# Scenario Search on the Grid of Environmental Data Sources

Mikhail Zhizhin* Alexei Poyda† Dmitry Mishin Dmitry Medvedev
Eric Kihn‡ Vassily Lyutsarev§

August 17, 2006

## Abstract

We present the Environmental Scenario Search Engine (ESSE), a set of algorithms and software tools for distributed querying and mining large environmental data archives. The principal requirement of the ESSE system is to allow the user to query the data in meaningful "human linguistic" terms. The mapping between human language and computer systems involves fuzzy logic. We use a data resource web service abstraction layer to virtualize spatio-temporal databases, providing our search engine with time-series of environmental parameters. The data resource interface is implemented as a set of OGSA-DAI components with simple input and output XML schemes. Time-series selected from a data resource in XML format can be mined for environmental events by the ESSE or used after XSLT transformation by other clients, Microsoft Excel 2003 being one of the examples.

## 1 Introduction and related work

Environmental informatics combines the research fields of Artificial Intelligence, Geographical Information Systems (GIS) Modeling and Simulation, and User Interfaces and is a rapidly expanding area of computer and natural science [1]. The increasing data volumes from todays collection systems and the need of the scientific community to include an integrated and authoritative representation of the natural environment in analysis requires a new approach to data mining, management and access [2]. The natural environment includes elements from multiple domains such as space, terrestrial weather, oceans and terrain. Systems such as the Global Change Master Directory (GCMD) from NASA[1] or the Master Environmental Library (MEL) from the DMSO[2] and others provide the ability to search metadata by keywords for links to archived environmental data sets distributed across the network, but the ability to search for specific scenarios (sets of conditions) within the environmental data does not yet exist outside of systems based on the ESSE technology.

At the same time, the environmental modeling community has begun to develop several archives of continuous environmental representations. These archives contain a complete view of the Earth system parameters over a regular grid for a considerable period of time. The numerical models used to reproduce environmental parameters take all available observational data as initial conditions, so the resulting petabyte-size data sets jointly may be considered an authoritative high-resolution representation of terrestrial weather and the near-Earth space during the last 50 years [3, 4]. For example, the spatial resolution for the NCEP/NCAR Global Circulation Model (GCM) is $2.5°$(latitude) $\times$ $2.5°$(longitude) $\times$ $10^2$ (environmental parameters) $\approx 10^6$ grid values. The high-frequency GCM outputs the data every six hours of simulation time, resulting in $\sim 400$ Mb of data per simulation day. By contrast, the worldwide daily meteorological observational data collected over the Global Telecommunications Sys-

*Geophysical Center, Russian Academy of Sciences, Moscow, Russia
†Moscow State University, Russia
‡National Geophysical Data Center, NOAA, Boulder, CO, USA
§Microsoft Research, Cambridge, UK

[1] http://gcmd.nasa.gov
[2] https://mel.dmso.mil/

tem, is $\sim 200\,\mathrm{Mb}$[3]. The NCEP/NCAR Reanalysis project has run the GCM for more than 58 years providing $\sim 8\,\mathrm{Tb}$ of weather data.

In this paper we describe the Environmental Scenario Search Engine (ESSE), a set of algorithms and software tools for distributed querying and mining large environmental data archives. The prime requirement of the ESSE system design is to allow the user to query the data in meaningful "human linguistic" terms. Natural language is not easily translated into the absolute terms of 0 and 1 which make up the digital world. The mapping between human language and computer systems involves fuzzy logic. Fuzzy logic is a superset of conventional (Boolean) logic that has been extended to handle the concept of partial truth – truth values between "completely true" and "completely false". It was introduced by L. Zadeh in the 1960's as a means to model the uncertainty of natural language [5].

The ESSE acts as a bridge between questions a user needs to ask of the environment and the data which describes it. Imagine for example that the end user doesn't need all the weather data covering Moscow region for the last 50 years, but rather needs an example of an atmospheric front near Moscow. Further imagine that the user needs satellite images of the front and he wants to know how often such fronts occur or if they have been increasing in the last 10 years. The ESSE search engine and the data mining portal are addressing such enquiries.

The relational data model proposed in 1970 by E. Codd [6] and its implementation in the form of SQL DBMS's with possible fuzzy logic extensions [7], which is so successful in business applications, is not universally adopted for environmental data archives. Petabyte sized data products [3, 4] are still delivered in the form of file collections because the file structure like NetCDF[4] or HDF[5] is better for representing a multidimentional array data model than a set of related rows from two-dimensional tables. The UNIDATA[6] THREDDS server with OpenDAP network data access protocol attempts to aggregate different file formats under a single array-oriented Common Data Model. This ongoing unification effort currently does not support an XML format for data export and is not compatible with the emerging e-Science Data Grid standards [8, 9, 10].

In this paper we use a data resource web service abstraction layer to virtualize sequential databases providing our search engine with a time-series of environmental parameters. The data resource interface is implemented as a set of OGSA-DAI [8, 11] components with simple input and output XML schemas. Time-series selected from the data resource in XML format can be mined for environmental events by the ESSE or used after XSLT transformation [12] by other clients, Microsoft Excel 2003 being one of the examples. We show that using XML output format with GZIP data compression [13] requires CPU time and network bandwidth comparable to the NetCDF binary file serialization. Compliance with the OGSA-DAI specification and use of Java/J# language allowed us to deploy our data source and mining services into most of the existing web service and grid service containers including Microsoft ASP.NET[7], Apache Tomcat/Axis[8], WSRF Globus Toolkit 4[9], OMII[10], and EGEE gLite[11].

The rest of the paper is organized as follows. In Section 2 we define the "environmental event scenario" and introduce mathematics of the ESSE fuzzy data mining. In Section 3 we present ESSE toolkit implementation and describe two authoritative data sources for space and terrestrial weather. In Section 4 we present interactive data mining use case. Section 5 offers conclusions and directions of future work.

## 2 Environmental scenario

The base data model in our study is a vector-valued time-series

$$\mathbf{X} = \{\mathbf{x}(t_1), \ldots, \mathbf{x}(t_N)\},$$

$$\mathbf{x}(t_i) = (x_1(t_i), \ldots, x_M(t_i)),$$

where $N$ is the number of time samples, and $M$ is the number of observed parameters. It can be

[3] http://ct.gsfc.nasa.gov/lys/data/question1.html
[4] http://www.unidata.ucar.edu/software/netcdf/
[5] http://hdf.ncsa.uiuc.edu/
[6] http://www.unidata.ucar.edu
[7] http://msdn.microsoft.com/netframework/
[8] http://ws.apache.org/axis/
[9] http://www.globus.org/toolkit/
[10] http://www.omii.ac.uk/
[11] http://glite.web.cern.ch/glite/

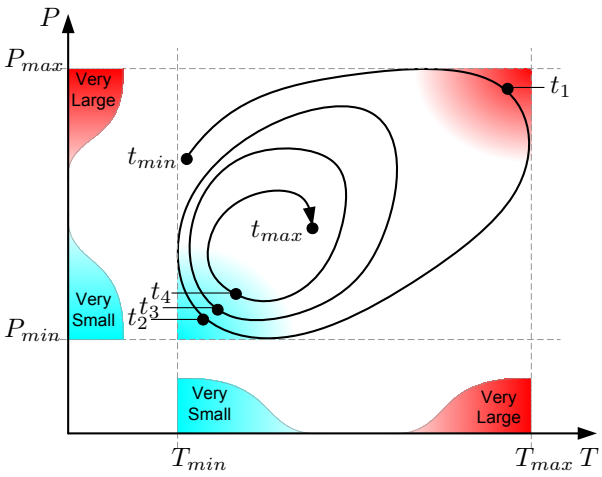


Figure 1: Time series as a trajectory in the two-dimensional phase space ($P$—pressure, $T$—temperature)

represented as a trajectory in the $M$-dimensional phase space $\mathbb{R}^M$. For example, in Fig. 1 we have a two-dimensional trajectory in the pressure–temperature ($P$–$T$) space.

