Sonoma Technology, Inc.
Air Quality Research and Innovative Solutions

THE INTERAGENCY FUELS TREATMENT DECISION SUPPORT SYSTEM (IFT-DSS) SOFTWARE ARCHITECTURE

Draft Software Architecture Design
STI-908038.04-3565

By:
Tami H. Funk
Roger G. Corman
Judd E. Reed
Sean M. Raffuse
Neil J.M. Wheeler
Sonoma Technology, Inc.
Petaluma, CA

Prepared for:
The Joint Fire Science Program
Boise, ID

February 2009

Sonoma Technology, Inc.
Air Quality Research and Innovative Solutions

THE INTERAGENCY FUELS TREATMENT DECISION SUPPORT SYSTEM (IFT-DSS) SOFTWARE ARCHITECTURE

Draft Software Architecture Design
STI-908038.04-3565

By:
Tami H. Funk
Roger G. Corman
Judd E. Reed
Sean M. Raffuse
Neil J.M. Wheeler
Sonoma Technology, Inc.
1455 N. McDowell Blvd., Suite D
Petaluma, CA 94954-6503

Prepared for:
The Joint Fire Science Program
3833 South Development Avenue
Boise, ID 83705

February 12, 2009

ACKNOWLEDGMENTS

[Mike Rauscher to draft this section]

VERSION CONTROL

Draft Version 1	February 2, 2009
Draft Version 2	February 12, 2009

TABLE OF CONTENTS

Section **Page**

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

ES.1 INTRODUCTION

During the past decade, software applications and analysis tools have proliferated in the fire and fuels management community. Due to the heterogeneity of available data, software systems, ad hoc tools, and workflow processes in the fire and fuels community, awareness of, access to, and distribution of data and software tools have decentralized. As a result, fuels treatment specialists and decision makers are left with an assortment of unconnected systems in various stages of development and little guidance with respect to the strengths and weaknesses of these systems. The Joint Fire Science Program (JFSP) staff, working in concert with the National Interagency Fuels Coordination Group (NIFCG), initiated the Software Tools and Systems Study (STS) in 2007 to develop a software architecture design for an Interagency Fuels Treatment Decision Support System (IFT-DSS) to assist the fire and fuels community and to provide a framework to help address the issues. The overall goals of the IFT-DSS are to

- simplify the fuels treatment planning decision support process and improve the overall quality of analysis and planning by more easily combining and reusing applications and providing new opportunities for data analysis and collaboration;
- encourage scientific collaboration by providing a framework, registration mechanism, and tools that allow and facilitate the integration of new software applications into the framework;
- and to promote interagency collaboration within the fire and fuels community.

ES.2 THE FUELS TREATMENT PLANNING DECISION SUPPORT PROCESS

At the most basic level, the fuels treatment decision support process centers on managing outcomes associated with disturbances in vegetation. The process to arrive at an outcome involves six general steps:

1. **Define project, landscape, vegetation data, and scale –** The appropriate vegetation data, data parameters, geographic coverage, and resolution must be identified to meet the analysis objectives.
2. **Prepare and quality-assure current year data –** In many cases, the fuels treatment specialist must manipulate raw vegetation data to produce a data set that is more representative of real-world conditions and/or that provides geographic coverage adequate to meet the analysis objectives.
3. **Simulate and analyze fire behavior –** Fire behavior simulations are performed to assess fire effects and/or values at risk.
4. **Analyze fire effects and/or risk –** Fire effects and risk assessments are sometimes performed to identify and prioritize geographic areas that may warrant treatment strategies.
5. **Design treatment strategies –** Treatment strategies are often developed to achieve desired target conditions (i.e., ecological, geophysical, socioeconomic).

6. **Simulate vegetation, geophysical, and fuel conditions** – Vegetation or weather conditions (or both) are iteratively simulated as needed to assess treatment effects.

The IFT-DSS architecture supports the six steps in the decision support process and is designed to include the data, software applications, and tools used for fuels treatment planning.

ES.3 THE IFT-DSS ARCHITECTURE

The IFT-DSS will integrate disparate vegetation data, vegetation simulators, fire behavior and effects models, and risk analysis tools (some of which are web-based) into a common graphical user interface (GUI). It will support the reuse of IFT-DSS applications and services over the Internet, and it will be a flexible, modern, web-friendly system. The overall architectural design approach for the IFT-DSS is based on the concept of service oriented architecture (SOA). SOA facilitates the integration of disparate software systems and applications by deploying them as distinct units, or services, that users can easily access, combine, and reuse as needed. SOA facilitates the integration of new and legacy software applications to streamline work processes. This architectural approach can also support inter-operability with other decision support systems in the fire and fuels domain such as the BlueSky Framework and the Wildland Fire Decision Support System (WFDSS).

The core of the IFT-DSS software architecture consists of five elements: (1) a multi-layered GUI, (2) a control database, (3) a dispatcher, (4) data and software services, and (5) a geospatial database. The following describes the general function(s) of the five key architecture components:

- **Graphical User Interface (GUI).** The GUI is the user access point into the IFT-DSS. It controls the user experience and provides the portal for all system inputs and outputs.
- **Control Database.** The control database stores all information that is input to the IFT-DSS and manages the user experience and the decision support process.
- **Dispatcher.** The dispatcher works with the control database to invoke specific functions and services to support the needs of the high-level decision support process.
- **Data and Software Services.** The fire and fuels treatment domain data, software models, and tools that support the decision support process needs.
- **Geospatial Database.** The geospatial database is the data management hub of the IFT-DSS. It stores all project-related input, intermediate, and output data. It is called the geospatial database because it can manage spatial data (map layers) as well as tabular data.

Figure ES-1 illustrates the five key components of the architecture and their relative architectural arrangement. Each component contains interconnected sub-components, or layers, that perform specific functions within the system.

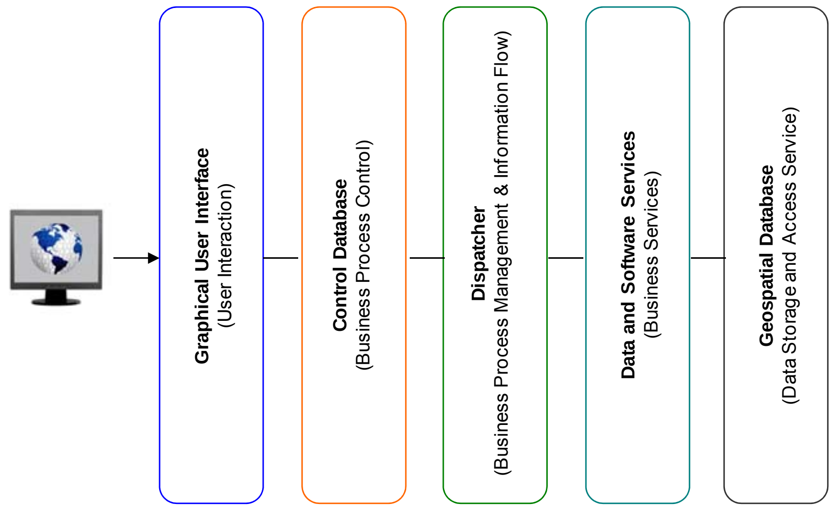

Figure ES-1. Illustration of the five key architecture components of the IFT-DSS.

The success of the IFT-DSS will depend on two key factors: (1) the participation, cooperation, and assistance of key stakeholder groups including fuels treatment specialists, developers of fire and fuels applications, fire and fuels data providers, and the National Wildfire Coordinating Group (NWCG) IT Investment Process coordinators and (2) community-wide adoption of standards for data, metadata, and application programming interfaces (APIs) to facilitate data and software application integration.

ES.4 IFT-DSS IMPLEMENTATION

It is expected that full functionality of the IFT-DSS could be implemented over the course of three years beginning in spring 2009. The first year of implementation (2009) could involve the development of a functional IFT-DSS prototype to serve as a proof of concept (POC) system. Assuming on-going support, full implementation of the architecture and the addition of system enhancements could occur in the second year (2010). By the beginning of the third year, the IFT-DSS could be fully functional and operational, and the focus in the third year of development could be system enhancement and fine-tuning. By the beginning of the third year, the IFT-DSS could be a stable system that can be transferred to the long-term hosting agency. The system developers could continue to enhance and fine-tune the system during Year 3 and could upgrade the operational system to the hosting agency by the end of 2012.

1. INTRODUCTION

A proliferation of software systems and analysis tools in the fire and fuels management domain over the past decade has made the task of managing and guiding the development of new tools and data sets increasingly challenging. Due to the heterogeneity of available data, data formats, software systems, ad-hoc tools, and workflow processes in the fire and fuels community, awareness, access, and distribution of data and software tools has decentralized. As a result, fuels treatment specialists and decision makers are left with an assortment of unconnected systems in various stages of development and little guidance with respect to the strengths and weaknesses of these systems.

The interagency Joint Fire Science Program (JFSP), through formal and informal interactions with its partners and clients, became convinced that the need for an integrated software architecture framework to manage the many models and data sets is a pressing issue currently facing fire and fuels analysts and decision makers. Acting in concert with the National Interagency Fuels Coordination Group (NIFCG), the JFSP initiated the Software Tools and Systems Study (STS) in 2007. The JFSP funded the Carnegie Mellon Software Engineering Institute (SEI) to perform a strategic analysis of the problem. This analysis was completed in March 2008, and SEI submitted a written report to the JFSP. A key finding of the SEI study was that the fire and fuels management community would greatly benefit from a software platform and systems architecture that support integration and collaboration.

Following Phase I of the STS study, the JFSP initiated Phase II of the STS study with the objective of designing a software architecture for an interagency fuels treatment decision support system (IFT-DSS) that would provide a framework for organizing and integrating the many data and applications that serve the fuels treatment community. The Interagency Fuels Treatment Work Group (IFTWG) was formed to help guide the Phase II project and a team from Sonoma Technology, Inc. (STI) was commissioned to help.

The overall process for software and systems development involves five general steps as illustrated in **Figure 1-1.** The first step is the development of the system architecture design; the subsequent steps are (2) preparation of a detailed design specification document; (3) system implementation and development; (4) testing and refinement; and (5) system deployment. Phase II of the STS represents the first step in the overall system development process—the design of the system architecture as indicated in Figure 1-1.

The software architecture design was developed through three general activities: (1) the development of the IFT-DSS conceptual design, (2) a current practices and needs assessment of the fuels treatment community, and (3) identification and development of the desired system functional characteristics and requirements, culminating in the development of the software architecture design. **Table 1-1** summarizes the activities and deliverables that make up Phase II of the STS. This document is the final deliverable of Phase II of the STS.

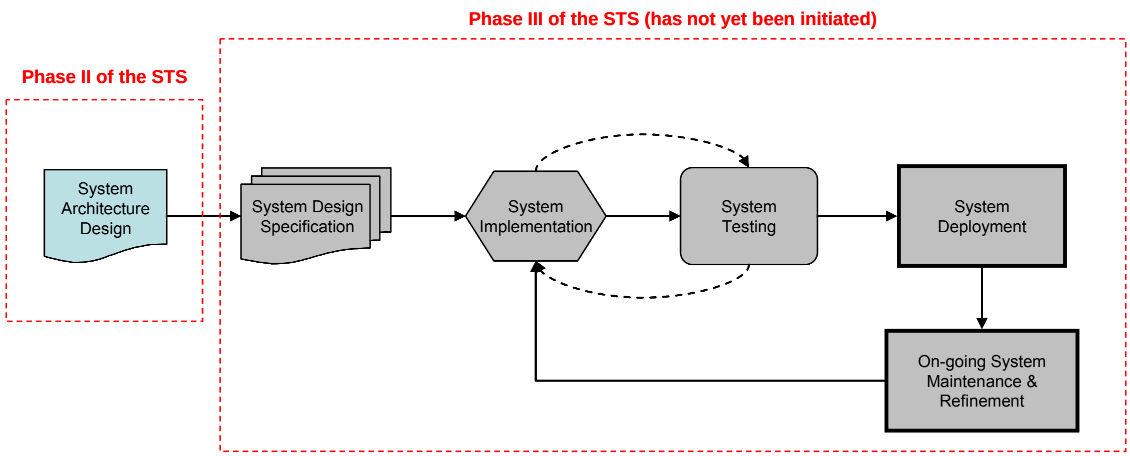

Figure 1-1. Overall process for software and systems development.

At the onset of Phase II of the STS, the JFSP and the IFTWG developed a vision and conceptual design for the IFT-DSS (Joint Fire Science Program, 2008). To ensure that the vision and conceptual design are consistent with current fuels treatment planning practices and that the IFT-DSS will support the needs of the fuels treatment community, the JFSP, the IFTWG, and STI worked collaboratively to assess the current practices and needs of the fuels treatment community. The product of the current practices and needs assessment was a technical memorandum documenting the activities and findings of the current practices and needs assessment (Funk et al., 2008).

In parallel with the current practices and needs assessment, STI built on the findings from Phase I of the STS and the work of the SEI to perform an assessment of existing decision support systems both within and outside of the fire and fuels domain. The purpose of the evaluation was to understand how other research communities have addressed the issue of integrating disparate software systems and tools through the use of service oriented architecture (SOA) technology. A secondary objective was to identify the functions and characteristics of other software systems that are desirable for the IFT-DSS and to understand how those functions and characteristics have been implemented from a software architecture perspective. The final product of the system evaluation was a draft paper discussing the assessment of selected SOA systems (Funk et al., 2008).

As shown in Figure 1-1, Phase III of the STS, which has not yet been initiated, will start with the preparation of a detailed system design specification document and implementation, testing, deployment, and on-going maintenance of the IFT-DSS. This Software Architecture Design document (this document) is intended to guide the design specification to be developed at the beginning of Phase III of the STS. The design specification document should include a more detailed view of the system, including implementation specifications such as the hardware and software deployment model, programming language specifications, data format specifications, and a detailed system implementation and deployment plan.

Table 1-1. Summary of the activities and documentation completed during Phase II of the STS study. The documents listed in the table below are available online at <http://frames.nbii.gov/portal/server.pt>.

Activity/Task	Objective	Deliverable/Documentation
Development of the IFT-DSS conceptual design document	To document the decision support process, current practices, and needs of fuels treatment community.	IFT-DSS Conceptual Design Document (JFSP, 2008)
Current practices and needs assessment	To confirm (and understand in more detail) the decision support process, current practices, and needs of the fuels treatment community.	Technical Memorandum: Findings of the Current Practices and Needs Assessment for the Interagency Fuels Treatment Decision Support System (IFT-DSS) Project (Funk et al., 2008)
Identify and document the desired functional characteristics and requirements for the IFT-DSS	To examine other distributed and collaborative decision support systems; identify those functional characteristics that are desired for the IFT-DSS; examine the software architecture approach to support desired functions.	Draft Paper: The Application of Service Oriented Software Architectures in the Fuels Treatment Community (Funk et al., 2008)
Develop a software architecture design for the IFT-DSS	To develop a strategic-level architecture design for the IFT-DSS.	IFT-DSS Software Architecture (Funk et al., 2008) **(this document)**

1.1 OVERVIEW OF THIS DOCUMENT

This software architecture design document provides a conceptual architectural overview of the IFT-DSS. It is organized in a way that provides different views of the IFT-DSS and system components and layers to varying degrees of detail. This document describes the functional and structural vision of the IFT-DSS. It defines what the system will do, the processes by which it will perform tasks, and the system components.

