Decentralized Proof-Term Library

Michael Nahas

affliliated with
Radboud Universiteit Nijmegen

New to the Research Field

Expert in:

- File systems
- Distributed computing
- Caching
- Error Correction
- ...

New to the Research Field

Not an expert (yet) in:

- ITP
- Type Theory
- Coq
- Isabelle/Isar, Isabelle/HOL, ACL2, Agda, ...
- Matita

New to the Research Field

Not an expert in:

- Libraries
- Machine Learning
- Information Retrieval

Decentralized Proof-Term ~~Library~~ *File System*?

What is a Library?

A storage place for information

Examples:

- Library (books)
- Dictionary (words + definitions)
- Phonebook (phone# and addresses)
- Webserver (hypertext)
- IMDB, Wikipedia, … Netflix! Facebook?

Why do we want a library?

Duplicate work!

“I know someone has already proved this!

… or something similar.”

“nanos gigantum humeris insidentes”

- Bernard of Chartres

Processes of a Library

- Collect
- Authenticate
- Index + Store
- Maintain
- Serve Users:
  - Browsing
  - Learning
  - Searching

Serving Users

- Browsing (meta-data + priorities)
  - “paging through the ToC / Index”
  - “reading the abstract”
- Learning (copying data into brain (“cache”))
  - “reading a textbook / paper”
- Searching (accessing data *without* caching)
  - “finding the citation for this paper”
- News (streams of new or changed data)
  - “Fermat's Last Theorem has been proved”

Processes of a ~~Library~~ Dictionary

- Collect
- Authenticate
- Index + Store
- Maintain
- Serve Users:
  - Browsing
  - Learning
  - Searching

Multiple Libraries are OK

- Dictionary
- Thesaurus
- Vocabularies (words to learn)
- Glossaries (sub-dictionary)
- Oxford English Dictionary (super-dictionary?)

Decentralized Proof-Term Library

- Contains every proof term ever written
- Assumes distributed *computation*, not storage.
  - Many servers, each with full copy of library
  - After checking term, server cryptographically signs
- Proof-terms are indexed by hash#

Decentralized Proof-Term Library

- Collect
- Authenticate
- Index + Store
- Maintain
- Serve Users:
  - ~~Browsing~~
  - ~~Learning~~
  - Searching

Decentralized Proof-Term Library

- Good:
  - Very little effort by users
  - Proofs are trustworthy
  - Lots of data for data mining
  - Exact search is fast
- Bad:
  - Search that requires refining the query
  - Proofs that require modification
  - Users will not learn

Other Approaches

- Single Publisher
  - Mizar's MML, Coq StdLib, SSReflect, ...
- Packages
  - Isabelle/HOL's AFP, ACL2's “books”
- Wiki
  - MathWiki
- Other
  - Matita
  - OpenTheory?

Discussion! (Bloodsport!)

- Do we want a large, stable library?
- How to handle versioning?
- What is the cost-benefit of using a library?
- How much user contribution can we expect?
- How important are checked proofs?
- What is Coq's relationship to the library?
- Proof-term or script?

End of Talk

- Supplementary / optional slides follow

Processes of a Library

- Collect
- Authenticate
- Index + Store
- Maintain
- Serve Users:
  - Browsing
  - Learning
  - Searching

Proof-term or Script?

Proof terms are:

- More precisely defined
- Definition is more stable
- Requires less code in the server
- Designed for manipulation, comparison, etc.
- Would not constrain UI development in Coq

Proof terms are ***data***, not a ***list of commands***.

Proof-term or Script?

Scripts are sequences of commands.

Can we recreate the script from a proof-term?

IF we had a specification of the commands

AND remembered hints about which subterms were created with each command.

Does anyone know a good language for specifying commands?

Searching 1TB of Proofs Quickly?

Mostly solved problem. (Google, Hoogle, etc.)

Some interesting topics here:

- Content-based identifiers
- Canonical forms for types
- Searching proofs on equivalent types
- Query languages
- Machine Learning (e.g., ML4PG)