



POESIA: *Public Open-source Environment for a Safer Internet Access*

POESIA Annual Report
Deliverable 1.3

Project name:	POESIA
Project number:	IAP 2117/27572
Date:	January 13, 2003
Document Id:	POESIA-WP1-1.3
Version:	1.0 (revision 1.13)
Deliverable:	1.3
Related WP:	1 - Project Management and Coordination
Authors:	All consortium partners
Status:	Public

Project Overview

The Safer Internet Action Plan

The Safer Internet Action Plan is an European Commission initiative motivated by the protection of European youth from inappropriate or harmful aspects of the Internet. It supports a safer Internet, through three main action lines. The POESIA project fits inside the second line: development of filtering and rating systems. The name of the project stands for **P**ublic **O**pen-source **E**nvironment for a **L**arge **S**afer **I**nternet **A**ccess. The project started on February 4th 2002 and lasts for two years.

Project goals

The overall goal of the POESIA project is to develop, test, evaluate and promote a fully open-source, extensible, state of the art, filtering and caching software solution, in order to achieve more effective filtering than is provided by existing products. The POESIA system will filter several channels (e.g. Web and Email) by combining highly innovative technologies. A key feature of the project is its aim of providing advanced content-based filtering, both on the basis of textual and image content. Filtering will thus cover a range of modes, including image filtering, natural language text filtering (for English, Italian, Spanish and – at a later stage – for French), URL, PICS and JavaScript filtering.

POESIA as an open-source project

A unique feature of the project, which is worth stressing here, is that POESIA is developed as an open-source (see <http://www.opensource.org/>) or free¹ filtering software system. This entails that every piece of software developed under the POESIA project is released under a free software licence (usually the GPL or LGPL, see <http://www.gnu.org/copyleft/>), in source code form. This source code is viewable, reusable and improvable by peer developers, and should be compiled into executables to produce working filtering software. Free availability of the POESIA system as an open-source software will provide numerous market opportunities to commercial corporations (both large and SME), in particular in network configuration and administration businesses, educational businesses, etc. It will also permit a wide deployment of filtering systems in many educational institutions.

Target Audience

The main targets of POESIA are educational institutions (Internet classrooms in primary or secondary schools, colleges, universities) and other collectivities (e.g. libraries, cybercafés, museums, corporate networks) in need of Internet content filtering, mostly with young users.

¹ We use both terms interchangeably -even if they are significant philosophical differences-, but we are more in the free (or libre) software side, since POESIA uses mostly the GPL or LGPL free software licenses.

The POESIA scenario

The POESIA system is designed to run on a GNU/Linux (free software) system, usually on a PC (unattended) box² or a semi-dedicated (e.g. the teacher's) Linux workstation. This GNU/Linux system should have two network interfaces: one for the connection to the outside Internet and another for the internal (classroom) local network. So every bit of information has to flow through the POESIA box before it hits a browser on the internal local network (e.g. the classroom): POESIA filtering is unavoidable, and does not require any specific browser (or configuration) on the classroom browsing stations (which can be of any kind and run any software compliant with Web standards). The POESIA project is developing filtering software which can be integrated with existing open-source software to provide a complete content filtering solution.

POESIA on the Web

The POESIA project has a dedicated web site on <http://www.poesia-filter.org/> where every public document and software is available.

Summary of first year activities

During the first year (February 4th 2002 - February 3rd 2003), the POESIA consortium (10 members) worked on two main directions: the “users” group defined detailed end-users requirements and provided test case examples, while the “developers” group defined a software architecture and started its implementation. Both groups actively and continuously interacted.

From End User Requirements to POESIA Recommendations

The End User team of the POESIA project surveyed current and on-going activities in the area of Internet Safety filtering, explored End User opinions and established needs and recommendations for the POESIA technical development team. End Users (decision-makers and administrators) were surveyed between June and July 2002, mainly from the three participating countries of the End User group (Italy, Spain and the UK). A wider audience has been sought through an on-line survey and questionnaires distributed at European conferences and workshops. These surveys have been continued in order to expand the sample size and their updated findings have been produced as academic presentations and conference papers.

The general requirements, proposed by the end users team for the design of the POESIA filtering system, were carefully evaluated by the developer partners to assess their feasibility and actual implementation in the project lifetime. This evaluation was done by taking into account a number of different factors: firstly, there are the resource limitations, in terms of time and labour that exist for all projects including POESIA, whose duration is only two years. Secondly, there are fundamental limitations to what we could expect to achieve using current techniques. This issue of what is fundamentally possible using current techniques interacts with a third factor, which is the need to produce a system whose processing speed characteristics make the system feasible for use in practical internet filtering.

