The Interagency Fuels Treatment Decision Support System Project: Final Report

H. Michael Rauscher
Rauscher Enterprises, LLC, Asheville, NC

John Cissel, Timothy Swedberg, Nate Benson
Joint Fire Science Program, Boise ID

Erik C. Christiansen
DOI Office of Wildland Fire, Boise, ID

Tami H. Haste, Stacy A. Drury, Neil J.M. Wheeler, David J. Noha, Lyle Chinkin
Sonoma Technology Inc., Petaluma, CA

Constance M. Bennett, Nanette Brown, Dan Doney, Linda Parker Gates, Suzanne Miller, M. Steven Palmquist, Patrick R. H. Place
Software Engineering Institute, Software Solutions Division, Carnegie Mellon University, Pittsburg, PA

Kim Ernstrom
Wildland Fire Management RD&A, National Interagency Fire Center, Boise ID

August 20, 2013

Abstract

After nearly 13 years of effort by members of the Joint Fire Science Program and five years of effort by members of the Wildland Fire Information and Technology (WFI&T) study, there can no longer be any question of the existence, characteristics, and importance of the "software chaos" problem in the fire and fuels management business mission of the US federal land management agencies. The broad outline of a solution to the "software chaos" problem were identified by the WFI&T Strategic Plan. This plan articulates a vision to enable an interagency, integrated approach to wildland fire information and technology management in support of mission activities. The plan establishes four concepts that guide wildland fire technology implementation:

- Mission requirements drive integrated, modular based applications and tools.
- Authoritative data are readily available for all uses and users.
- Interconnection and accessibility regardless of organization affiliation or user location.
- Technology, research, and innovation enable and enhance mission.

The Interagency Fuels Treatment Decision Support System (IFTDSS) was designed and implemented by the Joint Fire Science Program, working with a large number of stakeholders, to significantly mitigate the inefficiencies created by the "software chaos" problem and test the

solution vision described in the WFI&T Strategic Plan. IFTDSS is an existing service integration platform, currently with over 100,000 lines of software code, that provides command and control for software modules and datasets executing from within a common user interface. It provides capabilities for use and integration of standardized and custom datasets, supports treatment unit- and landscape-scale analyses, data visualization functionality, estimates of fire behavior and first-order fire effects, and quantitative hazard and risk assessments. It allows users to choose pre-designed solution pathways for the most commonly performed fuels treatment tasks. These pre-designed solution pathways, called workflows, were designed and reviewed by members of the user stakeholder community to ensure that offered functionality matched real needs. It is absolutely critical to understand that IFTDSS is not another new fuels treatment system. It is a service integration framework that organizes and makes available a large number of pre-existing software modules. IFTDSS is a web-based application that provides users with a single user interface to multiple software tools.

In 2012, the JFSP engaged the Software Engineering Institute (SEI), a Federally Funded Research and Development Center (FFRDC) operated by Carnegie Mellon University to perform an independent evaluation of how well the IFTDSS service integration framework succeeded in addressing the “software chaos” problem. After an extensive evaluation, the SEI concluded that IFTDSS (1) significantly improves the quality and efficiency of the fuels treatment planning process for the majority of user needs; (2) provides a concrete demonstration of one way to implement the four key WFI&T solution concepts above; (3) enables standardized, risk-based fuels management planning for a large part of the fuels specialist community; and (4) is near-ready for operational use. The evaluation concluded that if fielded as part of a comprehensive governance strategy, IFTDSS can be a major step in bringing order to the “software chaos” problem.

The objectives of this final report are to (1) describe the existing “software chaos” problem in the business mission of wildland fire and fuels management in the land management federal agencies of the United States; (2) describe the design of the solution the Joint Fire Science Software Tools and Systems study produced to ameliorate the “software chaos” problem; (3) present the IFTDSS service integration framework software architecture and functionality; and (4) discuss the results of an independent evaluation of the IFTDSS software in addressing the “software chaos” problem.

Introduction and Background

During the past two decades, a large number of data, software applications, and analysis methods have emerged in support of the business mission of wildland fire and fuels management in the federal land management agencies of the United States. Technological advances in spatial data availability, spatial analysis capabilities, simulation modeling technologies, and probabilistic risk assessment methodologies have changed how we plan and implement fire and fuels management strategies (Gollberg et al., 2001). These advancing software and hardware technological capabilities and the increasing understanding of how to apply these capabilities to the business of wildland fire management led to the production of numerous software applications. The complexity of fuel treatment management, especially with a probabilistically-based, landscape-scale risk assessment focus, requires many different types of software applications to support a solution process from beginning to end. Database management software applications are needed for viewing, organizing, and editing spatial as well as point-based data. Fire behavior applications simulate the ignition and spread of fire given specific fuel and weather conditions both within individual forest stands as well as forested and non-forested landscapes. Multiple fires are frequently simulated over the same landscape, fuel, and weather conditions "in order to analyze the uncertainty associated with wildfire events in terms of the timing, location, intensity, and duration" (Ager et al., 2011). Another set of software applications models the effects of fire on a variety of ecosystem components such as the vegetation, animals, fuels, soils, and air (Reinhardt and Dickinson, 2010). Yet more software applications simulate the change over time of the vegetation under various site conditions as affected by agents of change such as fire. A listing of existing software applications dealing with some aspect of the business mission of fire and fuels management quickly reaches into the multiple hundreds.

In June 1999, the Joint Fire Science Program (JFSP) sponsored a 3-day conference and workshop in Boise, Idaho, entitled '*Crossing the Millennium: Integrating Spatial Technologies and Ecological Principles for a New Age in Fire Management*'. Discussions at the workshop of this conference resulted in 7 recommendations to the Governing Board of the JFSP from the collective scientific and management experts attending this conference (Gollberg et al., 2001). These recommendations identified research, technological and management opportunities to improve the business mission of wildland fire and fuels management. The recommendations pointed out that software tool use in support of wildland fire and fuels management needed attention. It is worth reproducing them in their entirety (Gollberg et al., 2001):

- **Recommendation 1:** *Management tools including databases, maps, and models should be grounded in ecological research and principles. An emphasis should be placed on landscape level tools that– where appropriate–incorporate biological, biochemical, climatological, ecological, geological, and morphological factors.*
- **Recommendation 2:** *A more integrative and systematic national approach to fuel mapping and modeling that sets standards and protocols across multiple spatial and temporal scales and that includes a ground campaign to provide accurate model input data is needed. Within this framework remotely sensed data, GIS technology, and models should be better integrated and documented.*
- **Recommendation 3:** *The process of tool use including comparison, selection, acquisition, training, implementation, evaluation, and support needs national administrative focus, guidance, and support.*
- **Recommendation 4:** *Technology development, transfer, and communication need to be improved between developers and user communities.*
- **Recommendation 5:** *Collaborative approaches to research, development, and implementation of new information and decision support tools need to be encouraged.*

- **Recommendation 6:** *More precise and consistent definitions and standards are needed for fire severity, hazard, and risk.*
- **Recommendation 7:** *There should be a new emphasis on training that incorporates the latest developments in remote sensing, geographic information systems, information management, and communications technologies.*

The JFSP Board took these recommendations seriously. In 2005 and 2006, Tim Swedberg, JFSP Communications Director, was asked to perform a nation-wide follow-up analysis to determine the priority issues facing wildland fire and fuel managers. The JFSP wanted to align its program of work to support manager's needs. Swedberg met with the Interagency National Wildfire Coordinating Group (NWCG) committees, especially the Fuels Management Committee, attended fire and fuel managers winter meetings in all regions of the nation, visited local and regional fire and fuels managers at their home locations, and attended national training classes to query the opinions of participants.

What became characterized as the "software chaos" problem started to crystalize. Managers nationally identified the confusion and inefficiencies associated with the many existing software systems available for fire and fuels decision making as one of the most pressing problems they face. Managers face a large assortment of unconnected software applications in various stages of development with little or no guidance concerning the strengths and weaknesses of the various applications, and no framework for integration and fusion of data and outputs from these systems. Worse yet, any one of the available software applications perform only a portion of the necessary mission critical task. The software applications are isolated from one another which forces managers to manually transform data sets from one application so that the next application can import that data and perform the next stage of the analysis. These isolated, stand-alone software applications have proliferated in the last few decades in response to various funding initiatives without any central control or vision to bind them together. The JFSP Board of Directors recognized that their own program of work, among others, had significantly contributed to the creation of this "software chaos" by helping to fund it.

Having identified the "software chaos" problem (2001) and having gotten verification from the management and science community that it is real (2006), the JFSP in partnership with the National Wildfire Coordinating Group (NWCG) Fuels Management Committee initiated the Software Tools and Systems (STS) Study in 2007 to investigate what could be done to alleviate the problems caused by this "software chaos". The STS study was conducted in a series of phases to accommodate evaluation of the study between phases and to allow for planned course corrections throughout the duration of the study. Phase I (April 2007 – March 2008) was focused on problem identification and analysis. Phase II (April 2008 – March 2009) resulted in designing a solution pathway using the web-based, service oriented architecture methodology to build an example system called the Interagency Fuels Treatment Decision Support System (IFTDSS). Phase III (April 2009 – May 2010) produced a proof of concept version of IFTDSS to make sure the technology was adequate to accomplish the project goals. Phase IV (June 2010 – October 2012) saw the full implementation of IFTDSS ending with the completion of version 2.0. Phase V (November 2012 – September 2013) focused on an independent evaluation by the Carnegie Mellon Software Engineering Institute (SEI) of the entire STS study strategic vision and of the example implementation software, IFTDSS. Full and detailed documentation of all STS project products can be found at: www.frames.gov/iftdss .

Objectives

The objectives of this final report are to (1) describe the existing "software chaos" problem in the business mission of wildland fire and fuels management in the land management federal agencies of the United States; (2) describe the design of the solution the Joint Fire Science Software Tools and Systems study produced to ameliorate the "software chaos" problem; (3) present the IFTDSS service integration framework software architecture and functionality; and (4) discuss the results of an independent evaluation of the IFTDSS software in addressing the "software chaos" problem.

Defining the "software chaos" problem

Over the past two decades, many software tools have been developed to help fuels specialists decide where, when, and how to manage vegetation (natural fuels) to reduce the risk of wildfire. This proliferation of tools has come in response to various funding initiatives, with no guiding central governance or vision. All these tools can be effective in the right hands and for the appropriate purposes; however, the number of tools available, limited guidance on their use, and the time spent assembling data and learning each individual system has created frustration in the fuels planning community (Wells, 2009).

The interagency business mission of fire and fuels management and the software applications supporting that mission involves five major stakeholder communities (Fig. 1): fire and fuel operations managers (the users of software applications), scientist developers (the software application service providers), database stewards (the official data service providers), IT software managers (the enablers that host services, provide security, etc.), and agency senior managers (the leaders that provide governance oversight) (Rauscher et al., 2009). Members of these five stakeholder communities have developed specific roles and operating methods among themselves and between themselves that are important in understanding the existing "software chaos" problem.

At the present time, new software applications most frequently arise from research scientists, working in the fire and fuels management business mission, attempting to make their contributions useful to the local fuels treatment specialists. If these research developed tools meet a mission critical need, then the local fuels treatment specialists and their local immediate supervisors make independent, locally-based decisions to help fund this new software tool and its application in performing needed analyses (Fig. 2). In the fire and fuels management business mission, there are examples of large-budget, large scope, software applications that were planned and directed by members of the senior manager community. But they are the exception and are a recent development within the last 3 to 5 years. The local decision actions then result in a large and growing number of small scope software base models that perform some aspects of the larger fire and fuels management business function need. Middle and upper level agency supervisors, policy makers, and decision makers, i.e. the agency senior management community, have relatively little role to play in the existing condition. IT software managers as well have little role to play since almost all of the base model software tools in support of the fire and fuels management program are PC-based and developed by researchers quite independent of any IT software manager oversight. In fairness to the different stakeholder community members, the locally developed software typically begins its existence being classed as a research software application and thus outside the purview of the operational controls developed by senior managers and IT software managers. The current situation turns into the "software chaos" problem because these research-grade software applications are in fact used by the fire and fuels operations managers to plan, design, implement, and justify official mission critical work.