A (fuzzy) state $S$ in a phase space $\mathbb{R}^M$ is a fuzzy set which can be described by fuzzy logic expression, composed of predicates describing in numerical or linguistic terms the parameter values in each of $M$ dimensions. For example, the state $S_1$ corresponding to the red (upper-right) region in Fig. 1 can be described by the fuzzy expression:

$$S_1 = (\text{VeryLarge}\, P)\, \text{and} (\text{VeryLarge}\, T),$$

where the linguistic term "VeryLarge" is a predicate, and the operator "and" stands for the fuzzy logic conjunction. In the same way, the state $S_2$ corresponding to the cyan (lower-left) region is

$$S_2 = (\text{VerySmall}\, P)\, \text{and} (\text{VerySmall}\, T),$$

Now, combining the descriptions of the states with the time shift operator $\text{shift}_{\text{d}t}$ to describe transitions between the states, we can write the following symbolic expression for the environmental scenario "very low temperature and pressure after very high temperature and pressure":

$$(\text{shift}_{\text{d}t=1}\, S_1)\, \text{and}\, S_2.$$

The only pair of observations in Fig. 1 which fit the above scenario is the pair $(t_1, t_2)$. Our environmental scenario search engine, ESSE, is designed to mine for the phase space transitions like that in very large scientific databases.

In the following subsections we describe in detail mathematics behind that definition of the fuzzy event scenario.

## 2.1 Fuzzy logic expressions

A classical set $A$ in a space of objects $\mathbb{U}$ can be defined by its indicator function $I_A(u) : \mathbb{U} \to \{0, 1\}$, which is equal to 1 for all elements $u$ from the set $A$ and to 0 otherwise. Figure 2 shows the plot of an indicator function of the segment $A = [5, 8]$ as a subset of all real numbers $\mathbb{R}$.

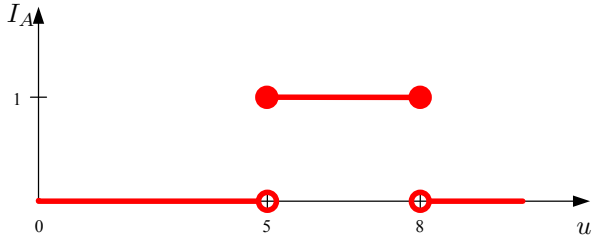


Figure 2: Indicator function $I_A(u)$ for the classical set $A = \{x | 5 \le x \le 8\}$

A fuzzy set expresses the degree to which an element belongs to a set. Hence the indicator function of a fuzzy set is allowed to have values between 0 and 1, which denotes the degree of membership of an element in a given set. A fuzzy set $A$ in $\mathbb{U}$ is defined by its membership function (or MF for a short) $\mu_A(u) : \mathbb{U} \to [0, 1]$, which maps each element of $\mathbb{U}$ to its membership grade between 0 and 1. Compare graphs of a MF for the fuzzy interval $[5, 8]$ and the indicator function for the classical segment $[5, 8]$ (Figures 3 and 2 respectively).

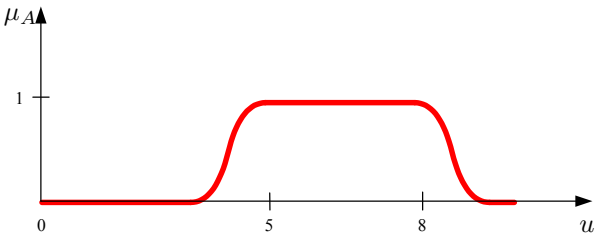


Figure 3: Fuzzy membership function $\mu_A(u)$ for the set $A = [5, 8]$

Basic operations of classical set theory (union, intersection, complement) and corresponding operations of mathematical logic (or, and, not) can be generalized for fuzzy sets and fuzzy logic in many different ways. The fuzzy generalization of intersection (logical "and") is usually called T-norm operator, generalization of union (logical "or") is called the T-conorm or S-norm operator, and the generalization of logical "not" is called fuzzy complement operator.

By definition [14], the T-norm operator is a function $T : [0,1] \times [0,1] \to [0,1]$ that satisfies the following properties: $T(0,0) = 0$, $T(1,a) = T(a,1) = a$ (boundary conditions); $T(a,b) \leq T(c,d)$, if $a \leq c$ and $b \leq d$ (monotonicity); $T(a,b) = T(b,a)$ (commutativity); $T(a,T(b,c)) = T(T(a,b),c)$ (associativity).

Any T-conorm operator $S : [0,1] \times [0,1] \to [0,1]$ has to satisfy the properties: $S(1,1) = 1$, $S(0,a) = S(a,0) = a$ (boundary conditions); $S(a,b) \leq S(c,d)$, if $a \leq c$ and $b \leq d$ (monotonicity); $S(a,b) = S(b,a)$ (commutativity); $S(a,S(b,c)) = S(S(a,b),c)$ (associativity).

The fuzzy complement operator $N : [0,1] \to [0,1]$ can be any continuous function, which meets the following axiomatic requirements: $N(0) = 1$ and $N(1) = 0$ (boundary conditions); $N(a) \geq N(b)$, if $a \leq b$ (monotonicity); $N(N(a)) = a$ (involution, optional).

One of the simplest generalizations from classical to fuzzy set theory for two MFs $\mu_A(u), \mu_B(u)$ is to use minimum of MFs for the intersection of fuzzy sets (fuzzy logic "and")

$$\mu_{A \cap B} = \min(\mu_A, \mu_B),$$

maximum of MFs for fuzzy sets union

$$\mu_{A \cup B} = \max(\mu_A, \mu_B),$$

and one complement for fuzzy set complement

$$\mu_{\overline{A}} = 1 - \mu_A.$$

In 1980 R. Yager introduced a parametric family of T-norms, T-conorms [15] and fuzzy complements [16]. Parameterized by $q > 0$ a family of fuzzy "and" aggregations for two MFs is defined by Yager's T-norm operator:

$$T_Y\left(\mu_A(x), \mu_B(x), q\right) = 1 - \min\left\{1, \left[(1-\mu_A(x))^q + (1-\mu_B(x))^q\right]^{\frac{1}{q}}\right\}.$$

