Logiweb - a system for web publication of mathematics

Klaus Grue

Dept.Comp.Sci, Univ.of Copenhagen (DIKU) grue@diku.dk

Abstract. Logiweb is a system for electronic publication and archival of machine checked mathematics of high typographic quality. It can verify the formal correctness of *pages*, i.e. mathematical papers expressed suitably. The present paper is an example of such a Logiweb page and the present paper is formally correct in the sense that it has been verified by Logiweb. The paper may of course contain informal errors like any other paper. Logiweb is neutral with respect to choice of logic and choice of notation and can support any kind of formal reasoning.

Logiweb uses the World Wide Web to publish Logiweb pages and Logiweb pages can be viewed by ordinary Web browsers. Logiweb pages can reference definitions, lemmas, and proofs on previously referenced Logiweb pages across the Internet. When Logiweb verifies a Logiweb page, it takes all transitively referenced pages into account.

1 Introduction

Logiweb is a web-like system that allows mathematicians and computer scientists to web-publish pages with high typographic quality and high human readability which are also machine verifiable. Among other, Logiweb allows pages to contain definitions of formal theories, definitions of new constructs, programs, lemmas, conjectures, and proofs. Furthermore, Logiweb allows pages to refer to each other across the Internet, and allows proof checking of proofs that span several pages that reside different places in the world. As an example, a lemma on one page may refer to a construct which is defined on another page situated elsewhere, in which case the proof checker must access both pages to establish the correctness of the proof.

Logiweb is accumulative and provides a medium for archived mathematics, c.f. Section 2.2. In contrast, the World Wide Web is a medium suited for information in flux, i.e. information which may be updated at any time without notice. There are many formats like MathML and OMDoc [1,2] which are suited for mathematics on the web. After many experiments and investigations, html, pdf, and TEX have been chosen as interface formats of Logiweb until further, but Logiweb is an open system for which individual users may add support for further formats. Logiweb also has its own, internal format and referencing system which is particularly suited for archived mathematic.

Like the Internet and the WWW, Logiweb is a robust, 'anarchistic' system that runs without any central authority.

Logiweb gives complete notational freedom to its users as well as complete freedom to choose any axiomatic theory (e.g. ZFC) as basis for their work. Logiweb also allows different notational systems and theories to co-exist and interact smoothly.

Logiweb may be used as it is but also has the potential to support other systems like Mizar [3,4]. Logiweb was originally designed to support Map Theory [5,6,7,8] which has the same power as ZFC but relies on very different foundations in that, e.g., it relies on λ-calculus *instead* of first order predicate calculus. However, Logiweb has been designed such that it supports all axiomatic theories equally well so the ability to support Map Theory should be seen as a widening rather than a narrowing of the scope.

Logiweb supports classical as well as intuitionistic logic, it supports theories built on first order predicate calculus as well as other brands of theories, and it supports theories (such as Map Theory) which admits general recursive definitions.

The absence of restrictions on the choice of logic of course makes it impossible to supply a general code-from-theorems extraction facility like `term_of` of Nuprl [9], but functions for manipulation of theorems and proofs of individual theories are expressible in the programming language of Logiweb.

One goal of Logiweb was to design a simple proof system which allows to cope with the complexity of mathematical textbooks. To ensure that the system can cope with the complexity of a full, mathematical textbook in a human readable style, two books [10,11] have been developed 1992-2002 to test the system.

Reference [10] is a discrete math book for first year university students and is of interest here because it has been possible to test the human readability of the book in practice. The associated course has been given ten times with a total of more than a thousand students. The course has been a success and runs as the first course on the computer science curriculum at DIKU in parallel with a course on ML.

Reference [11] is a treatise on Map Theory and is of interest here because it contains a substantial proof (a proof of the consistency of ZFC expressed in Map Theory) that can stress test Logiweb. To allow comparison with other proof systems and to ensure correctness, [11] has been ported by hand to Isabelle [12,13,7].

