FIRE SERVICES
COMMISSIONER
VICTORIA

REFERENCE ARCHITECTURE

VICTORIAN INFORMATION NETWORK FOR EMERGENCIES (VINE)

MAY 2013

Version 1.2

LEADERSHIP
INTEGRATION
ACCOUNTABILITY

WORKING IN CONJUNCTION WITH

CFA

MFB

Department of Environment and Primary Industries Victoria

SES

ESTA 000
Saving Time Saving Lives

1 Table of Contents

1.1 List of Figures

2 Executive Summary

The Information Interoperability Blueprint describes the future vision for information interoperability and decision support in the context of emergency management in Victoria. It recognises that cultural, organisational, political, legislative and technological reforms are all necessary to ensure a future where all decision makers have access to the information and tools they need, in a timely fashion and tailored to their context, to make the best possible decisions.

The technological platform required to deliver the outcomes detailed in the Information Interoperability Blueprint is defined as the Victorian Information Network for Emergencies (VINE). This document describes the technical architecture and high level roadmap for VINE.

As the name suggests, VINE is more than a single system, but rather an information network for all stakeholders before, during and after emergencies, including for the first time the community and private sector as critical participants in the process. The vision for VINE is to enable: a) all information relevant to emergency management to be gathered and integrated, b) the combined value-added information to assist all decision-makers, and c) information as well as decision-support tools to be made conveniently available to all stakeholders without unnecessary delays.

To comprehend VINE it is important to understand its aim. Although much of the discussion around VINE is about information, fundamentally VINE is about helping people make the right decisions before, during and after an emergency. Information is central to this goal, but it is not the ultimate end. What is VINE?

VINE is the collection of systems, standards and tools necessary to enable everybody affected by an emergency to make the best possible decisions for their current role and situation.

With this goal of decision support in mind and after significant consultation and reflection, it was concluded that VINE needed to provide the following functionality:

Unified identity and role management. At present, the identity of users is highly fragmented across systems: a single person may require up to 46 logins to perform their role at the State Control Centre. Not only does this present a burden to users, but it also makes it very difficult to combine information and present it to people in a unified way that enhances their ability to make decisions. It is a goal of VINE to gradually replace the current situation with a unified platform for identity and role management. This will serve to safeguard privacy and security, enhance coordination, improve accountability, and allow decision support that is customised according to users' preferences and roles.

Agreed standards for data representation and interpretation. In order to provide all stakeholders with a complete and up to date picture of any emergency scenario at any point in time, VINE needs to collect information from multiple sources and integrate it. This is possible only if all agencies agree on using the same set of standards for data representation and data exchange. In this way all stakeholders will be able to understand and consume any piece of information in VINE consistently, no matter if it was created by their organisations or others. Beyond the technical definitions of the standards a governance body needs to be put in place to make sure data standards are interoperable and extensible.

A flexible platform for data transport and storage. Information is at the core of VINE, therefore a platform for data transport and storage must be central to the VINE architecture. This platform should be flexible in terms of the type of information it can handle, and in the way that information is accessed. For data transport, VINE must support multiple communication protocols and take advantage of the publish/subscribe architectural approach for loosely coupled propagation of information.

Platform for innovation and foundation of an ecosystem. In order to engage with the community and to provide the best possible return of investment to the state of Victoria, VINE must not simply stand alone as the creation of those who built it. Instead, it must form the nucleus of a larger ecosystem by providing open and flexible access to all the services it provides. In this way, a broad constituency including ESOs, commercial organisations, community organisations, and universities can innovate around the core platform and continue to add value to it

3 Overview

This document describes at a high level the systems implementation goals that are necessary to build this functionality. This covers the set of systems and modules that need to be built, the way in which they need to be built, and the way in which all of these components will need to interact. In all cases, alternatives have been considered, and the document provides the reasoning used to arrive at the recommended approach.

It is planned that VINE will be constructed as a series of smaller, staged projects that build on each other until a point is reached where the core functionality of the system has been completed. This methodology is proposed for a number of reasons:

- Firstly, by building clearly-defined projects with an understood scope and well-specified benefit, it is less likely to succumb to building a large project with poorly understood costs, risks, and benefits;
- Secondly, clear value to the government and the community can be realised early in the process;
- Thirdly, by learning from the successes and failures of early projects, corrections and modifications can be made to the architecture of VINE in order to maximise the value it provides.

As well as providing an overall architecture for the system, this document also proposes a roadmap for a number of smaller projects that can provide early value while progressively building the full capabilities of the system—see Section 10.

A risk of such an approach is that complex core functions of the overall system are omitted from VINE as each small project defers it to some future time. Care needs to be taken that the development roadmap for VINE ensures that core functionality

is built methodically across the smaller projects such that the full vision for the system is realised over time.

It is not possible to cover off every technological issue in a high level reference architecture document. Furthermore, as components of VINE are designed and built there will be new insights and further detail that should be embodied into our documentation. Recognising this, it is intended that Version 1.1 of the Reference Architecture will be an ongoing work and the VINE Reference Architecture will be formally managed, modified (where required), and added to. This will support an ongoing architectural roadmap for all hazards and all agencies.

This is essential to ensure the overall IT investment is consistent with the 2013 Victorian Government ICT Strategy (released February 2013) and delivers the best possible emergency management outcomes for the State of Victoria.

3.1 Consultation Process

The following organisations were consulted with and feedback obtained:

- Country Fire Authority (CFA)
- Emergency Services Telecommunications Authority (ESTA)
- Victoria State Emergency Service (VicSES)
- Victorian Department of Sustainability and Environment (DSE)[1]
- Metropolitan Fire Brigade (MFB)

Presentations were also provided to and feedback obtained from:

- Victoria Police (VicPol)
- Ambulance Victoria
- Departments of Transport, Health and Human Services and Primary Industries
- Volunteer Fire Brigade Victoria

Emergency scenarios, drafted to ensure the High Level Reference Architecture addresses the correct needs (see Section 12) were proofed at a workshop where representatives from the following agencies were present:

- Victorian Fire Services Commissioner

[1] In April 2013 the DSE became part of the Department of Environment and Primary Industries

- Country Fire Authority (CFA)
- Metropolitan Fire Brigade (MFB)
- Victoria State Emergency Service (VicSES)
- Victorian Department of Sustainability and Environment (DSE) (Land and Fire Division and Water Division)
- Emergency Services Telecommunications Authority (ESTA)
- Victoria Police
- Ambulance Victoria
- Government of Victoria
- Environmental Protection Authority (EPA)
- Bureau of Meteorology (BOM)
- Victorian Department of Health (DoH)
- Victorian Department of Primary Industries (DPI)[2]
- Victorian Department of Treasury and Finance
- Municipal Association of Victoria (MAV)

[2] In April 2013 the DPI became part of the Department of Environment and Primary Industries

4 Background

Effective decision-making before, during and after emergencies can significantly improve emergency outcomes. How to provide the right information, at the right time, to the right person, is a pivotal task in the management of emergencies since it will significantly impact the quality of decision-making of first responders as well as of the general public. A comprehensive platform for emergency management can help people recognise an impending hazard, understand the risks and take effective steps to prepare beforehand, respond during an event and recover after the event.

With regards to emergencies, the requirements for management tasks are defined as follows:

"*These tasks require the combined expertise and resources of the emergency services, other government and private organisations, municipal councils and the people of the whole community.*" -Emergency Management Manual Victoria 2009[3]

Such a concept provides a basis for effectively dealing with disasters: a significant amount of cooperation and information sharing is required between government, emergency services and the community. A shared and integrated view of an emergency situation results in more coordinated action before, during and after an emergency. In order to achieve greater efficiency and interoperability between agencies, making all relevant information about an emergency available to all emergency services and the community is very important. However, as noted in the Floods Review (Commissioner, Review of the 2010-2011 Flood Warnings & Response, 2011), concrete barriers in organisational culture, communication, coordination, interoperability and information collation and sharing need to be

[3] Emergency Management Manual Victoria, October 2009, p.3

removed to realise this goal. The current state of emergency management and the information infrastructure require major reforms for improving the way information is shared and managed before, during and after emergency situation.

In order to enhance information management and communication before during and after an emergency, the Fire Services Commissioner (FSC) is leading with emergency services agencies a project that aims to develop a single structured platform for storing and sharing all relevant information about an emergency and making this information available to stakeholders and the community[4]. This project will build a unified and widely available open data platform that supports information exchange among different stakeholders in the phase of planning, preparing and responding to emergencies and helps stakeholders make the best possible decisions when emergencies occur. Furthermore, since this platform will provide an easily accessible and valuable repository of information relevant to emergency management, it will encourage research and innovation in emergency management and actively encourage contributions from the community.

Comprehensive emergency management requires participation by all emergency services and the individuals who understand their particular roles and responsibilities. Therefore, providing a resilient and open platform for sharing information between all stakeholders including the community will be the key requirements for dealing with disasters effectively and empower decision-makers to make better decisions.

[4] Achievements and Challenges: Annual Report 2011-12, p.11

5 Objectives

The fundamental objective of VINE is to support the vision for information interoperability set out in the Information Interoperability Blueprint, published by the Victorian Fire Services Commissioner. This Blueprint envisages a future in which all stakeholders in an emergency (emergency services, government and government agencies, NGOs, the private sector and, most importantly, the community) contribute to and work from a single, timely, coherent base of information: a common operating picture.

Information is essentially used for the purposes of making decisions and because VINE aims to provide a full, unified and timely view of information it is perhaps most accurate to represent VINE as a *decision support* platform. As a decision support platform, VINE must do more than simply provide common access to raw data. It must also be a place where information is digested, fused and correlated with other data, and enriched so that stakeholders are able to make informed decisions in a timely manner. This means turning road maps into evacuation models, satellite observations into bushfire risk assessments, and having the intelligence to recognise that a flood near a sewage-treatment plant requires the involvement of new agencies. VINE must therefore provide a flexible and extensible platform for a variety of processes that take raw data and value-add it into information that supports effective decision-making. It is also important to understand that as a platform VINE has to be able to interoperate with legacy systems. In other words, VINE should be able to integrate existing fire management information systems (e.g. Fireweb (DSE, 2012)) into its platform, so that the government can gain maximum benefit from a unified view of legacy systems and the new systems built on top of VINE.

The following are the key high-level objectives of the VINE program:

- Facilitate the acquisition of structured and unstructured (e.g. social media) real time data from multiple sources, including the community;

- Facilitate the sharing of data and information among different stakeholders in the emergency management community by using appropriate standards for information exchange and representation;
- Provide the maximum benefit through the federation and integration of existing systems and processes;
- Focus on delivering timely, relevant and tailored information to the community;
- Enable effective operational collaboration among ESOs and other stakeholders;
- Provide unified and flexible management of identities and roles. As discussed in Section 2, currently there is no central management of identities and roles between the systems operated by the multiple agencies;
- Provide complete support for information needs before, during, and after emergencies;
- Constitute the nucleus of an ecosystem that enables third parties to develop new capabilities and add value for all stakeholders;
- Support established and novel business processes to maximise the timeliness and quality of information and the effectiveness of actions;
- Provide a platform for powerful decision support using data fusion, analytics, and advanced modelling;
- Implement a governance model to oversee the process of setting appropriate standards and ensuring that they are interoperable between the agencies;
- Engage the community and the private sector as full participants in all stages of the emergency lifecycle; leverage the latent capabilities and resilience of the community and the private sector to improve the effectiveness of emergency management and response.

6 Architectural Approach

The reference architecture for VINE is designed to lay the foundation for a system that fulfils the objectives described above while creating an extensible asset for the people of Victoria that adapts and changes according to the community needs over the years. This will be achieved by separating the system into two well-encapsulated parts. A high-level view schematic of the architecture is shown in Figure 1.

The first part of VINE, representing the core infrastructure, is a relatively small and well-defined set of components that provide the critical functionality underlying the entire system. The second part of VINE is the modular services platform. It will allow further contributions and extensions of the capabilities of VINE throughout its lifetime.

The core and services platform parts are separated by a well-defined interface. In principle, the core infrastructure of VINE could be upgraded or replaced and, as long as the new implementation conforms to the same interface, the services implemented on the services platform will continue to function. This ability to swap out the core infrastructure has the potential to significantly extend the lifetime of the system, thus protecting the investment in a wide variety of cutting-edge services for emergency management in Victoria.

Core Interoperability Platform. The core interoperability platform is intended to provide a set of fundamental services upon which all further services can be built. Underlying all services in VINE is the access to information. As such, the core platform focuses on fundamental information services: a publish/subscribe **Event Bus** for the efficient propagation of real-time information; and a **Data Store** for the storage, manipulation, and querying of the many types of data that are relevant to the management of emergencies. Also as part of the core platform are the identity and role management service and several auxiliary services such as logging and

data transformation. See Section 9.2 for a more detailed discussion of the Core Interoperability Platform.

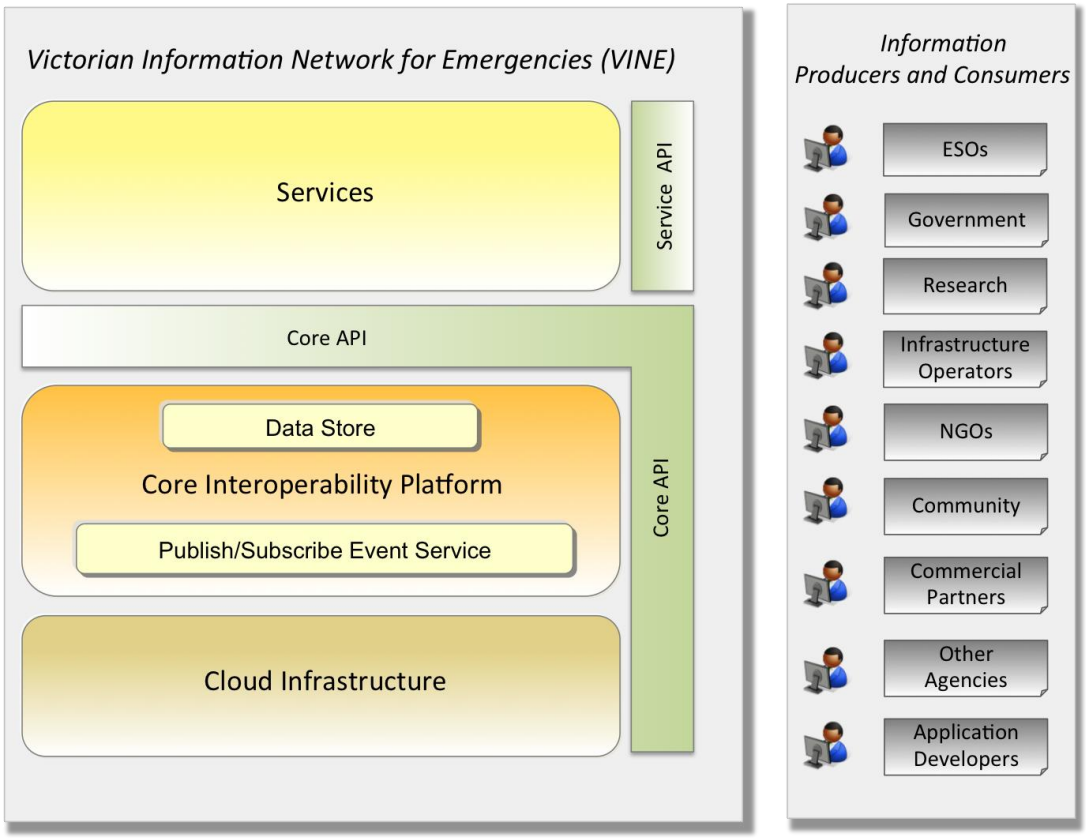

Figure 1 High-level VINE architectural components.

Services. The services layer consists of the components of VINE that are built with the aim of enriching the capabilities of the system but are not part of the core interoperability platform. It is separated from the core interoperability platform in order to allow the development and deployment by third parties of a variety of modules that interact with the platform using only the API. Because the platform is intended to be highly modular, the list of services in this layer is not fixed and is anticipated to grow as the state becomes increasingly sophisticated in its emergency management and integration with the community. Examples of services in this layer include tools for ingestion of external feeds, data validation tools, social media analytics tools, rule engines, modelling and forecasting tools, alerting and notification services, and many others.

API. The VINE API will provide the sole means of interacting with VINE for all users and services, including those services running as part of VINE within the modular services platform. The API will itself consist of a core API that allows access to and interaction with the core interoperability platform, and a set of API components that will provide additional access to functionality added by modules in the services layer. See Section 9.4 for a description of the API components.

7 High-Level Requirements

This section describes the set of high-level requirements in order for VINE to fulfil its mission as described in Section 6. It focuses primarily on requirements for the lower two layers of the platform illustrated in Figure 1: the Cloud Infrastructure and the Core Interoperability Platform, as well as the infrastructure necessary to allow services to be built within the Services Platform.

Reference to capabilities that are expected to be built as services within the services platform has been deliberately limited. This is because the architecture of VINE is designed to allow a great degree of modularity and expansion. VINE is planned to be a constantly evolving ecosystem of growing capabilities, therefore rather than defining the requirements for a particular number of services, this section defines the requirements for the services platform in general.

7.1 Functional Requirements

Functional requirements define specific capabilities that VINE must provide. They have been split into a number of groups, each specifying a set of specific requirements.

7.1.1 Identity and Authentication

Central to achieving interoperability and accountability within VINE is a secure, role-based identity management system. Each user will need to have a single identity throughout the VINE system that will be used to access any data or service. This behaviour will need to be consistent whether the users access data internal to their own organisation or provided by a different organisation.

Before assuming a unique identity within VINE, users will be required to authenticate to the system using one or more of a variety of best-practice authentication methods (e.g. strong passwords, tokens, biometrics etc.). Provisions

must be made to select those methods that prevent hacking or other types of attacks.

The following are the requirements for identity and authentication within VINE:

- Each user will have a single secure identity that they will use to interact with all features of VINE. Not only members of organisations but also members of the community will be able to register in order to authenticate and use VINE capabilities. Similarly, any member of the private sector (e.g. a contractor) who wishes to be accredited to provide services in emergency events will be able to register in order to authenticate themselves in VINE;
- Each identity will be associated with standard directory service metadata such as name, organisation, contact details, etc.;
- All actions taken within VINE will log the identity of the person who takes that action, enhancing accountability and post event reviews;
- Authentication can take place via a variety of factors (passwords, keys, smartcards, biometrics etc.) that can be customised and extended in accordance with the needs of various stakeholders and the level of security required;
- Where possible, Single Sign On (SSO) capability will be implemented within the system to minimise the number of times a user needs to provide credentials to verify their identity;
- Step-up authentication: for certain very sensitive data, an additional factor or stronger authentication may be requested prior to access. For example, accessing hospital records may require biometric authentication even if a user has rights to access this information and password authentication has already taken place.

7.1.2 Role Management and Authorisation

Interaction with VINE will be mediated by a role-based system for accountability and authorisation. At any given point in time users would have roles assigned; the roles will define their responsibilities and privileges within the system. As a unified record of the actions and responsibilities of each user at any point during the course of an emergency is retained accountability is enhanced.

VINE will be built on the principle that all information should be made as widely available as possible. Operating off a common base of information is critical to enabling full situational awareness and providing a common operating picture. However, in many cases it might be necessary to control access to certain actions or to sensitive pieces of information. Roles will be used to define the types and levels of capabilities users will have in VINE (i.e. to access data, take certain actions or decisions, assign resources etc.).

Role management module in VINE has therefore the following high-level requirements:

- Each user will have zero or more roles associated with their identity at any time;
- Each role will define the responsibilities assumed by the user, and it will be associated with a set of permissions in the system (i.e. what a person can access and do);
- New roles can be added and defined by VINE's administrator;
- Roles can be assigned to and unassigned from a given user at any time;
- All events relating to the change of a user's role status will be logged in a persistent log such that any user's roles at any given time can be reconstructed;
- A user's level of access to VINE will be defined by the union of the access rights of all roles currently assigned to the user;
- Roles may also be assigned to services, so that a service (such as a weather modelling service) may access the data and upstream services upon which it depends.

VINE's role-based access control system has the following requirements:

- Access to any data, facilities and services within VINE will be mediated by VINE's access-control layer;
- The access management system will support a range of data access levels, appropriate to different use-cases. For example, the system may allow a reconstruction agency to access the exact address of a damaged property, whereas for privacy reasons researchers would only be able to retrieve the number of damaged properties in a particular region. Or, access to the personal details of disabled persons in a community may be tightly controlled, but the total number of disabled persons in the community could be made available to a much broader set of users;
- Access rights will be only associated with roles and not directly with users;
- A change to users' role assignments will be immediately reflected in their level of access to VINE;
- Access-control system should support obfuscated access to data where required, for example by changing the precise coordinates of an event to the name of the region in which it is taking place, in order to protect the exact location when needed;
- A low-privilege default role will be assigned to unauthenticated users. This role will allow them to access the critical emergency information available in VINE but it will not allow access to other services in the system that require authentication.

7.1.3 Real-time Event Propagation

VINE must provide support for propagation of event messages in real-time or near real-time. Much of the information necessary for emergency management, especially during an incident, can naturally be modelled as events. For example, all of the following pieces of information can be seen as events:

- The reporting of an incident;
- Despatch of units to a particular location;
- The issuing of an alert or warning;
- The closure of a road;
- Updates to the status of an incident;
- The completion of a house, school or nursing home evacuation;
- Weather updates.

At all times, but especially during an incident, receiving relevant information about events in a timely manner is essential to coordination and effective decision-making. For this reason, VINE will feature a flexible publish/subscribe capability for the real-time transmission of event information to all interested, authorised parties. Such a system will allow for the near-instantaneous propagation of events from the person or agency that shares event information (the publisher) to anybody who is interested in events of that type (the subscribers) without any direct agreement or interaction between the parties in question.

The following are the high-level requirements for real-time event propagation:

- Events in the system are represented as a collection of hierarchical topic trees, where each topic represents a particular type of event.
    - Each topic in a tree may have zero or more subtopics;
    - Each topic (except the root topic for a tree) has exactly one parent topic;
    - A Common Information Model (CIM) (DMTF, 2012) is specified for each topic tree. All topics in that particular tree will comply with that CIM;
    - Decomposition of the tree into subtopics is domain-dependent and beyond the scope of this document. A common decomposition is geographical (see example in Figure 2), but other decompositions (such as organisational) may also be applicable. In Figure 2, the topics are divided into a two-level geographical hierarchy.
- Authorised users can publish new events to one or more topics;

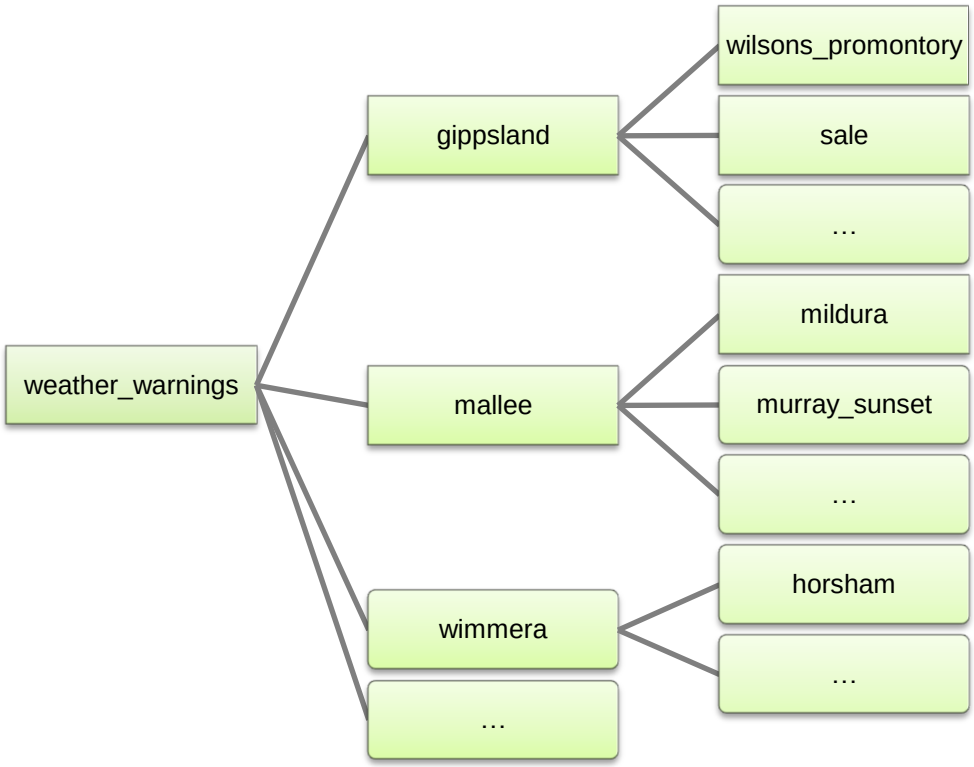

Figure 2 An example topic tree for weather warnings.

- Authorised users can subscribe to one or more topics;
- When subscribed to a topic, users will receive new events as soon as they are published to that topic
    - If subscribed to a topic that has subtopics, the user will also receive events that were published to these subtopics;
    - Subscribers will be able to specify filtering rules for each subscription, such that only certain events in the topic to which they are subscribed are delivered;
    - Subscribers will be able to use different communication protocols to maintain their subscription to a topic, including at least one that provides push notifications so that polling is not required.
- Events are maintained in the system until they have been successfully delivered to all users who were subscribed to the topic at the time of publication.
    - Optionally, events may be assigned with expiration dates, such that they would be removed from the topic after the expiration time.

Figure 3 shows an example of Publish/Subscribe message flow.

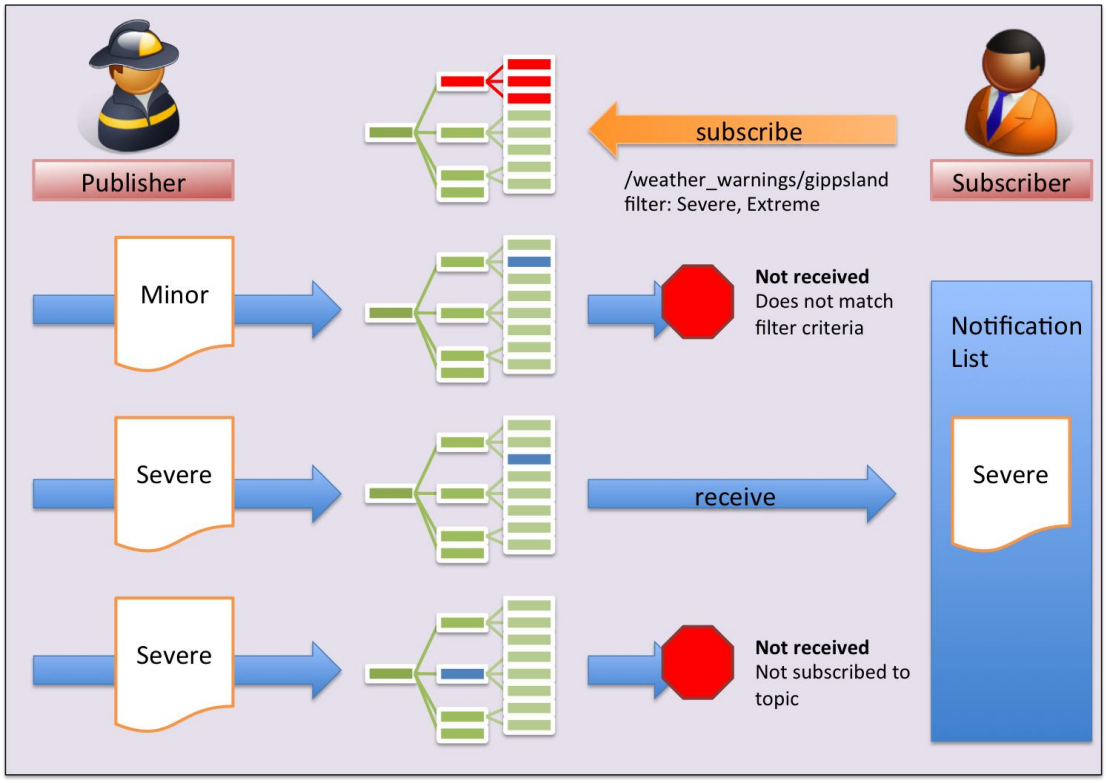

Figure 3 Sample Publish/Subscribe subscription flow.

7.1.4 Data Management and Access

Effective decision-making before, during, and after an emergency requires coordinated and timely access to a wide variety of potentially relevant data and information. Therefore, stakeholders in an emergency may need access to some of the following information:

- Maps and map overlays;
- The location of emergency service appliances and personnel, private sector appliances and personnel, vulnerable people and other members of the community;
- The duty cycle and status of appliances and personnel, for example the number of hours someone has been on duty, the level of fuel in a grader or whether someone has been provided with food and drink;
- Modelling and forecasts for events such as fires, floods and chemical spills;
- Preparedness plans;
- Event information from earlier emergencies;
- Vulnerable persons registers;
- Hazmat information;
- Resource deployment;

- Weather forecasts;
- Road conditions;
- Multimedia data (e.g. images, video);
- Social media inputs;
- Sensor network information (e.g. smart meters or temperature sensors on smart phones).

The list above is not intended to be comprehensive, but to illustrate the diversity of data that may be encountered in the course of emergency management. For VINE to fulfil the vision of a common operating picture it needs to mediate access and manage a wide variety of information pertinent to the management of emergencies. As a wide variety of access patterns must be anticipated, the system must also provide a rich application programming interface for querying and retrieving data. The following are the high-level requirements for data management and access:

- VINE will make it possible to discover and exploit all sources of data within the system without a need for direct communication between the producer and consumer of the data;
- The system will support transactional interactions with data in order to ensure consistency and accountability for key pieces of data. For example, the delegation of a role from one person to another need to take place in such a way, that roles and responsibilities in the system are clear at all times;
- The system will expose a uniform interface for data access, regardless of the physical location of that data;
- Both read and write access to the data will be mediated by the access control capabilities of VINE;
- The system will have the capability to define and enforce standards for the representation of data;
- Access to data will be supported by querying mechanisms that have relational as well as geospatial (GIS) capabilities (see Section 9.3.2.2 for a discussion of GIS features in the context of VINE).

7.1.5 Operational Support Capabilities

VINE is envisaged as a platform operated 24/7, providing support for all phases of the emergency management lifecycle, including prevention, planning and training. However, it is critical that it provides adequate operational support for the period immediately before, during and after an emergency, especially when dealing with a large, multi-hazard crisis. During such periods, the volume of alerts and other types of data exchanged reaches the peak; at the same time, the coordination of all emergency services required to be supported by VINE is the most critical.

The following capabilities will be provided in VINE, in order to assist with operational information flows and coordination:

- VINE will have integrated support for the creation of business process workflows that define a sequence of rules and steps for a given process. For example, certain events might require to be validated by an individual in a role of responsibility, prior to the information becoming available to subscribers.
    - It must be possible to define workflows across a particular data by the custodians of that data. For example, the Bureau of Meteorology must be able to define workflows around weather forecasts in VINE.
    - At all times, the workflow system must interface seamlessly with other components of the system, in particular the event bus.
- VINE will support existing standards and protocols for incident management, such as AIIMS (AIIMS, 2012).

7.1.6 Universal Access Interface

VINE is designed to enable the development of an ecosystem around emergency management. The intention is for many stakeholders (ESOs, community organisations, research organisations, the private sector, individuals) to be able to access and submit information in a flexible way, from many devices and locations, and to be able to build novel and useful tools to contribute to and make use of the information in VINE.

In order to meet its ambition of creating an ecosystem, VINE must provide a flexible and powerful interface to the data and services that it offers. This interface must not be restricted to certain systems, devices or networks, and it must not require a high level of technical proficiency or domain expertise in order to use. In order to build such a universal access interface, the following requirements must be met:

- VINE will expose an interface that allows access to all data and services contained within and flowing through the system;
- The interface will use standards-based communication protocols and be accessible via the Internet;
- All elements of the interface will be accessible to any users, including those who have not authenticated to the system
    - However, authorisation for access to data and services is mediated by the access-control layer. Those users who attempt to use the API to access data to which they are not authorised will receive an appropriate error message.
- The interface will have a real-time access element whereby subscribers to event topics are able to receive events as they occur, without repeatedly querying the interface.

- The core interface will expose of the following set of capabilities:
    - User authentication;
    - Management of event subscriptions;
    - Access to events in topics to which the user is subscribed;
    - Publication of new events;
    - Querying capabilities across all datasets in the system, including geospatial queries;
    - Mapping and location services;
    - Workflow interactions, including push notification of outstanding workflow tasks;
    - Basic interaction with services and modules within VINE; for example starting a new process;
    - Flexible write access to all datasets in the system;
    - Data catalogue;
    - User directory;
    - Role and user management;
    - System administration capabilities.
- Extension of the interface capabilities will be possible via a plugin model.
    - Plugins will be necessary to provide the flexibility to interact with new system services as they become available;
    - Once developed and deployed, the services provided by the interface plugin are available to all users (with the appropriate privileges) to access the underlying data and services.

7.1.7 Modular Services Platform

As mentioned previously, VINE is intended to serve as the foundation for a rich ecosystem of emergency management tools and services. The class of such services is potentially very broad and limited only by the different uses to which VINE can be put. Some examples may be a service for analysing social media data, a system for optimising the placement of resources, a fire-spread model or real time management of an evacuation.

As an ecosystem, many of these services may be initially built and operated externally from VINE, using only the interface. However, for an operational reliance to be allowed to develop on any given service, it must operate in a controlled and reliable environment that has the required level of redundancy and scalability. For

this reason, operationally critical services must run within VINE itself. This give rise to the concept of a modular service platform, whereby services can be developed externally (for example, by ESOs or research organisations) and can then be assimilated into VINE and operated from within the system.

Following are the requirements for the provision of a modular services platform:

- Modules will be executed within a well-isolated and capable compute environment running within VINE. This means that failure of a service would have no effect on any other services
    - The compute environment will provide a clear set of capabilities through a fully-defined interface;
    - The compute environment will be made available for installation outside of VINE, so that application development and research can take place independently prior to deployment of service on VINE.
- In order to improve security compliance and ease the transition between external and internal modules, modules will only have access to the system via the same API made available to external users;
- Basic interactions with modules (for example, starting and stopping processes in a module) will be provided by the core VINE interface;
- In order to provide a richer and more customised level of interaction with the specific domain of the module, it may be necessary to write interface plugins;
- All modules will be modelled as resources, and interaction with them will be subject to access-control in the same way as any other resources within VINE.

