

The Challenges of Creating a Real-Time Data Management System for TRU-Mixed Waste at the Advanced Mixed Waste Treatment Plant

S. W. Paff, S. Doody
Contemporary Technologies, Inc.
444 Liberty Ave Suite 1800, Pittsburgh, PA 15222

ABSTRACT

This paper discusses the challenges associated with creating a data management system for waste tracking at the Advanced Mixed Waste Treatment Plant (AMWTP) at the Idaho National Engineering Lab (INEEL). The waste tracking system combines data from plant automation systems and decision points. The primary purpose of the system is to **provide information to enable the plant operators and engineers to assess the risks associated with each container and determine the best method of treating it**. It is also used to track the transuranic (TRU) waste containers as they move throughout the various processes at the plant. And finally, the goal of the system is to **support paperless shipments** of the waste to the Waste Isolation Pilot Plant (WIPP).

This paper describes the approach, methodologies, the underlying design of the database, and the challenges of creating the Data Management System (DMS) prior to completion of design and construction of a major plant. The system was built utilizing an Oracle database platform, and Oracle Forms 6i in client-server mode. The underlying data architecture is container-centric, with separate tables and objects for each type of analysis used to characterize the waste, including real-time radiography (RTR), non-destructive assay (NDA), head-space gas sampling and analysis (HSGS), visual examination (VE) and coring. The use of separate tables facilitated the construction of automatic interfaces with the analysis instruments that enabled direct data capture. Movements are tracked using a location system describing each waste container's current location and a history table tracking the container's movement history. The movement system is designed to interface both with radio-frequency bar-code devices and the plant's integrated control system (ICS). Collections of containers or information, such as batches, were created across the various types of analyses, which enabled a single, cohesive approach to be developed for verification and validation activities. The DMS includes general system functions, including task lists, electronic signature, non-conformance reports and message systems, that cut vertically across the remaining subsystems. Oracle's security features were utilized to ensure that only authorized users were allowed to log in, and to restrict access to system functionality according to user role.

Challenges encountered during the design and implementation of the DMS included the development of specifications prior to construction of the plant; constraints on design; project team maturation into the NQA-1 Quality Assurance world; lack of involvement of key stakeholders, such as the system operators; and difficulties associated with islands of knowledge unavailable to system designers. The approaches used to overcome these challenges included facilitated discussions between subject matter experts and analysts; comprehensive review of requirements with operators and other stakeholders; classification of requirements and issues; and cross-module design reviews to ensure that the system operated successfully.

INTRODUCTION

The Advanced Mixed Waste Treatment Project (AMWTP) is a highly automated treatment facility for TRU wastes currently moving into production at the INEEL in Idaho Falls, Idaho. During early project phases, British Nuclear Fuels Limited (BNFL), the main contractor on the project, recognized the need for a computer system capable of handling the complexity of real-time management in an automated facility. This system was needed to track all waste containers as they moved throughout the various processes in the plant. It was also needed to optimize the classification and shipping of the waste. And finally, it was necessary to create a paperless system to audit all container movements and transactions.

BNFL, working in conjunction with Contemporary Technologies, Inc., has developed the DMS computer system for AMWTP and are in the process of implementing the first of a three-phase project. This presentation describes the concepts, issues and solutions embodied in the approach to creating software systems to manage the complexities inherent in the management and processing of large numbers of TRU waste containers.

Historical Context

During the 1970s and 1980s, TRU wastes from a variety of sources were delivered for interim storage to the INEEL site in Idaho Falls, Idaho. Agreements between the State of Idaho and the DOE have resulted in the construction of the AMWTP to treat and ship this waste from INEEL to its final disposal site at the WIPP in New Mexico. In 1996, BNFL was contracted by the DOE to construct and operate the AMWTP. Construction of the plant was completed December 27, 2002, and work began to commission and begin operations. Further work to be completed includes the processing Facility operations, in which volume reduction and special waste treatment operations will occur. These systems are anticipated to be operational by the end of 2003. Grenville Harrop, BNFL Design Manager, describes the design in his WM '03 paper "Design and Construction of the Advanced Mixed Waste Treatment Facility".

The AMWTP Process

In its simplest form, the AMWTP process consists of three phases:

- Retrieval and Characterization of the TRU waste
- Treatment of the waste in order to meet WIPP requirements
- Waste Payload assembly for shipment to WIPP

Retrieval operations involve removing containers from a large berm of materials located at the Transuranic Storage Area/Retrieval Enclosure (TSA-RE) building. As the containers are removed from the berm, they are inspected, associated with any existing historical, i.e., acceptable knowledge (AK) information that would provide information on their contents, and readied for transport to the characterization building. At characterization, Real-Time Radiography (RTR), radioassay (NDA), and HSGS operations are utilized to provide information concerning the contents and hazards associated with the container. Further characterization techniques, including coring, and VE are performed on select containers. Verification and validation reviews are conducted for all of these analyses. Based on the container type, size and results of the analyses, a decision is made to either send the container back to the INEEL M&O, ship it directly to the WIPP, or process it first and then ship to the WIPP.

