

## **Evolution of a Waste Information System - 9541**

D. Speed  
URS Washington Division  
P.O. Box 2078, Carlsbad, New Mexico 88221

### **ABSTRACT**

Managing information has become a pervasive task in our society and business activities. This is especially true in the arena of government facilities and nuclear materials.

Accomplishing the required tasks is not sufficient in the new millennium; plans are made, reviewed and approved, specifications for materials are developed, materials are procured and delivered, inspected, invoices are audited and paid. Activities are conducted to procedures with embedded quality checks and a final turn-over inspection is performed.

In order to make the most efficient use of our human capital, we turn to machines to assist us in managing the information flood. How best to address this task? This is new territory – there was no prior art at this level. The challenge is to exercise an appropriate level of control, and at the same time, add value. The key to accomplishing this goal is having a good team with a carefully engineered processes applying an appropriate level of automation.

At the Waste Isolation Pilot Plant (WIPP), information is managed about the facility, its performance (environmental monitoring), mining operations, facility services, cyber security, human resources, business processes, and waste information. This paper addresses experience gained with the management of waste information over the first decade of operation.

The WIPP Waste Information System (WWIS)<sup>(1)</sup> was created to fill both a gatekeeper function to screen waste for disposal at Waste Isolation Pilot Plant (WIPP) and the official record of the properties of the waste contained in the WIPP transuranic waste repository. The WWIS has been a very successful system as the monitor of waste acceptance criteria and data integrity; it is an integral part of the success of the WIPP operation.

The WWIS is now in its thirteenth year of operation. This period has included close regulatory scrutiny as a part of determining facility readiness for initial waste acceptance, and more than 40 significant software revisions to accommodate changes in the growing body of knowledge for waste disposal at WIPP. At the present time, the experience gained in the last ten years is being applied to the design of a replacement for this system which is being developed in parallel with ongoing disposal operations.

The core of the WWIS is a relational database which focuses on the waste container as the central object of interest. (container-centric). This database is the source of the information used for the final acceptance checks and tracking of the transuranic waste to be sent to the WIPP. The ‘front end’ or user interface of the database is the WWIS application, an Oracle Forms based program. The user submitted data consists of ten relational data tables with one hundred two separate data elements to describe a waste container and its contents. The requirements for information needed to maintain data integrity and enforce business logic increases these elements to nearly one thousand. Currently, information contained in the system exceeds one hundred seventy four thousand containers and seven thousand shipments; there are more than a million rows in several internal database tables.

This paper presents a discussion of the history of the system, changes implemented, and lessons learned.

## **HISTORY OF THE WWIS**

From its beginnings in the early 1980's as a simple tabulation of waste data, the WWIS (WIPP Waste Information System) has grown from a simple tool for the storage of waste data, to include logic for data integrity checking, regulatory compliance checking, production enhancement, tracking and accounting tools with interconnections to many external systems supporting users around the country. This paper is a brief description of this evolutionary process and lessons learned during this effort.

The WWIS formally began operations in 1996 in support of the WIPP site readiness review and waste receipt began in 1999. Shortly thereafter began a period of change which has continued to the present. The WWIS has averaged three major and several minor revisions to the package annually for a total of forty-three distinct software versions. The original gatekeeper function for WIPP Waste Acceptance Criteria (WAC) which formed the basis of the 1996 version has been combined with additional logic to add functionality:

- packaging (transportation) parameter checking (2000)
- terminal server system (2001)
- tracking packaging compliance for non- WIPP shipments (2008)
- real-time product tracking at WIPP facility with barcode (2008)
- reuse of TRAMPAC (TRU Waste Authorized Methods for Payload Control) logic at the Advanced Mixed Waste Treatment Plant in Idaho (2004)
- reuse of TRAMPAC logic at Hanford (2006)

The challenge to meet evolving operational needs and make needed improvements to the system with a small team was significant. An opportunity was sought to add to the original WWIS functions to better serve both the operational and regulatory users. In late 2007, a conceptual model of a replacement for the current WWIS package was prepared to communicate key concepts to the WWIS user community and stimulate discussion about requirements for a new system.

## **SYSTEM DESCRIPTION**

The WWIS is a container-centric system built using a multi-user database management system enables a team of users to submit the data elements, such as radioisotope data and headspace gas sample data which describe the waste's properties in logical increments, verify the waste's qualifications to be disposed, and finally assemble containers into transportation compliant payloads for shipment.

