Robotics 2000. ASCE Conference on Robotics for Challenging Environments.
February 28-March 2, 2000, Albuquerque, NM, 2000.

Automated Part Tracking on the Construction Job Site

William C. Stone[1], Lawrence Pfeffer[2], and Karen Furlani[3]
william.stone@nist.gov, pfeffer@nist.gov, karen.furlani@nist.gov

ABSTRACT

Efforts are underway at the National Institute of Standards and Technology (NIST) to develop a web-based system for rapid tracking, identifying, and locating manufactured components on the construction jobsite. The approach involves the use of RFID and barcode identification systems, 3D long range coordinate measurement technologies, portable/wearable computers, wireless communications, high speed networking, temporal project databases, web-based data analysis, and 3D user interfaces to provide as-is and as-built component data at the actual construction site. These same techniques may prove useful for planning and execution of construction operations on other planets. Present research is focused on developing compact, rugged, wireless part status managers, interoperability protocols for data transmission, and 3D site visualizers which reflect the current state of tracked components on the construction site. Ultimately, these field measurements will be used to generate automated material acceptance and payment transactions, damaged component rejection notices, and production rate change requests to fabricators. This paper discusses the methods employed to achieve this capability in a prototype system.

Introduction

The Construction Metrology and Automation Group (CMAG) at the National Institute of Standards and Technology (NIST) has initiated a project in Real-Time Construction Component Tracking to address the problem of identifying, locating, and tracking discrete construction components and sub-assemblies on a construction site. Discrete components in the context of the present discussion are manufactured, prefabricated items (such as wide flange steel beams) and assemblies which might logically carry an identification means (such as a bar code or radio frequency identification [RFID] tag). The project objectives are to: 1) develop means for real-time tracking of sub-assemblies and components; 2) develop standards for component identification and tracking that the construction industry will adopt; 3) develop standard means to wirelessly transmit that information to a construction project database; and 4) demonstrate the utility of these techniques on full-scale construction sites.

[1]Leader, Construction Metrology and Automation Group, NIST, 100 Bureau Dr., Stop 8611, Gaithersburg, MD 20899-8611/ USA
[2] Research Civil Engineer, Construction Metrology and Automation Group, NIST, Gaithersburg, MD 20899-8611/ USA
[3] Research Civil Engineer, Construction Metrology and Automation Group, NIST, Gaithersburg, MD 20899-8611/ USA