As seen in Figure 1, the DMS mirrors the physical processing at AMWTP and provides informational support for the decisions that must be made along the way. Each phase requires a large amount of data, which must be maintained, validated, and made available as appropriate for AMWTP personnel. Due to health and safety concerns associated with TRU waste, the TRU waste data must be carefully controlled and maintained in a secure and reliable manner. The high level of automation within the AMWTP complicates this task.

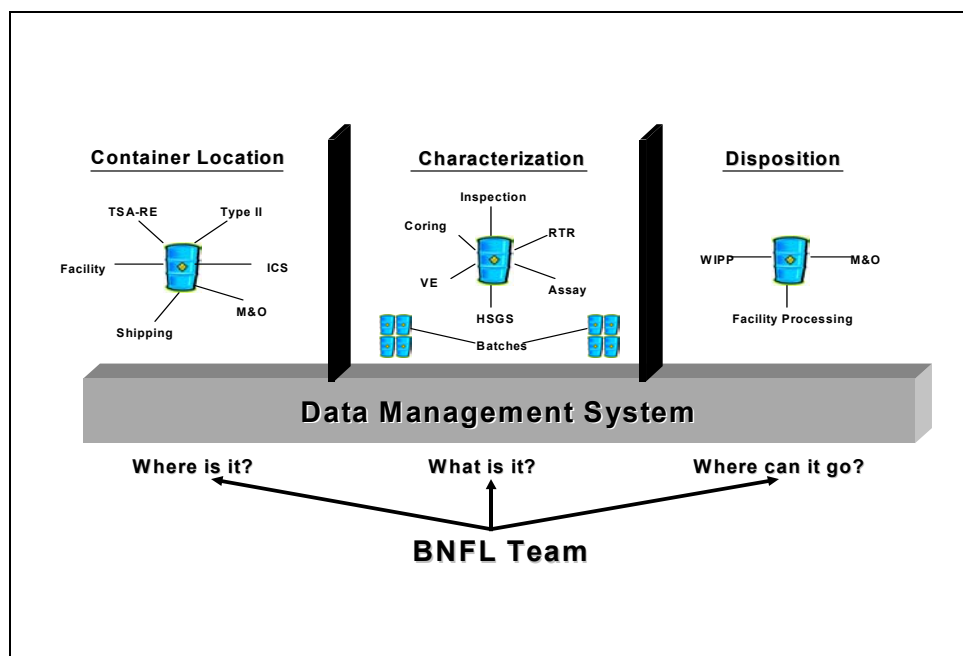


Fig. 1 DMS Purpose: The DMS mirrors the Physical processing of Materials throughout AMWTP

The DMS was designed and implemented to meet the NQA-1 quality assurance guidelines. These quality requirements necessitate adherence to standards and procedures for requirements gathering, design, coding and testing.

Unlike ordinary waste or analytical tracking systems, the DMS must operate in an environment where it is difficult, or even impossible, for an operator to physically contact the material. Therefore, the DMS has been designed to fully integrate with the plant's Integrated Control System (ICS). The ICS controls the physical placement and movements of materials. The integration of the DMS and ICS systems becomes complicated by the requirement to prevent movements in certain cases based on the results of analyses (for example, assay results that exceed the Fissile Gram Equivalent (FGE) limitations). Furthermore, because the analytical equipment utilized in the AMWTP is extremely complex and not readily available for system testing, the project represented special challenges for validation of the DMS Software.

Objectives of the DMS

The objectives of the DMS include the following:

- Track each waste container as it moves throughout the AMWTP;

- Track characterization results associated with each container;
- Support the verification and validation activities associated with each container;
- Provide basis for rejection/treatment of each container;
- Facilitate the determination of IDC codes, shipping codes, EPA waste codes and other information required for shipment;
- Provide information to support actions of the Integrated Control System (ICS);
- Provide information necessary to efficiently manage the plant and store the retrieved materials.

DMS DESIGN

The DMS package was designed principally as a client-server system with a series of interfaces to analytical equipment.

DMS Software Configuration

The AMWTP DMS was implemented using an Oracle 8i back-end database, and utilizing Oracle Forms and Reports 6i. Additional coding, for the RF-barcode system, for example, was written using C++ and a WaveLink™ middle tier. Interfaces were generally written using ODBC connections directly into the database. Individual workstations operate on a Windows 2000 platform, and the redundant servers operate with Windows and a Veritas fail-over system.

Design Approach

The DMS design was envisioned as a combination of vertical and horizontal functional areas. Vertical areas align with specific tasks or activities within the AMWTP process, such as an RTR station. Horizontal areas provide support for functionality across a broad set of vertical functions. These include electronic signature and verification and validation activities.