The Oracle database management system (DBMS) is the framework upon which the software is constructed. This package has performed very well; this was an appropriate choice for the system when the WWIS was built, and remains the system of choice. There has not been a situation when the software was limited by the choice of this for a base system. The Oracle Forms based WWIS application filled the original needs well, and served as a base for the expanded functions.

WWIS is a multi-user system (around three hundred authorized users at any given time) which allows staff to simultaneously work at the tasks of entering container data, reviewing/approving data, building overpacks/payloads/shipments, emplacing waste in the repository and querying the data in the database. By careful construction of the internal data integrity rules, the necessary constraints can be applied to data elements so that the different members of the team can work together closely to meet the waste disposal

requirements. This also enables team members to track the containers, overpacks, assemblies and shipments while in transit and finally into receipt and emplacement at the WIPP repository.

The client-server paradigm used by the WWIS keeps the data in the central repository but moves a copy to the user's desktop for some operations. This requires the client (WWIS application) software to be installed on the users desktop. The task of maintaining several hundred desktop installations in multiple DOE facilities across the breadth of the continent was very resource intensive, even when the user's aid was enlisted. With the advent of tighter cyber security and lessened user privileges, this became untenable.

The 1996 revision of the WWIS was based on the Client-Server model; changing Oracle software architecture left WIPP (and many others) with an unsupported system in 2007. Changing this required a major system redesign/rebuild to move to a different system design paradigm; there was no simple path forward. Scheduling and resource constraints complicated planning and execution of this effort.

## **DATA MOVEMENT TO THE WWIS**

In order for waste to be sent to WIPP for disposal, it is first characterized on the basis of generating processes designated as 'waste streams'. Waste stream profile forms are submitted by the generator sites for review at WIPP; data submitted for review as part of this effort is stored in the WWIS. Subsequently, the parametric requirements for the waste stream are added into the WWIS reference tables for the checks of subsequent waste container data from that waste stream.

The next step is to begin the collection of data on the properties for individual waste containers. This data is used to certify that waste is eligible for disposal at the WIPP repository. Waste bound for WIPP passes through multiple levels of review before shipment authorization is granted. Prior to being entered into the WWIS, the data is reviewed and approved at the generator sites. Once submitted to the WWIS, the data is reviewed at the container level, during the overpacking process if applicable, and again at the payload level. Checks on the data entered into the WWIS fall into several major categories:

- data integrity checks,
- checks for compliance with the Waste acceptance Criteria (WAC),
- checks for compliance with the WIPP site Documented Safety Analysis, and
- checks for compliance with the Type B packaging Certificate of Compliance (TRAMPAC).

Data integrity checks are exemplified by requirements like the uniqueness of a container identification number, or whether the total weight of the container is within 5% of the sum of the individual material weights, etc. Waste Acceptance Criteria checks verify that the waste is from an approved waste stream, is in an approved container, does not have any hazard codes prohibited at the WIPP site. Documented Safety Analysis (DSA) checks address parameters associated with the facility hazards assessment, for example PE Curies (Plutonium Equivalent curies – an index of hazard from inhalation from a hypothetical container breach). The TRAMPAC criteria is primarily associated with hydrogen gas generation from radiolysis.

These checks can be run either upon demand by the user, or automatically, when the user determines that the data is finalized and submits the container for final approval. These checks are separate, and run sequentially. The WWIS logic completes as many data checks as possible with a data set versus stopping the data checks when an errant value was detected; this way each checking session returned as much information to the user as possible.

Finally, waste containers, payloads and shipments are compared to the published packaging criteria (TRAMPAC) to determine if all information indicates a safe configuration for transport to the repository. Shipments meeting the regulatory criteria are sent to the WIPP repository. Upon receipt at the repository, the shipment's contents are checked against the database to verify that the proper containers have been sent. After verification, the waste containers are moved to their emplacement location in the WIPP underground galleries and their locations stored for future reference.

This data and the practices used to generate it are audited routinely by both internal and external organizations to ensure the quality of the product.

## **WWIS LOGIC EXTENSIONS**

The logical structure of the WWIS has been changed in order to add functionality. The initial WWIS logic was fairly simplistic and inflexible. Prior to actual operations, there was not an appreciation of the complexity of the problems associated with meeting the overlapping regulatory requirements which govern the assessment, packaging, movement and disposal of these wastes. Several significant changes to the waste disposal process have occurred since the system was put into production; largely, the repository requirements changes have been adaptive, rather than corrective. The original logic was correct, but too simplistic.