Each stakeholder community struggles with different issues as a result of functioning within the existing "software chaos" matrix:

- Fire and fuel operations managers
    - Too much time spent learning numerous software interfaces
    - No good way to choose among a profusion of unconnected, fragmented, and overlapping software applications. Maturity of the software application, documentation availability, and degree of developer support all vary widely.
    - Too much time spent training and retraining on using numerous software applications
    - Too much time spent in acquiring and preparing data for analysis
    - Hazard and risk analyses are too complicated for most fuels treatment planners to do and are not currently required in the applicable position minimum standards
    - Software application and data availability are dependent on agency and geographical location
    - Users cannot easily share work, compare notes, teach each other methods across agency and location resulting in most users working in isolation from each other on similar problems.
    - Available software systems do not typically support the entire mission critical fuels treatment planning process
    - No process exists for users to provide feedback requests for improvement in a software application to the senior leadership community for vetting, prioritization, and possible funding.
    - Agency security requirements limit the ability of some users to install independent "non-approved" software applications.
- Scientist Developers
    - Research scientists, motivated by the desire to transform original science into software applications for managers, face important constraints:
        - Resources required to develop and deploy a software application often compete with resources available to perform new science. It has been

    estimated that between 40 – 60% of total development cost is spent on programming the Graphical User Interface (Lewis and Rieman, 1994). Eliminating the need for research model developers to program a GUI would essentially reduce their share of the cost of software tool development in half.
    - Lack of professional software engineering and architecture skills frequently resulting in code that is not modular, not well documented, and very tightly coupled with the graphical user interface.
    - The level of resources available to develop and deploy a software application is usually well below what is really needed and well below the common software industry standard.
    - Frontier science necessarily has a narrow and deep focus so that results are typically much smaller in scope than the scope of mission critical problems that operational managers routinely deal with. This results in the development of numerous software applications with narrow scope from the point of view of operational managers.
    - Research scientists frequently have a much shorter timeframe in which they are interested in developing and supporting a particular software application than that of operational managers. Scientists naturally move on to other research topics, they retire, they transfer into administrative roles, and frequently funding shifts occur that remove resources to support existing software maintenance.
  - No widely agreed upon standards exist for developing research software applications such as graphical user interface standards, modularity standards, verification and validation testing standards, or documentation standards. This makes it difficult and costly update the underlying scientific models and to manage version control properly.
  - No process exists to enable developers to move a research-grade application to operational-grade under senior management governance control.
  - Research developers are forced to fund and manage the entire life cycle process of an existing application.
  - No process exists for research developers to obtain feedback from the user community in a controlled manner that is vetted and prioritized by the senior management community. This results in requests for improvements from individual users to individual developers with no funding resources made available to perform those requests or any consideration of how important those requests are in a more strategic frame of reference.
  - Developers have no good way to easily share work, compare notes, teach each other methods resulting in most software application development teams working in isolation from other software application development teams.
- Database Stewards
  - Data access restrictions commonly exist across federal agencies and location.

    - Authoritative, web-enabled databases of record to support the fire and fuels business mission are uncommon with the notable exception of the LANDFIRE Database.
    - Missing or outdated data for fire and fuels analyses at the appropriate scale creates enormous problems for the fuels treatment specialist especially for landscape level problems.
    - Data editing and viewing software capable of reading common input formats and producing data files in common output formats are not widely available to fuel treatment planners.
    - Standards and guidelines have not been developed to help fuels treatment planners in preparing scale-appropriate, analysis ready data sets for typical fire and fuels treatment analyses.
- IT Software Managers:
    - No interagency TT policies and security requirements exist.
    - Agencies maintain their own official server farms with different software installation, management, and cost requirements.
    - No effective and efficient process, standards, and guidelines exist for helping developers hand-off their research-grade applications to a software life-cycle management process supervised by the senior management stakeholder community.
    - No interagency Personal Privacy Information management and security requirements have been worked out and those that do exist seem to severely constrain, if not prohibit, user collaboration on inter-agency projects.
    - No inter-agency standards, guidelines and procedures have been developed to describe the minimum requirements of an officially sanctioned, operational-grade software application.
    - No process has been created to resolve the problem of research-grade software applications that have not been reviewed or specifically authorized for use in official analyses for the fire and fuels business mission.
- Agency Senior Managers:
    - Limited centralized knowledge about what fire and fuels business mission software applications are being developed and how much they cost to develop and maintain. The costs for officially recognized and supported software projects are well known, reported and tracked in such systems as the inter-agency IT Dashboard project (see www.itdashboard.gov/portfolio). Unfortunately, the majority of the software tools used most frequently for mission critical work in the fire and fuels management business mission are not officially recognized and supported. Funding decisions for improvements to these unofficial software applications and development of new ones are made at the local, field unit level in cooperation with research scientists. Senior leadership in the agencies does not require upward reporting and consolidation of costs associated with these unofficial software tools.
    - Supervision of and quality control for the research-grade software applications currently being used in official analyses for the fire and fuels business mission

are made at the local, field unit level. Senior leadership does not require a uniform and documented quality assurance, quality control process reported to the agency level. This results in the lack of a consistent, and quality assured fuels treatment analysis framework across agencies and location.

  - Supervision of and quality control of the functional processes that field users use to perform analyses for the fire and fuels business mission occurs at the local, field unit level. Senior leadership does not require a uniform and documented quality assurance, quality control process reported to the agency level.
  - Senior leadership has no strategic framework to provide a context for making resource allocation decisions concerning the development and maintenance of operational-grade software applications for the fire and fuels business mission.
  - No centralized process has been developed for accepting, vetting, and prioritizing user feedback ideas and improvement suggestions for existing and newly developed software applications in the fire and fuels business mission.
  - No centralized process has been developed to match existing software application capabilities with mission critical needs to pro-actively identify gaps and deficiencies and prioritize new software development projects for future funding. This results in new science and technology not being readily identified and integrated into mission critical fuels treatment planning processes in a timely manner.

"In summary, at present there is no overall governance of wildland fire investments, no agreed upon vision or strategy for making future investments, and limited standards or protocols for data and management. Each agency maintains separate, parallel organizations. The NWCG provides some coordination of user requirements and voluntary standards. The decision space of each organization is limited. A number of applications provide important support to wildland fire planning and operational activities. But significant inefficiencies exist in sharing of data, project management, and application support. There is no consensus view on business requirements and priorities, nor is there an agreed upon strategy or vision to guide new investments or evaluate the efficacy of current investments." (Douglas and Phipps, July 15, 2011).

Designing the Solution: Vision and Strategy

In 2008, members of the Carnegie Mellon Software Engineering Institute, a federally funded research and development center sponsored by the U.S. Department of Defense, completed a year-long analysis of the "software chaos" problem described in the previous section. The SEI team formulated two recommendations for the JFSP to consider (Palmquist, 2008):

- The wildland fire community should adopt a Service-Oriented Software Architecture Framework (Erl, 2005; Newcomer and Lomow, 2005), also known as a Service Integration Framework, as the basis for a technical solution to the software chaos problem. The framework provides a common user interface for the integrated scientific software modules, standardizes communications and control protocol for the modules, and manages the common shared data that serve as input and output to/from the modules. The user sees just one integrated application.

- The goal must be to design and deliver a "whole product," which is the technology introduced plus everything else needed for the technology to be accepted and used. This means that major stakeholder groups must be identified and their interactions between themselves and the software technology product defined. A service integration framework in isolation can rarely function properly without the relevant stakeholder communities being properly organized to take advantage of its power (Rogers, 2003).

Designing the IFTDSS Service Integration Framework

The JFSP convened a 10-member Fuels Treatment Working Group consisting of field practitioners as well as scientist developers with experience in the fire and fuels business mission of the USDA Forest Service and the DOI Land Management Bureaus. This Working Group recommended that (1) a web-based service integration framework application should be created to test the utility and practicality of the SOA software methodology recommended by SEI; and (2) that the test framework application focus on fuels treatment management. The scope of the entire fire and fuels business mission was judged too broad to tackle in a single Service Integration Framework given the time and resources available. The most pressing problem in the fire and fuels business mission was judged to be fuels treatment management. This test application received the name of Interagency Fuels Treatment Decision Support System (IFTDSS). As its first task, the Fuels Treatment Working Group produced the IFTDSS Conceptual Design Document to provide the functional guidance for the project (JFSP, 2008).

In 2008, Sonoma Technology Inc., Petaluma CA, was engaged to provide the software engineering expertise to guide the design of a software integration framework using SOA principles and methods. A review of existing SOA-based systems revealed five in use outside the fire and fuels arena and two in use within the fire and fuels business area—the Wildland Fire Decision Support System (WFDSS) and the BlueSky Framework (Larkin et al., 2009a). The WFDSS was developed to support strategic and tactical decisions regarding real-time fire management, and the BlueSky Framework was developed to analyze and manage smoke impacts from fires. The WFDSS combines desktop applications for fire modeling into a web-based system for easier data acquisition and provides an easy way for fire managers and analysts to accurately document their decision-making process. It organizes and manages its services to provide one standardized decision process and documentation system for all types of wildland fires. Because it is a web-based application, it facilitates analytical collaboration and sharing of analyses and reports across all levels of the federal wildland fire organization (http://wfdss.usgs.gov/wfdss/WFDSS_Home.shtml). The BlueSky Framework combines desktop applications for fuel consumption and emissions simulations to produce estimates of emissions from fires. It consists of software framework programming code and accompanying models (services) that can be downloaded from a website and run on a local desktop machine. The Framework offers various model (service) choices at each step of a smoke impacts assessment (Larkin et al., 2009b). While both of these framework systems represent significant advances in software engineering over the more common stand-alone, desktop software applications in the fire and fuels business mission, neither BlueSky nor WFDSS offered a generic SOA-design that could be readily modified for a different business function such as fuels treatment management.

It should be mentioned that one stand-alone, desktop software application exists for comprehensive fuels treatment planning that pre-dates the IFTDSS project and was also partially funded by the Joint Fire Science Program. ArcFuels is a fuels treatment management and risk assessment software application offering a limited integration of existing stand-alone software applications for designing and testing fuel treatments (Ager et al., 2011). ArcFuels is written "using the ArcObjects library and Visual Basic for Applications within ArcMap." ArcFuels does an excellent job of automatic data transformations between resident modules thus saving users much time and effort. However, ArcFuels retains the native user interface of each of the resident software applications and thus requires users to develop and maintain expertise in using them. ArcFuels is an extremely powerful application if used by experienced fuels treatment planners that have an expert level understanding of GIS theory as it has been implemented by the proprietary ESRI ArcMap software product. ArcFuels offers the most powerful and flexible risk assessment process for fuels treatment planning available today. Unfortunately, the tight coupling of ArcFuels with the proprietary ArcMap and the use of ArcFuels scripts that could not be used outside the ArcFuels application made it logistically extremely difficult to port into a web-based, service integration framework, such as IFTDSS. Discussions with the developer of ArcFuels early in the IFTDSS project lead to the important recognition that there exists a tension between offering a flexible, powerful risk assessment process that relatively few agency experts can use properly and a standardized, yet credible, risk assessment process that can be used by most fire and fuel operations managers. This issue will be discussed more fully later in the paper.

A field user survey was conducted in 2008 - 2009 to identify which software applications were actually being used for fuels treatment management (Rauscher 2009). It is estimated that the various federal agencies employ approximately 700 – 800 fuels treatment specialists. The 44 responses to this survey translates approximately into a 5% sample of the population of fuels treatment specialists. The sample was random in the sense that everyone had the same opportunity to respond. It was biased in the sense that the respondents were self selected and thus constitute a particular subset of the entire population with unknown characteristics. However, the point of the survey was not to produce a statistically valid result but rather to obtain an impression of what software tools fuels specialists used most frequently and conversely, which ones they did not use (Table 2). The results of this survey were used to prioritize the existing software applications that were to be brought into IFTDSS. The results of this survey were instrumental in reducing the volume of software applications that needed to be considered for inclusion in the service integration framework from the hundreds to less than 20.

