components required for provide low- or high-fidelity reports based on resource availability
Estimate: 80 hours

3.2.1.6 Task #6 – Result Analysis, Visualization and Refactoring
Based on the previous tasks, this task provides analysis of the BDA and visualization prototypes based on the resulting BDA reports. This task also allows for tweaking and updating based on analysis.
Estimate: 320 hours

3.2.2 Phase I Option Tasks

3.2.2.1 Task #7 – Extended Data Modeling
This task provides additional, expanded modeling.
Estimate: 120 hours

3.2.2.2 Task #8 – Data Source Integrations
This task provides integration and extension for existing data sources that can be used to contribute to BDA.
Estimate: 120 hours

3.2.2.3 Task #9 – Visualizations
This task further develops the visualization capability for the EWO.
Estimate: 120 hours

3.3 Milestone Schedule

3.3.1 Phase I Milestones

3.3.1.1 Milestone #1 – Platform and Infrastructure
Schedule – end of 2nd month of Phase I
Deliverable – technical review to include progress presentation and demonstration of platform with data source processing and initial BDA modeling results

3.3.1.2 Milestone #2 – BDA Generation
Schedule – end of 4th month of Phase I
Deliverable – technical review to include progress presentation and demonstration of generating digital network BDA reports

3.3.1.3 Milestone #3 – Prototype
Schedule - end of Phase I
Deliverable - technical review to include final report and demonstration of prototype with visualizations

3.3.2 Phase I Option Milestones

3.3.2.1 Milestone #4 – Extended Modeling
Schedule – midway into the Phase I option
Deliverable – Updated BDA modeling from extended research

3.3.2.2 Milestone #5 – Updated Prototype
Schedule – end of continuation period
Deliverable – Update of platform with extended model, data source integrations, and visualizations

3.4 Deliverables

3.4.1 Phase I Deliverables

Deliverable	Schedule	Description
Kick-off Meeting	2nd week	Initial meeting at APG
Progress Reports	Weekly	Technical progress updates for customer
Technical Review	Every 2 months	Technical presentation review via VTC for each Milestone
Prototype	End of Phase I	Delivery of prototype application supporting Phase I scope
Final Report	End of Phase I	Final report on Phase I scope

3.4.2 Phase I Continuation Deliverables

Deliverable	Schedule	Description
Continuation Kick-off Meeting	1st week of Continuation	Continuation meeting at APG
Progress Reports	Weekly	Technical progress updates for customer
Technical Review	Every 2 months	Technical presentation review via VTC for each Milestone
Prototype	End of Phase I Continuation	Delivery of updated prototype application supporting Phase I Continuation scope

4 Past Performance

Our experience from acting as the Chief Architect deploying the fusion framework currently used by the Distributed Common Ground System – Army (DCGS-A) program provides us with a unique insight into the requirements of developing fusion and correlation capabilities. The DCGS-Army fusion framework was developed in 2008 to address the need of data de-duplication within the Army's centralized data model called the Tactical Entity Database (TED). From this experience was gained an understanding of the need for efficient resource management, dynamic data connectivity, and distributed information utilization. It was realized that many of the techniques and methodologies employed in the de-duplication effort could be applied across a variety of analytic processes, such as association, correlation, aggregation, and temporal and geo-spatial trending. This also lead to an understanding that providing an integrated infrastructure with core support for data management and analytic processing engines provides a robust and easily deployed enterprise application solution for addressing specific data needs. This was proven at the Army's Empire Challenge in 2011 where the Army's first, cloud-enabled, mobile biometric fusion platform was demonstrated.

Our experience on the Army's Windshear II program in developing and demonstrating the first fully functional, cloud-based, mobile biometric solution at Empire Challenge 2011 further illustrates our awareness of the challenging problems of today's networked world. Windshear II was demonstrated successfully to the Undersecretary of Defense after only 4 months of architecture and development. The platform consisted of a cloud-enabled server in a vehicle that housed a data management capability, biometric analytics for facial recognition, voice recognition, retinal patterns, and personal identification. These elements were all collected using a Motorola's Atrix Android phones. As the data elements were collected from the phones, associations against an HDFS-based graph data store providing processing to alert back to the mobile devices when entities of interest were discovered. With a short timeline for demonstration and a relatively new application platform in Android, it was learned that open source solutions for data connectivity to mobile devices coupled with utilization of the then emerging HTML5 standard allowed a much faster path to integration of the mobile devices to the cloud-enable infrastructure. One of the stunning successes of this demonstration was the delivery of alerts not only to the mobile devices, but also through secure protocols to alert the higher echelons of the Army. This security driven effort provided the acknowledgement that an integrated, distributed, enterprise platform enables secure management of data.