At the request of the Department of Energy's Carlsbad Field office (CBFO), logic was added to the WWIS to address overpack creation and payload management needs. Each of these requests added complexity to the underlying database structure in order to ensure that the combined containers met repository and transportation requirements.

Overpacking is the process of placing legacy waste containers that do not have the proper integrity for shipping into a more suitable outer container for handling and shipping. As a significant portion of transuranic waste is legacy waste (i.e. generated at some time in the past and stored, awaiting a repository), there has been some container degradation requiring the original waste containers to be combined into an overpack. The solution to this was to make the overpack container data a virtual construct and apply the test requirements to the virtual package. Another example is the modification of the Fissile Gram Equivalent (FGE) limits for packages to address the presence or absence of neutron reflectors which have an effect on this limit.

Payload management is the process of combining wastes managed as transuranic waste, but being assessed in the final repository assay as slightly less than 100 nCurie/gram with higher value waste so that the waste payload container has a value of > 100 nCurie/gram.

With the advent of the nineteenth revision of the TRAMPAC, the calculations necessary to determine compliance with the TRUPACT-II Type B shipping container had become very burdensome. A computer code (eTRAMPAC) was developed to assist the sites in complying with the TRAMPAC. The WWIS team was asked to support these checks with the data that was contained in electronic format in the WWIS. This led to the development of the Contact Handled TRAMPAC Evaluation Software (CHTES) and related packages beginning in the 2003 time frame. There have been several stepwise improvements in the methodology used to calculate compliance with the TRAMPAC, each adding to the complexity of the model. For example, early modeling forced multiple parameters into a single two dimensional reference table. The result was conservatism in the calculations. This was addressed in the Version 5.0 WWIS release that supported the normalization of the TRAMPAC reference tables and produced more accurate results and more efficient shipments.

Shortly after the CHTES completion, the system logic was extended to handle the disposal and transportation of Remote Handled (RH) transuranic waste. The introduction of Remote Handled (RH) waste in 2007 required the inclusion of a significant addition in the system logic in order to be able to handle this different class of waste and two different Type B package configurations (72-B and 10-160B).

## **MANAGING CHANGE**

For most of the first ten years of operation, the combined tasks of operation and development of the system was a serious challenge. The rate of change was very high with an average of four software releases per year. Innovative management changes enhanced the waste flow to the WIPP repository this challenged the software to meet the scheduled changes while maintaining the required quality standards.

Several tactics were employed to aid in coping with the high change rate. To the extent practicable, system modifications were constructed so that logic changes could be implemented by changing values in reference tables, rather than in the logic. This was the first move towards the achieving the agility needed to match the needs of the disposal process. Even with these tactics, it became apparent that modifications to the system development philosophy were needed to move from reactive to proactive.

Rebuilding of the TRAMPAC logic in 2005 was an opportunity to start moving the system into a planned structure. Prior to this time, development efforts had been directed on an immediate need basis without a long range plan. With the benefit of operational experience, an architectural plan for a future system was developed where progress on the overall system could be made one building block at a time. This plan was followed with good results.

## **FACTORS INFLUENCING CHANGE**

In order to manage a system in an environment with rapid change, it is helpful to understand the driving forces in order to anticipate and control the effects of the change when possible. There are forces which act to move a system from its initial configuration.

Building the system logic into a structured object oriented system where there was a logical separation between the user interface logic, the business rules logic and the database logic, one piece at a time was a challenge, but progress was made daily. Changes became less interdependent and easier to test. Clearer requirements enabled both a better product and better (more accurate and faster) testing.

Business rules describe the definitions, logic and constraints that are applied by an organization in achieving its goals. The inflexibility of early business rules proved to be an unnecessary impediment to waste disposal, and it was recognized that change was needed. Initially, the repository database was designed to permit no change to the data; a container had to be deleted in order to have data related to the container to be changed. There was little appreciation of the need to change data associated with waste that had been submitted, approved or emplaced. For example, when the WWIS was implemented, the waste characterization process was envisioned as being sole responsibility of the waste generator, that is, a single certified entity would perform the waste qualification measurements. As the programs matured, it was found to be advantageous to allow two or more groups to perform different parts of the measurement suite. The changes needed to manage the checks of these activities in the database application are also adaptive.

