



CRYSTAL DATA



System and Method
for the Management and
Monitoring of Remote Live Sensor Data



Introduction..... 3

 SERVERINSIGHT™ OPERATIONAL FRAMEWORK.....4

 FUNCTIONAL DIAGRAM6

ServerInSight Communications Conduit (SICC)..... 7

ServerInSight Web Application Interface (SIWAI) 8

ServerInSight™ System Components..... 9

 CLUSTER9

 COLLECTORS10

 ZAILOTS..... 11

Sensor Web Enablement..... 13

 OBSERVATIONS & MEASUREMENTS SCHEMA (O&M) 13

 SENSOR MODEL LANGUAGE (SENSORML)..... 13

 SENSOR OBSERVATIONS SERVICE (SOS)..... 13

 SENSOR PLANNING SERVICE (SPS) 13

 SENSOR ALERT SERVICE (SAS) 14

 WEB NOTIFICATION SERVICES (WNS)..... 14

ServerInSight™ System Summary..... 15





ServerInSight™

System and Method for the Management and Monitoring of Remote Live Sensor Data

Introduction

Massively distributed data-gathering environments can be built quickly and managed easily through a hierarchy of specialized software and hardware components that minimize the number of operators and the skill sets required to manipulate them.

This abstract describes a scalable management system for the acquisition of data through the remote management of sensor devices and their associated data where a sensor device can be, among many other possibilities:

- Visible and invisible spectrum sensors such as cameras (in-situ, airborne, and satellite), LIDAR, radar, and motion detectors
- Audio monitors
- Seismic sensors
- Physical data sensors such as thermometers, voltage, pressure, and/or weight sensors
- Environmental sensors such as weather stations, stream gauges, and buoys

The management system includes collection methods and details for such information to be received from a remotely located live event, recorded for historical analysis, and transformed into an array of formats over a computer





network, such as the Internet. Any number of representations, virtual or realistic, may be produced from the collected and managed data. The totality of control over the remote sensors puts the collection of data in the purely robotic realm, and the hierarchical method of management allows a single (or a small number of) operator(s) to control many collection devices without the need to understand the location and/or network correlation to the device by the operators.

ServerInSight™ Operational Framework

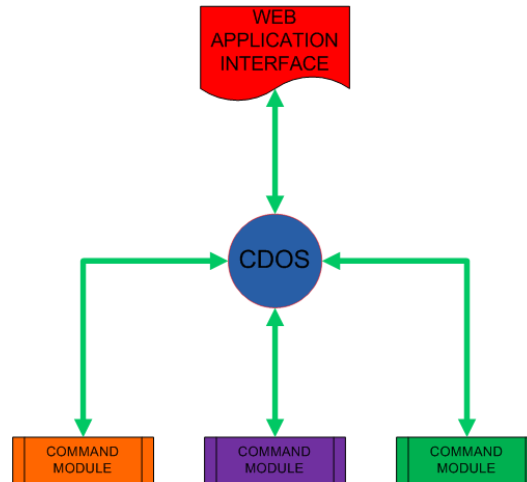
ServerInSight™ operates within an HTML/XML environment that serves as the human interface between a centralized communications device and various data collection devices. The framework is hierarchically redundant, with each device capable of performing all functions. Dynamic interchangeability enables flexibility during times of hardware or performance issues.

The system utilizes proprietary CDOS network messaging to manage communications between all devices. The messaging transport is made possible through the use of the ServerInSight Communications Conduit (SICC), which consists of a proprietary client program and a proprietary server daemon. Messages are sent by the client program and received by the server daemon where they are translated into specific system actions. These actions are initiated as tasks by the server daemon to the various command modules located on the data collection devices. Once the tasks are completed (successfully or otherwise), the result status is retrieved by the server daemon and sent back to the client program for display.

The ServerInSight Web Application Interface (SIWAI) – a non-proprietary Internet web browser and a proprietary Internet web daemon – can communicate indirectly through the SICC due to its ability to translate URL entries into commands that can be passed by the client program to the server daemon. In



this scenario, the SIWAI becomes the initiation point of all messages and the destination point of all results, creating a seamless interface for the user into the SICC. To create this environment, point-and-click navigational menus (with properly embedded URLs) are displayed by the SIWAI allowing the user to manage and communicate within the framework of the system.