These personal experiences brought to Reality Warp Software provide the key ingredients and foundation for its initial offerings and solution architectures. The Pirkolator, Determination Engine, and Trending Engine each provide a new, unique, and differentiated solution that provides the basis for RWS's approach to offering Big Data solutions. In providing these solutions as open source to the community, RWS's strategy is to provide industry specific analytic solutions built upon community driven and maintained tools. This allows for a focus on the real problems facing industry and not around the tools required just to get to the start of answering problems.

5 Key Personnel

5.1 Jeff Pirkey

- Principal Investigator

EDUCATION

- M.S., Engineering, University of Texas, 2008.
- B.A., Economics, University of Texas, 2004.

CURRENT POSITION AND RESEARCH

Jeff Pirkey currently acts as Chief Architect for Reality Warp Software and will be the Principal Investigator for this opportunity.

RELEVANT EXPERIENCE

Jeff Pirkey has over 20 years of experience in software engineering with Java and .NET-based technologies. He is an architect, manager, and developer with expertise in stand-alone, distributed, web-based, and visualization-based applications. He architected the white paper recognized at the 2012 Army Electronic Warfare industry day as the necessary solution for bringing the EWO into the fight. He lead the team that developed one of the first successful Army cloud edge nodes that was demonstrated at Empire Challenge 2011 using biometric services and mobile devices. He architected and developed the Fusion Exploitation Framework for the Army to provide the fusion framework for the DCGS-A platform. He has developed and fielded tactical airborne and ground-based SIGINT and MASINT data collection, mapping, and geo-location applications. He has written several papers on data processing infrastructures, including a paper presented at NSSDF in 2010 on Adaptive Sensor Processing Supporting Event-Driven Fusion.

5.2 Daniel A. Pass

- Expert Consultant

EDUCATION

- B.A. Political Science, Washington State University
- MBA Technology Integration and Management, Pacific Lutheran University
- M.S. Social Science and Middle Eastern Studies, Pacific Lutheran University
- Graduate - US Army Command and General Staff College
- Graduate - USMC Command and Staff College
- Project Management Professional (PMP)

CURRENT POSITION AND RESEARCH

- Adjunct Professor, Pacific Lutheran University, Tacoma WA
- Consultant ISR and Defense Programs

RELEVANT EXPERIENCE

Dynamic, focused and cross-cultural business executive, plus operations and data/cost analysis with an 18 year history of demonstrated leadership and success that includes award-winning results in highly competitive markets, both federal and commercial. An expert in the Defense and Intelligence communities, and tenacious in building and leveraging service delivery models and forging strong relationships with external business partners. Recent experience includes a year at the National Training Center, Ft Irwin CA where I was selected to be the Senior Training Director for a congressionally mandated Expeditionary Training Capability. Responsibilities included all Intelligence, Surveillance and Reconnaissance (ISR) training scenarios developed for certification training for all Army Brigade Combat Teams in preparation for deployment overseas. This included all Division and Echelon Above Division ISR perspectives for intelligence collection, analysis, unmanned aerial system deployment, airspace control and targeting. While in this assignment, I developed the first Cyber/EW training package for the ETC mandate.

5.3 Raymond Page

- Expert Consultant

EDUCATION

- M.S., Computer Engineering, University of Florida
- B.S., Computer Engineering, University of Florida

CURRENT POSITION AND RESEARCH

- Raymond Page is the founder and President of ICAT, LLC. He is responsible for all operations, research, and development at ICAT, LLC.

RELEVANT EXPERIENCE

Raymond Page has over 7 years of engineering experience and over 15 years experience with open source software. He is a cloud architect, reverse engineer, and developer of distributed communication systems. He supported the cloud edge node at the Army Empire Challenge 11, a blue print architecture for Army Red Disk infrastructure. He single handedly deployed, maintained, and improved the cloud deployment for INSCOM Futures' Red Disk. Prior to cloud development, he was a reverse engineer working with wireless communication systems and performing source code audits.

Figure 2: BDA Processing Platform