7.1.8 Alerting and Notification

VINE's universal data access interface will allow any user with an appropriate level of authorisation to receive alerts and notifications in a timely manner, simply by subscribing to the appropriate topics using the publish/subscribe event service. However, there will often be a need to send notifications (especially alerts) to people and agencies who are not full participants in the VINE ecosystem, as well as to devices that do not have the capability of connecting directly to VINE. For this reason, it is necessary for VINE to possess full alerting and notification capabilities to engage with agencies and the community across all available channels. The following characteristics are required:

- Full integration with VINE, especially with the event bus. The alerting system should be able to define rules based on incoming events. For example, it may subscribe to evacuation warnings on the event bus and issue an all-channels alert to people in the affected area;

- Support for alerting across multiple channels: telecom recorded message, SMS, sirens, radio, electronic signs, email etc.;
- Specific accessibility support to ensure that vulnerable community members do not miss out on alerting and notification. For example, consideration must be given for alerting mechanisms that are suitable for hearing-impaired members of the community;
- Support for multiple languages. The system should facilitate the issuing of alerts in multiple languages in order to reach members of the community who are not native English speakers. This can be achieved through a combination of automated translation techniques and human-sourced translations, for example through engaging community leaders to take part in a translation process.

7.1.9 Logging, Audit and Reporting

Accountability is fundamental in the management of emergencies. VINE's role-based identity management capability will record what users of the system were assigned a particular role at any given time, and what actions they took in that role. As such, it is well placed to assist in maintaining the accountability of all those involved in the management of an emergency. To assist with this, VINE must have logging, audit, and reporting capabilities, as follows:

- All role assignments and reassignments are permanently logged such that the set of responsibilities assumed by users at any given time can be reconstructed;
- All actions taken by logged-in users of VINE (including the action of accessing certain information) will be permanently logged such that a trace of actions taken by any given user can be reconstructed;
- All information within VINE will be associated with a timestamp defining when actions on that information were taken, such that all timestamps are consistent relative to each other. With this information, the relative timing of events within the system can be reliably determined;
- Timestamps for when a piece of information was created, transmitted and received at VINE (thereby being usable);
- The system will be capable of producing a variety of reports that allow for an ongoing process of accountability, as well as aiding forensic investigations following major disasters;
- The logging system will have a flexible interface that is supportive of audit and forensics operations on the log data;
- The logging and reporting capabilities in VINE will be compliant with common standards of audit for such systems.

7.2 Non-Functional Requirements

Non-functional requirements specify necessary general characteristics of the system defining expectations around its performance, maintainability, accessibility, and so forth. This section lists the high-level non-functional requirements for VINE, and discusses the measures necessary to meet each of these requirements.

7.2.1 Availability

Over time, VINE will play a critical role in many processes related to the management of an emergency - before, during and after. As such, it must be deployed in a reliable, high-availability configuration. Availability is not as straightforward as a measure of system uptime; it describes the degree to which users of the system are able to access the functionality of the platform. Availability can be affected by network issues, excessively high latency, service degradations due to maintenance, and other issues. To achieve high availability, the following principles must be adhered to:

- Instrumentation and supervisory control. Extensive monitoring systems are required, to provide ongoing statistics on system's availability and to provide an early-warning system that can be used to improve overall availability. Effective round-the-clock system administration capacity will be necessary to allow for rapid response and recovery from availability issues;

- Redundancy. Any component, no matter how reliable, has some chance of failing. Planning redundancy into every level of the system, although not inexpensive, is a critical aspect of enabling high availability by ensuring that single component failures no longer have the ability to take down the entire platform;

- Loose coupling. Isolating subcomponents of VINE such that the failure of one component does not lead to failures of others will reduce the impact of outages and improve overall availability. For further information on loose coupling and modularity, see Section 7.2.5.

7.2.2 Scalability and Elasticity

It is intended that VINE will begin life providing relatively limited and targeted functionality, and that it will grow over time, adding new capabilities and users. It is not possible to anticipate the precise load, quantity of data, or volume of traffic that VINE will have to handle over its lifetime. For this reason, VINE must be scalable. It is important to design VINE in such a way that its infrastructure can be scaled to handle larger workloads than the ones for which it was originally intended, yet without having to massively over engineer or overprovision the system at the early stages. Building VINE implies the following design considerations:

- Distributed systems paradigm. It is possible to grow capacity in a distributed system in a 'scale out' fashion by adding additional nodes, whereas the only alternative for a single-node system is to 'scale up' to a more powerful

system. Scaling out is generally more economical and less disruptive than the alternative of scaling up;

- Bottleneck avoidance. The overall performance of a system is governed by that of its slowest component. Large distributed systems often lose the ability to scale because of reliance on a single unthreaded process, or on a single database table. Such bottlenecks must be avoided in the design because they cannot be solved by adding extra resources to the system.

- Modularity. Increasing complexity and interdependency of software often acts as a barrier to scalability of systems. Modular system design is important in avoiding this issue; for more on modularity, see Section 7.2.5.

While scalability deals with the ability of VINE to grow its capacity in the medium-to-long term, elasticity describes a requirement that the system be able to adapt to variation (and, in particular, to spikes) in load over a much shorter timescale. It is easy to anticipate that VINE will be put under the most stress at precisely those times when it is needed most - when there is a major emergency taking place. Thus, it is critical that the architecture of VINE makes it possible for additional capacity to be added to and assimilated by the system at very short notice. The alternative to building VINE for elasticity is to permanently provision the system with sufficient capacity to handle the most extreme anticipated load spikes. However, this is not an economical alternative as it means leaving a vast quantity of resources idle for a majority of the time. Building VINE as an elastic system requires the following:

- Distributed systems paradigm, as above. Computational elasticity is largely premised on the possibility of scaling out to additional nodes as load increases (although in many cases scaling up to a more powerful node remains possible);

- Dynamic systems for provisioning, replication and load balancing. The basic mechanisms for allowing resources to be rapidly allocated and released are typically available as part of a cloud computing environment;

- Pool of computational resources shared amongst a diverse range of services. Elasticity is premised on the ability of a service to quickly provision and assimilate additional computational resources when necessary, but those resources must be available. Elasticity is most effective when a pool of resources is shared across a range of services that are unlikely to all see spikes in load at the same time. Under such circumstances, one service seeing a load spike is able to provision resources relinquished by another service experiencing a relatively light amount of load;

- Quality of Service / job priority. In some cases, services sharing a pool of resources will attempt to allocate more resources than physically exist. When such circumstances arise it is necessary to have a mechanism in place for ensuring that the most critical services gain access to the

resources they require at the expense of less important or auxiliary services[5].

7.2.3 Low Latency

As outlined in the Blueprint, one of the key aims of VINE is to improve the current situation where information shared by other organisations is often quite out of date by the time it is received and processed. In keeping with this ambition, it is a requirement that VINE offers low latency. It is difficult to generalise a definition of low latency for a system intended to be as diverse as VINE beyond keeping it in mind as a key requirement, but some guidelines for core parts of the system include:

- Time from publication of an event until it is available to subscribers in VINE should be under 10 seconds, in order to ensure that any delays in information transmission are not so long as to substantially affect decision-making;
- Latency for basic API calls (where the call invokes no significant computational burden) should be below one second, in order to ensure that responsive interactive applications can be written that utilise the API;
- When ingesting data feeds from external services, frequency of polling should not exceed one minute so that consumers of data from VINE can be confident that they are getting the latest information from the system.

7.2.4 Accessibility

VINE is intended to serve all members of the community, and especially those that have an increased level of vulnerability in the face of an emergency. In many cases, it will be the most vulnerable members of the community who will have the greatest difficulty in getting support from a system such as VINE. Accessibility issues include, among others, having a physical or intellectual disability, difficulty to work with computers, no access to Internet, a weak command of English or simply being uncomfortable interacting with an online service such as VINE.

In order to serve all members of the community, provisions must be made within VINE to maximise accessibility, including:

- Ensuring that critical websites and applications adhere to standard accessibility rules[6], that allow people with various limitations or impairments to interact comfortably with them;

[5] An example of such a mechanism is the MapReduce FAIR Scheduler's ability to pre-empt or kill tasks in favour of higher priority tasks (Zarharia, Borthakur, Sen Sarma, Elmeleegy, Shenker, & Stoica, 2009).

[6] For example, The W3C's Web Content Accessibility Guidelines (http://www.w3.org/TR/WCAG10/)

- Encouraging leaders in immigrant communities to engage with the VINE program and ensure their constituents are able to stay informed about emergency planning and response, for example by building a community-language website or by setting up a telephone tree[7] to spread information shared by VINE during an emergency;

- Bridging VINE with more traditional communication mechanisms (such as the voice telephony network) and media (such as radio and television) so that computer literacy or Internet access are not prerequisites for benefiting from information in VINE.

7.2.5 Modularity

VINE is envisaged as an information interoperability platform for emergency management. That means that it will consist of a diverse range of services that may be contributed by different stakeholders, and that its functionality will expand over time. Therefore it is important that VINE pursues a loosely coupled, modular architecture.

Modular architectures exhibit the following characteristics:

- Well-defined components. As opposed to an integrated or monolithic system, a modular system can be described as an aggregation of distinct components, each of which plays a well-delineated role. Ideally, each component should be responsible for performing only a single service;

- Narrow, clearly defined interfaces. Interaction between a module and the outside world, or between two modules, must take place using clearly defined protocols and interfaces only. These interfaces should be designed to be stable (change rarely if at all), narrow (the bare minimum necessary to expose the desired functionality), and semantically coherent;

- Isolation between components. A failure of one component should not cause failure in other components;

- Limited interdependence. Although components are able to interact, and can therefore come to depend on each other for their correct operation, uncontrolled interdependence reduces isolation between components and makes the overall system less robust. While in many cases some level of interdependence is unavoidable, both system governance and module design must work to minimise and regulate dependencies.

7.2.6 Security

As it evolves, VINE is expected to play an important role in an increasing number of emergency management activities. Furthermore, it will be exposed to the Internet,

[7] http://www.dse.vic.gov.au/effective-engagement/toolkit/tool-telephone-trees

making it potentially accessible to a broad range of malicious parties. Therefore, an effective approach to securing the system is essential.

Key components of an effective overall security policy include:

- Vigilant systems administration. Those responsible for operating VINE's infrastructure must ensure that they stay up-to-date with security advisories and that they apply security updates and patches to VINE's software components with alacrity;

- Robust authentication. Many systems are compromised as a result of weak authentication practices. Examples of such flaws include poor enforcement of password policies, storing passwords in clear-text or using weak or unsalted hashes, and failing to use adequately secured connections for authentication. VINE's authentication system must be built and deployed in accordance with industry best practices so that all stakeholders can have confidence in the integrity of the system and of their personal accounts and information;

- User education and the principle of least privilege. Many system compromises take place not through technical exploits but rather through what is known as 'social engineering', where users are manipulated or tricked into revealing information about the system or their account—the online phenomenon of phishing is one such example. In order to protect against such attacks, education of users with respect to correct practices and protocols is essential. To further protect VINE the 'principle of least privilege' should also be followed. This principle simply dictates that each user in the system should have the bare minimum level of access to the system that is necessary to perform their role, meaning that no unnecessary information is given away should that user's account be compromised;

- Security-minded programming practices. Software deployed as part of VINE must be built in accordance with secure programming practices (see (Seacord, 2006), (Swiderski, 2004)) that prevent vulnerability to common attacks such as code injection, cross-site scripting (XSS) and cross-site request forging (XSRF), buffer overflows, etc.;

- Active security processes. In addition to the passive security practices described above (which can be thought of as 'walls' against intruders), an effective security environment also requires that active security measures ('security guards') are put in place. These include systems for virus detection and elimination, intrusion detection, anomaly detection, Denial-of-Service (DoS) detection and countermeasures, and so forth. Although such measures are not 100% effective, they do provide an additional line of defence against certain types of security breach.

7.2.7 Privacy and Integrity of Access

As mentioned in the BLUEPRINT, VINE is intended to align with a new direction in emergency management of making information open by default. Despite this, it is

recognised that much of the information to be shared through VINE is sensitive and that access must be carefully controlled. Special attention must be paid to the question of privacy with regards to information stored or propagated by or about private individuals. In addition to general security and information protection practices described above, specific measures to be taken in order to safeguard user privacy include:

- Provision of and adherence to a clearly worded privacy policy;
- Adherence to special procedures when handling sensitive or Personally Identifying Information (PII)—for example, having two people in the room at all times when dealing directly with such information;
- Provision of anonymisation tools so that PII is not leaked when accessing other legitimate information. For example, when accessing information about people with impaired mobility in a nursing home for the purposes of evacuation planning, names, ages, and form of impairment should be purged from the data before making it available to the requestor.

7.2.8 Data Preservation

As a repository of valuable information, VINE must ensure that the data is well preserved at multiple levels so that it remains available to the community:

- Online replication and redundancy. By using replication to ensure that data is not stored on only a single device (a common practice is to replicate data to three independent devices) or accessible via a single server, VINE can tolerate a physical failure in its underlying infrastructure without severing access to the data on the platform. VINE should also be redundant across multiple sites, so that a site-wide failure (a power outage for example) does not interrupt service;
- Offline & offsite backup and restore. Offline backups are necessary to protect against disruptions such as large-scale hardware failure (e.g. caused by an uncontrolled power surge or data centre fire) or software-induced data corruption. A well-planned and diligently executed offsite data backup and restore process is essential to ensuring the long-term integrity of data stored as part of VINE;
- Data archiving. Over its lifetime, VINE will accumulate a wealth of valuable information that will be of value to posterity. An archiving plan is necessary to safeguard VINE's data and make it accessible to future generations, even once the system itself has been decommissioned.

8 Data Standards

This section describes a number of standards developed by the international standardisation bodies, to represent and exchange data in emergency scenarios. Then it looks at emergency and alerting systems developed in various countries throughout the world, to get a clear picture of the current standardisation efforts in emergency management. Finally, it describes the existing efforts and applications in the Australian emergency services context, followed by a number of findings and recommendations in the context of VINE.

8.1 International Standards for Emergency Data Exchange

OASIS is one of the main international bodies in standardisation, which aims to "drive the development, convergence and adoption of open standards for the global information society" (OASIS, 2012). OASIS has developed many emergency related standards, which are currently implemented and used throughout the world.

8.1.1 The EDXL family

EDXL is an OASIS initiative to create an integrated framework for a wide range of emergency data exchange standards for operations, logistics, planning and finance in the context of providing efficient emergency services.

This family of standards was created and it is currently maintained by OASIS via specialised emergency management working groups (OASIS EM TC, 2012). The members of EDXL family, in the chronological order of their approval by OASIS are:

8.1.1.1 EDXL-DE (Emergency Data Exchange Language Distribution Element) – May 2006

This is mainly a wrapper element that allows flexible message distribution. It can contain any XML-based or non XML-based payload (OASIS EDXL-DE, 2006).

The standard is currently at version v1.0 (second public review held in May 2012 for version v2.0). It has the ability to specify the sender and recipient details and a target area or specific addresses for recipients. It can also track distribution status (new, update, request, response, etc.). A sample EDXL-DE file is shown in Appendix A.

The rest of EDXL family members, described below, are XML-based recommended payloads for EDXL-DE.

8.1.1.2 EDXL-HAVE (Emergency Data Exchange Language Hospital Availability Data Exchange) - November 2008

EDXL-HAVE specifies an XML document format that allows the communication of the status of a hospital, its services, and its resources (OASIS EDXL-HAVE, 2009). These include bed capacity and availability, emergency department status, available service coverage, and the status of a hospital's facility and operations.

The standard is currently at version v1.0 with a first public review held in June 2012 for version v2.0. A sample EDXL-HAVE file is shown in Appendix B (contained in an EDXL-DE message wrapper).

8.1.1.3 EDXL-RM (Emergency Data Exchange Language Resource Messaging) - November 2008

This standard XML format is used to coordinate requests and responses to requests between systems in emergency situations (OASIS EDXL-RM, 2009). There are 16 message types that can be described using EDXL-RM. They support communication requirements for allocation of resources during the entire lifecycle of the emergency incident. This includes preparedness, pre-staging of resources, initial and ongoing response, recovery and release of resources.

The standard is currently at version v1.0. A sample EDXL-RM file is shown in Appendix C (contained in an EDXL-DE message wrapper).

8.1.1.4 CAP (Common Alerting Protocol) v1.2 – July 2010

CAP is a relatively simple but generic format used to exchange alerts and notifications (e.g. public warnings) over different networks (OASIS EDXL-CAP, 2012). It provides an open, non-proprietary XML format that does not address a particular application or communication method.

Although not directly part of the EDXL family of standards, CAP is very often included as part of the EDXL-DE payload for exchanging alerts and notifications.

CAP offers a number of capabilities such as flexible geographic targeting (geographical points or areas of coverage for the message), multilingual and multi-audience, customised effective times, onset and expiry date for the messages, facility to refer the receiver to a separate, related, digital image or audio file etc.

The standard is currently at version v1.2 (approved in July 2010) and contains a number of enhancements on the initial version v1.1 (October 2005). Work on CAP v2.0 has been planned to start this year (2012).

A sample CAP v1.2 file is shown in Appendix D (contained in an EDXL-DE message wrapper).

8.1.1.5 EDXL-CAP v1.2 Australia (AU) Profile Version 1.0 – April 2012

In April 2012 OASIS approved EDXL-CAP v1.2 Australia (AU) Profile Version 1.0 (also known as CAP-AP (OASIS EDXL-CAP-AU, 2012) or CAP-AU-STD (Australian Government, 2012)). This is not a separate standard for Australia, but a detailed interpretation of the OASIS CAP v1.2 standard as described above, with a number of requirements and recommendations for most of the fields in CAP v1.2 schema that are necessary to meet the needs of the Australian Government*.

Event codes utilised in CAP-AU messages needs to match the Australian Event Code List (AUEventLIST) (OASIS , 2012). An alert message conforms to CAP-AU if it is valid according to CAP v1.2 schema *and* meets all additional mandatory requirements from CAP-AU specification document.

A sample CAP-AU file template is shown in Appendix E.

8.1.1.6 EDXL-TEP (Emergency Data Exchange Language Tracking of Emergency Patients) - June 2012

This standard describes the standard XML schema for the exchange of emergency patient and tracking information from the point of patient encounter to patient admission or release (OASIS EDXL-TEP, 2012). EDXL-TEP allows patient tracking across the continuum of care and provides real-time information to care facilities in the chain of care and transport. Recipients of EDXL-TEP messages could be emergency departments, hospitals, incident command centres etc.

The standard is in a Working draft format at version v1.0. A sample EDXL-TEP is not available at the moment from OASIS EDXL-TEP Technical Committee.

8.1.1.7 EDXL-SitRep (Emergency Data Exchange Language Situational Reporting) – Public Review Draft 02 - May 2012

This standard (OASIS EM TC, 2012) describes a set of standard reports and elements that can be used for data sharing among emergency information

* *Further details on the Australian standardisation efforts are given in Section 8.3.*

systems, and that provide incident information for situation awareness on which incident commanders can base decisions.

The standard is in a Working draft format at version v2.0. A sample EDXL-SitRep is not available at the moment from OASIS EDXL Technical Committee.

8.1.2 Other standards for emergency management

Along with EDXL family there are a number of other standards that have been designed to facilitate information sharing in emergency scenarios. Some of them are:

8.1.2.1 NIEM (National Information Exchange Model) – February 2005

NIEM was designed to develop, disseminate and support information sharing processes across various domains such as Justice, public safety, emergency and disaster management, Intelligence, immigration etc. It was created in 2005 by the US Department of Justice and US Department of Homeland Security to leverage and extend the Global Justice XML Data Model (Global JXDM) (US Department of Justice, 2012) and to facilitate timely and secure sharing of information about people, places, materials, events etc. Global JXDM is currently at version 3.0.3 (September 2012) and NIEM 3.0 is planned to be released in autumn 2013 (US calendar time).

The basic building block in NIEM is a data component that describes a real-world object or concept using an XML schema. NIEM also works with concepts of Information Exchange Package Documentation (IEPD) and Information Exchange Package (IEP). IEPD contains schemas, description, a catalogue of artefacts, metadata etc. When IEPD is populated with real data from data sources, it becomes an IEP, which is the actual XML document transmitted / exchanged.

8.1.2.2 TWML (Tsunami Warning Markup Language) and CWML (Cyclone Warning Markup Language – 2006

These two standards were developed by National ICT Australia (NICTA, 2012) as a first attempt to define a structured semantic data model for tsunami bulletins and cyclone advice. Among multiple benefits, both allow for less ambiguity than in pure textual messages, improved consistency across different warning centres and improved opportunities for computer processing.

TWML (NICTA TWML , 2006) and CWML (NICTA CWML, 2006) can be used as payloads for EDXL-DE or together with CAP for routing the message to the right recipients and for prioritisation of alerts.

Both these standards are Working Drafts, at version v1.0. Their specifications have no formal standing in any Standards consortium or related group.

8.1.2.3 (JMX) Japan disaster prevention information XML – May 2010

JMX is a project of the Japan Meteorological Agency (JMA) (JMA, 2012), supported by W3C working group on XML standardisation (W3C, 2012) and in coordination with governmental organisations and mass media. The latter have a critical contribution in distributing disaster prevention information. The existing CAP protocol cannot represent and transmit effectively the detailed and much specialised information needed for disaster prevention in Japan. Therefore JMX was proposed as a local standard to cater for the domestic information needed by authorities in order to ensure people safety. The main goal of JMX is the ability to send multiple types of information in a single bulletin, from observations to forecasts.

8.2 International Efforts in Alerting and Emergency Management

8.2.1 Emergency management systems

This section lists some of the larger initiatives in implementing standards-based emergency management systems. The list is therefore not exhaustive and other initiatives exist throughout the world. The map below gives a visual representation of the spread of organisations and projects using CAP and other standards for applications and services in emergency management.

Figure 4 Emergency management systems throughout the world.

8.2.1.1 UICDS (Unified Incident Command and Decision Support) – US

UICDS (UICDS, 2012) is an initiative of the Department of Homeland Security in the US, for information sharing between commercial, government, academic and volunteer technology providers to support the National Incident Management

System. UICDS relies on emergency applications as sources of, and visualisation for, relevant incident data. These sources provide some of their data to UICDS, which in turn publishes it for subscribed applications.

Among the standards currently incorporated in UICDS are:

- NIEM (National Information Exchange Model) - see section 8.1.2.1
- CAP (Common Alerting Protocol) – see section 8.1.1.4
- EDXL-DE (Emergency Data Exchange Language Distribution Element) – see section 8.1.1.1
- EDXL-RM (Emergency Data Exchange Language Resource Messaging) – see section 8.1.1.3
- OGC (Open Geospatial Consortium) Web Mapping Service (OGC GML, 2012)
- KML (OGC KML, 2012), GeoRSS (GeoRSS, 2012), GML (OGC GML, 2012) and others.

To cater for the variety of standards which may overlap or even conflict, UICDS implements the concept of “core digest data” (Ucore Digest) which summarises the *Who*, *What*, *Where* and *When* of any event. (GeoRSS, 2012).

8.2.1.2 Integrated Public Alert and Warning System (IPAWS) - US

IPAWS (FEMA, 2012) is also an initiative of the US Department of Homeland Security under a broader umbrella aiming to “plan, prepare and mitigate” emergencies.

IPAWS also uses CAP to format the alerts. A supplemental Profile specification also exists to ensure compatibility with the previously existing systems used in US.

A pilot project called 'MyStateUSA” was developed as a proof of concept for IPAWS and an experiment was also run to demonstrate interoperability with Canada. CA/US Enhanced Resilience (CAUSE) experiment (IRCAN, 2011) run in June 2011 to share alerts and geo metadata information between IPAWS (US) and MASAS (British Columbia, Canada). See further below a description of Multi-Agency Situational Awareness System Information Exchange (MASAS-X) - Canada.

The Commercial Mobile Alert System (CMAS) (FEMA CMAS, 2012) in US uses IPAWS to allow alerting authorities to send a non-subscription based text message to cell phones in geo-targeted areas to alert or warn individuals about an imminent threat, an AMBER alert or a Presidential message.

8.2.1.3 National Alert Aggregation and Dissemination System (NAAD) – Canada

NAAD (Pelmorex, 2012) was launched in 2010 at the initiative of the Canadian Radio-television and Telecommunications Commission (CRTC), when a private

Canadian telecommunications company (Pelmorex) committed themselves to the task of broadcasting emergency alerts at no costs for subscribers. The system only accepts emergency alerts from authorised FPT (Federal, Provincial and Territorial) departments or agencies in Canada and disseminates them to the public via radio, cable television and satellite television. The system is looking to expand to include participation of cell phone companies, social media websites or other Internet and media distributors, so that even more Canadians can be alerted in time.

NAAD is based on the CAP protocol and, as at August 2012, it is at version r5.0.

8.2.1.4 Multi-Agency Situational Awareness System Information Exchange (MASAS-X) - Canada

MASAS-X (MASAS, 2012) is a Pilot Project of the Government of Canada. It is a first step in building a broader Multi-Agency Situational Awareness System (MASAS). MASAS-X aims to share relevant alerts, from road closures, severe weather or plumes warnings, to sharing information about water stations, shelters location and status.

The MASAS project slogan is to “move from restricting what we share to restricting what we don't”. As the name says, the major difference from NAAD (see 8.2.1.3) is that MASAS is a multi-agency effort, and non- government organisations can gain access to the system and are able to participate in information sharing if they are endorsed by a public agency.

MASAS was also triggered by the need of exchanging emergency alerts with US and establishing a cross-border communication framework.

MASAS is strongly supported by the Canadian Association for Public Alerting and Notification (CAPAN, 2012) and GeoConnections (Natural Resources Canada, 2012). Their shared view was that people are usually interested in alerts and notices only when pertain to their areas of interest, hence the situational awareness feature built into the system.

MASAS uses the CAP protocol Canadian Profile (CAP-CP) (CAPAN, 2012).

8.2.1.5 Interoperability of data and procedures in large-scale multinational disaster response actions (IDIRA) – European Union

The IDIRA project (IDIRA, 2012) was initiated in May 2011 by the European Union, in response to the large scale disasters, which happened repeatedly in Europe, where multinational help was required and the need to manage efforts during emergencies collectively was obvious. IDIRA is currently supported by 18 organisations in 7 countries.

The two major issues faced by IDIRA were around standards to be used for information handling and specific event codes existing in various countries, as well as communication barriers due to 23 official languages existing inside EU.

IDIRA uses and combines existing standards such as CAP, EDXL-SitRep, EDXL-RM, ATOM (Atom, 2012), etc.

8.2.1.6 Japan disaster mitigation and prevention – Japan

In Japan natural disasters happen very often and in a great variety (earthquakes, tsunamis, typhoons, volcanoes eruptions etc.). Japan Meteorological Agency (JMA) needs to disseminate the information about all natural hazards. One very important aspect is the low level of details that needs to be transmitted to all municipalities in the country (e.g. estimated and observed tsunami heights and arrival times, earthquake hypocentre and magnitude, seismic intensity at each site and others). JMA has a strict tsunami warning classification linked to the action required from the population, five different volcano alert levels and many other classification mechanisms for other types of disasters (torrential rain, flooding etc.).

While CAP could be used to express core information about the warnings, it could not express detailed observations and forecasts, where quantitative estimates and time-sequential values are provided. This is why JMA has implemented JMX (see Section 8.1.2.3). At the moment CAP and JMX are utilised together: CAP for higher level of details and for international users (inside or outside Japan) who cannot understand Japanese, and JMX for domestic users (local authorities responsible for the safe evacuation of the people).

8.2.1.7 Community-based Disaster Management Centre (Sarvodaya) - Sri Lanka

Sarvodaya (Sarvodaya, 2012) is Sri Lanka's largest community development non-government organisation that also acts as a hazard information hub and community disaster management system.

Their system performs two key operations for community-based emergency response: alerting and reporting. Two major complexities Sarvodaya faces in Sri Lanka are: a) the usage of different local languages in different areas of the country and b) the computer and technical literacy barriers for many people. Sarvodaya had therefore to look for efficient solutions to transform text to voice that allows the alerts to be communicated via voice message with language selection.

In their efforts, Sarvodaya rely on Sahana Software Foundation (Sahana Foundation, 2012) for free, open source software that uses CAP, EDXL-HAVE and EDXL-SitRep for alerts, hospital availability data exchange and situational reporting.

8.2.1.8 China Public Emergency Alerting System (CPEAS) - China

CPEAS was built by China Meteorological Administration (CMA) (CMA, 2012) with the aim to notify Chinese people about four main categories of alerts: natural calamity, accidents, public health emergencies and social safety issues. The system is based on CAP and it disseminates alerts by sending messages in a top down manner via CMA branches, going from the state level to 31 province levels, 342 prefecture levels and 2379 county level. The major issues faced are related to

integrating large categories of alerting sources and adopting multiple dissemination paths.

8.2.1.9 Google Public Alerts - US

This is an initiative of the Google Crisis Response (Google, 2012) team, aiming to be a platform to disseminate relevant emergency alerts to users when and where they are searching for them. The project was triggered by the fact that many people would search on Google in an emergency and there was no Emergency Broadcast System on the Internet.

As of September 2012, Google Public Alerts shows weather, public safety and earthquake alerts from US National Oceanic and Atmospheric Administration (NOAA), the National Weather Service and the US Geological Survey. Their intention is to expand the service by including different types of emergency alerts, not only weather.

Google Public Alerts use CAP v1.2 to format the alerts and also standards from the EDXL family (see Section 8.1.1), PFIF (People Finder Interchange Format) (Zesty, 2012) and KML (Keyhole Markup Language) (OGC KML, 2012).

8.2.1.10 Other initiatives

The map at the beginning of this section (see 8.2.1) shows a number of other organisations that use CAP or some other standards for the distribution of alerts, and which have not been described in detail in this section, as follows:

- U.S. National Weather Service (US NWS, 2012) – publishes weather watches, warnings, advisories, etc. in CAP and ATOM format.
- Global Disaster Alert and Coordination System (GDACS, 2012) – sponsored by United Nations and the European Community, they publish alerts in CAP format about earthquakes, tropical cyclones, floods and volcanos.
- U.S. Geological Survey (US GS, 2012) – publishes alerts for earthquakes and volcanoes in CAP format via IPAWS (FEMA, 2012) and also includes links to those CAP messages into their ATOM alerts.

8.2.2 Standardised terminology, vocabulary and symbology

Along with speaking a common language in terms of data exchanged in emergency scenarios, maps and visualisation of the information received are also critical for decision makers, especially given the time-sensitive context. This is why various legislators undertook projects to develop standard terminology or symbology sets, which are supposed to be used consistently by information providers in any phase of emergency management cycle (from prevention to preparation, response and recovery).

“Emergency Management Vocabulary – Canada” (Government of Canada, Translation Bureau, 2012) intends to promote adoption of standardised emergency

management terminology in both official languages in Canada (English and French). It contains terms and definitions for more than 200 emergency management concepts. A wide selection of sources was examined to extract the most useful terms.

8.3 Adoption Levels for Emergency Data Standards in Australia

Most of the current efforts in adopting data exchange standards for emergency scenarios in Australia are part of a larger federal initiative for implementing an Australian Emergency Management system. This work has been coordinated by the Australian Attorney-General's Department (AGD) (AGD, 2012).

8.3.1 Common Alerting Protocol

This national project was started as a response to the outcomes of the investigation of the Black Saturday bushfires in February 2009. A study was first conducted by AGD in 2009-2010 and it determined that CAP international standard (see Section 8.1.1.4) was the most suitable content standard available to date for emergency alerts and notifications. The project titled “Common Alerting Protocol (CAP)” received funding from the Australian Government under National Emergency Management Projects in three stages (AGD CAP, 2012).

The first two stages (2009-2010 and 2010-2011) focused on adopting the international CAP standard and then creating an Australian profile version that aims to respond better to the Australian community emergency communication requirements.

The current stage (2011-2012) aims to facilitate Australian contributions and influence future updates of the international CAP standard. In Victoria, the Department of Sustainability and Environment (DSE), the Country Fire Authority (CFA) and the Office of Emergency Services Commissioner (OESC, 2012) are participants in the CAP-AU Stakeholder Group (CAP SG) that develops the Australia CAP Profile.

Other emergency management projects, previously developed in Australia, are described in the following sections.

8.3.2 One Source One Message (OSOM)

This is a web-based messaging system, used currently in Victoria by CFA and DSE to deliver emergency warnings (Jackson, S., 2009). As the name says, the system aims to provide a single source of information to DSE and CFA websites where the information is received from Incident Control Centres and published in a few minutes from being authorised by an Incident Controller. The same source is also used for other organisations such as Victorian Bushfire Information Line (VBIL) (CFA, 2012) or emergency broadcasters such as ABC Radio (ABC, 2012).

OSOM uses CAP to format and send the alerts to the intended recipients (websites, other organisations, etc.).

8.3.3 Emergency Alert

This is a national telephone-based emergency warning system that has been used in all Australian states and territories since December 2009 (Australian Goverment, 2012). In the past 3 years Emergency Alert issued more than 7 million alerts, including storm, flood, tsunami, bushfire, chemical incident or missing persons.

Currently emergency alerts can be sent to landline telephones based on the location and to mobile phones based on the address of the service. Location-based alerting for mobile phones is planned to be rolled out by Telstra in November 2012, with other mobile service providers aiming to provide same type of service in 2013.

8.3.4 Joint Australian Tsunami Warning Centre (JATWC)

This warning centre aims to accurately detect, monitor, verify and warn of any tsunami threat to the coastline of Australia and its offshore territories (BOM JATWC, 2012).

It is jointly operated, 24 hours a day, by the Bureau of Meteorology (BOM, 2012) and Geoscience Australia (Geoscience , 2012) and funded by the Australian Government via the Attorney-General's Department (ADG).

JATWC also uses CAP to format and send the tsunami alerts.

8.3.5 Standardised terminology, vocabulary and symbology in Australia

In 1998, Emergency Management Australia (AGD, 2012) developed an “Australian Emergency Management Glossary” (Emergency Management Australia, 2012) in consultation with all key management organisations in Australia. This glossary provides a list of emergency management terms and definitions.

