

MOQA: Meaning-Oriented Question-Answering

Jim Cowie¹, Sergei Nirenburg², Tanya Korelsky³, Stephen Helmreich¹,
Benoit Lavoie³, Valeriy Sibirtsev¹, Steve Beale² & Ted Caldwell³

¹New Mexico States University,
Computing Research Laboratory
Las Cruces, NM 88003
shelmrei@crl.nmsu.edu
{jcowie, shelmrei, vasib}
@crl.NMSU.Edu

²University of Maryland
Baltimore County,
Institute for Language and
Information Technology
1000 Hilltop Circle,
Baltimore, MD 21250
{sergei,sbeale}@cs.umbc.edu

³CoGenTex, Inc.
840 Hanshaw Road Suite 1,
Ithaca, NY 14850
{tanya,benoit,ted}@cogentex.com

Abstract

The paper describes a system which uses a fact repository (FR) consisting of instances of ontological concepts to answer sequences of questions. The fact repository is populated automatically by analyzing texts to produce complex text meaning structures. These are then mapped to fact repository instances and stored in a relational database. The documents used by the system are news articles in English, Arabic, and Farsi.

The FR, the ontology and the lexicons for the three languages, together with the *working memory* that includes the intermediate results of the system operation, are the major static knowledge sources in the system.

A multimedia interface allows queries to the system to be posed in a variety of ways including natural language. The user interacts with the system through input text, tables, generated text, maps, and relationship diagrams. The results of a query session are stored and can be reviewed and used to generate reports or to restart the session at any point in its history.

1 Introduction

Question understanding in our approach is implemented by using text analysis processes to produce a “text meaning representation” or TMR, for the question. Deep analysis of user input is necessary because: a) “natural” ways of asking a question include many that do not even contain any question words (which renders research in the classification (“grammar”) of questions on the basis of key words and their combinations only partially useful); b) lack of word sense disambiguation (in string-matching methods) leads to lower precision in IR and IE and also makes the creation of queries against structured databases a task of equal complexity to general text analysis; and c) only semantic analysis can bring about real progress in reference resolution and, eventually, treatment of non-literal language, both necessary conditions for radically improving the quality of human-computer interaction and of automatic information processing results. Once the basic text analysis is completed, the resulting TMR is used as input to the question interpretation module. In what follows, we first describe our approach to text analysis and the production of “facts” from text. Then we examine its extension to the interpretation module. Finally we examine the development of a multimedia interface which provides feedback on question interpretation and allows a user multiple ways to examine the knowledge held by the system.

The data sources used by the system are newswires in all three languages. These documents are captured by a web-spider and stored and indexed using a statistical retrieval system which supports the Unicode character set. A subset of these documents was made by searching for key words related to travel and to titles related to political leaders. These have been used to support the design and implementation of the analysis system described here.

2 Text Analysis

As the starting point for developing the MOQA semantic analyzer, we took its previous implementation used in an information extraction project (Cowie et al, 2000). The static knowledge resources with which we started MOQA development were inherited partially from the same IE project but more broadly from the results of a machine translation project (Mahesh, 1995; Mahesh and Nirenburg, 1996; Beale et al., 1996). The processing environment (illustrated in Figure 1) was enhanced by several modified microtheories (e.g., those of modality, time and reference; Nirenburg and Raskin, forthcoming) and several new ones (e.g., those of quantification and approximation). The main static knowledge resources – the lexicon and the ontology – have undergone significant changes. In particular, a completely new ontological-semantic lexicon of about 12,000 word senses was developed within MOQA. Lexical and morpho-syntactic coverage for Arabic and Persian was also added.