This document does not contain an exhaustive list of system artifacts and their characteristics. These include specific data sets, software applications, and tools that will later become integrated with the IFT-DSS. The data sets and software applications have been inventoried and cataloged. This document describes the strategic-level IFT-DSS architectural approach. It is important to realize that this document is a work in progress, and modifications to the proposed approaches should be expected. This document represents the current state of the IFT-DSS proposed architecture.

Section 1 of this document provides an overview of the STS study, the IFT-DSS stakeholder groups, and the benefits that the system will provide, an overview of the fuels treatment decision support process, and the users of the IFT-DSS. Section 2 discusses the architectural goals, challenges, and requirements for the IFT-DSS and the factors that will determine the ultimate success of the system.

Section 3 describes the user experience and how the user will interact with the system during a typical project analysis session. Section 4 presents the architectural approach and describes the system components and their functionality. Section 5 discusses the need for data and software standards to support scientific collaboration within the fire and fuels community and how it relates to the IFT-DSS. Section 6 presents a summary of architectural decisions, assumptions, and implications for the IFT-DSS. Section 7 includes a proposed implementation plan for Phase III of the STS and Section 8 includes citations and references.

1.2 IFT-DSS STAKEHOLDERS

Phase II of the STS has been a collaborative effort and has involved the active participation of four key IFT-DSS stakeholder groups. The success of the IFT-DSS depends on continued active collaboration and active engagement with each of the stakeholder groups. It is vitally important that these stakeholder groups are engaged throughout the IFT-DSS design and development process. During Phase II of the STS, relationships have been established with representatives from each of the stakeholder groups. These groups should continue to be actively involved throughout the design, implementation, testing, and evolution of the IFT-DSS. The following describes the key stakeholder groups and their involvement in Phase II of the STS:

- **Fuels Treatment Specialists** (approximately 1,000 throughout the United States). A group of 40+ fuels treatment specialists has been formed to review and provide feedback on the design and development of the IFT-DSS. During implementation, it is intended that a sub-set of this group will provide early system testing and feedback.
- **Scientific Collaborators** (approximately 20 throughout the United States). Scientific collaborators are those individuals or teams that develop new scientific algorithms, software, and tools to aid the fuels treatment planning community. During Phase II of the STS, relationships were established with a sub-set of fire and fuels scientists and model developers (i.e., ArcFuels developer, INFORMS developer, IFP-LANDFIRE developers, Starfire developer, and other independent software model developers). Representation from this group should be expanded and included in the design and implementation of the IFT-DSS. The IFT-DSS system development team should work with scientific collaborators to ensure that analysis methods, models, and tools are properly implemented within the IFT-DSS. In addition, this group should be involved in establishing scientific collaboration guidelines and data standards for making models and tools available to the system.
- **Institutional Data Provider(s)**. Institutional data providers are large-scale programs such as LANDFIRE that may eventually provide data to the IFT-DSS system. This group should continue to be engaged in the IFT-DSS process and relationships should be further developed to foster inter-program collaboration.
- **Information Technology (IT) Administrators** (representing the IT Investment process and the hosting agency administrators). IT administrators will eventually administer and

maintain the IFT-DSS at the hosting agency (to be determined). During Phase II of the STS, the IT Investment team has been engaged to help the IFT-DSS architecture design team understand the interagency IT investment process requirements to ensure that the system will meet those requirements. They should continue to be engaged throughout the implementation effort.

1.3 IFT-DSS STAKEHOLDER BENEFITS

The overall goal of the IFT-DSS is to provide a software framework for organizing and managing the many existing data sets, software models, and tools available for fuels treatment planning and analysis and to foster collaboration within the community. The IFT-DSS software architecture was designed to provide the following key stakeholder benefits:

- Integration, guidance, and collaboration regarding the use of existing data, software models, and tools available for fuels treatment analysis and planning.
- An overall gain in productivity and efficiency in the fuels treatment planning process by developing a system that can eliminate the time associated with preparing and manipulating data and software applications.
- A framework that facilitates peer review and model validation for scientific algorithms and applications and a mechanism to more efficiently review, critique, and provide feedback to help improve the scientific work flow and decision support process.
- A framework to facilitate access, use, and validation of the many existing vegetation data sets and a mechanism to more efficiently review, critique, and provide feedback to the data providers concerning strengths and weaknesses found in the data; and
- A central framework that IT administrators can ensure is appropriately secure when needed and appropriately open for access when needed as well as functionally effective and user friendly. Organization of a myriad of software systems into a framework that meets the needs of users, IT security specialists, and managers equally well.
- Another vehicle to enhance interagency functionality and collaboration and to serve as a testing ground for identifying and testing acceptable governance issues that best support interagency operations.

1.4 THE FUELS TREATMENT PLANNING DECISION SUPPORT PROCESS

At the most basic level, the fuels treatment decision support process involves performing environmental assessments of fuels treatment options as mandated by regulatory requirements such as (but not limited to) the National Environmental Policy Act (NEPA). This process centers on managing outcomes associated with vegetation disturbances. The processes required to arrive at an outcome involve preparing a detailed vegetation dataset, modeling vegetation changes based on growth, treatments, and/or disturbance, and analyzing the outcomes of the modeled vegetation. The fuels treatment specialist then recommends which treatment option to apply. The following six steps define the fuels treatment planning decision support process:

1. **Define project, landscape, vegetation data, and scale –** Prior to a fuels treatment analysis, the appropriate vegetation data (i.e., treelist vegetation data versus gridded vegetation data, data parameters, geographic coverage, and resolution) must be identified to meet the analysis objectives. Geophysical data such as topography and weather data may also be required. For some use cases, no explicit data are input at all, and those data that are required are input directly by the user relying on expert judgment.
2. **Prepare and quality-assure current year data –** In many cases, the fuels treatment specialist must manipulate raw vegetation data to produce a data set that is more representative of real world conditions and/or that provides geographic coverage adequate to meet the analysis objectives. This step might include imputation of treelist data for a multi-unit analysis and/or updating gridded LANDFIRE data to better reflect actual vegetation conditions.
3. **Simulate and analyze fire behavior –** Next, an analyst might simulate fire behavior or effects of fire on values at risk either for a vegetation unit or on a landscape (spatial data) representing current or post-treatment vegetation conditions.
4. **Analyze fire effects and/or risk –** The analyst might then analyze fire effects to identify and prioritize areas that may warrant treatment strategies. If a risk analysis is required, the analyst would use risk assessment tool(s) to identify those areas within a landscape that are at risk given specified risk parameters or values. Expert judgment is then used to identify those areas at greatest risk and prioritize treatment areas.
5. **Design treatment strategies –** Based on the results of the fire behavior modeling and risk analysis, the analyst would determine if geophysical, ecological, and/or socioeconomic target conditions have been met. If target conditions are met, the analyst may conclude the analysis. If the target conditions are not met, the analyst would design treatment strategies and re-simulate vegetation or weather conditions (or both) to assess treatment effects for a single unit or landscape.
6. **Simulate vegetation, geophysical, and fuel conditions –** In the event that target conditions are not met, the analyst would re-simulate vegetation or weather conditions (or both) to assess treatment effects for a single unit or landscape. This step is repeated as necessary based on different treatment strategies or weather conditions.

1.5 USE CASES

During Phase II of the STS, the following use cases were identified:

Use Case 1 Comprehensive risk analyses and treatment strategies are applied for multiple vegetation units or landscape. This is an end-to-end analysis that involves identifying treatment areas and designing and evaluating treatment strategies following a risk assessment.

Use Case 2 Analyses performed at the strategic management level to identify and prioritize treatment areas for multiple vegetation units or landscape. The objective is to identify and prioritize areas within a landscape where treatment may be warranted for planning purposes. Treatment strategies are not applied or evaluated in this use case.

Use Case 3 Analyses to simulate treatment over time for a single vegetation unit or collection of individual units without taking spatial juxtaposition into account. The objective is to understand the fire and fuels related characteristics of specific fuels treatments and the duration of effectiveness for each vegetation unit.

Use Case 4 Analyses to determine under what weather conditions a prescribed burn should be conducted on a single vegetation unit.

More use cases may be identified or evolve over time. The IFT-DSS software architecture is designed to accommodate new use cases. **Figure 1-2** shows a work flow diagram developed to illustrate the decision support process and process steps. It represents the pathways that encompass the use cases identified above.

Many data sets, software applications, and tools have been developed to assist in the fuels treatment decision process. An inventory of the most commonly used data, software applications and systems, and tools was compiled as part of Phase II of the STS and is shown in **Table 1-2**. **Table 1-3** lists the data, software applications, and tools from Table 1-2 and where they are used in the decision support process.

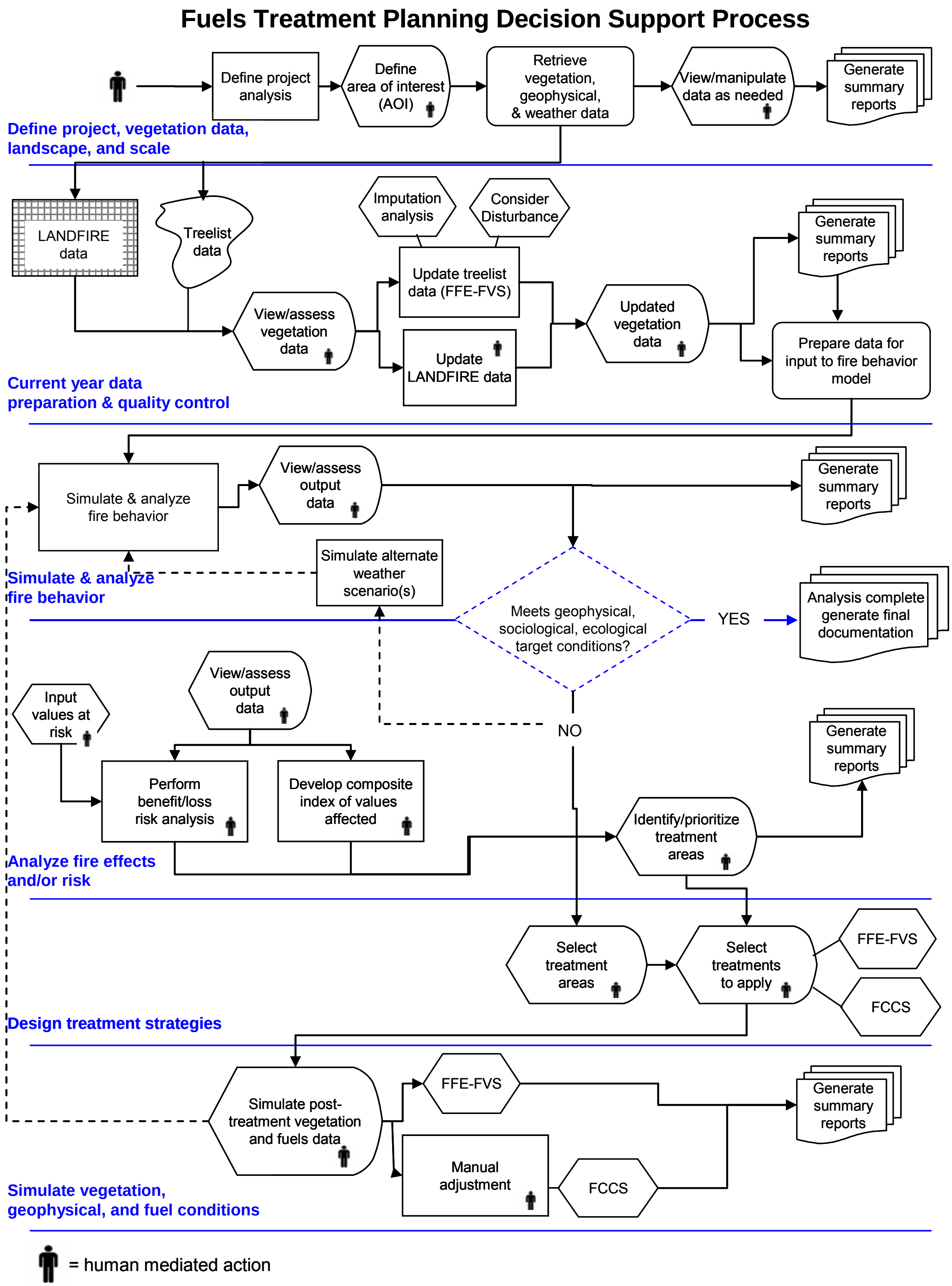

Figure 1-2. Work flow diagram illustrating the steps in the fuels treatment planning decision support process.

Table 1-2. Inventory and description of the most commonly used data sets and software applications identified during Phase II of the STS.

Names and Descriptions of the Data and Software Tools Commonly Used for Fuels Treatment Planning		
Data	Aspect	Direction to which a mountain slope faces
	Digital Elevation Model (DEM)	Digital representation of ground surface topography or terrain.
	FEAT/FIREMON Integrated (FFI)	Designed as a comprehensive, database-driven framework that organizes sampling protocols, stores field data, and provides analysis tools for vegetation sampling and monitoring programs.
	Field Sample Vegetation Database (FSVeg)	FSVeg contains plot vegetation data from field surveys such as FIA data, stand exams, inventories, and regeneration surveys.
	GNN-FVS	Vegetation data for the Pacific Northwest
	LANDFIRE (grid)	LANDFIRE data products are designed to facilitate national- and regional-level strategic planning and reporting of wildland fire management activities.
	Slope	Slope is used to describe the steepness, incline, gradient, or grade of topography or terrain.
	Weather	Weather Information Management System (WIMS) weather data.
Software Models and Tools	Area Change Tool (ACT)	Tool used to help design and delineate fuel and vegetation prescriptions as well as edit raster layers to accurately portray treatment outcomes. The ACT is part of the IFP-LANDFIRE tool kit.
	BEHAVE-Plus	Fire modeling system comprised of a collection of models that describe fire behavior, fire effects, and the fire environment and produces tables, graphs, and simple diagrams.
	CONSUME	Decision-making tool, designed to assist resource managers in planning for prescribed fire, wildland fire for use, and wildfire by predicting fuel consumption, pollutant emissions, and heat release.
	Ecosystem Management Decision Support System (EMDS)	Integrates GIS, logic, and decision modeling to provide decision support for a substantial portion of the adaptive management process of ecosystem management.
	FARSITE	Fire growth simulation model that uses spatial information on topography and fuels with weather and wind files
	Fire and Fuels Extension for FVS (FVS-FFE)	Integrates FVS with elements from existing models of fire behavior and fire effects. It predicts the effects of stand development and management actions on fuel dynamics, fire behavior, and fire effects
	Fire Behavior Assessment Tool (FBAT)	Provides an interface between ArcMap and FlamMap3, a fire behavior mapping and analysis program that computes potential fire behavior characteristics at a pixel level. The FBAT is part of the IFP-LANDFIRE tool kit.
	Fire Regime Classification Class (FRCC)	Discrete metric that quantifies the amount that current vegetation has departed from the simulated historical vegetation reference conditions
	First Order Fire Effects model (FOFEM)	Program developed to meet the needs of resource managers, planners, and analysts in predicting and planning for fire effects.
	FlamMap	Basic fire behavior calculations intended for characterizing fuel hazard in fire management planning.
	Forest Vegetation Simulator (FVS)	Individual-tree, distance-independent growth and yield model
	Fuel Characterization and Classification System (FCCS)	Program that builds, characterizes, and classifies wildland and managed fuels throughout North America.
	FuelCalc	Tool to compute surface and canopy fuel loads and characteristics from inventory data, to support fuels treatment decisions, and to provide linkages to stand visualization, fire behavior and fire effects programs.
	Most Similar Neighbor/Nearest Neighbor (MSN/NN)	Program is used to impute attributes measured on some sample units to sample units where they are not measured.
	Multi-scale Resource Integration Tool (MRIT)	Characterizes the composition of feature layers (i.e., spatial themes added to an ArcMap document) within user-defined reporting units (e.g., landscapes), then classifies those reporting units according to their relative status. The MRIT is part of the IFP-LANDFIRE tool kit.
	NEXUS	Crown fire hazard analysis software that links separate models of surface and crown fire behavior to compute indices of relative crown fire potential.
	PHYGROW	Hydrologic based plant growth simulation model. PHYGROW is still under development and testing.
	Smoke Impact Spreadsheet (SIS)	Simple-to-use planning model for calculating particulate matter (PM) emissions and concentrations downwind of wildland fires.
	Stand Visualization System (SVS)	Generates graphic images depicting stand conditions represented by a list of individual stand components, e.g., trees, shrubs, and down material.
	Vegetation Dynamics Development Tool (VDDT)	Tool that provides a modeling framework for examining the role of various transition agents and management actions in vegetation change.