Figure 2 shows the overall flow of waste containers through the process. The vertical areas of functionality for the DMS correspond roughly to the boxes within this figure.. These areas include:

- Inspection Station
- RTR
- HSGS
- Drum Assay
- Box Assay
- Coring
- VE (miscertification, closure, and new waste VE)
- Infrastructure (container)
- Interfaces (e.g. to ICS or INEEL lab)
- Type II Storage Areas

The horizontal areas of functionality support the functions above. These subsystems include:

- Container routing (task lists and plans)
- Batch processing and reporting
- E-signature processing and rules for each vertical module
- Container move processing

- Verification and Validation (reporting and audit trails)
- Ad hoc reporting
- Messaging
- Login, security and user qualifications
- Labeling
- Overpacking and repackaging
- Non-conformance reporting
- Plant Optimization functions and reports

Like a checkerboard, the horizontal functional areas cut across the vertical, and also tend to exist on a layer of abstraction above the more physically oriented vertical areas. For example, it is easy to correlate an RTR analysis with a particular area of the plant, a particular operator, and the container associated with it. However, the verification and validation (V&V) of that RTR examination may be associated with V&V of a series of other RTR analyses, related only because they happened to occur during roughly the same time period.

The overall system design was, by necessity, built around a *container* object. This object was maintained using the container table, and given a unique container_id number. The container table acts as a repository for summary information gained concerning a container. Much of the detailed information for a container, as per typical normalized design constructs, is found in associated analyses tables, such as in the assay or RTR result tables. In some cases, select summary information, such as fissile_gram_equivalent values, was deemed critical enough to be associated directly with the container, even in violation of normalization principles. Thus, schematically, coming off the container table are a series of related tables, as shown in the high-level process model in Figure 3(a).

Figure 3(b) gives a simplified view of the architecture for a horizontal subsystem: in this case the batch V&V design that collects several analyses together and allows for verification and validation reviews.

The purpose and design of the system components are described below.

Vertical Subsystems

Table 1 provides a summary of the purpose, major tables, output and external interfaces associated with each vertical subsystem.

Horizontal Subsystems

The functionality of horizontal subsystems, by definition, cuts across the vertical functions. Some, such as batching and verification and validation, primarily support characterization functions. Others, such as movement, task lists and plans, directly support the physical routing of the containers through the plant. Still others, such as security and login, are more administrative and general. The purpose, basic design and interaction with vertical systems are summarized in Table 2.

