**16th ICCRTS**

**"Collective C2 in Multinational Civil-Military Operations"**

**Paper 110:**

# Incident Management Systems Evaluation and Usability Assessment

**Topic 8:**

Architectures, Technologies, and Tools

**Author:**

Tania E. Randall

Defence R&D Canada – Atlantic

PO Box 1012, Dartmouth, NS B2Y 3Z7

(902) 426-3100x283

tania.randall@drdc-rddc.gc.ca

# Incident Management Systems Evaluation and Usability Assessment

**Tania Randall**
Defence R&D Canada - Atlantic
PO Box 1012, Dartmouth, NS B2Y 3Z7
tania.randall@drdc-rddc.gc.ca
(902) 426-3100x283

## ABSTRACT
In order for emergency managers to effectively track, organize and manage emergency events they require straight-forward tools with an adequate level of functionality. With the myriad of Incident Management Systems (IMS) available in the marketplace, it is difficult to know which one is the best fit for an organization. This paper discusses available features within IMS systems as well as variations in implementations across systems. Since these tools are typically used in high-stress, quick-paced environments, it is critical for these tools to be easy to use. Usability is both difficult to specify in a Statement of Requirements (SOR) and costly to evaluate. However, the potential 'cost' of not considering system usability is realized when users cannot or will not use a chosen system because it impedes getting work done rather than facilitating it. The paper describes an ongoing experimentation program where untrained participants complete core tasks in various IMS systems while completion times and subjective assessments of usability are captured. It is aimed at understanding which design implementations lead to the most usable systems, framing expectations for IMS system usability in general, and informing the process of specifying usability requirements in a measurable and effective manner.

## Author Keywords
Incident Management Systems, Requirements Analysis, Requirements Specification, Tender Process, COTS Software, Usability Experimentation

## INTRODUCTION
In times of emergency, the operations center team must (among many things) track details of the emergency, manage resources involved in the response or recovery effort and notify team members of pertinent developments while maintaining an overall situational awareness that facilitates effective decision-making and a coordinated response. While training, planning and preparation are clearly associated factors, another critical component is the software toolset that supports them.

For the purposes of this paper, the term 'IMS system' will refer to a web-based system supporting the management of an emergency through distributed incident logging, resource and task management, messaging, and the provision of geographic information system (GIS)-based situational awareness. Information may be collectively entered into or retrieved from an IMS system by emergency managers and supporting staff in the emergency operation centers (EOCs), on-site responders via handheld mobile devices, and any approved personnel that become relevant to a given emergency. Beyond supporting the response or recovery effort itself, IMS systems can produce detailed logs of all developments, actions and requests for review and analysis, as well as accountability purposes.

Given the distinct mandates of the various Emergency Support (ES) organizations (e.g., Department of Defence, Public Safety, Police, Firefighters, etc.), it seems reasonable that their requirements for a software support tool will also differ to some degree. Thus, it is unlikely that a single IMS system will be the best solution for all organizations or members. In addition to pointing out the much-talked-about requirement for IMS system interoperability (at least at some basic level), it also highlights the need for organizations to be able to understand and specify their individual needs in a way that is conducive to the writing a Statement of Requirements (SORs) to initiate the procurement process.

This paper provides a discussion of the breadth of functions that can be expected in an IMS system, and then focuses on a non-functional requirement: usability or 'ease of use', that should also be considered when selecting a product. While many tools may meet an organization's functional requirements, a smaller number may be appropriate in terms of the user experience. A tool that is too complicated to use is likely to be set aside, or worse, to hinder the progress of the emergency response that it was designed to aid.

While the need for an easy-to-use system is readily recognized by potential users and identified as a core requirement in an IMS system report produced by the US Department of Justice in 2002 [1], specifying this requirement in a measurable way for SORs is not a straight forward task. Generally speaking, usability issues are not uncovered until the software is in the hands of the user, at which time it is too late to affect the procurement process.

*Prepared for 16th ICCRTS,, "Collective C2 in Multinational Civil-Military Operations", June 2011, Quebec City, Quebec, Canada*

This paper discusses ways in which measurable usability requirements may be specified for inclusion in an SOR.

Arranging statistical trials as part of a competitive bidding process is impractical, not to mention likely to discourage bidders from engaging in the competition in the first place. It is suggested in [2] that an SOR state the required performance level (e.g., 75% of all users must be able to complete Task X without assistance), and leave it to the vendor to provide evidence to back this up.

The difficulty, however, for the contract writer is to know what a reasonable target value would be (i.e., '75%' in the above example). If the target value is set too low, all vendors may pass, providing no aid to the selection process. If the target value is set too high, there's a risk that no vendors will comply.

The remainder of this paper outlines a usability experimentation program that is underway at DRDC Atlantic. It has aims to provide overall clues about what makes an IMS system usable (or unusable) and to identify some target levels that can inform the process of specifying measurable usability requirements.

## DETERMINING REQUIREMENTS

Prior to initiating an investigation of available systems, it is important to identify the needs of the organization. Many potential users of a new IMS system are already familiar with some of the systems on the market. However, identifying the organization's goals for the new system can be completed without detailed knowledge of available systems. In fact, reviewing existing systems first could negatively influence the user's ability to focus on the true requirements as they become awed by features that ultimately may not serve to advance the effectiveness of the organization. This requirements gathering effort should include input from all groups affected by such a system; general users, managers of users, administrators (e.g., for account or contact management), and information technology (IT) personnel (e.g., for server maintenance), are likely candidates.

This initial list of *organizational requirements* should be goal-focused rather than feature-focused. That is, it should focus on what the users need to be able to do, rather than the technical implementation that will allow them to achieve it. This list may also address non-functional system requirements such as usability, reliability, and interoperability with other systems, without getting into the details of how these requirements could be assessed.