For the reasons listed above, it was agreed to focus the development effort for content-based filtering towards only certain domains and channels. In regard to channels, the focus is on web

² The box running POESIA could be a barebone or rackable or brick model, without any keyboard, mouse or screen.

pages and – in a more restricted way – on email. Concerning the latter, the filtering task is restricted to the specific aim of blocking porn-oriented SPAM messages, i.e. SPAM messages advertising pornographic web sites.

Within the web pages channel, which is the main focus of attention for content-based filtering, three different content categories are addressed. Firstly, there is the domain of pornographic web pages, which is the main focus of attention for both Natural Language Processing (NLP) and Image based filtering. Secondly, gross language content is filtered, the task being handled by the NLP-based components. Thirdly, the domain of violent content is addressed, but in a restricted fashion, i.e. by the use of image processing techniques to identify symbols associated with hate groups. The more general problem of violent content in text may be addressed later on in the project, depending of the rate of progress made with the problem of handling sexual content in text. Obviously illegal content (such as paedophilic pornography) is outside the realm of POESIA (because gathering illegal examples and testing with them requires sensitive specific authorizations from law officials that the consortium do not acquire), even if the technology developed within the project might catch part of it.

It should be noted that, although some combinations of domain and channels will not be addressed by content-based filtering, alternative filtering methods such as channel blocking and URL-based filtering, will still be available to prevent access to inappropriate material.

During the first year, the end user team also collected a significant amount of example URLs, manually classified according to their content (see D2.2 “Test Case Files”), which will be used to test, train and debug the filtering software and produced in consultation with all partners a wide-ranging dissemination plan for the project activities and outcomes (detailed in D9.5 “Dissemination Report and WWW Server”). Web site design and dissemination has also been addressed, the promotional POESIA Web site has been planned, designed and first version developed and an End User reporting site has been established (<http://www.hope.ac.uk/ebs/poesia/>). The End User team has developed a dissemination toolkit for partners to use, this includes leaflets and Powerpoint template. A number of conference presentations and publications have been produced and local, regional, national and international dissemination has been undertaken. The End User team have also begun to address usability issues and the testing and evaluation process which will take place in the second year of the project. Some of these aspects are now reflected in the latest End User report (D2.1).

Software architecture and development

The Developers team worked, with constant collaboration with the end-users team, to define and start prototyping an extensible software architecture for POESIA. The developers worked not only with the manually classified database of URLs collected by end-users (see above) but also by scanning the Web and by building appropriate training corpora of examples (namely, D6.1 “Image Subsets” and D7.1 “Domain Corpora for Each Language”).

Overall architecture of the system

The POESIA filtering system is organized around a central monitor, which deals with external information sources (e.g. web, email) and pre-processes and distributes the information to be filtered to the specialized filters – namely the language detector, the Natural Language Processing filters, the Image filter and the URL, PICS and JavaScript filter - and to the decision-making components. The overall architecture of the system is sketched in Figure 1 below:

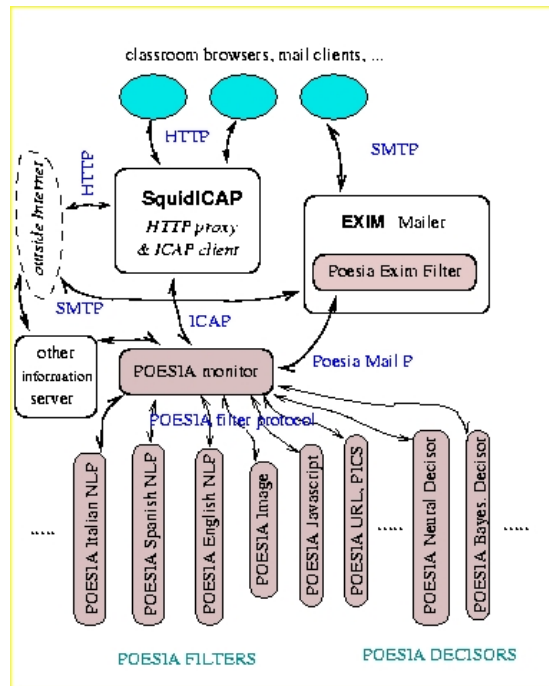


Figure 1: POESIA overall architecture

The POESIA monitor is the sole program communicating (directly) with filtering clients. Filters and decision-making components communicate directly only with the POESIA monitor. The POESIA monitor starts, schedules and communicates with POESIA specific filters (e.g. image or NLP filters) and sub-processes (like the decision mechanism). Filters can not only produce scores (which are handled by decisors) but can also ask (thru a continuation message) for another filter to process their intermediate result. For instance, in this way, a language detector filter can pass textual content to a language specific (Italian, Spanish, English) filter. Final scores are cached and stored by the monitor, which also deals with positive and negative lists of URLs. This mutualizes the filtering cost in the common case where the same content is fetched from several browsing stations. The filtering system can be configured to provide different levels of filtering at different times of day, and/or for different browsing clients, etc.

The architecture is explicitly designed with configurability and extensibility in mind: the same POESIA monitor should be usable to filter content types not yet filtered in this project, by addition of internal filters and reconfiguration. More details on the POESIA architecture can be found in the publicly available “Software Architecture Definition” report D3.1.

A first version of the POESIA filtering system was developed during the first year of the project (D4.1 “Alpha Release of POESIA Software”).

Image filtering

The POESIA system provides image filtering, in particular for filtering of Web sites. Only fixed pixelised images (GIF, JPEG, PNG, etc ...) are filtered within POESIA (so MPEG movies, Macroflash images, and vector SVG drawings are out of scope). Since images are embedded in HTML, image filtering requires pre-fetching of images contained inside an HTML page (before the browser requires them). Image filtering is focused towards two independent goals:

1. detection of pornographic images;
2. detection of inappropriate symbols (like nazi crosses or some sectorian cult symbols) against a configurable set of model symbols.

The basic pornographic filter starts with skin detection. Several algorithms were implemented for skin detection that perform uniformly better than the widespread colour model referred here as the

Baseline model. The pornographic image database collected by POESIA End-users was used to test the skin detection algorithms. The skin detector outputs for each input image a score, which is the normalized average skin probability of all the pixels in that given image. The normalized score is an integer in [0,100]. Two sets of images were used to test the skin detector: the first set, referred to as "porn + nude", consists of 1,654 pornographic images and nude images; the second one, "the rest", contains 3,739 other sorts of images. The first set contains images from the end users team of the project and from Compaq, whereas the other set is entirely from Compaq. The skin detector was run on these two sets of images to get the respective score distributions. The score distributions of these two categories were observed to be very distinct, which means that the score can be used to classify images as we expected.

Skin detection permits a large number of images that do not contain skin to be filtered. However, non-pornographic images as portraits or group portraits may also be filtered out. Hence POESIA proposes to use shape information from the skin image as well as from the original image in order to detect human anatomy. More complex pornographic detection carried out along the lines outlines above will be tackled during the second year of the project.

Symbol detection has been partly investigated during this first year. The goal is to compare a symbol entirely filling an image³ against a small fixed set of model symbols. This can be not only useful for content filtering of images (e.g. detecting every occurrence of a flag or logo) but - being available in open-source form - could also be reused in other settings (e.g. detection of navigational arrows). The guiding principle is to extract from a symbolic image a vector of descriptors (a tuple of real numbers) which is invariant for usual image operators like translation, scaling, limited rotation, and then to compare the descriptive vector against those of the model symbols.

From the image symbols collected by POESIA End Users, a catalogue of almost two hundred images was created and subdivided into two subsets: harmful and non harmful symbols. These symbols were collected from different web sites (racist, xenophobic, cults [i.e. religious intolerant sects], violence). Working has begun towards developing invariant descriptors for these symbols. During the first year of the project, only preliminary work has been undertaken for symbol detection. The remaining development of other descriptors and the integration inside the POESIA framework will be done in the second year.

Natural Language Processing text filtering

An important feature of POESIA is its ability to analyse textual content (in HTML and plain text only⁴). A tag stripper removes HTML tags from HTML. Then, such textual content is first processed by a language identification filter, which redirects it (using a continuation message to the monitor) to a language specific text filter. Language specific filters are first light filters (able to produce scores for most texts) and sometimes more time and resource-consuming heavy filters.