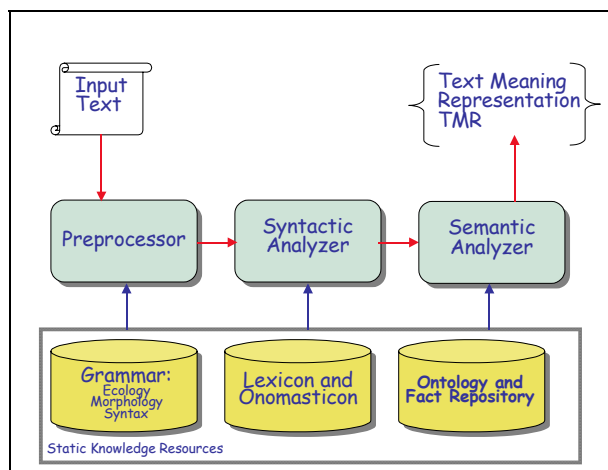


Figure 1: The architecture of the semantic analyzer

The development of the static knowledge resources supported one of the two main lines of research with respect to the analyzer, namely, *coverage* of input material. The other main line of research has to do with *robustness* of the analyzer in the face of unexpected input or, alternatively, residual ambiguity. In MOQA, we have introduced a facility for generating lexicon entries on the fly for elements of input that were not covered in the lexicon. The basic text meaning representation in the MOQA analyzer is derived by putting the input text through a couple of complex processing modules – the preprocessor and the syntactic analyzer – before it gets to the semantic analyzer proper. The preprocessor determines sentence boundaries, tags words and punctuation with their parts of speech, recognizes and normalizes dates, numbers, named entities, looks the words up in the system lexicon and carries out full morphological analysis of the words in the input (that is, determines their citation form and grammatical parameter values). Figure 2 presents a sample output from the preprocessor.

```

((674,677), root='hall',post=n,case=capitalized,text_form='Hall')
((674,677), root='hall',post=pn,case=capitalized,text_form='Hall')
((679,682), root='say',post=v,tense=past,case=lower,text_form='said')
((679,682), root='said',post=adj,case=lower,text_form='said')
((684,685), root='he',post=pron,case=lower,text_form='he')
((687,692), root='want',post=v,tense=past,case=lower,text_form='wanted')
((687,692), root='want',post=v,form=pastparticiple,case=lower,text_form='wanted')
((694,695), root='to',post=prep,case=lower,text_form='to')
((694,695), root='to',post=inf,case=lower,text_form='to')
((697,707), root='investigate',post=v,case=lower,text_form='investigate')
((710,716), root='report',post=n,number=plural,case=lower,text_form='reports')
((710,716), root='report',post=v,tense=present,person=third,case=lower,text_form='reports')

```

Figure 2: The output of the MOQA preprocessor

The syntactic analyzer takes as input the results of the preprocessor and, using information in the lexicon, produces a set of syntactic dependency structures. Figure 3 illustrates an input and an output of the syntactic analyzer.

```

((COLIN-POWELL N ((TYPE PN))) (ADDRESS V ((TENSE PAST))) (THE ART)
(UN-GENERAL-ASSEMBLY N) (TODAY DATE))
PRINTING OUT SYNTAX RESULTS AFTER PRUNING
0 - 5 : CL ((0 4 CL ADDRESS) (4 5 DATE TODAY)) ADDRESS 1
0 - 4 : CL ((0 1 NP COLIN-POWELL) (1 2 V ADDRESS) (2 4 NP UN-GENERAL-ASSEMBLY))
ADDRESS 1
0 - 1 : NP ((0 1 N COLIN-POWELL)) COLIN-POWELL 0
0 - 1 : N NIL COLIN-POWELL 0
1 - 2 : V NIL ADDRESS 1
2 - 4 : NP ((2 3 ART THE) (3 4 N UN-GENERAL-ASSEMBLY)) UN-GENERAL-ASSEMBLY 3
2 - 3 : ART NIL THE 2
3 - 4 : N NIL UN-GENERAL-ASSEMBLY 3
4 - 5 : DATE NIL TODAY 4
5 - 6 : SEM ((0 5 CL ADDRESS) (0 4 CL ADDRESS) (4 5 DATE TODAY)) NIL 5

```

Figure 3: An illustration of the operation of the MOQA syntactic analyzer

The semantic analyzer produces text meaning representations. Figure 4 illustrates, in a graphical form, the TMR for the sentence *Iran, Iraq and North Korea on Wednesday rejected an accusation by President Bush that they are developing weapons of mass destruction.*

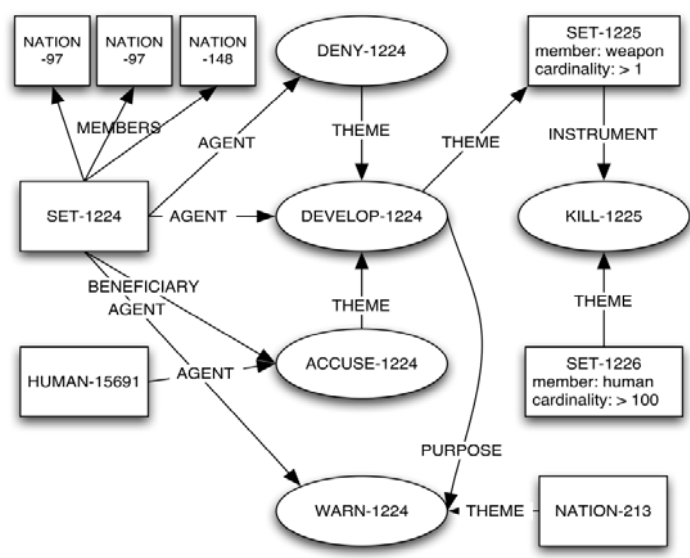


Figure 4: A sample MOQA TMR, presented in a graphical form

This representation is somewhat simplified. For example, it does not show the temporal relations among the various events in the input that the analyzer actually derives. TMRs are used at various stages of the question answering process. The process of text analysis that we briefly sketched above is used both in understanding the questions and in generating fact repository entries.

The operation of the analyzer is predicated on the availability of adequate knowledge resources. In the proposed system the latter include static resources – the *TMR language*; a general-purpose *ontology* stressing mutual (selectional) constraints on propositions and their arguments; an existing *fact repository*; a *semantic lexicon* for the support of both text analysis and text generation – and the resources that are dynamically created for information gathering for a particular analyst and for a particular Q&A session (or sessions). These resources include the dynamically added FR facts, the model of a specific user, an active set of goals, plans and scripts and the Q&A session log. In what follows, we give very brief descriptions of these static knowledge resources.