The WWIS supports a number of customers with differing needs. These can be categorized as; Regulators, Managers, Waste Certifiers, Waste Shippers, Waste Disposal Ops, and Business users. Data

is made available to a user based upon needs. The information available to the regulator users sets is based upon the applicable regulations. The regulatory body charged with regulating radioisotopes can see the radioisotope data; the regulatory body charged with regulating packaging compliance can see the packaging compliance information. Data views or queries are tailored for each regulator's area of responsibility. This policy helps keep users focused on their requirements.

The operating environment refers to the hardware/software environment which is used to host the system and move information. The version of the WWIS software which went into production was originally developed in 1996 - the Windows 3.1 era. Vendor support for the version of the database management system (DBMS) and application building tools (Oracle Forms) used has been formally discontinued. The replacement system has been designed in accordance with the architectural principles document adopted in 2004. This standard stresses a consistent and compatible software stack, flexibility, modular design, minimized dependence on proprietary solutions. Adherence to this paradigm has enabled a favorable leverage position to the recent CH and RH packaging check software into an easy reuse of a complex body of Java code which has conserved considerable time and resources..

## **LESSONS LEARNED**

The following is a brief summary of the major lessons learned.

Maintain a close connection to your customer: Interaction with the WWIS user base, both at their home sites and at periodic offsite meetings helped to gain and maintain understanding of operational needs. To date, six user focus meetings have been held. With a registered user base of around three hundred individuals, there have been between fifty and sixty attendees to each session. Special sessions to demonstrate new logic, such as overpacking have been held.

Plan for change. Nothing lasts forever, and the future cannot reliably be predicted. Build the system in logical pieces that can be changed and tested in a timely fashion. Start with a clear statement of purpose. Be very specific in the expectation of what the software will and will not do. It is essential that each program element (or in software engineering parlance, requirement) be referenced directly to the document that drives that requirement.

Adopt a strong process and follow it. The adoption of the rigorous IEEE software development process<sup>(2)</sup> has resulted in a better capture of issues which has enabled more accurate, responsive and documented changes. Operational experience in managing the transportation and repository data identified the need for additional system functionality. For example, increased error checking sub-routines to catch typographic errors has minimized rework in the data input end for manual data input. In addition, the adoption of a customized software requirements tracking package helped issue and resolution tracking immensely.

Efforts put into the system planning and documentation were evolutionary. The development efforts for this system were typical of the norm during a period of evolution in software engineering, and somewhat less rigorous than those required today. At the same time, the overlapping regulatory requirements for both WWIS data content, limits and reporting were also dynamic throughout this developmental period as the actual requirements were being negotiated.

Focus on requirements: Software is the implementation of a set of rules, or, in IEEE terminology, requirements. Most owners of a process understand their requirements very well, but are unable to verbalize them in a complete and succinct manner. The resources spent on the front end to gain a complete and in-depth understanding of the rules which govern a process cannot be over-stated. Having a

system that tracks these complex system requirements and assists in zeroing in on changes needed to resolve issues is a necessity that is not widely understood.

Measure performance: The performance of the system is not simply the number of milliseconds it takes to run a simple or complex query (though this is also useful information). Human performance can also be measured. Where are the questions coming from? Where are mistakes being made? These measurements also provide vital information that is beneficial to track and analyze in order to improve the product.

Logic is unforgiving: Unless the effort is made to allow for options and contingencies, software is very merciless about allowing exceptions. The need for these alternative paths must be captured in requirements and carefully built into the system. This is not a place for fuzzy thinking.

A quality staff is of paramount importance: Over the years, a number of individuals and organizations have supported the WWIS development efforts. The effectiveness of these elements varied greatly. Careful selection and cultivation of these resources played a major factor in the system's success.

Shift of mission focus: A higher level of benefit was reached with the acceptance of the fact that the data system could be more than just a hurdle to be crossed. With the backing of the CBFO, development moved to include the use the knowledge in the system to assist the user in completing their tasks by making use of the knowledge in the system (decision support). As the original system charter was as a gatekeeper rather than a dual role of gatekeeper/facilitator, this transition came later in the process than necessary.

## **WDS SYSTEM DESCRIPTION**

Based upon the high-level lessons described above, a replacement system, the Waste Data System (WDS) has been designed and implementation begun. The new system incorporates the original functionality of the WWIS and goes beyond. The following is a brief description of the new system and the elements which have been implemented.

The WDS was designed and programmed with separation of the logic which determines the user interface from the business rules and further separation from the database. This architecture allows for maximum flexibility and allows more rapid modification and testing.