Language identification. The languages addressed for text filtering in POESIA are English, Spanish, Italian and (at a later stage) French. Before text filtering can be done, the language used within a document must be identified. A component to perform language identification has been implemented in Java. The approach used computes statistics for letter n-grams in the supported languages, and compares these statistics to those obtained for each new document, to determine its most likely language. Two statistical models have been implemented: smoothed frequency probability distribution, compared using cross-entropy, and a simple frequency rank, which are compared using an 'out-of-place' measure.

³ Detection of a symbol as a small part of a larger image is out of scope.

⁴ Text extraction from image using OCR techniques is out of scope of this project, but might be useful, and could be integrated into POESIA framework.

Text Filtering. Within POESIA, both ‘light’ and ‘heavy’ filtering components are to be provided for each filtered language, where the former is simpler and faster, and the latter is computationally more expensive, making use of more complicated NLP techniques. Somewhat different language-dependent approaches are being used in the construction of heavy filters for the filtered languages, as detailed below. For light filtering, the techniques employed are rather more language independent, and there is some reasonable overlap in the approaches being developed by the different partners. Specifically, documents can be represented as vectors of weights for terms (e.g. word, stem, etc), where weighting depends on a term’s frequency both within and across documents. A statistical or machine learning technique can be applied to this representation to learn a model that is used to assign the most likely category to new documents.

Text Filtering for Spanish. A light filter has been implemented using various term weighting schemes, feature (i.e. term) reduction using an Information-Gain metric, and employing Support Vector Machines (SVM) to induce the categorisation model. The heavy filter being developed employs a similar approach to category learning but allows a more sophisticated representation of relevant “terms”, e.g. as phrase chunks, named entities, etc. Current work is investigating the automatic identification of significant multi-word-expressions using maximum-entropy techniques.

Text Filtering for Italian. A light filter has been implemented that employs a simple statistical word-based approach, after a feature reduction along the lines outlined above. The categorization scheme presently involves local term counting rather than global frequency computation. A heavy filter is being developed in which NLP-based pre-processing is first applied to documents, including multi-word expression recognition, morpho-syntactic tagging and lemmatisation. Category learning over the resulting representations is performed using a new entropy-based classification technique, called CASSANDRA (Complex Analysis of Sequences via Scaling AND Randomness Assessment).

Text Filtering for English. A light filter has been written in Java which implements three different categorisation methods, namely categorisation by cosine vector similarity (with inverse document frequency weighting), by cross-entropy (using a smoothed frequency probability distribution of terms), and by out-of-place rank (using simple frequency ranking of terms). The heavy filter being developed is based on recognising categorisation-relevant substructures within the results of NLP analysis of text. A number of the NLP tools needed to construct this filter are already available, and progress has been made in generating one of the key missing components, a fast robust chunk parser.

To summarize, during the first year of the project work mainly focused on: a) general architectural issues concerning heavy vs light NLP-based filtering for each language; b) language identification; c) preliminary definition of the internal architecture of the language-specific filters; d) first implementation and/or adaptation and tuning of some NLP modules. The use of NLP methods implies the need for a range of resources (namely, processing tools, computational lexicons and corpora), for use both as part of filtering systems and to aid their development. Some of the required resources already existed before POESIA and require tuning/adaptation to the POESIA task; other resources are being developed as part of the project. More details on the work being done can be found in Section 7 “NLP-based text filtering” of D3.1 “Software Architecture Definition Document”, and in D7.2 “Lexical Resources and Tools for Each Language”.

URL, PICS, and links based filtering

In addition to the sophisticated content-filtering techniques above, POESIA also provides URL based filtering, PICS based filtering, and links (both static and dynamic) based filtering (see the first version of D5.1 “URL, PICS and JavaScript Report”).

URL filtering is a simple but useful technology. It just means that an HTTP request is filtered by the requested URL only. In POESIA some DNS domains or URL prefixes may be left unfiltered or

always rejected. Also, a set of positive (= accepted) and negative (= rejected) database⁵ of URLs is managed by the POESIA system. These database of URLs are actually prefixes (with the hostname reversed, so it is possible to reject any URL from the "sex.com" domain).

POESIA also contains a PICS filtering module. PICS provides an effective and responsible way to filter Web content and helps serving diverse audiences the appropriate material. For this reason PICS has been proposed by the World Wide Web Consortium in 1995 and has since been used and endorsed by major software industry. The PICS filter translates PICS rating labels into POESIA scores. It was developed during the first year.

POESIA will also contain a static links filtering module, which will compute a score by getting all the static links inside an HTML page and comparing them against the database of URLs.