The ontology provides a metalanguage for describing the meaning of lexical units of a language as well as for the specification of meaning encoded in TMRs. The ontology contains specifications of concepts corresponding to classes of things and events in the world. In format, the Ontosem ontology that we are using in this project is a collection of frames, or named collections of property-value pairs, organized into an hierarchy with multiple inheritance. The expressive power of the ontology is enhanced by enhanced semantics of fillers for properties, specifically, in the use of what we call *facets*: DEFAULT, SEM, VALUE, RELAXABLE-TO that serve as multivalued selectional restrictions. The current ontology contains about 6,500 concepts, each of which has, on average, 16 properties defined for it.

The fact repository contains a list of remembered instances of ontological concepts. For example, whereas the ontology contains the concept CITY, the fact repository contains entries for London, Paris and Rome; and whereas the ontology contains the concept WAR, the fact repository contains the entry WWII. The main difference between an ontological concept and its instance is the nature of the fillers of properties for each. In the former, the fillers of properties are, typically, overridable constraints; in the latter, they are actual values (when known) or left unfilled when not known. A sample fact repository entry is illustrated in Figure 5.

HUMAN-15691	
NAME	George W. Bush
ALIAS	George Bush, President Bush, George W, the president of the United States, the US president
SOCIAL-ROLE	PRESIDENT
GENDER	male
NATIONALITY	NATION-1 (i.e., The United States of America)
DATE-OF-BIRTH	July 6, 1946
spouse	human-33966 (i.e., Laura Bush)

Figure 5: Sample Fact Repository entry

The ontological semantic lexicon contains not only semantic information; it also supports morphological and syntactic analysis and generation. Semantically, it specifies what concept, concepts, property or properties of concepts defined in the ontology must be instantiated in the TMR to account for the meaning of a given lexical unit of input. We illustrate the structure of the lexicon entry on the example of the first verbal sense of *alert* in Figure 6.

alert-v1	
cat	v
synonyms	warn
morph	regular
annotations example	"He warned us about the rain"
syn-struct	
subject root	\$var1 ;variables are used for linking to semantics
cat n	
root	\$var0
cat v	
indirectobject root	\$var4
cat n opt +	
pp-adjunct	root \$var2
cat prep	
root(or of about)	
opt +	
object root	\$var3
cat n	
sem-struct	
WARN	;WARN is the ontological concept underlying alert-v1
agent value	^\$var1
sem	human
theme	value ^\$var3
beneficiary	value ^\$var4
instrument	value ^\$var1
sem (or artifact event)	
^\$var2 null-sem	+

Figure 6: Sample semantic lexicon entry

For lack of space, we are not be able to discuss all the representational and descriptive devices used in the lexicon or the variety of the ways in which semantic information in the lexicon and the ontology can interact. **The Fact Repository**

The fact repository allows us to use a uniform format for all kinds of data. It is semantically anchored in general ontology. The repository is implemented as a PostgreSQL database and is implemented as a single table of instances -- concepts and properties drawn from Ontosem ontology.

The contents of the table are –

- Concept/Property (Class) name
- Description (Name)
- ID
- Pointer(s) to Other Instance(s)
- Document ID and offsets
- Reliability

Each ID in the Fact Repository is a unique instance. All names are stored as property fillers (HAS-ALIAS).

The reliability is a triple of three values which can range from 0.0 – 1.0. Textual reliability is based on modals, and hedges, which appear as epistemic modality (in TMR).

The contextual reliability is based on the estimated reliability of the source document. Finally the processing gives a reliability measure based on how much relaxation was needed to produce the TMR. The fact Repository ID is used both for document internal coreference and cross-document coreference. The analyser uses already existing facts to produce internal references in the TMR. Thus when a concept instance is used to populate the repository it may already use ID's from the repository. In this case the new information from the instance is added to the already existing information in the FR. The FR content was initially hand-crafted to allow concurrent testing of the interface. New data is now added automatically based on the TMR's produced by the analyser. Four kinds of concepts are acquired –

- Travel (TRAVEL-EVENTs)
- Contacts (COMMUNICATIVE-EVENTs)
- People (SOCIAL-ROLES)
- Organizations (ORGANIZATIONs)

Ancillary properties are also collected and stored. These include –

Semantic Roles

AGENT, THEME, SOURCE, DESTINATION, LOCATION, TIME-RELATION

Standard Properties

HAS-ALIAS, HAS-NATIONALITY, CAREER, HAS-RESIDENCE-CITY, IN-STATE, IN-COUNTRY

The original hand-coding covered meetings and travel events from 25 documents in English, Arabic, and Persian.

The automatic population uses a direct map from TMR to Fact Repository. Four English texts about Tony Hall have been processed and loaded into the fact repository.

Queries are passed to the fact repository by an intermediary process. This produces a structure which specifies the constraints on and concept types of the instances required. For structured queries which are produced by the interface menus there are a limited number of query topics: *organizations, people,*

travel, contacts and specialized mappings from Structured query to Fact Repository query are produced. In its turn the interface uses specialized presentation strategies and summarization/generation vocabulary.

A simple fallback strategy is used if the original query has no results; e.g. Did X go to Y on Z?

Not that we know, but
X1 did go to Y on Z
X did go to Y1 on Z
X did go to Y on Z1

These are offered to the user to allow him to see what information actually is held in the system.