There are a variety of techniques that can be employed for doing requirements analysis, and ideally the effort would be led by someone not immersed in the organization itself (thereby unintentionally influencing the results), and trained or experienced in human factors engineering and more specifically, requirements analysis. It is recognized, however, that time and budget restrictions will often not allow for this, so it is necessary to make do with the resources at hand. In [3] a variety of methods for needs identification are discussed: user surveys, focus groups with stakeholders from each user group, interviews, scenario and use case discussions, and 'future workshops' which require users to consider where they hope the organization will be 10 years from now.

The next step is to begin looking at commercially available systems and identifying *technology solutions* that satisfy the organizational requirements; these will form the basis of the *technology requirements* that ultimately appear in the SOR. Systems already in use by other communities that either work with the organization in question, or have similar mandates should be investigated. Internet searches will, of course, reveal many options as well. This step is not meant to be a comprehensive search of available products, but rather to confirm (or not) what can be achieved in typical systems, and to gauge what features will be required to meet the identified organizational requirements. In fact, the organizational requirements could be used directly in the SOR, and some of them may be.

The next section of this paper provides the reader with some user-relevant questions that they may wish to consider asking the vendor in order to inform the above activity.

## INCIDENT MANAGEMENT SYSTEMS

### Questions to Ask About General Features and Qualities

There are many IMS solutions currently on the marketplace. Generally there is a fair bit of overlap in their capabilities, but yet they all are unique in one way or another and are difficult to compare overall. Comparisons on a feature by feature basis may be justified, but determination of a single product as the 'best product' for all potential users is generally not feasible. It is only possible to determine the best fit for a given group of users, and this largely relates to determining which product offers a combination of features and qualities most in line with the identified requirements. The questions listed below can be asked of vendors to gain a broad understanding of the product; they are not exhaustive, but should be more than sufficient for an initial pass through of a given system and to highlight the main capabilities of IMS systems in general. The volume of questions should give the reader an appreciation for the complexity of this decision-making process.

Categorizing these questions is challenging. They do not necessarily fit into tidy groupings. For the purposes of this document, they are organized by relevant user group. There is overlap between the categories, as more than one user group may be concerned about the same thing (e.g., usability).

#### *For General Users*

These are the primary users of an IMS during an emergency. Questions relevant to these users include:

- Does the system support: Creation, modification and updating of new incidents? Attaching files to incidents? Linking of multiple incidents to a larger event (e.g., multiple road blockages following a hurricane) and creation of sub-incidents (e.g., car crashed into tree blocking road)? Is there a limit on the number of levels (i.e., can there be a sub-sub-incident)? What are the required fields for an incident? Does the user have control over who can see a particular incident?
- Does the system support: Tracking of resource availability? Assigning resources (people, supplies, response teams or vehicles) to incidents? Assign tasks to resources? Monitoring/updating task progress (e.g., task accepted and underway)?
- Does the system have a GIS-based mapping tool to indicate locations incidents, resources, etc.? Does it accept GPS feeds of responder locations (such as from an iPhone)? Does it support annotation on the map to share with others (e.g., exclusions zones, evacuation areas, etc.)? Is it leveraging an external mapping resource (e.g., Google Maps) or does it use a custom solution?
- Does the system include a document storage area? Does it support usage of electronic Standard Operating Procedures? If so, can they be used to generate checklists of items to achieve, and can progress status of each item by tracked/recorded? Can they be used to immediately create pre-defined tasks to assign to users when activated?
- Does the system support sending/receiving alerts or alarms of critical events in an overt way such that all users will immediately take notice?
- Does the system support sending/receiving automatic notifications of changes to a known incident or creation of a new incident?
- Does the system support sending e-mail messages external to the IMS system? Can it include links to relevant information within the IMS system that users can click on and open (given they have an IMS account)? For e-mails sent to those without an IMS account, can the e-mail automatically include a summary of an incident (without requiring retyping or copying and pasting)? Is it possible to attach files to e-mail messages? Is it possible to develop pre-formatted, event-relevant e-mails for quick dissemination when needed? Does it support all of the above items for SMS text messaging?
- Does the system offer an advanced notification feature (or add-on) that supports instant widespread messaging to many people at once (such as for recalling large groups of people)? If so, does it use various means to reach the people, such as phone, cell, e-mail, text, etc.? Does it persist until it receives confirmation of the message reaching its intended recipients?
- Does the system offer an integrated chat service to chat with other IMS users? Are time-stamps of messages visible in the chat window? What control is there over who sees and participates in chat messages?
- Are time-stamped logs available for all activity (incident creation and updates, resource assignments, etc.) for review at shift changeovers? Do logs include specification of who made the change and what exactly the change was? Are time-stamped chat logs available?
- Is the system configurable to only show menus, options and data relevant to a user's role?
- Does the system support display of RSS feeds chosen by user (e.g., weather, CNN, etc.)?
- Does the system support usage on a handheld mobile device? Is an additional product needed?
- Does the system have context-sensitive help? What are the hours of the help desk?
- What is the recommended training time for this system? Is it easy to use, and if so, what evidence exists to support this?
- What is the process for requesting bug fixes, changes to the system, or new features?

*For Administrative Users*

The name of this group is not meant to suggest that these users need to be administration staff; they may very well be general users with the additional responsibility of administering the system. These users have the role of supplying the system with complete and up-to-date information that will allow it to provide maximum benefit to general users when an event actually occurs. Questions relevant to the work of these users include:

- Does the system support creation of contact lists (names, phone numbers, e-mails, etc.) for IMS users as well as relevant external persons? Does it allow for importing of contact information from common office software such Microsoft Outlook? Does it offer any mechanism to help keep the contact information up-to-date? Does the system distinguish between general contacts, IMS users, people resources, etc.?
- Can groups of contacts be created and named to facilitate message sending during an incident (e.g., all police officers, or all information technology personnel)?
- Does the system support entry of details about potential resources? Can capabilities, suppliers (if not owned), requirements (e.g., takes three people to operate), limitations, and numbers available be indicated?
- Does it support creation/entry of SOPs? If so, are these uploaded documents (e.g., pdfs) or do they need to be typed from scratch (e.g., into checklists) to fully take advantage of this capability?
- What is involved in creating an account for a new user? What information is required? Which web-browsers are supported? Which browser plug-ins are required for the client-side browser? Who can create new user accounts?