Designing the Stakeholder Interaction Strategy

The literature of software transition from research and development to operations and maintenance, suggests that it is rarely sufficient to engage only the end-user community (Rogers, 2003). Technology development teams allied with the early adopter end-users rarely have the resources or the staying power to move a new software technology from innovation to institutionalization on their own. The goal must be to design and deliver a "whole product". A whole product is defined as the new software application plus everything else needed for it to be accepted and used by the stakeholders. It is the software technology surrounded by well organized stakeholder communities interacting with each other and the software technology that produces an effective solution to the stated business mission requirements (Rogers, 2003). The essence of designing an effective "whole product" solution is that members of each stakeholder community have both a responsibility toward the new software application while at the same time gaining significant advantages from it.

To deliver the IFTDSS as a whole product, a stakeholder communications plan was written and implemented in parallel with the IFTDSS software application development (Rauscher et al., 2009). Five stakeholder communities were identified: agency senior managers (governance), scientist developers, database stewards, IT software managers, and fire and fuel operations managers (Fig. 1). Each stakeholder community is composed of members that typically fall into the following categories according to their inclination to adopt and use new technology: innovators, early adopters, early majority, and late majority (Fig. 3). In addition, each member of a stakeholder community has a varying degree of familiarity with the IFT-DSS project. These degrees of familiarity can be categorized as: awareness, understanding, trial use, and adoption.

From the software engineering perspective, the IFTDSS was developed using an agile software development process to support the stakeholder interaction described in the communications plan (Rauscher et al., 2009). The agile software development process consists of a group of software development methods based on iterative and incremental development, where requirements evolve through collaboration between end-users and other stakeholders, and the software development team. It promotes adaptive planning and interim delivery of software functionality. The main benefit of the agile process is that it provides a mechanism to collect early feedback and encourages rapid and flexible response to change.

The IFTDSS stakeholder interaction strategy began with the identification of a small number of innovators from each of the stakeholder groups (Fig. 3). The innovators then worked closely with the IFTDSS development team to design, implement, and iteratively refine the first several versions. It is crucial that an effective feedback system be established at the very beginning of the project to allow innovators to provide the new requirements for the development team to implement. The agile software development process then generates relatively small but rapidly produced changes to the software application that the innovators can critique again. Such rapid response times keep the innovators engaged and force the developers to break difficult problems into smaller, more manageable pieces.

The next stakeholder segment, the early adopters, were then engaged once sufficient functionality had been put into place (Fig. 3). The same iterative process occurred with this now substantially larger group of test users generating feedback and guiding the development of both the user interface as well as the functional improvements of the software application. Once a reasonably good version, with some important functionality, had been produced, it became possible to engage members of the senior manager stakeholder community to begin to design how the developing research application would be transitioned into institutionally supported operations and maintenance. Once this transition has been accomplished, the early and late majority members of the stakeholder communities can be engaged (Rauscher et al., 2009).

Implementing the Solution: IFTDSS

The Interagency Fuels Treatment Decision Support System (IFTDSS) is an existing service integration platform, currently with over 100,000 lines of software code, that provides command and control for software modules and datasets executing from within a common user interface (Fig. 4). It provides capabilities for use and integration of standardized and custom datasets, supports treatment unit- and landscape-scale analyses, data visualization functionality, estimates of fire behavior and first-order fire effects, and quantitative hazard and risk assessments. It allows users to choose pre-designed solution pathways for the most commonly performed fuels treatment tasks. These pre-designed solution pathways, called workflows, were designed and reviewed by members of the user stakeholder community to ensure that offered functionality matched real needs. It is absolutely critical to understand that IFTDSS is not another new fuels treatment system. It is a service integration framework that organizes and makes available a large number of pre-existing software modules. IFTDSS is a web-based application that provides users with a single user interface to multiple software tools.

- **Fire and Fuel Operations Managers**: IFTDSS transforms a chaotic, ungovernable set of stand-alone, stove-piped software applications into a consolidated, manageable single software application focused on helping users solve their mission critical business needs. The primary orientation is to support local, project-scale analyses but as part of a landscape area of interest that could be up to 2 million acres in size. IFTDSS takes the model processing power of many different systems and brings them together into one place. The IFTDSS process is easy to understand and use. Users no longer need to learn and to use multiple tools with different interfaces thus reducing training and re-familiarization time. Users no longer need to spend most of their time on data transformation issues from one software system to another. Users can use the collaboration features of IFTDSS to share data and project analyses with other professionals. Finally, for the first time, users of IFTDSS have access to a credible, yet easy to learn and apply risk assessment process that they can perform themselves at their local office.
- **Scientist Developers**: Developers can reduce development costs by focusing only on the core scientific model functions because IFTDSS will handle the user interface, user support, integration across models, data transformations, web-based hosting of application, IT security, and CPIC. This reduction in costs will better match the actual

funding typically available to developers. A single library of software service modules will make version control and updating of existing modules much easier and cheaper. Developers can focus on improving the science contained in their software modules and leave issues of life cycle software management to the IFTDSS platform management team. Developers can focus on difficult problems with narrow scope and use the IFTDSS framework to provide real-world problem solving context for the results. IFTDSS offers the potential for ensembling of models so that their outputs can be compared easily for the same data inputs. IFTDSS makes available a set of standard coding practices to make it easier for developers to produce new modules to function as IFTDSS services.

- **Database Stewards**: IFTDSS provides database stewards a way to make authoritative databases of record available to fuels treatment managers. Feedback loops between users and database stewards can be established to the governance stakeholders to identify missing or poor data problems for fuels treatment analyses. Database stewards can provide guidance to IFTDSS users on how best to overcome known problems with data availability, quality, and coverage.
- **IT Software Managers**: IFTDSS can insure that modern software design processes are used leading to control of efficiency of operations and maintenance and ensuring security requirements are maintained. IFTDSS can help IT specialists work out practical standards and guidelines for moving a research-grade software module into an operational-grade module.
- **Agency Senior Managers**: IFTDSS can function as a centralized control point for senior leadership to ensure field users are using the best available science and the best software modules expressing that science to accomplish their mission critical tasks. User feedback on problems and new functionality can guide senior leadership in making informed funding decisions on improving currently available software and creating badly needed new software applications. By controlling money spent on software development in fuels management, senior leadership can match annual expenditure requests to annual funding availability. As new mission critical needs arise, such as risk assessment for fuels treatment, senior leadership can ensure that the necessary software support becomes available in a timely manner and in the quality needed.

Implementing the Stakeholder Interaction Strategy

The objective of the first step in the IFTDSS communications strategy then was to identify a small group of innovators in each stakeholder community, increase their level of familiarity with IFTDSS gradually from awareness to adoption and, with their guidance, ensure that their mission critical business needs were supported. A 10 member JFSP Fuels Treatment Working Group consisting of fuel planners, database stewards and model developers working in fire and fuels was assembled and given the task of creating a conceptual design for the IFTDSS service integration framework (JFSP, 2008). This working group identified the functionality that IFTDSS had to contain in order to properly support the mission critical needs of the fuels treatment field managers. The NWCG Fuels Management Committee, consisting of senior fuels managers from each of the federal land management agencies, were the innovators of the senior leadership/governance community.

With the conceptual design completed, the Fuels Treatment Working Group and members of the software engineering company, Sonoma Technology Inc., Petaluma, CA, designed and wrote the first software architecture design document for IFTDSS (Funk et al., 2009). At the same time, a 9 member JFSP Software Tools and Systems Study Advisory Committee was formed, consisting of senior managers and IT software managers, to provide guidance for the project to ensure that agency institutional needs such as software security, agency sponsorship, contracting supervision, and senior management support were properly identified and planned for.

Once the first few versions of the IFTDSS software application were completed under the guidance of the innovators, the next stakeholder group, the early adopters, were engaged (Fig. 5). A 45 member Test User Group representing the early adopters in the fuel planners community was established. A proof of concept version of IFTDSS was developed (http://www.frames.gov/partner-sites/iftdss/phase-iii/) and used to allow the Test User Group to guide the evolving IFTDSS service integration framework through feedback from hands-on workshops, webinars, and one-on-one interviews (Drury, 2009). By 2012, the early adopter Test User Group was expanded to about 250 members as IFTDSS became increasingly mature and stable with a growing degree of functionality.

The model developers of 13 stand-alone software applications, that eventually were integrated into IFTDSS, represented the early adopters of the model developer community (Table 3). The model developers helped the IFTDSS project to design and test how existing stand-alone software applications can be converted to function as modules within the IFTDSS service oriented framework. A developers guide resulted from this work providing detailed standards and guidelines to help developers build future science software modules that better integrate with IFTDSS (Rauscher et al., 2012).

Senior leadership of the NWCG, consisting of the fire directors of each of the federal land managing agencies, began their own efforts to reorganize the national wildland fire enterprise, which included the “software chaos” problem. The NWCG Planning Committee produced the NWCG National Wildland Fire Enterprise Architecture (NWFEA) Modernization Blueprint (NWFEA Project Team, 2008). The NWFEA modernization blueprint provided a high-level, interagency, comprehensive and strategic vision on how to organize the interagency fire and fuels business mission. The IFTDSS project team and the NWFEA project team collaborated to make sure that the IFTDSS application was designed to be an appropriate specific example of one aspect, the software technology application, of the NWFEA vision. The NWCG Wildland Fire Investment Review Board, representing the interagency governance stakeholder community, gave provisional acceptance of the IFTDSS service integration framework in August 2009 with full acceptance pending the completion of the full prototype. The Forest Service became the Interim Managing Partner for the IFTDSS integration framework in November 2009 and provided an IT specialist to work with the IFTDSS development team. In June of 2011, the DOI Office of Wildland Fire became the Managing Partner for the IFTDSS project.

An interagency analysis and review of the current state of information technology investments, governance, and capabilities in the wildland fire programs of the four Interior fire bureaus and the USDA Forest Service was conducted in order to provide a sound information base for designing a plan to implement the broad strategic recommendations of the NWCG NWFEA Modernization Blueprint (Douglas and Phipps, July 15, 2011). This review provided another independent confirmation that the "software chaos" problem identified earlier by the JFSP existed and that it was indeed a serious and growing concern to senior leaders of the Forest Service and the DOI. The interagency action plan entitled "Wildland Fire Information and Technology: strategy, governance, and investments" appeared a year later (Douglas and Phipps, March 23, 2012). The recommended solution strategy of the Wildland Fire Information and Technology (WFI&T) strategy closely mirrors the solution strategy of the IFTDSS project: web-based SOA approach as the enabling software technology; computer platform independence; available to users regardless of location or agency; integrated data environment; software modules linked in a framework as services; sharing of work between users; mission requirements drive the application; services organized to reduce workflow complexity to users; and research and innovation focused on enhancing business mission accomplishment.

Implementing the Service Integration Framework Architecture

The IFTDSS service integration framework was created using Service Oriented Architecture (SOA) principles and methodologies for designing and developing software in the form of interoperable services. SOA-based frameworks provide a generic software architecture designed to support a collection of services such as databases and software models that are typically modular and can be reused for other applications. SOA has well-defined software and data interfaces, facilitates the integration of new and legacy software applications, and facilitates inter-operability with other systems (Haste et al., 2012).

The IFTDSS service integration framework architecture consists of three major components: (1) a web-based user interface (2) the scientific modeling framework (SMF) and (3) the data and scientific models that provide the services (Fig. 6 & 7). The IFTDSS was created to be platform independent and is almost entirely built using open-source software solutions. The SMF and the SMF Web-Application are written in Java and JavaScript; the Data Storage server is based on PostgreSQL, and the geo-spatial engine was built using Web Mapping Services and Open Layers.

The first component, the IFTDSS Web-Application (user interface), provides the user experience and includes online help and documentation; model selection, connection, and input; spatial data visualization and editing; and collaborative features. The IFTDSS Web-Application is written in Java and utilizes JavaScript.