When a natural language query is passed to the intermediary process it first calls the TMR analyzer which produces an analysis of the query. If the event types in the TMR match those of a known structured query then the TMR is mapped to that query and the structure returned to the user interface for confirmation of the question analysis.

TMRs identify the speech act:

REQUEST-INFO for questions
ASSERTIVE-ACT (I-F (IMPERATIVE)) for requests

They also identify the type of query (contact, travel, person), the theme of the Speech Act, and the specific information asked for.

The same process is used to map TMR directly to a Fact Repository query since the structure and content of the TMR parallels the structure of the Fact Repository. This is not dependent on domain-specific query formulation or result presentation.

The ongoing work on the fact repository includes providing a generation lexicon to improve the presentation of results. The system currently uses identity matching for concepts. The ontology maps ~25,000 lexical senses to ~9,000 concepts and properties. Thus

“Tony Hall travelled to Baghdad” is a TRAVEL-EVENT
“Bill Richardson went to Baghdad” is a MOTION-EVENT

The retrieval process is being changed to allow sibling concepts to be included as potential answers. Both the above events are children of a CHANGE-LOCATION event.

4 Ontology-based Multimedia User Interface

For answer presentation in MOQA, we have experimented with a unique multimedia design. While most current question answering systems present answers using mainly text extracts, our multimedia presentations include tables, fluent text summaries of the tables, maps and social network graphs. A similar multimedia approach for information extracted from video streams is used by Ng *et al*, 2003.

The user interface in MOQA accepts queries in natural language (NL) and presents answers in multimedia produced by the answer formulation and presentation (AFP) component. For the initial domain of people, organizations, contacts and travel events, there is also a structured menu-based query interface, which is used both for query formulation and for validation of NL queries. Analyzed (NL or structured) queries are automatically paraphrased back to the user in English using a disambiguated form of lan-

guage, with terms from the fact repository and resolved references, informing the user of the system’s interpretation of the query. Such paraphrases are a crucial feature of ontology-based query clarification and repair. Figure 7 shows the query “Where did Tony Hall travel?” analyzed and presented to the user for validation.

The screenshot displays a web interface for query validation. At the top, there are two tabs: "Natural Language Query" (selected) and "Structured Query". Under the "Natural Language Query" tab, a text input field contains the query "Where did Tony Hall travel?". Below this, the "Structured Query" tab is active, showing search criteria. The "Search Category" is set to "Travel Events". The "Search Criteria" section includes five input fields: "People Involved" (filled with "Tony Hall"), "Person Title" (empty), "Point of Departure" (empty), "Destination" (filled with "?"), and "Time Period" (empty). Each field has a small red arrow icon to its right. Below the criteria, there is a checkbox labeled "Show query paraphrase for confirmation" which is checked. A yellow box contains the paraphrased query: "Find the destinations of all travel events: which involve travelers with name similar to Tony Hall." At the bottom of the interface are two buttons: "Clear" and "Search".

Figure 7: Query validation

Note that, since the repository contains facts extracted from foreign sources, the spelling of names can vary, hence the formulation in the validated query: “travelers with name similar to Tony Hall”.

The AFP component communicates with the text analysis component and the answer content determination component in XML. Figure 8 illustrates the architecture of this communication. The AFP component involves 5 main processing steps:

- (1) *Configuration*: lets the user configure the connections with the text analysis and answer content determination components, and set his/her display preferences;
- (2) *Query Elicitation, Validation and Repair*: lets the user specify his/her query and validate this query via automatically generated paraphrases of queries;
- (3) *Answer Formulation*: formulates the XML answer to the query returned by the ACD component in multimedia;
- (4) *Local Query Management*: exchanges data with the text analysis and ACD components and maintains the session history (see below);
- (5) *Reporting*: displays the session history.

More details about the processing are given below.