*For Managers of Users*

These users are the managers of the general users (and possibly other user types as well), and likely do not spend a lot of time entering information into the system themselves; rather they must be able to obtain whatever information they need, when they need it, from the system. Arguably they care about all of the items in each category since their ability to do their job is directly influenced by the ability of their employees to do theirs. They may have some specific concerns of their own as well, however, and these questions address some of those:

- Does the system include time-stamped, detailed logs of all system entries (including who entered them), suitable for determining accountability and for after-action-review?
- Does the system support generation of summary reports for managers, media, etc.? Does it support Incident Command Systems (ICS) forms? Does it support user customization of forms?
- Does the system allow for viewing of all spatially represented items (incidents, resources, etc.) for one incident or for multiple incidents, on a single map in order to provide quick situational awareness?
- What is the pricing model for the system (i.e., is it licensed by number of users (and if so, is it concurrent or total number of issued accounts?), by site, by organization, etc.)? What about costs for: database licensing fees, map-usage fees (if vendor leverages existing online GIS technology), text messaging services, set-up fees, maintenance agreements (for upgrades, support, etc), training fees, etc.?
- What organizations are currently using this software in an operational setting? How large is the client base? When did this product enter the marketplace? What is the history of the company?

*For Information Technology Personnel*

These users are involved in setting up and maintaining the system and its integrity.

- Are there options for both self-hosted servers and vendor-hosted servers? What system is in place to deal with the possibility of damage to the server location?
- What control mechanisms are available to prevent users from viewing or changing data without authorization? What security measures are in place for the handling of e-mail and storage of data? Does the system log who signs into the system and when? Does it capture failed login attempts, and provide notification of suspicious activity?
- Has exchange of incident data between this system and another IMS system been successfully demonstrated? What data exchange protocols are supported? Is purchase of a separate product required to aid with data exchange? If there was a requirement to integrate this system with another system, what process would need to be undertaken?
- If a self-hosted server solution is selected, can software updates and upgrades be installed by the organization's IT staff (perhaps with phone support), or would the vendor require a site visit? What do other clients typically do? Typically, how frequent are updates and upgrades?
- If a vendor-hosted solution is selected, what uptime can be expected from the server?
- Is the content of drop-down boxes or pick lists for the client-side interface easily populated or modified by IT staff without vendor support?

Admittedly these are a lot of questions to delve into before the work of accepting and evaluating bids begins. At this point finding the perfect product is unlikely to be of much benefit, nonetheless, going through this questioning process with a few vendors will help to refine the identified requirements. It may become clear that more detail is required for existing requirements to ensure applicability to the organization, perhaps new requirements will need to be added or old ones will be deleted. This information will aid the evaluators by allowing them to understand exactly what is desired upfront so that rating scales can be appropriately developed and so that vendors can better estimate the applicability of their product before entering the competition.

**Variations in Feature Implementations**

Different vendors tend to implement features of the same name in different ways. Thus it is necessary to adequately specify requirements upfront and to thoroughly investigate *how* a vendor satisfies a specific requirement.

*Incident Structure*

Each vendor may have a different idea of what should be contained in the basic incident structure (e.g., incident name, location, date/time, etc.). It is important to choose a product that aligns with the organization's 'mental model' of an incident as closely as possible. This serves two purposes: (1) minimizing the difficulty of figuring out where a specific piece of data should be entered and (2) limiting the number of non-applicable data fields; even when fields are not mandatory, users may feel obligated to fill them in.

Figure 1 shows some of the fields relevant to incidents for four commercially available products. Other products will no doubt vary from these as well. In these four systems alone, the number of fields that can be filled in varies from ten to well over thirty fields. The organization of the fields varies as well. For example, in two cases, the type of incident is specified before a name for the incident is specified. There is no right or wrong implementation; however, it is likely that some systems will be a better fit for a particular organization than others.

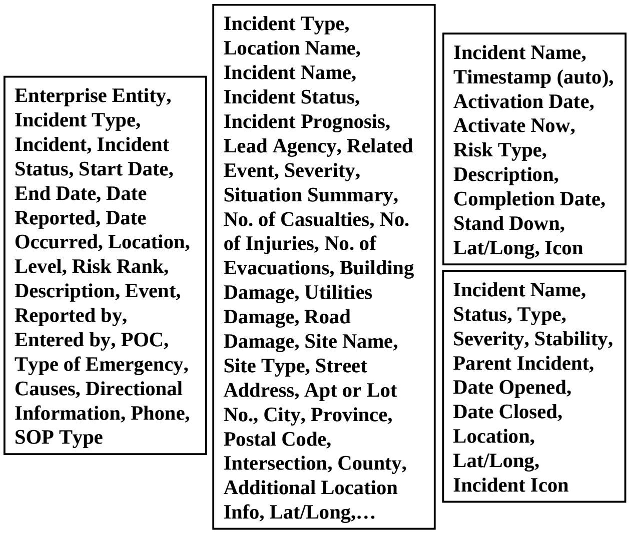


**Figure 1: Examples of Different Interpretations of Incident Structure**

*Data Logs*

The ability to log all data entered into an IMS system for potential future review is a key benefit of having a formal system for handling incidents. Presumably all IMS systems have some level of logging. However, the level of detail can vary considerably, as can the format that it is recorded in, affecting the ease with which it can be reviewed.