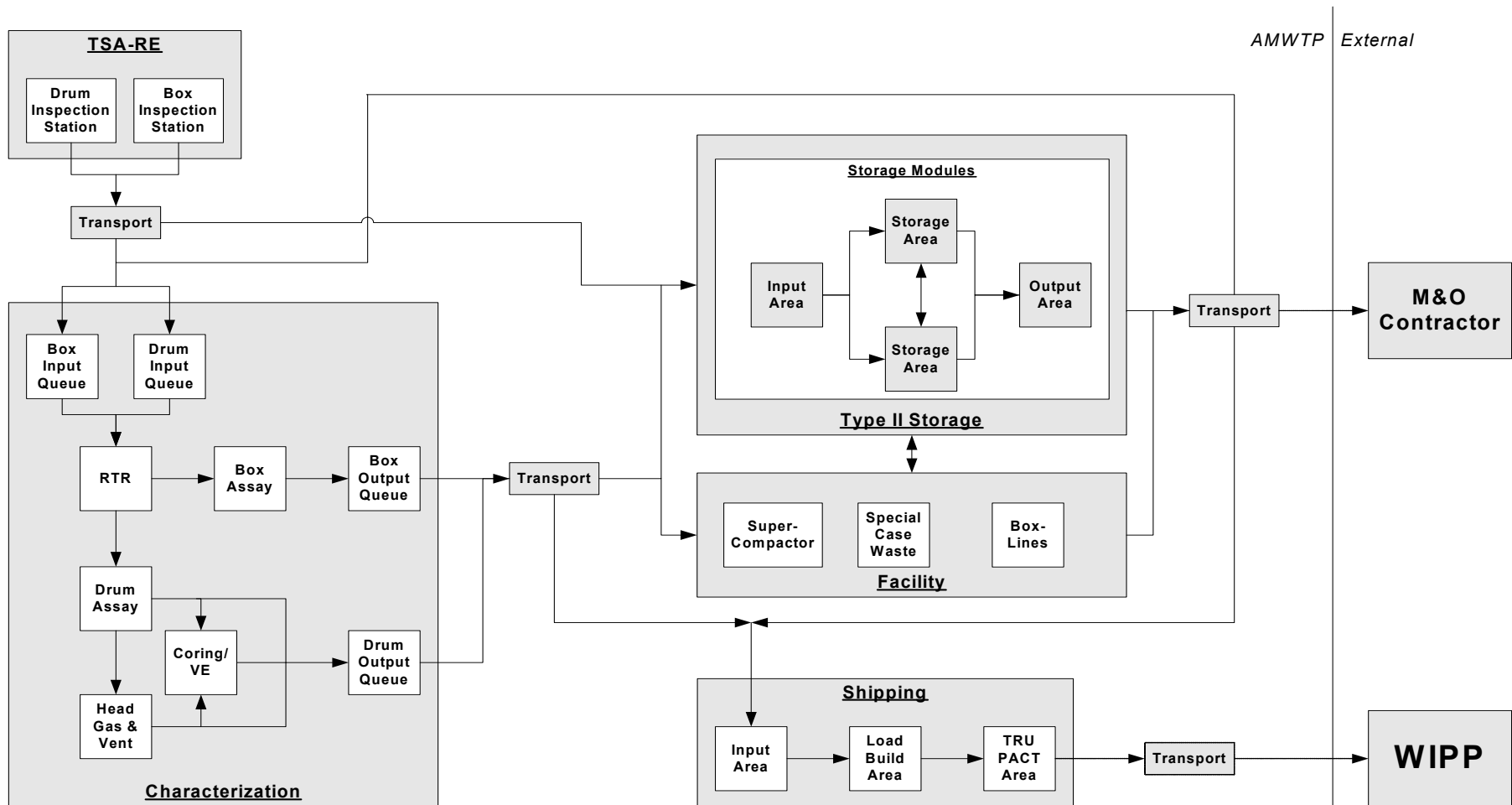


Fig. 2 Process flow diagram for a container through the AMWTP process.