The “All hazards symbology” project was developed by the Emergency Management Spatial Information Network Australia (EMSINA) in July 2010 (EMSINA, 2010) with the aim to develop a consistent symbology set that would be adopted and used by emergency management agencies across Australia and New Zealand. The need for the project was raised from an audit of mapping symbols used at the time by the emergency management agencies. Although a glossary of terms and definitions was in place (see the paragraph above about the “Australian Emergency Management Glossary”), the audit revealed not only physical differences in symbology used between various jurisdictions but also different terminology and definitions associated with the symbols. These differences impacted the ability to create a usable emergency map based on the information provided by multiple agencies, therefore an “All hazards symbology” matrix (EMSINA, 2012) was proposed by EMSINA.

8.4 Recommendations

From the current landscape of international and Australian efforts described in the previous sections, it is clear there is an increased focus on creating a message standardisation framework for emergency management, with the aim of enabling rapid information sharing between agencies and communities during disasters, and protecting human lives.

As can be seen from the map at the beginning of Section 8.2.1, most of the efforts are concentrated in North America (US and Canada), followed by Europe. Australia has just started on the path of data standardisation for emergency management scenarios and, *in the context of VINE, the following recommendations are made*:

8.4.1 With regards to data exchange standards to be utilised:

EDXL-DE is recommended as an overall wrapper for VINE messages, for message distribution with or without an XML-based payload (e.g. CAP-AU, EDXL-HAVE, EDXL-SitRep, etc.).

CAP-AU is recommended for sending alerts and notifications. A number of gaps have been identified in CAP v1.2 and CAP-AU, therefore the following further recommendations are made:

- Usage of one event type per alert message. This is because while CAP v1.2 and CAP-AU v1.0 allow multiple event types per message, they assign a unique identifier per message and therefore any subsequent update referencing one event type from the message would be seen as an update of all event types;
- Update the CAP-AU Event Codes List (OASIS , 2012) to include any other codes specific to the Australian context which are not currently in the list (e.g. planned burns);
- Work with OASIS to clarify the actual meaning of *urgency*, *severity* and *certainty* for the events in the Australian context, and reflect those in the CAP-AU Profile.

EDXL-HAVE is recommended for data exchange regarding hospitals, health services and health practitioners' availability.

EDXL-RM is recommended for requests and responses regarding various types of resources (e.g. fire trucks, rescue teams, etc.).

EDXL-TEP is recommended for emergency patients. This standard is still in draft and it is recommended that VINE governance committee collaborate with OASIS to publish a working version as soon as possible.

EDXL-SitRep is recommended for situational reporting. As above, work with OASIS to publish a working version.

There is no standard at the moment to exchange data regarding location and availability of shelters in emergency scenarios. It is recommended to work with OASIS on a standard for this purpose. Along with location and availability, features such as total capacity, level of occupancy, services, meals, amenities, catering for animals (among others) need to be catered for.

Tsunami and cyclones are not usual events in Victoria; however, work needs to be done in conjunction with NICTA and OASIS to ensure existence of official versions of TWML and CWML for these emergency scenarios, especially if a scalable national level VINE system is considered as a future possibility.

Given the large number of agencies planned to collaborate in VINE, by submitting and using data and information, the sender of any emergency message needs to be clearly indicated. If a decision is made to use EDXL-DE as a wrapper for any payload exchanged in VINE, a unique sender identifier can be used to identify each working party. In any other scenario, attention must be paid to including sufficient information in the respective message field so the sender can be easily recognised (e.g. use full agency name instead just abbreviations).

8.4.2 With regards to emergency management terminology

All agencies participating in VINE will need to use and promote a single, shared glossary of terms and definitions for emergency management concepts.

This set of definitions should reside in VINE and, if communication of emergency information to community in languages other than English is deemed necessary, the glossary should include corresponding translations in those languages. This will minimise the risk of errors and avoid using arbitrary translation tools in an emergency.

8.4.3 With regards to data governance from a standards perspective

For any data standard utilised in VINE there should be a single source of information for specifications, schemas and any other documentation. Given that some standards are novel and at early versions, this will ensure that all agencies use the correct version of each standard at any point in time. For example, OASIS Emergency Management Technical Committee (OASIS EM TC, 2012) must be the source for all OASIS approved standards for emergency management.

The same applies for Australian Profiles of any given standards (where applicable). At this early stage it seems that multiple references to the Australian Profile of CAP exist, and there are at least three different abbreviations used for it: the OASIS authoritative version (OASIS EDXL-CAP-AU, 2012) uses the abbreviation “CAP-AU', the same Profile is abbreviated “CAP-AP' in (OASIS , 2012) and there is a non-normative reference to the same version of the profile on the Australian Government “Govshare” website, as “CAP-AU-STD” (Australian Government, 2012). The multitude of abbreviations used, different names/titles and different storage locations for the same documentation will prove confusing for the VINE stakeholders, and *therefore attention should be paid to the governance aspect for all data standards used in VINE.*

8.4.4 With regards to international visibility and contribution to standards

VINE's governance committee will need to ensure visibility of the Australian efforts in terms of emergency management, as well as a solid presence in the groups working on various emergency data exchange standards (the ones described in the previous sections or any others, as applicable). The following issues need to be considered:

- Similar with the Canadian public alerting industry, the interests of VINE must be represented to the Organization for the Advancement of Structured Information Standards (OASIS), the standards body overseeing CAP:
- Increase presence of VINE stakeholders in the Emergency Management CAP Profiles Sub-Committee. Currently only one person from the Attorney-General's Department represents Australia on this sub-committee;
- VINE stakeholders need to increase participation in emergency policy workshops organised by OASIS For example, at the 2012 Emergency Alerting Policy Workshop (May 1-3, 2012, Montreal, Canada), Australia was represented by a single attendee from the Attorney-General's Department, compared with 42 Canadian attendees and 25 US attendees;
- The World Meteorological Organisation maintains a Register of Alerting Authorities (WMO, 2012) that is used as a reference by OASIS members and working groups. There are only two organisations on the Australian list, the Bureau of Meteorology (as authoritative agency for all hazards) and the Hydrological Services Program (for meteorological hazards only). VINE governance should work with WMO to update the list of official Australian alerting authorities to include those in Victoria.

9 Architectural Framework

This section describes in more detail the various architecture components of VINE. As originally discussed in Section 6, the architecture for VINE is separated into three layers: the cloud infrastructure, the core interoperability platform, and the services platform. In addition, the VINE API is a crucial component that defines access to the platform.

All of these components, when taken as a whole, constitute VINE. However, this document focuses primarily on the core interoperability platform and the API. In the case of the federated cloud infrastructure, a variety of commercial solutions exist that can satisfy VINE's infrastructure requirements. Furthermore, the precise choice of architecture is constrained by regulatory and governance considerations. commentary on the cloud infrastructure is limited more to a discussion of components and capabilities rather than to a complete architecture. In the case of the services platform, it is neither possible nor necessary to enumerate all the services that may reside on this platform; therefore it is necessary to define an architecture for hosting and delivering a variety of services, but not an architecture for those services themselves.

9.1 Cloud Infrastructure

According to NIST (Mell & Grance, 2011) cloud computing enables convenient, on demand and ubiquitous access to shared network, storage, applications and other computing resources that can quickly be provisioned and released. Essentially a cloud-computing infrastructure consists of:

- Virtual Machine (VM) image management
- Storage
- Networking

These resources are normally controlled and managed by a set of APIs, which make it possible to develop elastically scalable systems, to which resources can be added and relinquished on demand. Virtualisation—primarily of compute resources, but also of storage and networking—enables the consumption of these resources as *services*: they are made available on demand, utilised, and then given back when no longer needed. This approach transforms infrastructure and runtime environment into *utilities* and provides engineers and system architects with better tools and abstractions for creating potentially more scalable and fault-tolerant systems.

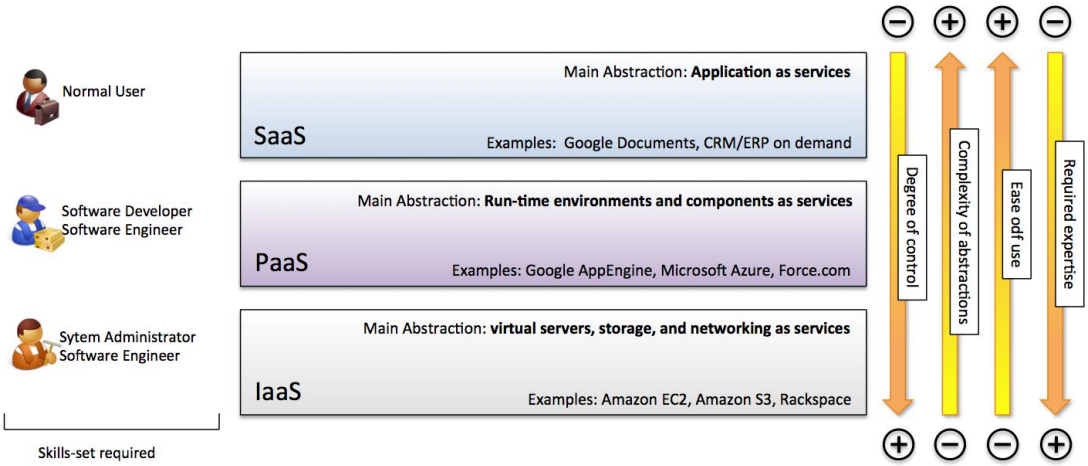

Figure 5. Reference classification of cloud computing service offerings.

As illustrated in Figure 5, cloud computing offers three different types of service models, which provides different capabilities and target different class of users. These are IaaS, PaaS, and SaaS:

- **IaaS (Infrastructure-as-a-Service)** provides processing, storage, networking and other computational resources used for a quick provisioning and deployment of services and applications by cloud users. The fundamental abstraction offered by IaaS to define applications and systems are (virtual) servers, storage, and networking;

- **SaaS (Software-as-a-Service)** provides applications running on top of cloud infrastructure. Users are able to use such applications without any need for low-level control and management of the cloud infrastructure. The fundamental abstraction offered by SaaS is an application, which is accessed and operated through a web interface;

- **PaaS (Platform-as-a-Service)** provides ways for clients deploy their own applications on the cloud infrastructure without any need to explicitly administer or control the cloud infrastructure. If IaaS offerings provide essentially infrastructure on demand and SaaS offerings are focus on leasing application services (e-mail, document and spreadsheet management), PaaS offerings provide cloud-based runtime environment and libraries that can be leveraged by developers to build scalable and

elastic applications, without exposing developers to infrastructure management. Therefore, it is more flexible than SaaS, but not as flexible as IaaS, since the application development model is imposed by the runtime environment and the APIs. The fundamental abstraction offered by PaaS is a scalable runtime environment accessible through APIs and a set of components allowing users to develop applications in one or more programming languages.

It has to be observed that this classification constitutes a reference model for exploring all the available offerings on the market. Over time as cloud computing become more popular and evolved, several vendors enriched their portfolio by integrating services of different categories, thus blurring the divisions between IaaS, PaaS, and SaaS. Amazon Web Services (AWS) and SalesForce.com are two classical examples of this trend. AWS initially started as a primarily IaaS solution and over time it has consistently added services which have turned the whole set of services offering in a PaaS-like environment. On the other hand, SalesForce.com initially offered a SaaS solution for Customer Relationship Management (CRM) and Enterprise Resource Planning (ERP). Over time by following the customer needs, who were demanding for a greater degree of flexibility and control for defining their applications, it opened up the underlying platform on which it was based: Force.com, thus expanding its service offering to the PaaS layer. Despite this, AWS and SalesForce.com still allow users to leverage IaaS and SaaS services for their needs, respectively. Rather than trying to tag a specific vendor solution with the SaaS, PaaS, or IaaS label, it is more important to understand which abstractions are primarily sought for cloud computing development and this defines the type of service being looked for.

With respect to VINE, the platform will deliver services belonging to all the three categories discussed above. This is because VINE aims to serve a large variety of stakeholders: community individuals and ESOs personnel will be mostly interested into the SaaS capabilities of VINE, researches and third party services developers will often need to interact with the PaaS and IaaS capabilities. Throughout this document we have referred to, and we will mostly refer to, VINE as a platform focussing on the scalable runtime environment and the reserve of services that are provided to build the foundations for information interoperability and advanced decision support in emergency management. Since VINE wants to provide an extensive set of capabilities to become a powerful ecosystem of application and services for emergency management, it will need to make available services that are primarily belonging to the underlying cloud infrastructure. In these cases, we will refer to VINE as the system including such infrastructure.

As a whole, VINE will provide a computing environment that can:

- Scale incrementally and elastically;
- Deliver highly configurable and isolated environments;

- Maximise the use of its own resources by dynamically change the virtual resources allocation of the cloud infrastructure according to the evolving and changing needs of the platform.

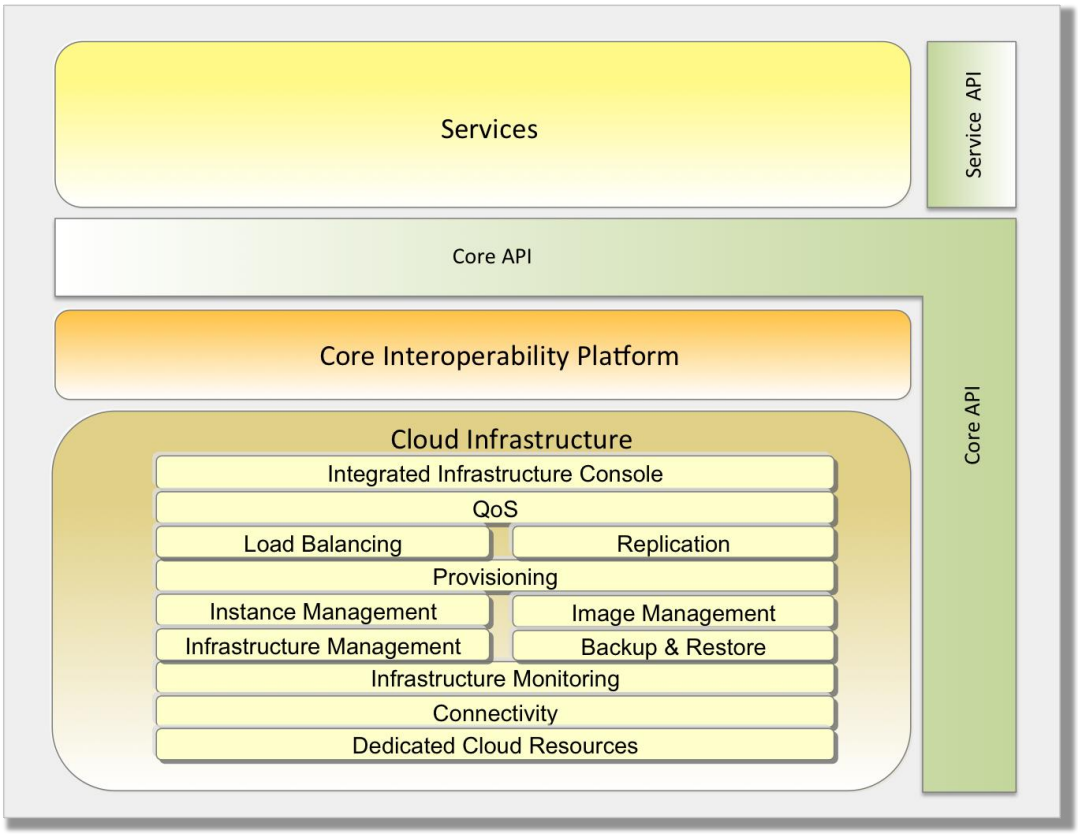

Figure 6 Desired features and expected characteristics of the infrastructure supporting VINE.

For what regards the underlying cloud infrastructure, Figure 6 provides an overview of the set of services that are expected from such infrastructure. These services cover a set of features that are now available in most of the major commercial IaaS offerings, and include:

- An *integrated console* that provides access to all the services of the infrastructures either via visual interface or APIs;
- Support for *Quality of Service (QoS)*, so that applications and services can be executed from very bottom layer of the computing stack under different performance and availability requirements;
- Capabilities for *load balancing and replication*, which will allow systems engineers to implement the ability of delivering the desired performance in terms of response and availability for critical applications;
- Capabilities for provisioning virtual servers on demand so that VINE can support surges in demand without allocating resources for more the time needed to address the load spikes;

- Capabilities for infrastructure and image management so that environment can be customised, composed, and configured with a high degree of control, thus reducing the limitations of VINE in providing support for building an ecosystem of services;

- Basic functionality for *infrastructure monitoring*, *backup* and *restore*, which will allow to maintain the underlying infrastructure constantly under control and with a recovery plan in place;

- *Connectivity* capabilities and *dedicated cloud resources* that compose the base layer of the cloud infrastructure.

All of these capabilities work together to provide support for the management and creation of a virtual computing infrastructure (servers, storage, networking) whose service model is essentially based on a pay-per-use strategy, where the cost associated to a resource becomes a function of its usage (uptime, space, or bandwidth, etc.). This approach provides the biggest benefits in case of public cloud infrastructure; users only pay for the operational costs associated to their systems, while the capital costs are eliminated since the infrastructure is leased. Public cloud infrastructure is not always a viable solution, especially when sensitive and critical information cannot be hosted on third-party infrastructure. An alternative solution is a private cloud infrastructure where the organisations that own the infrastructure are also the only consumers of the cloud services. In this case there is no elimination of capital costs, but still a pay-per-use model can be useful for a proper and effective allocations of costs to the different organisations (or divisions of a single organisation) according to the use they make of the shared infrastructure. A third option is represented by hybrid cloud infrastructure where an existing system is augmented with public cloud resources. In cases where the need of additional capacity is transient, provisioning additional resources from a public cloud could be a cost effective solution, since it avoids the need to purchase resources whose use can be very limited.

Commercial IaaS deployments have reached a level of maturity that enables them to host systems that have time-critical and reliability requirements. Moreover, improvements in the in the service offerings have made cloud technology more widespread and popular. The increased interest in cloud technologies has also opened the path to further development and advancement and currently it is quite common to have offerings that exhibit the characteristics discussed above (Figure 6).

IaaS solutions are used to implement public, private and hybrid clouds. Besides not bearing the cost of deploying and maintaining large data centres the advantage of choosing a public cloud solution resides in the possibility of leveraging a large number of mature vendors in the space, comparing their service offerings, and identifying the best trade-off between cost and services. The major disadvantage of a public cloud offering is given by the limitation of having to mostly rely on a single vendor to deliver the solution, and thus the risk of *vendor lock-in*. With respect to this risk, public cloud vendors who can provide the capability of exporting their systems into—or that are compatible to—existing cloud standards should be

considered as they provide a way to make vendor lock-in less limiting. In order to do that solution deployments should be developed in a way they could be easily migrated from the current provider to another one without many changes.

Clearly, this is not an easy task, as depending on the technologies adopted a reasonable amount of work may be required. The best approach is to use standards so that applications can be easily moved around different cloud vendors without any change. Unfortunately, there are no standards defined so far. However, initiatives such as OpenStack (The OpenStack Foundation), an open source effort to create a set of open source tools for cloud infrastructure, are trying to achieve that. Moreover, relying on a public cloud deployment could also bring concerns about the security and the regulatory compliance of the infrastructure, in cases where these aspects are particularly critical. As already mentioned, public clouds solutions rely on third party vendors infrastructure and systems and they require a lot of trust in the capability of these vendors (as well as their integrity) in providing a secure and safe computing environment.

A solution on the other end of the spectrum is represented by a private cloud deployment where the organisation owns the cloud infrastructure (i.e. data centres and middleware) and has complete control over the technology stack deployed on it. Whereas this solution is less cost effective than a public cloud offering, it is worth considering because it provides the opportunity of developing a cloud-computing platform that is based on the emerging standards in the field and will prevent future vendor lock-in. This solution incurs considerable upfront costs that need to be borne for the initial deployment, as well as maintenance costs for keeping the infrastructure operational and administering the system deployed on top the infrastructure. The drawbacks can be partially lessened by using an incremental strategy (e.g. starting with a single data centre and then adding more capacity as the system grows). Private cloud deployments will also provide the opportunity of controlling and managing the infrastructure. This will facilitate the creation of a computing environment that meets the required standards for security and other regulatory compliance needed by the systems and the data hosted on top of it.

Given the needs for resilience of VINE and the scale that this system will reach once put in production it worth considering a federated approach for the deployment of the virtualisation layer and the provisioning of cloud services. The concept of federation in this view plays a very peculiar role. Despite some advancement in the field the concept of cloud federation as a commercial solution is not yet sufficiently mature to support the demands of a system like VINE. At present, standardisation efforts in the area of virtualisation formats and cloud services—even though regarded with interest by multiple vendors—have not been endorsed to deliver the required level of interoperability to make a commercial cloud federation a reality. A different case is constituted by a private cloud deployment where the utilised infrastructure and the services built on top of it are shared among different organisations that partner together to reach the critical mass required to undertake the development and maintenance of a large infrastructure. This arrangement then leads to a federated cloud platform. Being based on a shared and private infrastructure this solution also facilitates the

creation of an environment meeting the regulatory compliance and the security requirement needed for applications, systems, and data.

9.2 Core Interoperability Platform

VINE's core interoperability platform is intended to provide a set of fundamental services for information interoperability, upon which many other capabilities may be built. In addition to meeting the previously stated goals for VINE as discussed in Section 5 (unified identity, real-time data propagation, etc.) the architecture of the core interoperability platform has two core principles: firstly, that the platform is as compact as possible with respect to the set of capabilities it offers; and secondly, all services outside of the core interoperability platform interact with the platform via a unified API that can be accessed, subject to permissions, by any service inside or outside VINE.

The core information interoperability platform will effectively function as a type of enterprise system, with a high level of interconnectedness between components, and very strict requirements in terms of system integrity and resilience. As such, it must be administered by a single dedicated entity, with no direct access for any other stakeholders. Keeping the platform as focused as possible and pushing functionality up to the services layer wherever possible will make the system easier to build and maintain, and will minimise the need to involve systems administrators when adding functionality to VINE.

Having all access to the core interoperability platform—whether from inside VINE's services layer or from outside the system entirely—mediated through the same web services API is intended to facilitate an open development model for VINE. Using this model, it will be easy for third parties (such as research institutions) to develop potentially useful service modules for VINE. If these modules are then adopted as part of VINE, quality assurance and testing will be necessary, but reprogramming will not. Providing a single API will ensure that sufficient power is offered to all stakeholders to facilitate the development of a true ecosystem of applications.

Figure 7 provides a schematic view of the core interoperability platform and its place within VINE. It consists of the following high-level components, which are discussed in more detail in the following sections:

- **Enterprise Service Bus.** The primary middleware mediating interactions between components in the core interoperability platform. To support these interactions, it will act as the transport infrastructure for VINE's publish/subscribe event service, which will be responsible for the real-time publish-subscribe functionality.

- **Data Store.** The repository for all data kept in VINE by its custodians. It will consist of a number of different types of storage paradigms, tied together by a data directory service that allows users to find the data they need.

- **User Management.** All services for user management: identity, authentication, authorisation, and role management will reside within the core interoperability platform.
- **Accountability Tools.** A customisable logging framework for keeping a detailed record of provenance and flow of events within the system.
- **Security Tools.** A number of active security measures to ensure that the system is not compromised.

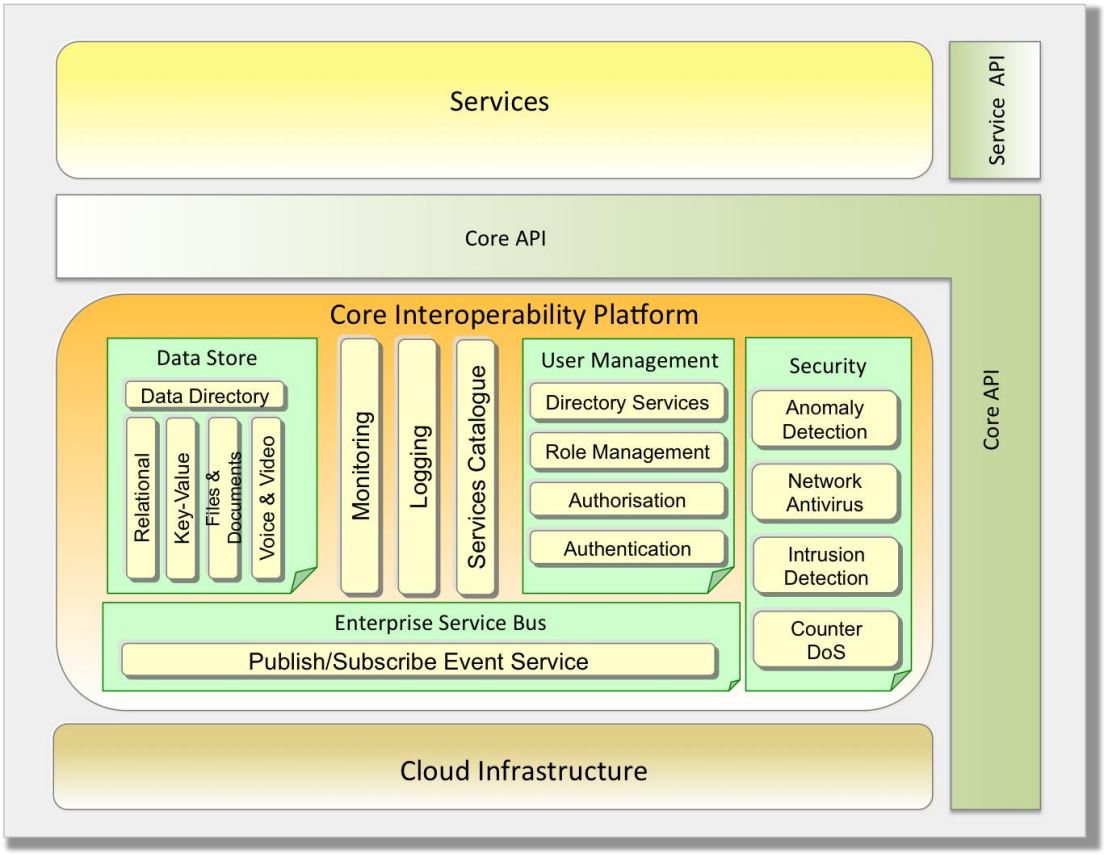

Figure 7 An expanded view of the core interoperability platform layer of the VINE architecture.

9.2.1 Enterprise Service Bus

The middleware for VINE's core interoperability platform will be an Enterprise Service Bus (ESB). ESB's lie at the core of Service-Oriented Architecture (SOA). There is no common definition of what an ESB is; depending on the context of discussion, it can mean one of two things:

- An architectural pattern – a middleware that acts as an intermediary between service providers and service clients.
- A product that offers integration, development toolsets and management environment.

In this document an ESB is defined as an architectural pattern that is deployed as a middleware that connects and mediates service connections. Such an approach minimises the complexity of managing connections between services and consequently provides the following advantages:

- It reduces the complexity of service interfaces, as they only need to provide one generic interface to be used by the ESB.
- Maintenance is better handled as any changes to service location and interfaces only affect the ESB itself.
- Because service clients do not communicate directly to service providers they (service providers) can be replaced without any changes on the client side.

Figure 8 illustrates the main concepts of an ESB, which is a logically centralised control structure with built-in features where multiple service providers and clients are connected through a message bus. ESB's also offer communication, data transformation, security, protocol binding and other features. These features will be leverages within VINE to offer services such as federated data store and the publish/subscribe event service.

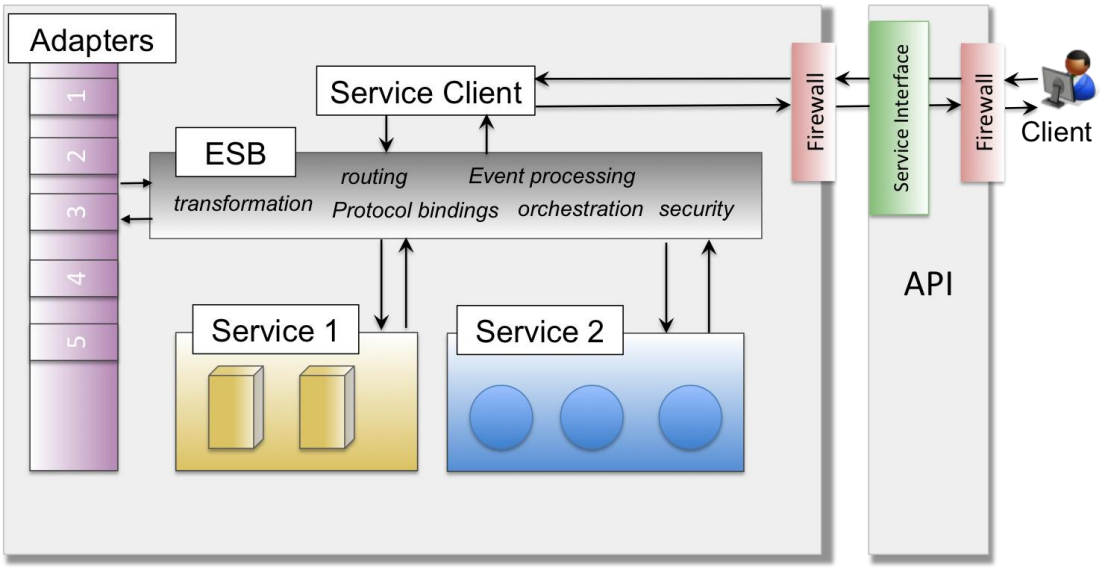

Figure 8 ESB high-level architecture.

Among the variety of functionalities provided by an ESB, below we describe the ones that will be central for VINE's capability:

- Routing – it is the ability of channelling a request to endpoints defined statically or dynamically. For instance, in VINE a request should be routed to a specific service based on who is requesting it (e.g. a user may not have authorisation to access such service). Another example is where service requests coming from a specific organisation should be routed to a particular service provider instance.

- Synchronous and asynchronous messaging – there is a need for supporting different types of message calls, depending on how the control is returned to the call originating point. In synchronous calls, the event bus invokes the target service immediately and control is not returned until the message has been received by the target service for processing. On the contrary, for asynchronous calls, the event bus uses a messaging service to deliver the message to the target service, while returning the control immediately, before the target service has received the message. ESBs provide this flexibility of supporting both styles without any additional work for both service providers and clients. The Publish/Subscribe Event service described in Section 9.2.2 is an example of why asynchronous messages are necessary.

- Protocol bindings – the event bus in VINE should support multiple protocol bindings for message delivery using synchronous/asynchronous, request/reply or publish/subscribe models. Existing/legacy services are often found using communication protocols that are very different from commonly used standards in today's SOA systems. ESB addresses this interoperability issue by providing the ability of binding legacy service protocols to a different communication protocol. VINE aims to be a platform where multiple systems could be effortlessly integrated as much as possible. Different agencies with legacy systems should be able to communicate with VINE and access its services. Also, as an intended platform for researchers, VINE should be flexible enough to be plugged with different types of research applications using different protocols. Therefore, it is important to be able to dynamically bind service providers for different communication protocols.

- Content transformation – it is the ability to convert structure and format of request messages to the expected structure and format of the service provider. It is important to business integration where different software solutions have to be integrated. One of the requirements of VINE is to provide agencies the capacity of storing and requesting content in different formats. In other words, inter-agency data sharing has to be supported by VINE, consequently making content transformation an important feature.

- Service mapping – it is the capacity of translating the exposed service to the real service implementation. More often than not, a service provider is exposed to its external clients differently from how it was originally implemented. Therefore, such "external" service has to be mapped to the original service provider. The Publish/Subscribe functionality offered by VINE is such an example. It is externally offered as a Publish/Subscribe service based on Web services specifications, however internally it is implemented by using the message queue broker that is part of the ESB core. Section 9.2.2 provides more details on the Publish/Subscribe service. Figure 8 shows an example of service mapping. An entity (service client) uses two others (services 1 and 2) to offer an external interface that is used

by external clients. The service client entity uses the service mapping to find services 1 and 2 and consequently communicate with them.

- Adapters – A key aspect of any ESB is the capacity of integration of multiple third-party, and usually, legacy applications. Adapters make the communication between third-party applications and service providers and clients possible, so that there is no need to change either the service provider or the application in order to make them "talk" with each other. It is a useful feature that makes the integration between applications easier.

- Quality of service features such as security, and persistence – ESBs provide security mechanisms such as authorisation, authentication, auditing, and encryption of messages in order to ensure confidentiality, integrity and accountability. Such features are provided out-of-the-box by mainstream ESBs and they do not demand much effort for integration and use. However, depending on the type of security offered it might be necessary to use other forms of security implementation. In VINE, features such as data encryption, in addition to message encryption, are necessary and should be implemented if they are not offered by the ESB solution. It is a requirement of VINE that all data shared by agencies and to be used by external and internal users should be encrypted. Another requirement is auditing. Any action taken by either users or service providers should be logged for future auditing and consequently possible accountability.

- Transaction management – it is also an important aspect and depending on the type of architecture used by the ESB it may be challenging. For instance, transactions over distributed ESB deployments are not a trivial problem. Transaction management provides a single work unit for a request regardless of how many distinct steps are required to complete it. In other words, a transaction should guarantee the coordination of multiple services used to process a single request. It is important to understand that the management of local transactions, usually supported by orchestration/choreography, does not support distributed transactions, which is the main characteristic of transaction management. Normally it is offered via the WS-Coordination specification (WS- Coordination Specification, 2009) or for Java-based ESBs via activity service specification (Robinson, 2006). VINE is designed to support a federated deployment with multiple ESBs distributed around different data centres, therefore supporting distributed transactions is of paramount importance.

9.2.2 Publish/Subscribe Event Service (PSES)

The Publish/Subscribe event service provides VINE users with the ability of subscribing for particular types of data depending on their interests (see Section 7.1.3 for more details). The service is based on the Publish/Subscribe message exchange model. This service makes it possible to connect different entities in a loosely coupled, scalable and dynamic manner. In VINE a subscriber is an entity (agency, user) interested in particular topics and/or data. A publisher is an entity that creates/posts data to VINE. There are multiple ways to implement a pub/sub

service. OASIS provides a Pub/Sub specification aimed at web services. The WS-Notification (Graham, Hull, & Murray, 2006) is a standard for Pub/Sub implementation over Web Services. It consists of a set of three specifications: WS-BaseNotification, WS-BrokeredNotification and WS-Topics. Mapping publishers and subscribers to WS-Notification specification, a publisher is called *NotificationProducer* and a subscriber a *NotificationConsumer*. WS-Topics represent Pub/Sub topics.