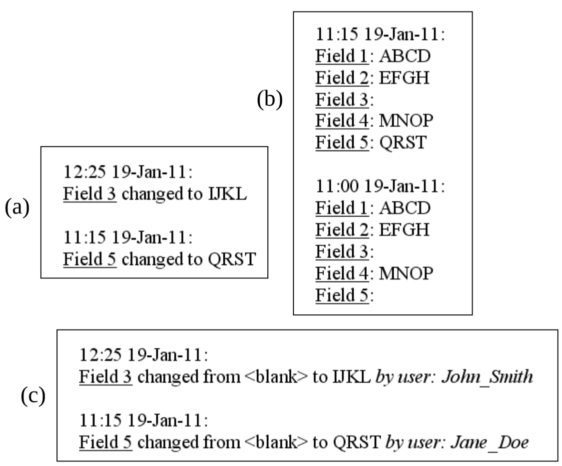


**Figure 2: Examples of Different Log Formats**

Figure 2 shows a few examples of log formats that were found in some commercially available IMS systems. In example (a), the log specifies when a change was made, which field was changed, and the new value entered. In example (b), complete copies of all fields in the record (e.g., an incident) are logged along with the time that the record was updated. In order to understand what change was made, it is necessary to compare this log entry to the previous log entry for this record. On the other hand, this method makes it easy to understand the status of the incident at that time, since the inclusion of all fields gives a more complete picture. In example (c), the log specifies when a change was made, which field was changed, the previous value of the field, the new value of the field, and the person responsible for making the change. This format adds some accountability information to the log, and may provide the reviewer with a better understanding of the purpose of the change since the original data is also available. Depending on the organization's requirements, any one of these options could be sufficient; however, one will likely resonate more with the organization than another.

*Alerts*

The definition of an alert varies from vendor to vendor just as alerting requirements likely vary from organization to organization. Some systems take a somewhat passive approach to alerting, using e-mail messages to deliver these high priority messages. Other systems provide the option of sending a critical message in a more obvious manner. For example, this can involve a window that pops up on top of the IMS system that requires acknowledgement before it will go away, or a message that scrolls horizontally across the top or bottom of the screen with flashy colors making it difficult to miss. While this method is advantageous in terms of its visibility, it is only effective when the intended recipients are logged into the IMS system.

*Mobile Support*

Many emergency response personnel have a requirement to access or enter information from a web-enabled mobile device (iPhone, Blackberry, iPad, etc.), in addition to working in a desktop environment. Since the systems focused on in this paper are all web-based (i.e., the user only requires a web browser and an account to access the IMS system), it is true that all products should be accessible from any mobile device with a web browser (assuming that the server itself is not behind a firewall). However, some vendors provide (for free or at an additional cost) a mobile version of the same product, or a second 'mobile' product that integrates with the main product. In these cases, buttons, menus and views are customized for small mobile devices such as a Blackberry, so that text remains legible and displays are appropriately designed for the limited real estate. In many cases, using a mobile device to access a web page that has not been designed with this in mind will prove to be an unpleasant and potentially unsuccessful experience.

## SOFTWARE USABILITY

### Definition

Usability consultant Jakob Nielson [4] suggests that the usability of a system is defined by five quality components: (1) *learnability*, the ease with which users can accomplish basic tasks the first time they use the system, (2) *efficiency*, the speed at which tasks can be performed once they are learned, (3) *memorability*, the ease with which infrequent users remember how to use the system when they return to it, (4) *errors*, the quantity, severity, and recoverability of errors made by users, and (5) *satisfaction*, the overall satisfaction that the user has with the system.

Developers will often say that it is impossible to create a system that will score highly in all five categories [5].

**Importance**

Software usability is desirable in any environment, but particularly in ones where:

- users do not have the need or opportunity to use the software on a regular basis, and thus may forget their training,
- there is a relatively quick turnover in users (e.g., military) so that it is hard to keep people trained or to develop 'power users',
- untrained personnel may be required to fill a position unexpectedly,
- there are potential users that cannot be trained ahead of time because they are simply unknown until the emergency occurs (e.g., the security officer of a random civilian building under threat),
- users that are not computer-savvy may have difficulty remembering or using non-intuitive interfaces even after training,
- users work in a time critical environment where lives are at stake, and
- users are already overloaded with software systems to monitor or manage and are naturally resistant to new ones; "lack of usability" will certainly not encourage usage of a new system.

**IMS System Usability Issues**

These are some examples of usability issues encountered while testing a number of commercially available products:

- *no indication of required fields* in a form until after the 'submit' button has been selected and errors are generated,
- *inconsistencies within the product*, that is, the user is required to follow different processes to achieve the same effect in different parts of the software,
- *actual errors or bugs* (while these are not design issues, they certainly contribute to the user's overall satisfaction with the product),
- *clunky maps*. Most users are accustomed to interacting with electronic maps in a certain way (e.g., through MapQuest, Google Maps, etc.) and expect other products to behave similarly, however, some IMS-based maps do not,
- *unnecessarily long navigation paths*. For example, a page that shows all currently assigned resources to an incident should have a button/option to 'assign a new resource'; it should not be necessarily to go back to the main page and navigate through menus to get there,
- *unclear rules*. For example, it may be necessary to assign a resource to an incident before assigning a task to the resource; when the user attempts to assign a task to an unavailable resource, the source of the resulting error may not be clear.

Clearly these issues are not apparent in all products, nor is this a comprehensive list.

## SPECIFYING USABILITY REQUIREMENTS (IN GENERAL)

While documentation on measuring usability is fairly easy to come by, documentation on specifying usability requirements appears sparser. However, the author in [2] discusses six styles of specifying usability requirements which can be applied to the specific case of specifying requirements for IMS systems. The styles are: Performance Style, Defect Style, Subjective Style, Guideline Style, Process Style, and Design Style.