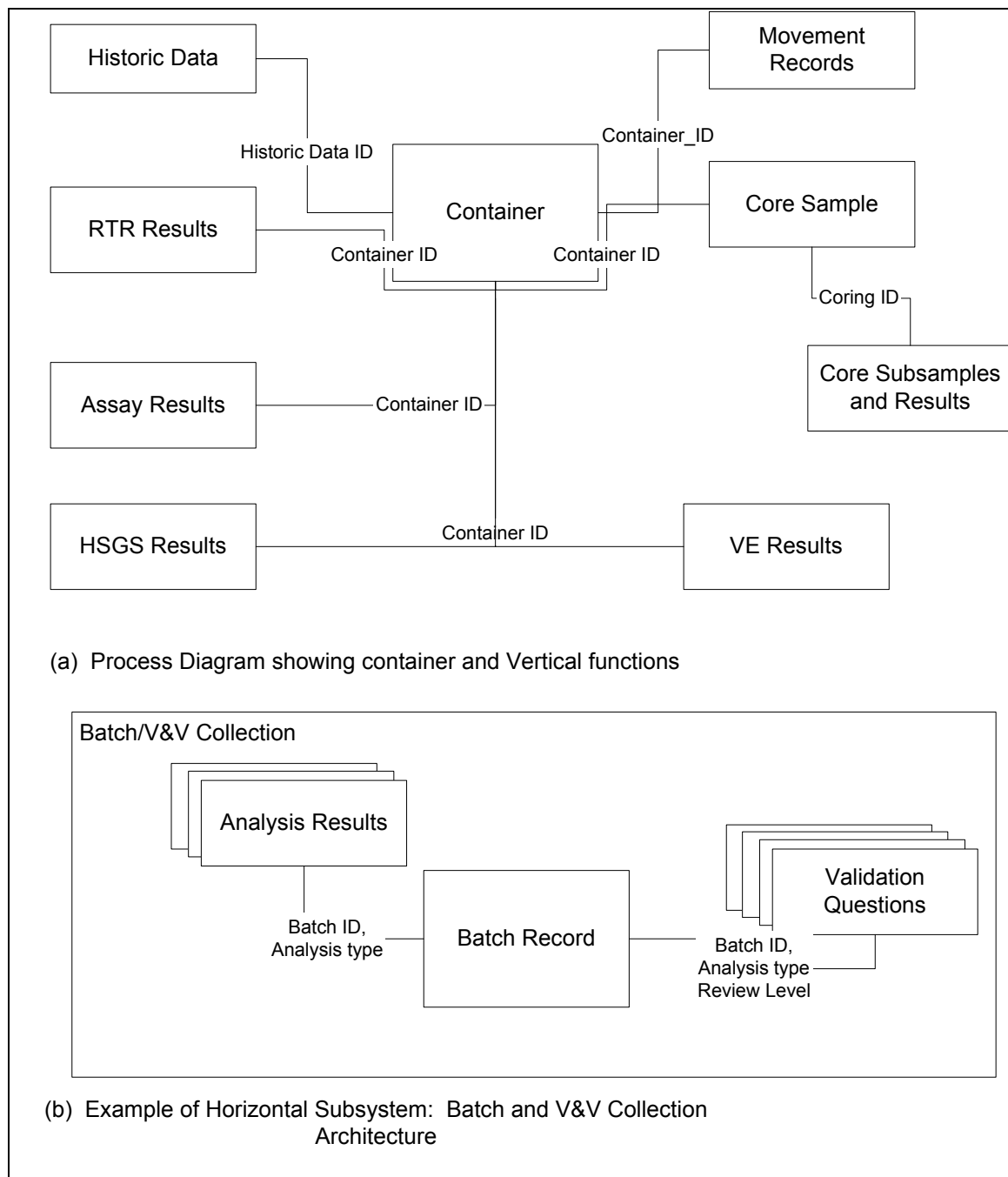


Fig. 3 Process Diagrams for vertical and horizontal functional areas.

Table 1: Summary design for vertical subsystems

Subsystem	Purpose	Major Tables	Major Outputs/ Decisions	External Interfaces
Inspection	Introduce containers into the DMS. Label, preliminary inspection.	Container	<ul style="list-style-type: none"> - Label the container - Preliminary determination of acceptability - Tie container to historic record - Determine need to overpack 	Historical data
RTR	Non-destructively determine drum contents	RTR_Results	<ul style="list-style-type: none"> - IDC Code - Exam Plan - Is container prohibited? 	None
HSGS	Measure organic output from container.	HSGS_Results,, various calibration tables	<ul style="list-style-type: none"> - What volatile organics are contained within container? - Is volatile organic concentration too high to allow safe shipment? 	HSGS Interface
Drum Assay	Non-destructively assess the radiological properties of a drum.	Assay_results	<ul style="list-style-type: none"> - Fissile gram equivalent (FGE) - Does drum meet safety constraints? 	Assay Interface
Box Assay	Non-destructively assess the radiological properties of a box.	Box_Assay_results	<ul style="list-style-type: none"> - Fissile gram equivalent - Does box meet safety constraints? 	Box Assay Interface
VE	Verify the RTR Results, or (for newly generated waste) maintain a record of container contents.	VE_Results	<ul style="list-style-type: none"> -Verify RTR result -Verify IDC 	None
Coring	Verify contents of a drum via a sampling technique	Core_Sample, Core_sample_Lot, Sub_sample, Sub_sample_Lab_Results	RCRA Data: Chemical analyses of a core, from the container.	Outside laboratory
Type II Storage	Track location of containers in storage	Movement_history	<ul style="list-style-type: none"> - Check container move permissions - Track movement history 	Bar-Code system

Table 2: Horizontal Subsystem Design Summary

Subsystem	Purpose	Interaction with Other Systems	Major Outputs/ Decisions
Container Routing (Task List and Plans)	Ensure each container is properly characterized according to regulatory and business rules.	Other systems provide input when tasks are completed. Also task results inform ICS/Move system prior to moves.	Directs ICS system to move the container to correct locations per analysis plan.
Batch Processing	Provides logical collection of analyses. Enforce analytical equipment rules regarding calibration, accuracy, drift and precision.	Batch opening/closing informed by analyses.	Logical collections of V&V activities. Prevents analyses when calibration criteria are not met.
Verification and Validation	Reviews batches of analyses for quality control purposes. Allows for reworking of results, or identification of invalid analyses.	Series of checklists created when batches are closed. Interacts with exam plan when analyses are rejected.	Were analyses conducted in an acceptable manner? Are the results consistent? Were the calculations performed correctly? Were the correct conclusions drawn?
Container Move Processing	Provides a mechanism (bar-code, manual or through control system) to allow for, and track container movements.	Interacts with storage locations, storage rules, container movement rules and other limitations.	Is container movement allowed? Is the destination location suitable, and does it have sufficient room?
Messaging	Provide messages to users or roles, generally pertaining to V&V or other logical activities (not physical movements).	Messages created directly based on V&V and other activities.	Allows users to plan their work activities. Provides direct channel to work that 'must be' completed.
Login, security and User Qualifications	Control access to the application. Control roles associated with each user.	Accessed by all users upon log-in. Re-checked when users electronically sign data.	Appropriate access to the appropriate users.
Labeling	Provide unique labels to containers. Replace labels when necessary.	Created at inspection, coring, and during repack operations.	Unique labels
Electronic Signature	Validate the identity of a user. Enforce rules regarding review and over-ride roles. Create historical record of what was signed.	Accessed in inspection, at each analysis, throughout V&V, and when supervisor overrides required.	Verifiable system, immutable record of data that was signed.
Non-conformance Reports	Provides ability to track non-conforming	Can be raised anywhere within the	Creates, tracks and maintains records of potentially adverse

	issues.	system, against containers, batches, equipment or any other aspect.	conditions. May prevent promotion of batches if NCR's are open.
Plant Optimization functions and reports	Allow for optimal placement of materials in storage, by category. Establishes storage areas. Support management decisions.	Inspection and post-characterization	Containers are placed at locations and grouped according to logical groups, such as FGE, weight, container types.

