



# SEER\*DMS

---

June 8, 2004



# Rationale

---

- Many registries have DMS systems developed as far back as mid 1980's that are nearing the end of their hardware & software life cycles
- Unified SEER DMS, including all core registry functions, implemented in individual registries would save development/implementation costs and have the following extra benefits:
  - Decreased system maintenance
  - Ability to quickly enhance and modify the DMS for all SEER registries
  - Increased sharing of knowledge and experience between registries
  - Improved data quality and consistency
  - Increased flexibility in data definition and collection
  - Increased efficiency and timeliness of data submission and reporting



# Summary

---

- Appliance Solution
- Modular Design
- Extensible and Scalable
- Architecture separates various layers
  - Client/user
  - Web/presentation
  - Object – tumor set, patient set
  - Database
- Workflow
- Standardization
- Customizable

# Design, Development, Deployment

---

- Project Definition
- Requirements Analysis/Needs Assessment
  - Business models – object, process and socio-political models
  - Business use cases
- Architecture Recommendation and Initial Design Recommendations
- External Validation and Input from NCICB
- Design and Development Phase
  - Wireframes – site map, module flows, navigational prototypes
  - UML models and design documents
  - Testing (proof of concept, unit, integration) and traceability
- System Deployment and Implementation
  - Analysis of current registry computer environment
  - Data migration
  - Training – initial, detailed, advanced
  - Physical deployment
    - Set up hardware and integrate with existing infrastructure
    - Convert production data
    - Load data and verify
    - Perform training
    - Testing
    - Conduct post-implementation evaluation
- System Enhancement and Maintenance

# System Components

---

## **Completed**

\*Data submissions

\*Screening

\*Matching

\*Report Engine

Auxiliary Data

Follow-back

Record Requests

Edit Engine

System Administration

## **Under Development**

Visual Editing

Consolidation

Individual Reports

Online Help

\* Components that will be demonstrated



# System Architecture

---

- 3 tiered client/server
- Server(s) runs under Linux OS on Intel hardware
- Underlying RDBMS is Oracle
- J2EE application server is JBOSS
- The system components written in JAVA
- Client - browser based