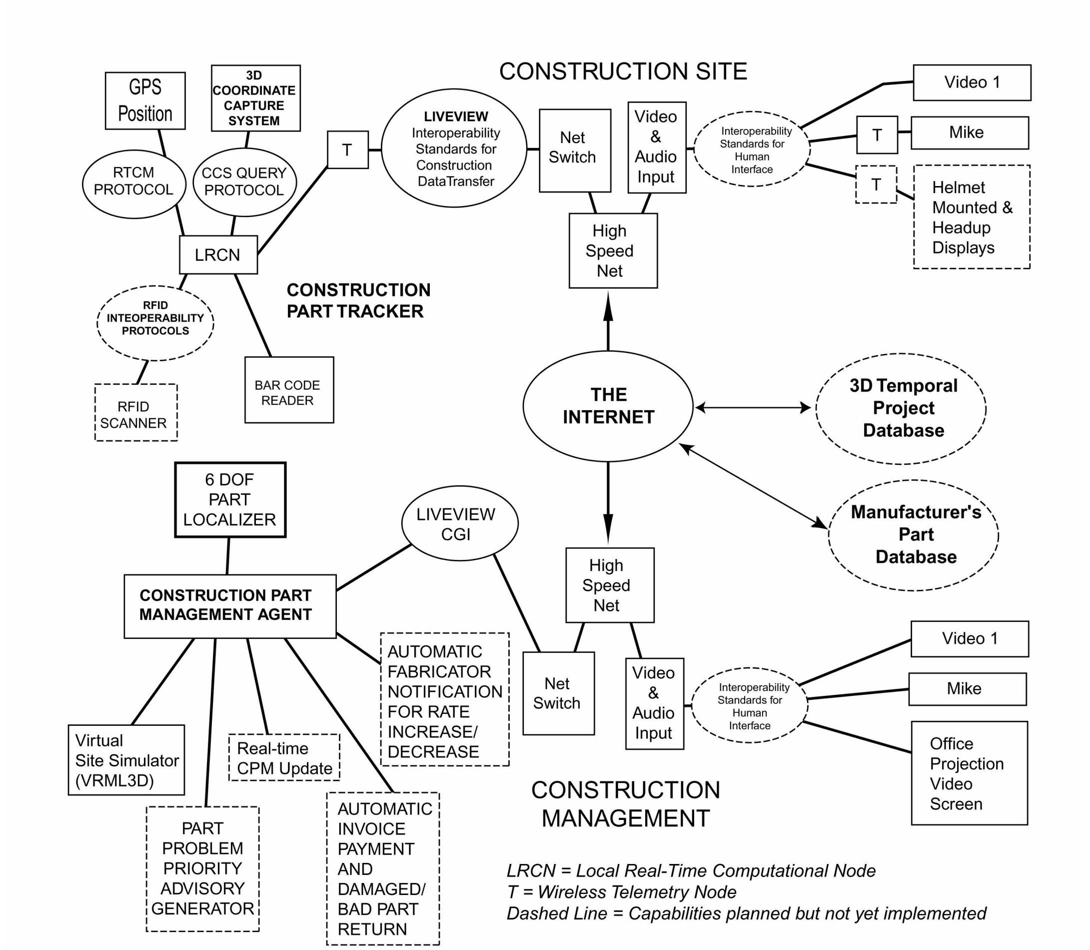

Figure 1: Network diagram for the inclusion of jobsite part tracking into the construction management process. This architecture was implemented and tested live at NIST before an audience of 300 on June 24, 1999. The audience played the role of the construction management team whose job was to assess the status of a remote construction site located one kilometer from the auditorium. Live data included audio, video, and identification and coordinate data for components, located by a quality control officer on the job site. The component packet data was automatically processed by a 6-DOF part localizer agent residing on a separate computer. The updated part position and orientation were then sent to a visualizer for display. The NIST LiveView communications protocol provides the interoperability standard for the uplink of all these data. The data backbone consisted of a 155 Mbs ATM fiber optic link.

Part Tracker System

The prototype component tracking system developed for this project integrates field sensors, portable computers, wireless communication, and real-time 3D coordinate measurement systems (see Figures 1 and 2). Individual objects scheduled for arrival on the construction site are tagged at the fabricators using bar codes or radio-frequency identification tags (RFID). The encoded information is scanned directly into a portable computer and is wirelessly relayed to a remote *temporal project database* (a repository of transient information associated with items anticipated to be on site along with reference pointers to other databases such as 3D part libraries and manufacturer's specifications -- currently under active development at NIST). A query to the *temporal project database* returns graphical representations (e.g. virtual reality modeling language [VRML, 1999] models) of scanned