Some key features of the software, built to ensure not only that the software works under currently understood conditions, but under future conditions at the plant, include:

- Container and batch centered design to act as repositories for container information and for the logical/review information required at characterization
- Rule-driven movements that prevent/allow movements based on container properties and/or business rules at each location
- Verification and validation with flexibility as to the levels of reviews and the questions during each review
- Electronic signature system that maintains data regarding each signature in a separate, controlled, set of tables, but that itself is implementable utilizing relatively simple forms
- A radio-frequency bar-code system that parallels most physical processes allowed within the system
- Rule-based routing system to determine the characterization elements required for each container, with flexibility to change the characterization plan based on results at each step of the process.

These key features were designed to meet BNFL's current requirements and also give the project a tool that could be used to meet whatever future requirements were encountered.

SOFTWARE DEVELOPMENT METHODOLOGY

Initially, the project team envisioned utilizing existing packages for the DMS, either CTi's EnviroWare™ software (a waste tracking system designed initially for the hazardous waste industry), or the WEMS system from Rocky Flats, with minimal customization and changes. However, careful evaluation of these existing systems led to the conclusion that they were not immediately suitable for the AMWTP, and that so many changes would have to be made to them that it would be better to create a custom package to better meet the complex functional requirements.

The software was created under CTi's Quality Assurance Program, which is in compliance with the nuclear industry's NQA-1 Subpart 2.7 standards. The methodology employed by CTi in designing the system included the following standard software engineering steps:

- Requirements analysis
- Design
- Software development
- Testing

The original specifications received from BNFL did not have identifiable or traceable requirements. The requirements documents received by CTi were rewritten to include numbered, identifiable, testable requirements. This provided the foundation for tracing requirements from the requirements statements, through the design, software development and testing.

In addition to following a process for software development based on the requirements, CTi expended considerable effort *developing* the requirements for the project. CTi utilized the efforts of business analysts, who interviewed BNFL design, operations, TRU waste and QA oversight personnel to determine the system requirements.

Custom software development project includes not only the development of software, but the development of the process and procedures required to develop that software within the business structure and culture of the organizations involved. Moreover, like any other project in the nuclear world, the project was required to adhere to strict standards for quality assurance. Therefore, significant effort was expended to ensure a process that worked consistently and reproducibly between the two organizations – CTi and BNFL—that also maintained QA compliance. The primary focus of software QA-related activities was to maintain the discipline of tracing requirements throughout the process, and the methodologies developed included:

- Identification and enumeration of the requirements statements
- Procedures to clarify the requirements when they were not understood or inconsistent
- Analysis and design based on the requirements
- Design reviews
- Software development
- Test scenario and test case development
- Review of test cases with BNFL
- Formal testing
- Customer Acceptance Testing (CAT)
- Installation testing

Automated document scanning programs were developed using Visual Basic to ensure traceability and completeness from requirements through design and testing. Standard approaches, methods of review and sign-off were developed QA activity.

CHALLENGES

Similar to other complex effort involving large numbers of people and different organizations, the DMS development had its share of challenges. These included:

- Deceptive complexity required of the software
- Concurrent design
- Autonomous requirements development
- Development process and coordination

Each of these challenges is discussed in more detail below.

Deceptive Complexity

On the surface, the AMWTP process is very straightforward: (1) dig TRU waste containers up from the berm (or, easier still, take them out of an existing warehouse where they are safely

housed), (2) determine what they are, the hazards posed by them, and the integrity of the container, and (3) then either ship them directly, crush them and ship them inside another container, or return them to the M&O. The initial phases of the DMS included only the first two steps. And the requirements for the DMS were to provide a method to track the waste containers as they were unearthed and characterized.

However, the simple became far more complex. For example:

Historical Interfaces As the container is unearthed, it was necessary to match it with whatever historical AK data was available. This included not one, but potentially three, historical databases, and necessitated a method to interface with each. Moreover, the historical information is not to be directly associated with each container, but rather used as suggestions for BNFL experts during characterization.

RTR The container is sent through an RTR scan, primarily to determine what it is in gross terms (its IDC code), and to determine if there are prohibited items. The DMS was required to track the results of the RTR analysis. Seems simple enough. However, RTR analyses involve some inherently subjective judgments, which means that the RTR results must be reviewed by a technically qualified person, may be repeated, and may require additional confirmatory visual examinations. The RTR examinations themselves are grouped together into batches for review purposes, and these reviews may further complicate matters by necessitating the technician to rework an RTR analysis. Finally, interpretation of the RTR results, ostensibly straightforward, since the container is either to proceed through the process or not, is complicated since the presence of prohibited items require different pathways.