The second component, the Scientific Modeling Framework (SMF), includes the SMF Core which manages data flow and communication throughout the system; the Executive which is a registry for locating SMF service hosts and models; the Data Storage server manages and stores multidimensional scientific data; the Aquisitor provides a mechanism to import or upload data from external sources; and one or more Model Hosts, which manage the execution of models. The SMF Web UI library is an optional component of the SMF that provides a set of SMF-aware user interface components for use in web applications. The SMF Web UI's interface components and user-triggerable actions interact with SMF elements such as data sets and models, while leaving the application with complete control over layout, data access, and model execution. The SMF is written almost entirely in Java and the SMF Data Storage server utilizes PosgreSQL.

The third component of the architecture defines how existing and new scientific models can be linked to IFTDSS (Haste et al. 2012; Rauscher et al. 2012). Models can be integrated into the IFTDSS by one of three methods: (1) direct integration into the system as a model subclass; (2) indirect integration by wrapping the model program using a custom interface, or model wrapper; or (3) through a web-service connection. While the direct integration method is the most efficient and provides the best control over process-level science, the other two methods provide needed support for legacy models and system interoperability capabilities.

Implementing the IFTDSS v. 2.0 Functionality

The IFTDSS version 2.0 service integration framework web-application can be accessed by anyone (http://iftdss.sonomatech.com/). This internet link brings users to the IFTDSS home page. From the home page, users can (1) sign in if they already have an account (2) request an account for a new user (3) view an introductory video of how to use IFTDSS and (4) view the detailed online help for IFTDSS. On the home page, viewers will also find a succinct explanation of what IFTDSS is and information on the latest functional capabilities.

The fuels treatment functions available in IFTDSS are grouped and are accessible in three ways (Haste et al., 2012): (1) by field user-designed workflow, (2) by scientist developer-designed workflows, and (3) by individual models and their exposed functions. IFTDSS field-user designed workflows are a set of business-oriented modeling pathways intended to capture the problem-solving needs of the fuels treatment analysis and planning community. They provide access to scientific models in a stepwise, intuitive pattern, reducing the emphasis of individual models. These workflows were developed based on direct user input from JFSP-sponsored fuels treatment working group and other test user groups.

The modeling tools are also grouped by scientist model developer design; that is, the tools are organized by the science teams that developed the models, and includes the model type and the outputs produced. The third way to access models within IFTDSS is to view a listing of all models and the functionality they provide.

Field User Designed Workflows: Five workflows have been identified and implemented in IFTDSS Version 2.0. Each workflow provides a logical, step-by-step process of using the various tools needed to perform the tasks of that workflow (Fig. 8).

1. The **Data Acquisition and Editing Workflow** is used to identify the appropriate vegetation, geophysical, and weather data for IFTDSS that will be needed for a project. IFTDSS goes to authoritative data sets, such as LANDFIRE, and automatically downloads the requested data coverage. Users may then view and edit the data acquired in order to customize it for a project analysis. The customized data set(s) may then be saved, output in selected file formats to a local computer, and/or shared with other IFTDSS users. Data acquisition and editing was identified as such a fundamental and important workflow that we highlighted it in the IFTDSS user interface as a separate, immediately visible Tab in the user interface.
2. The **Hazard Analysis Workflow** is used to identify potentially hazardous areas across a landscape. The focus of this workflow is to identify areas across a landscape where fuels treatment analysis may be warranted based on potential fire hazard. IFTDSS provides tools that support this workflow.
3. The **Risk Assessment Workflow** provides a first-approximation probabilistic risk assessment for fuels treatment planning.
4. The **Fuels Treatment Workflow** (a) simulates fuels treatment placement in areas of high fire hazard within an area of interest, (b) simulates post-treatment influences on fire behavior and fire effects potentials, and (c) evaluates the temporal durability of fuels treatments; that is, how long, in years to decades, a treatment will continue to reduce adverse fire behavior and fire effects within an area of interest.
5. The **Prescribed Burn Planning Workflow** provides the information needed to plan and document a proposed prescribed fire. IFTDSS provides tools that support this workflow; with these tools, users can
   - calculate the probability of ignition from lightning or a firebrand
   - assess and calculate fire behavior
   - assess and plan fire containment
   - calculate fire effects
   - create a prescribed burn plan (including printing out a Word document with many of the required burn plan elements filled in automatically by IFTDSS)

A detailed discussion and explanation for each workflow in IFTDSS v. 2.0 can be found in the Phase IV final report section 5 (Haste et al, 2012) and in the online Help system accessible on the home page of IFTDSS **(**http://iftdss.sonomatech.com/).

Scientist Developer Designed Workflows: IFTDSS also supports the organization of tools by scientist developer-designed workflows. The purpose of developer-designed workflows is to provide a way to use software models in ways other than the field manager designed workflow pathways. For example, a research scientist might be interested in simply evaluating the output of two models within IFTDSS given the same input values. These developer-designed workflows can be a single calculation or a series of calculations implemented in the developer's original tool set or application. Currently, only one developer's tool set has been incorporated into IFTDSS version 2.0 as a demonstration of what is possible: the Fire and Environmental Research Applications (FERA) Team's Fire and Fuels Application (FFA) (http://www.fs.fed.us/pnw/fera/). FERA is a USDA Forest Service research team focusing on fuels and fire and landscape ecology. IFTDSS supports several FERA tools:

- **Consume**. Predicts fuel consumption, pollutant emissions, and heat release based on a number of factors, including fuel loadings, fuel moisture, and other environmental factors.
- **Fuel Characteristic Classification System (FCCS)**. Stores and classifies fuels data as fuelbeds, calculates physical characteristics of fuels based on fuelbed data, and calculates fire potentials based on the intrinsic properties of fuels.
- **Fire Emission Production Simulator (FEPS)**. Predicts fuel consumption, emission rates, and heat release characteristics of prescribed burns and wildland fires. Total burn consumption values are distributed over the life of the burn to generate hourly emission and release information.
- **Digital Photo Series (DPS)**. Users can link to the DPS from within IFTDSS to obtain fuel loading information for a selected situation to replace default values.

Direct Access to Scientific Models: IFTDSS provides direct access to a range of tools through standalone user interfaces to assist users who wish to perform calculations with a single tool instead of going through an entire workflow process. There are 71 individual calculations that can be performed in IFTDSS Version 2.0 (Table 4). There are 13 software application modules (Table 3) that have been integrated into IFTDSS to provide the 71 model calculation services listed in Table 4.

The System of Systems Vision

While the IFTDSS service integration framework is specifically designed for fuels treatment planning, the generic service integration framework approach serves as an example and stepping stone toward a larger "system-of-systems" vision. The vision is that the fire management community will access modeling, analysis, reporting needs through a small number of interoperable service integration frameworks defined and organized by fire and fuels management business needs (Reinhardt and Dickinson, 2010: 138). For example, the BlueSky Framework (BlueSky) and Wildland Fire Decision Support System (WFDSS) are also service integration frameworks, each serving a different business need within the fire and fuels domain (Fig. 9). Each of these larger software integration frameworks would eventually access a common virtual library of component computational models. One of the major goals for the IFTDSS project is to demonstrate the feasibility and value of the system-of-systems concept.

A “system-of-systems” architectural approach is characterized by an independence of operations and management, meaning that the individual constituents of the system are able to act with relative independence. Effective management in this construct requires greater emphasis on strategic elements (e.g. core enabling services, data and interface standards, and coalition governance). These elements should encourage conformance while placing minimum constraints on the mission solution space - empowering technologists and operational elements to quickly incorporate and adapt preferred capabilities that meet diverse mission needs. In a decentralized environment, a system-of-systems architectural approach can yield greater agility, enhanced situational awareness and mission effectiveness, improved security posture, and reduced development and lifecycle costs.

A system-of-systems approach explicitly recognizes and enables independent, evolutionary development of constituent capabilities. This co-evolution is necessary given the wide range of continuously improving capabilities associated with wildland fire. Independence of change in individual constituents adds to the complexity of the interactions among constituents and of management and operations. Thus, in a system-of-systems, evolution must be explicitly recognized and managed. By facilitating change, systems can more readily integrate innovation resulting in greater mission impact (Bennett et al., 2013).

Evaluation Results

For the IFTDSS Evaluation Study, JFSP engaged the Software Engineering Institute (SEI), a Federally Funded Research and Development Center (FFRDC) operated by Carnegie Mellon University. The SEI was familiar with the issue of “software chaos” having conducted Phase I of the STS, but had not been involved in the subsequent phases.

For the evaluation, JFSP tasked the SEI with these four questions:

- *Does IFTDSS provide a significantly improved platform for the integration of computational models and data?*
- *Can IFTDSS improve the efficiency and effectiveness of model and software development and maintenance process as compared to existing procedures?*
- *Does IFTDSS measurably improve the quality and efficiency of fuels treatment planning?*
- *Does IFTDSS encourage and improve critical thinking and problem solving skills needed for fuels treatment planning?*

The SEI began their evaluation in October 2012 and delivered their final report on July 1st, 2013 (Bennett et al., 2013). They conducted the evaluation using document reviews, tool and system demonstrations, and individual/group interviews. The SEI also used a variant of the Architecture Tradeoff Analysis Method® (ATAM®). The ATAM uses scenarios to operationalize architectural drivers, after which a team of software architects and engineers as well as user representatives assess how well the architecture will perform in the scenarios.

Additionally, the SEI hosted a series of 11 day-long user workshops; these were held virtually as well as at seven physical locations. The SEI was supported in these workshops by the IFTDSS development contractor (Sonoma Technologies, Inc.) and by Kim Ernstrom of the Wildland Fire Management Research, Development, and Application (RD&A).

Overall, the SEI stated JFSP succeeded with its goals for IFTDSS. They concluded that if IFTDSS were fielded with a cohesive, inter-agency governance strategy, it would be a major step to bringing order to the software chaos problem (Bennett et al., 2013).

According to the SEI, IFTDSS:

- enables standardized, risk-based fuels management planning for a large part of the community
- demonstrates a consistent, common set of analysis processes that use common tools, and makes these available regardless of agency or location
- demonstrates a prescribed burn plan workflow that is designed around the *2008 Guide* and was also significantly influenced by users
- demonstrates a workflow for risk assessments that is accessible and useful for a majority of users

IFTDSS also demonstrates a vision of the WFI&T's principal concepts:

- mission requirements drive integrated, modular-based applications and tools
- authoritative data are readily available for all uses and users
- interconnection and accessibility regardless of organizational affiliation or user location
- technology, research, and innovation enable and enhance mission accomplishment

The SEI also stated that IFTDSS:

- contains a rich set of tutorial material that goes beyond mechanics into a knowledge management system that could serve as a framework for fuels management self-study and training support
- demonstrates a single-sign on for a host of currently stand-alone systems
- facilitates data entry and formatting
- allows users to create landscapes that most reflects their current, local knowledge, and then to run multiple tools across that landscape without the need for data reformatting
- can be used to share both the data and the analysis
- uses workflow-based navigation to demonstrate end-to-end guidance and processes that could be implemented when these processes are adopted across the wildland fire community
- provides the ability for agency-specific needs via tailored workflows
- provides spatial constructs for some non-spatial foundational models that allow a more-complete visualization of fire behavior

- provides a consistent user interface and data formatting approach

The SEI summarized their findings in Fig. 10.

However, the SEI was clear that they saw some risks for IFTDSS:

- IFTDSS was developed without significant investment in security. If the Scientific Modeling Framework (SMF) within IFTDSS is to be exposed as a service to other applications then identity and data confidentiality will need to be implemented in both SMF and the model hosts.
- IFTDSS was developed without defined performance standards. While performing "well enough" (the SEI only used this subjective term because there were no hard requirements), users occasionally complained of unexpected delays. While IFTDSS is structured such that additional hardware can be added to support additional users, the SMF model scheduler will need a monitoring capability of current loads and resource requirements to perform load balancing.
- Extending IFTDSS by adding new models will burden the governance process, in particular to qualify new models for deployment as well as stop current models from being deployed. Extending IFTDSS into new domains will require some refactoring of the architecture because the separation between the IFTDSS web application and the SMF is not as clean as depicted.
- While IFTDSS and the SMF "data acquisitor" obtain data from other sources, IFTDSS currently has no data export capability, which limits full interoperation.

