

Automatic Ontology-Based Knowledge Extraction from Web Documents

Harith Alani, Sanghee Kim, David E. Millard, Mark J. Weal, Wendy Hall, Paul H. Lewis, and Nigel R. Shadbolt, University of Southampton

To bring the Semantic Web to life and provide advanced knowledge services, we need efficient ways to access and extract knowledge from Web documents. Although Web page annotations could facilitate such knowledge gathering, annotations are rare and will probably never be rich or detailed enough to cover all the knowledge

these documents contain. Manual annotation is impractical and unscalable, and automatic annotation tools remain largely undeveloped.

Specialized knowledge services therefore require tools that can search and extract specific knowledge directly from unstructured text on the Web, guided by an ontology that details what type of knowledge to harvest. An ontology uses concepts and relations to classify domain knowledge. Other researchers have used ontologies to support knowledge extraction,^{1,2} but few have explored their full potential in this domain.

The Artequakt project links a knowledge-extraction tool with an ontology to achieve continuous knowledge support and guide information extraction. The extraction tool searches online documents and extracts knowledge that matches the given classification structure. It provides this knowledge in a machine-readable format that will be automatically maintained in a knowledge base (KB). Knowledge extraction is further enhanced using a lexicon-based term expansion mechanism that provides extended ontology terminology.

Artequakt

Many information extraction (IE) systems can recognize entities in documents—for example, that “Rembrandt” is a person or “15 July 1606” is a date. However, such information isn’t very useful without the relation between these entities—that is, Rembrandt was born on 15 July 1606. Extracting such relations automatically lets us acquire more complete knowledge to populate the ontology. Artequakt attempts to identify entity relationships using ontol-

ogy relation declarations and lexical information. Storing information in a structured KB supports diverse knowledge services—for example, reconstructing the original source material to produce a dynamic presentation tailored to user needs.

Previous work in this area has highlighted the difficulties of maintaining a rhetorical structure across a dynamically assembled sequence.³ Most dynamic narrative techniques have used robust story schema such as the typical news program format (a sequence of atomic bulletins).⁴ Building a story-schema layer over an ontology lets us create dynamic stories within specific domains. By populating the ontology through automatic-knowledge-acquisition software, we let users construct those stories from the Web’s vast wealth of information.

Artequakt combines expertise and experience from three separate projects:

- *Artiste*: A European project to develop a distributed database of art images. This has recently been succeeded by Sculpteur, which will extend the database to 3D objects and integrate with the Semantic Web.
- *The Equator IRC*: An Engineering and Physical Sciences Research Council-funded Interdisciplinary Research Collaboration that uses narrative techniques to structure and present information.
- *The AKT IRC*: An EPSRC-funded IRC examining all aspects of the knowledge life cycle.

During Artequakt’s first stage, we created an ontology for the artists and paintings domain. We developed

Artequakt
automatically extracts
knowledge about
artists from the Web,
populates a knowledge
base, and uses it to
generate personalized
biographies.

and applied several IE tools and techniques that automatically populate the ontology with information extracts from online documents on the basis of the given ontology's representations and WordNet lexicons. The system stores information in a KB and analyzes it for duplicates. In the second stage, we developed narrative construction tools to query the KB through an ontology server, search and retrieve relevant facts or text, and generate a specific biography on demand.

Automatically generating tailored biographies offers two benefits. First, it can provide biographies of artists for whom available information is sparse and strewn across the Web. This might mean constructing text from basic factual information or combining text from several sources. Second, the method provides biographies tailored to a user's interests and requirements. To provide a project focus and a demonstration corpus of data, we chose the subject domain of impressionist artists and their paintings. The techniques developed do apply, however, to other domains.

Architecture

Artequakt's architecture (see Figure 1) comprises three key areas. First, knowledge extraction tools glean information items along with sentences and paragraphs from Web documents manually selected or obtained automatically using appropriate search engine technology. The tools pass the information fragments to the ontology server along with metadata derived from the ontology's vocabulary. Second, the ontology server stores and consolidates the information so that the biography generation tool can query the KB using an inference engine. Third, the Artequakt server takes user requests to generate narratives through a simple Web interface. The user might request a particular style of biography, such as a chronology or summary, or a specific focus such as the artist's style or body of work. The server uses story templates to render a narrative from the KB.