Assay The container is assayed, and the results are automatically sent from the assay instrument into the DMS, where they are stored, retrieved and reviewed. To this extent, the assay system is a straightforward interface. The complexity within the assay system occurs when you consider that the instrument calibration, and feedback between assay and RTR. Calibration standards require necessitates that each assay instrument pass calibration *prior* to a given set of assay results, and also *after* the results are completed. If any of the calibrations fail, a complex series of requirements govern whether the results can remain validated, or the contains require re-assaying. Assay results further inform the previously completed RTR. The assay machine determines the total weight for the container. This weight is passed back to the RTR system, which distributes the weight according to the weight parameter estimates made during RTR.

Coring A fraction of the containers are cored, with the cores analyzed to determine the hazardous constituents contained therein. Only some containers are to be cored, based on random selection techniques. Because coring involves opening the container, most cored containers will be visually examined, which means that the container selection for coring must be coordinated with that for visual examination (based on container contents). The container contents are determined at RTR (related to the so-called IDC codes). The *coring analyses* to be performed vary according to these contents – some containers are targeted for metals analyses, while others are targeted for volatile organics. The coring system is required to recommend the analyses to be performed on each core. Finally, the analytical results must be tracked back not only to the core, but also to the underlying containers.

Container Routing Container routing is determined and re-analyzed at each step of the process, based on information available at that time. This never-fully-determined approach is necessitated by the needs for analyses, but it complicates what is normally a straightforward question—where does this drum or box need to go? Additionally, many of the rules are complex, and require

contingencies for interaction. A container with batteries present is normally marked as 'Special Treatment'. The batteries will be removed during special treatment. Similarly, a container with a lead-lining is marked as 'Return to M&O'. But what if the same container has a lead-lined object and batteries? Which rule takes precedence? The routing system requires rules about the rules themselves to establish an order of precedence (in this case, 'Return to M&O' rules). Moreover, some container dispositions directly affect routing. A drum containing compressed gas canisters is sent to 'Special Treatment' to remove those canisters prior to assaying, but the assay is still required. But it would be a potential disaster if the same drum were marked for coring – where it will be subject to a drill bit that could puncture the compressed gas canisters. The routing system must guard against such exigencies.

A few design principles were employed to allow the DMS to tolerate the complexity.

- *Although the primary system was conceived as container-centered, each characterization step was treated as a separate sub-system.* This enabled the use of relatively simple data models where appropriate (a single parent-child system in the case of RTR, for example), and much more complex structures where the requirements necessitated them (Coring, which is a multi-level, multi-table system that tracks everything from container selection, to the taking of cores, to the chain of custody records, and analytical results)
- *Where possible, the concepts were built using rule-based approaches, with soft-coded features, rather than hard-coded rules and logic.* This allowed the necessary flexibility as the rules were developed. Batches were treated as logical collections of containers, with rules and criteria for opening, closing, and promoting the batches. The exam plan (routing) system was built as a container-centric task list, with the tasks marked off as they were completed, and simple procedures for adding required tasks to a container. This provided a framework for a rule-based system that implemented the more complex interactions among the tasks. The rules of precedence were built into the order in which the rules are applied, and controlled using a single parameter mechanism.
- *Interfaces were used to buffer complexity on either side.* The assay instrument includes a complex software package designed to interpret the various results received and determine the assay results. Although the process may use this information, the DMS receives only the *results* through its interface. The instrument performs all the required calculations, induction and system checks. Likewise, the integrated control system checks with the DMS prior to making moves. Although the DMS may require complex checks to determine whether a move is appropriate, it returns a simple 'Yea' or 'Nay' to the move, and lets the ICS respond accordingly. This approach tended to reduce the complexity within the DMS.

These approaches mitigated the affect of the system complexity without compromising too much flexibility.

Concurrent Design

The DMS was not designed for a well-understood, existing plant. The DMS development often paralleled – or even preceded—the plant construction, and it generally preceded development of operating plans for the plant. The AMWTP was a new, unique facility. In some cases, as questions developed within the DMS design team, it became apparent that the necessary expertise was unavailable, or perhaps not even hired yet. Thus, it was initially difficult to determine business or operations requirements. For example, the coring system was redesigned multiple

times because the necessary process experts were not available to the project until the system had been designed and even coded to earlier specifications. The box assay system was originally envisioned to be identical to that for drum assay. However, it quickly became apparent that the system was not the same, and a new specification was written to encompass its requirements.

Project schedule pressures forced the DMS team to build many sub-systems when the requirements were suspected to be incomplete or incorrect. Normally, under such circumstances, CTi would deploy a series of rapid prototypes that articulated a basic design according to the requirements as understood at the time. These prototypes would then be reviewed by the relevant experts, and used to convey additional requirements or changes to the design. This standard approach was challenged by the schedule pressures for the project, which often demanded that the software be 'fully operational' although the requirements were not fully understood. Moreover, the QA requirements to produce a system traceable back to requirements presented a further challenge to the prototype approach, because prototypes are by definition not fully tested and designed.