The SEI cited risks cited in other areas, but in general these were not about IFTDSS *per se*, but rather were about the processes and governance that will be needed for IFTDSS (or any system) to be effective. These included:

- There is no agreed-upon, consistent, end-to-end fuels treatment planning process in use across the federal wildland fire community.
- There are no standardized policies and mechanisms to transition research projects into operational production; many key operational applications are supported out of research environments.
- There is no consistent software lifecycle management model.
- There is no coherent software portfolio management process to guide new investments or evaluate the efficacy of current fuels management applications.
- There is no curriculum of required training for fuels management specialists.
- There is no consistent, community-wide hosting strategy.

The SEI recommended that the federal wildland fire community fund IFTDSS as a system of record and continue using IFTDSS in a controlled prototype effort as it is currently configured. This would allow for continued user feedback while the issues they cited are addressed.

The SEI summarized their recommendations in Fig. 11.

Discussion and Conclusion

After nearly 13 years of effort by members of the Joint Fire Science Program and five years of effort by members of the Wildland Fire Information and Technology study, there can no longer be any question of the existence, characteristics, and importance of the "software chaos" problem in the fire and fuels management business mission of the US federal land management agencies. The broad outline of a solution to the "software chaos" problem were identified by the WFI&T Strategic Plan (Douglas and Phipps, 2012). This plan articulates a vision to enable an interagency, integrated approach to wildland fire information and technology management in support of mission activities. The plan establishes four concepts that guide wildland fire technology implementation:

- Mission requirements drive integrated, modular based applications and tools.
- Authoritative data are readily available for all uses and users.
- Interconnection and accessibility regardless of organization affiliation or user location.
- Technology, research, and innovation enable and enhance mission.

An extensive, year-long, independent evaluation finds that IFTDSS (1) significantly improves the quality and efficiency of the fuels treatment planning process for the majority of user needs; (2) provides a concrete demonstration of one way to implement the four key WFI&T concepts above; (3) enables standardized, risk-based fuels management planning for a large part of the fuels specialist community; and (4) is near-ready for operational use (Bennett et al., 2013). The evaluation concluded that if fielded as part of a cohesive governance strategy, IFTDSS can be a major step in bringing order to the "software chaos" problem.

While IFTDSS represents a significant step forward from a technology perspective, resolving the "software chaos" problem requires changes to the way the stakeholder communities are organized and operate. Specifically, there are many missing elements that need design and implementation to finally resolve the long-standing "software chaos" problem (Bennett et al., 2013):

- There is no authoritative, cohesive development agenda and roadmap for fuels management software.
- There are no incentives or organizational support structures for the achievement of common architectures, integrated workflows, common user interfaces, and common data models across applications.
- There is no coherent governance and review process for model approval and workflow accreditation.
- There are no processes in place to collect and evaluate field user feedback to prioritize software function improvements along with funding support.

- Quality standards for software coding, testing, configuration management, release management, change management, and framework architecture compatibility are missing or not held to common production software level disciplines.

- There is no robust, unified hosting strategy.

- The transition between research tool and operational application is blurred and the process has not been defined.

- How to transition legacy software systems into operation applications following the developing WFI&T governance process has not been worked out.

- A unified vision of a long-term fuels management roadmap that could inform future research and model development efforts is missing.

- No agreed-upon consistent risk and/or hazard assessment framework has been developed. This will require an inter-agency agreement on a high level risk workflow that also allows for agency and region specific tailoring.

IFTDSS was designed primarily as a tool to guide and facilitate fuels management planning by front-line forest managers. To accomplish this, it "hides" certain complexities of the fuels management models which are not required to accomplish standard fuels management planning scenarios. A consistent concern raised by researchers regarding IFTDSS was that it could lead to a naïve, "black-box" approach to running and interpreting model results. Researchers believe that their current involvement in operational support activities provides some safeguards against this risk, both because they can educate users on model assumptions, and can themselves step in and conduct modeling for complex high-risk scenarios. Any organizational adoption and transition to IFTDSS must ensure adequate training on model assumptions as well as thorough explanations of the default parameters that were selected for incorporation into IFTDSS. In addition, consultation services for modeling complex high-risk scenarios will still be required. Training should be provided both on how to recognize these high-risk scenarios and the mechanisms by which to escalate them for evaluation by those with enhanced expertise. In addition, the "expert" tools should be identified and plans should be made for their support and enhancement. One potential route is that the expert tools will reside within the IFTDSS framework and will be the same ones used by front line managers. The only difference would be that experts would have access to a full and robust set of model parameters. Another potential path is that IFTDSS itself would contain only user level modeling capabilities and a different tool set would be supported and maintained for expert usage.

Acknowledgements: Many individuals have contributed to the Interagency Fuels Treatment Decision Support System (IFTDSS) software development effort. The JFSP Fuels Treatment Working Group (FTWG)—Michael Beasely, Dennis Dupuis, Mark Finney, Glen

Gibson, Randi Jandt, David Peterson, Tessa Nicolet, and Brad Reed—has guided our work and provided initial critique and innovative ideas for the project. The JFSP Software Tools and Systems Study Advisory Committee—Pat Andrews, Nate Benson, Mike Hilbruner, Mike Hutt, David Peterson, Carol Saras, Paul Schlobohm, Shari Shetler, and Tim Swedberg—has kept the study headed in the right direction and made sure that all of the various components of a successful solution were considered.

We would like to specifically acknowledge the fuels treatment specialists who have agreed to participate in this effort to serve as the IFTDSS Test User Group—Brad Reed, Brenda Wilmore, Eric Miller, Gary Curcio, Gary Fildes, Gwen Lipp, Jim Roessler, Jon Wallace, Jonathan Olsen, Karen Folger, Mack McFarland, Nikia Hernandez, Perry Grissom, Randi Jandt, Randy Stiplin, Scott Weyenberg, Sean McEldery, Tessa Nicolet, Alison Forrestel, Brian Sorbel, Daniel Rasmussen, Dennis Fiore, Dennis Page, Glenn Gibson, Jeremy Spetter, Joshua Keown, Kim Kelly, Kyle Jacobson, Loretta Duke, Mike Uebel, Sam Amato, Anna Payne, Dave Pergolski, Jennifer Croft, Jeremy Bennet, John Washington, Ken Rodgers, Mathew Weldon, Paul Maday, Rance Marquez, Albert Savage, William Aney, Yanu Gallimore, Lauren Miller, Kristen Allison—who have provided, and will continue to provide, valuable feedback regarding the functionality and usability of the IFTDSS. We also thank the broader group of 49 field fuels treatment specialists who helped develop and refine the fuels treatment decision support process.

We would like to acknowledge our appreciation of the fire and fuels science and software development community—Eric Twombly, Mark Finney, Joe Scott, Alan Ager, Nick Crookston, Larry Gangi, Woodham Chung, Kurt Kruger—for their cooperation and feedback related to integrating data and software applications that will be a part of the IFTDSS. Thank you also to the Fire and Environmental Research Applications (FERA) team of Dave Petersen, Roger Ottmar, Susan Prichard, Paige Eagle, and Kjell Swedin for their support and assistance implementing new software modules into the IFTDSS. We would also like to acknowledge the managers of the Wildland Fire Decision Support System (WFDSS)—Tom Zimmerman, Rob Seli, Mitch Burgard and their development team—and the BlueSky Framework—Sim Larkin—for their willingness to work collaboratively to develop software systems that can communicate with one another to create efficiencies in the fire and fuels domain.

We would also like to thank the IT specialists who helped ensure that we have considered agency IT requirements—Brad Harwood, Laura Hill, John Gebhardt, and John Noneman. Finally we want to thank Michelle Tae of CommonThread Resources, Boise ID for her excellent guidance in developing the IFTDSS Conceptual Design Document.

References

Ager, Alan A., Nicole M. Vaillant, and Mark A. Finney. 2011. Integrating Fire Behavior Models and Geospatial Analysis for Wildland Fire Risk Assessment and Fuel Management Planning. Journal of Combustion, Article 572452. doi:10.1155/2011/572452

Bennett, Constance M., Nanette Brown, Dan Doney, Linda Parker Gates, Suzanne Miller, M. Steven Palmquist (PE), Patrick Place. Final Report of the Interagency Fuels Treatment Decision Support System (IFTDSS) Evaluation Study. 2013. Software Engineering Institute, Carnegie Mellon University, Special Report CMU/SEI-2013-SR-017, 142 p.

Calkin D.E., Ager A.A., Gilbertson-Day J., Scott J.H., Finney M.A., Schrader-Patton C., M. Q.T., Strittholt J.R., and D. K.J. (2010) *Wildfire Risk and Hazard: Procedures for the First Approximation*, D.E. Calkin, A.A. Ager, and J. Gilbertson-Day, eds., USDA Forest Service, Rocky Mountain Research Station, Fort Collins, CO (Gen. Tech. Rep. RMRS-GTR-235). Available on the Internet at http://www.fs.fed.us/rm/pubs/rmrs_gtr235.pdf.

Douglas, Jim and John Phipps. July 15, 2011. Implementing the National Wildland Fire Architecture Blueprint. Unpublished Interagency report, http://www.frames.gov/documents/iftdss/Fire_IT_Report_Signed_7-15-11.pdf .

Douglas, Jim and John Phipps. March 23, 2012. Wildland Fire Information and Technology: strategy, governance, investments. Unpublished Interagency report http://www.frames.gov/documents/iftdss/Signed_IT_Report_March_23_2012.pdf .

Drury, Stacy A. 2009. Summary of Fuels Specialist Feedback on the IFTDSS Proof of Concept Functionality and Work Flow Scenarios Manuscript. STI-909029.02-3686. Unpublished IFTDSS project report, http://www.frames.gov/documents/ift-dss/fuels_specialist_feedback_20090901.pdf .

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

Funk, Tami H., Roger G. Corman, Judd E. Reed, Sean M. Raffuse, Neil J. M. Wheeler. 2009. The Interagency Fuels Treatment Decision Support System Software Architecture: Draft Software Architecture Design. STI-908038.04-3565. Unpublished IFTDSS project report, http://www.frames.gov/documents/ift-dss/task2_tech_architecture_draft_20090212.pdf .

Gollberg, Greg E., Leon F. Neuenschwander, and Kevin C. Ryan. 2001. Introduction: Integrating spatial technologies and ecological principles for a new age in fire management. Intl. J. Wildland Fire 10: 263 – 265.

Haste, Tami H., David J. Noha, Stacy A. Drury, Erin M. Banwell, Neil J. M. Wheeler, Michael D. Haderman. 2012. Implementation of the Interagency Fuels Treatment Decision Support System. STI-910901-5550-FR. Unpublished IFTDSS project report, http://www.frames.gov/documents/iftdss/IFTDSS_PhaseIV_Final_Report.pdf .

JFSP. December 10, 2008. Interagency Fuels Treatment Decision Support System Conceptual Design. Unpublished IFTDSS project report, http://www.frames.gov/documents/ift-dss/task2_conceptual_design_v1_0_20090121.pdf .

Larkin, N.K., Tami Funk, Judd Reed, Sean Raffuse, Lyle Chinkin, Mike Rauscher. 2009a. Application of Service Oriented Software Architectures in the Fuels Treatment Community. Sonoma Technology Inc., Petaluma, CA. Unpublished IFTDSS project report, http://www.frames.gov/documents/ift-dss/task2_service_oriented_architectures_20090106.pdf .

Larkin N.K., O'Neill S.M., Solomon R., Raffuse S., Strand T.M., Sullivan D.C., Krull C., Rorig M., Peterson J., and Ferguson S.A. (2009b) The BlueSky smoke modeling framework. Int. J. Wildland Fire, 18 (8), 906-920. doi:10.1071/WF07086

Lewis, Clayton and John Rieman. 1994. Task-Centered User Interface Design: a practical introduction. Lewis and Rieman, Boulder, CO. http://hcibib.org/tcuid/index.html .

Newcomer E. and Lomow G. (2005) Understanding SOA with web services, Addison-Wesley Professional.