Table 1-3. Summary of the most commonly used vegetation data sets, software applications, and/or tools used in the fuels treatment planning process and where they are applied in the fuels treatment analysis process.

	Define Vegetation Data, Landscape and Scale	Current Year Data Preparation and Quality Control	Simulate and Analyze Fire Behavior	Analyze Fire Effects and Risk	Design Treatment Strategy	Simulate Vegetation, Geophysical, and Fuel Conditions
Vegetation data	FSVeg (treelist) FFI (treelist) GNN-FVS (treelist) LANDFIRE (grid)					
Software models and tools		MSN/NN FVS-FFE SVS ACT	FVS-FFE FlamMap FARSITE NEXUS BEHAVE-Plus FBAT SIS	MRIT SFRAS RAVAR FlamMap FCCS/ CONSUME FOFEM Vsmoke	EMDS MTT/TOM	FVS-FFE PHYGROW VDDT FRCC FCCS- CONSUME ACT FuelCalc
Geophysical data	DEM Slope Aspect Weather (WIMS)					

In addition to developing a work flow diagram (Figure 1-2) and creating an inventory of the various data sources and software applications available for fuels treatment planning, a more detailed dissection of the sub-processes that occur in each of the six process steps indicated in Figure 1-2 was performed to understand how an analysis is performed from a fuels treatment specialist perspective. Narratives that describe and outline the six process steps and associated sub-processes are included as Attachment A of this document. Also included in Attachment A are more detailed work flow diagrams for each of the six steps in the decision support process. The narratives in Attachment A were used to inform the IFT-DSS architecture design.

1.6 IFT-DSS USERS

The following four user groups that form the key stakeholder groups (discussed in Section 1.2 above) have been identified for the IFT-DSS:

1. Approximately 1,000 **fuels treatment specialists** located throughout the United States at multiple federal and state agencies. Fuels treatment specialists will be the primary users of the system to perform year-round fuels treatment planning activities.
2. Several (20) **scientific collaborators** who will provide new or updated science, models, and tools to the system. Scientific collaborators will be periodic contributors to and users of the IFT-DSS.

3. **Data providers** who will provide data to the system:
   - A few (2 to 5) institutional data providers who will make large-scale databases available to the system (i.e., LANDFIRE and FSVeg)
   - Fuels treatment specialists who will upload or create local data sets
4. **System administrators** who will operate and maintain the system over time.

2. ARCHITECTURAL GOALS, CHALLENGES, AND REQUIREMENTS

2.1 ARCHITECTURAL GOALS

The overarching goal of the IFT-DSS architecture is to improve the practice of fuels treatment planning by integrating the variety of data, software models, and tools into a framework with an easy to use graphic user interface (GUI). More specifically, the architecture is driven by the following objectives:

- Simplification of the fuels treatment planning decision support process and improvement in the overall quality of analysis and planning by more easily combining and reusing applications and providing new opportunities for data analysis and collaboration.
- Scientific collaboration by providing a registration mechanism and tools that allow the integration of new software applications into the framework.
- Simplification of project documentation and audit-trail tracking to support regulatory requirements.

2.2 CHALLENGES

The JFSP, along with its partners and clients, form a community of professionals including fuels treatment specialists, researchers, data providers, and others, who currently use a variety of data sources and software applications. These applications (listed in Table 1-2) are generally developed by different individuals and organizations, and do not share common data structures or (in many cases) terminologies. Modification and maintenance of these applications is beyond the scope of this project, that is, they will necessarily continue to evolve and be maintained by the community. New applications will continue to be developed and will need to be integrated with the existing applications over time.

The success of the IFT-DSS project depends on the participation, cooperation, and assistance of the developers of these applications. In most cases, some modification(s) to existing applications will be required to properly integrate with the IFT-DSS framework. A key challenge of this project is to encourage the participation of developers of these applications and to begin to create common data and software integration standards. This architecture document suggests tools and strategies to achieve this cooperation in Section 2.3.2, Community Requirements.

From a technical point of view, applications run on different platforms, are written in different software languages, and require different data types and formats. A key technical challenge is to design a framework that supports the diversity of platforms and languages and allows the applications to be driven from a common GUI. These issues will be discussed in Section 2.4, Technical Requirements.

2.3 REQUIREMENTS

Requirements define the intended purpose of a system under development. System requirements describe what the system will do and how it will be expected to perform. There are different types of requirements:

- **Strategic-level requirements** describe the key high-level requirements of the system and architecture.
- **Community requirements** describe tools, services, and standards that must be provided to the scientific community to achieve the integration goals.
- **Technical requirements** describe the high-level technical or platform issues.
- **General requirements** describe attributes of the system that generally apply to any complex software system.
- **Functional requirements** describe what the system must do and the functional attributes and characteristics that describe the functions that the architecture must support.
- **IT requirements** describe the interagency IT requirements.

The remainder of this section details the requirements for the IFT-DSS.

2.3.1 Strategic-level Requirements

The IFT-DSS architecture will achieve its goals in several ways:

- Development of a unifying software framework to integrate applications.
- Centralization, organization, and management of fuels treatment data, software models, and analysis tools.
- Development of a registry system for new applications or updates to be distributed to the scientific community.
- Development of one or more scientific collaboration tools that can be used by application developers to assist in modifying the various software applications to function within the new framework. These tools would make it easier for software developers to make their applications available to the IFT-DSS and would support conformance verification, reusable software components, and other features to be determined.
- Specification of data standards, which will be supported by all integrated applications.
- Assistance and training for the scientific community, as necessary, to achieve the integration of the various applications.
- Training of fuels treatment specialists, as appropriate, via software user guides, help pages, and training programs.

2.3.2 Community Requirements

The variety of data and software applications used in the fire and fuels community differ in terms of manner of invocation, robustness, generality, types of modeling, execution platform, and in other ways. Integrating these applications poses both technical and structural challenges. This section discusses the structural challenges, or more precisely, the challenges of bringing together a diverse community of application developers and users to achieve a common goal.

Developers of individual applications are likely to have limited time, if any, to integrate their products with the proposed framework. To provide encouragement and incentive, the IFT-DSS development team should provide:

- Software tools to simplify and streamline the process of integrating new models into the IFT-DSS.
- Technical assistance including software application programming interface (API) documentation and email or phone support to application developers.
- Easy registration of components, and simplified delivery of applications and updates to users.
- Clear specifications for data standards, and specifications of required APIs that the software applications are expected to support.

In some cases, there may be key applications, in which the original developer is not available or is unwilling or unable to make modifications that are necessary for integration with the IFT-DSS. In these cases, assuming the source code is available in the public domain and can be decoded, the IFT-DSS development team may choose to integrate the application by

- obtaining source code, making necessary modifications, and integrating the application into the IFT-DSS;
- "wrapping" the original software application in a wrapper application, that is itself integrated into the framework; and
- re-engineering the application in a way to allow it to integrate into the framework.

It should be anticipated that in the early stages of IFT-DSS development, several of the existing software applications will require integration into the IFT-DSS by one of the methods described above. Over time, when a sufficient number of key applications support the new framework, it is likely that there will be significant user pressure on remaining applications (and new applications being developed) such that voluntary support for the framework will be common.

The IFT-DSS development team can expect the scientists and application developers (and users) of the various applications to appreciate and support the value of integration from the IFT-DSS framework. There is an obvious benefit to the community in general, and the expected value justifies the efforts that will be required. This concept is beginning to come to fruition effecting a similar type of integration for the smoke modeling community through effort on the BlueSky Framework, <http://www.getbluesky.org/>, and has proven effective in the air quality monitoring community through the U.S. Environmental Protection Agency's (EPA) AIRNow program, <http://www.airnow.gov/>. The IFT-DSS development team should draw on these established collaboration and community building strategies when possible to benefit the IFT-DSS community.

2.4 TECHNICAL REQUIREMENTS

Existing fuels software applications perform a variety of functions and can be combined together to perform complex simulations. The variety of existing applications developed with different goals and requirements, at different times, by different organizations, presents an integration challenge. Applications are written in different software languages, may currently run on different operating systems, have sometimes overlapping functionality, and require differing data formats.

The design of the IFT-DSS software architecture will be driven by the requirement to allow as many applications as possible to work together (including applications not yet developed) with a minimum of additional effort required by those application developers to support the framework. This objective can be achieved by designing an architecture that is adaptable and generic enough to accommodate a broad variety of applications and functionality.

Government and state agencies often have rules and regulations regarding installation of new software on government computers and workstations. These regulations make it difficult to install software applications directly onto agency desktop computers. One of the requirements of the IFT-DSS is that it can be run from a browser by any desktop, laptop, or workstation computer connected to the Internet so that users will not have to install any software on their local desktop computer. This feature implies that the system will have to be hosted on an accessible server. See additional requirements in the Section 2.6, Information Technology Requirements.

The GUI for the application framework will necessarily be designed to run in a standard web browser. However, an additional requirement is that local computers or workstations may be configured to run the system locally in the event there is no Internet connectivity. While a web-based application can be packaged and installed locally, this requirement affects the selection of the hosting operating system. For example, if the web-based application framework is designed to be hosted on a Linux server, any native application components will not be able to run on a Windows system due to cross-platform incompatibility. To enable both on-line and off-line system access, either the target hosting server should be Windows (allowing simple redeployment on any Windows desktop computer or workstation) or multiple versions of all native applications will need to be developed and maintained. Developing a system that can run on both operating systems will likely be more expensive, and maintenance and testing costs are likewise impacted. Note that applications written in languages such as Java will not impact cost as Java applications are easily re-deployable across different platforms.

2.5 GENERAL SOFTWARE REQUIREMENTS

The IFT-DSS architecture should have a long useful lifespan and will support the following software attributes during the architecture, design, and coding of the system:

- **Modularity** – each service (i.e., software applications, data sets, and tools) is developed in a way such that the component may be used independently of other software components. This can benefit the user, when such modules can be used independently of other program functions or combined with other modules. It also benefits the system developers and maintainers, who can often reuse such modules in multiple applications.

- **Extensibility –** the system can be expanded over time to support the incorporation of new tools and data as they become available. This is a key requirement; as new applications are developed or old ones modified, they can be easily added to the framework.
- **Flexibility –** the system is flexible to allow users to customize data and model execution to fit their specific project analysis. This also has great benefits to the maintainers of a system. As platforms and requirements change, the software can be adapted.
- **Portability –** the system is easy to access and use from any standard desktop computer and does not require proprietary software or systems.
- **Ease of use –** the system is straightforward and practical to use through a well-designed interface. Specialized training or software skills are not required to run the system.
- **Maintainability –** the system should include clear structure, and good quality technical documentation to allow system maintainers to easily maintain the system as features are added, platform requirements change, or defects are addressed.

2.6 FUNCTIONAL REQUIREMENTS

The requirements listed here apply to the GUI application and to user experience or to the underlying framework. The system must

- support the decision support process, analysis steps, and software tools commonly used for fuels treatment planning;
- support visualization of spatial and tabular data, data editing, and user interaction at each processing step;
- provide data choices (i.e., standard treelist data, standard gridded data, and/or locally generated data) ;
- have a data processing and transformation mechanism to acquire or create and transform input data (i.e., the ability to combine vector and raster data formats; support for vector-to-raster transformations and vice-versa). The system should contain mechanisms to streamline data preparation and processing;
- have a quality control, documentation, and audit-trail mechanism to support regulatory requirements;
- provide guidance (i.e., sub-model choices) based on geographic scale and the type of analysis being performed;
- be able to be stopped or started at any processing point;
- support analytical collaboration; that is, the system should provide a mechanism for fuels treatment analysts to publish and share methods and algorithms with other system users via a central system library;
- have a mechanism to perform sensitivity analyses;

- recognize user error and explain alternate action; and
- support scientific collaboration; that is, the system must be able to incorporate new models and tools as they become available through an authorship and publishing mechanism.

2.7 INFORMATION TECHNOLOGY REQUIREMENTS

The following requirements apply to the general IT operational requirements. The system

- must have an operation and maintenance plan, long-term hosting agency with allocated servers, equipment, and maintenance staff;
- should have a dedicated hosting agency and software and hardware resources available;
- should be fully operational (24/7) and reliable;
- should be designed to function with high-speed Internet access (assumes users will have such access at a minimum);
- must support ArcGIS data formats and other commonly-used geographic information system (GIS) data formats; and
- must be designed for inter-operability with other decision-support systems in the fire and fuels domain (i.e., BlueSky Framework and WFDSS).

2.8 PERFORMANCE AND SCALABILITY REQUIREMENTS

The following requirements apply to the overall performance and scalability of the IFT-DSS. The system

- should be able to accommodate up to a maximum of 250 simultaneous users (about 25% of the total user community), running up to 500 computations per hour. These numbers can be increased with additional back-end hardware (i.e. a second server could be added to double those numbers);
- will ensure a response time of three seconds or less whenever a user invokes a command in the GUI. In cases of heavy load, or if the user is in the process of running an operation, the system response should be an indication that the command has been recognized by the system and that the server is busy. In the latter case, some indication of progress such as an active status bar should be provided;
- should ensure that if a user repeatedly invokes an operation (because of issues with slow response time, for example) the repeated commands will not interfere with system stability (redundant commands will be ignored or deactivated to prevent this).
- will store all intermediate calculations as they are produced by simulations, and any data the user enters will be stored as soon as it is received. In the case of a server-crash or if the system needs rebooting for any reason, all stored data will be available when the user

logs back in. In addition, routine server back-up processes will be employed to ensure that data are not lost in the case of a server-failure;

- and will shield the user from server crashes during system use by invoking automatic recovery processes and prompting the user to log back into the system without data loss.