The WDS database was built using the Oracle version 10g R2 relational database. This system was chosen for several reasons. First, it met WIPP needs. Second, WIPP owned licenses for the software, and had a significant investment in staff training for this package. Oracle is still the standard enterprise class database management system for performance, scalability and security. A decision was made to limit the system build to constructs that are supported only by the Oracle package. In general, this move was made to allow for a reasonable migration to another DBMS package in the future, if warranted.

Both the User Interface and the Business Logic elements were developed with Java using an Object Oriented architecture, and developed on the Eclipse IDE platform. This Open Source toolset is an industry standard and enhances production, documentation and testing.

Using this structure for the WDS completes the implementation of the design plan initiated in late 2004 with the forerunner of the CHTES package. The significant investment made in this architecture over the years was reused with minimal modification. Content delivery is web-based using Open Source tool sets.

This is advantageous in that code reuse becomes very feasible. The downside is that Java programmers can be more difficult to find in some markets at this time..

Beyond the Information Technology (IT) side, the WDS design philosophy changes reflect the lessons learned. Development of WDS adopted a 'data delivery' paradigm. Instead of presenting the data in a standard format of 'one size fits all', the needs of the user base were analyzed and an effort to determine what information was needed to be able to do their jobs efficiently, and how they needed to be able to find this information. This is the starting point of the dashboard design – the dashboard being a metaphor for putting the information needed to operate the vehicle in front of the operator. For the Waste Certification Officials (WCOs), the login screen shows them the containers which they have ownership in, and offers suggestions for payload configurations.

WDS introduces tools to assist the process in addition to the original compliance checks. In the past, the WCO/TCO population used the compliance checks as a gauge; with the manual system available to them, this was very labor intensive.

WDS implements increased data qualification requirements and thereby reduces errors. Over the years, there has been a steady increase in the number of data integrity checks to catch data entry errors in the early stages of the process. As a compliance tool, making a determination of acceptability is a clear decision – it passes or it fails. But there is more information we can pass to the user about the container submitted based upon the system logic. For example, a filter installation date must occur after a drum closure date.

WDS introduces decision support tools. By targeting the needs of the users with dedicated dashboards we can meet our goal of moving from eighty percent custom queries and twenty percent canned reports to twenty percent custom queries and eighty percent canned support. This will reduce resource needs by focusing users on core information. Today, multiple individuals maintain SQL scripts which report on the shipments made during a period of time. Over the years, each of these scripts was requested individually, and more often than not, with slight differences in specification. This leads to several problems: first, different questions are bound to return different answers, and the efforts needed to reconcile these minute differences (e.g., was a shipment received or emplaced during a particular week?) are not productive. Second, these queries are, at present, uncontrolled; they are not tracked or documented, mostly due to resource constraints. This presents potential issues when version changes are implemented and the (now out-dated) scripts return unexpected values.

WDS also adds communications tools. Adopting a lesson from the software development tools used in the project, a function to allow actions/questions to be assigned from user to user. These functions have proven their value in helping to get timely notification of the user's needs.

WDS adds user-friendly query tools. The new database structure is more normalized for clarity and efficiency. Unfortunately, this does not make it easier for humans to join the tables together to get the information that they seek. In order to bridge this gap and move away from the custom queries to the extent possible, a custom query engine was added to the system for selected users. The initial version of the software will have a limited capability; future enhancements will be added based upon developing needs.

WDS also changes the change adoption paradigm. The introduction of new elements in the system is being introduced in the manner consistent with that recommended by the *Information Technology Infrastructure Library* (ITIL) Consortium<sup>(3)</sup>. The decision support tools beyond the very basic are being introduced slowly and simply, to allow an evaluation of their effectiveness while minimizing the impact on the user base.



## CONCLUSION

The management of transuranic waste is a data intensive activity that has evolved over the last decade. Careful planning is necessary to assist in starting a project with the tools needed to effectively collect, validate, and communicate the information needed to safely and efficiently manage radioactive wastes. Following the lessons learned noted in this paper will assist in producing the needed tools with an efficient use of resources and assist in the delivery of a product which adds value to the enterprise.

## REFERENCES

1. Washington TRU Solutions LLC, DOE/CBFO-97-2273, Revision 15 WIPP Waste Information System Users Manual, May 15, 2008
2. IEEE Software Engineering Standards Collection, Institute of Electrical and Electronics Engineers, Inc. New York, NY, 2003.
3. *Information Technology Infrastructure Library (ITIL) Consortium*  
<http://www.itil-officialsite.com/home/home.asp>