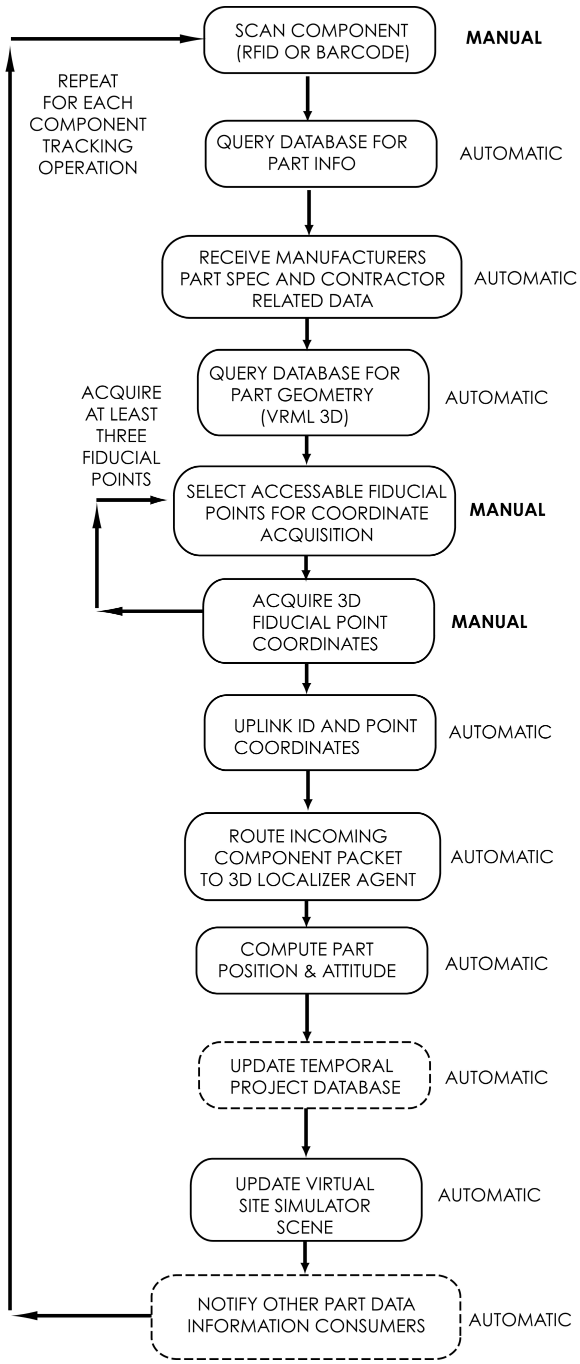

Figure 2: Flowchart for the NIST construction job site part tracker. The system is presently implemented on a laptop computer that is wirelessly linked to the Internet. Peripheral devices that have been integrated into the laptop include a barcode scanner and a 3D long-range coordinate measurement device. The system is being modified to handle multiple coordinate input sources, including RTK GPS. Dashed lines indicate components planned but not yet implemented.

objects and additional information as appropriate. Most notably, ancillary information includes a visual description of unique *fiducial* points assigned to the component (see Figure 3e). These models, coupled with web browsing software, guide a Field Inspector through the acquisition of the key fiducial point positions using a long-range coordinate measurement device. This data is sent to a non-local part localizer to determine the object's position and orientation in the job site coordinate system. The database then uses this data to track the current state of the object.

Prototype Testbed

A construction lay-down area testbed has been established at NIST to provide an initial means for testing and demonstrating the effectiveness of a prototype tracking system. This testbed contains a representative assortment of construction components (e.g. I-beams). Each component is tagged with a bar code tag, encoding the unique identification number previously assigned to that object in a testbed database. A laptop computer presently serves as host for the bar code field sensors, the construction database, and the web pages which guide users through the component identification process (see Figure 3). The database for the discrete component tracking tests is based on MicroSoft Access97. The entire system runs through an Internet Explorer 4.1 graphical user interface (GUI) written in Microsoft FrontPage98, driven by a VisualBasic script [Furlani et.al, 1999]. The web pages prompt the user to scan a barcode tag, and then queries the database to return information associated with the scanned component, including a 3D VRMLmodel of the object. A Proxim wireless Ethernet card allows for wireless data transfer with a master site integration computer hosting the project database. Presently the site computer is an SGI workstation.