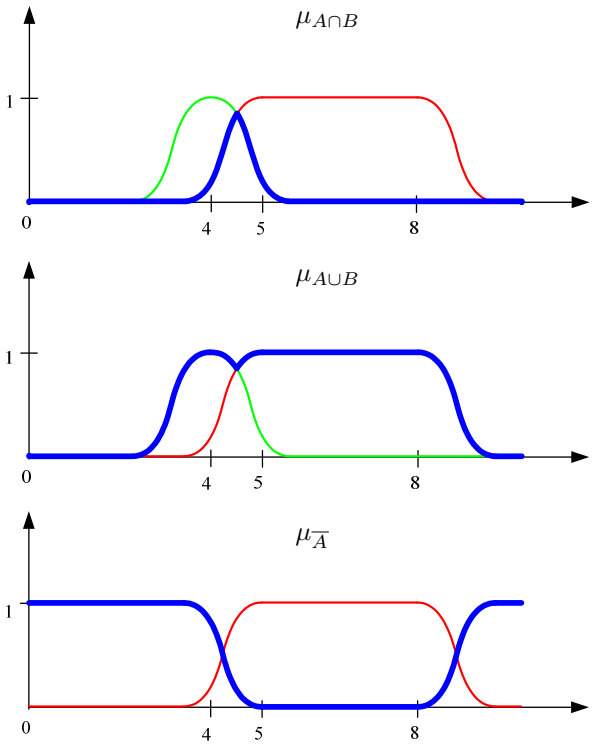


Figure 4: Fuzzy logic operations

A more general formula for the parametric Yager's T-norm operator for fuzzy "and" aggregation of any $M > 1$ MFs $\mu_m(x), m = 1 \dots M$ is

$$T_Y\left(\mu_m(x), q\right) = 1 - \min\left\{1, \left[\sum_{m=1}^{M}\left(1-\mu_m(x)\right)^q\right]^{\frac{1}{q}}\right\}.$$

The resulting surface of values for the multi-dimensional MF is more smooth than using a simple minimum of the aggregating MFs, which is the limiting case of Yager's T-norm for $q = 1$.

The parametric family of fuzzy "or" aggregations for two MFs is described by Yager's T-conorm operator

$$S_Y\left(\mu_A(x), \mu_B(x), q\right) = \min\left\{1, \left[\left(\mu_A(x)\right)^q + \left(\mu_B(x)\right)^q\right]^{\frac{1}{q}}\right\}.$$

A more general formula for Yager's "or" aggregation of any $M > 1$ MFs is

$$S_Y\left(\mu_m(x), q\right) = \min\left\{1, \left[\sum_{m=1}^{M}\left(\mu_m(x)\right)^q\right]^{\frac{1}{q}}\right\}.$$

Yager's fuzzy complements are defined by formula

$$N_Y\left(\mu(x), q\right) = \left(1 - \left(\mu(x)\right)^q\right)^{\frac{1}{q}}$$

The ESSE search engine is designed to support different libraries of T-norm, T-conorm and complement operators. The results below are obtained using the Yager's formulas with the order $q = 5$.

## 2.2 Fuzzy logic predicates

People often use qualitative notions to describe such variables as temperature, pressure, wind speed. In reality, it is difficult to put a single threshold between what is called "warm" and "hot". Fuzzy set theory serves as a translator from vague linguistic terms into strict mathematical objects.

The scenario editor from the ESSE user interface is used to formulate a set of conditions to be satisfied by the candidate events. The search conditions may be specified in a number of ways depending on the user's familiarity with the region/data of interest. An expert user can specify numeric thresholds

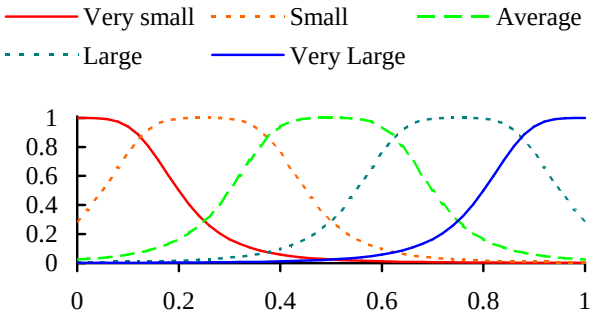


Figure 5: Membership functions of the ESSE "linguistic terms"

and/or limitations that must be maintained on certain parameters. Conditions can also be specified via abstract natural language definitions for each parameter. For instance, temperature limitations can be specified as "hot", "cold", or "typical". The default ESSE library of MFs formed for each variable (phase space dimension) uses the generic bell "mother" function [14]:

$$\mu_{gbell}\left(\tilde{x}; a, b, c\right) = \frac{1}{1 + \left|\frac{\tilde{x}-c}{a}\right|^{2b}}$$

Here, $\tilde{x}$ stands for normalized for range [0,1] scalar data variable, $c$ stands for centre of the symmetrical "bell", $a$ for its half-width, and $b/2a$ controls its slope. We use here simple range normalization for the variable $x$:

$$\tilde{x} = \frac{x - x_{min}}{x_{max} - x_{min}},$$

where $x_{min}$ and $x_{max}$ stand for the minimal and maximal observable values of $x$, respectively.

Five MFs for a linguistic term set {"very small", "small", "average", "large", "very large"} are plotted in Figure 5.

Center, slope, and half-width of the bell functions for these linguistic terms are listed in the Table 1.

On the next plot (Fig. 6) we present examples of four MFs from the ESSE numerical fuzzy term set {"less than", "about", "between", "greater than"}.

For the the normalized variable $\tilde{x}$ the center, slope, and half-width of the bell functions for numerical terms are listed in the Table 2.

Table 1: Parameters of MFs for linguistic terms

| Linguistic term | Center | Slope | Half-width |
|---|---|---|---|
| Very Small | 0 | 5 | 0.2 |
| Small | 0.25 | 5 | 0.2 |
| Average | 0.5 | 5 | 0.2 |
| Large | 0.75 | 5 | 0.2 |
| Very Large | 1 | 5 | 0.2 |

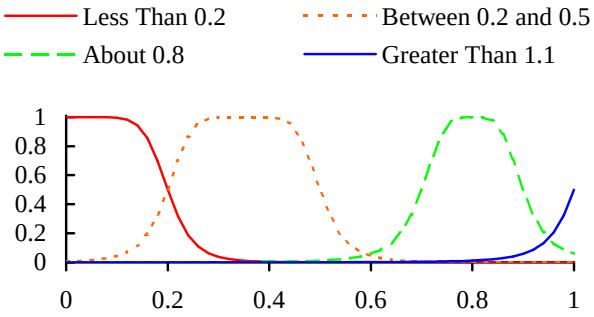


Figure 6: Fuzzy MFs for numerical terms

Table 2: Parameters of MFs for numeric terms

| Numerical term | Center | Slope | Half-width |
|---|---|---|---|
| Less Than $\tilde{v}, \tilde{v} < 0$ | $\tilde{v}$ | 10 | 0.1 |
| Less Than $\tilde{v}, \tilde{v} \geq 0$ | 0 | 10 | $\tilde{v}$ |
| About $\tilde{v}$ | $\tilde{v}$ | 10 | 0.1 |
| Between $\tilde{v}$ And $\tilde{w}$ | $(\tilde{v} + \tilde{w})/2$ | 10 | $\lvert\tilde{w} - \tilde{v}\rvert$ |
| Greater Than $\tilde{v}, \tilde{v} < 1$ | 1 | 10 | $\tilde{v}$ |
| Greater Than $\tilde{v}, \tilde{v} \geq 1$ | $\tilde{v}$ | 10 | 0.1 |

## 2.3 Fuzzy event scenario

In the ESSE applications we are searching for events in the environment where the parameters depend on time, as well as MFs – fuzzy logic predicates $\mu_m(\mathbf{x}(t))$ and fuzzy expressions $E_Y(\mu_m(\mathbf{x}(t)), q)$, which are composed of T-norms, T-conorms and complements over the predicates. We consider the values of the resulting time series $E_Y(\mu_m(\mathbf{x}(t)), q)$ as the "likeliness" of the environmental state to occur at the time moment $t$, or, in other terms, to visit a sub-region of the phase space described by the fuzzy expression (see Fig. 1). We search for the highest values of the $E_Y(\mu_m(\mathbf{x}(t)), q)$ and consider these to be the most likely candidates for the environmental event.