Figure 9 shows a high-level architecture of a possible implementation of the WS-notification specification on top of an ESB. The Publish/Subscribe Event service implements the WS-notification interfaces that allow external users to publish and subscribe to topics (WS-topics) and get notified when content matching their subscriptions is published. Notification producer and consumer are internal service providers that implement the *WS-NotificationProducer* and *WS-NotificationConsumer* interfaces respectively. The Publish/Subscribe service wraps them in order to offer a unified interface to external users. Subscriptions and topics content are stored by the data store service.

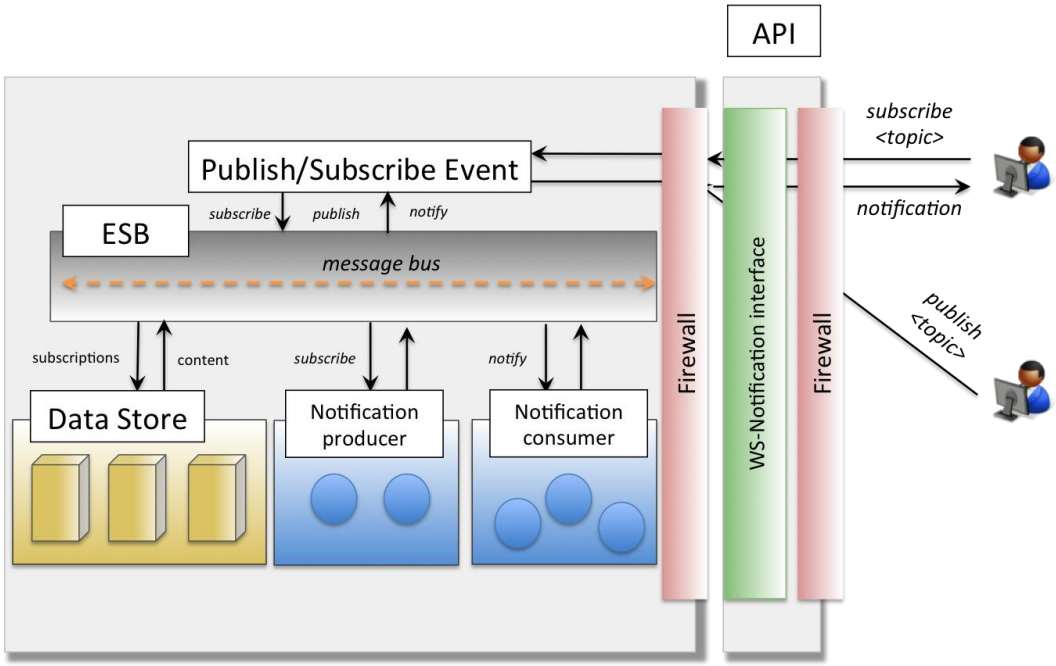

Figure 9 High-level architecture of a WS-notification implementation.

One of the cases where PSES is used is the real-time propagation feature described in Section 7.1.3. One of the high-level requirements is the ability to support hierarchical topic trees, which it is completely supported by WS-Notification (Huang & Gannon, 2006). In addition to that, the standard is adopted by all major ESB vendors, making WS-Notification relevant and recommended for VINE.

9.2.3 Data Store

The data store constitutes one of the very basic services offered by the core platform and it is aimed at providing capabilities for storing, discovering, querying, and in general interacting with the information managed or accessed through VINE.

From an architectural point of view the data store is organised as a collection of interconnected services that are accessible through the *enterprise service bus (ESB)* and referenced in the *services catalogue* (see Section 9.2.4). These services are mainly the *Dataset Registry*, the *Dataset Deployment*, and a variable set of services that directly manage the access to data.

The data store utilises the abstraction of *dataset* for representing of the data in VINE. A dataset encompasses a collection of information elements that are semantically grouped together. Given the wide span of VINE, it is envisaged that a variety of different datasets will be hosted. These may require different interaction patterns for querying, retrieving, adding, updating, and deleting data. As one of the core aspects of VINE is to become an authoritative source for information supporting emergencies, it is expected that associated to a dataset there are one or more custodians, which are entities responsible for the content of the datasets and that define the privileges required for accessing and interacting with the datasets.

The data store will provide services for hosting and exposing different types of datasets and an extensible mechanism for improving such support with new types of datasets in the future. In its first instance the data store will provide support for the following types of datasets:

- *Relational datasets:* these datasets will be queried and manipulated by an SQL-like language. Whether full support to SQL or a limited subset of the language will be exposed goes beyond the characterisation of the data store in this document. Also, the specifics of providing access to the datasets (i.e. SQL connection or service based interface) will not be discussed at this level. These fine grain aspects will be part of the architectural solution.

- *Key-value datasets:* these datasets are meant for providing storage and querying support for extremely large structured/semi-structured data. The basic operations that will need to be supported are insertion of a key-value pair, retrieval of an element by key, update of the value of a given key, and deletion of a given key-value pair. As discussed in the previous point the specific details of all the supported operations allowed on these types of datasets goes beyond the purpose of this document and will be part of an architectural solution.

- *File-based datasets:* flat files constitute a more elementary way to store data. The basic capabilities for retrieving from and importing a file into the store must be provided. Also, file-based datasets might expose a collection of files rather than a single file, since a dataset is often divided into multiple files. Therefore the capability of enumerating the files into the dataset is an important aspect. A particular case are metadata-rich files, these are often used in scientific experiments for storing results of simulations or store information in a more convenient format. As VINE is envisaged to be a platform that also supports research it is considered important to provide support for storing and accessing scientific data in a convenient and

efficient way. This means for example, providing support for navigating and accessing these files through specific interfaces, rather than providing the basic capability of uploading and downloading these files. Given that these files are often very large, more advanced interfaces, as well as highly efficient communication protocols, may be required to effectively access the data.

The data store will maintain a registry of all available datasets stored in VINE. This capability, as well as the capability of providing a client with access to specific datasets, will be exposed as data store services and filtered by authentication and authorisation rules that rely on the role-based user management systems defined in VINE. These rules are defined and managed by the custodian(s) of the dataset.

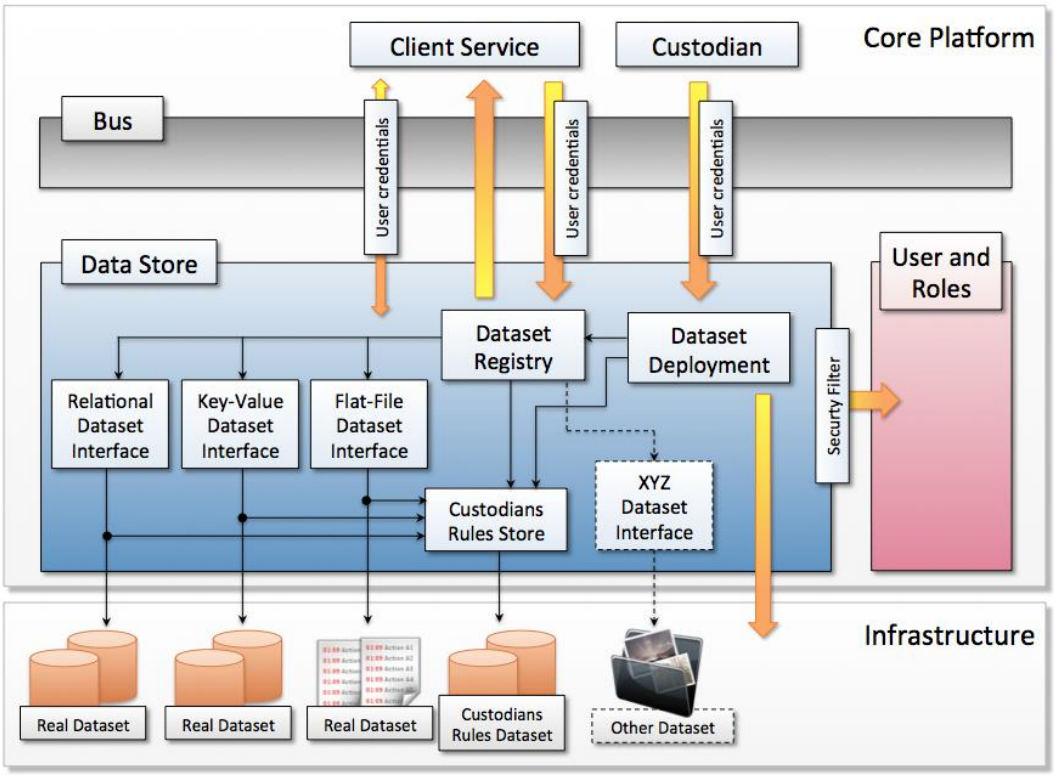

Figure 10 Data store conceptual model.

The diagram depicted in Figure 10 provides a conceptual reference model for the organisation of the data store and its interaction with other components in the system. The main access point to the store is the *Dataset Registry*. This service contains the information about all the registered and available datasets in the store. A client to the data store gets access to the *Dataset Registry* through the ESB and by querying the registry obtains a service address, together with metadata about the interface used to interact with the dataset and the information on the specific dataset selected. Access to this information is granted by checking the user credentials against the security infrastructure. A request should only be served if the attached credentials are validated against the security infrastructure and the identity associated with them covers the roles required by the custodian of the dataset. The information that maps the operations against the datasets to the roles

(for each of the datasets) is managed in the *Custodians Rules Store*, which validates every operation performed on the datasets.

Once a client has received the information required to access a dataset it will be able to contact the appropriate interface through the ESB. The preferred implementation of these interfaces is by means of (web) services accessible through the ESB. In cases when the nature of a specific dataset type makes inappropriate a web-service interface different interface (an perhaps more effective for the specific case) should be considered. The access to the dataset is filtered by the interface returned by the service registry and, as happens for the dataset registry, the access is filtered by interacting with the *Custodians Rules Store* and the security filter that checks that the given credentials have the required roles.

Another component of the data store is the *Dataset Deployment* service, which allows clients to load and register new datasets into the store. The custodian performs this operation by specifying the type of datasets and the set of rules for its access. The deployment service will also provide interfaces for loading/importing the dataset into the store. As happens for the interaction with the stored datasets, these operations are subject to security restrictions that depend on the roles of the identity associated to the client interacting with the Dataset Deployment service. For instance, it is unlikely that a client accessing the service under a set of credentials having the sole role of *citizen* could have rights to register a new dataset or import data into an existing dataset. A different case is then given by the staff of the BoM that can be allowed to define a new dataset regarding information about weather.

The architecture of the data store is modular to allow the integration of new types of dataset over time, possibly via a plug-in mechanism, that allow authorised entities to deploy and register new interfaces so that new capabilities can be supported.

9.2.4 Services Catalogue

A services catalogue is an integrated information infrastructure from which users can locate resources and services available in VINE. Each accessible resource is made available through the services catalogue with appropriate metadata describing its characteristics. For instance, the Data Store, previously described, will be made available through the services catalogue as well as any other service that is intended to be accessible.

The core capabilities delivered by this service are:

- Locating the available services in the platform. The services catalogue is used to store information and provides access to network resource information without the client having to know the specifics of how or where the resource is physically connected. The information appears as a single database although it may be stored among different physical locations. This could include completely external systems such as those operated by the Bureau of Meteorology.

- Acting as a services catalogue that provides metadata about the available services. The services catalogue exposes information about users, applications, file and print resources, access control, and other resources, into a service that is available to users and applications.

The metadata exposed by the services catalogue needs to provide sufficient information for clients to connect to the services they are looking for (service address, format and protocols required for the interaction).

The services catalogue generally has the following characteristics:

- Providing a convenient way to access the full information about the capabilities of a system. For example, the Domain Name System (DNS) maps the computer host names and other forms of domain name to IP addresses. With the name service type of catalogue, users can locate computers on a network and access resources without remembering complex numerical IP address.

- Providing a clear and current picture of all the existing services. The services catalogue unifies all the network resources and enables clients or applications to think in terms of the entire network instead of the individual servers. Also, the services catalogue supports the update functions in order to increase availability and reliability.

- Providing a unified way to access the information, this way does not change if the capabilities of the system change. The services catalogue defines the namespace for the network. It ensures each entity has a unique and unambiguous name through a set of rules that specifies how network resources are named and identified. The services catalogue maintains the correspondence between name and address. When a resource changes, the administrator simply changes the resource's address on its object in the services catalogue's database and keeps the resource's name the same. Therefore, network users and applications only have to know the name of any resource they need as the servers have the information in order to locate the resources.

- Facilitating the discovery of capabilities by offering multiple types of searches. The services catalogue provides robust and flexible ways to find data and acquire information, allowing searches on individual attributes of entries. It supports a range of different types of searching and matching on entries, such as word matching, stem matching, component matching and so on.

The services catalogue will manage and update all available resources stored in VINE. As part of the core platform components of VINE, it provides all users and applications with a single, well-defined and standard interface to access the information. The services catalogue interface is not only used internally but also supports access for external clients.

The diagram showed in Figure 11 describes an interaction between the services catalogue and external clients outside VINE. Imagine that such a client wants to access a dataset stored in VINE. In order to locate the service that will provide access to this dataset, the client queries the services catalogue, which will respond by informing the client about the information that can be accessed. More precisely, the client will use the catalogue interface API, which eventually sends the request to the services catalogue through the ESB. The services catalogue will resolve the query requested by the client and will send back information about the data store. From there on, the client will be able to interact with the data store through appropriate APIs.

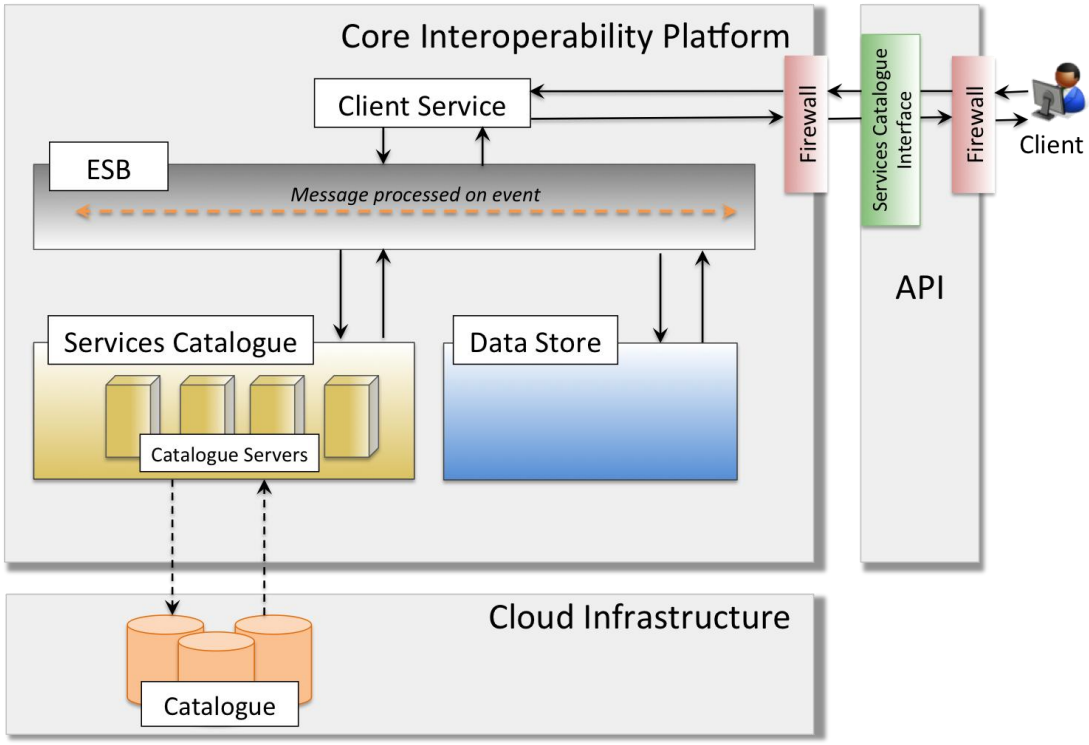

Figure 11 External access to services catalogue.

A similar interaction occurs when a service hosted in VINE will require locating one of the available services needed for its activities. The diagram depicted in Figure 12 provides an interaction between the services catalogue and other internal components in the system. A weather model is one of the modelling components deployed in the services layer. The model will provide weather forecasts and current weather information that will help VINE to provide support for decision makers. Suppose the model will access a dataset stored in VINE. In order to locate the accessible resources, the weather model will interact with services catalogue through sending message via ESB. The services catalogue will process the message received from the weather model and send back information about the data store. Then the weather model is able to obtain the available resources by interacting with the data store.

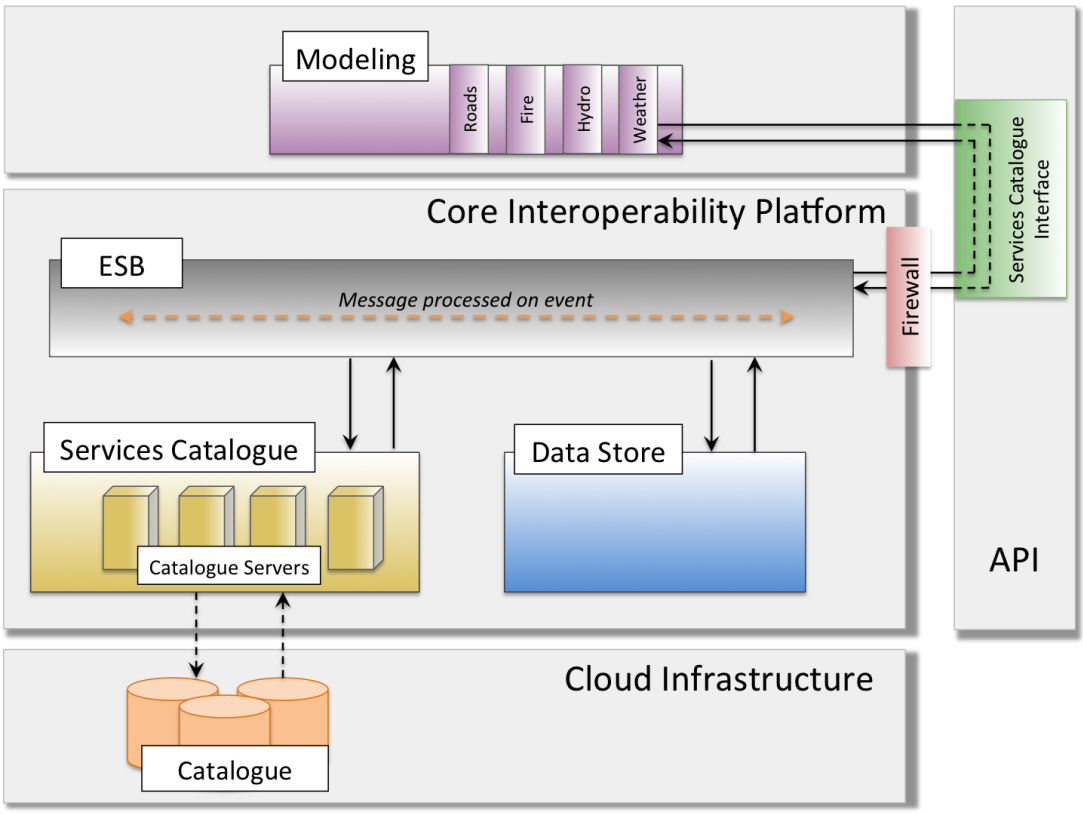

Figure 12 Internal access to services catalogue.

In order to make scenarios happen, the services catalogue will need some sort of data store capability to maintain all the metadata about the available services. It is expected that it will leverage the service offered by the data store. Therefore, during deployment and then its activation, the services catalogue will keep a reference to the data store and to the specific interface required to access its data. This information is obtained through either configuration or interaction with the ESB. During operations, the services catalogue will simply use the cached interface to retrieve the required information.

In order to provide a robust services catalogue, some other requirements are specified as follows:

- **Replication and Redundancy:** The services catalogue leverages replication and redundancy to avoid the loss of a single server causing the service to become unavailable. A reliable replication and redundancy ensures the most recent data is available to the clients and will continue to be available even in the event of failure.
- **Scalability:** The services catalogue provides a centralised management of network resources. These resources are distributed across multiple catalogue servers that hold different namespaces and are interconnected to form a distributed catalogue service therefore increasing its scalability.
- **Load Balancing:** To maintain the ability to read data in the catalogue, a suitable load balancing strategy must be put in place. Load balancing provides a solution for distributing the load across multiple replicated servers. The solution can also monitor the state of each server and to manage its participation in the load balancing.

9.2.5 Logging System

Essential for VINE is the capability of logging a diverse and evolving range of information that includes behaviour of users and services as well as other events that might be determined by different conditions at different points in time. In order to accommodate these capabilities the logging service should be flexible enough to allow VINE administrators to finely tune what information is recorded. This goes beyond having a basic logging service that collects the information sent to it by other services, but requires also additional intelligence that is able to capture information originally not meant to be logged. This functionality is necessary because VINE is intended to be an ecosystem and therefore it is not possible to define a priori which type of information can be relevant and that need to be recorded. To achieve this we suggest the logging system to be composed by two modules: a logging service and an event listener.

9.2.5.1 Logging Service

The logging service subscribes to a specific topic on the publish/subscribe event service (e.g. 'logging' topic) and logs all such messages. It enables applications to send formatted information to one or more output destinations such as consoles, files, databases and so on. Such functionalities are similar to the Java JSR Logging API (Oracle America, Inc. , 2002), which is implemented by multiple Java logging frameworks.

The service can be configured to collect information at different severity levels for different modules. The output data is stored in log files that can then be leveraged by administrators for tracking users' actions, analysing traffic patterns, auditing the system usage and reviewing troubleshoot.

As a services-based and unified information platform, VINE is able to sustain a diverse ecosystem of users and applications, and is capable of automatically filtering, processing and analysing data for supporting decision-making. The logging service generates an audit trail that can be used to understand the activities of the system and the interaction between applications and users, as well as an information source to diagnose problems. We can make use of logging service to:

- *Logging messages in a file under a specify directory*. The logging service will create audit logs to record logging-type events and store the formatted messages to multiple output destinations through log streams for keeping track of various actions taking place in the system. For example, since VINE have a role-based access control system built in, the event of handing over duties from one person to another should be permanently logged in order to facilitate reconstruction of users' responsibilities at any given time.

- *Logging messages based on their severity level and formatting events into certain types.* Logging requests may come from any services in VINE directly or from the event listener and contain a severity level. In order to meet different requirements, the logging service provides a configuration of

logging threshold that is a level of filtering. The log file will be generated by the service when its request with level higher than the threshold. Once at the output destination, a log record is subject to output formatting rules, which are configurable and public.

9.2.5.2 Event Listener

The event listener is designed to allow certain types of events to be logged even though the source of the event has not issued an explicit logging call. The event listener will function by eavesdropping on all events (regardless of type) in the ESB, such as access denials and approvals, authentication and authorisation events, and so on. The event listener will use a series of rules to convert some of these events into logging-type messages that are then handled by the logging service. Every time a service in VINE generates an event, it communicates to the listener following the rules configured in the rule engine that is designed for declaring the desired information the listener will report. If the event satisfies the requirement of predefined rules, it is then transformed into a logging-type event and it will be sent to the logging service.

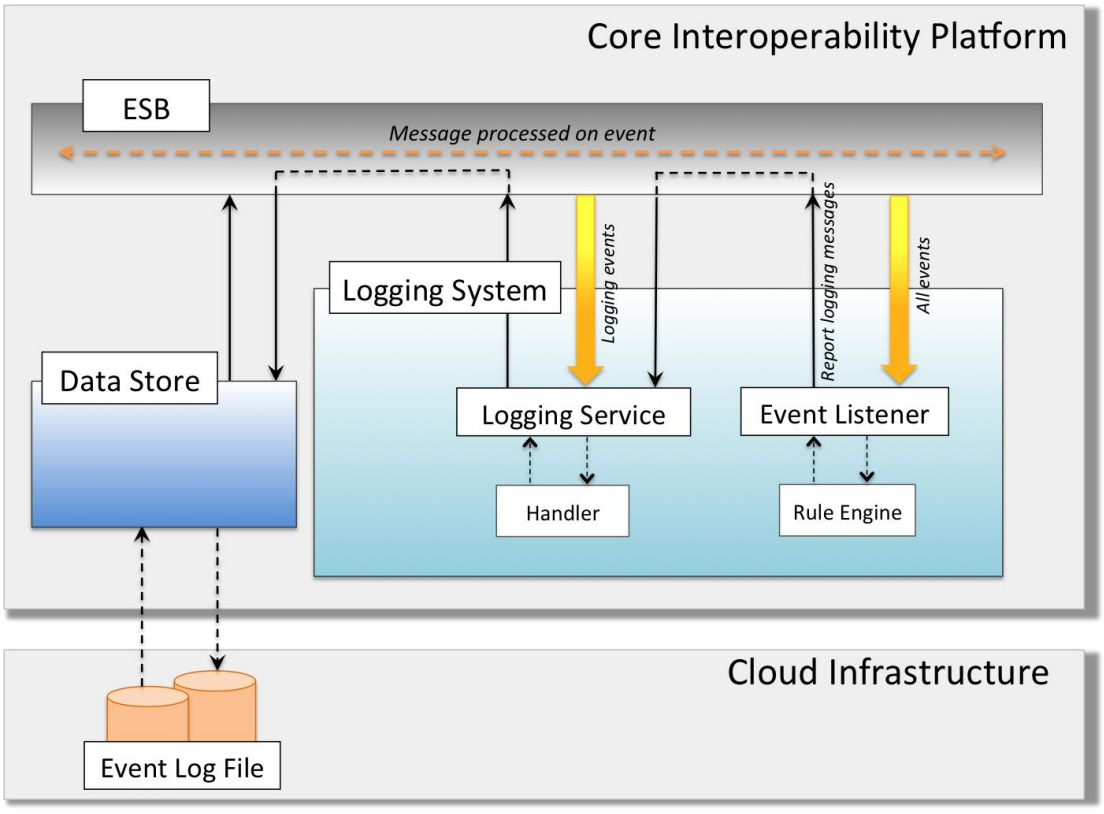

Figure 13 The logging system.

The diagram showed in Figure 13 describes an interaction between the logging system and other internal components within the system. The logging system consists of two modules: a logging service and an event listener. The logging service only captures the logging-type events through ESB and sends them to the data store after the handler formats the messages. These logging-type events are generated either directly by services in VINE or by the event listener. The event listener listens to all types of events occurring in the system through ESB. Once it

receives information, the rule engine will be triggered and filter the messages for particular events according to a series of rules. The listener then converts the filtered messages into logging-type events and interacts with the logging service through ESB for recording of the logging information.

9.2.6 User Management

The goal of the user management system in VINE is to support a dynamic and seamless integration of services and resources in VINE and ESOs with users by:

Providing a mechanism to identify and validate users from different ESOs, using a single identity for each end-user who is recognised across all the federated members

Providing each authenticated user with seamless access to resources within that trusted federation based on their attributes and authorisation rules

Providing mechanisms for dynamically propagating any change of user attributes to the ESOs to streamline access to federated resources

This separation of authentication from the applications themselves allows for greater flexibility to support users logging into applications with a single username and password as long as it is from a trusted Identity Provider.

User management in VINE will be based on a federated identity model. Federated identity technology allows organisations using disparate authentication and authorisation methods to interoperate, extending the capability of each organisation's existing services rather than forcing their replacement. The result of this functionality is that users are able to seamlessly access resources provided by multiple federation partners.

Within a federation relationship, the majority of the traditional user management functionality will now be retained by the Identity Provider, i.e., VINE. As an end user's Identity Provider (IdP), VINE will be responsible for the identity lifecycle management of this user including account creation, user provisioning, password management, and general account management. Having a single point for identification relieves the federation members of the burden of managing equivalent data for the user. However, each member may still manage local information for a user, even within the context of a federation, but are in synchrony with VINE in terms of transactional attributes, such as access privileges. Each ESO will leverage their trust relationships with VINE to accept and trust information provided by it on behalf of a user, without the direct involvement of the user. This trust relationship simplifies the creation and maintenance of federation trusts as each ESO must only configure their applications to trust one Identity Provider, which is VINE.

For enabling federated identity management, there exist standards and specifications such as SAML, WS-Federation, OpenID, OAuth, that include an aspect of session lifecycle management as well as single sign-on enablement through account linking. The sharing of user information across federation partners

is based on the secure exchange of tokens, or assertions, referring to a user, their attributes, privileges, etc.

Identity Federation has following solution areas:

- Identity lifecycle
- Application based Web Services Security
- Single Sign-On

Identity Lifecycle refers to the ability to manage the comprehensive phases of an identity, from login through termination.

Single Sign-On (SSO) is an access control mechanism that provides authentication of a user's access across multiple software systems and different services based on the user's permissions, while reducing extra logins when the user switches applications within one session.

WS-Security is a flexible and feature-rich extension to SOAP to apply security to web services. It is a member of the WS-* family of web service specifications. The protocol specifies how integrity and confidentiality can be enforced on messages and allows the communication of various security token formats, such as SAML, Kerberos, and X.509.

Utilising the user provisioning, role governance, and service capabilities within the federation, VINE should take control of defining and automating multiple processes that allow individuals and services with identities to be appropriately handled according to resource access policies.

An Example Scenario

Tim (a fictitious character for illustration purposes only), an ESTA Planning Operations Manager, is responsible for assessing and authorising dispatch of ESO resources to the emergency incident sites. He needs to get a clear picture of the number of resources being dispatched from each ESO and their current status in the field before making a decision to dispatch more resources. From his office in Warragul (rural Victoria), he has access to each of the software systems provided by the ESOs that keeps track of their individual resources in the field.

Enabling Tim to access all of the systems in a manageable way is a federated identity system that includes all the ESOs as members. Under the system, Tim maintains a single digital identity with VINE. When he accesses any ESO system, he logs in using VINE's username and password, and VINE, his identity provider, confirms to the ESOs that his credentials are legitimate. Tim does not need to create dozens of user accounts and passwords for the ESOs or for services that VINE itself provides. The member organisations, here each ESO, do not need to maintain separate accounts for Tim. Tim later takes advantage of the federated identity system to share dispatch data with other personnel from various ESOs, by simply updating their roles to access his data, in VINE. VINE propagates these changes to each ESO transparent to the user. And because commercial providers

(e.g. Active Directory, Google, Facebook, Windows Live, etc.) are also members of the federation, Tim can access VINE and ESO services as well while maintaining a single digital identity. Likewise, a normal citizen having access to the federated identity but only with the access rights for reporting incidents, could use his/her digital identity, which is trusted across all the identity providers, for reporting emergencies to VINE without having access to its restricted resources.

9.2.7 Security Services

Effective security practices must pervade every aspect of the development and deployment of VINE: physical security of data centres, fault tolerance, proper firewalls, data encryption, data retention mechanisms, authentication mechanisms, password policies, user education, and security-minded software development practices such as input sanitisation all must play their part in ensuring that VINE remains secure. In addition to these, VINE's core interoperability platform will host a number of systems and services whose role it is to bolster the security of VINE.

- Intrusion Prevention Systems (IPSs). IPSs can be deployed at the network and host level. Network IPSs are widely used because they are easy to install – they typically come as an appliance – and can be placed at selected, strategic network locations. Network IPSs are of reduced use if the network traffic is encrypted; solutions that try to circumvent the problem first decrypt the traffic, send the plain network data to the IPS, and encrypt the network traffic again. In today's data centres where virtualisation plays a dominant role, one also has to ensure that IPSs are able to analyse the traffic between virtual machines, even if this traffic happens within a single physical box that hosts multiple virtual machines.

- Data Leakage Protection (DLP): DLP is a special form of network monitoring. DLP products ensure that no sensitive data is sent – maliciously or accidentally – to a remote location. This also implies that there is a definition of what sensitive content is, and that on the fly documents can be checked for sensitive content.

- Usage control. Identity and Access Management are cornerstones of a good security architecture. While it is key to properly design and implement who has access to what information, simple access control is no longer sufficient to protect data. Usage control monitoring systems analyse access logs and look for deviations from normal or expected behaviour. For example, if a user accesses more documents than they usually do, this can be flagged and reported. The idea is to detect misbehaving users but also intruders as early as possible.

- Network Antivirus (NAV): It is important to protect VINE against viruses that may try to infect services running within VINE. In addition to local antivirus protection, VINE should also have network-based antivirus capabilities. Some IDS/IPS provide virus protection. Firewall services with deep packet inspection facilities can also intercept ingress and egress of viruses and other malware.

- Denial of Service (DoS) Mitigation. Denial of service attacks against VINE cannot be prevented, but a number of measures can be taken to mitigate their impact (Societe Generale, 2011). These include appropriate configuration of firewalls and routers, and vigilant systems administration that can manually respond to DoS attacks by blacklisting certain origin addresses, for example. However, more sophisticated attacks can be mitigated by routing services that efficiently detect and filter bad packets before they can deny service at the application layer; in many cases such services use dedicated hardware (Cisco Systems, 2010). It has to be mentioned that in case of severe DoS attacks close collaboration with the network service providers is needed. The goal is to stop DoS attacks as close to their origin as possible.

- Security Information and Event Management (SIEM). An increasing number of security solutions are deployed in enterprises. In order to avoid that the devices operate in silos, it is important to correlate the information provided by the various security solutions and to have a single console where the security operator gets on overview of the overall security status. SIEM products allow one to collect security information from a wide variety of sources and to automatically analyse and visualise the collected information.

None of the services mentioned in this section are a substitute for good security practices elsewhere in VINE, and although they will play a role in keeping VINE secure they should not be relied upon to detect malicious behaviour in all circumstances.

9.3 Services layer

VINE's services layer will function as a cloud infrastructure for hosting software services that extends VINE's capabilities. The core interoperability platform will provide access to data and a basic set of services on top of which new emergency management capabilities can be built. At the same time the virtualised cloud infrastructure will enable a great degree of isolation and loose coupling between the different services in the system. What separates the services layer from a simple cloud IaaS is the VINE API framework (see Section 9.4), which when combined with good governance (see Section 9.4.4) will allow a coherent and consistent API to be provided for accessing the multitude of services hosted in this layer.

