CIRP Journal of Manufacturing Science and Technology xxx (2017) xxx–xxx

Contents lists available at ScienceDirect

CIRP Journal of Manufacturing Science and Technology

journal homepage: www.elsevier.com/locate/cirpj

Reference architecture to integrate heterogeneous manufacturing systems for the digital thread

Moneer Helu(3)*, Thomas Hedberg Jr., Allison Barnard Feeney

Engineering Laboratory, National Institute of Standards and Technology (NIST), 100 Bureau Drive, Gaithersburg,MD 20899, USA

ARTICLE INFO

Article history:
Available online xxx

Keywords:
Data architecture
Digital thread
Systems integration

ABSTRACT

The increasing growth of digital technologies in manufacturing has provided industry with opportunities to improve its productivity and operations. One such opportunity is the digital thread, which links product lifecycle systems so that shared data may be used to improve design and manufacturing processes. The development of the digital thread has been challenged by the inherent difficulty of aggregating and applying context to data from heterogeneous systems across the product lifecycle. This paper presents a reference four-tiered architecture designed to manage the data generated by manufacturing systems for the digital thread. The architecture provides segregated access to internal and external clients, which protects intellectual property and other sensitive information, and enables the fusion of manufacturing and other product lifecycle data. We have implemented the architecture with a contract manufacturer and used it to generate knowledge and identify performance improvement opportunities that would otherwise be unobservable to a manufacturing decision maker.

CIRP.

1 Introduction

Digital technologies and solutions have grown tremendously in manufacturing. For example, only 8% of firms in 1999 used data warehouses to support their operations [1]. Today, industry has become focused on concepts like smart manufacturing, Industrie 4.0, and cyber-physical systems [2–4]. The steady adoption of data standards, such as MTConnect [5] and OPC Unified Architecture (OPC UA) [6], have enabled a growing market of digital solutions for data-driven, web-enabled manufacturing [4]. The increasing accessibility and growth of these standards and technologies has provided industry with opportunities to leverage data to reduce costs, improve productivity, ensure first-pass success, and augment existing workforce capabilities [2,4]. These opportunities can also address evolving industry challenges created by the increasingly distributed nature and growing complexity of modern manufacturing systems and global production networks [4].

One opportunity enabled by digital technologies that has been gaining attention is the “digital thread” concept. The digital thread links disparate systems across the product lifecycle and throughout the supply chain [7]. It enables the collection, transmission, and sharing of data and information between systems across the product lifecycle quickly, reliably, and safely. This concept drives data-driven applications that can generate domain-specific knowledge for decision support, requirements management, and ultimately improved diagnosis, prognosis, and control of design and manufacturing processes. Such capabilities address a significant need for manufacturing where traditional decision-making and paper-based processes often neglect the far-reaching implications of specific actions on the product lifecycle.

The development of the digital thread concept (and in fact many digital technologies) in manufacturing has been challenged by the inherent difficulty of aggregating and applying context to data from systems across the product lifecycle [4,7–9]. These systems generate and require various types of data of different formats stored using different means in different locations [7,9]. Commercial systems exist that attempt to organize product lifecycle data, but these systems often lock users into homogeneous suites of solutions throughout the enterprise. This can add additional expense onto products that are already extremely expensive and out of reach for many organizations, especially small-to-medium enterprises (SMEs). These solutions also do not often address the “silo effect” between engineering and manufacturing functions or between different organizations across the supply chain. There is a strong need for data infrastructures and management concepts that are scalable, integrate with heterogeneous systems, cut across many domains, and enable industry to determine where best to leverage data. The goal of this paper is to describe our development

* Corresponding author.
E-mail address: moneer.helu@nist.gov (M. Helu).

http://dx.doi.org/10.1016/j.cirpj.2017.04.002
1755-5817/CIRP.

and implementation of a reference architecture to enable the digital thread in manufacturing.

2 Background

Much of the research on the curation and use of data for decision support has focused on the development and implementation of applications rather than the management of data itself [1]. The lack of well-established architectures for digital manufacturing has limited the use of decision-support systems since an application cannot successfully generate knowledge without appropriately managing the flow and contextualization of data. Addressing this need of industry is especially important because traditional architectures in manufacturing have been challenged by the growing use of digital standards and technologies [4].

2.1 Traditional architectures in manufacturing

There is no single unified data architecture that is used across all industrial sectors of manufacturing [3,10]. Instead, firms have deployed one-, two-, and three-tier architectures to address the various use cases that each has faced [3]. An n-tier (or multilayer) architecture approach describes the separation and modularization of capabilities in a computing environment. These capabilities are referred to as logic mechanisms, which manage application commands, logical decisions, and computations as data moves between layers of the architecture. Separating logic mechanisms provides developers and users with flexibility when implementing and maintaining solutions to address needs and requirements. Specifically, developers and users do not need to develop new or redevelop entire applications whenever change is required. The primary goal of the n-tier architecture approach is to enable solutions within each tier that are specialized to a specific task needed to manage, contextualize, and present data.