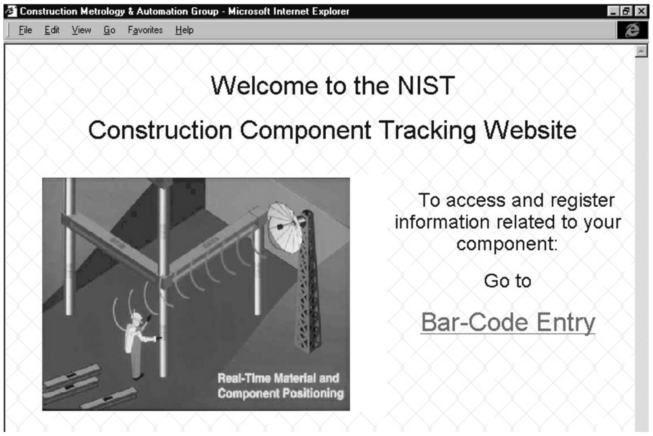

Figure 3a: Home page on the NIST part tracker. The Field Inspector at the job site selects the type of entry (barcode or RFID) and is then prompted to conduct an ID scan.

Figure 3b: Field Inspector acquires the component ID using a barcode scanner; the same information may be automatically acquired using non-contacting RFID technology. Part ID and coordinate acquisition tasks are integrated through a local laptop computer which is wirelessly connected to the Internet. Work is underway at NIST to integrate these functions into a compact, field portable device.

Figure 3c: A successful part identification brings up the component ID number. The Field Inspector then wirelessly submits a part query through the web for further information.

The field laptop is also connected to a 3D long range (50 meter radius) coordinate measurement system -- a Vulcan system from Arc-Second [Arc-Second, 1999]. An interface C program was written at NIST to interface the laptop to the Vulcan sensor. A LiveView [Latimer, 1999] CGI script provided the protocol for the transmission of both the part ID and its coordinates to a remote part locater agent (see below). The integration computer is linked to other labs throughout NIST via an ATM 155 megabit/s fiber network. The weak link in the system is presently the wireless LAN, which is limited by bandwidth and range. Once the part locator agent develops a best fit for the position and attitude of the part these data become available for display using the Virtual Site Simulator (VSS), a data-driven augmeted simulation written in-house using the WorldToolkit graphics/simulation libraries [World Toolkit, 1999].

Part Localizer Agent

The part locator is a data interpreter -- a conceptually separate computational "server" whose role in is to take the measured fiducial-point locations and certain data from the CAD model of the component, and compute an estimate of the part's position and orientation in the site's coordinate system. The part locator acts as an aggregator, that is, it combines different types of information (point locations and CAD model), and determines something -- the components position/orientation -- that cannot be determined from either type of information alone.

Although conceptually separate, the part locator can either be run on separate computer (or not), depending on the particular constraints of the implementation. In fact, in a large-scale implementation, there could be multiple part locators, dividing up location requests. In the two implementations we have done to date, the part locator

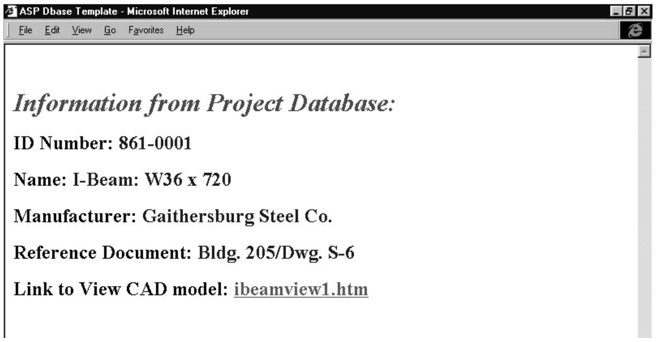

Figure 3d: The remote web query returns further confirming information associated with the part. Although significant amounts of such information can be stored local to the part using RFID technology, the web-based system allows greater flexibility.