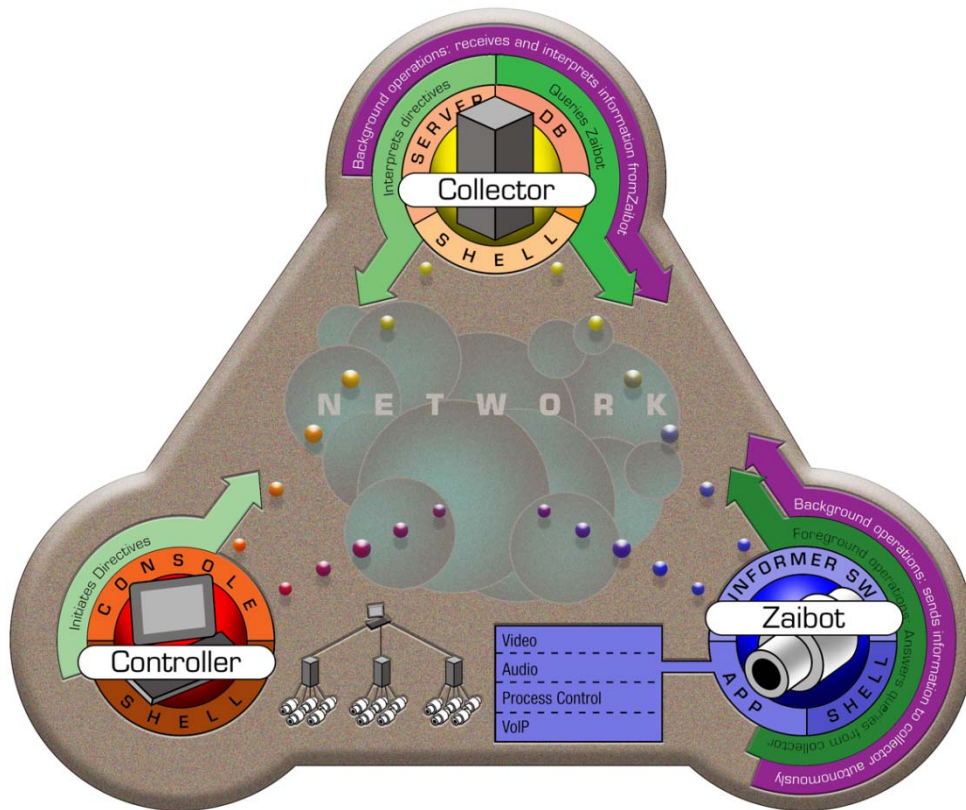
By choosing one of the available selections, various intra-computer and inter-computer communications are generated by the effected devices and relayed along the SICC. The resulting communications are interpreted by the SIWAI and displayed within a graphical interface to the user for visual interpretation and further action or intervention.

This hierarchical system is comprised of devices called Controllers, Collectors, and Zaibots. The Controller is the device managed by the user, while the Collector is responsible for all communications along the SICC to the data collection devices, known as Zaibots.

A logical association of multiple Controllers, Collectors, and Zaibots is deemed to be a Cluster.

Functional Diagram

In the Functional Diagram below, the Controller, Collector, and Zaibot components are represented within a computer network. The figure displays the relationship between the Controller, Collector and the Zaibot as they communicate using the ServerInSight Communications Conduit within a single Cluster.





ServerInSight Web Application Interface (SIWAI)

While the ServerInSight Communications Conduit (SICC) enables communications between two proprietary computer processes, the ServerInSight Web Application Interface (SIWAI) may be utilized for communicating between the computer and the end user. It consists primarily of two different components.

- WebCDOS server daemon
- WWWCDOS module

As was previously discussed, WebCDOS is a proprietary server daemon responsible for executing HTML or XML directives. These directives may be HTML text files containing only HTML tags and text, XML defined data, or they may be procedures that adhere to the Common Gateway Interface (CGI) standard, which is a method whereby applications are integrated into the HTML process. These procedures are typically programs such as a PERL program, an AJAX function, or a BASH shell script that dynamically create HTML text based upon user responses or system configuration parameters.

In the ServerInSight Web Application Interface, the collection of files, both text and CGI, are referred to as the WWWCDOS module.

Due to the proprietary nature of WebCDOS, WWWCDOS does not have to conform to the standards of other web server applications and is capable of using non-standard syntax to invoke desired system activities and uniquely display system responses.