Ontology

An ontology conceptualizes a domain into a machine-readable format. For Artequakt, we created an ontology to represent the artist and artifacts domain. We implemented this ontology in Protégé (<http://protege.stanford.edu>), a graphical ontology editing tool that also stores the knowledge base. We constructed this ontology using sections of the CIDOC Conceptual Reference Model ontology

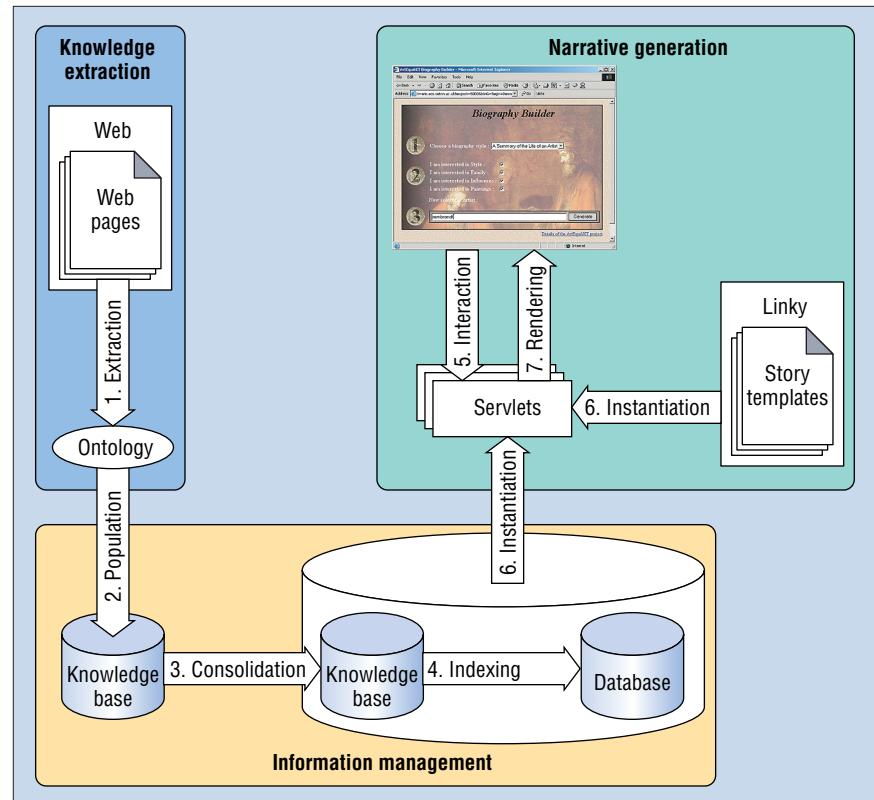


Figure 1. The Artequakt architecture.

(<http://cidoc.ics.forth.gr/index.html>). The ICOM/CIDOC Documentation Standards Group developed CRM to represent an ontology for cultural heritage information and gather existing disparate museum and cultural-heritage information sources into one coherent source.

The CRM ontology represents artifacts and their production, ownership, location, and other attributes. We modified this ontology for Artequakt and enriched it with additional classes and relationships to represent artists' personal information, family relations, relations with other artists, details of their work, and so on. The Artequakt ontology also stores text with its source URLs so that the system can later reorganize them using the ontology as a guide.

Knowledge extraction

Much Web information exists as natural language documents. A promising approach to accessing this knowledge uses IE to reduce the documents to tabular structures from which you retrieve document fragments as answers to queries. However, exploiting such techniques for practical purposes remains difficult because of the time and effort needed to man-

ually annotate many texts and the need for templates that stipulate information types to extract.⁵ Many IE systems rely on predefined templates and pattern-based extraction rules or machine learning techniques to identify certain entities in text documents. Web documents, however, use limitless vocabularies, structures, and composition styles to define approximately the same content, making it hard for any IE technique to cover all writing pattern variations. For example, even if two biographic documents have similar content, they might use different expressions or linguistic structures. More important, traditional IE systems lack the domain knowledge required to pick out relationships between the extracted entities.