2.9 INFORMATION TECHNOLOGY INVESTMENT REQUIREMENTS

The National Wildfire Coordinating Group (NWCG) is in the process of developing a "cohesive interdepartmental-interagency method to manage the complex wildland fire Information Technology (IT) investment portfolio." The IT Investment Process stewards IT projects from planning through development, operations, and finally to retirement. The details of the process are under development, but the overall structure and flow have been documented, with participants, interfaces, inputs, outputs, process, and thresholds outlined for each step.

As part of Phase II of the STS, the IFT-DSS team met and instituted a dialog with the creators of the IT Investment Process. It is not expected that the IT Investment Process will be fully realized by the time the IFT-DSS development is expected to begin. Also, the IT Investment Process includes the planning and proposal phases, which the IFT-DSS has already begun. However, it is expected that the IFT-DSS development effort will serve as a pilot for the IT Investment Process, which will benefit both the IFT-DSS project and the IT Investment Process development. For example, issues in the IFT-DSS computer security plan might be identified, while gaps in the IT Investment Process will be discovered and rectified.

The IFT-DSS development team should continue to engage and work with the NWCG and the IT Investment Process coordinators to ensure that the IFT-DSS adheres to the Investment Process and the interagency IT requirements.

3. THE IFT-DSS USER EXPERIENCE

An important aspect of the IFT-DSS architecture is system behavior and user experience. This section describes how the system will behave and how the user will interact with the IFT-DSS without prematurely specifying GUI design. This discussion is intended to inform the development of the system design specification (which is the next step moving forward) and to provide substance for discussions with future system users.

The IFT-DSS is a strategic planning tool that will be operational, reliable, and accessible over the Internet from a standard desktop computer. The IFT-DSS website will contain a user login screen as the main point of entry into the system. The first time that users log into the IFT-DSS, they will be prompted to create a user profile that will be saved to the system. From a user perspective, a project analysis will involve three phases: (1) project set up and planning, (2) software model execution and iterative analysis, and (3) project finalization, documentation, and archive. The remainder of this section describes the general user experience during the three phases of a project analysis using the IFT-DSS.

3.1 PHASE I – PROJECT SET UP AND PLANNING

The project set up and planning phase of a fuels treatment analysis will mainly be performed by the user and will involve the creation of a project profile, acquisition or generation of the project input data, and the selection of the software models to be used for the analysis. Depending on the analysis being performed, the user may also enter information about the desired target conditions or result values. For each project analysis, a vegetation scenario is analyzed and adjustable parameters (e.g., the type of fuel treatment(s), the timing and spatial extent of treatments, etc.) are modified until the resulting outputs meet the desired target conditions.

The first phase of a project analysis will involve making available selections and filling in several online forms that will be administered by the system through the user interface. The forms will be dynamic so that as specific selections are made, subsequent forms only request information that is relevant to a particular type of analysis. The type of information that will be provided will include: a project name and description, the project analysis objective(s), the spatial location and extent of the area of interest, the vegetation data to be used, the weather data to be used, and the software models and tools that will be used for the analysis. User-provided information regarding vegetation data inputs will enable the IFT-DSS to acquire data from external data sources such as FSVeg or LANDFIRE while other types of input data will be provided directly by the user depending on the choices of software models and tools that are selected for the analysis.

Once all of the project analysis input parameters and associated data are in place, the user will access a sophisticated data visualization and editing tool. This editing tool will be the main mechanism through which an analyst interacts with the input, intermediate, and output data sets. In the early phase of a project analysis, the user would use the editing tool to confirm that all data (e.g., spatial and/or aspatial) and map layers are acceptable for use as modeling inputs. The editing tool will be powerful and will allow the user to edit any of the spatial (map layers) and/or aspatial (tabular) data. In addition to providing spatial visualization tools (i.e., map viewers) it will provide graphical displays of aspatial data such as time-series graphs and/or tabular data summaries.

Another aspect of project planning will involve selecting the software models to be used for the analysis. The IFT-DSS will provide choices of model(s) available for analysis. The selection of model choices will provide the flow control or pathway through the IFT-DSS. A design goal of the IFT-DSS is to emulate flow control, data, and software models that are found in existing systems including (but not limited to) ArcFuels, INFORMS, and the IFP-LANDFIRE process and make these available to the IFT-DSS user. The user will also have the choice of creating customized flow control(s) through the system.

3.2 PHASE II – SOFTWARE MODEL EXECUTION AND ITERATIVE ANALYSIS

Upon completion of the project set up and planning phase, the user will direct the IFT-DSS to initiate the processing and analysis phase. Scheduling and prioritization of work will be done interactively by the user and the modeling run(s) will then be submitted to the system. Users will be given project tracking numbers with which they can monitor the IFT-DSS processing progress. The heavy computer processing will occur on remote server(s) and will be returned to the user via the user interface browser. When results are available, the user will review, adjust, and reiterate the analysis as desired. Multiple iterations may be required to compare different scenarios or possibilities.

When a run is complete (and while it is still in progress at certain points), the user will be able to view results using the data visualization and editing tool (the same tool used in the set-up phase described above). Controls will be provided to interrupt processing, modify input parameters, and restart processing. It will also be possible access the project planning tool to refine the model scenario and project analysis parameters. This iterative process may be done continuously or it may be broken into brief sessions that occur at various times over a period spanning hours or days depending on the complexity and scale of the project.

3.3 PHASE III – PROJECT FINALIZATION, DOCUMENTATION, AND ARCHIVE

When all iterations of the analysis have been completed and the user has validated the results, the project will move into the last phase of finalization, documentation, and archival. During this final phase, electronic documentation in the form of text, maps, data tables, and/or graphs will be generated by the IFT-DSS. The system will also generate documentation outlining the analysis steps performed, the interim and final data products, and any human mediated changes that were made to the project analysis. When a project analysis is complete, the project data will be archived in two ways: (1) project information, data, and results will be permanently archived in a database within the system, and (2) an electronic project “bundle” will be created and downloaded to the user’s local computer that consists of all of the elements required to restore the project analysis to the system for future exploration or re-analysis. The purpose of this archive is to provide supporting data in the instance that the decision process is reviewed at a later date.

4. SOFTWARE ARCHITECTURE

This section discusses the overall software architecture for the IFT-DSS and presents different views of the system. The software architecture, or logical view, describes the system components and layers and how they interact. The proposed hardware configuration, or structural view, describes how the system might physically be constructed.

4.1 ARCHITECTURAL OVERVIEW

The requirements for the IFT-DSS system include integration of disparate applications and systems, some of which are web-based. In addition, to support the reuse of IFT-DSS applications and services over the Internet, a flexible, modern, web-friendly system architecture is required. The overall architectural design approach for the IFT-DSS is based on the concept of SOA. SOA facilitates the integration of disparate software systems and applications by separating business functions into distinct units, or services, that can be made accessible so that users can easily access, combine, and reuse individual services as needed (Erl, 2005).

A key benefit to the SOA approach is that it also enables the same software models and processes to be combined and made available through multiple GUIs so that different communities can make use of the system. SOA facilitates the integration of data, new software systems, and legacy software systems to streamline work processes. It is a widely used architectural approach for addressing the same problem that the fire and fuels treatment community faces—the decentralization of disparate software applications written in various programming languages and of different vintages. A key recommendation resulting from Phase I of the STS was that the fire and fuels treatment community would greatly benefit from an SOA solution (Palmquist, 2008).

The architecture presented here will offer several benefits to users and developers of existing fire and fuels software applications. In addition to providing coordinated hosting of the diverse models used to assess fuels treatment options, this system will deliver many advantages over independent execution of the models as currently required.

- The processes and services (i.e. the applications themselves) will be separated from data loading and unloading activities.
- A small suite of data interchange standards will be adopted.
- All processes and services will be adapted to receive and produce data in a standard format.
- Models with aspatial processes, many of which are currently limited to a specific type of input format, will be empowered to work with any combination of aspatial or spatial data formats.
- Resolution, coordinate system, and units incompatibilities will be resolved automatically by a central control system.
- The system will recognize and exploit redundancies in the input data streams and in the modeled conditions. This feature will allow optimal performance and storage efficiency.

Some of these key features result from architectural details that are distributed throughout the system. Others are focused in one or another component of the system. Many of the key features are embodied in the dispatcher which is described in Section 6.4, the structural architecture section. The dispatcher, in concert with other framework components, will handle all or most of the overhead associated with formatting and passing data from one application to the next which will free users from the burden of managing data formats and model processing steps in great detail.

4.2 THE IFT-DSS SOFTWARE ARCHITECTURE DESIGN

The IFT-DSS software architecture design provides the framework for implementing the decision support process and software tools used by fuels treatment planners. The core of the IFT-DSS software architecture consists of five components: (1) a graphical user interface (GUI), (2) a control database, (3) a dispatcher, (4) data and software services, and (5) a geospatial database. The following describes the general function(s) of the five key architecture components:

- **Graphical User Interface (GUI).** The GUI is the user access point into the IFT-DSS. It controls the user experience and provides the portal for all system inputs and outputs.
- **Control Database.** The control database stores all information that is input to the IFT-DSS and manages the user experience and decision support process.
- **Dispatcher.** The dispatcher works with the control database to invoke specific functions and services to support the needs of the high-level decision support process.
- **Data and Software Services.** The fire and fuels treatment domain data, software models, and tools that support the business needs.
- **Geospatial Database.** The geospatial database is the data management hub of the IFT-DSS. It stores all project-related input, intermediate, and output data. It is called the geospatial database because it can manage spatial data (map layers) as well as tabular data.

Figure 4-1 illustrates the five key components of the architecture and their relative architectural arrangement. Each of the five components contains interconnected sub-components, or layers, that perform specific functions within the system. **Figure 4-2** presents an overall view of the architecture itself including the five high-level architectural components, their sub-components (layers), and the connection points and information flow between each layer.

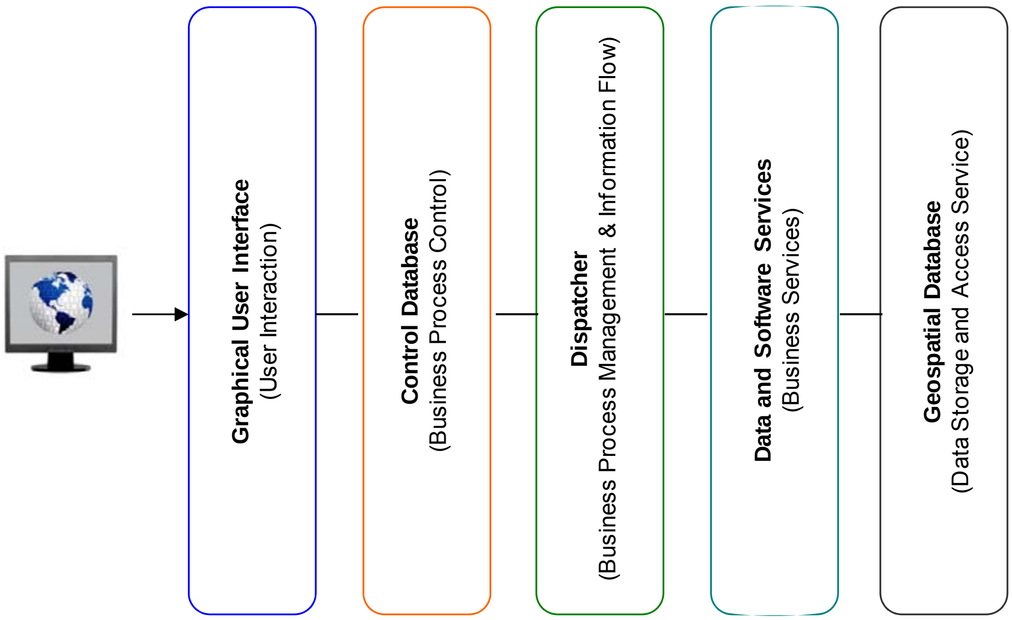

Figure 4-1. Illustration of the top-level IFT-DSS architectural components.

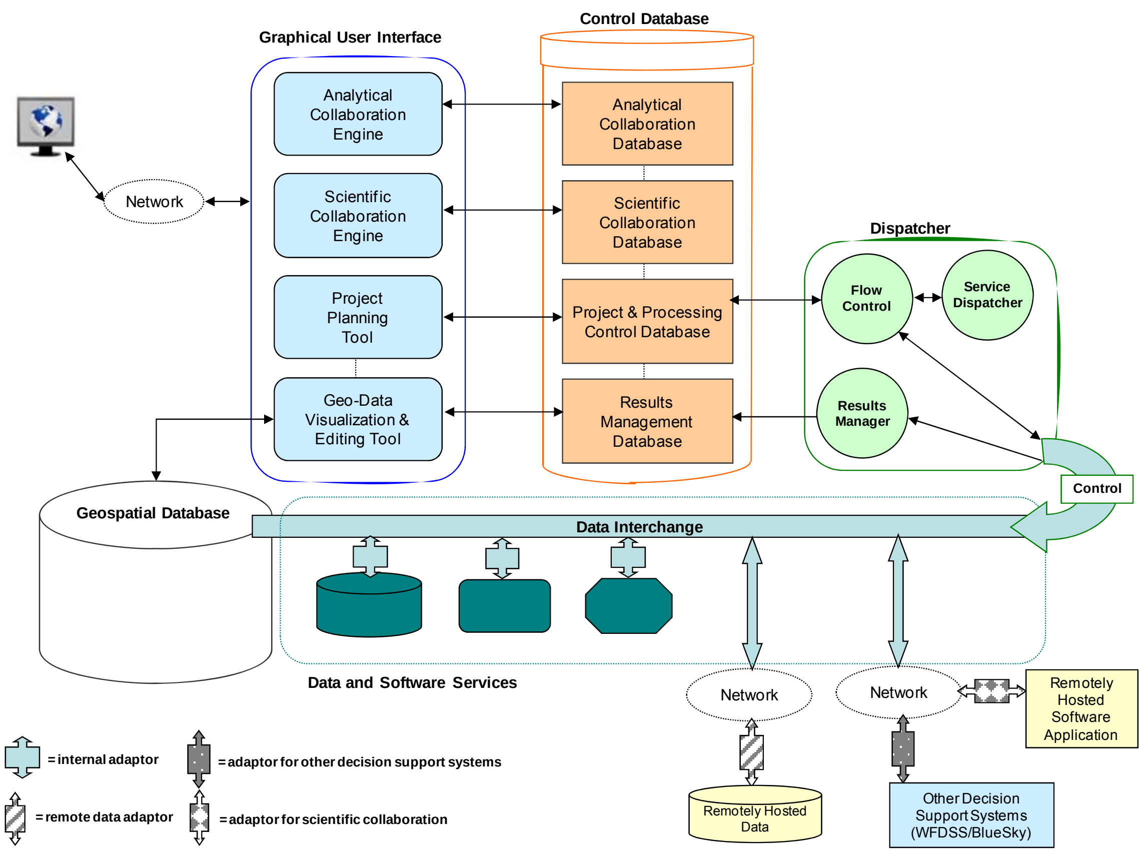

Figure 4-2. Illustration of the IFT-DSS software architecture and sub-components (layers).