During 2005 and 2006, Logiweb has been used on two graduate courses in logic (c.f. 'Student reports' at `http://logiweb.eu/`) and Logiweb is being adapted according to user requests. After that, it is the intension to run first [11] and then [10] through the system. Running those two books through Logiweb requires adaption of the books to the current syntax of the Logiweb compiler plus programming of a number of proof tactics that are described but not formally defined in the books. Running [11] through Logiweb will also allow a comparison with Isabelle.

Logiweb includes a programming language, macro expansion facilities, proof checking facilities, means for expressing proof tactics, a protocol for exchanging information about pages, a format for storing and transmitting pages, machinery for rendering, machinery for compiling programs, machinery for referencing, and many other facilities. Each facility is simple, but the sum of features makes it impossible to cover everything here. For a comprehensive introduction to Logiweb, consult Logiweb itself at `http://logiweb.eu/` and read the 'base' page.

Section 3 gives examples of theories, lemmas and proofs expressed using Logiweb. Section 2 gives some background information about the structure of Logiweb and Section 4 gives a short description of how Logiweb has been used in teaching mathematics. Section 5 gives an overview of the implementation of the system.

2 Description of Logiweb

2.1 Vectors and references

Logiweb stores and transmits Logiweb pages in its own, internal format named the *Logiweb interchange format* and interacts with other systems using standardized formats like html, PDF, Lisp S-expressions, and many other formats. We shall refer to pages expressed in the Logiweb interchange format as *vectors*. Vectors are sequences of bytes.

Each Logiweb page has a unique Logiweb reference. A reference is a sequence of about thirty bytes. When expressed base 64 [14], the reference of the present paper is BokG7o6Za5JH5cFpE6nduncQ-rtRhuYtj-e_iigBB. The reference contains a protocol version number (always one, reserved for future extensions), a RIPEMD-160 [15] hash key, and a time stamp.

One may look up the present page on Logiweb following the link http://logiweb.eu:8080/logiweb/server/relay/64/BokG7o6Za5JH5cFpE6nduncQ-rtRhuYtj-e_iigBB/2/index.html. When clicking that link, a CGI-program at http://logiweb.eu:8080/logiweb/server/relay/ relays the reference BokG7o6Za5JH5cFpE6nduncQ-rtRhuYtj-e_iigBB to a Logiweb *server*, which resolves the reference and directs the users browser to the page.

Once a Logiweb page is submitted to Logiweb, its Logiweb reference remains fixed whereas its location may change. At any time there may exist many copies of each page on Logiweb and all the copies may be moved around, but all Logiweb servers cooperate on keeping track of all Logiweb pages at any time.

2.2 Immutability

Logiweb pages are *immutable* in the sense that if a user looks up the same Logiweb reference twice, even with years in between, and if Logiweb still has a copy of the page somewhere in the world, then the user can be sure to receive exactly the same Logiweb page the two times. If there are no copies left, then the user can be sure to get a message that the page does not exist anymore. Any user of Logiweb can mirror any Logiweb page and thereby ensure its continued existence. In this way, Logiweb pages are as stable as papers published in paper journals: If all copies of a paper journal burn, the contents is lost, but as long as at least one copy remains, the contents is still available.

An important feature of paper journals is that one cannot change a paper after publication. That makes it safe to refer e.g. to a definition in a published paper since the definition is fixed after publication. Logiweb mimics this feature through immutability.

2.3 Rendering

Given a Logiweb vector, i.e. a Logiweb page expressed in the Logiweb interchange format, Logiweb is able to *render* and *verify* the page.

Logiweb may render pages in arbitrary formats. The present page is rendered as one html and two PDF files, located the following places:

- http://logiweb.eu:8080/logiweb/server/relay/64/BokG7o6Za5JH5cFpE6nduncQ-rtRhuYtj-e_iigBB/2/body/tex/page.pdf
- http://logiweb.eu:8080/logiweb/server/relay/64/BokG7o6Za5JH5cFpE6nduncQ-rtRhuYtj-e_iigBB/2/body/tex/appendix.pdf [16]
- http://logiweb.eu:8080/logiweb/server/relay/64/BokG7o6Za5JH5cFpE6nduncQ-rtRhuYtj-e_iigBB/2/body/tex/page.html

The first link points to the present paper. The second link points to an electronic appendix with definitions that would bore most readers (such as definitions of how each construct should be rendered). The second link is included in the BibTeX bibliography [16] for easy reference. The third link points to a table of contents.