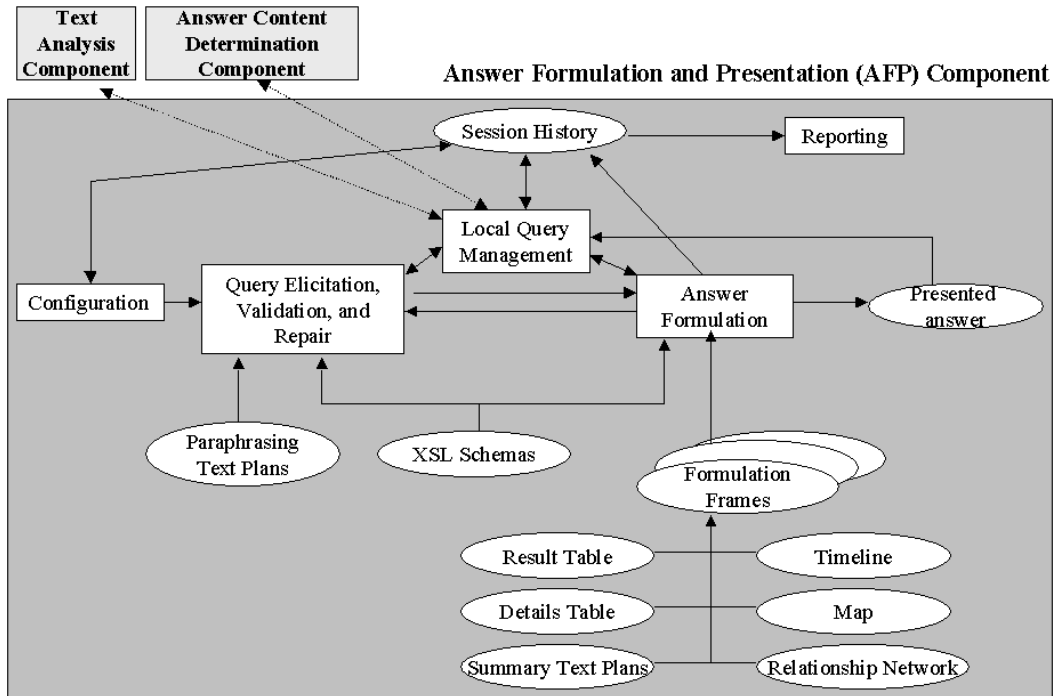


Figure 8: AFP component architecture

Figure 9 shows the XML specification of the natural language (NL) query illustrated in Figure 7 and three associated representations: (1) the TMR corresponding to this NL query; (2) a domain independent XML specification of the NL query; and (3) a domain specific XML specification of the NL query. This domain-specific XML schema was used in the first version of the system, which was built for the domain of people, their travel and contacts. Currently the system is being extended to cover arbitrary domains and the communication between components is performed using a domain-independent XML schema. However, the support for domain independent queries is still in progress, and the examples used in the remaining of this Section are based on the fully implemented domain specific XML specifications.

```

<?xml version="1.0" encoding="UTF-8" ?>
- <Query status="undefined">
  <Query-NL-Content text="Where did Tony Hall travel?" />
  <Data-TMR-Content text="{(HUMAN-27 (AGENT-OF (VALUE
TRAVEL-EVENT-28)) (HAS-NAME (VALUE 'TONY-HALL'))
(ROOT-WORDS (VALUE (TONY-HALL))) (WORD-NUM (VALUE 2))
(INSTANCE-OF (VALUE HUMAN))) (TRAVEL-EVENT-28 (THEME-
OF (VALUE REQUEST-INFO-29)) (AGENT (VALUE HUMAN-27))
(ROOT-WORDS (VALUE (TRAVEL))) (WORD-NUM (VALUE 3))
(INSTANCE-OF (VALUE TRAVEL-EVENT))) (REQUEST-INFO-29
(THEME (VALUE (DOT LOCATION) TRAVEL-EVENT-28)) (ROOT-
WORDS (VALUE (*QUESTION-MARK*))) (WORD-NUM (VALUE
4)) (INSTANCE-OF (VALUE REQUEST-INFO))))}" />
- <Query-Instance-Content>
  - <Instance>
    <Concept label="TRAVEL-EVENT" />
    - <Property>
      <Name label="LOCATION" />
      <Value label="requested" />
    </Property>
    - <Property>
      <Name label="AGENT" />
      - <Instance>
        <Concept label="HUMAN" />
        - <Property>
          <Name label="HAS-NAME" />
          <Text label="Tony Hall" />
        </Property>
      </Instance>
    </Property>
  </Instance>
</Query-Instance-Content>
- <Query-Structured-Content>
  - <Travel>
    - <Person>
      <Name label="Tony Hall" />
    </Person>
    - <Destination-Time-Period>
      - <Location>
        <Name wh="true" />
      </Location>
    </Destination-Time-Period>
  </Travel>
</Query-Structured-Content>
</Query>

```

Figure 9: NL query and associated representations

Multimedia presentation of answers in MOQA includes tables, maps, a time line, graphs for representing social networks and various types of contacts, and textual summaries of the information in tables. Textual summaries complement graphics and tables by highlighting salient facts and events and clustering data in meaningful ways. In particular, summaries group values in the table columns and, for large tables, list these groups in decreasing order. Text plans for summaries are domain-independent and written in XSL. For instance, the following summary is generated for the query illustrated in Figure 7:

The search found three travel events involving people with name similar to "Tony Hall". The destinations of these trips include Baghdad, North Korea, and Iraq. The time periods of these trips include 2000-04-17, ? - 2000-04-17, and 2000-04-16. All of these trips have no known point of departure.

This summary is generated using a domain-specific lexicon with lexical entries like those illustrated in Figure 10. Those entries are defined using an XML tag set similar to the one used to define the query and the answer content. Currently we are transitioning to a domain-independent lexicon which will support generating summaries for answers to arbitrary queries.

```

<xsl:when test="name() = 'Organization' and parent::Person">belonging to</xsl:when>
<xsl:when test="name() = 'Organization' and parent::Contact">between</xsl:when>
<xsl:when test="name() = 'Nationality' and parent::Person">with nationality from</xsl:when>
<xsl:when test="name() = 'Travel' and parent::Person">involved in</xsl:when>
<xsl:when test="name() = 'Person' and parent::Travel">involving</xsl:when>
<xsl:when test="name() = 'Person' and parent::Contact">involving</xsl:when>
<xsl:when test="name() = 'Person' and parent::Organization">involving</xsl:when>
<xsl:when test="name() = 'Time-Period' and parent::Contact" />
<xsl:when test="name() = 'From-Time-Point' and parent::Time-Period and not(//To-Time-Point)">since</xsl:when>
<xsl:when test="name() = 'From-Time-Point' and parent::Time-Period">between</xsl:when>
<xsl:when test="name() = 'To-Time-Point' and parent::Time-Period and not(//From-Time-Point)">before</xsl:when>
<xsl:when test="name() = 'To-Time-Point' and parent::Time-Period">and</xsl:when>
<xsl:when test="name() = 'Name' and parent::Location and ancestor::Location-Time-Period">reported to be in</xsl:when>
<xsl:when test="name() = 'Name' and parent::Location and ancestor::Contact">taking place in</xsl:when>
<xsl:when test="name() = 'Location' and parent::Source-Time-Period and ancestor::Travel">departing from</xsl:when>
<xsl:when test="name() = 'Location' and parent::Destination-Time-Period and ancestor::Travel">with destination</xsl:when>