It is anticipated that the variety of services found at this layer will grow as uptake of the system continues, and that it will host contributions from a variety of stakeholders. Typically a new service will at first be developed and hosted by a third party, accessing data and other VINE services using the web services API. In the case where such a service has been in use for some time and has proven to be useful for stakeholders beyond the original developer, it can be migrated to the services layer and assimilated as a component of VINE. The configurable operating environment afforded by cloud technology, combined with the fact that VINE's API

can be accessed from any Internet-connected device, will significantly reduce the cost of adding externally developed features and off-the-shelf components to VINE. Although new services will have to undergo quality-control processes and be harmonised with VINE's API, no code changes or porting activities should be necessary. This is critical to allowing VINE's ecosystem to feed back and strengthen the core system.

9.3.1 Types of Services in VINE

The services layer is designed as an open platform that will continue to host new functionality over VINE's service life. As such, it is neither necessary nor possible to provide a complete enumeration of the services that may be hosted in this layer, as it depends very much on future priorities and needs. However, this section briefly describes some of the types of services that may prove useful in extending the core functionality of VINE, as illustrated in Figure 14. However, this discussion should not be regarded as exhaustive, nor should it be interpreted as specifying a set of priorities for this layer.

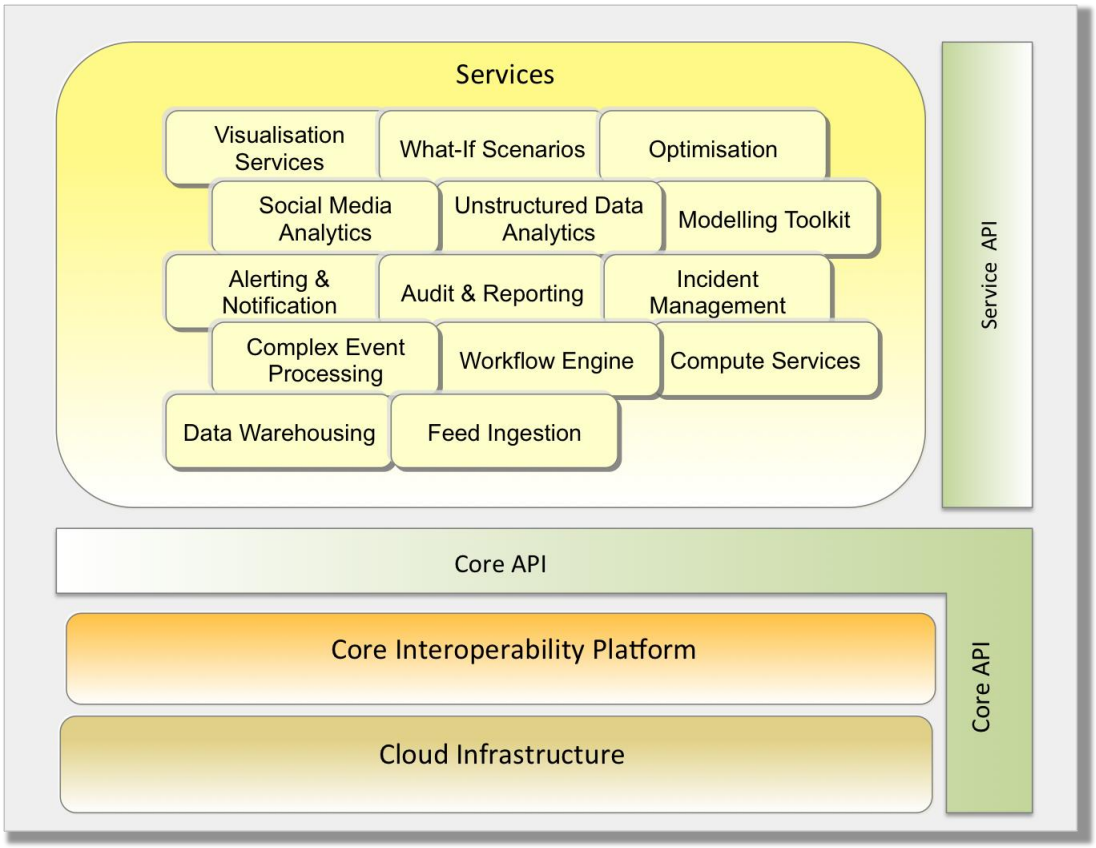

Figure 14 VINE's services layer.

- **Modelling Toolkit & Service.** Modelling provides a mathematical or computational representation of the real world that can be used to improve prediction and understanding, allowing better decisions to be made. Examples are models of runoff that improve flood prediction, models of human behaviour that can help in evaluating the effectiveness of a potential evacuation, and, weather forecasting models. Models can enable 'what-if'

evaluation, where the consequences of various actions can be predicted in order to choose the best one. Many different models can be built using the same underlying set of techniques; in order to make it faster and cheaper to incorporate new models into VINE, it may be valuable to build a modelling toolkit and service. Such a toolkit could allow users to define models on a common platform, as well as provisioning a service that could take care of many of the details of data consumption, computation, connecting different models, and so on.

Optimisation Toolkit & Service. Optimisation tools attempt to find the best solution to a problem from a set of possible solutions. For example, optimisation can help determine the best locations to deploy appliances in order to minimise property damage. In some cases, modelling and optimisation tools work together, as would be the case when defining an optimal evacuation plan given a model of the local road network. As with the modelling toolkit and service, it may be valuable for there to be a toolkit and service for optimisation on VINE such that the many common and operational aspects of different optimisation processes can be taken care of, reducing the time and cost needed for bringing new optimisation models into VINE.

- **Workflow Engine.** The VINE architecture will enable automatic filtering, processing, and analysis of data in a way that supports more informed decision-making. However, human judgment will still be critical in evaluating and classifying information in many instances. For example, fire-spread forecasts may need to be assessed for veracity and plausibility by experienced personnel prior to being published and acted upon. VINE will need to incorporate appropriate workflow capabilities that can be used to employ the skills of experts in classifying and verifying information as it becomes available. This capability will be supported by the workflow engine component of the services layer and should: (i) expose a workflow specification API where the manual processing steps for particular types of information can be defined, including actions to be taken in cases of timeouts or if qualified personnel are not available, (ii) allow dynamic reconfiguration, enabling both the role information and the workflow specifications to be changed at any time, with the changes reflected immediately in the execution of the workflow, and (iii) provide suitable integration with the publish/subscribe event service to make the definition of workflows straightforward.

- **Data Warehousing.** Each stakeholder may have particular analytics and data mining needs that would best be served by a data warehouse. Although VINE will not have a data warehouse of its own due to the differing needs of its many stakeholders, its open API will make it possible for agencies and other stakeholders to build warehousing utilities that export the necessary data to from VINE build private data warehouses. This will be subject to governance constraints and oversight to ensure that such

warehousing doesn't have the potential to violate privacy or confidentiality conditions.

- **Unstructured Data Analysis Service.** Data from social media feeds, peoples' phone calls to the emergency agencies, videos and images of disaster-affected areas are all examples of unstructured data: data that is irregular and free-form and therefore difficult for computational processes to use directly. VINE will need to be able to process such unstructured data from various sources in order to provide the best possible situational awareness to all stakeholders; therefore, VINE must the capability to process and analyse these types of data. Techniques such as data mining, text analytics, entity detection, speech recognition and object recognition can be used to find patterns in and interpret various types of unstructured information. An unstructured data analysis service could provide a set of APIs that allow unstructured data to be marked up. For example, submitting a document to the service may return key terms from the document; submitting an audio file may return a transcript of the speech in that file.

- **Feed Ingestion Service.** Services built on VINE will need to operate with different types of data from multiple sources such as social media feeds, newswire feeds, sensor feeds, weather and traffic model feeds and so on. In many cases, a feed ingestion process will need to be defined that checks for new data, converts it into a suitable format (in some cases this will require unstructured data analysis), and publishes it to the appropriate topic within VINE's publish/subscribe event service. A feed ingestion service would provide a framework to handle much of this process. New ingestion jobs could be configured in terms of what data to ingest, where it comes from, the protocol by which the data must be retrieved, the frequency of polling for new data, rules for its transformation and the topic to which it should be published.

- **Social Media Analytics Service.** Social media has proven to be a great resource for sharing and obtaining information during disasters, as it provides the opportunity both to reach and gather information from many people in real-time. However, though it has the potential to provide great value, social media as a source of data is unstructured and untrustworthy. A social media analytics service would build on the unstructured data analysis service to provide analytics tools that are specifically useful for social media. Services provided for social media analytics could include emergent topic detection, trust and reliability analysis, sentiment analysis, opinion analysis and graph/network structure analysis. When combined, these services would be able to overcome the shortcomings of social media data and transform it into a useful input for decision making for emergencies.

- **Visualisation Services** One significant feature of VINE is that it will be used by different groups of people for different purposes. The same information may need to be visualised differently depending on the use to which it is being put. Therefore, it would be useful for VINE to support a variety of services to assist with data visualisation. Such services could

include generic services for mapping and charting, as well as other services that might provide specific visualisations for particular applications. An example of the latter could be a service that plots current flood conditions on a map, or that maps a predicted smoke plume (that has been modelled by a separate service). In order to support different client applications, visualisation services would need to provide visualisations in different formats such as markup (KML [8] , for example), vector and raster representations.

- **Alerting and Notification Services.** VINE must be able to provide alerting and notification services across multiple channels, and to integrate with existing systems that provide similar functionality. Alerting and notification services should have the following characteristics: (i) gateway functionality between VINE and external systems, especially with respect to the publish/subscribe event service (ii) support for alerting across multiple channels: telecom recorded message, SMS, sirens, radio, email, etc. (iii) specific accessibility support to ensure that vulnerable community members do not miss out on alerting and notification, and (iv) support for multiple languages: the service should facilitate the issuing of alerts in multiple languages in order to reach members of the community who are not proficient in English.

- **Compute Services.** VINE services will operate using cloud computing resources provided by the underlying cloud infrastructure of the platform. This means that service owners provision machine instances and operate software on them, interacting via the API with other services. In some cases, however, it might make sense to provide computation environments as a higher-level service that can be utilised directly to execute certain types of processes. This is particularly true when the cost of setting up and maintaining the environment is high, so that such overheads can be shared across many users. In the case of large-scale parallel computation this is often the case, as users typically wish to use a large, carefully-configured computational resource for a relatively short period. Amazon Web Services (AWS) provides an Elastic MapReduce service for such parallel computation (Amazon, 2012). It might be valuable to build a similar service for parallel computation, stream computation, and other specialised computation models within VINE.

- **Incident Management Services**. At present, many agencies operate independent Incident Management Systems (IMS) for situation management during an emergency. A barrier to interoperability is that each agency's IMS typically draws from its own store of data, which may be different to and even inconsistent with the data used by another agency's IMS. Over time, it is desirable to draw much of the data, processes and business rules used to drive incident management into VINE so that a common operating picture is shared by all agencies' IMS in an emergency.

[8] See section 9.3.2.2.1

Incident management services encompass the variety of services necessary to keep track of incidents within VINE. Examples of services that fall within this category include incident reporting, duty cycle management, personnel and appliance tracking, CAD gateway, AIIMS automation and so on.

- **Audit & Reporting**. VINE's logging system (see Section 9.2.5) will be able to store detailed records of events within VINE, including assignment of roles and responsibilities, data access, service invocation, administration activities and so on. Such records will be very useful both for producing reports summarising various aspects of system use and performance, as well as for any routine or forensic auditing activities that may take place. Audit and reporting services will support these activities by, for example, producing certain types of reports on demand or performing certain types of searches that are useful for auditing purposes.

9.3.2 Additional Services

In addition to the services listed above in Section 9.3.1 above, several suggested services require more detailed discussion, which takes place in the following subsections.

9.3.2.1 Complex Event Processing Service

Complex event processing (CEP) deals with evaluating multiple events with the goal of identifying the meaningful characteristics within an application. As there could be multitude of event types and each of them may occur over a varying period of time, CEP provides the functionality of filtering, aggregating and pattern matching for event co-relation analysis. CEP is also useful for various analytics activities such as event tracking, alerting, prediction and forecasting, and so forth in addition to event data processing. For this reason, CEP's functional characteristic aligns perfectly with VINE's requirement for extracting information out of streams of events occurring in real-time.

Figure 15 depicts the building blocks of CEP that are embedded as part of the VINE architecture. Its components are spread across services, core interoperability platform, and services API. The data store component present at the core interoperability platform is used for storing event processing rules and their metadata, as well as for caching information useful for correlating with future events. At a higher level, the components present at the services layer carry out series of analysis and record on incoming stream of data. Invocation of appropriate services is coordinated by the service invocation logic that is constantly monitoring activities of the event processing engine, correlating matched patterns with rules, grouping events that match certain patterns, and mapping them to any pre-defined business workflows. The invocation of these pre-defined business workflows will result in one or more or combination of services mining the events of interest.

Using the services API, information producers and consumers are given access to development and management tools that help them in defining CEP functionalities, such as event matching rules, service invocation workflows, information visualisation, real-time update for event subscribers, and so forth.

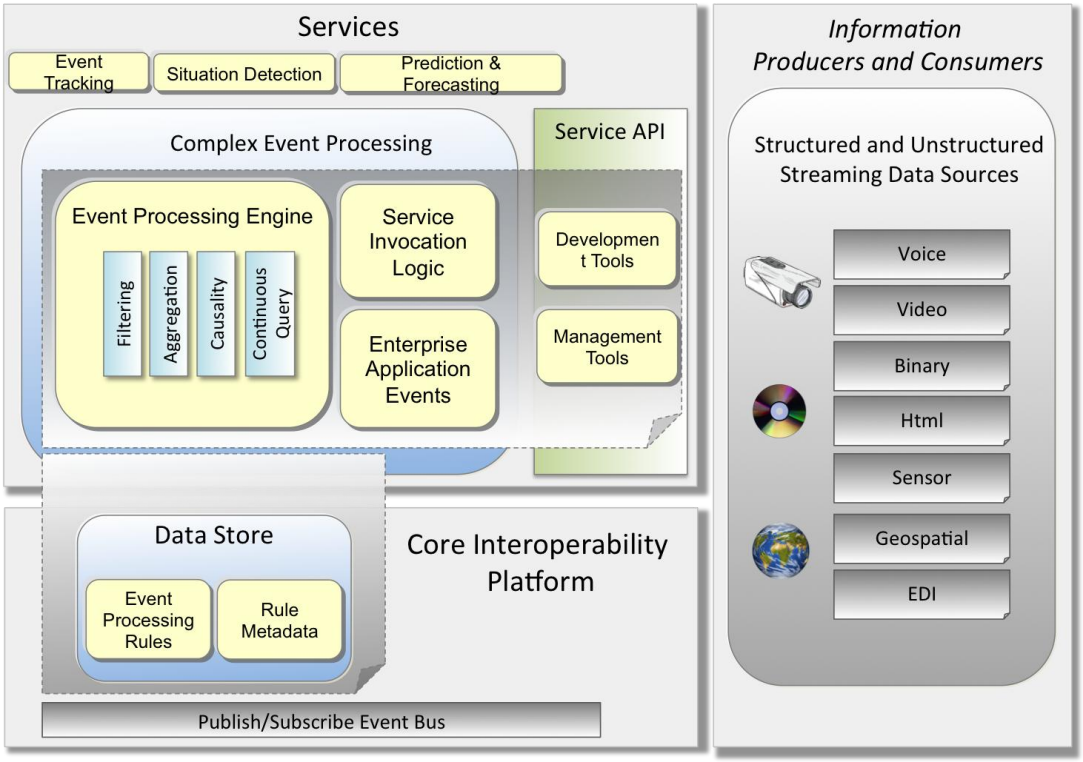

Figure 15 Integration of complex event processing component in VINE.

The capability of CEP that VINE would benefit the most is its ability to analyse incoming data in real-time. By the employment of sophisticated event interpreters, event-pattern definition and matching rules, and correlation techniques, it has the capability of correlation of events across multiple sources, correlation of events across time and space, and identification of interesting events that match a defined pattern. This feature could be applied to disaster management, and more specifically for early detection and warning systems. Relating to the scenario of "Fire in the Dandenongs", CEP real-time analysis could correlate the altered forecast to the controlled burn regions operated by DSE and issue a warning to CFA, which in turn classifies the fire from controlled to "Severe". The monitoring tools used by emergency services personnel at the burn region are connected to VINE via the services API, which provide them the warnings in real-time.

The aggregation capability of CEP is associated with combining data from multiple sources, including the ability to combine streaming and static data spanning large time windows, producing high-level summary and statistics. Taking the reference of the scenario of hazardous chemical crisis in the city, CEP could be used to assist the traffic model for creating diversions. It would aggregate the streaming data obtained from current traffic conditions (derived from VicRoads SCATS data) and historical data to generate probabilistic values of traffic flow that could be used to generate the set of traffic diversions and predict the disruption to the local traffic.

The publish/subscribe event service plays a pivotal role in integrating CEP functionalities with the rest of VINE. As the most important function, it is associated with the routing of messages arising from CEP components to services and end-points responsible for taking actions on them. It also helps mediate communication

protocol between various loosely coupled components, by readily transforming message formats and reliably delivering messages with or without transactional integrity.

Seamless integration of CEP into the VINE's platform will help decision makers by allowing them to benefit from real-time analysis of structured and unstructured streaming data as soon as it becomes available.

9.3.2.2 Geospatial Services

Geospatial data are becoming an increasingly important component in decision making processes and planning efforts across many information sectors, and especially for emergency management. For instance, visualisation of geospatial information derived from satellite remote sensing and combined with other available sources of data could provide valuable situational awareness.

A broad range of information types, including geographic data, remote sensing imagery, three-dimensional object representations and other location-based information, characterises the geospatial domain. These information types span a wide variety of data structures: vector and raster; unstructured and topological; discrete and continuous. In order to manage these geospatial data types, as well as to integrate them with non-spatial data, VINE will need services capable of collecting, storing, managing, retrieving, analysing, and portraying them.

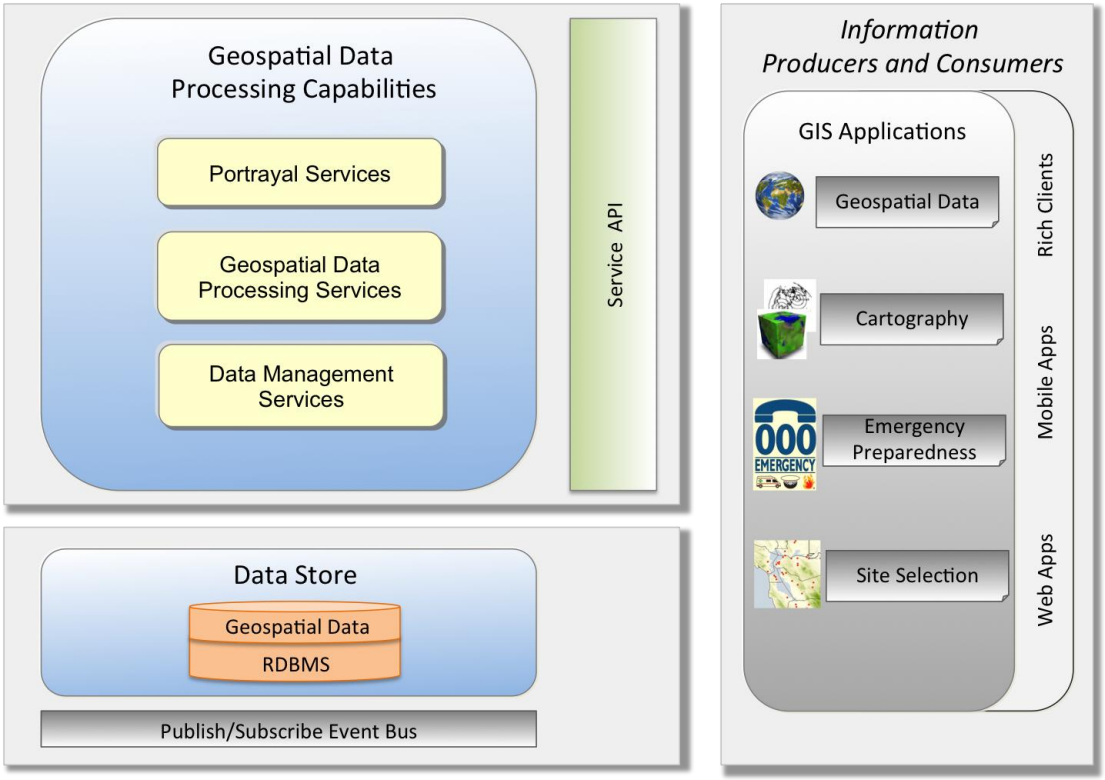

Figure 16 Geospatial data processing capabilities in VINE.

In general, the necessary geospatial data capabilities of VINE can be grouped into the following categories:

- Data management services
- Geospatial Data Processing services
- Portrayal services

There are at present two primary geospatial standards efforts led by the Open Geospatial Consortium, Inc. (OGC) (OGC, 2012) and the International Organization for Standardization (ISO) Technical Committee 211 (TC211) (ISO, 2012).

OGC is leading the development of standards for geospatial and location based services to develop publicly available interface specifications. OGC specifications support interoperable solutions that “geo-enable” the Web, wireless and location based services, and mainstream IT. On the other hand, ISO/TC211 propose a standard framework for the description and management of geographic information and geographic information services. OGC and ISO/TC211 work in conjunction in order to avoid duplication of efforts.

9.3.2.2.1 Data Management Services

Data management services are responsible for geospatial data import/export from repositories and databases; spatial data Extraction, Transformation, and Loading (ETL); and providing catalogue services. In simple terms, these services facilitate access to collections of content stored in the data store of VINE. They are supported by catalogue services, which provides the ability to publish and search collections of descriptive information (metadata) for data, geospatial processing services, discovery and binding of end-users to those services, and related information objects.

Storage of and access to spatial data is managed using the support of relational databases. Simple Feature Access (SFA; also called ISO 19125), is both an OpenGIS (OGC AS, 2012) and ISO Standard that specifies a common storage model of geographical data (point, line, polygon, multi-point, multi-line, etc.) (ISO, 2012). The geometries are also associated with spatial reference systems. The data storage and management framework found in a variety of commercial and open source software implement the OGC and ISO SQL data type for spatial. Data management services provide the benefits of a relational database, such as scalability, reliability, security, backup, and integrity. Moreover, DBMSs are efficient at moving the type of large binary objects required for geospatial data.

In addition to SFA, the OGC Web Feature Service (WFS) specifications facilitate dissemination of geospatial data in XML format. To provide access to vector data, OGC defines an XML grammar, the Geography Markup Language (GML) (OGC GML, 2012). GML serves as a modelling language for geographic systems as well as an open interchange format for geographic data related interactions on the Internet. Keyhole Markup Language (KML), a descriptive language developed by Google, complements GML as a format for describing, storing, and visualisation of geographical information, including three-dimensional objects.

Using these standards, geospatial data in VINE could be made interoperable and accessible to Web clients (REST, SOAP, XML, KML, etc.), OGC compliant services

(such as GML, WFS, WPS, WMS, etc., as described in earlier sub-sections), for enterprise integration (SOAP, XML, EJB, SQL, etc.), or to rich clients and standalone applications.

9.3.2.2.2 Geospatial Data Processing Services

One of the key functions of geospatial data processing services is to integrate geospatial and spatiotemporal data with other available sources of information. The fusion of these data sources significantly enhances the ability of a decision maker to comprehend changing situations and make decisions based on the dynamic nature of various data sources over a given time period.

VINE data processing services will provide one or more access mechanisms by which a client may submit a processing task to a server to be completed. The services will provide access to its “server instance” as an entity, which may provide data processing service implementations in the form of processes or tasks. In this manner, any given server may be able to perform multiple processes on behalf of the requesting client. The processes could include implementation of any algorithm, calculation or model that operates on spatially referenced data, such as site selection models, dispersion/plume models, road network analytics, raster analytics, image processing, and so on.

For emergency management, a decision maker may want to integrate and utilise various models and processing services to search for best options for the near real-time dispatching of resources. For instance, the decision maker could execute flood and traffic models to find the optimal solution under the constraints of a flooded area and a near-real-time traffic report, resulting in the final shortest path being obtained. By using the capability of data processing services in VINE, the decision maker could make a rational decision for dispatching emergency vehicles on the shortest path obtained.

This capability conforms to the OGC Web Processing Service (WPS) (OGC WPS, 2012), which is a specification for exposing geospatial analysis operations through web services. WPS does not specify the kind of processes that could be implemented as a web service. Instead, it defines a generic mechanism that can be used to describe and web-enable any sort of geospatial process.

9.3.2.2.3 Portrayal Services

Portrayal Services provide capabilities for visualisation of geospatial information. Portrayal Services are components that, given one or more inputs, produce rendered outputs (e.g., cartographically portrayed maps, perspective views of terrain, annotated images, views of dynamically changing features in space and time, etc.).

The portrayal service uses the functionality of data management and geo-processing services to transform, combine, or create formatted outputs. By offloading the generation of the visualisation of integrated geospatial data (i.e. vector data, satellite imagery, and raster maps) to the portrayal services, thin clients such as mobile and web-apps could simply display the superimposition of various geospatial and non-spatial data at their end in the form of tiled images.

The visualisation of transportation networks, demographics, physical environments, etc. on maps (2D) and globes (3D) are all examples of the capabilities of a portrayal service. With respect to the evacuation scenario at the Dandenongs described in Section 12.1, a portrayal service would be able to combine data pertaining to emergency resources dispatched, incident locations, weather conditions, hospital status, and so on alongside traffic flow, terrain, elevations, and map data, so that the entire situation is portrayed visually to stakeholders.

The portrayal service functionality should conform to the OGC Web Map Service Specifications (WMS).

9.3.2.2.4 Streaming Data Analysis

The interfaces exposed by the portrayal services or geospatial data processing applications require access to spatial databases to extract both geographic data and associated metadata. On the other hand, with the emergence of sensor networks as an important source of situational data, VINE will need capabilities to analyse the sensor observations in real-time or near-real time while also supporting analysis using spatial data repositories.

Standardisation for sensor enabling web services is still ongoing. OGC's Sensor Web Enablement (SWE) initiative is focused on developing standards to enable the discovery, exchange, and processing of sensor observations. It has proposed several encodings for describing sensors and sensor observations, through several standard interface definitions for web services. VINE could leverage the capabilities of any commercial off the shelf software suite that implements these interfaces for web services.

9.3.2.2.5 Extensibility and Customisation

In addition to the core capabilities listed above, there are efforts being made to standardise emerging geospatial technologies and tools. Although not part of the OGC or ISO/TC211 standards, more specialised extensions are available from software vendors for specific geospatial data processing needs, such as 3D mapping and analysis, distributed geo-processing, and so forth. Where necessary, these additional capabilities could be integrated into VINE's services layer.

Furthermore, any geospatial service integrated with VINE will need to provide the capability to integrate user-defined application specific logic and domain expertise. These extensions could be supported through existing APIs that integrate functional characteristics of several services. As an example, the catalogue specification (OGC CSW) provides an underlying framework that is information model agnostic, and as such can be implemented to support multiple information models (ISO, ebRIM, and so forth).

9.4 VINE API

As a cloud platform, all interaction between VINE and external parties will be mediated through its web services API. The API must therefore handle a broad variety of tasks, ranging from authentication through to data transport and on to administrative tasks such as service deployment. It must also be appropriate for a

many types of applications from small mobile apps and websites through to agency IMS's and large batch data processing jobs. Furthermore, it must be possible to extend the API as new services are added to VINE and new uses are found for existing services.

An inflexible, unreliable, or difficult-to-use API will hinder the uptake of VINE, no matter how powerful and useful the services being offered by the system. As such, it is essential that the API be well designed and well governed so that it can play its proper role as an effective intermediary between VINE and the rest of the world.

The VINE API consists of two parts: the core API and the services API. Although to an outside user there should be no apparent distinction between the two, there is a difference in the level of access the two APIs have to VINE. The core API exposes services found at the infrastructure and core interoperability layers of VINE, and as such must have direct access to these layers. The services API can interact with processes operating at the services layer but has no direct access to the lower layers of VINE.

9.4.1 API Hosting Model

VINE's API will be hosted on a set of application servers. Defining a component of the API will be a matter of using a framework to register handlers for one or more API calls, and defining the set of actions to be taken and data to be returned to the caller for each of these calls. As much as is possible, the application server framework should handle the operational issues surrounding making calls to the API component, leaving to the component developer only to specify the appropriate operational and business logic. Some of the matters that should be handled by the API servers include:

- Load management and load balancing
- Multiple serialisation formats, for example XML and JSON (see Section 9.4.2)
- Connection management
- Failure handling, retries etc.
- Security (e.g. input sanitisation)

Specifying the framework to use for developing VINE's API is beyond the scope of this document. Commonly used frameworks used for the development of web services such as JAX-WS (Oracle, Inc.) and JAX-RS (Oracle, Inc.) in Java for SOAP (web services) APIs and RESTful APIs respectively offer useful guides for the framework's feature set, as well as potentially being usable directly as foundations for the framework's development.

9.4.2 Data Serialisation

It is important that the VINE API be flexible with respect to data serialisation. In many cases, XML is used as the canonical representation for data standards. This

is due both to the widespread support for this technology as well as to the powerful facilities for defining and enforcing schema compliance. However, in many cases there is good cause to represent structured data using different wire formats for reasons of compatibility or efficiency. A common example of this is the use of JSON rather than XML when writing a JavaScript web application; this is preferred because the native support for JSON within JavaScript makes data manipulation easier. At other times, such as for large files, efficiency considerations with respect to file size or deserialisation cost mandate other wire formats such Binary XML (EXI Working Group, 2011) or Protocol Buffers (Google, Inc., 2012).

9.4.3 API Component Model

VINE's API must appear coherent to outside users, but internally it should be implemented as a set of components that can be independently managed. The component model reflects the fact that VINE's API will be very broad. In the case of the services API, it will act as an interface to a variety of components built by different stakeholders. For this case in particular it is essential that the API consist of individual components so that the service owner can develop the component of the API that corresponds to their service. To counterbalance the risks of decentralisation a strong governance process will be necessary; see Section 9.4.4 for more detail.

As mentioned above, VINE's API will consist of core API components and services API components. These two types of components differ in their level of privilege within VINE. They are described in the following subsections.

9.4.3.1 Core API Components

The core API consists of the set of API components that mediate interaction with VINE's cloud infrastructure and core interoperability platform. To do this, core API components will require direct access to these layers of VINE. As the core interoperability platform uses an ESB paradigm for inter-process communication and service invocation, it is the role of the API model to act as a bridge between service requestors and the ESB. It is anticipated that the core API components will make heavy use of the ESB's service mapping capabilities. A straightforward example of interaction with a core API component is illustrated in Figure 17, where a synchronous API request for a single service is serviced by the ESB. In the figure, a request from the user is made against the Core API's role management interface (1). This request is formulated as a message that is placed on the ESB while the API handler blocks, waiting for a response (2). The message on the ESB is handled by the role management service (3), which places a return message on the ESB (4). The API handler receives the return message (5), reformulates it into an API response and returns it to the user (6).

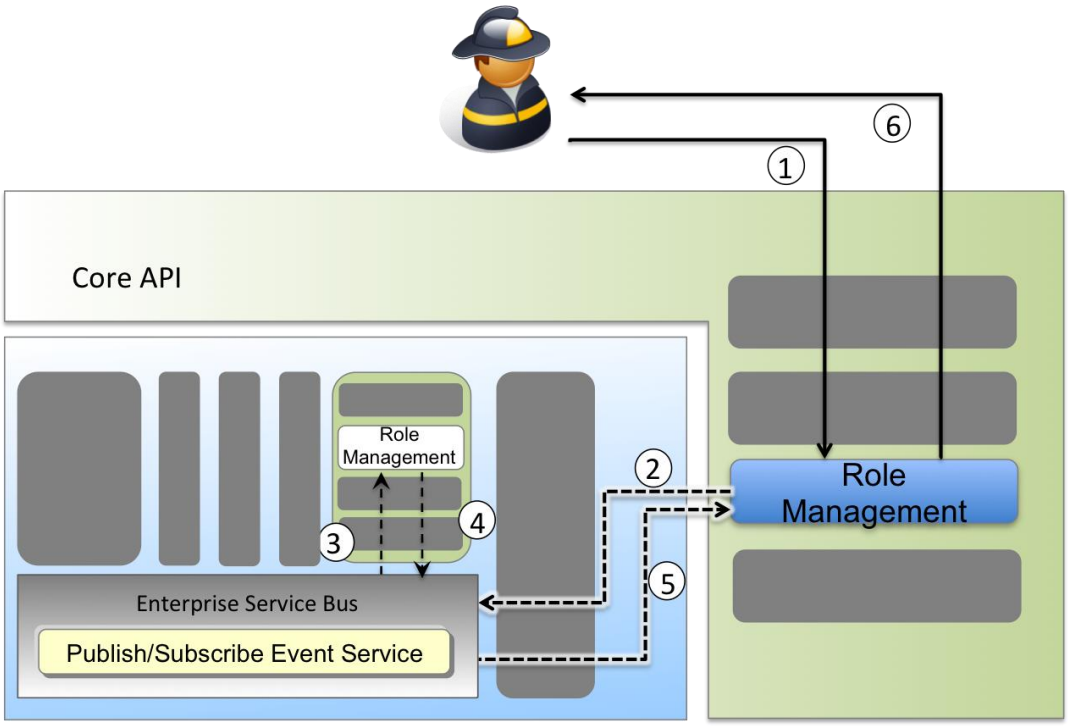

Figure 17 Sample interaction with VINE using the core API.

More complex scenarios are possible; for example, asynchronous requests or requests that require orchestration between multiple services. Different scenarios can be handled by the operational logic of each API component, with assistance from ESB features where available.

9.4.3.2 Services API Components

The services API consists of components that mediate interaction with the various services hosted on VINE's services platform. As new services are added to VINE over time, the new functionality will need to be exposed in the API in order to allow all stakeholders to access it. Service API components are intended for this purpose.

Services hosted in VINE will each expose an API for their functionality. However, as each service will be developed independently (using different development platforms in many cases) both the type and the functionality of the API may vary. For the purposes of reliability, functionality and consistency, these APIs will not be directly exposed outside of VINE. Instead, a component in VINE's API hosting framework will handle external requests and forward them to the service's internal API. In doing so, the API framework can perform services such as connection management, serialisation format conversion etc. described in Section 9.4.1. Where necessary, the services API component will also be able to handle translations between VINE's web services interface and the service's interface which may use different remote procedure call (RPC) infrastructure. An example of interaction with a services API component is illustrated in Figure 18. In the figure, a user makes a web services request for a map visualisation (1). The API component passes the request to the service, converting it to match the internal service's API

(2). The request times out, so the service automatically retries the request (3). The visualisations service returns a result to the API (4), which converts it to a JSON serialisation per the user's original request and returns it to the user (5).