POESIA will also perform dynamic links filtering thru Javascript static analysis. Since most current HTML pages contain Javascripts, it is required to do some static code analysis of these scripts (either inline, or embedded, so pre-fetched by POESIA). This code should be statically analysed, because it cannot be interpreted outside of a client browser - since it depends upon user actions. By a detailed study of the actual use of Javascripts in current pages, it was assessed that usual (but quite complex) static analysis techniques such as abstract interpretation are not enough for effective analysis of Javascript. More heuristic based analysis is required to approximate the set of dynamic links reachable thru scripts (and then compute a score with it). Preliminary and exploratory work on Javascript analysis was achieved during the first year; most of this effort will happen (as scheduled) in the second year of the project.

Decision mechanisms

The decision mechanisms (or decisors) are given as input the scores of the given content (usually each score is computed by some filters, but it can also have been cached by the monitor) and produce an accepting or rejecting decision. Decisors are managed by the monitor. A decisor may produce a decision without having a score from every relevant filter: for instance, a page could be rejected just because it contains strong porn images, without having completed its textual analysis.

Effective decision mechanisms use some kind of machine learning (in its broadest sense) approach and requires training using real filtering components. During the first year, a trivial decisor has been implemented. Implementing, and most importantly, tuning, other decisors mechanisms will be done in the second year, and availability of a working alpha-version (which goes out at the end of this first year) of the entire system is a prerequisite for this. Decisors mechanisms will follow the progress of the POESIA filters. More details can be found in the preliminary version of D8.1 "DM Filtering Components Software".

Networking, dissemination and deployment

Many potential users (including persons making decisions in educational institutions) have been contacted and expressed interest in this project. Several presentations of the project have been undertaken both in academic and promotional conferences and also through informal meetings.

Various presentations and communications have been undertaken about POESIA, notably by contacting several potential users (schools, ...) in partner countries and by attending workshops, conferences, exhibitions and other events.

Last 15th October 2002 the Fundació Catalana per a la Recerca (Barcelona, SPAIN) (in conjunction with European SchoolNet) organised a major European workshop "Educating and Protecting

⁵ But building and maintaining a realistic and actual set of positive and negative URLs is outside the scope of this project, and provides significant market opportunities to European corporations.

Children on the Information Society - Lessons from European Projects", which was an exchange between different Safer Internet Action Plan projects. There was success in gathering together many European educational institutions, media and members of the European Commission. The ICRA (Internet Content Rating Association) has been contacted, and as a result collaboration with the PRINCIP (a fellow IAP) project has been initiated. Evaluation and deployment of the POESIA software will occur after availability of the first alpha or beta releases of the filtering system. A dissemination specific Web server is being built. The POESIA project and outcomes were also presented to participants (the 'industry', media, researchers, consumer organisations) at the European Commission eSafe launch event in Luxembourg and at the very large BETT (British Educational Technology and Training) exhibition in January 2003 in London.

Remaining future work

For the Developers team, the bulk of the future work is to continue the development effort, taking into account the continuous feedback of End Users, and also using real examples from the Web to tune and enhance the software.

For the End User team, the main aspects of future work will be to establish a final testing, evaluation and implementation process, to support Developers through undertaking trials, tests and evaluation and to disseminate to End User groups the outcomes of the project. The End User Group will also continue to make contacts and connections to the new Internet Awareness projects (Safeborders, SIFKAL etc.) to seek to promote POESIA to a wide European audience.

A momentum begins to be created around this project, which will probably continue in some other settings.

Deliverables produced during the first year of the project

Deliverable No.	Deliverable Title	Date Due
D 1.1.a	Progress Report 1	t6
D 1.2.a	Management Report 1	t3
D 1.2.b	Management Report 2	t9
D 1.3	Annual Report	t12
D 2.1	End-user Requirements Report	t6
D 2.2	Test Case Files	t6, t12
D 3.1	Software Architecture Definition Document	t6, t9
D 4.1	Alpha Release of POESIA Software	t12
D 5.1	URL, PICS and JavaScript Report	t9 (t18)
D 6.1	Image Subsets	t12
D 7.1	Domain Corpora for Each Language	t12 (t18)
D 7.2	Lexical Resources and Tools for Each Language	t12
D 8.1	DM Filtering Components Software	t12 (t21)
D 9.5	Dissemination Report and WWW Server	t9 (t24)