ServerInSight™ System Components

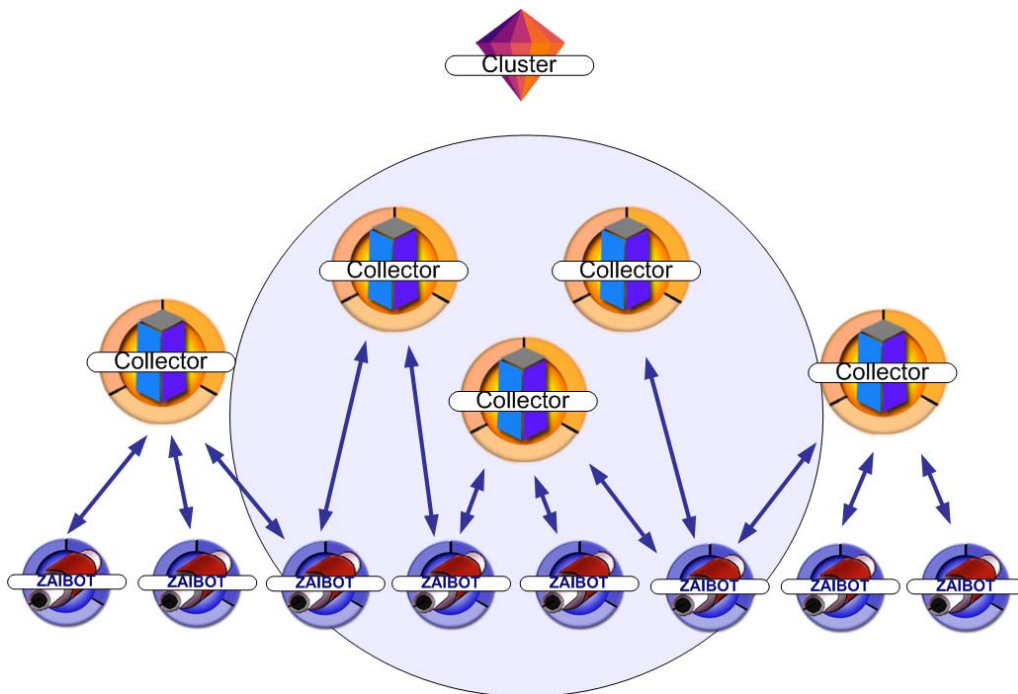
Clusters are created by using the ServerInSight Communications Conduit (SICC) and the ServerInSight Web Application Interface (SIWAI) to logically connect the ServerInSight System components – Controllers, Collectors, and Zaibots.



Clusters

A Cluster is a logical communication relationship between Collectors and Zaibots that creates a system for managing and monitoring several Collector environments within a single interface (the **Controller Console Interface**).

The Cluster concept is the basis for the hierarchical or parallel management scheme.





A typical Cluster configuration is comprised of a single Controller, which is used to manage various Zaibots via a single Collector. However, the Controller may be associated with more than one Collector, creating a multi-tiered Cluster.

The hierarchical relationship between Controllers and Zaibots can be graphically displayed within the **Controller Console Interface** in a tree representation. By viewing the Cluster tree, a user can easily determine which Collectors receive statistical and data streams from the available Zaibot realm.



Controllers

The Controller is responsible for the graphical user interaction between Collectors and Zaibots. Many configuration parameters are maintained and data collection devices are controlled from within its graphical user interface, the **Controller Console Interface**.