2.4 Verification

When verifying a page, Logiweb collects all definitions and verifies all claims found on the page. As an example, one may define

$$[\mathsf{n}! \doteq \textbf{if } \mathsf{n} = 0 \textbf{ then } 1 \textbf{ else } \mathsf{n} \cdot (\mathsf{n} - 1)!]$$

and one may claim

$$[3! = 6]^{\cdot}$$

which makes Logiweb verify that 3! equals 6 according to the given definition of the factorial function.

The claim above is a rendering of the bytes in the vector whose correctness is established by the verifier. This ensures that a human reader and the machine verifier see the same claims.

Verification ignores all text that has no formal meaning. The $\doteq$ and $[\cdots]^{\cdot}$ constructs have formal meaning in that they introduce definitions and claims, respectively. Verification ignores running text such as the present sentence and also ignores formulas like 3! which neither form a definition nor a claim. Section 3 gives examples of more advanced definitions like definitions of axiomatic systems and more advanced claims like mathematical proofs (where the claim to be verified is that the proof is correct).

2.5 Bibliographies

Each Logiweb page contains a Logiweb bibliography i.e. a list of references to other Logiweb pages. The Logiweb bibliography of the present paper (not to be confused with the BibTeX bibliography at the end of the paper) references two other Logiweb pages:

– http://logiweb.eu:8080/logiweb/server/relay/64/BAGTyrgl5Qmkb-DUmGqAWUda0vx9aHftYSe_iigBB/2/ [17]
– http://logiweb.eu:8080/logiweb/server/relay/64/BsPa2QtKsY6_LULMdLl-THnDv58AVac-Mbe_iigBB/2/ [18]

The references above have been included in the BibTeX bibliography for easy reference but there is in general no link between the Logiweb and the BibTeX bibliographies.

The definition

$$[\mathsf{n}! \doteq \textbf{if } \mathsf{n} = 0 \textbf{ then } 1 \textbf{ else } \mathsf{n} \cdot (\mathsf{n} - 1)!]$$

of the factorial function refers, among other, to multiplication $\mathsf{x} \cdot \mathsf{y}$. [17] both defines how to evaluate and how to render multiplication, and multiplication is available on the present page because [17] is included in the Logiweb bibliography of the present page. The immutability of [17] ensures that $\mathsf{x} \cdot \mathsf{y}$ denotes multiplication every time $[3! = 6]^{\cdot}$ is verified.

3 Example proofs

3.1 Theories

We now give a very simple example of a theory, a lemma, and a proof. The theory is classical propositional calculus L as defined in [19]:

– [**Theory** L]
– $[\mathrm{L}$ **rule** $\mathrm{MP}\colon \Pi\mathcal{A}\colon \Pi\mathcal{B}\colon \mathcal{A} \vdash \mathcal{A} \Rightarrow \mathcal{B} \vdash \mathcal{B}]$
– $[\mathrm{L}$ **rule** $\mathrm{A1}\colon \Pi\mathcal{A}\colon \Pi\mathcal{B}\colon \mathcal{A} \Rightarrow \mathcal{B} \Rightarrow \mathcal{A}]$
– $[\mathrm{L}$ **rule** $\mathrm{A2}\colon \Pi\mathcal{A}\colon \Pi\mathcal{B}\colon \Pi\mathcal{C}\colon (\mathcal{A} \Rightarrow \mathcal{B} \Rightarrow \mathcal{C}) \Rightarrow (\mathcal{A} \Rightarrow \mathcal{B}) \Rightarrow \mathcal{A} \Rightarrow \mathcal{C}]$
– $[\mathrm{L}$ **rule** $\mathrm{A3}\colon \Pi\mathcal{A}\colon \Pi\mathcal{B}\colon (\neg\mathcal{B} \Rightarrow \neg\mathcal{A}) \Rightarrow (\neg\mathcal{B} \Rightarrow \mathcal{A}) \Rightarrow \mathcal{B}]$

Logiweb macro expands [**Theory** L] into a definition of L. That definition defines L as the meta-conjunction of all rules attributed to L.