Most of the remainder of this section describes the arrangement and general functions of the five key architectural components and layers. The following discussion in the remainder of Sections 4-2 through 4-5 describe the system architecture from a development perspective. Section 4.6 presents a narrative example of the IFT-DSS functionality from a user's perspective, explaining how data and control flows through the system.

4.2.1 Graphical User Interface

The user experience and the manner in which a user interacts with the system are vitally important. The GUI provides the mechanism by which the user interacts with the system and largely dictates the user experience. The GUI also serves the underlying system components and decision support processes and will be customized to facilitate the fuels treatment decision support process. Note that the SOA design of the IFT-DSS allows the creation of multiple GUIs to serve specific purposes or user communities. For example, a Management GUI could be created that would provide a streamlined interface showing active projects, status, and results. A light-weight GUI could also be developed for use on mobile devices in the field. The GUI described here represents the core interface that will be developed to support fuels treatment specialists and scientific collaborators. The GUI will be web-based and will operate in a standard web-browser. It will be separated into four layers for ease of use and to allow the use of different user interface technologies (i.e., HTML, AJAX, Flex, etc.). The four layers of the GUI will consist of: (1) the analytical collaboration engine, (2) the scientific collaboration engine, (3) the project planning tool, (4) and the geo-data visualization and editing tool. These layers will provide the following general functionalities:

- Analytical collaboration engine will provide the main access point into the IFT-DSS and the analytical collaboration environment users of the IFT-DSS. It will contain login screens, user profiles, question and answer forums, a project inventory and authorship library to facilitate analytical collaboration.
- The scientific collaboration engine will provide researchers and scientists a mechanism for registering new software models and tools (services) with the system so that they can be made available to users of the IFT-DSS. It will contain an authorship library containing information about the model developers, how the models should be applied, and any other relevant information pertaining to the data and software available through the IFT-DSS.
- The project set up and planning tool will serve as the project analysis and flow control mechanism to the IFT-DSS. This function allows users to create new project analyses, set up the project parameters, and select the software models to be used for the analysis.
- The geo-data visualization and editing tool will allow users to interact with both spatial and aspatial input, intermediate, and output data throughout a project analysis.

When a user interacts with the GUI, all information input by the user is stored and managed by the control database. As shown in Figure 4-2, each layer of the user interface has a corresponding control database layer.

4.2.1.1 Analytical Collaboration Engine

The analytical collaboration engine will be the main access point to the IFT-DSS. From a website perspective, it serves as the home page. It will consist of a main IFT-DSS webpage and several screens, including a user login screen, a user profile set up screen, a list of current users and affiliations (if they choose to be listed publicly), an interactive chat page where analysts can post questions and information, a library of past project analyses performed and a description of each (if a user wishes to share an analysis), and links to other relevant information. When a user establishes a profile, he or she will indicate the type of collaboration that they wish to engage in (e.g., open, limited, or closed). This choice will control where and to what degree a user's personal information and project analyses are published and accessible by others within the system.

4.2.1.2 Scientific Collaboration Engine

The scientific collaboration engine will be available through the IFT-DSS home page and will provide the access point for scientists and researchers to register new or updated software models, data services, or other tools to the system (collectively referred to as services). The collaboration engine will provide input screens that scientific collaborators can fill in to register their service with the system and make it available to system users. For example, it is possible that software models could be registered with the system and incorporated into the IFT-DSS. Alternatively, a scientific collaborator may wish to host a software model on a website and have the IFT-DSS access it remotely through the use of web services.

The scientific collaboration engine will provide information and guidance about collaboration standards that define how a service can be made available to the system. It will also provide the GUI mechanism for collaborators to register and activate their services within the IFT-DSS.

4.2.1.3 Project Planning Tool

The project planning tool will be the GUI layer where the user will set up a new project analysis. All information related to a specific project will be input through the project planning tool including: the project name; the analysis objectives and background information; the geographic location and extent of the study area; the choice of vegetation data to be used (e.g., LANDFIRE, treelist data, local data). After a project has been defined, the project planning tool will walk the user through a series of screens to make selections allow users to select the software models to be used for the analysis.

The project planning tool will have mechanisms that inventory and track parameters associated with a particular project such as time intervals associated with treatments or disturbances and values at risk. It will contain a catalog of approaches to solve particular types of problems and this catalog will evolve as users contribute to the analytical collaboration project library.

The project planning tool and the geo-data visualization and editing tool (Section 6.2.1.4) will provide the mechanisms by which an analyst will view intermediate output, edit data values, modify analysis parameters, and initiate model iterations. These two components of the GUI will be tightly coupled so that they work efficiently together to facilitate the iterative processes within the IFT-DSS.

4.2.1.4 Geo-Data Visualization and Editing Tool

The geo-data visualization and editing tool will be a powerful GIS-based application that allows analysts to view, edit, perform spatial manipulations, stack map layers, and perform calculations on both spatial and aspatial data. It will contain a GIS-based viewer as well as a tabular data viewer. It will contain controls to quickly and easily perform unit conversions on specific data sets, create .LCP files, and convert vector maps to raster maps and vice versa.

During an analysis, the geo-data tool will allow users to perform the functions that are required during an iterative treatment scenario analysis including modifying vegetation conditions, designing and applying alternate treatment scenarios, and reviewing model output data. It will also contain controls to handle many of the manual overhead steps associated with working with vegetation data such as data conversions, calibration, imputation, and vegetation simulation.

Behind the scenes, GIS-specific functions such as map projections will be handled automatically. It will also contain controls to export files in specified formats and view spreadsheets of data values (and map attribute data) and will provide controls to export data in standardized formats that are compatible with other commonly used GIS systems such as ArcMap and Google Earth.

4.3 CONTROL DATABASE

The control database will serve as the knowledge base of the IFT-DSS. As shown in Figure 4-2, almost all the information that is input by the user through the GUI is stored and managed by the control database. The control database manages the user experience through the analytical collaboration database and the project flow by dynamically recognizing user options and providing the appropriate responses and choices throughout a project analysis session. It manages and administers the data sources, software models, and tools available in the IFT-DSS through the scientific collaboration database.

The database will comprise a relational database (e.g., MS SQL Server, Oracle, MySQL) and an associated file repository for document data and storage.

4.3.1 Analytical Collaboration Database

The analytical collaboration database is part of the control database. It will receive information from the analytical collaboration engine (GUI) and store and manage all of the data associated with the users of the system. It will also provide an archive of the interactive chat page(s) and will store and manage the catalog of past project analyses and all associated information. It will serve as a library of project analyses and analytical collaboration activities.

4.3.2 Scientific Collaboration Database

The scientific collaboration database is part of the control database. The scientific collaboration database will receive, store, and manage information from the scientific collaboration engine (GUI). The scientific collaboration database will contain data and information associated with the software models and services available to the IFT-DSS users. The information contained in the scientific collaboration database will control the registration and management of services within the system. The scientific collaboration database will contain an inventory of models and information such as model name, model author, data requirements, required inputs and outputs, scientific documentation, and appropriate use scenarios.

To effectively manage and control the IFT-DSS components, the scientific collaboration database will contain an inventory of all IFT-DSS resources including the server components of the system that perform processing and information flow; the data and software models available within (and external to) the system; data converters and transformers; an index of available data and metadata; and indexes into the geo-database.

4.3.3 The Project and Processing Control Database

The project and processing control database will be part of the control database and will connect to the project planning tool (GUI) to receive, store, and manage all of the information associated with a project analyses. The project and processing control database will pass project-specific information and instructions to the dispatcher (Section 6.4).

4.3.4 Results Inventory Database

The results inventory database is also part of the control database. It acts as the data portal between the GUI and the processing steps. When a processing step or project analysis is complete, the results are temporarily stored in the results inventory database and served to the Geo-Data visualization and editing tool (GUI) where the user can view and edit intermediate or final results. When a project analysis is complete, all data and associated information will be transferred from the results inventory database to a permanent storage database where they will be archived for the life of the IFT-DSS.

4.4 DISPATCHER

The dispatcher will function as the logistics manager of the IFT-DSS. It will receive information and instructions from the control database and will execute and manage data processing and information flow among the selected software models in an efficient and optimized manner. It will serve as the communication link between the control database and all other system components. For a specific project analysis, it will pull the appropriate data from the geo-database; it will transform the data as needed; it will then pass the data to the selected software models; it initiates the model runs; it then returns the results back to the geo-database.

Some processes and models in fuels treatment analysis are processing and data intensive and currently take significant time (many hours) to run. The dispatcher will be seek to both reduce processing time, and deal with long-running processes. The dispatcher will be intelligently designed to eliminate or reduce redundant data processing and modeling by only processing data content that are unique or that have been modified. For example, if an analysis is in its intermediary stages and a user makes a change to a data parameter or analysis choice, the system will only re-process and update data and processes that are associated with the change(s). Additionally, the dispatcher will recognize data redundancies and will execute system processes to minimize processing burden. Some processes may still take hours to run. The dispatcher will handle active processes even without user connections, and will provide “tracking numbers” and notifications to the user via email when a process has completed and results are ready.

4.5 GEO-DATABASE

The geo-database is the dynamic data repository of the IFT-DSS. It is called the geo-database because it will have the ability to store geographic information in the form of map layers and attribute information as well as tabular data. The geo-database will contain and manage all of the input and output data generated during a project analysis. It will load and store data related to projects in progress as defined by the project planning inputs. This geo-database contains data tables and relationships that will manage all model inputs, intermediate data files, and outputs including all associated attributes and data parameter units.

The geo-database will contain semi-static geophysical data such as digital elevation model (DEM), elevation, slope, and aspect that are used by some software models during fuels treatment planning. The geo-database will not contain data that are updated regularly or may be input to the system by the user (i.e., LANDFIRE and WIMS). These data will likely be available to the IFT-DSS through remote data services where they can be accessed by the system and used for analysis but the entire data sets will not be permanently archived in the system. The data that are specific to a particular project analysis (i.e., vegetation and weather data) will reside in the geo-database throughout the life of a project analysis. When a project analysis is complete, project-specific data will be transferred to a permanent data archive within the system.

4.6 PROCESS NARRATIVE

This section is a discussion of an example use case narrative to provide another way of understanding the roles of the individual components of the IFT-DSS architecture. It does not provide the full list of features. The narrative refers to the sub-components shown in Figure 4-2.

> An analyst would like to identify and prioritize treatment areas within a landscape. The analyst navigates to the IFT-DSS main web page from her desktop workstation’s web browser. The analyst logs in and the Analytical Collaboration Engine passes her

credentials to the Analytical Collaboration Database for authentication. She then switches to the Project Planning Tool page and enters information about her current project. This information is stored in the Project Processing and Control Database.

At this point the analyst is ready to begin the steps of the fuels treatment planning process, as outlined in Figure 2-1. She will be alternating between the Geo-Data Visualization and Editing Tool and the Project Planning Tool, which will be tightly coupled in the GUI. In the Geo-Data Tool, she selects her area of interest. The Service Dispatcher queries both the Geospatial Database and any Remotely Hosted Databases for vegetation and ancillary data within her domain. She selects the desired data, and the Data Interchange reformats the data (if necessary) and displays it. From here, the analyst can view parameters and edit the data as desired. Edits are stored in the Results Management Database.

When the analyst is satisfied that the data are of sufficient quality for analysis, she develops her process flow using the Project Planning Tool. The Project Planning Tool connects to the Project and Processing Control Database, which provides the inventory of available models and processes that can be run, what inputs they require, what outputs they produce, and guidance on when they should be applied. The Project Planning Tool uses this inventory to generate a dynamic series of screens that walks the analyst through a process. The analyst selects tools and options for preparing data for the analysis year, simulating fire behavior, and analyzing fire effects.

Once the analyst is satisfied with the project plan, she begins the processing. The Project and Processing Control Database then passes the request to the Dispatcher Flow Control. The Flow Control initiates the desired Software Services. As results are returned, the Results Manager sends them to the Results Management Database for viewing by the analyst. If a process is expected to take a long time (several minutes or more), the analyst is alerted and can then log off. The Dispatcher sends an email to the analyst when the job is complete. When the analyst logs back in, the data are ready for viewing in the Geo-Data Visualization and Editing Tool.

If the analyst wants to continue to adjust parameters or try other scenarios, she can make changes in the Project Planning Tool, and the Dispatcher will only run those processes which have been changed. After the analyst is satisfied, she can complete her project and export reports. Data from the Results Management Database will be moved to long-term storage for future retrieval through Analytical Collaboration Engine.

The above narrative covers most of the components of the IFT-DSS; however, it does not cover the Scientific Collaboration Engine and Database because they are meant to serve the developers of the models and data within the IFT-DSS, not their users. A scientific collaborator would use the Scientific Collaboration Engine to register new models or data for use by others.

4.7 IFT-DSS SERVICES AND STANDARDIZED INTERFACES

An IFT-DSS service (or *process* within the IFT-DSS) is an operation that acts on input data to produce output data, or that provides data to the IFT-DSS. The IFT-DSS architecture provides the framework for connecting services, each of which can be anything from a simple script to a complete scientific model. Within the IFT-DSS, users will be able to choose and connect available services in many different configurations. A standardized interface is a specification for the input or output of data to and from a service. Web services often provide standardized interfaces via SOAP standard WSDL specifications. Java applications typically provide program interfaces via the Java interfaces packaged in a jar file. Standardized interfaces provide a mechanism that allows services to connect and interoperate with one another. Standardized interfaces act as input plugs and output plugs, connecting services based on a client application or framework supplying the appropriate type of data.

Standardized interfaces allow a system to be extensible by providing the infrastructure and specification for services to be added to the system over time. These interfaces are shown in Figure 4-2 as *service adaptors*. Service adaptors can accommodate different types of service access including (1) services that are integrated and stored internal to the IFT-DSS, (2) standardized data services provided by large institutions such as geo-portals that the IFT-DSS can access externally (i.e., LANDFIRE), (3) local user-provided data services, and (4) services that are part of other decision support systems such as WFDSS or non-standardized services remotely hosted by individuals.

Due to a general lack of standardization among software models and systems, data providers and services, and other decision support systems, specific service adaptors are required for the IFT-DSS to communicate with different types of services. The IFT-DSS will handle the integration of different service types in the following ways:

A. Services that are integrated and stored internally within the IFT-DSS. Services that are integrated into the IFT-DSS will be modularized versions of existing software programs (or components of these) that are currently in use. The modularization is performed by extracting the scientific algorithms from a model and making them available within the IFT-DSS as an individual service. Alternately, existing models can be added to the system as they currently exist through a process called wrapping. A wrapper takes data from the standard format and translates it into the specific format required by the model, invokes the model, and then takes the model output and translates it into the standard format. In this way, a model that manipulates data of a non-standard format may be included in a framework and treated as if it handled data in the standard way.