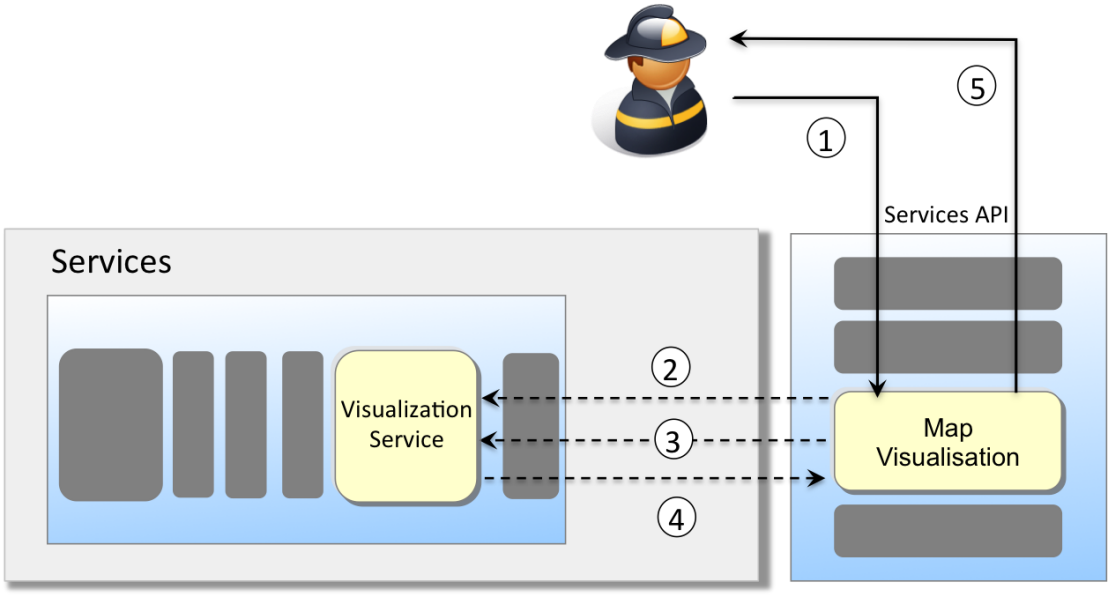

Figure 18 An example of a services API component for map visualisation.

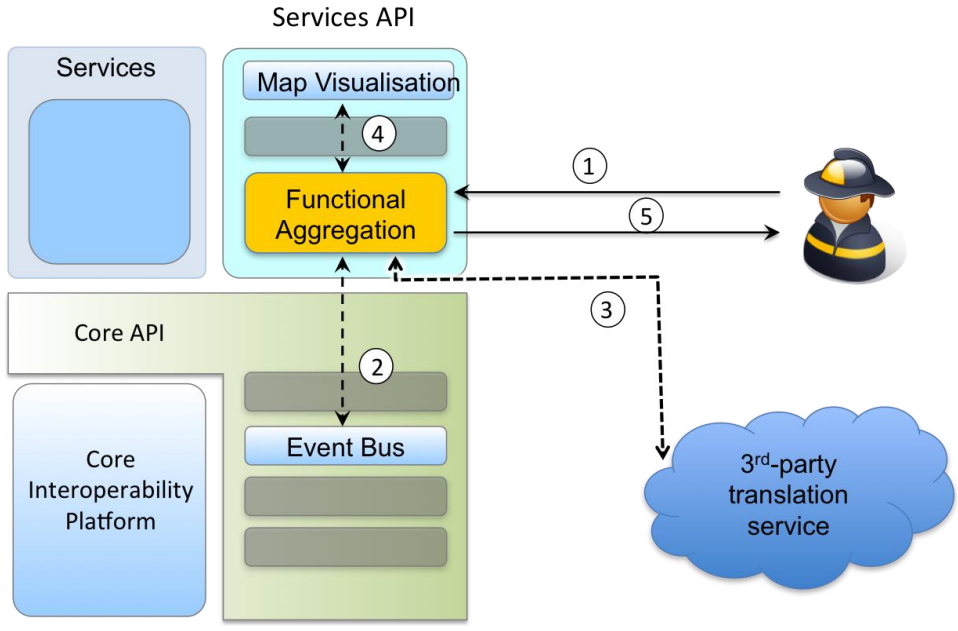

Figure 19 An example of a services API component for functional aggregation.

Another type of services API component does not have any single corresponding service but rather performs *functional aggregation* between a number of services both within and outside VINE. The purpose of such a component is to package existing API functionality in a more convenient manner for a particular application or specialised use, or to provide precomposed mashups of VINE with other online

resources. An example of the former would be a functional aggregation plugin that provided a way for a user to subscribe to all events for a given region across many topics. An example interaction with the latter is presented in Figure 19, where the API takes a request from a user (1), gathers all current warnings (2), translates them using a third-party service to Vietnamese (3), and places the translated warnings on a map visualisation (4) before returning to the user (5).

9.4.4 API Governance

VINE's API must meet the dual goals of being flexible and extensible for the variety of developers using the system, while at the same time providing a reliable, resilient and coherent interface for outside applications and users to interact with VINE. As these goals are somewhat at odds with each other, a process of technical governance will need to be put in place to ensure that the API meets the needs of all stakeholders. Governance will need to manage the following concerns:

- **Reliability.** A trade-off must be made in terms of the degree of effort required to add or modify API components at various levels as against the east with which such changes can be made. Core API components constitute part of VINE's core infrastructure, and as such must be subject to rigorous quality control. Some service API components may provide access to auxiliary services that are less critical; in such cases flexibility may be valued more highly. The governance process must apply appropriate procedures to changes in various parts of the API, and must also set user expectations appropriately.

- **Correctness and coherence.** It must be ensured that new API components as well as modifications to existing components serve a useful purpose and do not unnecessarily bloat or complicate the VINE API, and also that they are stable and correct with respect to their intended functionality.

- **Consistency.** VINE's API must "feel" like a single API, with common semantics, terminology, and idioms throughout. The governance process must provide suitable guidance as to the style and semantics of the API, and must also enforce this guidance as the API is deployed, extended and modified.

- **Dependency management.** The resilience of the API can become compromised if the level of interdependency between different components becomes too high. Although there are legitimate reasons for dependencies to be formed, the governance process must work to ensure that dependencies are well managed and are created for good reasons.

- **API Versioning.** External applications will make use of the specification of VINE's API as it is at the time they are developed. Changes to the API have the potential to break the expected contract between existing applications and VINE. As such, management of API versions will be necessary,

including maintaining access to older versions of the API for specified periods in order to provide time for existing applications to transition.

10 Development Roadmap

In this section, a development roadmap is proposed based on the requirements and architectural framework.

10.1 Single point log in for existing systems

Making early progress on the issues of identity and authentication is critical to the success of VINE. The current situation of multiple accounts and logins hampers emergency response capability and decreases the security of systems, as maintaining good security practices across so many accounts proves difficult in practice. In order to progress towards the goal of a single identity and common authentication, a two-pronged approach is suggested.

One solution is to implement client-side password-management ("keychain") software tool as is commonly available on web browsers for management of user authentication across many websites. This will effectively be a piece of software that stores a user's credentials in encrypted form on their end-device, and will act as a broker to other pieces of software, providing credentials to the software on the user's behalf. Such a keychain tool will have two immediate benefits: firstly, to reduce the user's need to remember and type usernames and passwords for all the software they need to use while undertaking their duties; and secondly, by relieving the user's burden of remembering many passwords, it allows stricter password policies to be applied by the many underlying systems. To access and update this keychain tool, the user may choose to use a master password. However, this solution has some limitations. Firstly, the password-management tool may not support multiple software or hardware platforms. Different compilations of the same application need to be released and maintained in order to support heterogeneous end-user devices and hardware configurations. Moreover, if the end-user is using a multitude of devices to access the existing software systems, it is a non-trivial task to keep the password consistent across all of them.

In terms of overcoming the limitations of the first solution, an alternative answer is to implement a single-point authentication service, which may be hosted by a recognised third party that provides managed authentication to the end-users to access the existing legacy systems. This service will act as a middleman between the end-users and the agencies. Instead of connecting to the agencies directly, the end-user initiates a connection request by passing a master username and password to the hosted single-point authentication service that will in turn connect to the agencies requested by the user if the user's identity exists in their system. This allows end-users to log-in once to access multiple systems without having to log into each of them individually, for instance 46 times to use 46 different login systems currently in operation. Such a service has three obvious advantages. Firstly, it provides a centralised and portable service to end-users no matter where they are or which device they use. Secondly, it reduces the identity proliferation that causes user confusion and leads to the use of weak or poorly secured usernames and passwords. Thirdly, it facilitates the end-users' identity management by providing a single place to implement and manage authentication and authorisation policies across multiple systems, which simplifies the process of updating, migrating, and synchronising passwords.

In the longer term, the existence of a password-management system can act as a bridge to a situation where identity is truly federated. As software migrates to a single-identity model, the password manager can continue to handle authentication on behalf of users. An ideal password management system will be extensible enough to integrate with applications, seamlessly with existing infrastructure such as authentication system, and so on.

10.2 Planned Burn Scheduler

Burning off on public or private land may result in an uncontrollable fire when it is ignited in a high-risk area or when it is affected by the change of windy conditions or periods of weather. Therefore, ascertaining suitable times for ignition that relates to weather, location, and other planned burns is important to reduce the risk of the fire becoming uncontrollable and fast moving. Currently, when DSE selects the most suitable time to light a public planned burn, a number of factors such as type of fuel, the fuel moisture, landscape, weather, and risks to population and surroundings have to be taken into account. Because the burn is dependent on many conditions that may change unpredictably, such as weather, it is usually not easy to calculate an optimal time to light the burn in a short period of time, which may lead to postponement of the burn or making a decision that is not well planned for. For a private planned burn, a permit from the local municipal office and a prior notification to ESTA are required for burning any rubbish or vegetation. However, the person who gets the permit of private burning may not know clearly when to burn and how the weather affects his or her decision.

The planned burn scheduler will be a smart tool for suggesting good times to conduct specific burns safely through combining planned burn data with contributing models, such as weather, traffic, hydrological and similar models. The capability of this tool is to provide good estimation of appropriate times for ignition.

It not only could be used by DSE to determine best times for ignition quickly, but also helping the community who are planning private burns by allowing them to query suitable burn dates and alerting them when the conditions are optimal for ignition. This smart tool has the following characteristics:

- Providing a convenient and efficient way for the fire emergency services or the community to get suggestions of best times for ignition. It facilitates the conduction of a safe burn through estimating the optimal lighting time by allowing users to enter a burn and related information.
- Providing a near real-time visualisation of the factors that affect the way the fire behaves and the impact on nearby communities and the local environment through the analysis of planned burn data combining with forecasting and simulation models.
- Alerting any potential risk based on the weather and related conditions, such as the type of fuel, landscape and so forth, at the time of a planned burn. For example, a planned burn needs calm and mild weather. It cannot go ahead if there is a sudden, unexpected change to hot, windy or stormy conditions. Under this situation, the planned burn scheduler will work out guidance for the next best ignition time according to the current weather forecast and other related environmental conditions.

10.3 An Emergency Information Portal – http://www.emergency.vic.gov.au

Victorians need a single, integrated website that provides emergency information to the public. Currently, numerous websites provide emergency information to Victorians -- http://firecommissioner.vic.gov.au

A Community Emergency Information Website needs to be a single source of emergency information that provides near real-time information about a range of emergencies that agencies, including SES, CFA, MFB and DSE are managing. This website would have the following properties:

- A trustworthy source of information governed and managed by the government agencies
- Publicly accessible
- Near real-time information updates
- A collection of all major emergencies, such as fire, flood, weather, traffic, etc.

This website will be an aggregation of all the information in a single location. Building such a site will provide the community a single point of access to all emergency information.

In order to provide seamless access to the underlying emergency data published by each ESO in VINE, a set of open data API is provided that forms part of the services API of VINE. By using this API, the emergency.vic.gov.au website and other stakeholders will be able to access the ESO data in any standard form for further use.

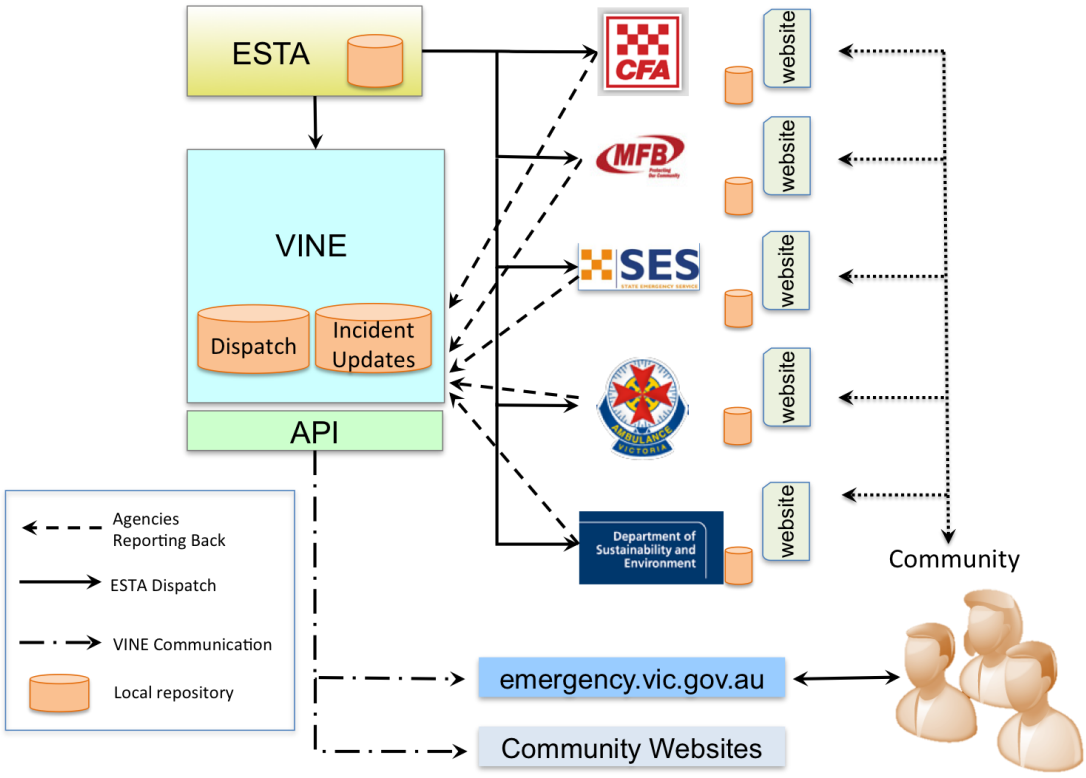

Figure 20 Data flow schematic for the emergency information API and website envisioned as an early-phase deployment for VINE.

Examples of the types of data that may be hosted using this website includes weather warnings, fire danger information, evacuation advisories, traffic hazards and road closures, planned burn and burn-off data, air quality information, flash flooding information and so on.

The website is a very good candidate for being implemented as an early win for VINE. It requires the cooperation of agencies and initiates important processes such as the standardisation and governance of data, while at the same time engaging the community by providing near real-time emergency information for public good.

The infrastructure required to create the website and expose its features through an open data API is relatively minimal. As a stepping-stone to this outcome the website could initially display planned burns data. The data collected from various agencies, such as MFB, CFA, DSE, SES and ESTA is stored in VINE and then presented on the *emergency.vic.gov.au* website. As the data is being made public, there will be no need for implementing the authentication and authorisation mechanisms. The ESOs simply publish information on planned burns to VINE

using its pub-sub API set. The published information is then stored using the standard Common Alerting Protocol (CAP) formatting. The *emergency.vic.gov.au* website would then subscribe to this data and present them on website in a comprehensible manner in near real-time. Using the same pub-sub interface, community websites could also subscribe to the emergency information.

As the website is envisioned to present all emergency information, in addition to planned burns, more data, such as floods and weather related emergencies, could be integrated following the initial planned burns project.

10.4 Visibility of Resources During an Emergency

For effective and efficient management of resources during emergencies, there should be a system in place that keeps track of all the resources including appliances and personnel available at any time and those dispatched to attend the emergencies.

The current system used by ESTA for resources tracking and decision-making during emergencies, namely the CAD, has several shortcomings. They are listed as follows:

1. Use of proprietary protocols and closed communication systems.
2. Lack of persistent mechanisms for historic data storage and record keeping on resources dispatch and location.
3. Lack of physical and logical links to connect ESOs to ESTA's resource tracking and management system.
4. Absence of personnel tracking and reporting capability during emergencies.
5. Lack of the capability to integrate geolocation information obtained from smart devices, such as smart phones and tablets, with the current CAD system.

The resource tracking system that ESTA currently has in place is proprietary and uses its own set of tools and technologies. For example, the GPS enabled Mobile Data Terminals (MDT) mounted in ESO vehicles send location packets over the Mobile Data Network (MDN) to CAD system located in ESTA. Rural Ambulance Victoria (RAV) vehicles send location and status data from GPS enabled trunked radio terminals through their proprietary network to CAD as well. Except for vehicles and devices, ESTA CAD system does not have a personnel tracking capability. The use of proprietary protocols and a closed system poses a serious limitation on extensibility and availability of the data to the ESOs.

At present ESTA's CAD system makes use of only the most recent location tracking data in for its "resource recommend" function and to manage resources at an incident. No history of location is maintained in the CAD database, and the CAD workstation ignores all but the most recent Automatic Vehicle Location (AVL) packet from the network. Due to the lack of historical data on vehicle tracking and

management, a substantial capability to look back at past resources management decisions and learn from them to optimise resource allocation in the future is entirely absent. Moreover, ESTA does not have any physical infrastructure or any logical links at their end that could connect ESOs to their CAD system in possession of AVL data. Finally, if the CAD system goes down for any reason and ESTA must go into 'manual' mode, none of the location data is available to the manual operators.

Currently, only closed networks and proprietary protocols are used to transmit data without leveraging other available networks, including public network and/or popular smart devices. As smart devices are becoming more and more pervasive, and public communication continue to increase in their coverage area, their use by personnel at the field in emergency situations is becoming more common alongside the use of agency provided devices. As the CAD system operational at ESTA is proprietary and closed in terms of extensibility, ESTA is unable to add any functionality to it unless it procures additional software tools from the same vendor. In summary, ESTA lacks the *effective* capability to use smart phones and/or public network channels for personnel and vehicles tracking.

The current system is illustrated in the following figure:

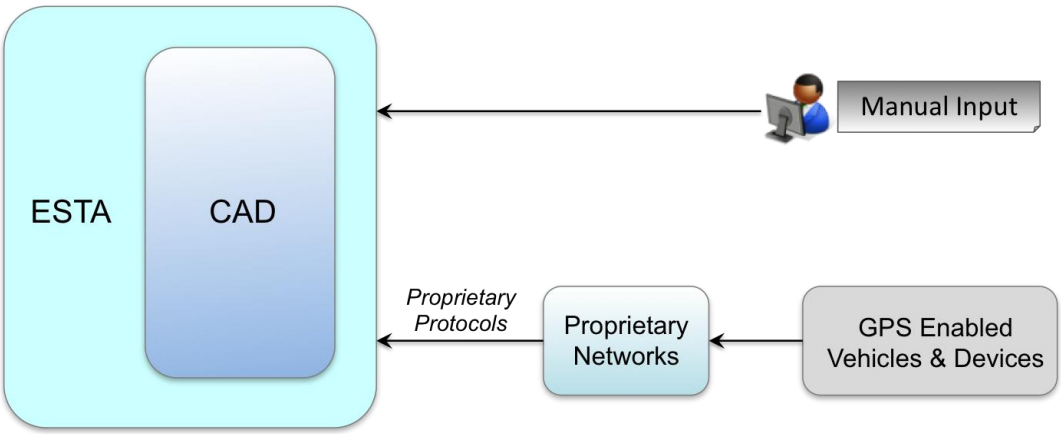

Figure 21 Existing mechanism for location reporting.

A project to overcome the limitations of the current solution for resources tracking could constitute an early win for VINE. Such a solution could make location-tracking information available to all stakeholders including the community, take advantage of GPS-equipped smart devices to gather additional location information from vehicles and personnel, and store an historical record of resource location for later analysis.

Without any disruption to the current tracking system operational at ESTA, we propose two ways of achieving resources and personnel tracking that could be shared across ESOs:

- Create an API to obtain the location data from CAD at near real-time. This is a mirroring service that simply replicates the data coming to CAD into

VINE. But VINE will record all the changes in the database for future analysis.

- Create an API to obtain the location data directly from the sources without the bridge to CAD. That is to say, VINE needs to consume different format data coming from different sources.

The first approach is preferred as it does not require interfacing with a variety of proprietary networks and protocols, reduces the overall project complexity and allows earlier completion.

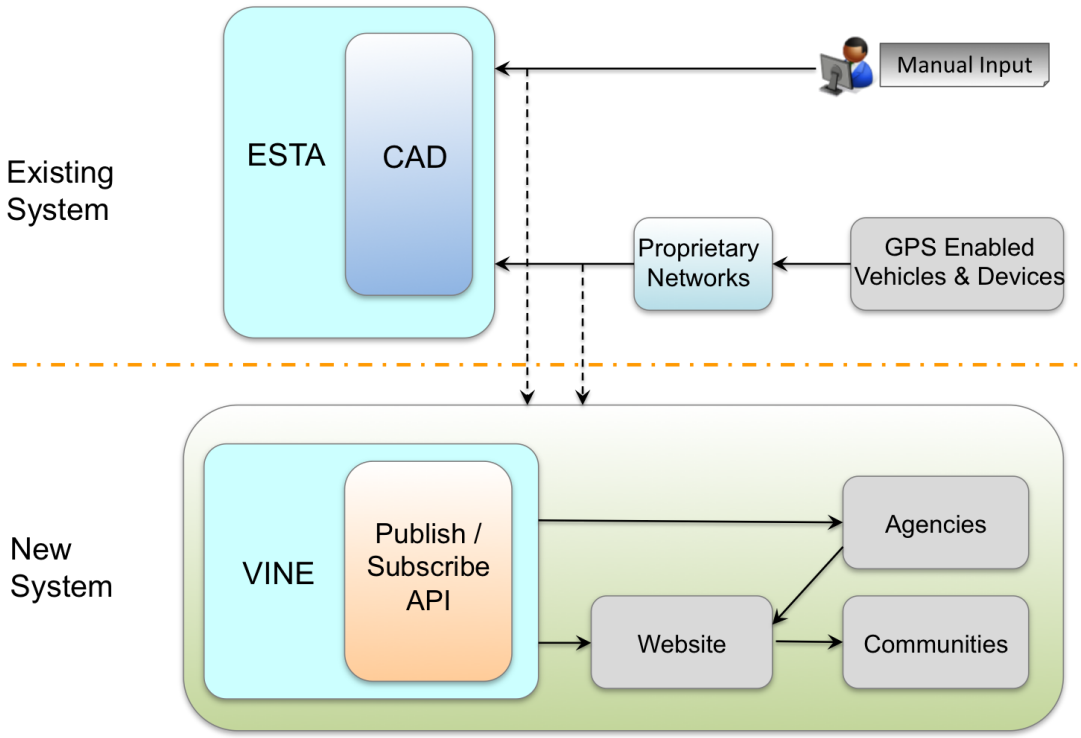

Figure 22 Architecture of new location tracking and sharing system.

A second stage to this project will build the capability to record and track appliances and personnel through smart devices using public commercial wireless networks. The proposed scheme is:

- Build an interface to obtain geo-location data from smart devices, such as mobile phones and tablet devices that are mounted on dispatched vehicles and/or being carried by personnel in the field.

- Different Agencies could subscribe the location information from VINE directly through the publish/subscribe event service.

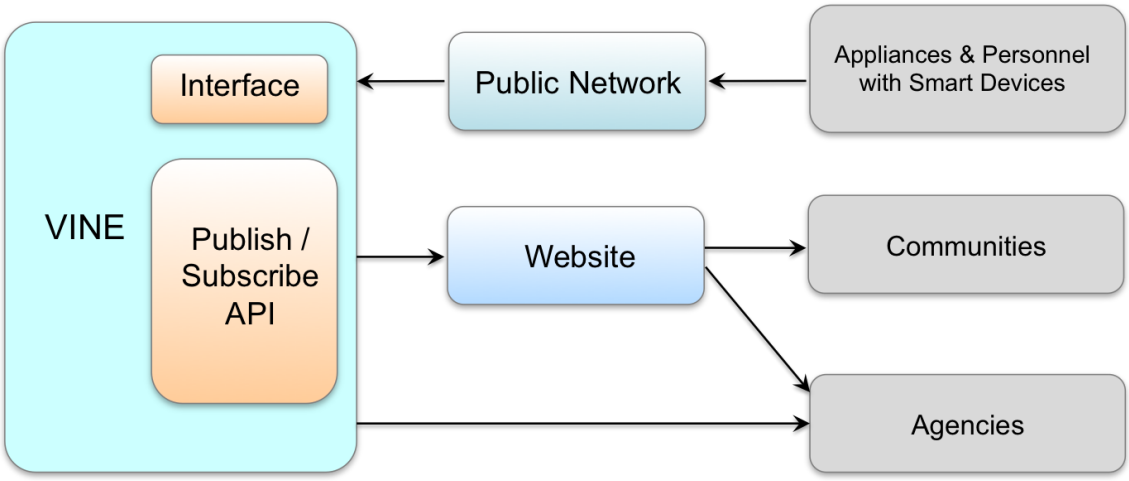

Figure 23 Architecture of location tracking and sharing system based on public commercial wireless networks.

The following high-level tasks will need to be completed to achieve the functionality described above:

- Identification of suitable standards and common information models for representing resource tracking and auxiliary data for this application, using a consultative process across stakeholders.
- Build underlying VINE infrastructure for hosting the publish/subscribe event service, parts of the data store, and the web services API.
- Build a component of VINE's authentication and authorisation services capable of supporting the necessary sharing of data across ESOs
- Provide an API to which resource location information can be submitted.
- Provide a resource tracking API by which stakeholders can access current and historical resource location data.
- Modify ESTA's CAD to publish the location information to VINE's API.
- Build a mobile application to submit location from smart devices to VINE's API.

11 Discussion

11.1 Potential Risks

Any large and complex ICT project carries significant risk of delays, cost increases, partial completion, and poor outcomes (failure to meet objectives). The Victorian Ombudsman's investigation into public sector ICT projects in Victoria (Victorian Ombudsman, 2012) discusses many of these risks, their causes (technical and organisational), as well as measures to mitigate them. There are many similarities between a number of the projects in the report and VINE, and in general the report serves as a good cautionary guide that should be consulted when developing an organisational and technical framework for the implementation of VINE.

In this section, a number of issues have been identified as potentially carrying significant risk for VINE. These issues are discussed, and mitigation strategies proposed.

11.1.1 Integration Costs

Integration with existing systems and data sources forms a major component of large-scale projects such as VINE—especially when the project is intended to bring together systems across multiple organisations. Furthermore, such costs are notoriously difficult to estimate accurately and are often quite severely underestimated. Such underestimates often result in cost blowouts and time delays and, in many cases, project failure as is illustrated in a number of cases by the Victorian Ombudsman's report.

11.1.1.1 Mitigation Strategy

Integration will inevitably form a major part of the cost of deploying VINE. However, the proposed development model whereby VINE is implemented as a sequence of distinct projects with clearly-defined scope limits the risk of unanticipated blowouts

in integration costs. Because VINE is intended to be a platform rather than a monolithic system, cost-benefit analyses undertaken prior to each potential deployment can manage risk and ensure that the benefits outweigh the integration cost in each individual case.

As part of the process of undertaking each new phase of VINE, a thorough analysis of integration requirements associated costs must be conducted in order to estimate the risk, cost, and timeframe associated with that particular project.

11.1.2 Concentration of System Failure Risk

It is intended that VINE be implemented to high standards of reliability and availability. However, by concentrating a variety of information and services within a single system, there is inevitably a concentration of risk. If not properly mitigated, a failure of VINE's services during a critical period could have a significant impact on emergency management capability for Victoria.

11.1.2.1 Mitigation Strategy

The principles of resilience were discussed in some detail in the BLUEPRINT. Resilience in a system is about more than just robustness, but is also about tolerance to failure: accepting that failure is sometimes inevitable and designing systems and processes to be able to keep functioning in the face of such failures. Such resilience operates at many levels. VINE itself will be built for high availability by using methodologies such as redundancy, QoS, and service isolation.

At the next level, major stakeholders (such as ESOs) must develop contingency arrangements—both technical and procedural—such that they can continue to operate effectively should VINE fail. Technical contingencies include local replication and caching of data at various points, as well as the maintenance of alternative methods for sharing information. At the organisational level, organisation must continue to invest in preparedness for using manual procedures in a major emergency. For example, even if VINE were to provide automated support for AIIMS procedures, training and preparedness to revert to the original manual process should be maintained. Effective training is a crucial part to mitigating risks of system isolation or failure: large-scale training exercises should simulate not just external emergencies, but also disruption to internal systems in order to ensure a resilient emergency management capability.

11.1.3 Data Standardisation and Governance Process

The potential benefits of VINE cannot be realised without a cooperative and consultative process for the establishment of data standards and common information models, and an effective process for data governance to ensure that data is adhering to the commonly agreed formats. The risk is that building the infrastructure of VINE in accordance with the proposed architecture without investing heavily in the process of data governance will result in a 'Tower of Babel' situation, whereby all the information necessary for a common operating picture is available through VINE, but due to incompatible data formats each stakeholder is

only able to interpret the data that they themselves contributed. Information interoperability is only possible with a common understanding of information.

11.1.3.1 Mitigation Strategy

A strong governance process with clear roles and accountability will be established at the earliest stages of VINE. Regular auditing of compliance with data standards will be a part of this governance process. To ensure that the governance of standards remains effective, there will also be an ombudsman who will retain oversight over governance and investigate concerns brought forward by VINE stakeholders.

At the inception of each new phase of VINE, definition of data standards and common information models to be used for that phase will be a mandatory part of the planning process.

11.1.4 Exclusion of Established Solutions

The architecture for VINE as presented in this document is intended to leverage industry standard components for building out the core interoperability platform. However, as discussed in Section 9.1, it is intended that the core interoperability platform be kept as compact as possible. As such, a number of quite important services within this architecture are envisaged as being implemented within the modular services platform, and to interact with other components using the core API. The intention of this design is to minimise vendor dependence within VINE and establish a long-term asset for the state of Victoria.

A risk of this approach is that the architecture may exclude established solutions that are unable to fit the proposed model. For example, some vendors may offer solutions that in many ways meet the needs of VINE quite well but because they are integrated solutions they cannot be run at the services layer. In general, imposing the requirement that systems interact via VINE's API implies that some level of integration work will be required for off-the-shelf software to be deployed at the services layer.

11.1.4.1 Mitigation Strategy

VINE's core interoperability platform makes use of an ESB, which will provide a variety of tools designed to assist with integration of various packages. This should allow for the selection of appropriate solutions for each individual package, as integration can take place at the ESB level. The ESB will not be available at the services layer, but a range of data and service integration tools should be used to enable integration of distinct software packages; off-the-shelf software should not be excluded from consideration for any role within VINE, though the cost of integrating that software with the rest of the system should be taken into account as part of the selection process.

Furthermore, it is important to note that while this document sets out a high level vision for the configuration and deployment of VINE, it is not intended to be a procurement document and as such, has not made any recommendations about

particular vendor solutions that may or may not be suitable for various components of VINE. As the agenda for VINE moves forward and more detailed procurement processes begin, it is important to be pragmatic and secure the best solution for meeting VINE's long term objectives for the state of Victoria. This also includes remaining true to the standards and modular principles expressed in this document as it is these principles that will protect the 'State Asset' and its longevity. In some cases, expediency may result in selecting solutions that deviate from the architecture laid out in this document. So long as the overarching framework is held true and the trade-offs are understood that can still be the correct decision.

Because of the high level nature of this document, it should not be used on its own to rule out any possible solutions. Instead, it should be used to provide a framework and set of guidelines by which proposed solutions may be judged. Furthermore, it is intended to be a living document; as the process of building VINE progresses, lessons learned from procurement should be used to adjust the high level architecture in accordance with the thinking at the time.

11.2 System Administration and Governance

VINE's success will only be possible if it is perceived by the many stakeholders that make up the platforms constituency as being reliable, trustworthy, useful and responsive to their needs. A variety of initial and ongoing system administration and governance activities will be required to ensure that VINE's performance earns the trust and goodwill of its constituency, and that it continues to meet their needs. Thus, VINE must be supported by an organisation, accountable to all stakeholders, that is dedicated to its administration and governance.

The following administration activities will be necessary both at the development stage of VINE and for its ongoing maintenance:

- Standard systems administration and maintenance tasks for ensuring the ongoing availability, reliability, scalability and security of VINE.
- Setting of appropriate Service Level Agreement (SLA) targets and putting measures in place to ensure that these targets are met.
- Secure data backup; planning for rapid disaster recovery in the event of system outages or data loss.
- Database administration tasks such as the creation of new datasets, setting of access rights and ensuring that database performance continues to be satisfactory.
- User management tasks such as the creation of new roles in the system and setting of appropriate access rights.
- Quality assurance for the deployment of new services.
- Release management: developing and enforcing procedures for upgrading VINE's infrastructure and platform.

The following governance activities are necessary for the successful establishment and continued usefulness of VINE:

- Selection of appropriate data standards for VINE through consultation with relevant stakeholders.
- Participation in committees for the development of new standards.
- Ongoing engagement and outreach activities with the full range of VINE stakeholders—especially the community—to ensure that it continues to meet and adapt to stakeholder needs over the life of the system.
- Effective, accountable project oversight and roadmap planning for the development and expansion of VINE.
- Maintenance of a lexicon for technical and emergency management nomenclature in the context of VINE.
- Enforcement of compliance with mandated standards for data representation.
- Establishment and enforcement of consistent style and semantics for VINE API components.

12 VINE Scenarios

The following three scenarios were developed as part of the VINE architecture process in order to paint a picture of how VINE will one day function in an operational context, and to validate the proposed architecture against the full vision of information interoperability and decision support as envisaged in the Blueprint. The scenarios were presented and discussed at a workshop in August 2012 at which stakeholders from all Victorian ESOs as well as a number of other stakeholder agencies and departments were represented. The discussions that took place at that workshop were a source of valuable insight that has influenced the content of this document.