Axiom A1 above says that $\mathcal{A} \Rightarrow \mathcal{B} \Rightarrow \mathcal{A}$ holds for all terms $\mathcal{A}$ and $\mathcal{B}$. The meta-quantifier $\Pi\mathcal{A}\colon \mathcal{B}$ states that $\mathcal{B}$ holds for all *terms* $\mathcal{A}$ as opposed to the object quantifier $\forall \mathsf{x}\colon \mathcal{B}$ which states that $\mathcal{B}$ holds for all *values* x.

The electronic appendix [16] defines priority and associativity such that $\Pi\mathcal{A}\colon \Pi\mathcal{B}\colon \mathcal{A} \Rightarrow \mathcal{B} \Rightarrow \mathcal{A}$ means $\Pi\mathcal{A}\colon \Pi\mathcal{B}\colon (\mathcal{A} \Rightarrow (\mathcal{B} \Rightarrow \mathcal{A}))$.

Inference rule L above expresses the rule of modus ponens. $\Pi\mathcal{A}\colon \Pi\mathcal{B}\colon \mathcal{A} \vdash \mathcal{A} \Rightarrow \mathcal{B} \vdash \mathcal{B}$ means $\Pi\mathcal{A}\colon \Pi\mathcal{B}\colon (\mathcal{A} \vdash ((\mathcal{A} \Rightarrow \mathcal{B}) \vdash \mathcal{B}))$.

The rules of a theory may be stated together as above or may be scattered throughout one Logiweb page. Presentations of mathematical logic often scatters the axioms of a theory over several chapters (c.f. the definition of set theory in [19]). Support for scattering is not included in Logiweb itself; it is implemented in [17] and is available to all users of Logiweb who reference that page. Users who want another style of presentation may replace [17] by their own alternative.

3.2 Lemmas and proofs

The first formal proof in [19] proves the following lemma:

$$[\mathrm{L}\ \textbf{lemma}\ \mathrm{I}\colon \Pi\mathcal{A}\colon \mathcal{A} \Rightarrow \mathcal{A}]$$

The lemma says that $\mathcal{A} \Rightarrow \mathcal{A}$ holds for all terms $\mathcal{A}$ in propositional calculus. The classical proof of that reads:

L **proof of** I:

L01:	Arbitrary $\gg$	$\mathcal{A}$	;
L02:	A1 $\gg$	$\mathcal{A} \Rightarrow (\mathcal{A} \Rightarrow \mathcal{A}) \Rightarrow \mathcal{A}$	;
L03:	A2 $\gg$	$(\mathcal{A} \Rightarrow (\mathcal{A} \Rightarrow \mathcal{A}) \Rightarrow \mathcal{A}) \Rightarrow$ $(\mathcal{A} \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}) \Rightarrow (\mathcal{A} \Rightarrow \mathcal{A})$	;
L04:	MP $\rhd$ L02 $\rhd$ L03 $\gg$	$(\mathcal{A} \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}) \Rightarrow (\mathcal{A} \Rightarrow \mathcal{A})$	;
L05:	A1 $\gg$	$\mathcal{A} \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}$	;
L06:	MP $\rhd$ L05 $\rhd$ L04 $\gg$	$\mathcal{A} \Rightarrow \mathcal{A}$	$\Box$

The proof above is rendered in a formalistic, tabular style close to that of [19]. Other styles, such as rendering as running text is also possible.

3.3 A proof by induction

Reference [18] defines Peano arithmetic S as follows:

- [**Theory** S]
- [S **rule** S1: $\Pi\mathcal{A},\mathcal{B},\mathcal{C}\colon \mathcal{A} = \mathcal{B} \vdash \mathcal{A} = \mathcal{C} \vdash \mathcal{B} = \mathcal{C}$]
- [S **rule** S2: $\Pi\mathcal{A},\mathcal{B}\colon \mathcal{A} = \mathcal{B} \vdash \mathcal{A}' = \mathcal{B}'$]
- [S **rule** S3: $\Pi\mathcal{A}\colon \neg 0 = \mathcal{A}'$]
- [S **rule** S4: $\Pi\mathcal{A},\mathcal{B}\colon \mathcal{A}' = \mathcal{B}' \vdash \mathcal{A} = \mathcal{B}$]
- [S **rule** S5: $\Pi\mathcal{A}\colon \mathcal{A} + 0 = \mathcal{A}$]
- [S **rule** S6: $\Pi\mathcal{A},\mathcal{B}\colon \mathcal{A} + \mathcal{B}' = (\mathcal{A} + \mathcal{B})'$]
- [S **rule** S7: $\Pi\mathcal{A}\colon \mathcal{A} \cdot 0 = 0$]
- [S **rule** S8: $\Pi\mathcal{A},\mathcal{B}\colon \mathcal{A} \cdot (\mathcal{B}') = (\mathcal{A} \cdot \mathcal{B}) + \mathcal{A}$]
- [S **rule** S9: $\Pi\mathcal{X},\mathcal{A},\mathcal{B},\mathcal{C}\colon \langle \mathcal{B} \equiv \mathcal{A} | \mathcal{X} := 0 \rangle \Vdash \langle \mathcal{C} \equiv \mathcal{A} | \mathcal{X} := \mathcal{X}' \rangle \Vdash \mathcal{B} \vdash \mathcal{A} \Rightarrow \mathcal{C} \vdash \mathcal{A}$][1]
- [S **rule** MP: $\Pi\mathcal{A},\mathcal{B}\colon \mathcal{A} \Rightarrow \mathcal{B} \vdash \mathcal{A} \vdash \mathcal{B}$]
- [S **rule** Gen: $\Pi\mathcal{X},\mathcal{A}\colon \mathcal{A} \vdash \forall\mathcal{X}\colon \mathcal{A}$]

[1] $\langle \mathcal{A} \equiv \mathcal{B} | \mathcal{X} := \mathcal{C} \rangle$ says '$\mathcal{A}$ is identical (except for naming of bound variables) to $\mathcal{B}$ where $\mathcal{X}$ is replaced by $\mathcal{C}$ in $\mathcal{B}$, c.f. [16]

– [S **rule** Ded: $\Pi \mathcal{A}, \mathcal{B}$: Ded$(\mathcal{A}, \mathcal{B}) \Vdash \mathcal{A} \vdash \mathcal{B}$]
– [S **rule** Neg: $\Pi \mathcal{A}$: $\Pi \mathcal{B}$: $\neg \mathcal{B} \Rightarrow \neg \mathcal{A} \vdash \neg \mathcal{B} \Rightarrow \mathcal{A} \vdash \mathcal{B}$]

Furthermore, [18] proves the following lemma taken from [19]:

[S **lemma** Prop 3.2c: $\Pi \mathcal{A}, \mathcal{B}, \mathcal{C}$: $\mathcal{A} = \mathcal{B} \vdash \mathcal{B} = \mathcal{C} \vdash \mathcal{A} = \mathcal{C}$]

Since the Logiweb bibliography of the present page points to [18], system S and the lemma above are available in the present paper. We may use that to prove $\forall x\colon 0 \cdot x = 0$ by induction in x. The induction step reads:

[S **lemma** Prop 3.2l′: $\Pi \mathcal{A}$: $0 \cdot \mathcal{A} = 0 \Rightarrow 0 \cdot \mathcal{A}' = 0$]

S **proof of** Prop 3.2l′:

L01:	Arbitrary $\gg$	$\mathcal{A}$	;
L02:	Block $\gg$	Begin	;
L03:	Arbitrary $\gg$	$\mathcal{A}$	;
L04:	Premise $\gg$	$0 \cdot \mathcal{A} = 0$	;
L05:	S8 $\gg$	$0 \cdot \mathcal{A}' = 0 \cdot \mathcal{A} + 0$	;
L06:	S5 $\gg$	$0 \cdot \mathcal{A} + 0 = 0 \cdot \mathcal{A}$	;
L07:	Prop 3.2c $\rhd$ L05 $\rhd$ L06 $\gg$	$0 \cdot \mathcal{A}' = 0 \cdot \mathcal{A}$	;
L08:	Prop 3.2c $\rhd$ L07 $\rhd$ L04 $\gg$	$0 \cdot \mathcal{A}' = 0$	;
L09:	Block $\gg$	End	;
L10:	Ded $\rhd$ L09 $\gg$	$0 \cdot \mathcal{A} = 0 \Rightarrow 0 \cdot \mathcal{A}' = 0$	$\Box$