```

Figure 10: Part of XML-based generation lexicon

Multimedia in presentations are interconnected by hyperlinks, providing the analyst with comprehensive support for intuitive browsing and follow-up queries. Multimedia functionality implemented for MOQA covers a large subset of the recommended information visualization functionality, best suited for a particular type of presented data (Hearst, 1999) and follows the intuitive concept of the information visualization mantra: overview, zoom & filter, details-on-demand (Shneiderman, 1997). Follow-up links are generated for domain concept instances and for source documents; selecting a domain concept instance generates a details page for this instance. Figure 11 shows the answer presentation for the query in Figure 7. The map is produced by the third party tool MapObjects from ESRI (<http://www.esri.com>) which is integrated into the system.

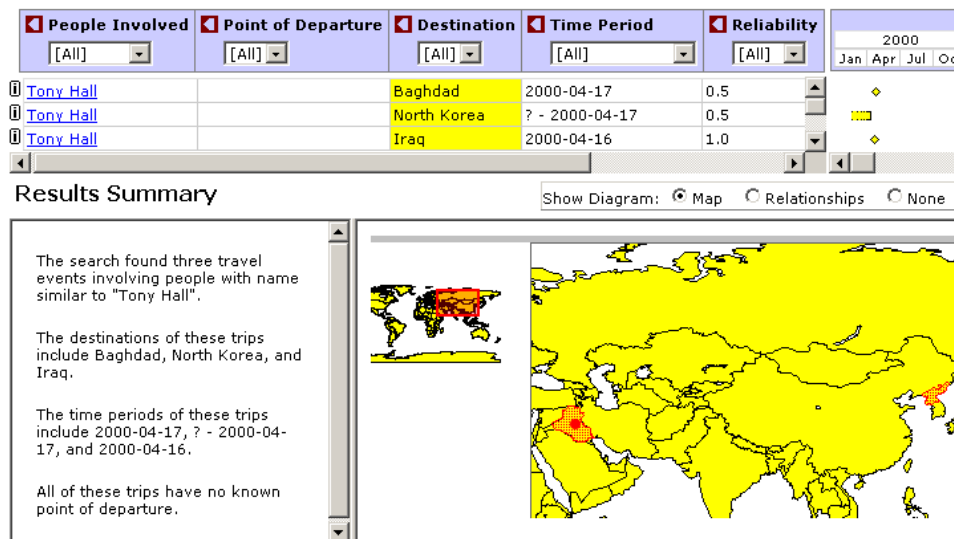


Figure 11: Answer presentation for query in Figure 7

```

<?xml version="1.0" encoding="UTF-8" ?>
- <Query-And-Result>
- <Query>
  - <Query-Structured-Content>
    - <Travel>
      - <Person>
        <Name label="Tony Hall" />
      </Person>
    - <Destination-Time-Period>
      - <Location>
        <Name wh="true" />
      </Location>
    </Destination-Time-Period>
  </Travel>
</Query-Structured-Content>
</Query>
- <table border="1" mainTopic="Travel" time-period-max-year="2000" time-period-min-year="2000">
- <thead>
  - <tr>
    - <th filter-value="[All]" sem="Person" topic="Person" wh="false">People Involved</th>
    <th filter-value="[All]" sem="Travel-Source-Location" topic="Travel-Source-Location" type="filter"
      wh="false">Point of Departure</th>
    <th filter-value="[All]" sem="Destination-Time-Period" topic="Destination-Time-Period" type="filter"
      wh="false">Destination</th>
    <th filter-value="[All]" sem="Time Period" topic="Time Period" type="filter" wh="false">Time Period</th>
    <th filter-value="[All]" sem="Reliability" topic="Reliability" type="filter" wh="false">Reliability</th>
  </tr>
</thead>
- <tbody>
  - <tr MaxRowSpan="1" id="0" report-url="" rowspanid="0" uid="10">
    - <td topic="Person">
      - <ul>
        <li sem="Person" uid="2">Tony Hall</li>
      </ul>
    </td>
    <td sem="Travel-Source-Location" topic="Travel-Source-Location" />
    <td sem="Destination" topic="Destination-Time-Period">Baghdad</td>
    <td sem="Time Period" topic="Time Period">2000-04-17</td>
    <td rowspan="1" sem="Reliability" topic="Reliability">0.5</td>
    <td type="start">2000-04-17</td>
    <td type="finish">2000-04-17</td>
  </tr>

```

Figure 12: XML query and formulation table extract

Figure 12 illustrates the XML representation used to generate the presentation shown in Figure 11. This representation contains the XML query and an answer formulation table produced from the XML answer content (which is passed to the presentation component by the answer determination component) and annotated using a semantic tag set similar to the one used in the query and answer content. This answer formulation table is used by different presentation sub-components to produce the HTML table, the text summary and the map shown in Figure 11.