NWFEA Project Team. July 1, 2008. National Wildland Fire Enterprise Architecture Blueprint. NWCG Program Management Office, Boise, ID. http://www.nwcg.gov/nwfea/dnloads/part1.pdf .

Palmquist, Steven M. 2008. Working Summary of the SEI's Engagement with the Joint Fire Science Program. Unpublished project report, http://www.frames.gov/documents/ift-dss/palmquist_sei_report_2008.pdf .

Rauscher, H. M. Feb. 16, 2009. Summary of Fire and Fuels Specialists Software Tools Survey. Unpublished IFTDSS project report, http://www.frames.gov/documents/ift-dss/task1_fuels_specialists_survey_results_20090216.pdf .

Rauscher, H. M., Tim Swedberg, Paul Schlobohm, Eric Christiansen, Tami Haste. 2009. Communications Strategy for Community Development Surrounding the Interagency Fuels Treatment Decision Support System (IFTDSS). Unpublished IFTDSS project report, http://www.frames.gov/documents/ift-dss/task7_comm_plan_20090722.pdf .

Rauscher, H. M., Tami Haste, Stacy Drury, John Stilley, and Jennifer DeWinter. 2012. Developer's Guide for Integrating Models into the Interagency Fuels Treatment Decision Support System (FTDSS). Unpublished IFTDSS project report, http://www.frames.gov/documents/iftdss/IFTDSS_Developers_Guide_FINAL.pdf .

Reinhardt, Elizabeth D. and Matthew B. Dickinson. 2010. First-Order Fire Effects Models for Land Management: Overview and Issues. Fire Ecology 6(1): 131-142. doi: 10.4996/fireecology.0601131

Rogers, Everett. 2003. Diffusion of Innovations. Free Press, 5th edition.

Wells, G. 2009. A powerful new planning environment for fuels managers: the Interagency Fuels Treatment Decision Support System. Fire Science Digest 7. Joint Fire Science Program, Boise, Idaho. http://www.firescience.gov/Digest/FSdigest7.pdf .

List of Figures

Figure 1. Stakeholder Communities

Figure 2. Existing Condition Description

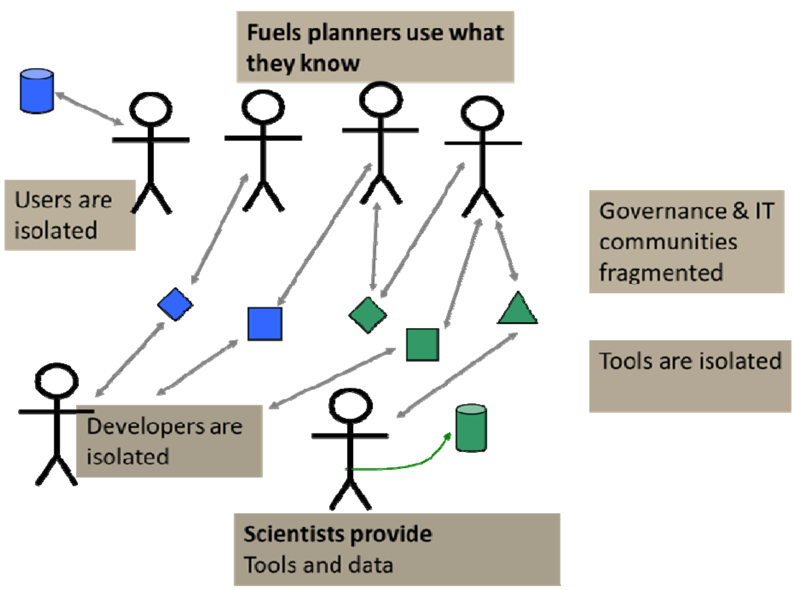


Fig. 3. The Diffusion of IFTDSS Awareness and Use in a Stakeholder Community

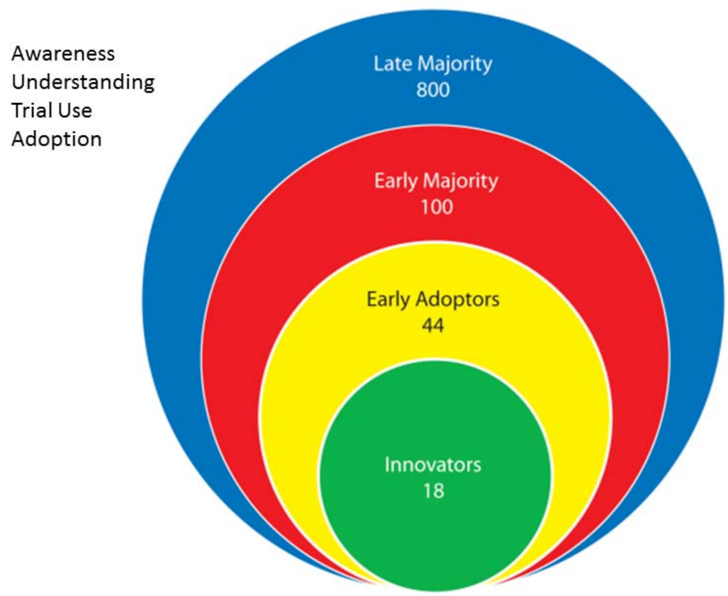


Figure 4. The IFTDSS vision

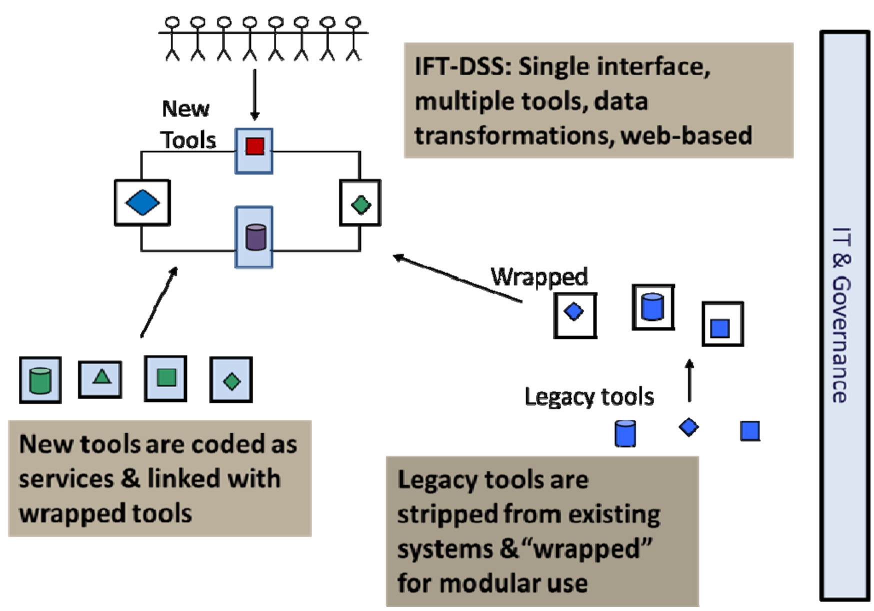


Fig. 5. IFTDSS Test Users

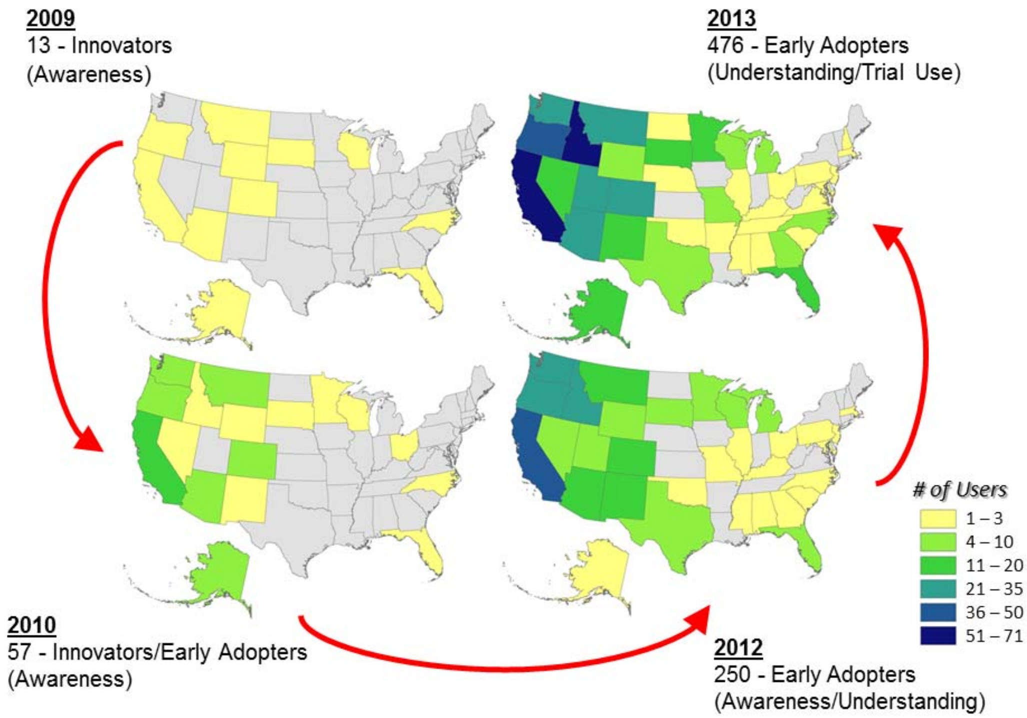


Fig. 6. The three main components of the IFTDSS software integration framework.

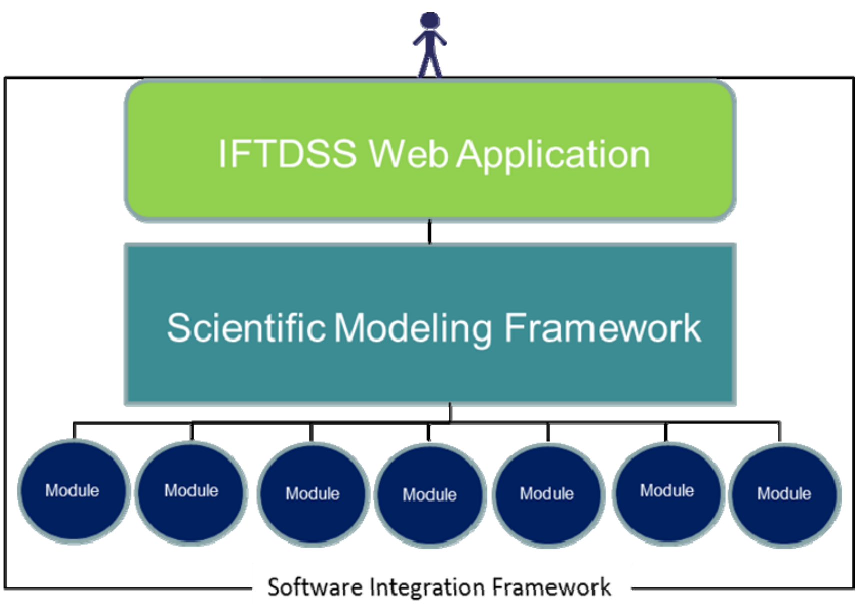
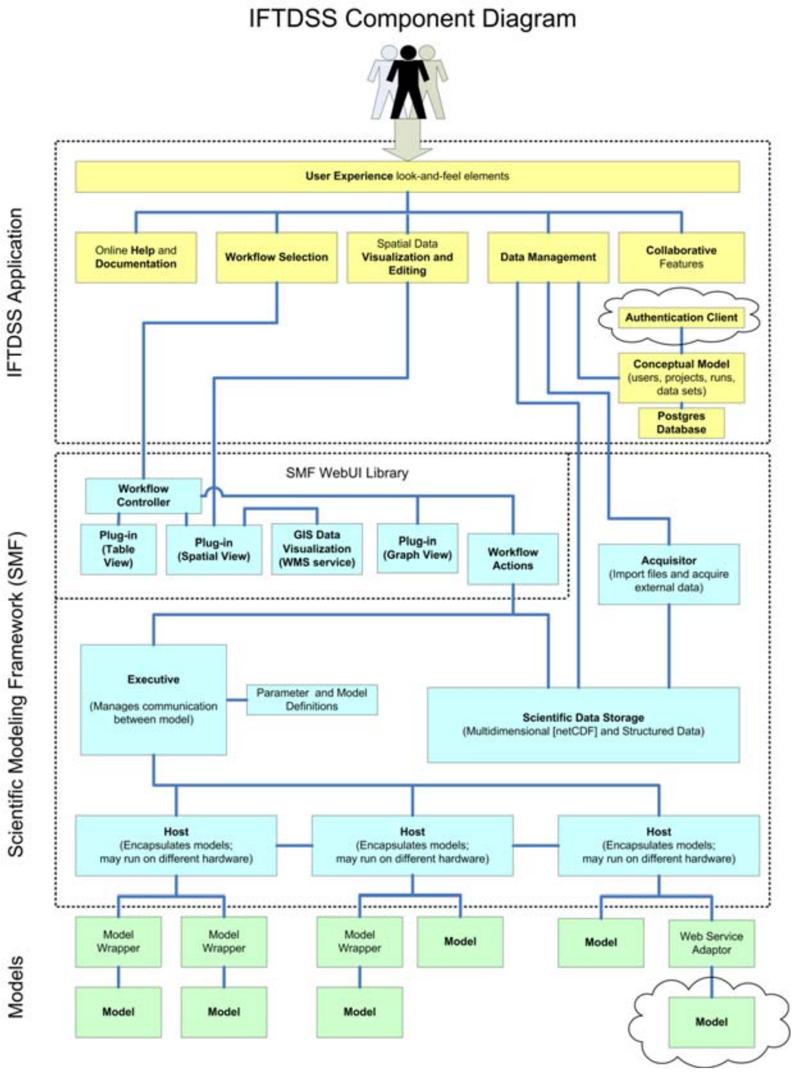