In the proof above, the begin-end-block in Line L02-L09 proves $0 \cdot \mathcal{A}' = 0$ under the assumption $0 \cdot \mathcal{A} = 0$. The assumption in L04 involves the meta-variable $\mathcal{A}$, which has an arbitrary, fixed value according to line L03. Line L10 then uses deduction to conclude $0 \cdot \mathcal{A} = 0 \Rightarrow 0 \cdot \mathcal{A}' = 0$ where $\mathcal{A}$ is arbitrary according to L01 (L03 has no effect outside the block).

The premise in L04 has to be expressed using a meta-variable $\mathcal{A}$. A statement with a free object variable x is understood to hold for all values of the object variable so that e.g. $0 \cdot x = 0$ implicitly means $\forall x\colon 0 \cdot x = 0$ which is not suitable in line L04 above. Ordinary text books on mathematics typically leave it to the reader to guess the scope and kind of each variable, but we have to be more pedantic here to make the proofs machine verifiable.

We may now prove [S **lemma** Prop 3.2l: $\forall x\colon 0 \cdot x = 0$]:

S **proof of** Prop 3.2l:

L01:	S7 $\gg$	$0 \cdot 0 = 0$	;
L02:	Prop 3.2l′ $\gg$	$0 \cdot x = 0 \Rightarrow 0 \cdot x' = 0$	;
L03:	S9 @ x $\rhd$ L01 $\rhd$ L02 $\gg$	$0 \cdot x = 0$	;
L04:	Gen $\rhd$ L03 $\gg$	$\forall x\colon 0 \cdot x = 0$	$\Box$

The proofs given so far are rather trivial but still illustrate what the Logiweb system can do. For more complex examples, consult `http://logiweb.eu/` and [10,11].

4 Uses of Logiweb

Until further, Logiweb has been used in a course on mathematical logic for graduate students. The course was given in the spring of 2005 and again in the spring of 2006. Predecessors of the course, using predecessors of Logiweb, have been given a number of times since 1986.

During the above mentioned course on mathematical logic, the students form groups and choose a theorem they want to prove. After that each group writes a report which formally proves the theorem, verifies and publishes the report using Logiweb, and hands in the Logiweb reference of the report by e-mail.

Immutability (as guaranteed by RIPEMD-160 [15]) ensures that the report is not changed after it is handed in. Students will typically publish many attempts before they produce a version they want to hand in and, if they want, they may continue publishing new versions after handing in the report, but the students get a mark based on the version that matches the reference they send by e-mail.

Students typically base their reports on other Logiweb pages such as [17] and [18] which define elementary notions. The students typically locate the pages they want to reference by ordinary html browsing e.g. starting at `http://logiweb.eu/` or the home page `http://www.diku.dk/undervisning/2006s/202/` of the course.

It is the intension to use Logiweb also for teaching of mathematics for computer science based on [10]. In particular, it is planned to let the students answer exam questions using Logiweb during a four hour written exam. This scenario has been tested with success on a small group of first year students in 2004 using an old version of Logiweb.

5 Implementation overview

A user may use the World Wide Web as shown in Figure 1. In the figure, the user may use the text editor to construct an html page and store it in the file system within reach of the http server. Then the user (or another user) may use the html browser to request the html page from the http server which in turn retrieves the html page from the file system.

Figure 2 shows how a user may use Logiweb. To write a Logiweb page, the user prepares a source text and invokes the Logiweb compiler on it. This is similar to running TEX on a TEX source [20]. Actually, much of a Logiweb source consists of TEX source code.

If the compiler succeeds in interpreting the source, then the compiler translates the source to a Logiweb vector, checks the correctness of the vector, and stores the vector back in the file system within reach of the http server. The compiler also renders the page in PDF so that users without a genuine Logiweb browser can view it. After that, any user that knows the url of the page can retrieve it using an html browser.