These observations prompted us to use an ontology coupled with WordNet ([www.cogsci.princeton.edu/~wn](http://cogsci.princeton.edu/~wn)), a general-purpose lexical database, and GATE⁶ (General Architecture for Text Engineering, <http://gate.ac.uk>), an entity recognizer, to identify knowledge fragments consisting of not just entities but also the relations between them. We used automatic term expansion to widen the scope of text analysis and cover syntactic patterns that imprecisely matched our definitions.

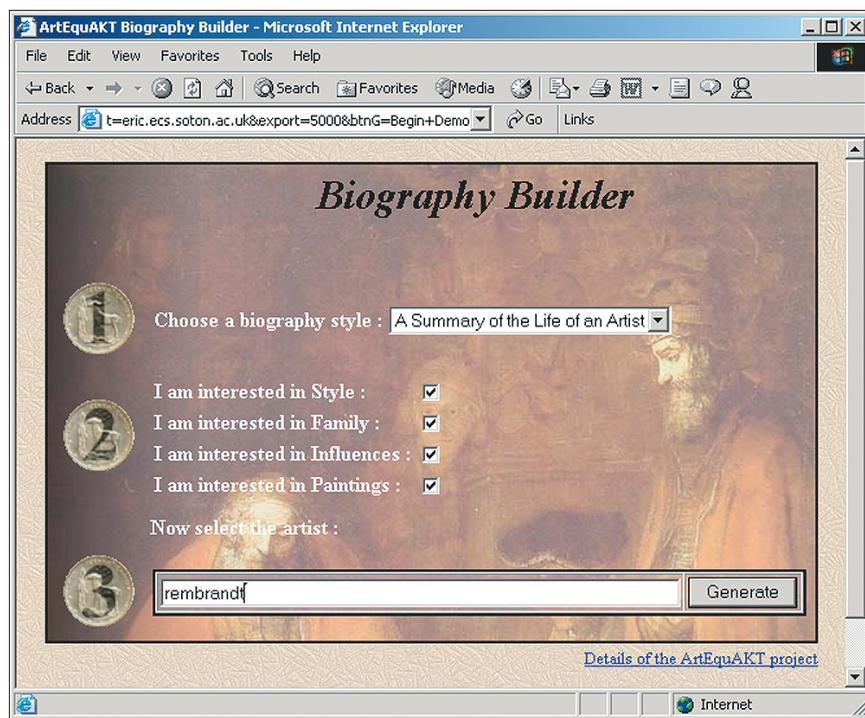


Figure 2. Artequakt's user interface.

Table 1. Filtered set of Web documents about Rembrandt.

Accepted sites	Rejected sites
www.ibiblio.org/wm/paint/auth/Rembrandt	www.rembrandt-s.com
www.mcs.csuhayward.edu/~malek/Rembran.html	www.rembrands.com
www.artchive.com/artchive/R/rembrandt.htmls	www.hotelrembrandt.co.uk

Figure 2 shows the prototype interface for searching for an artist and selecting the profile of interest. When a user enters an artist's name, such as *Rembrandt*, the system quickly searches the KB to see whether biographical knowledge for this artist exists.

If the given artist is new to the KB, the system deploys a script to search the Web using the artist's name as input. Because we needed a way to filter out irrelevant documents from the search engine results, we searched by example, using sample documents from trusted sites to measure similarity between query and search results. We obtained exemplars from the Web Museum site (www.ibiblio.org/wm/paint), which provides short artist biographies. Artequakt applies a vector similarity measure to compare the Web Museum page on a given artist with the search engine results, selecting only those with a similarity above a certain threshold.

Table 1 shows some of the accepted and rejected Web sites for *Rembrandt*. The accepted sites, filtered out of more than 60

URLs returned by the Yahoo and AltaVista search engines, provide information about Rembrandt's life and paintings. The rejected sites represent restaurants and hotels named "Rembrandt."

This filtering method doesn't always locate a good example document, however, especially when the query seeks a relatively new or unknown artist. We could expand queries with related terms to improve the results, but this probably won't remove the need for filtering. We're experimenting with techniques to measure document relevancy using term vectors constructed directly from the ontology terminology.