*Performance Style Requirements*

The Performance Style addresses specifying performance-based usability requirements; it involves identifying user groups and key system tasks where usability is likely to be an issue for a given group. In this case user groups may be more specific than those that have been previously identified in this paper. For example, 'General Users with no training or experience' may be one group, and 'General Users with one week of training' might be another, since there would be different expectations on their capabilities to perform in the system. An example of a performance-based requirement specification could be, "Demonstrate 95% confidence that 75% of untrained, inexperienced users are able to enter a new incident within 5 minutes".

*Defect Style Requirements*

The Defect Style addresses specifying requirements based on an acceptable level of defects (problems), which are classified according to how severely they affect the user's ability to accomplish their goals. An example of a Defect Style specification could be: "Demonstrate 95% confidence that no more than 20% users will fail to enter a new incident on their first attempt".

*Subjective Style Requirements*

The Subjective Style addresses specifying requirements based on subjective satisfaction; it involves collecting opinions from system users on their overall satisfaction with the system. There are some standard questionnaires for measuring satisfaction of software or websites, including the System Usability Scale (SUS) [6]. The SUS questionnaire consists of 10 questions to answer on a scale of 1 to 5. The questions alternate between being positively and negatively phrased, to help counter the bias of subjects to agree with any statement. A score between 0 (worst) and 100 (best) is calculated based on the answer to all ten questions. So, an example of a subjective satisfaction-based requirement could be, “Demonstrate 95% confidence that 75% of new users score at least a 60 on the SUS questionnaire”.

*Guideline Style Requirements*

The author in [2] suggests that the Guideline Style is not relevant to a tendering situation since it is unlikely that the developer was following any standard style guidelines (e.g., MS-Windows or Common User Access (CUA)). And even when guidelines are followed, it is difficult to inspect adherence since guidelines may include hundreds of rules.

It may be possible, however, to specify a requirement for adherence to some specific, relevant guideline rules, rather than an entire standard guideline. Or, the guideline rules could be original based on the organization’s experience and familiarity with other software. An example of a specification could be: “The application must include a ‘Home’ or ‘Main’ button accessible from every page. Clicking this button returns the user to the entry page.” Another example could be: “The map component of the application must respond to mouse clicks and scrolls in the same manner as Google Maps”. Such guidelines are related to both the learnability and memorability components of usability, and possibly extend to the other components as well.

*Process and Design Style Requirements*

Neither the Process Style nor the Design Style are applicable to systems that have already been developed; recommending incremental usability testing of prototypes and providing sketches of desirable interfaces are not relevant with respect to off-the-shelf software.

**Determining Target Values**

All of the sample usability requirements provided above could aid in the selection of a usable system; they would certainly be more effective and measurable than a “must be easy to use” requirement statement. Still, how does one determine the appropriate numbers (i.e., the *target values*) to include in such statements (e.g., should it be 75% of users or 85% of users? 5 minutes or 7 minutes?)?

Using appropriate numbers in these requirement statements is critical to them adding value to the evaluation process. If the target values are too low, all or most products may be compliant in which case extra work has been created for both the vendors and the evaluation team, without notable benefit to the product selection effort. On the other hand if the target values are too optimistic, products may be unnecessarily eliminated or vendors may choose to not enter a bid at all if they are uncertain of their ability to meet the stringent requirements. In [7] the author concedes that not having baseline data related to efficiency and satisfaction can prevent the development of meaningful SOR usability requirements.

In [2] the author suggests that one way to handle this is to have the vendor provide values for their product, and then compare the results across bids, rather than specifying the exact requirement. Assuming vendors have these data available (or are willing to obtain them before entering a bid), the evaluation team will need a creative way to rate/rank the responses without having any reference point. There is a risk, however, that none of the products will be sufficiently usable for the organization; this cannot be guarded against through a purely relative ranking.

Another option is to measure the current process that will be replaced/enhanced by the prospective system. These numbers can then be used as a minimum acceptable requirement to ensure that the new system does not leave users less effective or less satisfied than they already are.

A third option is to measure the usability of related systems and use those results as a reference point.

**IMS USABILITY EXPERIMENTATION**

This section of this paper discusses an experimentation effort underway at DRDC Atlantic that will assess the usability of four commercial IMS systems, all of which are currently in use in Canadian government offices. It is aimed at understanding which design implementations lead to the most usable systems, framing expectations for IMS system usability in general, and informing the process of specifying usability requirements in a measurable and effective manner. By predetermining the values achievable by a subset of commercial products, an understanding of what is obtainable will be realized; this knowledge can feed into the development of performance, defect and subjective style requirements. Through participant comments and the analysis of troublesome features identified through the trials, insight into guideline style requirements can be fostered. The discussion of the experimental plan may also prove valuable to the buyer or vendor that ultimately wishes to test a system for compliance with a written requirement, or to measure a current process.

This experimental plan was approved by the DRDC Human Research Ethics Committee (HREC) in September 2010, a requirement for all experiments at DRDC establishments involving human participants. To date the experimental infrastructure is in place and a pilot study has been run to test the experimental procedures and data logging tools. Completion of all runs (24 in total) is anticipated by the end of March 2011. The outcomes of this experiment should aid with the identification of appropriate target levels for usability requirements and of general IMS usability guidelines.

**User Group Selection**

This experiment will focus on tasks relevant to the General Users group. Usability is of particular importance to this group since they will typically be operating in fast-paced, high-stress environments; they will have limited to no time to look at help menus or seek customer support so the software must be straightforward. Furthermore the experiment will focus on untrained users with no experience with the software being evaluated. There are two reasons for this. The first is that the motivation for this effort came from an operations centre where it was very common for an untrained person to be called into the operations centre and be expected to perform. Second, it is much easier to find potential volunteers without any exposure to these products than it is to find them with experience (not to mention the difficulty in quantifying the varying degrees of experience).