By clicking on the hyperlink “Tony Hall” in the table the user specifies a follow-up query for more details about this person. Figure 12 shows the answer presentation to that second query. The details page for a person, in addition to personal data plus contacts and travel events involving the person, contains a graph that shows the person’s membership and contacts in organizations. Different icons are used for different types of contact: meetings, conversations, traveling together.

The contacts graph allows the user to trace the person’s contacts in a convenient way because all the names of people and organizations, as well as the depicted contact events in the graph are enabled with hyperlinks. The user can ask the system for an “extended graph” which includes contacts and memberships of all the people who had immediate contacts with the person in focus. The graph is implemented using the Tom Sawyer Software’s Graph Editor Toolkit (<http://www.tomsawyer.com>).

Each data item in the Fact Repository is justified by a pointer to the source document, coming both from English and foreign publications (Arabic, Farsi). These pointers are presented with icons associated with hyperlinks as illustrated in Figure 13. Selecting one of these icons displays the corresponding source document extract: Figure 13 shows the source document extract obtained by selecting one of Tony Hall's Arabic aliases displayed in this Figure.

MOQA Search Results: Travel Events

People Involved	Point of Departure	Destination	Time Period
Tony Hall		Baghdad	2000-04-17
Tony Hall		North Korea	? - 2000-04-17
Tony Hall		Iraq	2000-04-16

Person Details [Results Summary]

	Person Name <ul style="list-style-type: none"> Tony Hall
	Alias <ul style="list-style-type: none"> U.S. Rep. Tony Hall Congressman Hall تونى هال تونى هول
	Title <ul style="list-style-type: none"> U.S. Representative, D-Ohio United States Congressman
	Nationality <ul style="list-style-type: none"> United States of America United States of America
	Residence
	Organization <ul style="list-style-type: none"> U.S. House of Representatives United States Congress United States Congress United States Congress
	Travel <ul style="list-style-type: none"> to Baghdad, 2000-04-17 - 2000-04-17 to North Korea, ? - 2000-04-17 to Iraq, 2000-04-16 - 2000-04-16
	Contact <ul style="list-style-type: none"> meeting in Baghdad, 2000-04-17 - 2000 meeting in Baghdad, 2000-04-17 - 2000 meeting in Baghdad, ? - ?
	Source Extract

Source Document Extract (Arabic):

BBC Arabic News | News | برلمانى أمريكى يطلع على آثار الحصار

تم تحديث الموقع في الساعة 17:32 بتوقيت جرينتش برلمانى 00/04/17
 أمريكي يطلع على آثار الحصار

هول مع وزير الصحة العراقي توميد محنت: زيارة لاسياسية لاطلع عضو في مجلس النواب الأمريكي لنفسه على الأوضاع السائدة في مستشفيات للأطفال في العراق

بعد تحدث تونى هول إلى الأطفال المرضى وإلى نوابهم، كما اجتمع نوابي الصحة العراقي توميد محنت مدارك وذلك في نطاق زيارة لبتحاد يتم خلالها الأوصاح الانسانية في العراق

Diagram: A hierarchical diagram showing Tony Hall (U.S. Rep. Tony Hall) at the center, connected to the United States Congress (U.S. Congress) and U.S. House of Representatives. Below him is Omed Medhat Mubarak (اميد محنت مبارك).

Figure 13: Answer presentation for follow-up query and source document extract

Another feature of the presentation component is its management of the session history. From the outset of this effort intelligence analysts have pointed out to us the importance of capturing the session history. In their day-to-day work, report production plays a central role; hence they have a critical need to manage the information found, in a convenient way which would not disrupt the information gathering process. In the reported effort we have tried to make the first steps towards such a process. For each session, the presentation component produces the session history in HTML, capturing the sequence of queries, including clicks on hyperlinks, together with the presentations of answers in tables and text. The user can reorder or delete items to retain the most important information and export the history to the text editor. Figure 14 shows an extract of the session history resulting from the above queries. The display is done in a tree-like structure to provide contextual information.