We use a simple climatology analysis to obtain normalization limits $x_{min}, x_{max}$ used in calculations of linguistic predicates like "very large" from Table 1. The limits are set to the minimum and maximum parameter values observed within the continuous or seasonal intervals given by the time constraints of the fuzzy search.

To be able to search for events like a "cold day" or a "cold week" we introduce the concept of event duration which may be any multiple $k$ of the time step $\Delta t$ of the input, $k\Delta t$. For example, the time step in the NCEP/NCAR reanalysis is $\Delta t = 6$ hours, so the minimum event duration is also 6 hours, but the event duration may also be 1 day ($4\Delta t$), 1 week ($28\Delta t$), etc. We do a moving average of the input parameters with the time window of the event duration before calculation of MFs in the fuzzy expression:

$$\overline{x}\,(t_i) = \frac{1}{k} \sum_{j=i}^{i+k-1} x\,(t_j), \quad t_i = t_0 + i\Delta t.$$

For example, when searching for a "cold day" in the NCEP/NCAR reanalysis, first we have to smooth the air temperature using a time window of 1 day ($k = 4$), then calculate the linguistic predicate "low" $\mu_{low}\left(\overline{T}\,(t_i)\right)$, sort the fuzzy scores in descending order, and finally take the several first times with the highest scores as the candidate events.

The important difference of the averaging operator is the dependence from the input of the neighbour observations in time. Another operator of

that class is the time shift, defined by the formula:

$$\text{shift}_k\, \mu\,(t_i) = \mu\,(t_{i-k})\,.$$

The difference between the averaging and the shift operators is that we average input values, but we shift in time the values of fuzzy membership function. Thus we have to investigate the properties of the time shift in relation to the fuzzy logical operators T-norm, T-conorm, and complement. For any fuzzy logic expression $E$ we have:

$$\text{shift}_k \,\text{shift}_l = \text{shift}_{k+l},$$

$$\text{shift}_k\, E\,(\mu_1, \mu_2, \dots) = E\,(\text{shift}_k\, \mu_1, \text{shift}_k\, \mu_2, \dots)\,.$$

We need the time shift operator to define multiple-state event scenario. For example, to find an abrupt air pressure drop, we can use a two-state scenario with fuzzy "and" of the "very large" and time-shifted "very small" predicates for pressure $P(t)$:

$$S(t) = T_Y\Big(\mu_{\text{VeryLarge}}\,(P(t))\,, \text{shift}_1\ \mu_{\text{VerySmall}}\,(P(t))\,, q\Big).$$

Following this example, a two-state scenario with the fuzzy expressions for states $E_1$ and $E_2$ with the time delay between the steps $k\Delta t$ can be defined as a Yager T-norm conjunction of the time-shifted expressions

$$S(t) = T_Y\,(E_1, \text{shift}_k\, E_2, q)\,.$$

Generalization of the formula for more than two states is straightforward.

To have the result of the fuzzy search in the form of a ranked list of the K-most likely dates (times) of the events, we sort the scenario MF $S(t)$ and select the times of several maximum values separated in time by more than the event duration $k\Delta t$.

### 2.4 Importance of the input parameters

The fuzzy search request may contain conditions which never or very rarely take place at the same time at the specified location, although they can be observed there separately at different time moments. For example, very high precipitation rate and very high air pressure are unlikely to occur simultaneously. The fuzzy search for such a combination of conditions may return an empty set of candidate dates and times. We decrease the probability of the empty fuzzy search results by introducing the concept of importance of the input parameters.

The importance $\alpha_n$ is a constant weight of a given parameter in the range between 0 and 1. More important parameters are given higher weight, with the condition that the highest priority is then normalized to one. Then instead of MFs $\mu_n(x(t_i))$ in the fuzzy expressions we use "optimistic" values $\max\,(\mu_n(x(t_i)), 1-\alpha_n)$. For parameters with the importance 1 we use the original MFs as before, and the parameters with the importance 0 are not used in the search at all.

## 3 Search engine implementation

At the core of the ESSE architecture is our fuzzy logic engine that accepts event definitions in the form of fuzzy expressions, reads the data from one or more time series streams generated by data resources and performs a search for and statistical analysis of the distribution of the identified events. Both the fuzzy logic engine and the data sources are implemented as web services. This allows parallel mining of several distributed data archives, possibly from different subject areas and enables third-party applications to feed their data to the engine and/or post-process the results of fuzzy search.

The ESSE system includes our own user interface implemented as a web application. In the web application it is possible:

- to discover data sources by keyword-based metadata search;
- to define the searching event as a combination of fuzzy conditions on a set of environmental parameters (e.g. "high temperature and low relative humidity") for data mining;
- to review the statistics of detected events;
- to visualize data related to the selected event;
- to download the event data in self-describing format (NetCDF or XML) to the user's workstation.

Table 3: ESSE data sources

| Data source | Sample parameters | Temporal coverage | Spatial coverage | Size, Gb |
|---|---|---|---|---|
| Meteo NCEP/ NCAR | Wind speed, temperature, cloud cover | 1948 – present | Global @2.5 Deg. | 250 |
| Space SPIDR | Kp index, sunspot number | 1933 – present | Global by observatory or satellite | 30 |

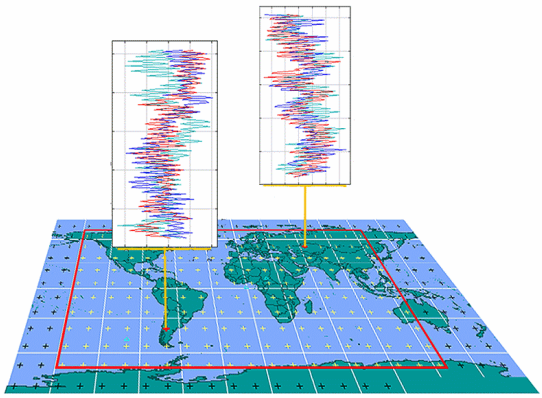

Figure 7: Data stored on the NCEP/NCAR cluster

Section 4 contains more detailed description of this web application capabilities.

## 3.1 Authoritative data sources

The real connection between the ESSE system and a given user community is a set of data sources that expose compatible web services interfaces. It is relatively easy to add a new data source to the ESSE through the web services interface, so the list in Table 3 should not be taken as limiting but rather as a starting point that demonstrates the ESSE functionality.

The first thing to notice is the relatively large size of the archives. Using the distributed database concept allows us to perform interactive mining on these substantial data sources. The second thing to notice is the long temporal ranges. The ESSE is most useful when the size of the archive prohibits or makes impractical searching by hand. As has already been mentioned, the NCEP/NCAR reanalysis data archive [3] was derived from numerical weather prediction model runs. It represents gridded output on a regular time step (6 hours) and fixed spacial grid step (2.5°). The model uses data ingest procedures to assimilate current observational data into model results to produce a consistent picture of the terrestrial weather since 1948.