Fig. 7. Architectural view of the IFTDSS software application



IFTDSS Component Diagram
IFTDSS Application
User Experience look-and-feel elements
Online Help and Documentation
Workflow Selection
Spatial Data Visualization and Editing
Data Management
Collaborative Features
Authentication Client
Conceptual Model (users, projects, runs, data sets)
Postgres Database
Scientific Modeling Framework (SMF)
SMF WebUI Library
Workflow Controller
Plug-in (Table View)
Plug-in (Spatial View)
GIS Data Visualization (WMS service)
Plug-in (Graph View)
Workflow Actions
Acquisitor (Import files and acquire external data)
Executive (Manages communication between model)
Parameter and Model Definitions
Scientific Data Storage (Multidimensional [netCDF] and Structured Data)
Host (Encapsulates models; may run on different hardware)
Host (Encapsulates models; may run on different hardware)
Host (Encapsulates models; may run on different hardware)
Models
Model Wrapper
Model Wrapper
Model Wrapper
Model
Model
Web Service Adaptor
Model
Model
Model
Model

Fig. 8. High-level description of all IFTDSS workflows

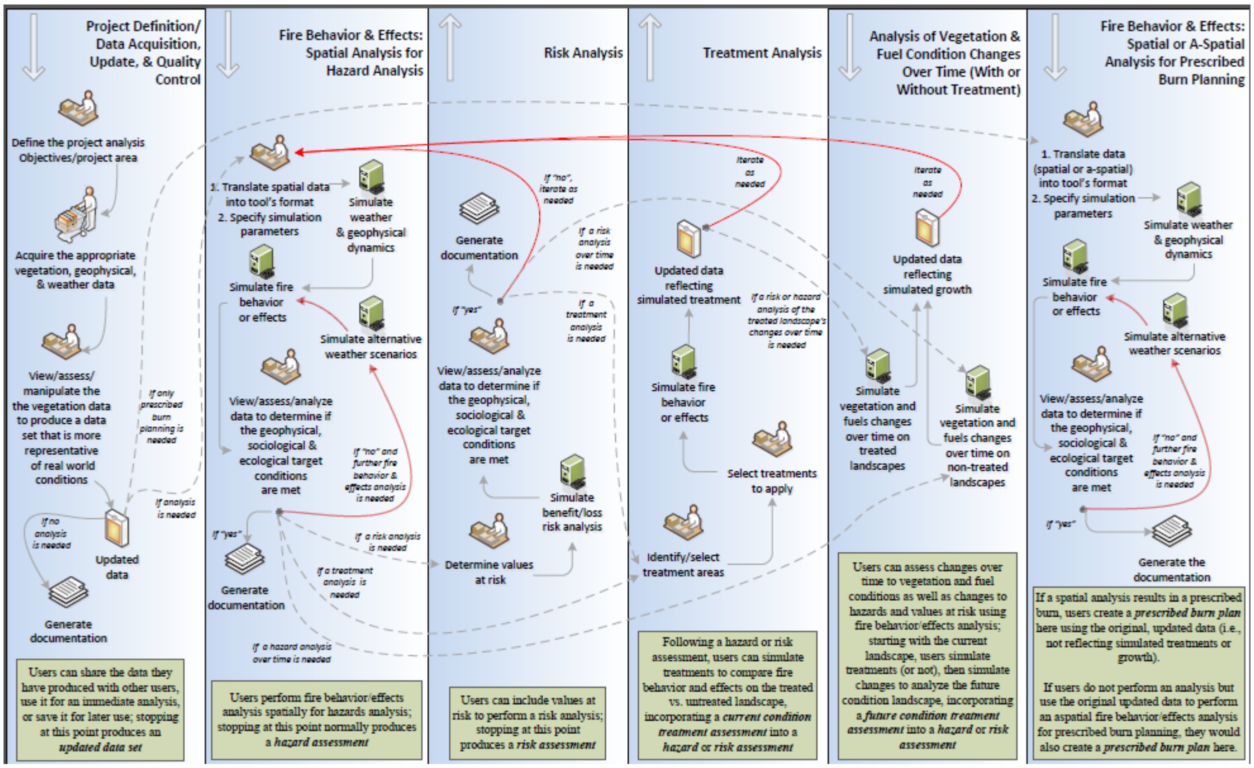


Fig. 9. The System of Systems Vision

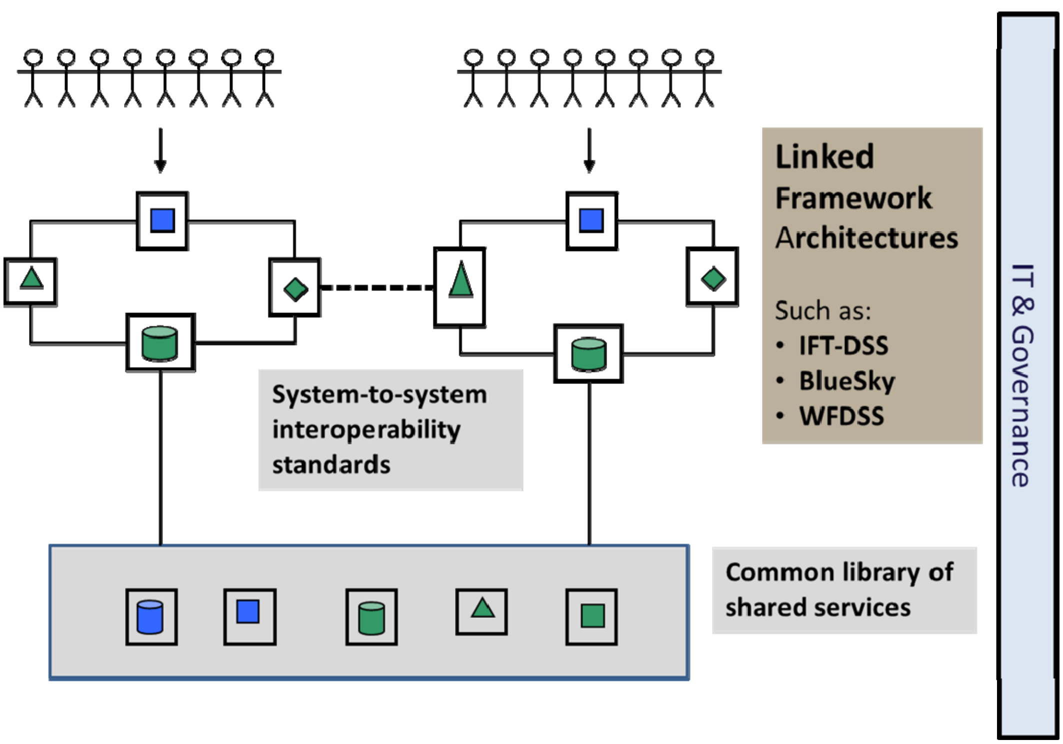


Figure 10: Summary of SEI Evaluation Findings

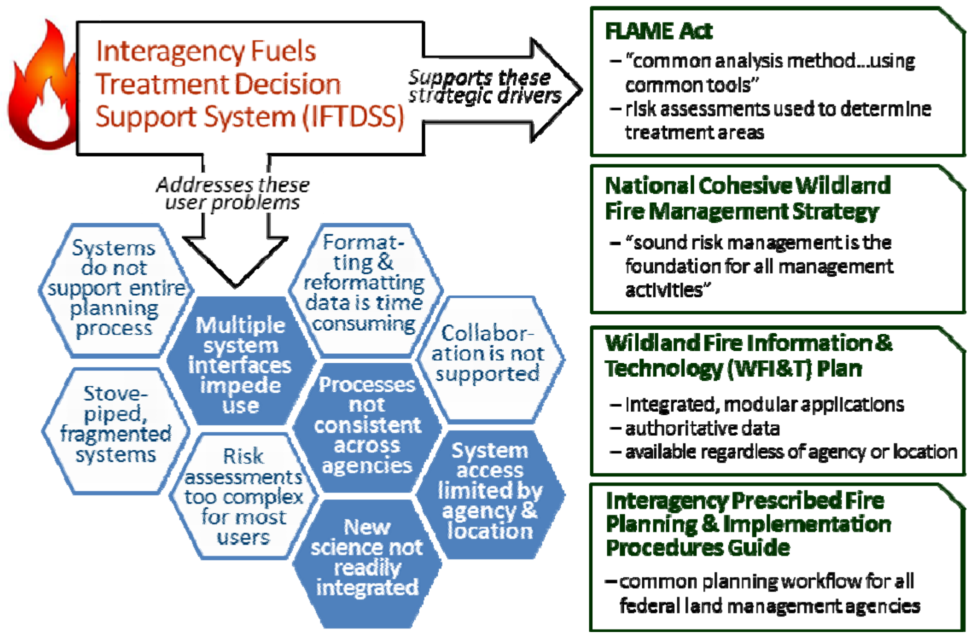
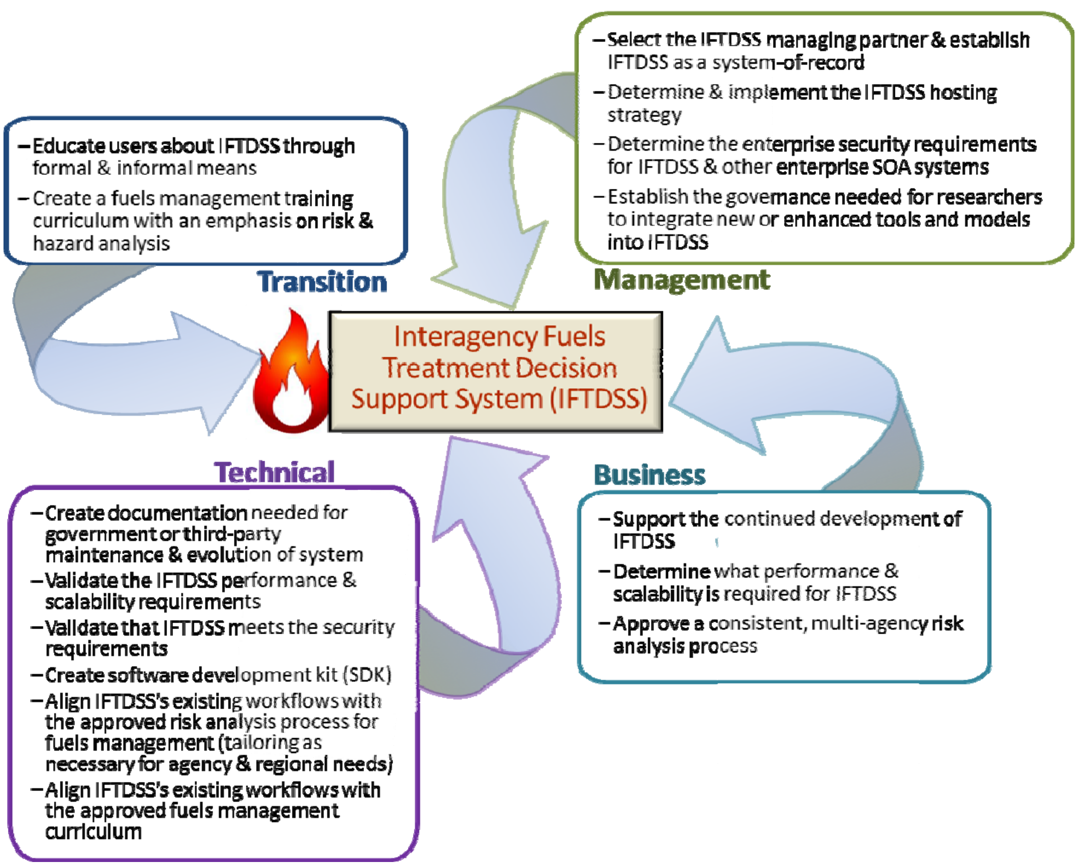