The scenarios are reproduced here in exactly the same form as they were presented at the workshop.

12.1 Scenario 1: Bushfire and Evacuation in the Dandenongs

12.1.1 Synopsis

Due to sudden weather condition changes, a burn on private land turns into an uncontrolled fire.

The community is able to stay informed of planned and private burn activity in their area through a variety of channels. The combined planned burns website provides real-time information about the planned and on-going burns. Individuals can subscribe to customised notifications about planned burns in a given area (e.g. their suburb or local Council). Through the use of applications on smart devices, the information is made available “anytime anywhere”, being tailored to the actual location or area of interest for the user.

Local media (e.g. radio) communicate the planned burns that are taking place. The broadcasted information is the same as that delivered through smart devices but in this case it is communicated through a different channel. Text to voice automated conversion software helps in delivering the information over the radio or phone.

12.1.2 Detailed Scenario

1. The planned burn begins.

- The landholder updates the status of the planned burn as it is happening based on the information received from the field. This information is immediately available through the planned burns web sites as well as portable apps.
- Location, status and details of the burn are visible to other agencies.
- The number of calls from the community reporting the smoke and burn to ESTA is reduced. ESTA is able to respond quickly and effectively to calls reporting the planned burns.

2. A weather forecast update shows a sudden change in the wind direction and a significant increase in wind speed, leading to dangerous fire conditions.

- Before being registered as a current forecast for the area of interest, a workflow begins where the forecast is allocated automatically for validation to an expert fire prediction specialist whose details have been previously entered into VINE. The specialist is interstate but receives the specific information on his smart device and verifies the forecast.
- As a result of this workflow process, emergency personnel are notified or alerted about the updated forecast.
- The forecast is flagged as requiring a change in the fire danger rating. Together with this information, current burns underway are brought to the attention of the emergency personnel. This is achieved by correlating the available information of the burns reported in VINE to the new event represented by the forecast update.

3. CFA cross-references this information with the combined planned burns database and concludes that a planned private burn currently taking place in the Dandenongs is a risk factor. Local communities and groups in the surrounding areas of the location of interest are informed of the status of the fire risk.

- The new situation is evaluated and the potential risk is confirmed for the planned burn occurring in the Dandenongs.
- The landholder is advised of the change to the weather and advised by CFA to extinguish the fire immediately. At the same time, local community members who have subscribed to VINE are informed about the updated

condition. Because the state change has been published in VINE, media can also inform the community through their standard channels.

4. Before the change arrives, CFA and DSE have already drawn up contingency plans in case the fire escapes its containment boundary. Some resources are deployed to suitable locations to fight the fire and put in a state of alert. Community and area health services are alerted of an increase in the fire spread risk and asked to 'be ready to leave'.

- The information about the current state of crews, fire truck locations and other resources such as private water dams, tanks and catchments are used to define a strategic plan.
- The strategic plan is then composed by combining the above information with other relevant data-sources (databases of hazardous material, road conditions, location of power lines, schools etc). VINE provides access to these datasets and relevant information, contextualised to the current condition (based on geo-location and possibly more complex relations), is extracted from them.
- CFA and DSE have a complete view of the situation and decide to allocate additional units to prevent major disasters. The evolution of the current conditions is tracked in VINE and this information is propagated appropriately.
- Given the potentially critical conditions, an aircraft is deployed to monitor the state of the fire.

5. The change arrives and, as feared, the landholder updates the status of his fire as out of control. The alerted teams respond to the fire, but the fire continues to spread. Extra resources are provisioned across different agencies to support the existing units. The fire spreads significantly. The Metropolitan Fire Brigade is requested to send adjoining units.

- Officers in the field inform the incident control centre about the evolution of the fire via mobile devices.
- The community reports that there is a fire and starts to tweet and blog about it.
- The information from the social networks, which has been "sensed" by VINE, is automatically cross-referenced with the updates coming from the people in the field as well as with reports and photos showing the fire boundaries coming from the surveillance aircraft flying over the area.
- The aggregate information brings evidence of potentially new fires spawning from the original one and the surveillance aircraft is directed to where the social media information suggested new fire spots.

- As the fire is spreading and new fire spots are identified there is the need of additional resources. The Metropolitan Fire Brigade is asked to provide support in fighting the spot fires and provide asset protection.

- Workflow procedures implemented in VINE facilitate the request of assistance via ESTA from the MFB.

- By tapping into the same information made available through VINE and used by CFA and DSE, the MFB Incident Control System is able to immediately provide the MFB personnel (staff and operatives) with an understanding of the situation happening in the Dandenongs. This information is constantly updated as the fire trucks move to the identified area.

- Once MFB is on the scene, the integration of AIIMS in VINE allows the operatives to quickly understand the command structure and have a complete awareness of the available resources.

6. The State Fire Controller, in conjunction with the Regional Controller, reviews the fire-spread model and they conclude that several communities are now at risk. The Incident Controller discusses this with the Evacuation Manager at Victoria Police who immediately prepares plans for evacuation, considering road conditions and available shelters and resources. Affected communities are alerted to begin preparing their vehicles for evacuation. Community groups are engaged to assist with the evacuation of vulnerable individuals in the community.

- The updated and real-time information from the field feeds the fire spread model constantly and contributes to a complete and unambiguous view of the fire evolution to the State Control Centre personnel, Regional and Incident Management Teams.

- The latest fire spread model simulation indicates that the communities are now increasingly at risk. This is cross-checked with the reports coming from the CFA, MFB and DSE as it is happening in the field.

- The intensity of the fire calls for immediate evacuation as there is little time for people to enact their fire plans. Fire Services, in consultation with Victoria Police and other key agencies issue a targeted alert to people in the affected area, with specific instructions to prepare for an evacuation. This alert is spread through the many channels leveraged by VINE. At the same time, community individuals registered as volunteers in VINE are notified about the possible need of help.

7. An evacuation model provided by the University of Melbourne shows a suggested evacuation plan taking into account real time road conditions and demographics as well as the fire-spread model. It suggests that for an evacuation to be effective, an order must be issued in the next 15 minutes. The Incident Controller evaluates the suggested evacuation plan, and decides to enact it. Customised alerts are immediately issued to all

community stakeholders, as well as to emergency services personnel and auxiliary services in the area.

- In each case, the alert contains information that is relevant to the role of that particular stakeholder in the evacuation. In particular, individuals who have suitable devices (e.g. smart phones) receive customised evacuation directions relevant to their actual location. All emergency service officers in the field also receive information about the evacuation routes and are posted to intersections to direct traffic.

- In order to plan for an effective evacuation, a probability heat map generated by the fire spread model is used to determine the evacuation schedule for the affected communities.

- The evacuation model is constantly updated with real-time information about the road network conditions, demographics data about member at risks in the community, data from nearby hospitals about people who decide to disclose their health conditions for emergency purpose, and resources (shelters, hospitals, ambulances, community groups and volunteer in the fields checked in through AIIMS).

- Along with the devised evacuation plan, the model also informs the Incident Controller and agency Commanders about the urgency of executing the plan given the increasing risk and specifically advises that, in order to be effective, the plan needs to be put in action in the next 15 minutes.

- All stakeholders receive customised alerts and clear instructions on how to evacuate and where to look for help. Emergency services personnel receive tailored information from VINE. Community groups and volunteers can leverage either a direct communication channel to VINE or receive information via broadcast media.

- The evacuation starts and the State Control Centre is constantly updated through VINE. Also, the social media are continuously sensed, in order to obtain a better picture of the evacuation.

8. A burning tree blocks one of the two main roads out of the evacuation area. A large number of tweets about this event are registered and this is issued to the Incident Controller as an unverified report. The evacuation model immediately starts working on a revised plan to account for this contingency. This task, as suggested by VINE, needs to be completed in 20 minutes. While this is happening, a ground crew and helicopter dispatched to verify the road closure confirm that it is indeed closed and request resources.

- Trend and event detection models operating in VINE correlate a collection of tweets reporting a burned tree blocking a road with the evacuation process.

- A workflow process implemented in VINE is triggered and after appropriate filtering and validation this information is brought to the attention of the Incident Controller. Because the Incident Controller classifies the information as 'highly relevant', it is propagated automatically to the commanders in the field.

- As a secondary step in the workflow, a new evacuation simulation is started with the updated information while the unverified report is assessed on the field by ground crews and air observers. This task, as suggested by VINE, needs to be completed in 20 minutes.

9. The revised evacuation plan is issued. Emergency personnel in the area are informed of the new plan and immediately begin redirecting traffic from the closed road to nearby designated shelters that are registered in VINE. At the same time, suitable local resources such as tractors or other heavy vehicles that can be used to remove the road blockage are identified. A local contractor (who is accredited in VINE for carrying out this type of work) is identified close by as having a large truck-mounted crane, rated as able to move loads of the size of the fallen tree.

- As soon as the road closure is verified, the new evacuation plan is already available and therefore dispatched as an update of the current status to all the stakeholders involved.

- The local truck operator receives the request over VINE on his smart phone. He confirms receipt of the request and confirms that he is proceeding to the job to remove the tree. Progress is monitored in real time.

- The personnel responsible for the revised evacuation are able to set the process in place for the removal of the tree and the progress of the truck-mounted crane to where the road blockage is.

10. The road blockage is removed, and residents taking shelter are given the all-clear to use the reopened road to complete the evacuation.

- The Incident Control Centre updates the situation in VINE and the evacuation model is run again with the latest conditions (i.e. traffic conditions, location of people etc.) in order to ensure that is now safe for people to leave the shelters.

- An update to the situation is issued to all interested parties and the original evacuation plan proceeds.

11. The evacuation is successful. The fire fight goes on for several more hours before the fire is under control. A number of properties have been destroyed but no lives have been lost.

- VINE is used to notify all the agency commanders and fire-fighters with the latest situational information augmented with data contributed by the community.

- As CFA, MFB and DSE personnel view damaged properties they enter this information into VINE along with photos taken on their smart device to register damage and commence recovery activities.
- A single and unambiguous view of the ongoing fire being put under control is made available on publicly accessible channels of VINE.

12. Using the information recorded in VINE as the fire proceeds, recovery operations are planned before the fire is even extinguished, and begin within a short time of the fire taking place.

- Insurance companies can cross reference claims on property damages on the VINE database and be confident that the claims are not fraudulent. This enables early payments to the population, where applicable.
- The Department of Human Services, Department of Health, Local Government, Red Cross and other agencies are able to act immediately to assist individuals who have been displaced by the fires.
- A complete log of the operations and the decision taken to fight the fire is stored in VINE and kept for review as well as a knowledge base for any similar future events.

12.2 Scenario 2: Urban Fire and Toxic Plume

12.2.1 Synopsis

Fire in a chemical storage facility in Fishermans Bend causes the release of a toxic gas plume that threatens to affect the crowd at an AFL finals match in Docklands Stadium.

12.2.2 Detailed Scenario

1. At 3:00pm on a Saturday afternoon in September, a fire triggered by an electrical fault breaks out at a car assembly line in Fishermans Bend. Fire crews are alerted and nearby units begin to fight the fire as per normal procedures. Information about the fire is entered into VINE via ESTA, which triggers a number of automated processes. One of these processes scans for nearby hazards that may be a risk, and this identifies a chemical storage facility immediately adjoining the scene of the fire. Senior personnel are alerted about the new threat.

- The fire is first reported by a 000 call and CCTV footage to ESTA, resulting in dispatch of fire crews to the scene. Both the location of the fire and the resources provisioned to fight it (including specific skills sets) are recorded in VINE as part of the dispatch process.
- The existence of the fire in Fishermans Bend is immediately made available through the single emergency service website as well as any third-party apps that subscribe to the relevant information in VINE.

- As a result of the dispatch, the status of the closest MFB fire stations as well as the work rosters and skill sets of the fire fighters in those stations, are immediately made available to the Incident Controller. This information helps the Controller to rapidly identify potential needs.

- VINE contains a detailed registry of hazardous sites as part of Victoria's emergency preparedness program. When the fire is verified in the system, this triggers an automatic process to cross-reference the location of the fire with the hazardous sites registry of manifests to identify potential risks in the event that the fire spreads.

- This process identifies a large hazardous materials storage facility immediately adjacent to the fire, and alerts the Incident Controller of this fact, with a suggestion that the situation is extremely serious and that senior personnel should also be made aware of it.

2. Through VINE they also discover that the fire is out of control and the existing units are unable to handle it. Using their situational awareness, the Controller and agency commanders conclude that it is too late to stop the fire from reaching the chemical storage facility. Using information stored in VINE, they learn about the types of chemicals being stored at the facility and immediately dispatch specialised fire fighting units in order to extinguish and contain the chemical fire as quickly as possible. An evacuation plan for the immediate area is formulated with the assistance of simulations in VINE, and these are discussed with relevant members of the emergency management team.

- As soon as the threat to the chemical facility becomes known, Department of Health officials, WorkSafe and EPA are alerted and commence planning. Nearby ambulance units and hospitals are put on alert for the possible consequences of impacts from the chemical spill or toxic plume.

- Reports from the field are directly entered into VINE, where they are aggregated to get a unified view of the situation. Incident Controllers and agency commanders use their VINE-connected IMS to determine that the fire is out of control. A map view supports their understanding of the situation.

- Once the Incident Controller and agency commanders make the decision to send specialised units to the scene, they query VINE to identify which stations can provide skilled personnel and equipment specialised in handling the types of chemicals being stored at the facility. These units are immediately dispatched.

- The Incident Controller and agency commanders enter a number of what-if scenarios into VINE to get a better understanding of the potential ways in which the emergency may take shape. Among these, an evacuation scenario is considered for people in the nearby park and go-karting complex.

- As the risk of starting a chemical fire becomes more imminent, the Department of Health, as well as nearby hospitals and ambulance units, are alerted and updated with the latest view of the situation.

3. As soon as Incident Controller and agency commanders become aware of the threat of a toxic chemical release, they use VINE to begin modelling the shape of any potential gas plume. The result of this analysis suggests a possible threat to the spectators at an AFL finals match at Docklands Stadium, but that due to gentle wind conditions low level risk there is no immediate need to evacuate the facility. As a precaution, stadium operators are contacted and requested to close the roof of the stadium which will reduce the volume of gases entering the stadium.

- Incident controllers initiate plume modelling from within VINE. The model takes into account the types of chemicals being stored at the facility, current weather conditions, as well as a high-resolution wind forecast that is triggered for the area immediately surrounding the fire.

- The predicted path of the gas plume is projected on to a map, where it become apparent that it poses only a minor threat to Dockland Stadium, where an AFL finals match is in progress.

- Incident controllers look up contact details for stadium operators in VINE, contact them, and request for the roof to be closed.

4. The fire reaches the chemical storage facility at about the same time as the specialised firefighting units arrive on the scene. Thanks to the advance warning and information about the chemical payload they received from VINE, the firefighters are able to contain and extinguish the fire much faster than they otherwise could have. However, they are not able to prevent a significant discharge of toxic gas from the facility. Firefighters on the scene, in consultation with plant personnel who have been contacted via a registry of critical contacts maintained in VINE, are able to estimate the volume of toxic material that has been released. This information is entered immediately into VINE .

- *Department of Health officials* contact a world-renowned expert on the health effects of toxic materials. His home is located several hours' drive from the State Control Centre, however he is assigned a new role with greater access privileges and using an application on his tablet device (e.g. iPad) he is able to monitor the situation.

- As soon as the fire is extinguished, fire fighters on the scene begin to enter data about the quantity and type of gas that has been discharged by counting the number of storage vessels that have been compromised by the fire.

- Incident Controllers immediately grant the expert additional data access privileges in VINE so that he can carry out his role during the incident. They are also able to direct him to a website where he can download specialised

software for his tablet device that gives him an appropriate view on the data, and where he can guide the plume modelling process in accordance with his expertise.

5. The presence of the plume in the CBD sky captures the attention of a lot of people out in the city on that Saturday afternoon. Alarmed by the strange-looking plume of gas, large numbers of people begin to contact the authorities, many of them by using their mobile devices to look up the emergency services website for knowing the nature of the plume and getting updates on the ongoing situation. The surge in demand of information puts the Victorian Emergency Information Line and VINE infrastructure under considerable stress, but its elastic infrastructure allows the system to provision additional computing resources where they are most needed to address the spike in demand without failing.

- Having a web service that continues to remain responsive and available in the face of this large demand allows people to remain informed through official channels and minimises the need to resort to rumour and unnecessary 000 calls. Furthermore, the relatively large number of police officers on the ground in the CBD are able to share the same view of the situation via their portable devices that are connected to VINE.

- VINE is implemented on a flexible and scalable virtualised infrastructure which allows the easy replication of services and the reallocation of resources from less essential or underutilised services to those where there is the greatest need and demand. This allows it to withstand the spike in web traffic and serve critical information to the community.

- Because the same consistent foundation of information is used to inform both the public as well as emergency services personnel such as police, the actions undertaken and the information provided by the police on the ground is consistent with the information that the public is able to access via websites and other forums such as radio.

6. At the same time, the public make large numbers of posts to social media sites expressing their concern at the situation and discussing what it might be. Social media analysis begins to take place in order to monitor community sentiment and understanding of the situation. Posts to social media sites are filtered, aggregated, and clustered to get a sense of the magnitude of public concern as well as to track the spreading of rumours and inaccurate information. By monitoring the social media discussion clusters, emergency services media officers can proactively react to rumours and misinformation with corrective posts of their own. Social media analysis is also used to look for potential panic hotspots.

- Through such monitoring, emergency services become aware that panic is spreading through the large crowd of spectators watching the AFL finals match at the Docklands Stadium. Stadium operators are contacted and requested to announce corrective information to reassure the crowd that

there is no threat. Semi-automated tools allow emergency services agents to respond to large numbers of social media postings in a short period. Victoria Police monitors the social media situation and decides to send a number of mounted police to the Docklands Stadium as a contingency against any disorder that may occur.

- Social media monitoring capabilities built into VINE retrieve information from a number of social networks (eg. Twitter, Facebook) and aggregate and collate the information in a way that is easy to analyse by incident controllers.

- A cluster of tweets containing the words "Melbourne" and "nuclear accident" is identified by this process, suggesting that a rumour is spreading that a nuclear accident has taken place. Social media officers working for the emergency services are able to directly respond to this cluster of posts and direct people to the official information, ensuring that they have a good understanding of what is actually going on.

- A geographic clustering of postings also shows that a very high volume of negative and misinformed posts is circulating amongst people located in the Docklands Stadium. This raises concern that a large number of panicked people in a confined space may cause a dangerous stampede.

- Updates about the status of the plume and its real nature are sent to the stadium operators in order to avoid the spread of panic.

7. The immediate threat of a stampede passes, but many people have lost the appetite for football and large numbers of people start leaving the stadium in a disorderly fashion. Train platforms become dangerously overcrowded as the timetable is not designed to cope with such a large number of people at this time. An alert is issued to the Department of Transport and the railway operator about the dangerous situation. A Department of Transport traffic model is able to take all the real-time location information for trains on the network and devise an optimal plan for diverting them to Southern Cross station to clear the crowd.

- Further social media monitoring confirms what the stadium operators have notified, which is that large numbers of people are leaving the stadium and crowding on to platforms at Southern Cross station.

- Incident Controllers realise that a dangerous situation will develop unless additional trains arrive at the station to take the people away. This is not straightforward as the schedule does not support this and few trains are in a position to directly divert to Southern Cross without alterations to the routes of other trains.

- An optimisation model of the Melbourne metropolitan train network derives a plan for getting as many trains to Southern Cross as quickly as possible, given the current state of the rail network.

- Once the plan is verified instructions are sent to the rail network operator to follow the new schedule immediately. The operator reprograms the scheduling system, which informs drivers of the change in their route.

- Train drivers inform passengers of the changes to the route and request them not to get off at Southern Cross station.

8. Although the threat to the city does not justify an evacuation, in order to manage the major flood of people exiting the city the decision has been taken to close off all ingress to the city centre by road. VicRoads and personal navigation systems are updated with this information. Personnel in the control centre are able to dynamically define an exclusion zone, and a traffic routing algorithm identifies optimal alternate routes and diversions to avoid the area that is subject to the plume. This model takes into account real-time traffic-flow information provided to VINE by VicRoads, and is intended to ensure that the road closures do not lead to traffic jams that have the potential to make it difficult to evacuate the CBD, should that become necessary.

- Incident controllers make the decision to exclude incoming traffic from an area around the plume in order to make sure that the outbound flow and volume is as smooth as possible.

- At the point at which road closures are contemplated, VicRoads representatives are contacted to take part in the planning process.

- The traffic model is initialised to create diversions for two different-sized exclusion zones: one that only covers the exact predicted area of the plume, and another than defined a buffer zone around the area to account for forecasting inaccuracies. In each case, the model will use current traffic conditions (derived from both VicRoads SCATS data as well as social media traces) and historical data to generate an optimal set of diversions, and to predict the degree of disruption to traffic throughout the area.

- The result of the model suggests that the larger exclusion zone will lead to traffic jams throughout the inner city, so incident controllers decide to only activate the smaller exclusion area.

9. Throughout these events, the toxic plume expert has been monitoring both forecasts and the current data on the progress of the plume. He is able to use this information to calculate the concentration of toxins within the plume. As time passes and the plume dissipates the expert is able to conclude that the toxin concentration has fallen below the level where it is a threat to human life and EPA are also able to confirm this outcome. He enters this assessment into VINE, which is then immediately conveyed to the State Control Centre. The Incident Controller is satisfied with the expert's assessment that the threat has passed and declare the emergency over.

- The expert (still located at home) receives the latest plume information directly from VINE. Once he is satisfied that it no longer presents a threat,

he initiates a workflow within the system. This workflow requires multiple approval steps from senior emergency services officer before the all-clear is issued.

- One of the officers on the approval chain doesn't respond within the period required of them. The workflow automatically reassigns the approval to another officer with equal or higher authority.
- Once all the approvals have been gathered, the gas plume is declared harmless and the emergency is over.
- The end of the emergency is propagated through VINE to all parties affected by it, including VicRoads, the stadium operator, the train operator, etc. The general public is also notified using various channels, including social networks. The task of returning all conditions to normal begins.

10. Data that has been collected during the course of the emergency (performance of the plume modelling, effectiveness of the road and rail diversions that took place) is all captured in VINE and analysed to further calibrate these systems in preparation for the next emergency.

- Continuous tracking and monitoring capabilities of workflows, events, and responses to these events as well as the evolving status of resources and threats are built into VINE. These capabilities allow a complete data collection that increases the knowledge base and improves preparedness for future emergencies.

12.3 Scenario 3: Urban Fire and Toxic Gas Release

12.3.1 Synopsis

Major floods occurring in Victoria stress emergency management services. A small community in the mountains must survive the floods despite loss of power and limited emergency services support.

12.3.2 Detailed Scenario

1. Weather forecast predicts extremely heavy rains in the Alpine National Park on the following day. The heavy weather alert causes hydrological models in VINE to be run, in order to predict the extent of flooding. Modelling suggests that the high country community of High Falls, as well as the High Falls Hydroelectric plant, could be subject to flooding or isolation due to road closures.

- A weather forecast of heavy rain in the mountains is received.
- Bureau of Meteorology issues a flood warning for the high country, which is automatically published to VINE.

- Flood warning automatically triggers a DSE hydrological flood model via Floodzoom for the area identified. The flood model uses current soil moisture information and dam levels, combines this with the forecasted rainfall and a detailed topographic map data from VicMap in order to create a water runoff model from the coming rains.

- Floodzoom uses information from VINE to generate the flood model showing location of communities, critical infrastructure and other assets that will be impacted.

- A workflow is initiated to relevant agency personnel with tailored information to support the development of the plans for relief and recovery.

- Concerns about High Falls and the hydro plant are serious enough that they are escalated to an Incident Controller. An emergency readiness and response plan begins to be formulated.

2. The hydroelectric plant is identified as at-risk essential infrastructure. Furthermore, as it has a dam, it could potentially make the flooding worse. The dam operator is engaged, and with reference to a DSE module connected to the Floodzoom model which considers different discharge schedules. Using real-time feedback from the model, Incident Controller and agency commanders together with the plant operator, decide on the best option, and the sluice gates are opened.

- VINE stores the information of responsible personnel within the hydroelectric plant who can be contacted for emergency response. The plant operators are contacted and have an emergency conference call to discuss possible responses.

- The dam operator explains different operational possibilities in terms of release schedules, flow rates, etc. These scenarios are fed into the Floodzoom hydrological model, which rapidly updates its predictions based on each scenario. Emergency services personnel, together with the dam operator, decide on a plan that best balances the actions, potential consequences and needs for those upstream and downstream of the dam.

- The selected plan is published to VINE, along with the predicted flooding as the result of enacting that plan; a workflow also commences resulting in the dam operator opening the sluice gates.

3. Because of the imminent isolation of communities resulting from the flood, along with possible contamination of the town water supply as well as disruption of the power supply in High Falls, it is decided that water, food and other basic supplies must be stockpiled at strategic points in the town. Nearby supermarkets and distributors are engaged through VINE and volunteer significant supplies. An optimisation algorithm is used to identify where in the town to stock supplies and how to use the limited trucking resources in the area to get the supplies to the necessary locations before the flood hits.

- VINE is queried for nearby supermarkets and food distribution warehouses. These are all contacted to see if they are willing to provide supplies for the imminent emergency. Pledged supplies are entered into VINE and cross-referenced with available depots of emergency supplies.

- An optimisation model takes information about the location of government buildings, the town road network, and topography (high ground) in order to select the best possible set of temporary supply depots for the town.

- Once a set of depots has been selected, a further optimisation model combines this information with data about current supply location (eg. supermarkets) and information about local transportation (trucks) and road network in order to plan how the supplies will be picked up and redistributed.

- Personnel (including private truck drivers) are informed via VINE of the plan, including personalised instructions for the route options; based on the current situation and predictions.

4. It starts raining and the flood begins. Although High Falls is on higher ground and escapes the worst of the flooding, the only road into town is washed out, isolating the community. Furthermore, several properties in nearby low-lying areas are flooded, stranding people on the roofs of their houses. The local emergency services personnel coordinate on site to rescue the stranded people and move them to safer locations preregistered in VINE, including houses whose owners have volunteered to accept people in an emergency. A registry is maintained of first-aid qualified residents in the area who are willing to volunteer their time. Medical staff are allocated where necessary to provide additional medical support. This assignment is registered via role delegation in VINE.

- The status of access to High Falls is constantly available from VINE. This includes information from VicRoads, local emergency services, and social media.

- A helicopter and satellite imagery, combined with social media monitoring provides up to date information the flood extent. Flood levels, as well as reports of isolated residents, are constantly updated into VINE. This allows emergency personnel to begin rescuing isolated people requiring rescue even if they have been unable to call triple zero (000), while updated information about flooding is fed back into the hydrological model in order to refine its predictions.

- VINE is also queried to locate all the possible local resources that can be used to facilitate rescue and assistance to endangered people. This includes registered residents with first-aid training, medical staff, people who are accredited to perform water rescues with boats or any other special equipment which can assist with rescue, including location of sandbags and filling locations. Those who are able to be contacted and agree to

participate in the emergency response are registered in VINE as having a specific role and are immediately granted access rights to data in the system that is required for the fulfilment of their responsibilities. The tasking of responsibilities is tracked and stored in VINE.

5. Continued heavy stream flow from higher up the mountain places the hydroelectric plant under stress, eventually causing severe damage to some components and rendering the plant inoperative. Combined with earlier damage to the area power grid, High Falls and the surrounding area are now left with a significant electricity shortage. VINE is used to identify power use in critical facilities such as nursing homes and medical clinics, but there is not enough power remaining to serve households. Food in peoples' refrigerators begins to perish.

- As part of the emergency response, the duty managers at the power plant have been given additional privileges within VINE, allowing to enter information about the state of the plant directly into VINE.

- When the plant fails, the operator is able to immediately provide notification of the loss of power via VINE. Limited power remains available from a secondary transmission line connected to the national grid, but is insufficient to meet demand. Commanders are informed of the situation via VINE, and with the help of a simulation of power usage at critical facilities and advice from the electric utility company, devise a plan for keeping critical facilities (hospitals, nursing homes, cellular telephone network, the local control centre) operational.

6. Due to the threat of contamination in the water supply, the community is advised not to drink any tap water and instead to get bottled water from their nearest supply depot. Emergency services personnel use VINE to find trustworthy community members (e.g. doctors, bank managers, etc.) to whom they can delegate the duty of handing out supplies from the central depots. Using VINE, information about the distribution is communicated along multiple channels.

- With households' food supply dwindling due to the power outage, as well as the risk of contaminated water, commanders begin to mobilise their plan for distributing supplies to the townspeople.

- Given the knowledge of location of supply depots, as well as an up-to-date picture of the situation in town (including a number of road closures due to mudslides) optimisation algorithms are rerun in order to determine which segments of the town to direct to which supply depot.

- Personnel with the responsibility of distributing supplies are given instructions on how to proceed. As some of them are volunteers who, due to the power outage, have limited access to communication technologies, detailed operational information is also broadcast on ABC radio, who themselves obtain their information from VINE.

- Once the people in charge know what they have to do, the information on getting supplies is shared with the community. The central importance of radio in the context of power outage means that commanders work closely with the local ABC presenters to disseminate the necessary information. This is facilitated by a shared information view on top of VINE.

- Personnel handing out supplies use smartphones to record what was issued. This allows VINE to have an accurate picture of the quantity of remaining supplies. If they become critical, aircraft can be called to drop additional supplies.

7. Five days after the initial flooding, supplies of bottled water are running low. Emergency services personnel use portable water quality monitoring kits to continuously evaluate the quality of the local water supply, entering the results into VINE, from where it can be monitored by the Department of Health. As soon as the readings return below the critical safety levels, DoH issues an alert via VINE that the water supply is once again safe for drinking. Personnel in charge of supply depots receive an alert to issue no more bottled water.

- Being the information about water possible source of panic in the wider community, this has to be vetted by an authoritative entity. Department of Health personnel already have access to VINE and have roles matching their capability and expertise and are in a position to evaluate these readings and then communicate that the water can be drunk again.

- Once Department of Health issues the "all clear", personnel at the supply deposits are immediately told to stop distributing bottled water and to communicate to anyone that tap water is now safe again. This maximises the amount of bottled water available in case the supplies become contaminated again.

- During the five days after the flooding personnel on the field (emergency services personnel, volunteers, etc.) entered into VINE useful information about the condition of facilities, such as roads, power network, public buildings, as well as casualties. This information is used in VINE to define a recovery plan for the town of High Falls, which is place even before the town is accessible.

- This reduces the total amount of time of the recovery process as required resources can be mobilised and prepared in advance.

8. The weather conditions further deteriorate and it is clear that Melbourne is at risk of serious flash flooding. Recognising the potential consequences VINE provides a list of additional suitably qualified personnel to respond to the Melbourne flooding. The predictive tools indicate large extents of Victoria will be inundated for extensive periods. The State Flood Controller assesses the risks across the state, and plans for the most effective utilisation of the available resources which will include information about equipment and staff,

their skills, their location and their rostering across government and the private sector. People assuming their roles gain immediate access to all information and tools in VINE that their new role allows.

- The Floodzoom model processes all information available from the Bureau of Meteorology and other sources. A flash flood alert is issued by the Bureau of Meteorology for Melbourne and surrounds and is published into VINE. VINE immediately notifies the relevant people that they need to begin planning for a new emergency.

- The State Flood Controller makes the decision that some of the personnel and resources currently being used to manage the emergency in High Falls will be redeployed to the potentially more urgent situation in Melbourne. All locations of personnel and equipment and the number of hours they have been deployed are taken into account in making the redeployment decisions.

- Once suitable people are identified to take on the responsibilities of departing personnel, workflows within VINE are generated in order to facilitate an orderly handover. This includes alerting personnel to their new responsibilities, providing situational reports relevant to those responsibilities, and opening a channel of communication with those who previously had their role. Through VINE, personnel immediately have access to all information and analytics that are pertinent to their newly assigned role. At all points in time it is clear which person is responsible for a particular role.

- The handover procedures and delegation of roles are in accordance with the protocols and conventions set up under AIIMS for which VINE provides support in terms of tracking and logistics.

13 References

ABC. (2012). *ABC Emergency*. Retrieved September 2012 from http://www.abc.net.au/news/emergency/.

Adam, F., Philips-Wren, G., Teixeira, C., Respicio, A., & Telhada, J. (2010). Bridging the Socio-Technical Gap in Decision Support Systems: Challenges for the Next Decate. *Frontiers in Artificial Intelligence and Applications , 212*.

AGD. (2012). *Australian Emergency Management*. Retrieved September 2012 from http://www.em.gov.au/Pages/default.aspx

AGD CAP. (2012). *Common Alerting Protocol project*. Retrieved September 2012 from http://www.em.gov.au/cap

AIIMS. (2012). *The Australasian Inter-service Incident Management System*. Retrieved October 2012 from http://knowledgeweb.afac.com.au/aiims

Amazon. (2012). *Amazon Elastic MapReduce (Amazon EMR)*. Retrieved October 2012 from http://aws.amazon.com/elasticmapreduce/

Anton, P., Anderson, R. H., Mesic, R., & Scheiern, M. (2004). *Finding and Fixing Vulnerabilities in Information Systems: The Vulnerability Assessment and Mitigation Methodology.* Rand Publishing.

Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Kinwinski, A., et al. (2010). A View of Cloud Computing. *Communications of the ACM , 53* (4), 50-58.

Atom. (2012). *Atom Syndication Format (Atom)*. Retrieved October 2012 from http://en.wikipedia.org/wiki/Atom_(standard)

Australian Goverment. (2012). *Emergency Alert*. Retrieved September 2012 from http://www.emergencyalert.gov.au/

Australian Government. (2012). *Common Alerting Protocol – Australia Profile (CAP-AU-STD)*. Retrieved September 2012 from https://govshare.gov.au/xmlui/handle/10772/6493

Bertino, E., Terzi, E., Kamra, A., & Vaka, A. ".-p. (2005). Intrusion detection in RBAC-administered databases. *Proceedings of the 21st Annual Computer Security Applications Conference.* IEEE.