To accelerate typical data requests for the ESSE search engine, we have developed a special parallel database cluster and optimized the database schema. The year field is used for data partitioning, so each cluster node stores several years of NCEP/NCAR reanalysis data. For example in 10-node cluster the first node stores data for years 1950, 1960, ...; the second node stores years 1951, 1961, ..., and so on. For each year we have a separate database, one table for one parameter, such as temperature or pressure. The table has a simple structure: latitude, longitude, height (optional), and a blob of data with floating point time series for one year at one latitude-longitude-height location (Fig. 7). The data records are indexed by the location.

The Space Physics Interactive Data Resource (SPIDR)[12] is an observational data source which includes the output of numerical models. The SPIDR system currently handles the following: Defense Meteorological Satellite Program (DMSP) visible, infrared and microwave browse imagery, ionospheric parameters, geomagnetic variations, geophysical and solar indices, GOES satellite x-ray, plasma, and magnetometer data, cosmic rays, and solar radio telescope data sets.

## 3.2 OGSA-DAI framework

The implementation of the environmental data access system that incorporates the ESSE engine is based on the OGSA-DAI framework [10, 8], the emerging standard for representing databases in Grids. OGSA-DAI is a middleware product which supports the representation of various data resources, such as relational or XML databases, on

[12] http://spidr.ngdc.noaa.gov/

to Internet and Grids. The basic abstraction introduced in OGSA-DAI is the notion of a data resource which is able to perform data access and data transformation activities. Typically each database is represented as a separate data resource, but the concept is general enough to represent heterogeneous databases as well. Data resources may differ in a set of activities they are able to perform. For example, a data resource representing a relational database may execute SQL queries while XPath queries may be submitted to a data resource representing an XML database. The advantage of OGSA-DAI is that clients use standard Web services/Grid service protocols to submit queries and obtain results. Additionally data resources may be orchestrated in such a way that result set from one of it goes as an input data directly to another data resource.

For each data resource OGSA-DAI exposes a web service endpoint, to which Web service messages can be addressed. The message with a "perform" operation contains XML perform document as its parameter and returns XML response document as a result. The perform document describes one or more activities. Data resource performs different kinds of query, transformation, delivery or manipulation operations depending on the activities types and parameters. Each activity may have input and output channels that may be linked with each other within one perform document. Thus OGSA-DAI perform document describes a simple workflow to be performed by the data resource. Special types of delivery activities enable linking channels on remote OGSA-DAI servers.

Along with the web service interface OGSA-DAI offers an easily extensible object-oriented programming framework. Each data resource and each activity is represented by a single object with a simple interface. The OGSA-DAI middleware engine dispatches processing over these objects as described by the perform document and outputs the result of this processing to the response document.

## 3.3 Data source API

The interface to data archives created for geophysics and other environmental sciences can not easily and efficiently utilise standard query languages like SQL or XQuery due to the fact that these languages do not directly support multidimensional array data type. Thus we had to create a separate OGSA-DAI data resource object and a set of corresponding activity objects (Table 4). This API to the virtual environmental data sources has a higher-level query language compared to the array subsetting and hyper-slabbing implemented in the OpenDAP protocol[13].

Table 4: Data resource and activity components added to the OGSA-DAI framework by the ESSE system

| Component | Description |
|---|---|
| `EsseDataResource` | Represents environmental database |
| `GetMetadataActivity` | Query activity. Returns the description of the data maintained by the `EsseDataResource`. |
| `GetXmlDataActivity` | Query activity. Returns one or several time series from the `EsseDataResource`. |
| `GetNetCdfActivity` | Query activity. Serializes a data subset into a NetCDF file and returns a URL to that file. |
| `FuzzySearchActivity` | Transformation activity. Receives one or more time series from `GetXmlData` and returns fuzzy membership function values. |

First three activities mentioned above expose a specialized environmental database query capabilities. In a real system the portal application that has the user interface creates perform document, invokes it to remote OGSA-DAI server and displays the result to a user.

The data from the OGSA-DAI service may come to a user in different formats. We've implemented the binary NetCDF format[4] commonly used in environmental sciences and more general purpose XML format. Unlike business environments, in en-

[13] http://www.opendap.org/

vironmental sciences domain the XML format is not yet recognized as a standard for data transfer. This is due to generally observed higher file sizes and larger processing times for XML data compared to data in binary format. File sizes are even more important in distributed systems where large transfers may easily saturate the network. The solution adopted in the present paper is in the use of compression algorithms for XML data transfer. The table below compares the amount of data transferred from the server to the client for the same query. We compare the NetCDF file with data, serialized to temporary directory, and XML document sent over the OGSA-DAI output data channel and saved to a file at client side. In both cases the client requests one month of data for one parameter with 1 min time step (Table 5).

Table 5: Data load for binary and XML data serialization

| Activity | Description | Load, Kb |
|---|---|---|
| `getNetCdfData` | Binary NetCDF file | 924.5 |
| `getXmlData` | Response document containing data in XML format | 1,771.1 |
| `getXmlData`+ `gzipCompression` | Response document containing base64 encoded and GZIP compressed XML data | 123.5 |

The ESSE engine is wrapped with the `FuzzySearchActivity`, the data transformation activity which is not linked to a specific type of data resource. This makes the whole data mining system extremely flexible. One can search an environmental scenario over several parameters stored in a local database. This is accomplished by combining several query activities with the fuzzySearch activity in a single workflow within a single perform document (Fig. 8).

In a more advanced scenario it is possible to combine data search from several OGSA-DAI resources. Figure 9 illustrates the situation when part of the scenario is queried and evaluated on a remote server, the resulting fuzzy membership function is transferred using OGSA-DAI data transport operations where it is combined with the rest of the data and produces the final mining results.

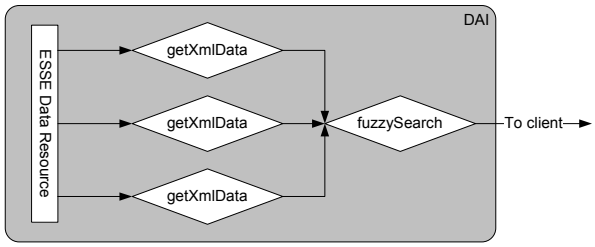


Figure 8: OGSA-DAI activities on a single server

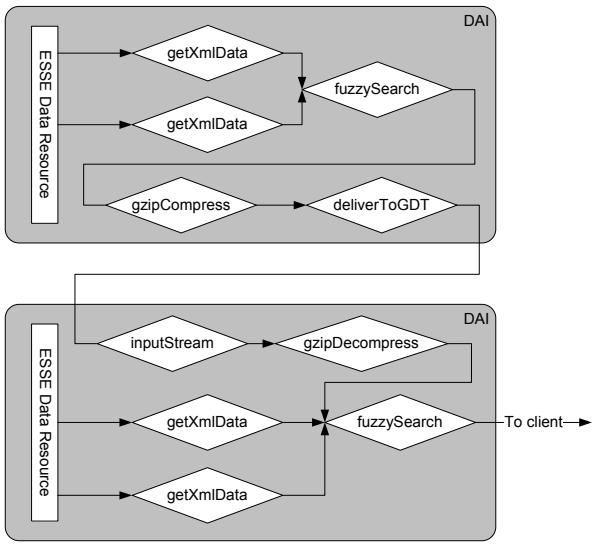


Figure 9: Distributed OGSA-DAI queries

## 3.4 Cross-platform portability

The original OGSA-DAI framework is written in Java and has open source code. There are flavours of the product providing slightly different external interfaces, the WS-I compatible interface when run in Jakarta Tomcat environment and WSRF interface when deployed in Globus 4.0 container.