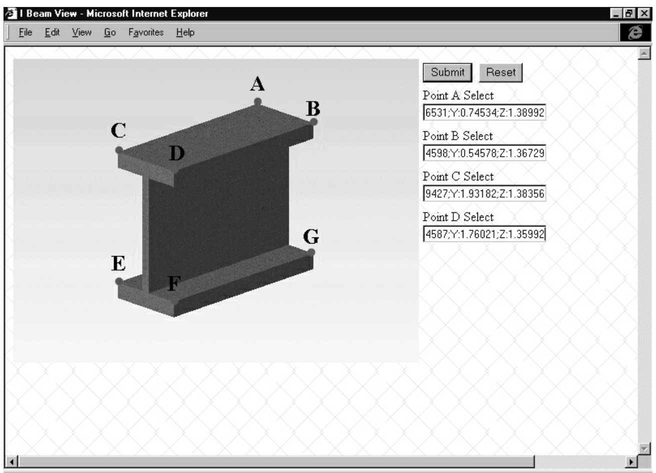

Figure 3e: The Field Inspector will then download a 3D model of the part. This provides not only crucial visual confirmation that the scanned part is what the inspector thinks it is, but also visual queues for the location of key fiducial points necessary for component location and orientation acquiisition.

Figure 3f: The Field Inspector acquires at least three fiducial point locations (a process that takes about 30 seconds total). The part ID and the fiducial point coordinates are then wirelessly uplinked to the part localizer.

process was run on a separate computer. The problem statement is as follows: Given two sets of inputs:

(1) the measured locations (x, y, z) of three or more (not all colinear) fiducial points (labelled, or in a clear order), measured in some coordinate frame (the "site" frame), and

(2) data from a CAD model, of the corresponding nominal (x, y, z) locations of the fiducial points in a frame fixed in the part (the "part" frame),

determine

(1) the "best" estimate the position (three translational degrees of freedom) and orientation (three degrees of rotational freedom) of the part frame in the site frame, and

(2) determine a metric that indicates the fidelity of the position/orientation estimate.

To assess the fidelity of the position/orientation estimate, one has to define the metric that measures "goodness" of fit in some reasonable, and readily calculable fashion. This metric may also be used as the basis to find the position and orientation, by using it as the cost function in a numerical optimization (search.) There are many such cost functions. Probably the most intuitive family of costs for this problem is based on the differences between the measured positions of the fiducial points and their corresponding "best-fit" positions, computed from the candidate part position/orientation, the CAD data, and rigid-body kinematics. Our current implementation uses a member of this family, namely the sum of the squares of the distances between the measured fiducial point positions and the corresponding rigid-body-kinematic positions, calculated from the estimated position/orientation and the CAD data (see Equation 1). Thus, the

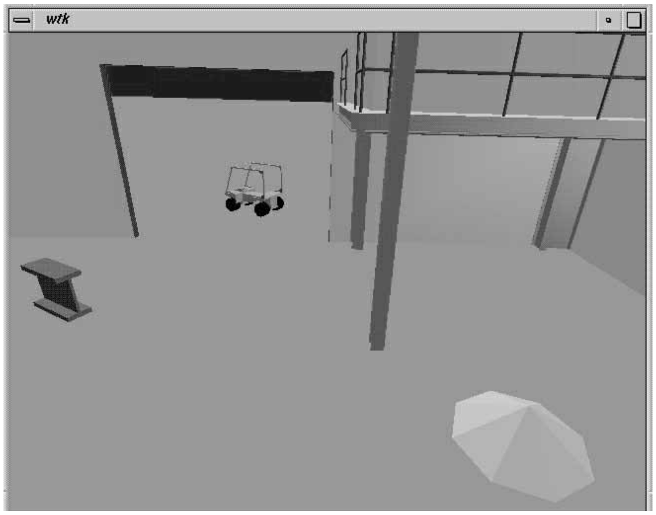

Figure 3g: "Live" visualization of the site. The field data collector acquires up-to-date locations for part fiducial points and makes these avaliable the construction management system. A 6 DOF part localizer uses the data provided. It performs an optimization analysis to best locate the part origin and the yaw, pitch, and roll for the part. This derivative data is then made available to the temporal project database. Another automated agent, known as the VSS (for Virtual Site Simulator) is a 3D Augmented Simulation system that creates a data-driven three dimensional view of the construction site showing the current status of tracked components, scanned terrain data, and tracked machinery. The I-beam described in the previous figures is shown at left center, properly updated after being moved and re-measured. Updates are generally discrete, quantized changes, since measurements will typically only be taken at the conclusion of each move of the component. However, the NIST system can handle real-time update rates that permit live tracking of a component placement as well as the tracking of the machine manipulating the object.