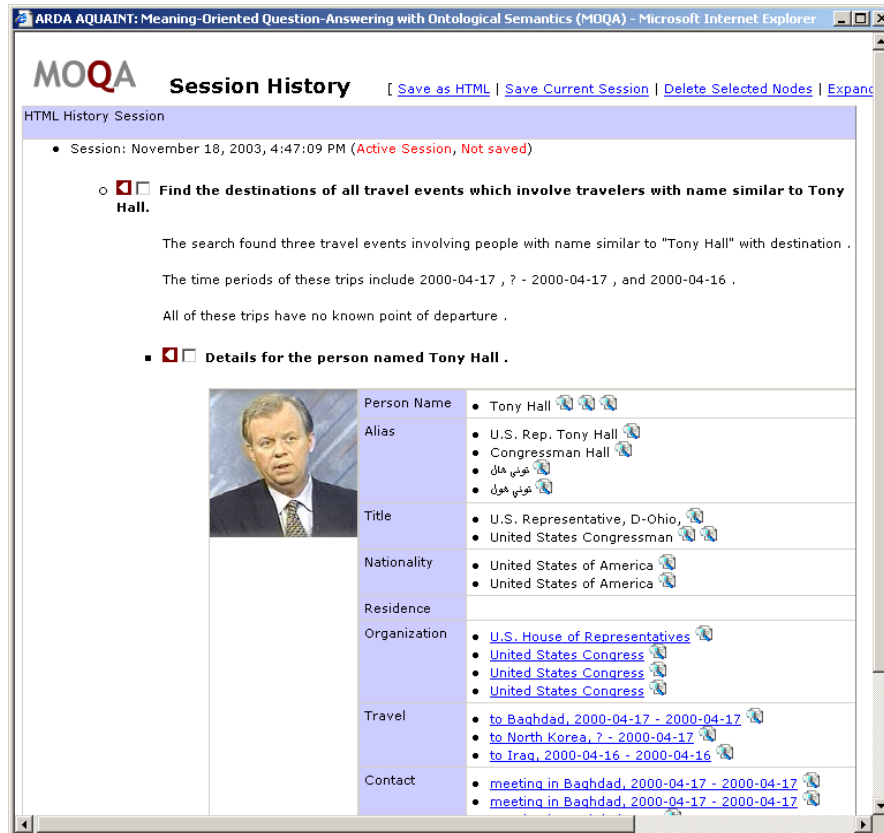


Figure 14: Extract of a session history

We have demonstrated the MOQA interface at AQUAINT workshops to intelligence analysts from the very beginning of the project, starting with mock-ups. The goal was to elicit as much feedback as we possibly could to be sure we are evolving the multimedia presentation design in the right direction. The feedback has been very positive; analysts have appreciated the benefits of comprehensive views of information presented in multimedia and the convenience of browsing the repository via follow-up queries using hyperlinks.

Current work includes making media display more flexible and better coordinated. For example, the level of detail on the map display (area, country, city) will coincide with the level of detail in the tabular or textual answer. We are also extending answer formulation and presentation to arbitrary queries, outside of the initial MOQA domain. The crucial task there is to adapt the analysis lexicon for text generation, to be able to generate the textual part of answers to arbitrary queries.

5 Future Work and Evaluation

The fully functioning system has only recently been integrated. Several major components have not been described in this paper. These include document management which allows a user to examine source materials. Arabic - English and Farsi - English translation engines are also part of the system. The analysis system has been modified to handle Arabic and Farsi syntax. Ontological lexicons for these languages are being extended to provide better coverage for the analysis process. The preprocessor used by the system has had morphologies for both Arabic and Farsi added. Date recognition and conversion modules are now being incorporated.

We are in the process of carrying out glass-box evaluations of the processing components of the system. A usability study is being carried out on the interface component. Once this is complete the interface will be improved based on the study results and it will then be subjected to a task based evaluation. This will require training analysts in the use of the system and then observing their use of the system during execution of the task.

Acknowledgements

This work has been supported by the ARDA AQUAINT program under contract number 2002*H167200*000.

Bibliographical References

- Beale, Stephen, Sergei Nirenburg and Kavi Mahesh. 1996. HUNTER-GATHERER: Three Search Techniques Integrated for Natural Language Semantics. In *Proceedings of AAAI-96*. Portland, Or.
- Cowie, Jim, Evgeny Ludovik, Hugo Molina-Salgado, and Sergei Nirenburg, and Svetlana Scheremetyeva. 2000. The week at a Glance - Cross-language Cross-document Information Extraction and Translation. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000)*, Saarbrücken, Germany.
- Hearst, Marti. 1999. User Interfaces and Visualization. In *Modern Information Retrieval*, Ricardo Baeza-Yates and Berthier Ribeiro-Neto (eds), Addison-Wesley Longman Publishing Company, 257–323.
- Korelsky, Tanya, Benoit Lavoie, and Ted Caldwell. 2003. Ontology-based Multimodal User Interface in MOQA. In *Proceedings of the AQUAINT 18-Month Workshop*, San Diego, California.
- Mahesh, Kavi, and Sergei Nirenburg. 1995. A Situated Ontology for NLP. *Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing*, International Joint Conference on Artificial Intelligence (IJCAI-95), Montreal, Canada.
- Mahesh, Kavi. 1996. Ontology Development for Machine Translation: Ideology and Methodology. *Memoranda in Computing and Cognitive Science, MCCS 96-292*. Computing Research Laboratory, New Mexico State University, Las Cruces, NM.
- Ng, Tobun, Michael Christel, Alexander Hauptmann, and Howard Wactlar. 2003. Collages as Dynamic Summaries of Mined Video Content for Intelligent Multimedia Knowledge Management. In *AAAI Spring Symposium Series on Intelligent Multimedia Knowledge Management*, Palo Alto, CA, March 24-26.
- Nirenburg, Sergei, and Victor Raskin. Forthcoming. *Ontological semantics*. The MIT Press. 350pp.
- Ben Shneiderman. 1997. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley, Reading, MA, third edition