In order to enable true platform independence the team has created OGSA-DAI.NET, a lightweight OGSA-DAI compatible component using Microsoft ASP.NET web services infrastruc-

ture. OGSA-DAI.NET and OGSA-DAI have the same external and internal interfaces. The same portal user interface can consume either of implementations. The same Java/J# code for ESSE specific activities run on the Open Source and on the Microsoft platform.

The Microsoft .NET Framework includes a comprehensive set of classes that supersedes many of the commonly used open source libraries. This is especially true for XML processing libraries and enables easy creation of wrapper classes ensuring the portability of the source code created in this project.

# 4 Data mining portal

A web-portal serves as an agent between the user and the ESSE framework. It performs two main functions. The first function is metadata management, which allows for fast and efficient collection-level metadata search. Here by metadata we mean general descriptions of data resources, stored as a managed set of XML documents (owner info, geographic coverage, time coverage, data description, etc.). The web-portal called the Integrated Distributed Environmental Archive System (IDEAS) allows users to register new data resources by adding their own metadata. The metadata is being constantly updated both manually and automatically (see Fig. 10). Our metadata collection works much the same as other similar resources, like GCMD[1] or MEL[2]. For more detailed discussion of the role of metadata in distributed data networks see [9]. The second function of the web-portal is data access. In Fig. 10 the IDEAS web-portal is shown as a client, which connects to numerous data sources, retrieves the requested data, and delivers it back to the user.

The typical workflow of the IDEAS web-portal is shown in Fig. 11. This figure shows the two components: the IDEAS Portal and the ESSE Grid framework. During the workflow the IDEAS portal dynamically generates requests to ESSE Grid components. Some of the metadata, like the list of available parameters, is also retrieved from ESSE data sources.

All user operations on the IDEAS web-portal are accomplished via interactive web forms. The interface is flexible and easy to use.

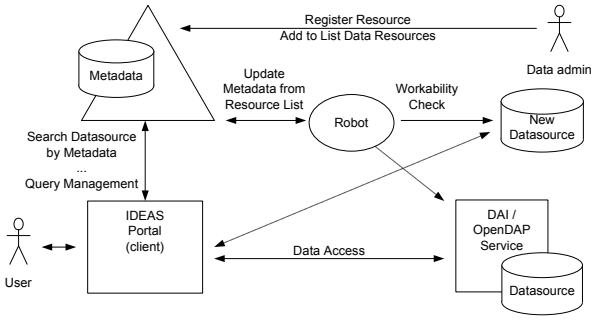


Figure 10: IDEAS web-portal as a client for ESSE services

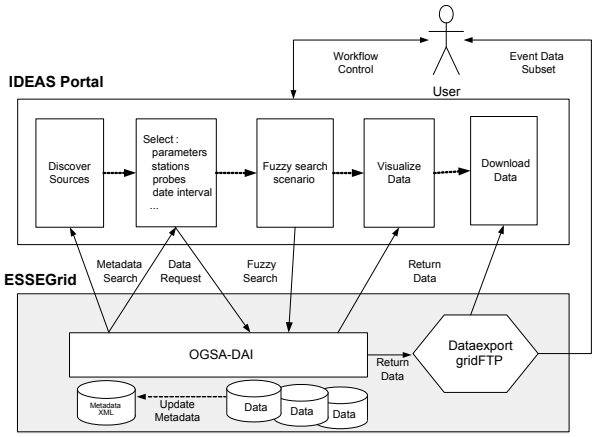


Figure 11: IDEAS Web-portal workflow

The next subsection provides an example of a typical ESSE workflow, performed via the web-portal.

## 4.1 Use case

In the following example we will search for a E-W atmospheric front near Moscow described by three parameters “air pressure”, “E-W wind speed” (U-wind) and “N-S wind speed” (V-wind) with subsequent fuzzy states:

1 : (Small pressure) and(Large V-wind-speed)
2 : (Large pressure) and(Small U-wind speed) and(Small V-wind-speed).

The typical data mining protal use case involves 3 actors, namely a user, the web application, and the fuzzy search engine web service, and consists of the following steps:

1. The user logs in to the IDEAS portal and receives a list of the currently available (distributed) data sources. For each data source the list has abridged metadata like name, short description, spatial and temporal coverage, parameters list and link to full metadata description.

2. The user selects environmental data source based on the short description or by metadata keyword search (e.g. NCEP/NCAR Reanalysis). The portal stores the data source selection on the server side in the persistent “data basket” and presents a GIS map with the spatial coverage of the data source (Fig. 12).

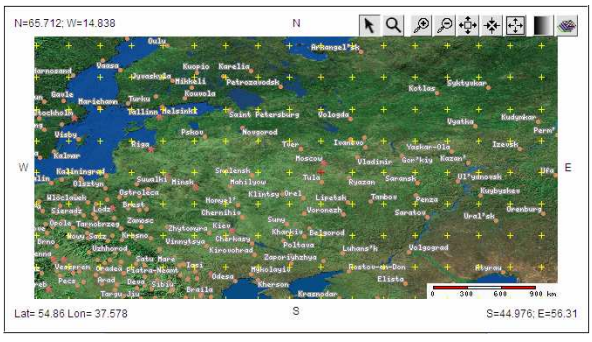


Figure 12: Selecting probe near Moscow using GIS map

3. The user selects a set of “probes” (representing spatial locations of interest, e.g. Moscow) for the the searching event. IDEAS stores the selected set of ”probes” and presents a list of all the environmental parameters available from the selected data source and a fuzzy constraints editor on the parameters values which represent the event (Fig. 13).

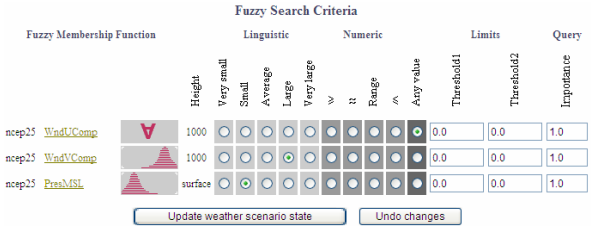


Figure 13: ESSE fuzzy state editor

4. The user selects some of the environmental parameters and sets the fuzzy constraints on them for the searching event (e.g. low pressure, high V-wind speed).

5. Multiple subsequent environment states can be grouped to form the actual environmental scenario. For example, we need to define the two different states mentioned above. Adding and removing fuzzy states is done via a Web-form shown in Fig. 14.