B. Standardized data services. Service adaptors will be developed to allow the IFT-DSS to communicate with institutional data services. Large, institutional data providers commonly follow established and widely accepted data standards that facilitate inter-service communication. A service adapter is a wrapper (see A above) for a web service.

C. User-provided data services. IFT-DSS users will have the option to connect the IFT-DSS to a personal (or agency-specific) database housed on their local computer system (or agency network) by providing the data in a common and standard format such as a Web Coverage Service. The IFT-DSS will contain a service adaptor that allows connection to specific standard data services.

D. Services that are part of other decision support systems or non-standardized services hosted by individuals. It is a goal of the IFT-DSS program to collaborate with the scientific community and the developers of other decision support systems within the fire and fuels domain to establish inter-system communication standards. A special form of model adaptor will be developed and used to incorporate models that are part of other decision support systems (e.g.: BlueSky or WFDSS) or are stand-alone modules provided by collaborators. Models that are hosted outside of the IFT-DSS will be accessed through remote model adaptors which may be served by the IFT-DSS or which may be served by other systems. Remote model adaptors are more sophisticated because they require command and control functions as well as input and output data handling mechanisms.

A goal of the IFT-DSS project is to facilitate the process of adding new models to the system through the use of standardized interfaces and specifications that should be established through working with scientific collaborators during the IFT-DSS design specification and implementation process. By publishing and supporting standardized interfaces, the IFT-DSS project will allow existing applications and services to be modified to easily integrate with the new system. In addition, new applications and services can be designed to support the standardized interfaces. These standardized interfaces would likely include the following specifications:

- Java Interface specifications. Modules written in Java could easily be integrated by supporting these interfaces.
- Command-line application standard. Native applications that can run from a command line could support some standard command-line arguments and input and output formats.
- Web services. Standard web service method names and input and output types would be specified.

In addition, specifications for all standard data types, in an XML syntax, will be supported throughout the system. Applications and services that support these standard data types and standard interfaces will be simple to integrate into the framework. The standard interfaces and data types used within the IFT-DSS will likely not be developed specifically for the IFT-DSS. Rather they will draw upon previously developed standards (see Section 7, Collaboration and Standards).

4.8 THE IFT-DSS ARCHITECTURE AND THE FUELS TREATMENT DECISION SUPPORT PROCESS

This section illustrates a more detailed view of the fuels treatment decision support process and how the IFT-DSS architecture supports the process. **Figures 4-3 through 4-8** show a more detailed breakdown of the six steps that comprise the fuels treatment decision support process (as shown in Figure 1-2) and describes how the IFT-DSS and architecture components will support each of the steps and sub-steps from the user perspective.

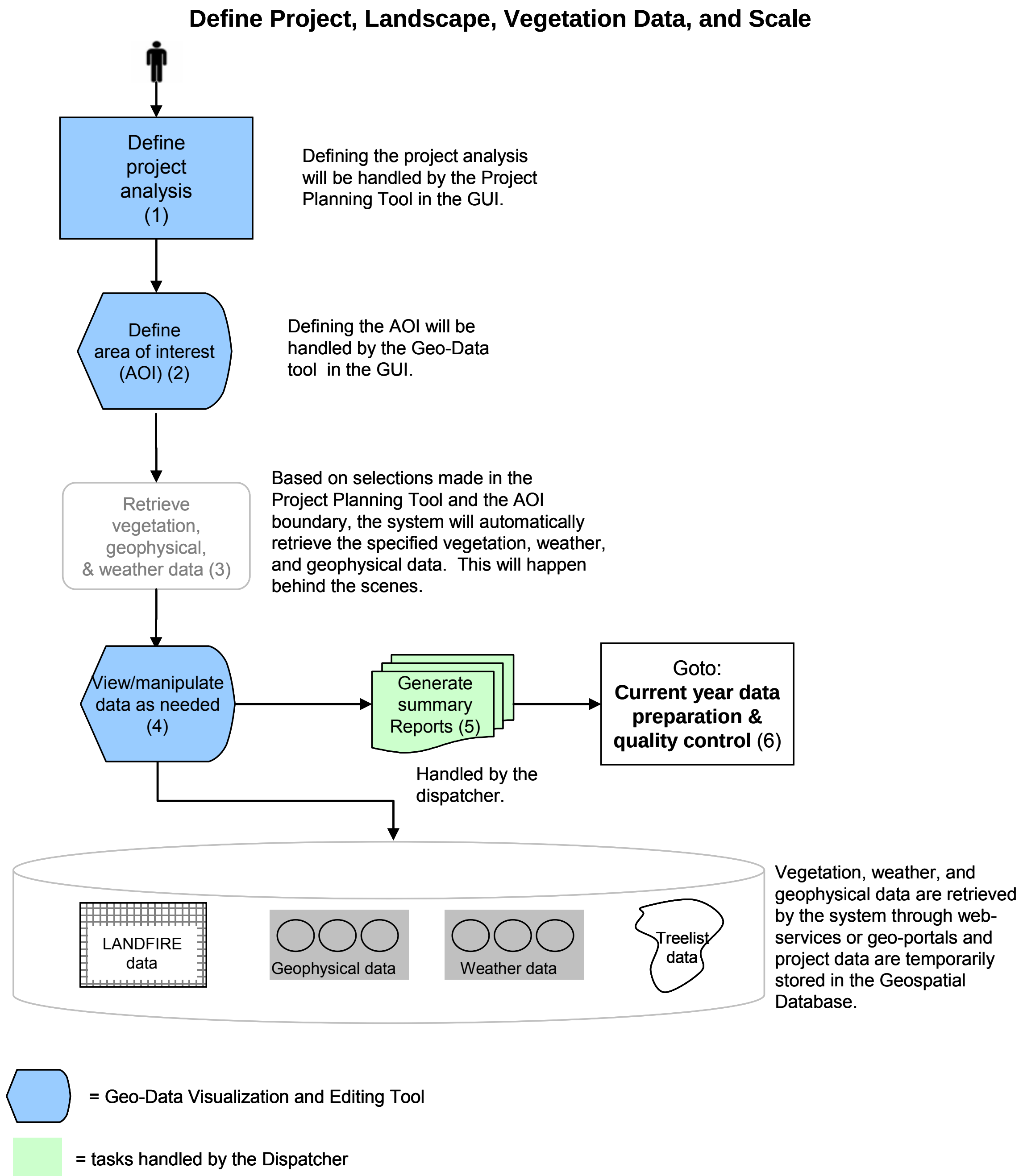

Figure 4-3. Work flow diagram for the first step in the decision support process – define project, landscape, vegetation data, and scale.

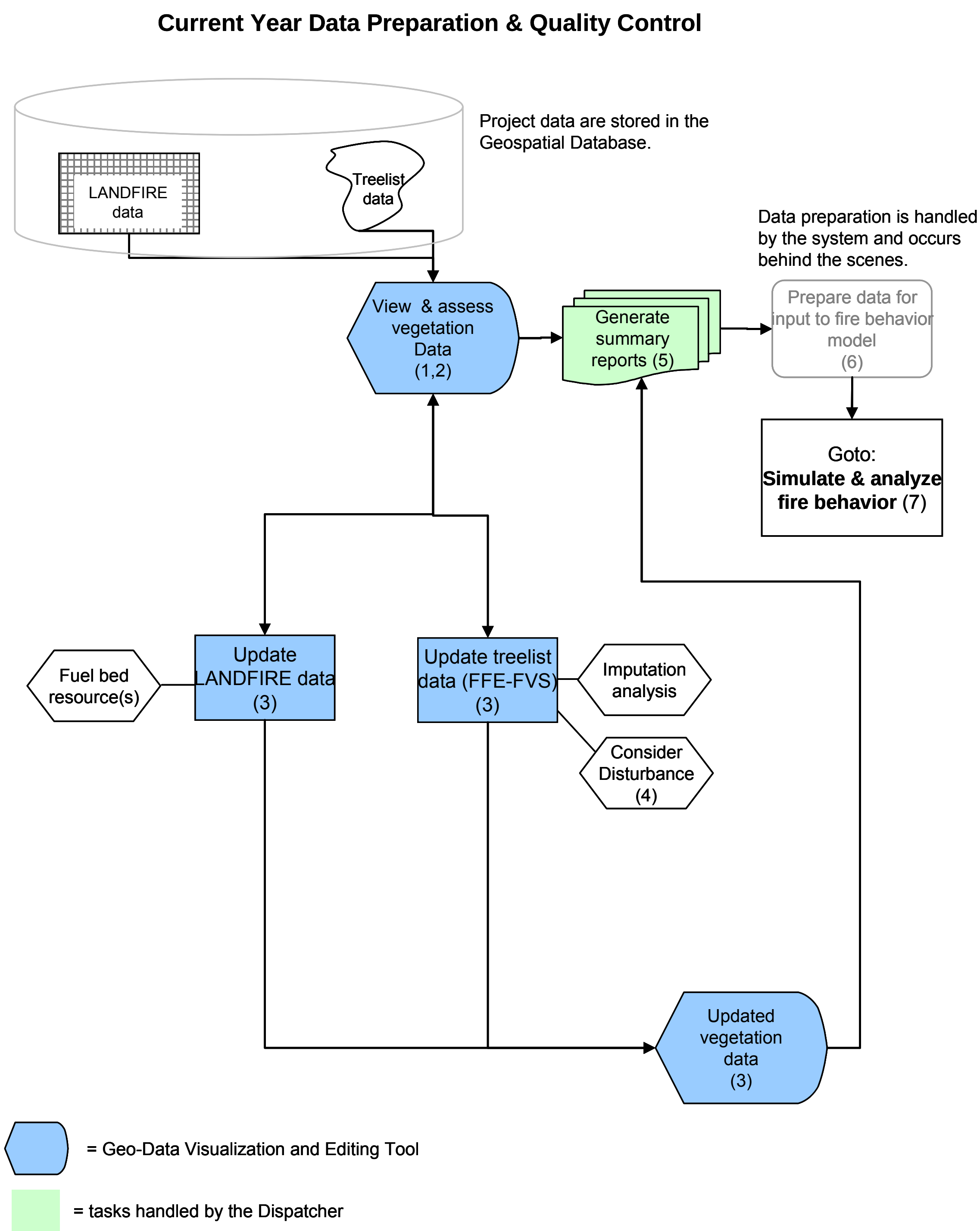

Figure 4-4. Work flow diagram for the second step in the decision support process – current year data preparation and quality control.

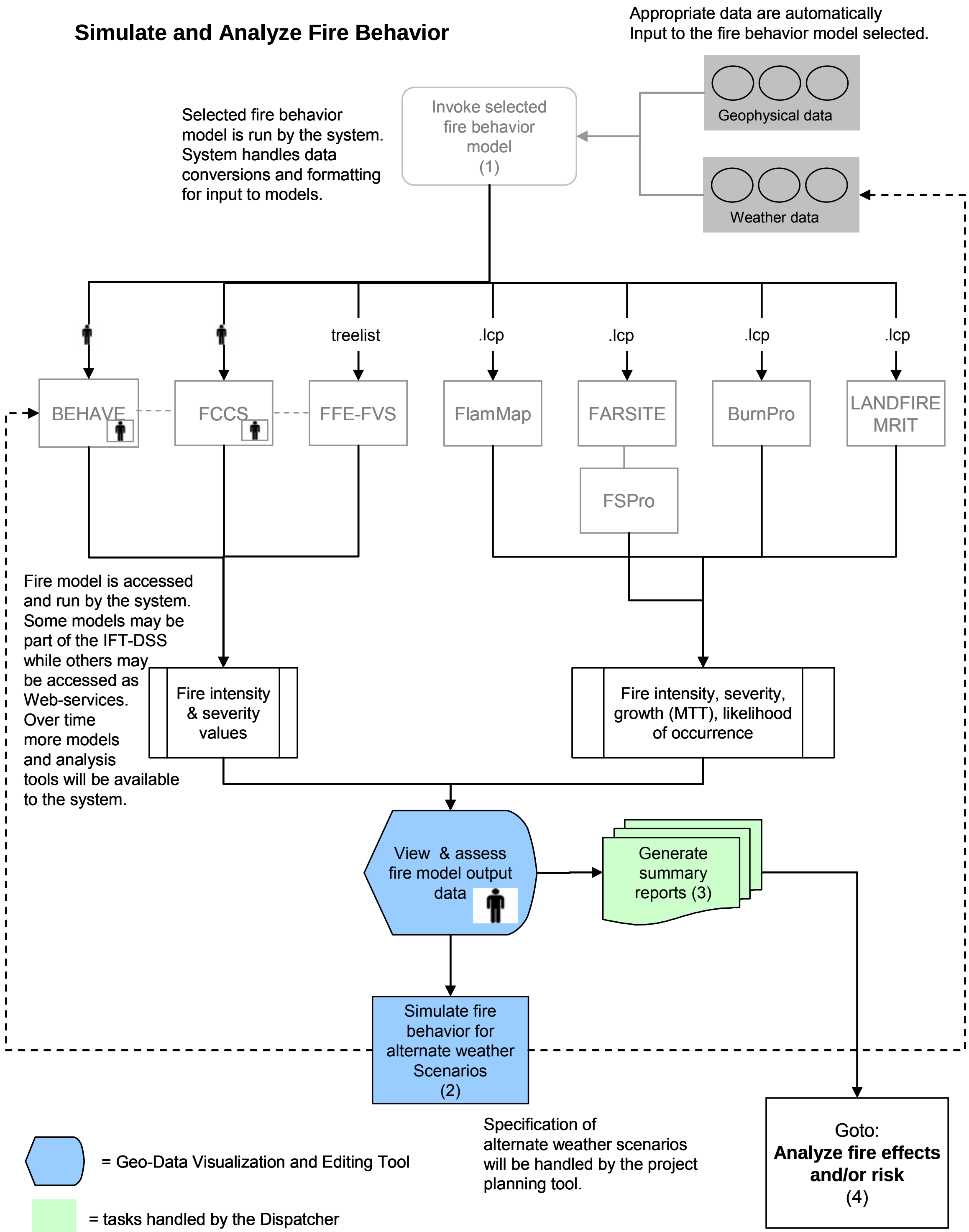

Figure 4-5. Work flow diagram for the third step in the decision support process – simulate and analyze fire behavior.