**Participants**

Twenty-four individuals will be recruited to participate in this study. Participants will be male and female DRDC Atlantic civilian and military employees, members of the Canadian Forces, and/or members of the general public and must be at least 18 years of age. Participants must also consider themselves competent users of basic computer software (e.g., office software) and be familiar with webpage navigation. In addition, they can not have any experience with any of the four products being evaluated. Each participant must read and sign a consent form, will be required for approximately two and a half hours, and will be compensated according to DRDC guidelines.

**Experimental Infrastructure**

Participants will sit in front of two monitors, controlled by a single computer, mouse and keyboard. The monitor on the left will display the system being evaluated, and the monitor on the right will display instructions and questionnaires for the participants via a custom-made Microsoft Access database (see Figure 3). Chairs, keyboard and mouse will be offset to the left such that the subject is sitting in front of the IMS system. CaptureWizPro screen recording software will run on each IMS system display, and a mouse logging utility will capture all mouse clicks (time, type, location) at each participant station. The Access database will capture the times at which the user starts and ends each task.

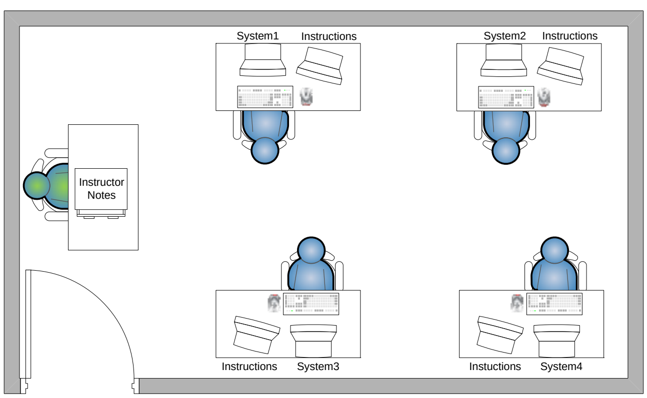


**Figure 3: Experiment Room Set-Up**

**Task Selection**

Four tasks were selected as the basis for this experiment. These tasks were chosen because they are common tasks for general users to perform in an IMS system, and also because they can be achieved in all four systems being assessed. The tasks are:

<u>Task 1</u>: Add an incident/emergency and use the map feature to define its location.

<u>Task 2</u>: Update a second (different) incident.

<u>Task 3</u>: Assign a resource to a third incident.

<u>Task 4</u>: Using the map feature, estimate the distance between a particular incident and a fixed point.

The instructions provided to users will include complete details for each of these tasks, such as incident names, resource types, etc. Participants will be instructed to only fill in the data provided, and to leave all other non-required fields blank.

**Experimental Procedure**

Participants will run through the experiment four at a time. Each run is divided into 4 sessions, one for evaluating each system. During each session, each of the four participants will evaluate a different system. As well, all twenty-four participants (spread across six runs) will evaluate the systems in a different overall sequence in an attempt to counterbalance any learning effects as well as any biases that may result from reviewing one system before another (e.g., if one system has really poor usability, the next system reviewed might seem especially good in comparison to the first system and be given more positive scores than it would have been given if it had been used first). Obtaining data from twenty-four participants should also allow for meaningful statistical analysis of the results.

In each session, the participant will perform all four tasks two times. The second time they complete the four tasks, the details will vary slightly (e.g., name and location of the incident), but the task structure will hold and the process to achieve it will be the same. The reason for completing such similar tasks a second time is to capture how difficult the task is to achieve during the very first attempt, and then to capture how difficult it is to achieve once it has already been learned. In some cases users may roam around the system trying to figure out the task the first time, and then eventually have an “Ah ha” moment. Once they have figured out the ‘trick’ to the system, they may achieve the task much faster. There is also the possibility that they will not be able to complete the task at all. In this case, they can click the ‘give up on this task’ button that only becomes enabled after a pre-set amount of time (to prevent them from giving up too quickly). Participants are not permitted to talk to each other, nor are they allowed to seek help from the instructor. They are, however, encouraged to use any built-in help features provided by the system if required.

Each session lasts a total of 30 minutes, regardless of how quickly a participant may complete the tasks. At the completion of each session, the participant is set up to review the next system in their sequence, until they have reviewed all four systems.

### Questionnaires

Prior to beginning their first session, each participant will fill out a background survey that queries their general familiarity and comfort level with software applications, as well as their job category and age group.

After completing each task, the participant will answer the Post-Task questionnaire, consisting of only three questions, using a scale of 1 to 5, where 1=very low and 5=very high:

1. For each step of this task, it was clear to me what I needed to achieve.
2. Navigation through this system to achieve this task was straight forward.
3. The number of steps required to accomplish this task was reasonable.

These questions are roughly inspired by the Cognitive Walkthrough process [8]. They are also given a comment box to fill in if desired.

After completing each session (i.e., all eight tasks in one system) the participant will answer the SUS questionnaire, once again using a scale of 1 to 5, where 1=very low and 5=very high. The SUS questions are as follows [6]:

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I would need to learn a lot of things before I could get going with this system.

Following completion of the SUS questionnaire, they are presented with a list of 50 adjectives, both positive and negative. They are asked to read the list and select all words (5 or more) that apply to the product they just assessed. Once they accept their answers to this page, they are presented with a reduced list of only the words that they selected. They are then asked to select the five words that best represent that product.

### Other Data Collection

In addition to the data collected directly from the participants, time to complete each task, number of mouse clicks to complete each task, and screen captured video of all sessions are recorded. As well, there will be post-session checks performed by experiment investigators to see that the tasks were indeed completed correctly.