The concurrent design led to many instances where subsystems were specified, designed, built and then had to be rebuilt one or more times. This included instrument interfaces, V&V, Coring, assay, NCR, routing and HSGS.

CTi and BNFL utilized formal mechanisms to dialog about requirements, while still maintaining QA compliance. The formal apparatus, the BNFL Request for Information (RFI), allowed CTi to ask questions and confirm new or dropped requirements in an on-going manner. Once a base set of requirements was enumerated, the system allowed only drops and adds of new requirements (changes to a given requirement were expressed as a drop and an add) to simplify traceability. The software verification and validation plan (SVVP) was used as a scoping tool to identify the set of requirements associated with a given release of the DMS. Various other custom-built, automated tools were used to confirm that these requirements were addressed both in the system design, and in the test cases. As the tests were completed for any given release, a traceability matrix was developed that tracked the requirements from the SVVP, through the design (and therefore the software based on that design) and into the test cases. Thus, it was possible to show that a given requirement was not only designed for, but tested, and the test passed or failed.

This process allowed for the inevitable changes to the software during development. It was coupled by a series of in-depth analyses of various subsystems (starting with coring and head-space gas analysis). The changes produced by these activities were incorporated into the next revision of the SVVP, and subsequent releases of the software as appropriate.

Autonomous Requirements Development

System requirements were initially developed and reviewed by a group of subject matter experts. The requirements documents tended to be developed individually, however. This led to differences in terminology and meaning that complicated the development of a coherent, integrated system. The term 'Batch Closing', for example, appeared to have multiple different meanings depending on which specification it was found in. In addition, in many cases, what was presented as requirement statements were actually a series of design proposals.

To address this issue, CTi initially organized the analyses efforts with individuals assigned broad areas, rather than one or two specifications. This, at a minimum, enabled the identification of many of the disconnects within the specifications.

Over time, analysis teams were employed at CTi, with analogous teams ('blue ribbon panels' of experts) at BNFL. This combination enabled both groups to identify disconnects and produce a much more coherent, integrated system.

Development Process and Coordination

The final challenge for development occurred because both BNFL and CTi underestimated the level of effort required to develop the DMS. The teams were initially relatively small, and team sizes were stepped up only as necessary. Over time, the CTi team was organized into three major groups: Analysis and Design, Development, and Testing. Support activities (DBA, documentation, software configuration, and QA) were provided across the teams. The project management structure included a project manager and the team leads from each of the main teams. As the software was released on site, a commissioning/training team was deployed to assist BNFL in implementation.

On the BNFL side, the initial configuration included primarily the BNFL design team. This team was quickly augmented with staff from the TRU waste group, who provided expertise on system requirements and procedures. As additional operations personnel were hired, they joined the team. Thus, the eventual configuration included members from design, TRU waste, and operations with QA oversight.

This project organization improved the quality and pace of development of the system, and ensured that changes were reviewed by the appropriate team(s) of people.

Overall, therefore, the project overcame the challenges presented to it, and produced a sophisticated software that will enable BNFL to operate the AMWTP.

CONCLUSIONS

The development of the DMS provided many lessons learned. The key broad, organizational/management recommendations include:

- Start as early as possible;
- Involve all key stakeholders as early as possible in the project;
- Focus and define the requirements

Further more software-specific recommendations include:

- Remember that custom software development requires customization and control of the *development process* itself. The process should be established early in the project, and maintained throughout.
- Assure that careful attention to QA requirements should be maintained throughout the development effort. This includes ensuring traceability of the requirements throughout the development process.
- Do not accept unverified requirements.
- Aim for a clear articulation of the requirements. Identify the appropriate subject matter experts, and team them with business analysts to gain an understanding of what the system is required to do at each step along the way.
- Beware of deceptively simple processes. Engineers taking a first cut at requirements often leave out 'gotchas' and project/process exceptions.

- Build flexibility wherever reasonable. Some of the successful, flexible, encapsulated elements of the DMS design include:
 - Batch system
 - Electronic signature
 - Verification and validation for the various characterization steps
 - Container routing (exam plan)
 - Container location system
 - Movement rules
- Avoid flexibility where it will obscure the design. (In other words, ensure that the design includes autonomous or semi-autonomous subsystems where this is appropriate.) In this case, that was in the development of separate characterization modules, each with their own set of tables, for RTR, assay, coring, VE and HSGS.
- Develop test scenarios early in the process and utilize them to confirm the design team's understanding of the requirements.

In spite of the challenges, or perhaps because they needed to be overcome, the BNFL and CTi teams have developed a close working relationship that has enabled the project to move forward. It appears that the AMWTP is poised for successful operation, with DMS as a central element during operations. The BNFL and CTi teams are confident of the ability of the DMS to provide the necessary support throughout the AMWTP project.