$$J(X) = \sum_{i=1}^{n} \left| p_{meas(i)} - p_{kin(i)}(X) \right|^2$$

Equation (1): Cost function for part localizer.

$$X = \left[x, y, z, \theta_x, \theta_y, \theta_z \right]^T$$

Equation (2): Position and orientation estimate vector for the component being submitted to the part localizer.

cost (J) is a function of a six-parameter vector (X) that represents the estimate of the position and orientation of the component, with three components for translation (x, y, z) and three more for orientation (the θ terms, see Equation 2). Note, for orientation, we are using rotations about site-fixed axes, applied in the order x, y, and then z.. Other conventions are, of course, possible. This cost function has the advantage that it has an analytic expression for its gradient.
A closed-form expression for the gradient of the cost can significantly speed up the numerical search for the best-fit position/orientation.

Network Interfaces to the Part Locator

Since the part locator is a separate "server" engine, that generally resides on another computer, the other system components that need to communicate with it need agreed-upon interface protocols to send inputs to the locator, and to receive the results it calculates. For flexibility and generality, we have chosen to use simple, network-based interfaces. At present, our system uses two different network protocols, one to receive input data for locating a part (LiveView), and another network protocol (tetSock) for reporting the calculated position/orientation that results. We plan to unify both of these under the LiveView protocol shortly. The source of the input data may not, itself, care about the results; however, other entities in the system will generally need this information.

LiveView Interface

In our initial implementation, the input data to the locator (i.e. the measurements of fiducial points) is sent to the locator using the LiveView protocol [Pfeffer and Latimer, 1999], using the *pointCloud* message type. (Note, LiveView is a network protocol being developed at NIST, that is based on the IEEE 1278 protocol family [IEEE, 1998] for distributed interactive simulations.) The *pointCloud* message type is used to communicate an ordered sequence of Cartesian (x,y,z) triplets, defining the locations of points of interest (in this case, the measured locations of the fiducial points.) The CAD data are not transmitted by the field data collector (which may not always have that data available). Instead, the locator is responsible for obtaining that information. At present, the locator has data tables for the few objects of interest; in the future, the locator will query a separate database for the CAD data it needs, probably using further developments of the LiveView protocol.

Once the part locator has computed the position and orientation of the part, it can provide this information to other systems that can make use of this information, such as a project database, or a site vizualization or planning system. In our initial implementation, the data are provided in real-time to a visualization system, the NIST Virtual Site Simulator (VSS) [Stone, et.al.,1999; Pfeffer and Latimer, 1999; Furlani and Stone, 1999]. Thus, when a component is re-scanned (after a move), the visualization updates itself, using that "live" data, to show the current state of that component on the site. For the initial demonstration of component tracking, the interface to the VSS used an older NIST-developed network protocol, called tetSock [Stone,1999]. The tetSock protocol (named for *TETRA socket*, was originally used to in conjunction with the TETRA 6-DOF tele-operated crane [Bostelman,1996].