Prior to running the experiments, 'expert users' (i.e., the investigators) will also determine the minimum number of clicks required to achieve each task in each system, given a particular starting place.

### Data Analysis

Most of the data output from this study will be fairly straight forward, but the adjective selection activity and the collection of mouse clicks per task, warrant an explanation.

The results of the adjective selection task will be tallied across all participants and used to create a Word Cloud [9] for each system. A Word Cloud provides a quick visual representation of how desirable the user perceives the system to be. Words in the largest and darkest font indicate the words most frequently selected, while those chosen less frequently tend to fade into the background. Use of word clouds to represent system desirability was addressed in [9] and is an extension of the card sort techniques discussed in Microsoft's Desirability Toolkit documentation [10]. Figure 4 shows an example of a word cloud found in [9].

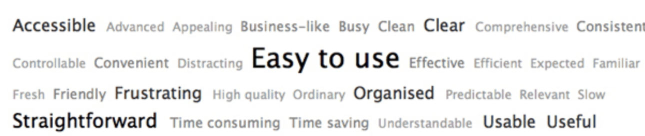


**Figure 4: Example of a Usability Word Cloud**

The number of mouse clicks required for each task by the novice user will be compared to the number of mouse clicks required by an expert system user to accomplish the same task. Any mouse clicks beyond the 'required' number of mouse clicks will be identified as unproductive (i.e., time spent on searches, help menus, indirect or erroneous navigation through the system, or making and correcting errors). Video files of screen captures will allow investigators to 'dig deeper' when they believe there are issues that deem further investigation. Thus, if there is a question about the volume of clicks, investigators can pull up the corresponding video file(s) and see what was happening.

Ultimately, analysis of data collected is expected to produce the following information for each system:

(a) Mean time to complete each task without prior experience,
(b) Mean time to complete each task once learned,
(c) Mean number of unproductive mouse clicks for each task without prior experience,
(d) Mean number of task failures for each task per user without prior experience (where a 'failure' implies the user could not complete the task or did so incorrectly),
(e) Mean rating for each question of the Post-Task questionnaire for each task,
(f) Mean SUS score (between 0 and 100),
(g) Desirability Word Cloud, and

(h) Identification of troublesome design features as highlighted by user comments, review of video files, and interpretation of other metrics (time to complete, number of failures, etc.).

For items (a) through (f) standard deviations and confidence intervals about the mean will also be identified. Further to this, the percentage of users achieving a certain outcome level (e.g., completing a task in 2 minutes) will be determined.

**Usability Components Addressed**

Referring back to the definition of usability discussed at the beginning of this paper, it is possible to map experimental outputs to most of the usability components. However, the *memorability* component cannot be addressed by this study since it is not a longitudinal study; it all happens within a two and a half hour period. It is conceivable to invite these same participants back for a second review at a later date to see how their scores compare to those obtained during their initial attempt at the tasks; this may be considered as a follow-on effort to this experiment.

Time to complete tasks without prior experience is an indication of *learnability*, as are the first two questions of the Post-task survey (albeit subjectively). Time to complete tasks once learned is the definition of *efficiency*, and the third question of the Post-task survey also addresses efficiency level. Both unproductive clicking and task failures are examples of *errors* of varying severity. Finally, the SUS scale is designed to measure user satisfaction, and the word cloud addresses desirability of the system which itself relates of features has potential to apply to any or all components of usability.

**Design Implementations and Usability**

An analysis of troublesome areas identified through a combination of long task times and notable unproductive clicking should help to identify specific design implementations that cause difficulty for the end user. Similarly, an alternative implementation in another system that resulted in shorter task times and less clicking may highlight better design choices. By generalizing such findings and comparing them with standard design guidelines, some general design rules for usable IMS systems will be developed.

**Preliminary Results**

In March 2011, following a pilot study earlier in the year, fifteen participants ran through the described experiment with only two of the four IMS systems. In one case, the collapse of the company supporting the chosen IMS system left DRDC with an un-supported system that had encountered serious errors that could not be remedied by DRDC staff. In the other case, obtaining a sufficient level of access (e.g., an ability to delete incidents) to the system was not possible within the desired timeframe. Nevertheless, it is worth examining the data that was obtained to see what can be learned. To date, only simple calculations have been done and no attempt has been made to generalize these preliminary results to the broader population of tools or people.

The following statements can be made about the sample population when evaluating “System1”:

- Average time to (1) add a new incident: 10m35s, (2) modify an incident: 1m14s, (3) assign a resource: 9m27s and (4) obtain basic information from the map: 3m37s on first attempts (based on 13-14 participants that claimed to complete the task),
  - On average it took 25m10s for all four tasks,
- Average responses to Post-Task questions 1-3 during the first round were 3.6, 2.6, 3.1; an overall ‘Neutral’ response, and
- The average SUS score was 58 (out of 100).

The resulting usability word cloud for this system is shown in Figure 5.

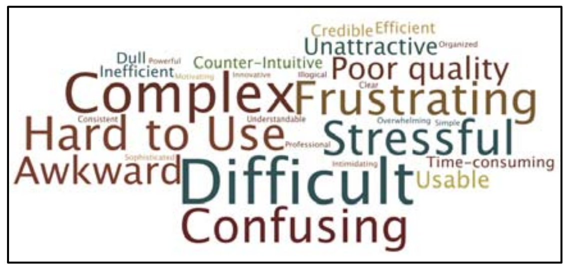


**Figure 5: Usability Word Cloud for “System1”**

The following statements can be made about the sample population when evaluating “System2”:

- Average time to (1) add a new incident: 7m29s, (2) modify an incident: 2m03s, (3) assign a resource: 3m48s and (4) obtain basic information from the map: 3m31s on first attempts (based on 8-16 participants that claimed to complete the task),
  - On average it took 16m51s for all four tasks,
- Average responses to Post-Task questions 1-3 during the first round were 3.2, 2.7, 3.2; an overall ‘Neutral’ response, and
- The average SUS score was 58 (out of 100).