Figure 11: Summary of SEI Evaluation Recommendations



List of Tables

Table 1. Terminology

Term	Definition
Software Integration Framework	A framework that integrates scientific models, built for different purposes, into a larger structure capable of addressing a broader question or decision support need. Software integration frameworks may or may not have a graphical user interface (GUI) and are typically platform independent. Software integration frameworks require model or module packages.
Software Application	An application that integrates scientific models, built for a specific purpose, to address a specific and somewhat narrowly defined question. Software applications typically have a GUI that is tightly coupled with its underlying software models and are typically platform-specific. Many applications utilize models or dynamic link libraries (.dll).
Model	The source code, or model calculation software that performs a specific mathematical algorithm. Models are command-line driven and have no GUI.
Module	A collection or grouping of mathematical models.
Model or Module Package	A package of software code containing model calculation software, metadata, and some kind of model implementation (e.g., wrapper) to allow one model (model package) or a collection of models (module package) to communicate with a larger software integration framework.
Scientific Modeling Framework (SMF)	The underlying modeling framework used by the IFTDSS application.
SMF-Modeling Binary Interface (SMF-MBI)	The application binary interface that describes the low-level software interface between a model and the SMF. The SMF-MBI defines how a model can communicate with the SMF.

Table 2. Most frequently used software tools of all types in 2009 by fire and fuels operations managers in the federal land management agencies.

Software Tool	Times Mentioned in the Survey	Percent of Respondents[1]	Data or Model Integrated in IFTDSS	Functionality Implemented or Connection to Other System
Behave (All Variants)	39	89%	x	
FOFEM	19	43%	x	
FireFamily Plus	16	36%		x[2]
ArcGIS	16	36%		x
FARSITE	15	34%		
FlamMap	13	30%	x	
LANDFIRE Data	10	23%	x	
FMA +	10	23%		
FVS	10	23%		
FFI/Firemon	7	16%		
Google Earth	6	14%		x
Nexus	6	14%		x
RERAP	6	14%		
CONSUME	5	11%	x	
FRCC	5	11%		
WIMS Data	5	11%		
SIS	4	9%		
LANDFIRE Tools	2	5%	x	
PROBACRE	2	5%		
SASEM	2	5%		
VSmoke GIS	2	5%		
NFDRS Calculator	1	2%		x[2]
Wind Ninja	1	2%	x[2]	
FCCS	1	2%	x	
FlamMap MTT	1	2%	x	
Other Tools Mentioned Once[3]	1	2%		

[1] Number of survey respondents: 44

[2] Functionality partially integrated in IFTDSS

[3] Twenty tools not listed in the table were each mentioned once:
BlueSky, Compare 4 Fuel Model Spreadsheet, FACTS Reporting System, FlamMap TOM, FSPRO, KCFAST, Map Tech Terrain Navigator Pro, Microsoft Digital Image Suite, NFPORS, NRIS Reporting System, Parcel Quest, Rainbow Series, RAMS, Simple Graphical Smoke Screening System, Spatial Data Analyzer (formerly INFORMS), Startfire, SVS, Topo 4.0, VDDT, Wind Wizard

Table 3. Descriptions of models that IFTDSS calculations are based on.

Model	Description
BehavePlus	The BehavePlus fire modeling system is a collection of models that describe fire behavior, fire effects, and the fire environment.
Consume	Consume 3.0 is designed to import data directly from the Fuel Characteristic Classification System (FCCS), and the output is formatted to feed other models and provide usable outputs for burn plan preparation and smoke management requirements. Additionally, training and a user's manual are available. Consume can be used for most forest, shrub, and grasslands in North America. http://www.fs.fed.us/pnw/fera/research/smoke/consume/index.shtml
Model	Description
FCCS	The Fuel Characteristic Classification System (FCCS) offers consistently organized fuels data along with numerical inputs to fire behavior, fire effects, and dynamic vegetation models. http://www.fs.fed.us/pnw/fera/fccs/index.shtml
FEPS	The Fire Emission Production Simulator (FEPS) manages data concerning consumption, emissions, and heat release characteristics of prescribed burns and wildland fires. http://www.fs.fed.us/pnw/fera/feps/index.shtml
FireFamilyPlus	FireFamilyPlus analyzes and summarizes an integrated database of fire weather and fire occurrence. It combines the functionality of the programs PCFIRDAT, PCSEASON, FIRES, and CLIMATOLOGY. FFP can be used to calculate fire danger rating indices and components and to summarize both fire and weather data. It offers options for jointly analyzing fire and weather data. http://www.firemodels.org/index.php/firefamilyplus-introduction/firefamilyplus-overview
FlamMap	FlamMap is a fire behavior mapping and analysis program that computes potential fire behavior characteristics (such as spread rate, flame length, and fireline intensity) over an entire landscape for constant weather and fuel moisture conditions. http://www.firemodels.org/index.php/national-systems/flammap
FOFEM	FOFEM (a First Order Fire Effects Model) is a computer program for predicting tree mortality, fuel consumption, smoke production, and soil heating caused by prescribed fire or wildfire. FOFEM provides quantitative fire effects information for tree mortality, fuel consumption, mineral soil exposure, smoke, and soil heating. http://www.firelab.org/science-applications/fire-fuel/111-fofem
FVS	The Forest Vegetation Simulator (FVS) is a family of forest growth simulation models. FVS answers questions about how forest vegetation will change in response to natural succession, disturbances, and proposed management actions. http://www.fs.fed.us/fmsc/fvs/
MTT	FlamMap's Minimum Travel Time (MTT) is a two-dimensional fire growth model that calculates fire growth and behavior by searching for the set of pathways with minimum spread times from a point, line, or polygon ignition source, keeping environmental (fuel moistures and winds) conditions constant for the duration of the simulation http://www.wildfirelessons.net/uploads/fire_behave_factsheet.pdf
OptFuels	OptFuels integrates existing fire behavior (FlamMap), vegetation simulation (FVS-FFE), and land management planning (MAGIS) tools into one decision support system that supports long-term fuel management decisions. http://www.fs.fed.us/rm/human-dimensions/optfuels/main.php
RANDIG	RANDIG simulates fire spread using the minimum travel time methods and inputs on wind, fuel moisture, and topography.

SVS	The Stand Visualization System (SVS) is a post-processing program for FVS. http://www.fs.fed.us/fmsc/fvs/software/index.shtml
VSmoke	VSmoke is a model that estimates downwind concentrations of particulate matter at 31 fixed distances. http://webcam.srs.fs.fed.us/tools/vsmoke/index.shtml

Table 4. Separate calculations that can be performed in IFTDSS.

Calculation (Module)
1. All fuel characteristics (FCCS)
2. All potentials (FCCS)
3. Available fuel potential (FCCS)
4. Calculate burn probability across a landscape (IFT-RANDIG)
5. Calculate consumption and emissions (IFT-FOFEM)
6. Calculate crown fire behavior (IFT-crown)
7. Calculate fire behavior across a landscape (IFT-FlamMap)
8. Calculate fire behavior for individual stands (IFT-FlamMap)
9. Calculate fire effects across a landscape (IFT-Consume)
10. Calculate minimum travel time (IFT-MTT)
11. Calculate probability of ignition from a firebrand (IFT-ignite)
12. Calculate probability of ignition from lightning (IFT-ignite)
13. Calculate safety zone size (IFT-safety)
14. Calculate spotting distance from a burning pile (IFT-spot)
15. Calculate spotting distance from a wind-driven surface fire (IFT-spot)
16. Calculate spotting distance from torching trees (IFT-spot)
17. Calculate surface fire behavior (FCCS)
18. Calculate surface fire behavior (IFT-surface)
19. Calculate tree mortality (IFT-FOFEM)
20. Computer generated treatment location (OptFuels – W. Lake Tahoe)
21. Computer generated treatment location (OptFuels)
22. Consume (activity fuelbeds)
23. Consume (manual loadings, activity fuelbeds)
24. Consume (manual loadings, natural fuelbeds)
25. Consume (natural fuelbeds)
26. Consume/FEPS (one activity fuelbed)
27. Consume/FEPS (one natural fuelbed)
28. Create a burn plan document
29. Crown fire potential (FCCS)
30. Estimate containment resources (IFT-contain)

Calculation (Module)
31. FCCS
32. FCCS/Consume (activity fuelbeds)
33. FCCS/Consume (activity fuelbeds) – batch
34. FCCS/Consume (natural fuelbeds)
35. FCCS/Consume (natural fuelbeds) – batch
36. FCCS/Consume/FEPS (one activity fuelbed)
37. FCCS/Consume/FEPS (one natural fuelbed)
38. FEPS (manual)
39. FEPS (pile burning)
40. Fire Weather Statistics (IFT-FireFamilyPlus)

Table 4. Separate calculations that can be performed in IFTDSS.

Calculation (Module)
1. Fuel consumption (activity fuelbeds, Consume)
2. Fuel consumption (natural fuelbeds, Consume)
3. Fuel loading (FCCS)
4. Heat release (activity fuelbeds, Consume)
5. Heat release (natural fuelbeds, Consume)
6. Manual treatment location (user-defined treatments) (IFT-FlamMap)
7. Manual treatment location (user-defined treatments) (IFT-MTT)
8. Manual treatment location (user-defined treatments) (IFT-RANDIG)
9. Manual treatment location (user-defined treatments) (Worst Case FL – Risk)
10. Manual treatment location (FVS treatments)
11. Mastication (IFT-FVS)
12. NTLL to LCP (IFT-FVS)
13. No Treatment (IFT-FVS)
14. Pile burn surface fuel (IFT-FVS)
15. Pollutant emissions (activity fuelbeds, Consume)
16. Pollutant emissions (natural fuelbeds, Consume)
17. Predict crown scorch height (IFT-scorch)
18. Predict fire size and spread distance (IFT-size)
19. Predict surface fire behavior, size, and spread distance (IFT-surface+size)
20. Prescribed Burn (IFT-FVS)

Calculation (Module)
21. Risk Assessment - Worst Case Flame Length
22. Risk Assessment - by Flame Length Probabilities
23. SVS for No Treatment (IFT-SVS)
24. Surface fire behavior (multiple fuelbeds, single scenario, FCCS)
25. Surface fire behavior and potentials (FCCS)
26. Surface fire potential (FCCS)
27. Thin From Below (IFT-FVS)
28. Thin a species across a Dbh range (IFT-FVS)
29. Thin with fuel piled and burned (IFT-FVS)
30. Total carbon (FCCS)
31. Smoke dispersion (VSmoke on the BlueSky Cloud)