When the compiler succeeds in translating a Logiweb page, it also computes the Logiweb reference of the page and notifies the Logiweb server (c.f. Figure 2). The Logiweb server keeps track of the relationship between http urls and Logiweb references and makes the relationship available via the Internet using the Logiweb protocol. The Logiweb protocol allows Logiweb servers to cooperate on indexing pages such that each server merely has to keep track of local pages plus

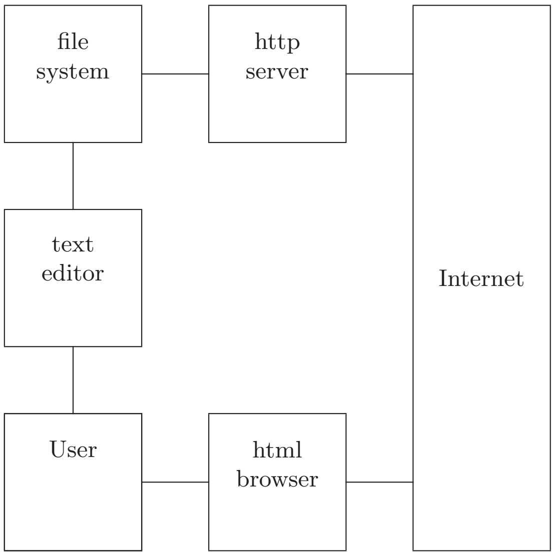

Fig. 1. World Wide Web

some information about which other Logiweb servers to refer non-local requests to.

When the compiler translates a Logiweb page that references other Logiweb pages (which is the normal case), it uses the Logiweb server to locate the references and then transitively loads the referenced pages so that all definitions on transitively referenced pages are available.

A Logiweb relay (c.f. Figure 2) is a CGI-program which, given a reference, contacts the nearest Logiweb server, translates the reference to an ordinary url, and returns an html indirection to that url. This instructs the html browser of the user to fetch the associated page. The net experience for the user is that clicking a Logiweb reference in an html page makes the html browser navigate to the referenced Logiweb page. One may construct a reference by appending the following:

- an url like http://logiweb.eu:8080/logiweb/server/relay/ of a relay,
- a Logiweb reference like 64/BokG7o6Za5JH5cFpE6nduncQ-rtRhuYtj-e_iigBBin base 16, 32, or 64,
- a number of levels like /2/ to go upwards in the directory structure (/2/ corresponds to /../../), and
- a relative reference like index.html

Referencing from Logiweb pages to html pages is trivial but not necessarily advisable since the immutability of Logiweb pages makes it impossible to repair broken links.

For more details on Logiweb see `http://logiweb.eu/` or [21].

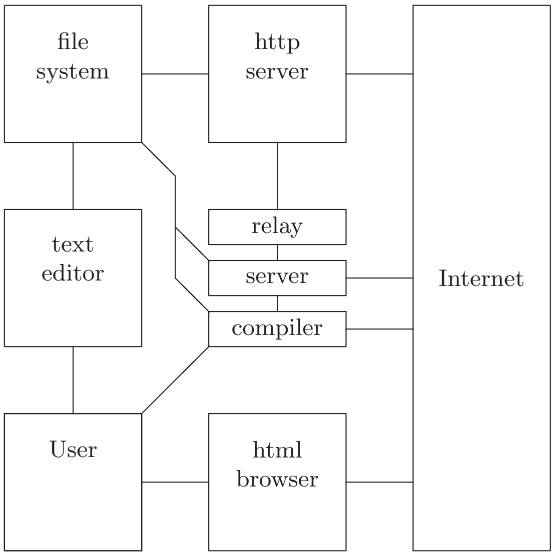

Fig. 2. Logiweb

6 Status

Logiweb allows users to author, render, publish, verify, retrieve, and read pages that contain formal mathematics. At the time of writing, more than 600 kilobyte of Logiweb source text has been verified by Logiweb. In addition, in 2005, ten graduate students have written eight reports that have been formally verified by a test version of Logiweb (c.f. http://logiweb.eu/), and the present version is being used in the spring of 2006 for another group of students. 800 kilobyte of formal proofs [11] await verification.