BOM. (2012). *Bureau of Meteorology*. Retrieved September 2012 from http://www.bom.gov.au/

BOM JATWC. (2012). *Joint Australian Tsunami Warning Centre*. Retrieved September 2012 from http://www.bom.gov.au/tsunami/about/tsunami_warnings.shtml

CAPAN. (2012). Retrieved October 2012 from Canadian Profile of the Common Alerting Protocol (CAP-CP): http://capan.ca/index.php/en/cap-cp/

CAPAN. (2012). *Canadian Association for Public Alerting and Notification (CAPAN)*. Retrieved September 2012 from Http://ww.capan.ca

CFA. (2012). *Victorian Bushfire Information Line (VBIL)*. Retrieved September 2012 from http://www.cfa.vic.gov.au/warnings-and-incidents/bushfire-info-line.htm

Cisco Systems. (2010). *Defeating DDoS Attacks.*

CMA. (2012, July). *China Meteorological Administration (CMA)*. Retrieved October 2012 from http://www.cma.gov.cn/en/

Commissioner. (2013, May). *Information Interoperability Blueprint.* Victorian Fire Services Commissioner, Melbourne.

Commissioner. (2011). *Review of the 2010-2011 Flood Warnings & Response.* Fire Services Commissioner Victoria. Melbourne: Premier of Victoria.

DMTF. (2012). *Distributed Management Task Force Inc.* Retrieved October 2012 from Common Information Model (CIM): http://dmtf.org/standards/cim

DSE. (2012). *Department of Sustainability and Environment*. Retrieved October 2012 from FireWeb: http://www.dse.vic.gov.au/fire-and-other-emergencies/wildfire-control-equipment-technology

Emergency Management Australia. (2012). *Manual 03 – Australian Emergency Management Glossary.* Retrieved October 2012 from http://www.em.gov.au/Documents/Manual03-AEMGlossary.PDF

EMSINA. (2012, July). *All Hazard Symbology.* Retrieved September 2012 from http://knowledgeweb.afac.com.au/__data/assets/pdf_file/0006/41838/EMSINA_ALL_HAZARDS_SYMBOLOGY_REPORT_FINAL_July10.pdf

EMSINA. (2010). *Emergency Management Spatial Information Network Australia*. Retrieved September 2012 from http://www.emsina.net/

Eugster, P. T., Felber, P. A., Guerraoui, & Kermarrec, A.-M. (2003). The Many Faces of Publish/Subscribe. *ACM Computing Surveys , 35* (2), 114-131.

EXI Working Group. (2011, March 10). *Efficient XML Interchange (EXI) Format 1.0.* Retrieved October 30, 2012 from W3C: http://www.w3.org/TR/2011/REC-exi-20110310/

FEMA CMAS. (2012). Retrieved October from Commercial Mobile Alert System: http://www.fema.gov/commercial-mobile-alert-system#7

FEMA. (2012). *Integrated Public Alert and Warning System (IPAWS)*. Retrieved October 2012 from http://www.fema.gov/integrated-public-alert-warning-system

GDACS. (2012). *Global Disaster Alert and Coordination System*. Retrieved October 2012 from http://www.gdacs.org/

GeoRSS. (2012). *GeoRSS*. Retrieved October 2012 from http://georss.org/Main_Page

Geoscience . (2012). *Geoscience Australia*. Retrieved October 2012 from Australian Government: http://www.ga.gov.au/

Google. (2012). *Google Crisis Response*. Retrieved September 2012 from http://www.google.org/crisisresponse/index.html

Google, Inc. (2012, October 30). *Protocol Buffers - Google's data interchange format*. From Google Code: http://code.google.com/p/protobuf/

Government of Canada, Translation Bureau. (2012). *Emergency Management Vocabulary.* Retrieved September 2012 from http://www.btb.gc.ca/publications/documents/urgence-emergency.pdf

Graham, S., Hull, D., & Murray, B. (2006). *WS-BaseNotification Specification.* Retrieved October 2012 from http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-os.pdf

Gruber, T. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition , 5* (2), 114-131.

Hollnagel, E., Woods, D., & Leveson, N. (2006). *Resilience Engineering: Concepts and Precepts.* Hampshire, England: Ashgate Publishing Co.

Huang, Y., & Gannon, D. (2006). A Comparative Study of Web Services-based Event Notification Specifications. *International Conference on Parallel Processing Workshops* (p. 8). IEEE.

IDIRA. (2012). *Interoperability of data and procedures in large-scale multinational disaster response actions (IDIRA)*. Retrieved September 2012 from http://www.idira.eu/

Internet Engineering Task Force. (2011). *RFC 6101 - The Secure Sockets Layer (SSL) Protocol, Version 3.0.* Retrieved October 2012 from http://tools.ietf.org/html/rfc6101

IRCAN. (2011, June). *CA/US Enhanced Resilience (CAUSE) Experiment* . Retrieved September 2012 from http://ircan-rican.gc.ca/news/763

ISO. (2012). *International Standards Organisation*. Retrieved October 2012 from Geographic infomation/Geomatics: http://www.isotc211.org/

ISO. (2012). *ISO 19125-1:2004*. Retrieved October 2012 from Geographic information - Simple feature access: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=40114

Jackson, S. (2009, September). *One Source One Message (OSOM)*. Retrieved September 2012 from http://www.cfaconnect.net.au/news/awesome-web-based-messaging-system.html

JMA. (2012). *Japan Meteorological Agency*. Retrieved September 2012 from http://www.jma.go.jp/jma/indexe.html

MASAS. (2012). *MASAS Information eXchange (MASAS-X) Portal*. Retrieved September 2012 from Government of Canada: http://www.masas-x.ca/

Mell, P., & Grance, T. (2011). *The NIST Definition of Cloud Computing.* NIST - National Institute of Standards and Technology, Computer security division. NIST.

Natural Resources Canada. (2012). *GeoConnections*. Retrieved October 2012 from http://geoconnections.nrcan.gc.ca/home

Newcomer, E., Robinson, I., Feingold, M., & Jeyaraman, R. (2009). *WS-Coordination Specification.* From http://docs.oasis-open.org/ws-tx/wstx-wscoor-1.2-spec-os/wstx-wscoor-1.2-spec-os.html

NICTA. (2012). Retrieved October 2012 from National ICT Australia: http://www.nicta.com.au

NICTA CWML. (2006, December). *Cyclone Warning Markup Language.* Retrieved October 2012 from http://nicta.com.au/__data/assets/pdf_file/0005/8645/CWML-10.pdf

NICTA TWML . (2006, December). *Tsunami Warning Markup Laguage (TWML).* Retrieved September 2012 from http://www.nicta.com.au/__data/assets/pdf_file/0007/7567/TsunamiWarningML-V10.pdf

OASIS . (2012). *CAP-AP EVENT CODES LIST.* Retrieved September 2012 from https://www.oasis-open.org/committees/download.php/41768/CAP-AP%20Discussion%20Paper.2.pdf

OASIS EDXL-CAP. (2012, July 1). *Emergency Data Exchange Language (EDXL) Common Alerting Protocol (CAP) Version 1.2. OASIS Standard.* Retrieved September 2012 from http://docs.oasis-open.org/emergency/cap/v1.2/CAP-v1.2-os.doc

OASIS EDXL-CAP-AU. (2012, April 26). *Emergency Data Exchange Language (EDXL) Common Alerting Protocol (CAP) v1.2 Australia (AU) Profile Version 1.0.* Retrieved September 2012 from http://docs.oasisopen.org/emergency/edxl-cap1.2-au/v1.0/cs01/edxl-cap1.2-au-v1.0-cs01.html

OASIS EDXL-DE. (2006, May 1). *Emergency Data Exchange Language (EDXL) Distribution Element (DE) Version 1.0. OASIS Standard EDXL-DE*. Retrieved September 2012 from , http://docs.oasis-open.org/emergency/EDXL-DE/V1.0

OASIS EDXL-HAVE. (2009, December 22). *Emergency Data Exchange Language (EDXL) Hospital Availability Exchange (HAVE) Version 1.0. OASIS Standard Incorporating Approved Errata.* Retrieved September 2012 from http://docs.oasis-open.org/emergency/edxl-have/v1.0/errata/edxl-have-v1.0-os-errata-os.html

OASIS EDXL-RM. (2009, December 22). *Emergency Data Exchange Language (EDXL) Resource Messaging (RM) Version 1.0. OASIS Standard Incorporating Approved Errata.* Retrieved September 2012 from http://docs.oasis-open.org/emergency/edxl-rm/v1.0/errata/EDXL-RM-v1.0-OS-errata-os.html

OASIS EDXL-TEP. (2012, June). *Emergency Data Exchange Language (EDXL) Tracking of Emergency Patients (TEP) Version 1.0. OASIS Working Draft 01.* Retrieved September 2012 from https://www.oasis-open.org/committees/download.php/46264/edxl-tep-v1.0-wd01.odt

OASIS EM TC. (2012). *OASIS Emergency Management Technical Committee*. Retrieved September 2012 from https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=emergency#technical

OASIS. (2012, september). *OASIS*. From https://www.oasis-open.org/

OASIS Open. (2004, October 19). *UDDI Version 3.0.2*. Retrieved October 29, 2012 from UDDI Specification: http://uddi.org/pubs/uddi_v3.htm

OESC. (2012). *Office of the Emergency Services Commissioner Victoria*. Retrieved October 2012 from http://www.oesc.vic.gov.au/

OGC AS. (2012). *Open Geospatial Consortium Abstract Specification*. Retrieved October 2012 from http://www.opengeospatial.org/standards/as

OGC GML. (2012). *Open Geospatial Consortium - GML*. Retrieved October 2012 from Geography Markup Language (GML): http://www.opengeospatial.org/standards/gml

OGC KML. (2012). *Open Geospatial Consortium - KML.* Retrieved September 2012 from Keyhole Markup Language (KML): http://www.opengeospatial.org/standards/kml/

OGC. (2012). *Open Geospatial Consortium*. Retrieved October 2012 from http://www.opengeospatial.org

OGC WPS. (2012). *Open Geospatial Consortium - Web Processing Service*. Retrieved October 2012 from http://www.opengeospatial.org/standards/wps

Open Geospatial Consortium. (n.d.). *Web Map Service*. Retrieved October 2012 from http://www.opengeospatial.org/standards/wms

Oracle . *Oracle Complex Event PRocessing : Lightweight Modular Application Event Stream Processing in the REal World".*

Oracle America, Inc. . (2002, May 9). *JSR 47: Logging API Specification*. Retrieved October 29, 2012 from Java Community Process: http://jcp.org/en/jsr/detail?id=47

Oracle, Inc. (n.d.). *JAX-RS Refrence Implementation*. Retrieved October 30, 2012 from Glassfish: http://jersey.java.net/

Oracle, Inc. (n.d.). *JAX-WS Reference Implementation*. Retrieved October 30, 2012 from Glassfish: http://jax-ws.java.net/

Paton, D., & Johnston, D. (2006). *Disaster Resilience: An Integrated Approach.* Charles C Thomas Pub Ltd.

Pelmorex. (2012). *National Alert Aggregation and Dissemination System (NAAD)*. Retrieved 2012 September from Pelmorex Alerting Services: http://alerts.pelmorex.com/en/

Robinson, I. (2006, May 11). *J2EE Activity Service for Extended Transactions.* Retrieved October 20, 2012 from Community Development of Java Technology Specifications: http://jcp.org/aboutJava/communityprocess/final/jsr095/index.html

Robinson, I. (2006). *J2EEtm activity service for extended transactions.* From http://jcp.org/aboutJava/communityprocess/final/jsr095/index.html

Sahana Foundation. (2012). *Sahana Software Foundation*. Retrieved September 2012 from http://sahanafoundation.org

Sarvodaya. (2012). *Lanka Jathika Sarovdaya Shramadan Sangamaya*. Retrieved September 2012 from http://www.sarvodaya.org

Scarfone, K., & Mell, P. (2007). *Guide to intrusion detection and prevention systems (idps).* NIST.

Seacord, R. (2006). *Secure Coding in C and C++.* Upper Saddle River, NJ: Addison-Wesley.

Smoot, S. (2011). *Private Cloud Computing: Consolidation, Virtualization, and Service-Oriented Infrastructure.* Morgan Kaufmann.

Societe Generale. (2011). *DDos Incident Response: Guidelines to handle Distributed Denial of Service incidents.* Incident Response Methodology.

State Government of Victoria. (2011). *Review of the 2010-2011 Flood Warnings and Response Final Report.*

Swiderski, F. &. (2004). *Threat Modeling.* Redmond, WA: Microsoft Press.

The OpenStack Foundation. (n.d.). *OpenStack*. From OpenStack Cloud Software: http://www.openstack.org/

UICDS. (2012). *Unified Incident Command and Decision Support*. Retrieved September 2012 from http://www.uicds.us/

US Department of Justice. (2012). *Global Justice XML Data Model (Global JXDM).* Retrieved September 2012 from http://www.it.ojp.gov/jxdm/

US GS. (2012). *U.S. Geological Survey*. Retrieved October 2012 from http://www.usgs.gov/

US NWS. (2012). *National Weather Service (U.S.)*. Retrieved October 2012 from http://www.weather.gov/

VBRC. (2010, July 31). *2009 Victorian Bushfires Royal Commission Report.* Retrieved October 2012 from http://www.royalcommission.vic.gov.au/Commission-Reports

Victorian Ombudsman. (2012, April). *The Victorian Ombudsman's investigation into ICT-enabled projects.* Retrieved October 2012 from http://www.ombudsman.vic.gov.au/resources/documents/Procurement_and_contract_management_VGSO_presentation_by_Erin_Barlow_26_April_2012.pdf

W3C. (2012). *Extensible Markup Language (XML)*. Retrieved October 2012 from http://www.w3.org/XML/

WMO. (2012). *WMO Register of Alerting Authorities*. Retrieved October 2012 from http://www-db.wmo.int/alerting/authorities.html

Wu, E. (n.d.). High-performance complex even processing over streams.

Yi, H., & D., G. (2006). A comparative study of Web services-based event notification specifications,. *International Conference on Parallel Processing Workshops*, (p. 8).

Zarharia, M., Borthakur, D., Sen Sarma, J., Elmeleegy, K., Shenker, S., & Stoica, I. (2009). *Job scheduling for multi-user mapreduce clusters.* University of California, EECS Department, Berkeley.

Zesty. (2012). *People Finder Interchange Format (PFIF)*. Retrieved September 2012 from http://zesty.ca/pfif/

14 Glossary

AGD	Australian Attorney-General's Department
AIIMS	Australasian Inter-service Incident Management System
API	Application Programming Interface
AUEventList	Australian Event Code List
AVT	Automatic Vehicle Tracking
BOM	Bureau of Meteorology
CAD	Computer Aided Dispatch
CAP	Common Alerting Protocol
CAP SG	CAP-AU Stakeholder Group
CAP-AP	CAP Australian Profile
CAP-AU-STD	CAP Australian Standard
CAPAN	Canadian Association for Public Alerting and Notification
CEP	Complex Event Processing
CFA	Country Fire Authority
CIM	Common Information Model

CMA	China Meteorological Administration
CPEAS	China Public Emergency Alerting System
CSE	Core Service Extension
CWML	Cyclone Warning Mark-up Language
DNS	DOMAIN NAME SYSTEM
DoH	Victorian Department of Health
DPI	Victorian Department of Primary Industries
DSE	Department of Sustainability and Environment
EDXL	Emergency Data eXchange Language
EDXL	Emergency Data Exchange Language
EDXL-DE	Emergency Data Exchange Language – Distribution Element
EDXL-HAVE	Emergency Data Exchange Language – Hospital Availability Data Exchange
EDXL-RM	Resource Messaging
EDXL-SitRep	Emergency Data Exchange Language – Situation Report
EDXL-TEP	Emergency Data Exchange Language Tracking of Emergency Patients
EMSINA	Emergency Management Spatial Information Network Australia
EPA	Environmental Protection Authority
ESB	Enterprise Service Bus
ESO	Emergency Service Organisation
ESTA	Emergency Services Telecommunications Authority
FPT	Federal, Provincial and Territorial
GDACS	Global Disaster Alert and Coordination System
GIS	Geographic Information System
Global JXDM	Global Justice XML Data Model
IaaS	Infrastructure-as-a-Service

ICT	Information and Communication Technology
IDIRA	Interoperability of data and procedures in large-scale multinational disaster response actions
IEPD	Information Exchange Package Documentation
IMS	Incident Management System
IMS	Incident Management Systems
IPAWS	Integrated Public Alert and Warning System
JATWC	Joint Australian Tsunami Warning Centre
JMA	Japan Meteorological Agency
JMX	Japan disaster prevention information XML
JSON	JavaScript Object Notation
KML	Keyhole Mark-up Language
LDAP	Lightweight Directory Access Protocol
MASAS-X	Multi-Agency Situational Awareness System Information Exchange
MAV	Municipal Association of Victoria
MDN	Mobile Data Network
MDT	Mobile Data Terminals
MFB	Melbourne Fire Brigade
MSE	Modular Service Extension
NAAD	National Alert Aggregation and Dissemination System
NGO	Non-Government Organization
NICTA	National ICT Australia
NIDS	Network Intrusion Detection Systems
NIEM	National Information Exchange Model
NOAA	National Oceanic and Atmospheric Administration
NWS	National Weather Service
OASIS	Organisation for the Advancement of Structured Information Standards

OESC	Office of Emergency Services Commissioner
OGC	Open Geospatial Consortium
OSOM	One Source One Message
PaaS	Platform-as-a-Service
PFIF	People Finder Interchange Format
PSES	Publish/Subscribe Event Service
REST	Representational State Transfer
RPC	Remote Procedure Call
SaaS	Software-as-a-Service
SES	State Emergency Service
SLA	Service Level Agreement
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SSO	Single Sign On
TWML	Tsunami Warning Mark-up Language
UICDS	Unified Incident Command and Decision Support
USGS	U.S. Geological Survey
VBIL	Victorian Bushfire Information Line
VicSes	Victorian State Emergency Services
VINE	Victorian Information Network for Emergencies
VM	Virtual Machine
XML	eXtensible Markup Language

15 Appendixes

15.1 Appendix A

Sample of an EDXL-DE message with a CAP v1.1. payload (highlighted).

```
<?xml version="1.0" encoding="UTF-8"?>
<EDXLDistribution xmlns="urn:oasis:names:tc:emergency:EDXL:DE:1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:emergency:EDXL:DE:1.0 EDXL-DE.xsd">
<distributionID>ieam_e3_2</distributionID>
<senderID>XML2005</senderID>
<dateTimeSent>2005-11-15T16:53:00-05:00</dateTimeSent>
<distributionStatus>Exercise</distributionStatus>
<distributionType>Update</distributionType>
<keyword>
<valueListUrn>http://www.niem.gov/EventTypeList</valueListUrn>
<value>Explosion</value>
</keyword>
<targetArea>
<polygon>33.4745,-112.1174 33.4745,-112.0238 33.4238,-112.0238 33.4238,-
112.1174 33.4745,-16 112.1174 </polygon>
</targetArea>
<contentObject>
<contentDescription>CAP message from DOT advising best alternate
Routes</contentDescription>
<xmlContent>
<embeddedXMLContent>
<alert xmlns="urn:oasis:names:tc:emergency:cap:1.1">
<identifier>Vendor generated</identifier>
<sender>AZ DOT</sender>
<sent>2005-11-15T16:58:00-05:00</sent>
<status>Exercise</status>
<msgType>Update</msgType>
<scope>Public</scope>
<info>
```

```
<category>Transport</category>
<event>Traffic Routes</event>
<urgency>Immediate</urgency>
<severity>Moderate</severity>
<certainty>Likely</certainty>
<description>Traffic adjustments ensure clear routes to St. Josephs Hospital and
Phoenix Childrens Hospital on Thomas Rd. </description>
<area>
<areaDesc>Best Routes</areaDesc>
<polygon>38.91655012246089,-77.02016267943407 38.91655012246089,-41
77.0117098391165 38.907662564641285,-77.0117098391165 38.907662564641285,-
77.0201626794340742 38.91655012246089,-77.02016267943407 </polygon>
</area>
</info>
</alert>
</embeddedXMLContent>
</xmlContent>
</contentObject>
</EDXLDistribution>
```

15.2 Appendix B

Sample of an EDXL-DE file with a EDXL-HAVE payload (highlighted).

```
<?xml version="1.0" encoding="UTF-8"?>
<EDXLDistribution xmlns="urn:oasis:names:tc:emergency:EDXL:DE:1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:emergency:EDXL:DE:1.0 EDXL-DE.xsd">
<distributionID>edxl_d1</distributionID>
<senderID>XMl2005</senderID>
<dateTimeSent>2005-11-15T16:53:00-05:00</dateTimeSent>
<distributionStatus>Test</distributionStatus>
<distributionType>Report</distributionType>
<keyword>
<valueListUrn>http://www.niem.gov/EventTypeList</valueListUrn>
<value>Pandemic Flu</value>
</keyword>
<targetArea>
<polygon>33.4745,-112.1174 33.4745,-112.0238 33.4238,-112.0238 33.4238,-
112.1174 33.4745,-112.1174</polygon>
</targetArea>
<contentObject>
<contentDescription>HAVE message reporting bed capacities</contentDescription>
<xmlContent>
<embeddedXMLContent>
<HospitalStatus xmlns="urn:oasis:names:ec:emergency:have:1.0:">
<Hospital>
<OrganizationInformation>
<OrganizationID>XXX1234</OrganizationID>
<OrganizationIDProviderName>AHA</OrganizationIDProviderName>
<OrganizationName>ABC Hospital</OrganizationName>
<OrganizationTypeText>Hospital</OrganizationTypeText>
<OrganizationLocation>
<StreetFullText>123 Main Street</StreetFullText>
<LocationCityName>Fairfax</LocationCityName>
<LocationCountryName>USA</LocationCountryName>
```

```
<LocationStateName>Virginia</LocationStateName>
</OrganizationLocation>
</OrganizationInformation>
<EmergencyDepartmentStatus>
<EMSTraffic>
<EMSTrafficStatus>Normal</EMSTrafficStatus>
</EMSTraffic>
<EMSCapacity>
<TriageRed>40</TriageRed>
<TriageYellow>40</TriageYellow>
<TriageGreen>40</TriageGreen>
<TriageBlack>40</TriageBlack>
</EMSCapacity>
<EMSCensus>
<TriageRed>20</TriageRed>
<TriageYellow>20</TriageYellow>
<TriageGreen>20</TriageGreen>
<TriageBlack>20</TriageBlack>
</EMSCensus>
<EMSAmbulanceStatus>
<EMSOffloadStatus>Normal</EMSOffloadStatus>
<EMSOffloadMinutes>20</EMSOffloadMinutes>
</EMSAmbulanceStatus>
</EmergencyDepartmentStatus>
<HospitalBedCapacityStatus>
<BedCapacity>
<Bed>AdultICU</Bed>
<Capacity>
<CapacityStatus>Vacant/Available</CapacityStatus>
<AvailableCount>10</AvailableCount>
<BaselineCount>30</BaselineCount>
<AdditionalCapacityCount24Hr>5</AdditionalCapacityCount24Hr>
<AdditionalCapacityCount72Hr>5</AdditionalCapacityCount72Hr>
</Capacity>
</BedCapacity>
</HospitalBedCapacityStatus>
<ServiceCoverageStatus>
<Burn>True</Burn>
<Cardiology>True</Cardiology>
<InfectiousDiseases>True</InfectiousDiseases>
<Neonatology>True</Neonatology>
<Neurology>True</Neurology>
<Orthopedic>True</Orthopedic>
<Surgery>
<General>True</General>
</Surgery>
</ServiceCoverageStatus>
<HospitalFacilityStatus>
<EOCStatus>Active</EOCStatus>
<EOCPlan>Active</EOCPlan>
<ClinicalStatus>Normal</ClinicalStatus>
<DeconCapacity>Inactive</DeconCapacity>
<MorgueCapacity>Open</MorgueCapacity>
<FacilityStatus>Normal</FacilityStatus>
<SecurityStatus>Normal</SecurityStatus>
</HospitalFacilityStatus>
<HospitalResourcesStatus>
```

```
<Staffing>Adequate</Staffing>
<FacilityOperations>Adequate</FacilityOperations>
<ClinicalOperations>Adequate</ClinicalOperations>
</HospitalResourcesStatus>
<LastUpdateTime>2001-12-17T09:30:47.0Z</LastUpdateTime>
</Hospital>
</HospitalStatus>
</embeddedXMLContent>
</xmlContent>
</contentObject>
</EDXLDistribution>
```

15.3 Appendix C

Example of an EDXL-DE file with a EDXL-RM payload (highlighted). It contains a resource request for electrical power restoration team.

```
<?xml version="1.0" encoding="UTF-8"?>
<EDXLDistribution xmlns="urn:oasis:names:tc:emergency:EDXL:DE:1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:emergency:EDXL:DE:1.0 EDXL-DE.xsd">
<distributionID>ieam_e3_2</distributionID>
<senderID>XML2005</senderID>
<dateTimeSent>2005-11-15T16:53:00-05:00</dateTimeSent>
<distributionStatus>Exercise</distributionStatus>
<distributionType>Update</distributionType>
<keyword>
<valueListUrn>http://www.niem.gov/EventTypeList</valueListUrn>
<value>Explosion</value>
</keyword>
<targetArea>
<polygon>33.4745,-112.1174 33.4745,-112.0238 33.4238,-112.0238 33.4238,-
112.1174 33.4745,-16 112.1174 </polygon>
</targetArea>
<contentObject>
<contentDescription>CAP message from DOT advising best alternate
Routes</contentDescription>
<xmlContent>
<embeddedXMLContent>
<RequestResource xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:emergency:EDXL:RM:1.0:msg EDXL-
RMRequestResource.xsd" xmlns="urn:oasis:names:tc:emergency:EDXL:RM:1.0:msg"
xmlns:rm="urn:oasis:names:tc:emergency:EDXL:RM:1.0"
xmlns:xpil="urn:oasis:names:tc:ciq:xpil:3" xmlns:xnl="urn:oasis:names:tc:ciq:xnl:3"
xmlns:xal="urn:oasis:names:tc:ciq:xal:3" xmlns:geo-
oasis="urn:oasis:names:tc:emergency:EDXL:HAVE:1.0:geo-oasis"
xmlns:gml="http://www.opengis.net/gml">
<MessageID>urn:au-qld-eoc:12332</MessageID>
<SentDateTime>2006-03-21T11:58:00+10:00</SentDateTime>
<MessageContentType>RequestResource</MessageContentType>
<OriginatingMessageID>urn:au-qld-eoc:12332</OriginatingMessageID>
<IncidentInformation>
<rm:IncidentDescription>Cyclone Larry</rm:IncidentDescription>
</IncidentInformation>
<ContactInformation>
<rm:ContactRole>Sender</rm:ContactRole>
```

```
<rm:AdditionalContactInformation>
<xpil:PartyName>
<xnl:PersonName>
<xnl:NameElement xnl:ElementType="FirstName">Alex</xnl:NameElement>
<xnl:NameElement xnl:ElementType="LastName">Jones</xnl:NameElement>
</xnl:PersonName>
<xnl:OrganisationName>
<xnl:NameElement>Dept of Emergency Services</xnl:NameElement>
</xnl:OrganisationName>
</xpil:PartyName>
<xpil:ContactNumbers>
<xpil:ContactNumber xpil:CommunicationMediaType="Telephone"
xpil:ContactHours="9:00AM - 5:00PM">
<xpil:ContactNumberElement
xpil:Type="CountryCode">61</xpil:ContactNumberElement>
<xpil:ContactNumberElement xpil:Type="AreaCode">7</xpil:ContactNumberElement>
<xpil:ContactNumberElement xpil:Type="LocalNumber">3000
1234</xpil:ContactNumberElement>
</xpil:ContactNumber>
</xpil:ContactNumbers>
<xpil:ElectronicAddressIdentifiers>
<xpil:ElectronicAddressIdentifier>alexj@emergencyservices.gov.au</xpil:ElectronicAdd
ressIdentifier>
</xpil:ElectronicAddressIdentifiers>
</rm:AdditionalContactInformation>
</ContactInformation>
<ResourceInformation>
<ResourceInfoElementID>003</ResourceInfoElementID>
<Resource>
<TypeStructure>
<rm:ValueListURN>urn:x-hazard:vocab:resourceTypes</rm:ValueListURN>
<rm:Value>Electrical Power Restoration Team</rm:Value></TypeStructure>
</Resource>
<AssignmentInformation>
<Quantity>
<rm:MeasuredQuantity>
<rm:Amount>2</rm:Amount>
</rm:MeasuredQuantity>
</Quantity>
<AnticipatedFunction>Restore power to critical infrastructure in and around the Innisfail
area </AnticipatedFunction>
</AssignmentInformation>
<ScheduleInformation>
<ScheduleType>RequestedArrival</ScheduleType>
<DateTime>2006-03-22T08:00:00+10:00</DateTime>
<Location>
<rm:TargetArea>
<gml:Point><gml:pos> 146.03 -17.53 </gml:pos></gml:Point>
</rm:TargetArea>
</Location>
</ScheduleInformation>
</ResourceInformation>
</RequestResource>
</embeddedXMLContent>
</xmlContent>
</contentObject>
</EDXLDistribution>
```

15.4 Appendix D

Example of an CAP V1.2 file (highlighted) containing a severe thunderstorm warning, as payload in a EDXL-DE wrapper message.

```
<?xml version="1.0" encoding="UTF-8"?>
<EDXLDistribution xmlns="urn:oasis:names:tc:emergency:EDXL:DE:1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:emergency:EDXL:DE:1.0 EDXL-DE.xsd">
<distributionID>ieam_e3_2</distributionID>
<senderID>XML2005</senderID>
<dateTimeSent>2005-11-15T16:53:00-05:00</dateTimeSent>
<distributionStatus>Exercise</distributionStatus>
<distributionType>Update</distributionType>
<keyword>
<valueListUrn>http://www.niem.gov/EventTypeList</valueListUrn>
<value>Explosion</value>
</keyword>
<targetArea>
<polygon>33.4745,-112.1174 33.4745,-112.0238 33.4238,-112.0238 33.4238,-
112.1174 33.4745,-16 112.1174 </polygon>
</targetArea>
<contentObject>
<contentDescription>CAP message from DOT advising best alternate
Routes</contentDescription>
<xmlContent>
<embeddedXMLContent>
<alert xmlns = "urn:oasis:names:tc:emergency:cap:1.2">
<identifier>KSTO1055887203</identifier>
<sender>KSTO@NWS.NOAA.GOV</sender>
<sent>2003-06-17T14:57:00-07:00</sent>
<status>Actual</status>
<msgType>Alert</msgType>
<scope>Public</scope>
<info>
<category>Met</category>
<event>SEVERE THUNDERSTORM</event>
<responseType>Shelter</responseType>
<urgency>Immediate</urgency>
<severity>Severe</severity>
<certainty>Observed</certainty>
<eventCode>
<valueName>SAME</valueName>
<value>SVR</value>
</eventCode>
<expires>2003-06-17T16:00:00-07:00</expires>
<senderName>NATIONAL WEATHER SERVICE SACRAMENTO CA</senderName>
<headline>SEVERE THUNDERSTORM WARNING</headline>
<description> AT 254 PM PDT...NATIONAL WEATHER SERVICE DOPPLER RADAR
INDICATED A SEVERE THUNDERSTORM OVER SOUTH CENTRAL ALPINE
COUNTY...OR ABOUT 18 MILES SOUTHEAST OF KIRKWOOD...MOVING
SOUTHWEST AT 5 MPH. HAIL...INTENSE RAIN AND STRONG DAMAGING WINDS
ARE LIKELY WITH THIS STORM.</description>
<instruction>TAKE COVER IN A SUBSTANTIAL SHELTER UNTIL THE STORM
PASSES.</instruction>
<contact>BARUFFALDI/JUSKIE</contact>
<area>
```

```
<areaDesc>EXTREME NORTH CENTRAL TUOLUMNE COUNTY IN CALIFORNIA,
EXTREME NORTHEASTERN CALAVERAS COUNTY IN CALIFORNIA,
SOUTHWESTERN ALPINE COUNTY IN CALIFORNIA</areaDesc>
<polygon>38.47,-120.14 38.34,-119.95 38.52,-119.74 38.62,-119.89 38.47,-
120.14</polygon>
<geocode>
<valueName>SAME</valueName>
<value>006109</value>
</geocode>
<geocode>
<valueName>SAME</valueName>
<value>006009</value>
</geocode>
<geocode>
<valueName>SAME</valueName>
<value>006003</value>
</geocode>
</area>
</info>
</alert>
</embeddedXMLContent>
</xmlContent>
</contentObject>
</EDXLDistribution>
```

15.5 Appendix E

Example of a CAP-AU message containing a notification about a small planned burn in Buninyong area.

```
<?xml version="1.0" encoding="UTF-8"?>
<alert xmlns="urn:oasis:names:tc:emergency:cap:1.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:emergency:cap:1.2
http://docs.oasis-open.org/emergency/cap/v1.2/CAP-v1.2.xsd">
<identifier>120800114</identifier>
<sender>info@cfa.vic.gov.au</sender>
<sent>2012-10-01T15:42:49+10:00</sent>
<status>Actual</status>
<msgType>Alert</msgType>
<source>CFA</source>
<scope>Public</scope>
<code> urn:oasis:names:tc:emergency:cap:1:2:profile:CAP-AU:1:0</code>
<note>Active</note>
<references>120800114</references>
<info>
<language>en-AU</language>
<category>Fire</category>
<event>Bushfire</event>
<responseType>None</responseType>
<urgency>Immediate</urgency>
<severity>Unknown</severity>
<certainty>Observed</certainty>
<eventCode>
<valueName>urn:oasis:names:tc:emergency :cap:1.2:profile:CAP-
AU:1.0:AUeventLIST:1.0</valueName>
```

<value>bushFire</value>
</eventCode>
<onset>2012-08-01T10:12:00+10:00</onset>
<expires>2012-08-02T10:12:00+10:00</expires>
<senderName>CFA</senderName>
<headline>PLANNED BURN, BUNINYONG</headline>
<description>SMALL PLANNED BURN, MIDLAND HWY, BUNINYONG,
CFA</description>
<contact>CFA</contact>
<area>
<areaDesc>MIDLAND HWY, BUNINYONG</areaDesc>
<polygon>-37.65670004108198,143.90896999331812</polygon>
</area>
</info>
</alert>

LEADERSHIP
INTEGRATION
ACCOUNTABILITY

firecommissioner.vic.gov.au

Published by the Fire Services Commissioner
121 Exhibition St
MELBOURNE 3000