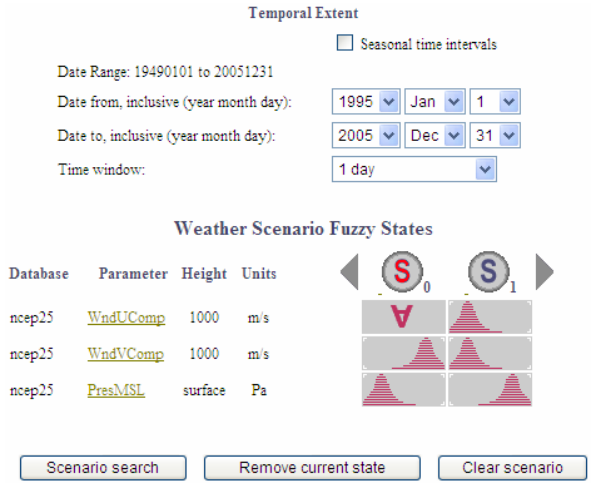


Figure 14: Environmental scenario states form

6. ESSE stores the searching environment states and sends them to the fuzzy search web-service in the XML format.

7. The fuzzy search web-service collects data from the data source for the selected parameters and time interval, performs the data mining, and returns to the IDEAS web application a ranked list of candidate events with links to the event visualization and data export pages (Fig. 15).

**Weather Scenario Search Results**

| Rank | Score | Date | Time | Probes | ROI | Satellite |
|---|---|---|---|---|---|---|
| 1 | 0.18 | 2005-10-26 | 18:00:00 | Plot | Vis5d | DMSP |
| 2 | 0.13 | 1996-09-16 | 0:00:00 | Plot | Vis5d | DMSP |
| 3 | 0.13 | 2003-03-12 | 18:00:00 | Plot | Vis5d | DMSP |
| 4 | 0.04 | 1995-03-29 | 0:00:00 | Plot | Vis5d | DMSP |
| 5 | 0.02 | 2003-04-05 | 6:00:00 | Plot | Vis5d | DMSP |
| 6 | 0.02 | 2002-04-01 | 6:00:00 | Plot | Vis5d | DMSP |
| 7 | 0.02 | 1998-02-06 | 0:00:00 | Plot | Vis5d | DMSP |
| 8 | 0.01 | 1995-01-03 | 0:00:00 | Plot | Vis5d | DMSP |
| 9 | 0.01 | 1996-10-22 | 6:00:00 | Plot | Vis5d | DMSP |
| 10 | 0.01 | 1999-01-29 | 0:00:00 | Plot | Vis5d | DMSP |

Figure 15: List of event candidates

8. The user visualizes interesting events and requests the event-related subset of the data for download from the data source in the preferred scientific format (XML, NetCDF, CSV table). Currently there are three visualization types available: time series (Fig. 16), animated volume rendering using Vis5D (Fig. 17), and DMSP satellite images (Fig. 18).

In Figure 16 rectangles show the two environmental states within the found event on October 26, 2005: (low pressure - high wind) and (high pressure - low wind).

Vis5D[14] is a system for interactive visualization of large 5-D gridded data sets such as those produced by numerical weather models. One can make isosurfaces, contour line slices, colored slices, volume renderings, etc of data in a 3-D grid, then rotate and animate the images in real time.

In the associated day-time DMSP satellite images in IR and visible bands (Fig. 18) we can clearly see the E-W front passing above Moscow. There are two types of DMSP images available: visible and infrared images. Images are shown together with the

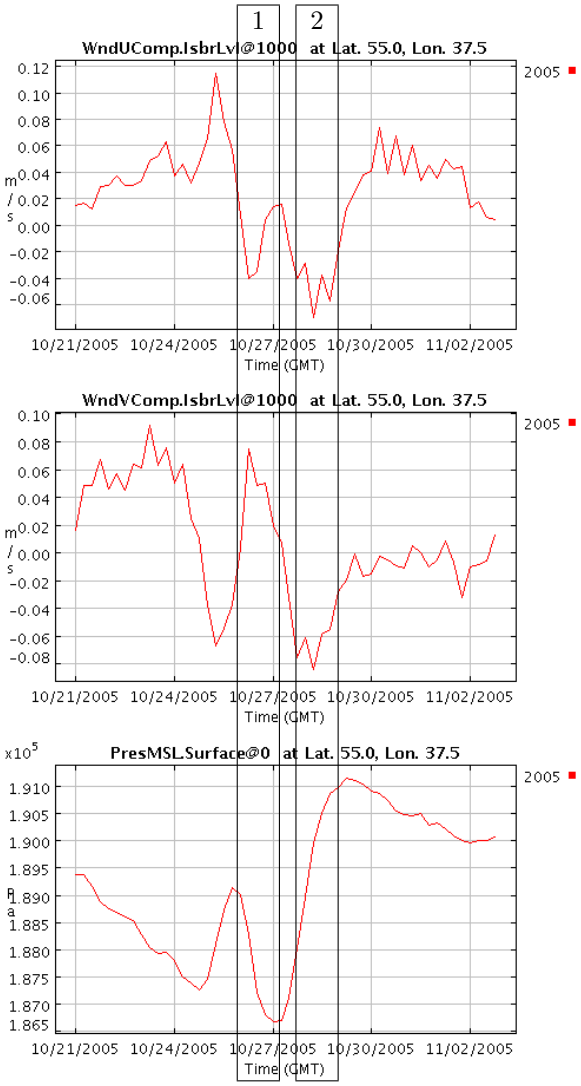


Figure 16: Air temperature (top) and pressure (bottom) in Moscow for the event of June 13–14, 2002

actual satellite orbit and visibility sector. Satellite images along with other kinds of visualization may serve as an additional means of verification for query results.

The use case variations include data mining for the environmental events, described by parameters from multiple data sources, which may consist of multiple states (e.g., extremely cold day followed by magnetic storm).

[14] http://www.ssec.wisc.edu/ billh/vis5d.html

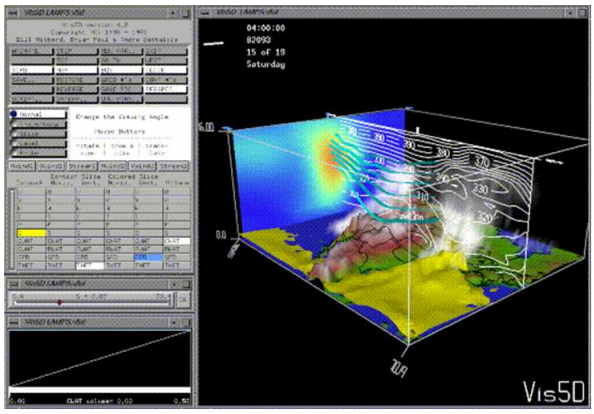


Figure 17: Animated volume rendering using Vis5D

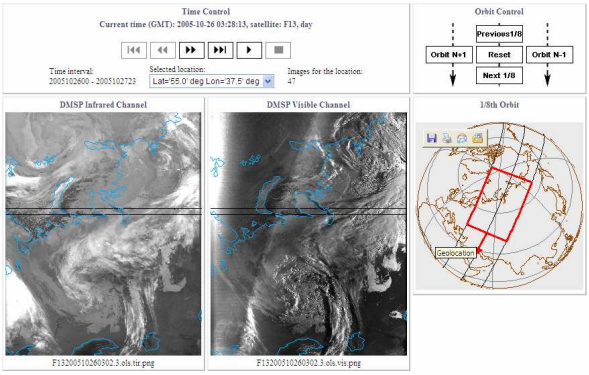


Figure 18: DMSP weather images

# 5 Conclusions and future work

As more and more data archives become available through projects like Earth System Grid[15] of DOD, Comprehensive Large Array-data Stewardship System[16] of NOAA, Earth Observing System Data and Information System[17] of NASA and other network accessible data systems, the tools to extract information from them become more important. ESSE can help users sift through the vast quantities of data available online and point at the interesting bits. This means that even with the volume of data increasing so rapidly and the number of researchers remaining relatively level we can hope to extract the most valuable information from the observations and carry that back to the relevant scientific communities.