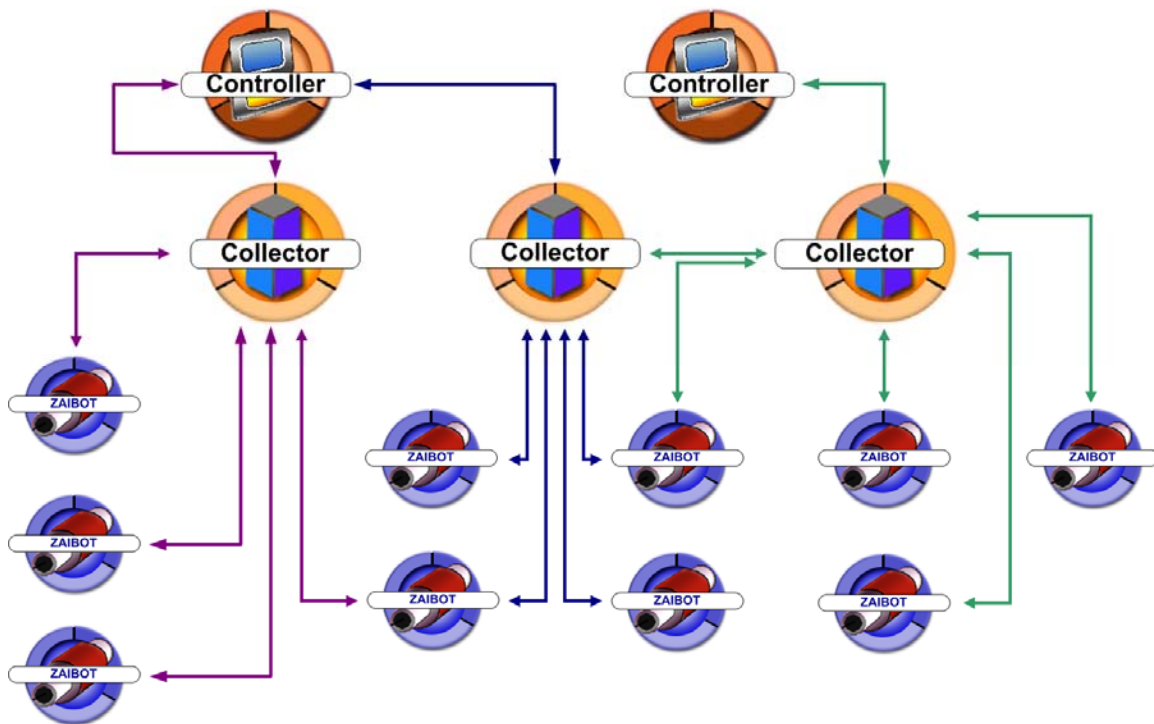
Collectors

Collectors are computer systems that acquire sensor data from the Zaibot data collection devices and provide the structure for displaying the information to the end user. They are responsible for collecting and redistributing all the information gathered from the Zaibots.

Collectors may be arranged in either a hierarchical or a parallel relationship. The data received by a Collector may be shared among multiple Collectors or may be limited to a single Collector. Collectors may be deployed in relationships where



one or more Collectors respond to conditional demands made by one or more other Collectors.



Collectors use an array of messaging protocols including HTML, XML, and plain-text through socket-based, TCP/IP communications using the ServerInSight Communications Conduit (SICC). Messages may be stored in a relational database, reviewed for conditions by a Controller configured for that purpose, acted upon in real-time, or any combination of these uses.



Zaibots

Zaibots are computer systems comprised of hardware and software components that can monitor, record, or transcode (transcoding is the automatic repurposing of content to meet various display, recording, or transmission demands) the data





streamed by a variety of sensors to which they are attached. Zaibot data collection devices can retrieve information in a number of formats that include, but are not limited to:

- NTSC video
- Digital Video
- Analog or Digital audio
- Serially collected data
- Direct sensor data

The output of Zaibot data collection devices is either sent to Collectors as it is gathered or as the data meets conditional parameters, such as a sensor reaching levels deemed necessary to report. Collectors may be configured to query Zaibots for sensor information under human control or autonomously.

Typical data exchanges between Zaibots and Collectors include information concerning, but are not restricted to:

- Current tasks
- Current operating conditions
- Hardware malfunctions

The communications to and from the Zaibot are handled via TCP/IP utilizing the ServerInSight Communications Conduit. The device must be accessible via a public IP address. However, Network Address Translation (NAT) is acceptable, but the defined port ranges must be made available to the NAT enabled machine.



Sensor Web Enablement

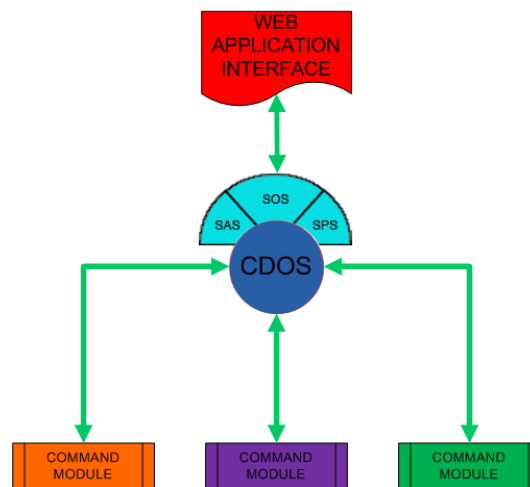
To meet certain sensor communication requirements, Sensor Web Enablement (SWE) technologies, developed under the auspices of the Open Geospatial Consortium (OGC), have been incorporated into the ServerInSight™ architecture. This platform enables a vast collection of sensors to be easily integrated into a manageable environment. The SWE standards that have been integrated into ServerInSight™ include both XML schema manipulation and web services, including:

- **Observations & Measurements Schema (O&M)**
Standard models and XML schema for encoding observations and measurements from a sensor, both archived and real-time.
- **Sensor Model Language (SensorML)**
Standard models and XML schema for describing sensors systems and processes; provides information needed for discovery of sensors, location of sensor observations, on-demand processing of low-level sensor observations, and listing of taskable properties.
- **Sensor Observations Service (SOS)**
Standard web service interface for requesting, filtering, and retrieving observations and sensor system information. This is the intermediary between a client and an observation repository or near real-time sensor channel.
- **Sensor Planning Service (SPS)**
Standard web service interface for requesting user-driven acquisitions and observations. This service is the intermediary between a client and a sensor collection management environment.



- **Sensor Alert Service (SAS)**
Standard web service interface for publishing and subscribing to alerts from sensors.
- **Web Notification Services (WNS)**
Web service for asynchronous notification through email, Small Message Service (SMS), phone, Extensible Messaging and Presence Protocol (XMPP), or through a URI

The SWE framework provides open standards for discovery, access to, and processing of sensor observations, as well as the tasking of sensor systems and publishing of sensor alerts. ServerInSight™ provides a reliable yet flexible method by which these systems may be analyzed and controlled in a manageable environment.





ServerInSight™ System Summary

ServerInSight™ is a specialized product that allows one or a small number of operators to affect a large number of intelligent Controllers, which in turn may be employed to securely gather, manage and archive massive amounts of data from any number of remote locations.

The core of the ServerInSight system resides within two major components: the ServerInSight Communications Conduit (SICC) and the ServerInSight Web Application Interface (SIWAI). The interaction between these two components allows the ServerInSight system to communicate with diverse hardware and software devices to complete uniquely diverse tasks.

The system's components are specifically designed for their assigned tasks, and thus are faster, more secure and more interoperable without human intervention than any other technique known for the remote gathering, management and distribution of video, audio and other information from a variety of sensors. These devices are controller by individual module directives, which are easily adaptable to new environmental requirements.

There are three components that comprise the ServerInSight operational environment; Controllers, Collectors, and Zaibots. The hierarchical combination of these devices is deemed a Cluster.

Controllers are computer systems that provide system operators with the ability to communicate with Collectors. They are the human interface into the ServerInSight system. Controllers allow operators to configure, monitor, and instruct Collectors as to whether or not to take action upon data acquired by the Zaibot Sensor Units. The Controller is capable of communicating with virtually





any number of Collectors. This communication is managed under the same socket-based protocols used by the Collectors.

The Collector operates via a server daemon designed especially for operation within this system. It provides the triple advantages of making the best use of bandwidth allocation for the task at hand, running faster than a non-proprietary server designed to handle a wide latitude of tasks, and being more secure than a non-proprietary server.

The Collector keeps track of the network addresses and locations of the Zaibots, alleviating the need for the human operator(s) to perform that task and thereby simplifying the operation of a vast network of data collection that would otherwise be unmanageable. The Collector is capable of communicating with virtually any number of Zaibots.

The Zaibot's relationship with the Collector also provides advantages that provide superior information concerning remote environments, while minimizing the number of human operators and their degree of specialization to perform such tasks. The Zaibot's reporting mechanism constantly "discusses" its information with the Collector's shell. This automatic transfer of information includes the abilities to:

- Reject any information from the Controller or a device representing itself as the Controller that the Zaibot does not recognize
- Perform the task demanded by the Controller





- Inform the Collector (and if so configured, the operator) that the task cannot be performed, with a reason why the task could not be performed (if so configured)
- Push information to the Collector without being queried for it

The ServerInSight realm of capability can be easily expanded by the creation of new communication modules, which allow the system to quickly and efficiently manage new devices. The operational framework of the core routines was designed specifically to allow manageable adaptation into new environments. The communication parameters need only to be identified and integrated into the ServerInSight Communication Conduit messaging protocols, while the ServerInSight Web Application Interface can be easily modified to allow users to manage these new resources.

The implementation of Sensor Web Enablement within the ServerInSight™ architecture adds yet another layer of communications and control for sensor array management, while adding an open source platform that decreases the time required for including a vast array of sensor responses into the ServerInSight environment.

The ServerInSight System Method for the Management and Monitoring of Remote Live Sensor Data is uniquely positioned to provide user interactivity and management authority over all remote live sensor devices.