Fig. 1 provides an overview of common one-, two-, and three-tier architectures. Systems within the lowest tier in each architecture (labelled "Data") curate the data that is to be accessed and potentially used for analysis. Systems within the highest tier in each architecture (labelled "Client") consume the data to provide knowledge to a user or external system. Systems within the middle tiers in each architecture provide services that manage the transformation, translation, and/or transaction of data between curators and consumers.

A typical one-tier architecture in manufacturing is composed of individual workstations, such as manufacturing equipment or computers, that both provide and consume data and information. There is little to no connectivity between these systems. For example, part programs may be generated on-machine or via computer-aided manufacturing (CAM) software on a computer and then copied to the machine using a storage device (e.g., floppy disk, memory stick). Such solutions are colloquially called "sneaker net," which describes the manual process of copying files from one system to another. One-tier architectures silo manufacturing systems and prevents these systems from coordinating activities.

Two-tier architectures begin to address the issues of one-tier architectures described previously. An example of a two-tier architecture in manufacturing is composed of a file or application server (e.g., to host a CAM license and generate G-code to control a machine tool) and client (e.g., the machine tool that consumes and uses the G-code). Each client typically exists on a network that enables it to communicate directly with databases or servers in Tier 2. Most logic mechanisms exist within the client, which can create additional computing burdens on these clients. While two-tier architectures provide some value (e.g., hosting part programs in a centrally-managed location), they do not usually provide sensing, monitoring, and control mechanisms outside of basic human observation. Any data that is collected must be extracted manually from the manufacturing system and entered manually into a server via a client terminal, which makes the use of this data for data-driven applications cumbersome and error prone.

Three-tier architectures build upon two-tier architectures by beginning to separate and modularize the functions and roles in information technology (IT) and operational technology (OT) systems. A typical three-tier architecture in manufacturing is composed of a database, application server, and thick client (i.e., dedicated software applications used to access other tiers, such as the user interface of SAP or Siemens Teamcenter). The application server provides the interface between the database and client by giving the client access to business logic mechanisms, which generate dynamic content from the data in Tier 1. By removing these logic mechanisms from the client, a three-tier architecture simplifies the design and use of the client and eases IT maintenance requirements and functionality upgrades. Three-tier architectures in manufacturing have given rise to product data management (PDM), manufacturing execution system (MES), and enterprise resource planning (ERP) solutions [11].

In the same way that IT systems in other domains have progressed from one- to two- to three-tier architectures, traditional architectures in manufacturing have evolved to better manage the data collected to support production operations. Despite this evolution, though, one-, two-, and three-tier architectures co-exist currently across industry. This lack of consistency has challenged the development of digital manufacturing because different types of content have different requirements for storage, processing, service, and observation [11]. These requirements have forced organizations to deploy different mechanisms and systems for each content type. This situation is also partly due to the lack of vertical integration between systems from the shop floor to the operations and enterprise levels [12]. The International Society for Automation (ISA) 95 standard [13] defines manufacturing systems as being composed of five levels: physical processes (Level 0), sensing and manipulating (Level 1), monitoring and control (Level 2), workflow and operations (Level 3), and business planning and logistics (Level 4). Typically, Level 1 comprises sensors and actuators, Level 2 comprises supervisory

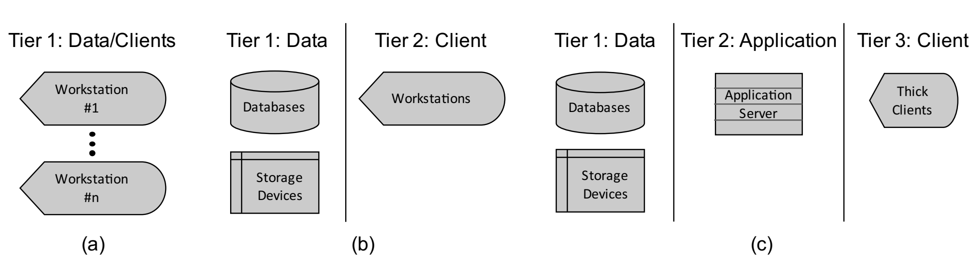

Fig. 1. Schematic examples of n-tier (or multilayer) architectures used in manufacturing: (a) one-tier; (b) two-tier; and (c) three-tier.