The resulting usability word cloud for this system is shown in Figure 6.

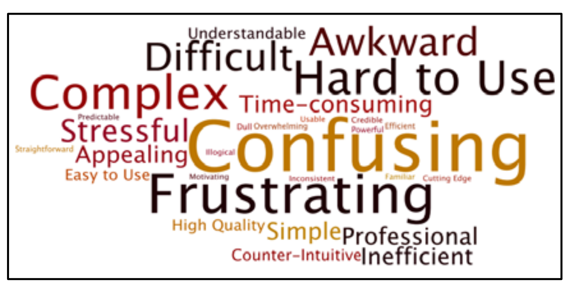


**Figure 6: Usability Word Cloud for “System2”**

These statements are based on data from all participants. It will be necessary to further consider how these results change based on participant comfort levels with computers and new software (addressed in the background survey). The majority of participants, however, indicated that they used computers every day and were very comfortable with new software.

The statements about task completion times are based solely on the participant's assessment of whether or not they completed the task correctly. At the end of each system evaluation, the systems were 'reset' by the experimentation coordinator as quickly as possible by manually undoing the work completed by each participant. During this process, notes were taken about the completeness and correctness of each response and still need to be compared to the self-reported data. This information will have to be taken into account in further analysis.

Additional analysis (such as click volumes per task) will require more time to complete as it requires integration of data from multiple sources. Similarly, participant comments and video captures of all runs are available for further consideration but will take more time to work through.

**Implications of IMS System Usability in General**

Since further analysis of existing data is required and experimentation with additional IMS systems is still needed, it is too early to make statements about IMS system usability in general. It is hoped that as a class of tools, the systems are rated reasonably high in terms of usability. If the results suggest otherwise, it will highlight the need for readily available training, support and reference materials (e.g., cheat sheets or cards focusing on specific tasks, desktop-accessible videos walking users through tasks, etc.) and the potential risk of having untrained or non-current users operating these systems in times of emergency.

**Guideline and Target Value Development**

At the conclusion of this exercise, some reference data should be available for future SOR writers to use when constructing their usability requirement specifications. While they may be evaluating or considering different products than those evaluated in this study, understanding what is achievable will help with the definition of measurable requirements that are strict enough to be useful but lenient enough to be met.

**In Pursuit of Product Improvement**

This experiment focuses on obtaining some broad measures of usability and captures all data electronically; unfortunately, there is no opportunity for participants to stop and talk about the problems they are having as this would hinder the ability to capture accurate task time measurements. However, through comment boxes following each task, and analysis of task times, mouse clicks and video files, it is anticipated that some specific problems will be uncovered. This information (as well as the time and error data) will be provided back to the vendors of the products being reviewed. At least one vendor of the products under review is welcoming this feedback and plans to work it into their next product review cycle.

## CONCLUDING REMARKS

Users of IMS systems are often in the position of dealing with events that affect both property and lives. Ensuring that they have the proper tools to meet their needs is critical. Many such organizations will purchase a COTS product for this purpose. In order to obtain the right tool it is necessary to understand the organization's goals and how these translate into technology solutions that can be detailed in an SOR. In addition to selecting a tool that 'does the right things' it is important to select a tool that 'does things well'. Ease of use for an emergency management tool can be the difference between a tool that helps the organization and a tool that hinders it. Stating that a product must be 'easy to use', however, is not sufficient to ensure the required results; identification of user groups, core tasks, and measureable usability-related requirements is necessary. These requirements can be stated in terms of performance requirements, acceptable error rates, subjective assessments, and/or style guidelines. In each of these cases, having some perspective on what can be reasonably expected of IMS systems could simplify the formulation of the formal requirements. An experimental program assessing IMS usability is underway at DRDC Atlantic that aims to provide this perspective as well as some general guidelines about designs that make these systems usable.

## REFERENCES


1. Ashcroft, J., Daniels, D., and Hart, S., ***Crisis Information Management Software (CIMS) Feature Comparison Report***, U.S. Department of Justice, National Institute of Justice Special Report 197065, October 2002.
2. Lauesen, S. and Younessi, H., ***Six Styles for Usability Requirements,*** Proceedings of the 4th International Requirements Engineering: Foundation for Software Quality Workshop, Pisa, Italy, June 1998.
3. Maguire, M. and Bevan, N., ***User Requirements Analysis***, Proceedings of IFIP 17th World Computer Congress, Montreal, Canada, August 2002.
4. Nielsen, J., ***Usability 101: Introduction to Usability,*** http://www.useit.com/alertbox/20030825.html.
5. Lauesen, S., ***Usability Requirements in a Tender Process***, Proceedings of the Australasian Computer-Human Interaction Conference (OZCHI) '98, Adelaide, South Australia, 1998.

6. Brooke, J., ***SUS - A Quick and Dirty Usability Scale***, Redhatch Consulting Ltd., UK, published in "Usability Evaluation in Industry" (1996), pp. 189-194.
7. Jokela, T., ***Determining Usability Requirements into a Call-for-Tenders. A Case Study on the Development of a Healthcare System,*** Proceedings of NordiCHI 2010, October 2010, Reykjavik, Iceland.
8. Cognitive Walkthrough: http://www.pages.drexel.edu/~zwz22/CognWalk.htm (accessed May 2011).
9. Travis, D., ***Measuring satisfaction: Beyond the Usability Questionnaire,*** published in UserFocus online, March 2008: http://www.userfocus.co.uk/articles/satisfaction.html (accessed May 2011).
10. Benedek, J. and Miner, T., ***Measuring Desirability: New Methods for Evaluating Desirability in a Usability Lab Setting***, Usability Professionals' Association Conference Proceedings, 2002.