TetSock is a simple, TCP-socket-based point-to-point protocol (rather than multicast.) Once a connection is established, it uses a request/response scheme, in which the application that needs the component location issues request messages over the socket, and then receives the corresponding data via the same connection. tetSock implements three different requests: Cartesian, joint (actuator), and tool coordinates, and three corresponding response messages. For this work, only the Cartesian request/response pair was used. All tetSock messages are ASCII-encoded strings, with a fixed length (256 bytes.) The Cartesian message contains six ASCII fields, delimited by whitespace. Each field is an ASCII-encoded floating point number. The six fields represent respectively:

X displacement from origin (in meters),
Y displacement from origin (in meters),
Z displacement from origin (in meters),
rotation about X axis (in radians),
rotation about Y axis (in radians),
rotation about Z axis (in radians),

The rotations are applied in the order Y, X, Z. A weakness of this scheme is that the origin and axes of the TETRA coordinate frame are not sufficiently rigorously defined -- the origin and axes can vary, from one TETRA setup/calibration to the next. The original tetSock implementation took it's coordinate zero and axis frame (home frame) to be wherever the TETRA manipulator had been moved to when its most recent "home" command was executed. One strength of the IEEE 1278 (and thus *LiveView*) system of reporting position and orientation is that it rigorously defines an earth-centered, earth-fixed coordinate scheme as a single common (and literally), global coordinate system. (IEEE 1278 also defines a fairly rigorous scheme for defining entity (e.g. vehicle-centric) local coordinate frames.)

Conclusions:

The procedures and techniques described here were implemented in actual hardware and software and demonstrated live before an audience of 300 at NIST on June 24th, 1999. The audience was located a kilometer distant from the “construction site” where the components were being tracked. They were able to follow both the component movements via live video images as well as through the data-driven 3D Virtual Site Simulator. Plans are presently underway to use this technology on a construction job on the main NIST campus in the spring of 2000, and to conduct return-on-investment studies as well as assess the impact of information provided by this new approach that never before existed. The field data collection system is being integrated into a rugged hand held computer that will handle different coordinate measurement sensors (including both laser-based and GPS). Plans are to track both discrete component movements (as described herein) as well as live motion of large assembly placements.

References

Arc-Second (1999), ArcSecond, Inc., 44880 Falcon Pl. #100, Dulles, VA 20166.

FIATECH (1999), Brochure and video tape available from CII/FIATECH Consortium, c/o CII, 3208 Red River St, Suite 300, Austin, TX 78705-2697, http://construction-institute.org

Furlani, K., Pfeffer, L., Latimer, D., and Stone, W.C., (1999), "NIST Construction Automation Program Report No. 5: Real-Time Vehicle and Discrete Component Tracking for Construction," National Institute of Standards and Technology, Gaithersburg, MD 20899, USA.

Furlani, K.M, and Stone, W.C., (1999), "Architecture for Discrete Construction Component Tracking," Proceedings, 16th IAARC/IFAC/IEEE International Symposium on Automation and Robotics in Construction, September 22-24, 1999, Madrid, Spain, pp 289-294.

IEEE, (1998), "IEEE Std. 1278-1a-1998, Standard for Distributed Interactive Simulations - Application Protocols," IEEE Customer Service, Piscataway, NJ 0855, (800) 678-IEEE, (refers to 1995 ver.)

Latimer, D.T., (1999), "Construction Metrology Standards in Orbiting Facility Construction, Maintenance, and Operation," Proceedings, ASCE Robotics 2000 Conference, Albuquerque, NM, February 28-March 2, 2000.

Pfeffer, L.E., and Latimer, D.T., (1999), "Toward Open Network Data-Exchange Protocols for Constructin Metrology and Automation: LIVEVIEW," Proceedings, 16th IAARC/IFAC/IEEE International Symposium on Automation and Robotics in Construction, September 22-24, 1999, Madrid, Spain, pp 117-122.

Stone, W.C., Reed, K., Chang, P., Pfeffer, L., and Jacoff, A., (1999), "NIST Research Toward Construction Site Integration and Automation," ASCE Journal of Aerospace Engineering, April 1999, Vol. 12, No. 2, pp 50-57

VRML, (1999), International Standard, ISO/IEC 14772-1:1998, The Virtual Reality Modeling Language (VRML).

WorldToolkit, (1999), SENSE8 Corp., 100 Shoreline Highway, Suite 282, Mill Valley, CA 94941.