The application of fuzzy logic based data tools goes far beyond simple event selection. For example an ever present issue when dealing with these large data sets is quality control. There is simply too large a volume to reasonably screen by hand. Using techniques such as peer-matching and expert systems we can extend the ESSE to monitor data and alert data managers to changes and anomalies. As the computational power available expands we can extend the system into areas such as data classification whereby we can identify modes of the environment and perhaps identify new unknown relations in specific regions.

Finally the emergence of a network infrastructure for data access is providing new opportunities for the scientific researcher. It is now fairly trivial to reach out across discipline boundaries and access data in an immediately useable format. This is true for example in the case of the terrestrial weather community being able to make use of the space data made available through SPIDR. With these opportunities come challenges. As researchers expand into domains in which they may not be expert they will come to rely on intelligent tools to support them.

The mission of the ESSE is fundamentally to help a user distil the vast amount of available data down to a manageable amount of information. The in-

[15] http://www.earthsystemgrid.org/
[16] http://www.class.noaa.gov
[17] http://nasadaacs.eos.nasa.gov/

creasing data volumes available in the future demand different techniques to handle it and the ESSE framework is one exceptional method for a user to handle it.

# 6 Acknowledgements

Our thanks to Dmitry Kokovin for the help with illustrations in the report and design of the ESSE web-portal, including the Fuzzy Scenario Editor. This work was supported by the Russian Foundation for Fundamental Research grant 04-07-90362.

# References

[1] L. M. Hilty, B. Page, F. J. Radermacher, and W.-F. Riekert. Environmental informatics as a new discipline of applied computer science. In N. M. Avouris and B. Page, editors, *Environmental Informatics - Methodology and Applications of Environmental Information Processing*, pages 1–11. Kluwer Academic Publishers, 1995.

[2] Alexander Szalay and Jim Gray. 2020 computing: Science in an exponential world. *Nature*, 440(7083):413–414, March 2006.

[3] E Kalnay, M Kanamitsu, R Kistler, W Collins, D Deaven, L Gandin, M Iredell, S Saha, G White, J Woollen, Y Zhu, M Chelliah, W Ebisuzaki, W Higgins, J Janowiak, KC Mo, C Ropelewski, J Wang, A Leetmaa, R Reynolds, R Jenne, and D Joseph. The NCEP/NCAR 40-year reanalysis project. *Bull. Am. Meteorol. Soc.*, 77(3):437–471, 1996.

[4] S.M. Uppala, P.W. Kallberg, A.J. Simmons, U. Andrae, V. da Costa Bechtold, M. Fiorino, J.K. Gibson, J. Haseler, A. Hernandez, G.A. Kelly, X. Li, K. Onogi, S. Saarinen, N. Sokka, R.P. Allan, E. Andersson, K. Arpe, M.A. Balmaseda, A.C.M. Beljaars, L. van de Berg, J. Bidlot, N. Bormann, S. Caires, F. Chevallier, A. Dethof, M. Dragosavac, M. Fisher, M. Fuentes, S. Hagemann, E. Holm, B.J. Hoskins, L. Isaksen, P.A.E.M. Janssen, R. Jenne, A.P. McNally, J-F. Mahfouf, J-J. Morcrette, N.A. Rayner, R.W. Saunders, P. Simon, A. Sterl, K.E. Trenberth, A. Untch, D. Vasiljevic, P. Viterbo, and J. Woollen. The ERA-40 re-analysis. *Quart. J. Roy. Meteor. Soc.*, 131:2961–3012, 2005.

[5] Lotfi Zadeh. Fuzzy sets. *Information and control*, 8:338–353, 1965.

[6] E. F. Codd. A relational model of data for large shared data banks. *Communications of the ACM*, 13:377387, 1970.

[7] F. Petry. *Fuzzy Databases, Principles and Applications*. Kluwer Academic Publishers, 1996.

[8] Konstantinos Karasavvas, Mario Antonioletti, Malcolm P. Atkinson, Neil P. Chue Hong, Tom Sugden, Alastair C. Hume, Mike Jackson, Amrey Krause, and Charaka Palansuriya. Introduction to OGSA-DAI services. In Pilar Herrero, María S. Pérez, and Víctor Robles, editors, *SAG*, volume 3458 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2004.

[9] María A. Nieto-Santisteban, Jim Gray, Alexander S. Szalay, James Annis, Aniruddha R. Thakar, and William O'Mullane. When database systems meet the grid. In *CIDR*, pages 154–161, 2005.

[10] N. Paton, M. Atkinson, V. Dialani, D. Pearson, T. Storey, and P. Watson. Databases access and integration services on the grid. UK e-Science Programme Technical Report UKeS-2002-03, National e-Science Centre, 2002. http://www.nesc.ac.uk/technical_papers/dbtf.pdf.

[11] Mario Antonioletti, Malcolm P. Atkinson, Rob Baxter, Andrew Borley, Neil P. Chue Hong, Brian Collins, Neil Hardman, Alastair C. Hume, Alan Knox, Mike Jackson, Amrey Krause, Simon Laws, James Magowan, Norman W. Paton, Dave Pearson, Tom Sugden, Paul Watson, and Martin Westhead. The design and implementation of grid database services in OGSA-DAI. *Concurrency - Practice and Experience*, 17(2-4):357–376, 2005.

[12] World Wide Web consortium. XSL transformations (XSLT). W3C recommendation REC-xslt-19991116, W3C, 1999. http://www.w3.org/TR/xslt.

[13] P. Deutsch. GZIP file format specification version 4.3. Request for Comments 1952, IETF, May 1996. http://www.ietf.org/rfc/rfc1952.txt.

[14] J.-S. R. Jang, C.-T. Sun, and E. Mizutani. *Neuro-Fuzzy and Soft Computing*. Prentice Hall, 1997.

[15] R. Yager. On a general class of fuzzy connectives. *Fuzzy Sets and Systems*, 4:235–242, 1980.

[16] R. Yager. On the measure of fuzziness and negation, part I: membership in the unit interval. *International Journal of Man-Machine Studies*, 5:221–229, 1979.

# 7 Appendix

Table 6: Properties of the classical and fuzzy logic operators

| | classical set theory | min-max fuzzy logic | Yager + $N(a) = 1 - a$ |
|---|---|---|---|
| $A \cap \neg A = \emptyset$ | yes | no | no |
| $A \cup \neg A = X$(universe) | yes | no | no |
| $A \cap A = A$, $A \cup A = A$ | yes | yes | no |
| $\neg\neg A = A$ | yes | yes | yes |
| $A \cap B = B \cap A$, $A \cup B = B \cup A$ | yes | yes | yes |
| $(A \cap B) \cap C = A \cap (B \cap C)$, $(A \cup B) \cup C = A \cup (B \cup C)$ | yes | yes | yes |
| $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$, $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$ | yes | yes | no |
| $A \cup (A \cap B) = A$, $A \cap (A \cup B) = A$ | yes | yes | no |
| $A \cup (A \cap B) = A \cup B$, $A \cap (A \cup B) = A \cap B$ | yes | no | no |
| $(A \cup B) = A \cap B$, $(A \cap B) = A \cup B$ | yes | yes | yes |