References

1. Kohlhase, M.: OMDoc: An open markup format for mathematical documents (version 1.1) (2003) http://www.mathweb.org/omdoc/index.html.
2. Miner, R., Schaeffer, J.: A gentle introduction to MathML (2001) http://www.dessci.com/en/support/tutorials/mathml/default.htm.
3. Trybulec, A., Blair, H.: Computer assisted reasoning with MIZAR. In Joshi, A., ed.: Proceedings of the 9th International Joint Conference on Artificial Intelligence, Los Angeles, CA, Morgan Kaufmann (1985) 26–28 http://www.mizar.org/.
4. Muzalewski, M.: An Outline of PC Mizar. Foundation of Logic, Mathematics and Informatics, Mizar User Group, Brussels. (1993)
5. Berline, C., Grue, K.: A κ-denotational semantics for Map Theory in ZFC+SI. Theoretical Computer Science **179**(1–2) (1997) 137–202
6. Grue, K.: Map theory. Theoretical Computer Science **102**(1) (1992) 1–133
7. Skalberg, S.C.: An Interactive Proof System for Map Theory. PhD thesis, University of Copenhagen (2002) http://www.mangust.dk/skalberg/phd/.

8. Vallée, T.: "Map Theory" et Antifondation. Volume 79 of Electronic Notes in Theoretical Computer Science. Elsevier (2003)
9. Constable, R.L., Allen, S., Bromly, H., Cleaveland, W., Cremer, J., Harper, R., Howe, D., Knoblock, T., Mendler, N., Panangaden, P., Sasaki, J., Smith, S.: Implementing Mathematics with the Nuprl Proof Development System. Prentice-Hall, Englewood Cliffs, New Jersey (1986)
10. Grue, K.: Mathematics and Computation. 7 edn. Volume 1–3. DIKU, Universitetsparken 1, DK-2100 Copenhagen (2001)
11. Grue, K.: Map theory with classical maps. Technical Report 02/21, DIKU (2002) http://www.diku.dk/publikationer/tekniske.rapporter/2002/.
12. Paulson, L.C.: Introduction to Isabelle. Technical report, University of Cambridge, Computer Laboratory (1998)
13. Paulson, L.C.: The Isabelle reference manual. Technical report, University of Cambridge, Computer Laboratory (1998)
14. Berners-Lee, T., Masinter, L., McCahill, M.: Uniform Resource Locators (URL). RFC 1738, Internet Engineering Task Force (1994) http://rfc.net/rfc1738.html.
15. Dobbertin, H., Bosselaers, A., Preneel, B.: RIPEMD-160: A strengthened version of RIPEMD. In: Fast Software Encryption. (1996) 71–82 http://citeseer.ist.psu.edu/dobbertin96ripemd.html.
16. Grue, K.: Logiweb - a system for web publication of mathematics - appendix. Technical report, Logiweb (2006) http://logiweb.eu:8080/logiweb/server/relay/64/BokG7o6Za5JH5cFpE6nduncQ-rtRhuYtj-e_iigBB/2/body/tex/appendix.pdf.
17. Grue, K.: A logiweb base page. Technical report, Logiweb (2006) http://logiweb.eu:8080/logiweb/server/relay/64/BAGTyrgl5Qmkb-DUmGqAWUda0vx9aHftYSe_iigBB/2/.
18. Grue, K.: The logiweb sequent calculus. Technical report, Logiweb (2006) http://logiweb.eu:8080/logiweb/server/relay/64/BsPa2QtKsY6_LULMdLl-THnDv58AVac-Mbe_iigBB/2/.
19. Mendelson, E.: Introduction to Mathematical Logic. 3. edn. Wadsworth and Brooks (1987)
20. Knuth, D.: The TeXbook. Addison Wesley (1983)
21. Grue, K.: Logiweb. In Kamareddine, F., ed.: Mathematical Knowledge Management Symposium 2003. Volume 93 of Electronic Notes in Theoretical Computer Science., Elsevier (2004) 70–101