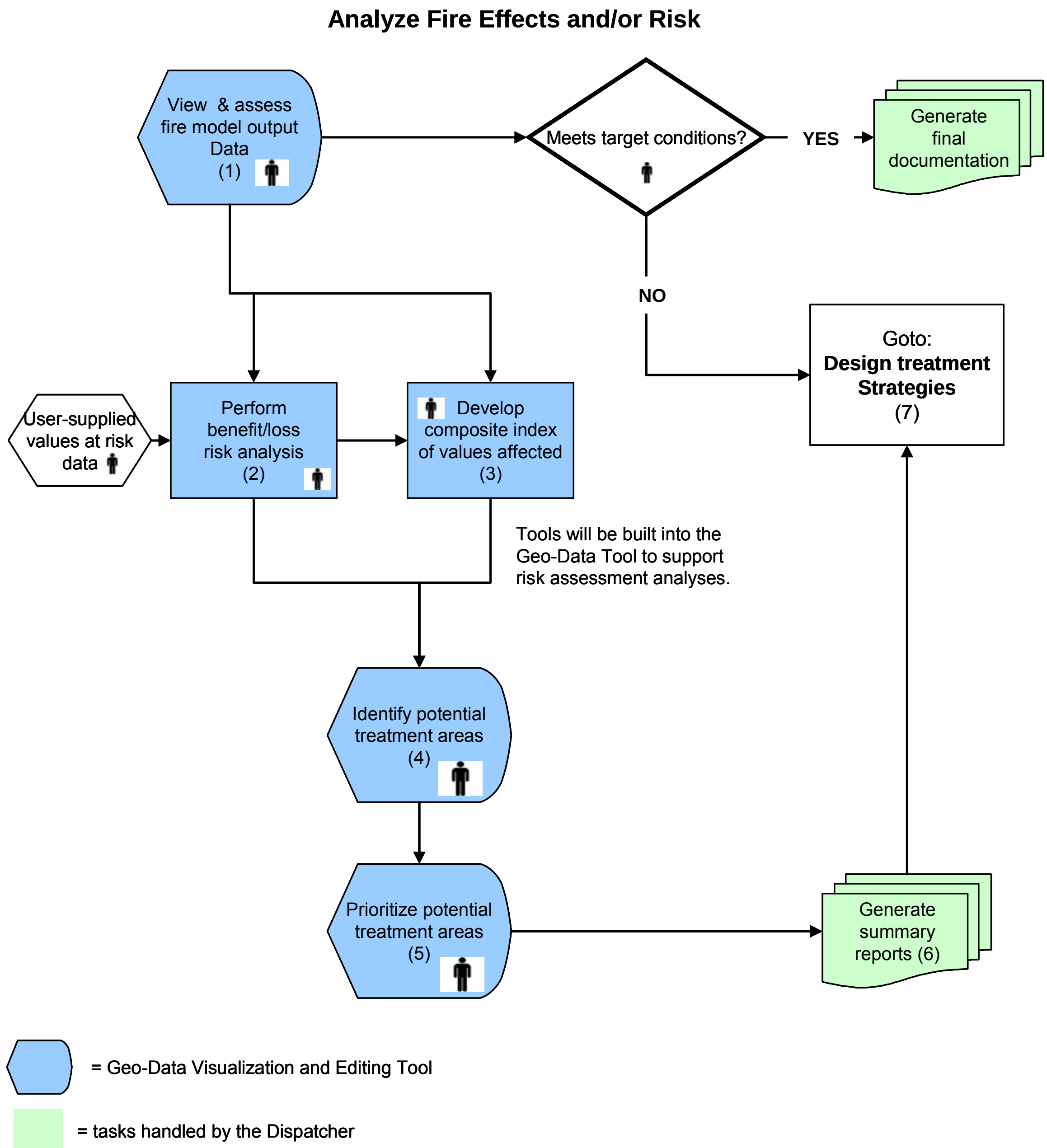

Figure 4-6. Work flow diagram for the fourth step in the decision support process – analyze fire effects and/or risk.

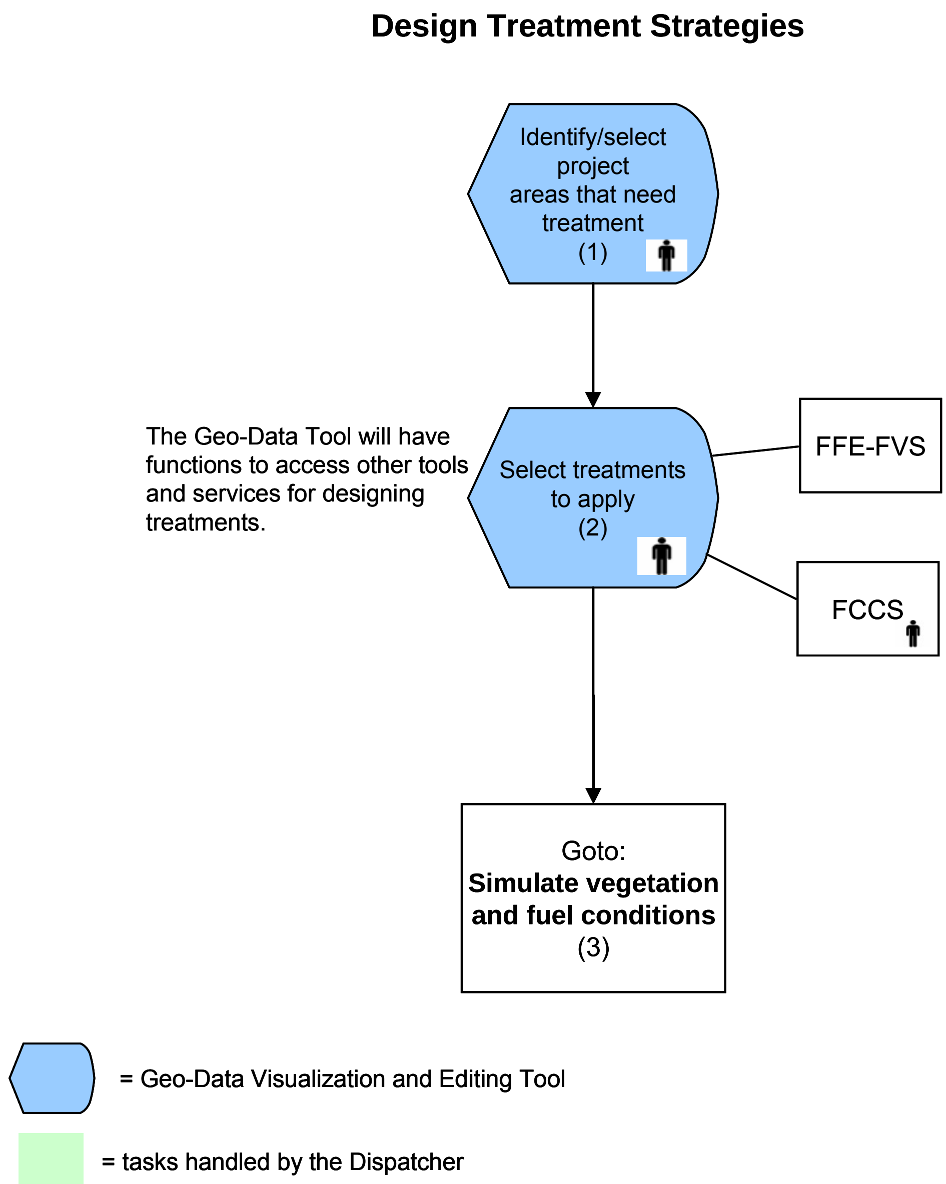

Figure 4-7. Work flow diagram for the fifth step in the decision support process – design treatment strategies.

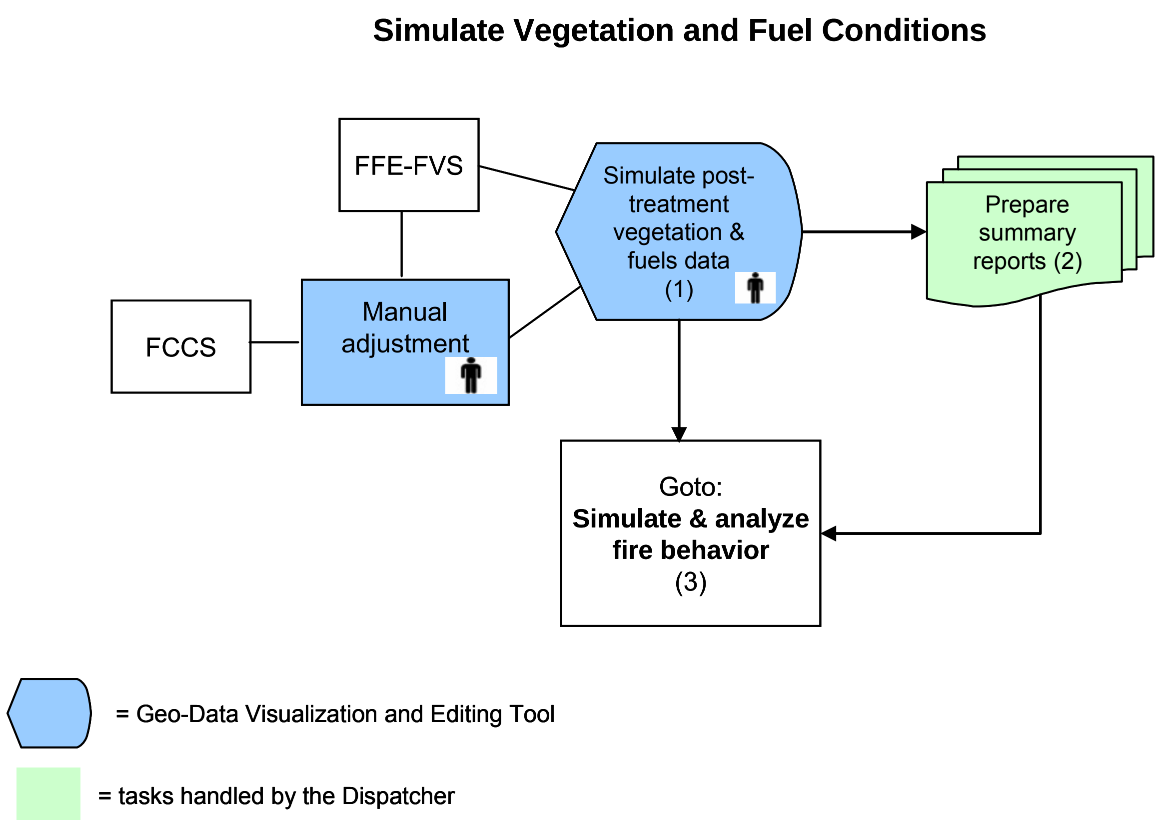

Figure 4-8. Work flow diagram for the sixth step in the decision support process – simulate vegetation and fuel conditions.

4.9 PROPOSED HARDWARE DESIGN

The IFT-DSS is envisioned to be deployed as a centralized network server. This server, which will most likely consist of several computers and networked data storage devices, will be accessed through a single website. This system will house all of the systems data and, with rare exceptions, be able to perform all the system's processes. A large, permanent data and project archive will also be associated with the system.

Clients will access the system through a web browser. All processing that must be performed on the client's computer will execute within the browser. These processes will be implemented with technologies such as Java or Flex. The most demanding aspect of the user interface will be the Geo-Data Visualization and Editing tool.

Since the system appears to the user as a website, there is an obvious need for network connectivity, but the network connection need not be either fast or reliable. It is possible for the system to be used with a minimum of network interaction. The design will take into account that simulations may take a long time to execute, and the user may not stay logged in during that time.

Client desktop computers or other systems may also be set up to host the IFT-DSS application framework. Thus, researchers and collaborators can create a customized system and avoid the performance overhead of running over a wide area network (the internet). In addition, it could allow running many of the services without an internet connection (offline).

4.10 TECHNICAL CONSIDERATIONS

A requirement of the IFT-DSS is that users should not be required to install proprietary or specialized software on their local machine(s) and that intensive computational processing will occur on high-performance, remotely hosted servers. This architecture specifies that client interactions will occur via a user interface (GUI) running in a browser on the client laptop or workstation, while the significant computational processing (simulations) will run on a hosted server to avoid users having to load software onto their local machine(s). This approach has trade-offs, and one key aspect of the system design is to ensure that large data sets are not passed between the remote computing server and the client machine. This activity can be done by designing the system so that only presentation data (intermediate and final results) are passed to the client browser to avoid passing large amounts of data across the Internet. This approach will alleviate potential network bottlenecks in the client-server interface. The details of how this interface will be handled by the system should be part of the IFT-DSS prototype design specification document.

There will be significant requirements on the back-end servers to support a potentially large number of users running simulations on the system and storing a large amount of data. The scalability of the system is the capability of the system to accommodate a large number of users and data. The architecture will support increasing the number of servers, increasing the number of central processing units (CPUs), and increasing the amount of disk space and/or database size as necessary to support the number of system users. System scalability should be addressed specifically in the design specification document for the IFT-DSS. That design will include specifications for testing the system and the specific performance expectations that the system will be expected to meet.

System stability and recovery will be handled by ensuring that the system is designed so that data and work are saved at each stage of a project analysis. The only work lost would potentially be any that was in the process of being edited or generated during a server crash. Software is not usually designed to ensure that no work is ever lost, and it is better to spend design effort developing mechanisms to prevent system crashes. It is envisioned that in the case of a server crash, the user would lose their connection to the system and would have to log back in; however, their intermediary data would remain in the system.

5. COLLABORATION AND STANDARDS

A key challenge in the successful design and development of the IFT-DSS is that the many different data sets and applications to be integrated within the IFT-DSS require different inputs and produce different outputs in many formats and structures. To enable collaboration, standards for data, metadata, and application programming interfaces (APIs) will be used. This need is magnified by the desire to enhance collaboration between the IFT-DSS and other systems within the fire and fuels domain (e.g., BlueSky and WFDSS). Fortunately, a wealth of useful standards have already been created by various scientific, geospatial, and computer science communities and adopted by global standards organizations.

While there are too many data and process standards to detail in this document and the work of identifying the most useful standards is not complete, there are three families of standards likely to be useful within the IFT-DSS—OpenGIS, OpenMI, and DAP:

- **OpenGIS Standards and Specifications** detail a suite of interfaces and encodings for spatial data. They were developed by the Open Geospatial Consortium (OGC), which is a non-profit, international, standards development organization. The geospatial standards developed by the OGC have been widely adopted by the geospatial community as well as by government agencies such as USGS and NASA. OpenGIS standards that may be used in the IFT-DSS include the Web Map Service Interface Standard (WMS), Web Feature Service Interface Standard (WFS), and the Web Coverage Service Interface Standard (WCS). These interface standards each specify service APIs (including operations and message syntax) for moving spatial data over the Internet. These standards are open and the specifications are freely available (http://www.opengeospatial.org/standards).
- **OpenMI** is a standard developed within the hydrology community for linking computational models. Model components that comply with the standard can be linked and run without requiring any programming. The standards and software developed by the OpenMI Association represent a simple approach to dealing with connecting modeling systems with multiple spatial and temporal dimensions, <http://www.openmi.org/>. OpenMI currently supports modeling systems written in Java or .NET languages.
- **Data Access Protocol (DAP) 2.0** specifies a standard way to store data in a hierarchical repository (logically similar to a remote file system). DAP is especially useful for unstructured data which are large and not suited to a relational database. Scientific data are often of this type, where a data set can consist of a very large set of values, which need to be kept together in some large file. A DAP repository can handle such data efficiently, even if the data are binary and compressed. DAP was originally developed for oceanographic data, but has since been generalized and adopted by other communities, such as the meteorological and space science communities. OpenDAP is an open source implementation of the DAP 2.0 standard, <http://opendap.org/>.

In addition to these open source standards and protocols for passing data and model information within the system, the IFT-DSS will need to support other formats for import and export. For example, the ESRI Shapefile format is a de facto standard for spatial data, and the

IFT-DSS will need to provide Shapefile import/export capabilities. Other file types that the IFT-DSS will support include common formats in the fire and fuels community, such as the landscape (.LCP) file, which is used to drive several fire behavior models.

6. ASSUMPTIONS AND IMPLICATIONS

Developing a software architecture involves making many solution assumptions and decisions that have implementation implications. In this section, the key architectural decisions that have the greatest implications on either the success of the IFT-DSS, the deployment, or both, are described. **Table 6-1** summarizes the decisions and implications of the IFT-DSS architecture approach proposed in this document.

Table 6-1. Summary of the architectural decisions and associated implications for the IFT-DSS.

Page 1 of 2

Architectural Assumption	Implication
1. The IFT-DSS will integrate and support existing and future data and software applications.	Success of the IFT-DSS program will rely on the participation of data and application developers and the adoption of collaboration standards. Data standards and API standards that are developed or chosen for IFT-DSS must be carefully considered, so that they are likely to satisfy the needs of existing fire and fuels applications as well as new applications (to the extent possible).
2. The IFT-DSS will implement data and software applications as services that can be made available to other decision support systems.	The IFT-DSS project will need to include documentation and testing of services being published. The hosting system will need to provide links to associated documentation, and the hosting agency may need to supply technical support. Security must be considered for each published service, and the design must consider that the number of potential users includes not only direct IFT-DSS users but also users of the other systems that interface to it.
3. The IFT-DSS will be web-based to facilitate accessibility and collaboration within the fire and fuels community.	Use of the IFT-DSS will require Internet connection. An option that might be considered is development of both a web-based system and a downloadable version of the system that could run on a local desktop machine. The implication is increased development and maintenance costs. There are security implications, and the design will need to ensure that user data are secure, and that malicious users or web scripts do not compromise the availability of the server. The entire design will need to be reviewed with respect to security issues.
4. The IFT-DSS will have a simple-to-use but sophisticated web-based GUI.	A web-based GUI simplifies deployment to client machines, but puts significant requirements on the hosting server. During design, decisions will need to be made about which web browsers will be supported.

Table 6-1. Summary of the architectural decisions and associated implications for the IFT-DSS.

Architectural Assumption	Implication
5. The IFT-DSS will make use of geospatial technology and adhere to commonly used GIS data standards.	The IFT-DSS will contain sophisticated GIS functionality and will support commonly used GIS data standards. The GIS platform has not yet been determined.
6. The IFT-DSS will have a dedicated hosting agency and maintenance plan and will be operational 24/7.	The IFT-DSS will require a dedicated hosting agency, appropriate hardware and software resources, and ongoing support and maintenance. The system will adhere to the IT Investment Process and will meet interagency IT requirements.
7. Technological implications for the proposed structural architecture.	The servers that host the IFT-DSS system will need to reside in the same physical location, on a high-bandwidth local network, to ensure good performance. These servers will include one or more application servers, and one or more database servers. It may also include one or more DAP servers.
8. The user will have the ability to log off the system while long simulation processes are continuing.	System performance can suffer when too many long processes are invoked at once. When users are logged off, it can be difficult for the server to detect and remove unnecessary processes. The design will need to include strategies for avoiding and handling overload situations. This will include a way of notifying the user if a batch process has been terminated (possibly via email).

7. IMPLEMENTATION PLAN

7.1 IMPLEMENTATION TIMETABLE

Full functionality of the IFT-DSS could be implemented over the course of three years. Assuming that implementation begins in 2009, the first year (2009) would involve the development of a functional IFT-DSS prototype to serve as a proof of concept (POC) system. Full implementation of the architecture and the addition of system enhancements would occur in the second year (2010). By the beginning of the third year, the system would be fully functional and operational. Beyond the third year, system enhancement and fine-tuning would occur in parallel with the ongoing operation of the system. During the first two years of development, the system could be hosted by the IFT-DSS development organization. By the beginning of the third year, the IFT-DSS would be a stable system that could be transferred to the long-term hosting agency. The system developers could continue to enhance and fine-tune the system during Year 3 and would upgrade the operational system (hosted by the agency) at the end of Year 3.

By the end of Year 2, the goal is that the IFTWG members would have access to and use the IFT-DSS on a routine basis. It is expected that by the end of Year 2 (2010), the IT Investment process will be complete, and the NWCG will be ready, willing, and able to institutionalize the IFT-DSS by 2011. At the end of 2011, the development organization would perform a final transfer (system update) of all system software, programming code, sub-models (services), etc. to the hosting agency. It is important that administrative and maintenance training be provided to the appropriate personnel when the IFT-DSS is finally delivered to the hosting agency. **Figure 7-1** illustrates the IFT-DSS implementation schedule and the requirements that can be met over the next three years.

IFT-DSS Implementation (STS Phase III)	2009	2010	2011	2012
	Year			
Implementation of Functional IFT-DSS Prototype	◆			
Implementation of Full IFT-DSS Functionality		◆		
Deliver Operational Version of the IFT-DSS to Hosting Agency			◆	
Continue to Refine and Enhance the IFT-DSS			◆	
Delivery of the Complete IFT-DSS to Hosting Agency & Training				◆

IFT-DSS Functional Requirements Implementation Timeline	2009	2010	2011	2012
Support for the decision support process, analysis steps, and software tools commonly used for fuels treatment planning.	√			
Provide data choices (i.e., standard treelist data, standard gridded data, and/or locally generated data).	√			
Have a quality control, documentation, and audit-trail mechanism to support regulatory requirements.	√	√		
Be able to be stopped or started at any processing point.	√			
Have a mechanism to perform sensitivity analyses.		√		
Support scientific collaboration; that is, the system must be able to incorporate new models and tools as they become available through an authorship and publishing mechanism.		√		
Support visualization of spatial and tabular data, data editing, and user interaction at each processing step.	√			
Have a data processing and transformation mechanism to acquire or create and transform input data (i.e., the ability to combine vector and raster data formats; support for vector-to-raster transformations and vice-versa). The system should contain mechanisms to streamline data preparation and processing.	√			
Provide guidance (i.e., sub-model choices) based on geographic scale and the type of analysis being performed.	√	√	√	
Support analytical collaboration; that is, the system should provide a mechanism for fuels treatment analysts to publish and share methods and algorithms with other system users via a central system library.		√		
Recognize user error and explain alternate action.		√		
Must have an operation and maintenance plan, long-term hosting agency with allocated servers, equipment, and maintenance staff.			√	
Should be fully operational (24/7) and reliable.		√		
Must support ArcGIS data formats and other commonly-used geographic information system (GIS) data formats.	√			
Should have a dedicated hosting agency and software and hardware resources available.			√	
Should be designed to function with high-speed Internet access (assumes users will have such access at a minimum).	√			
Must be designed for inter-operability with other decision-support systems in the fire and fuels domain (i.e., BlueSky Framework and WFDSS).		√	√	
Should be able to accommodate up to a maximum of 250 simultaneous users (about 25% of the total user community), running up to 500 computations per hour. These numbers can be increased with additional back-end hardware (i.e. a second server could be added to double those numbers).		√		
Should ensure that if a user repeatedly invokes an operation (because of issues with slow response time, for example) the repeated commands will not interfere with system stability (redundant commands will be ignored or deactivated to prevent this).	√			
Will shield the user from server crashes during system use by invoking automatic recovery processes and prompting the user to log back into the system without data loss.	√			
Will ensure a response time of three seconds or less whenever a user invokes a command in the GUI. In cases of heavy load, or if the user is in the process of running an operation, the system response should be an indication that the command has been recognized by the system and that the server is busy. In the latter case, some indication of progress such as an active status bar should be provided;		√	√	
Will store all intermediate calculations as they are produced by simulations, and any data the user enters will be stored as soon as it is received. In the case of a server-crash or if the system needs rebooting for any reason, all stored data will be available when the user logs back in. In addition, routine server back-up processes will be employed to ensure that data are not lost in the case of a server-failure.	√	√	√	
Will adhere to the IT investment process requirements.		√		

Figure 7-1. Illustration of an implementation schedule for the IFT-DSS.

7.2 OVERVIEW OF THE IFT-DSS PROTOTYPE SYSTEM FUNCTIONALITY

It is important that a functional prototype of the IFT-DSS be developed early to demonstrate its usefulness. An IFT-DSS prototype should support a subset of the most common fuel treatment planning use cases and should contain enough functionality to serve as a POC system. This section describes a functional prototype for the IFT-DSS. Note that this proposed IFT-DSS prototype is a work in progress and may change as Phase III (system implementation) of the STS evolves. **Figure 7-2** depicts the IFT-DSS generic work flow diagram (shown in Figure 1-2) adapted to illustrate the components of the system that are proposed for the IFT-DSS prototype.

The proposed IFT-DSS prototype functionality is based on input and discussion from the IFTWG meeting in December 2008 (Boise, Idaho). It was discussed among the IFTWG that a system that provides the functions shown in Figure 7-2 will be immediately useful for practitioners and will quickly demonstrate the usefulness of the system. The data, software models, and tools that are proposed to be included in the prototype are FSVeg treelist data, LANDFIRE gridded data, the Most Similar Neighbor (MSN) imputation tool, the Forest Vegetation Simulator (FFE-FVS), FlamMap, BEHAVE, and the Fuels Characterization and Classification System (FCCS). This subset of tools is proposed because they are among the most commonly used by fuels specialists as indicated by the results of a survey of fuels specialists performed as part of Phase II, Task 1 of the STS. This subset of data, applications, and tools will also make it possible for users to perform a subset of commonly performed analyses that are currently available through the use of existing systems such as INFORMS and ArcFuels.

In addition to the existing software models and data listed above, the Geo-Data Visualization and Editing tool would be developed to view, select, manipulate, edit, and convert polygon and raster data. This tool will handle the data conversions and formatting required to make data transfers and conversions seamless. In the IFT-DSS prototype, the Geo-Data tool would provide the capability to perform .LCP file conversions. The .LCP file is a common data format for several fire behavior models including FlamMap.

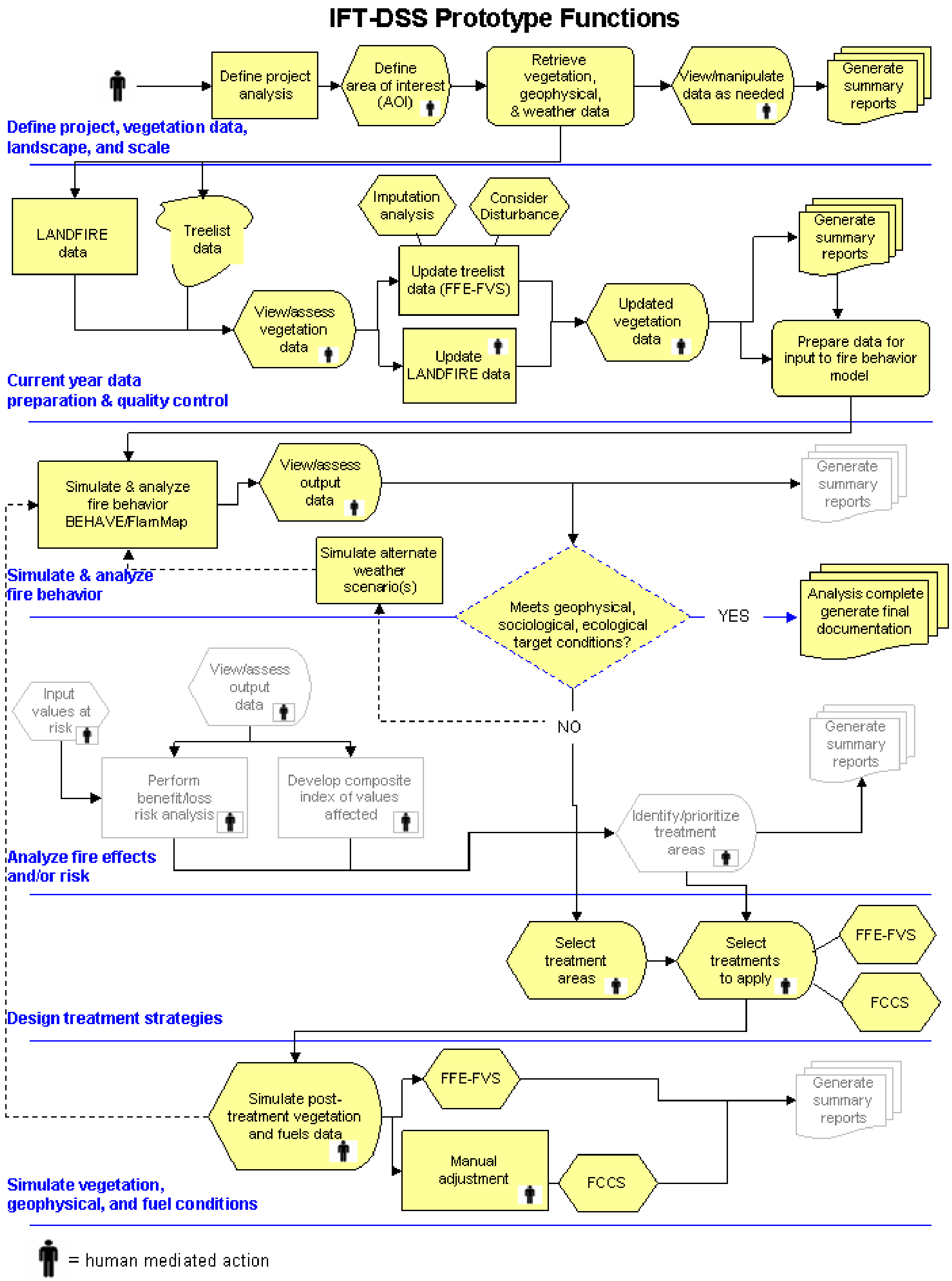

Figure 7-2. IFT-DSS generic work flow diagram (as shown in Figure 1-2) adapted to illustrate the components of the system that are proposed for the IFT-DSS prototype.

8. REFERENCES

Erl T. (2005) *Service-oriented architecture: concepts, technology, and design,* Prentice Hall PTR, Upper Saddle River, NJ.

Funk T.H., Rauscher M., Raffuse S.M., and Chinkin L.R. (2008) Findings of the current practices and needs assessment for the Interagency Fuels Treatment Decision Support System (IFT-DSS) project. Technical memorandum prepared for the The Interagency Fuels Treatment Work Group (IFTWG), by Sonoma Technology, Inc., Petaluma, CA and the Air Fire Science Team, Seattle, WA, STI-908038.01-3504, December.

Joint Fire Science Program (2008) Interagency fuels treatment decision support system conceptual design, v 0.9. Sponsored by Joint Fire Science Program, National Interagency Fuels Coordination Group, and National Interagency Fuels Management, August 18.

Palmquist M.S. (2008) Working summary of the SEI's engagement with the Joint Fire Science Program. Report prepared for the U.S. Department of Defense by the Acquisition Support Program, Software Engineering Institute, Carnegie Mellon University, April.