Once the extraction system has selected the documents, it divides them into paragraphs, then sentences. It analyzes each paragraph syntactically and semantically to identify relevant knowledge. The Apple Pie Parser (www.cs.nyu.edu/cs/projects/proteus/app) groups phrases that the syntactical analysis determines to be grammatically related. Using semantic analysis, the tool then locates

a sentence's main components (subject, verb, object, and so on) and identifies named entities (for example, *Rembrandt* is a person and *Leiden* is a place) using GATE and WordNet. The system also uses GATE to resolve anaphoric references (personal pronouns).

The following is a paragraph extracted from the first Web document selected:

Rembrandt Harmenszoon van Rijn was born on 15 July 1606 in Leiden, the Netherlands. His father was a miller who wanted the boy to follow a learned profession, but Rembrandt left the University of Leiden to study painting. His early work was devoted to showing the lines, light and shade, and color of the people he saw about him.

To extract binary relationships between a pair of entities, we need domain-specific semantic knowledge, which we can infer from the ontology and use to determine required and expected relations between the entities. At this stage, Artequakt submits a query to the ontology server to obtain such knowledge. It also uses three WordNet lexical chains (synonyms, hypernyms, and hyponyms) to reduce linguistic variations among relations defined in the ontology and the extracted text. These include synonyms, hypernyms (broader terms), and hyponyms (narrower terms).

For example, the system can match the concept of *depict* with *portray* (synonym) and *represent* (hypernym). Because a relation can have multiple WordNet entries (polysemous words), the mapping between a relation and an entry in WordNet accounts for syntactic and semantic clues in a sentence. For example, the *date_of_birth* relation maps onto the concept of *birth* which, according to WordNet, has four noun senses and one verb sense. The system selects the first noun sense because one of its hypernyms, *time period*, has *date* as a hyponym. The extracted synonyms for the verb sense include *give birth* and *bear*. By giving the IE process direct access to the ontology concepts and relations, our approach bypasses the need for predefined external templates.

Figure 3a shows the extraction results from the sentence "Rembrandt Harmenszoon van Rijn was born on 15 July 1606 in Leiden, the Netherlands." GATE and WordNet annotations highlight that *Rembrandt Harmenszoon van Rijn* is a person's name, *15 July 1606* is a date, and *Leiden* and *Netherlands* are locations. Artequakt extracts relations by the categorization result of the verb *bear*, which matches with two potential relations:

`date_of_birth` and `place_of_birth`. Because both relations are associated with *15 July 1606* and *Leiden* and *Netherlands*, respectively, this sentence generates two knowledge triples about Rembrandt (see Figure 3b).

Each extraction process yields an XML representation of the facts, paragraphs, sentences, and keywords identified in the selected documents. The extraction process terminates by sending the new XML files to the ontology server to be inserted into the KB.

Currently, Artequakt only launches knowledge acquisition when the KB contains no information about the queried artist. We plan to extend the system to let users submit a specific URL for analysis or request a new search if the biographies presented lack the required information. Also, users could periodically reinitiate the extraction process, searching for new Web sites to add to or update the KB's knowledge stores.

Automatic ontology population

To provide valuable, consistent ontology-based knowledge services, we must populate ontologies with many high-quality instantiations. By ontology population we mean the insertion of information into the KB following the ontology domain representation. Manual ontology population is labor intensive and time consuming. Some semiautomatic approaches create document annotations and store the results as assertions in an ontology. One method adds relationships automatically between instances only if they already exist in the KB; otherwise user intervention is needed.² Another framework applies user-driven ontology-based annotations,¹ reinforced with the IE learning tool Amilcare.⁷ However, this framework can't reliably identify relationships.

Artequakt moves toward fully automatic feeding of Web-based knowledge to the ontology. XML files, one per document, represent information extracted in Artequakt with respect to a given ontology using tags mapped directly from ontology class and relationship names. Figure 4 shows an example of this XML representation and how Artequakt asserts it in the ontology.

The ontology server we're experimenting with uses Java sockets and connects to the Artequakt KB through the Protégé application programming interface. A limited inference engine built on this server lets us query the KB to retrieve specific information—for example, all paragraphs that mention a specific artist's birth date, a painting's creator, or

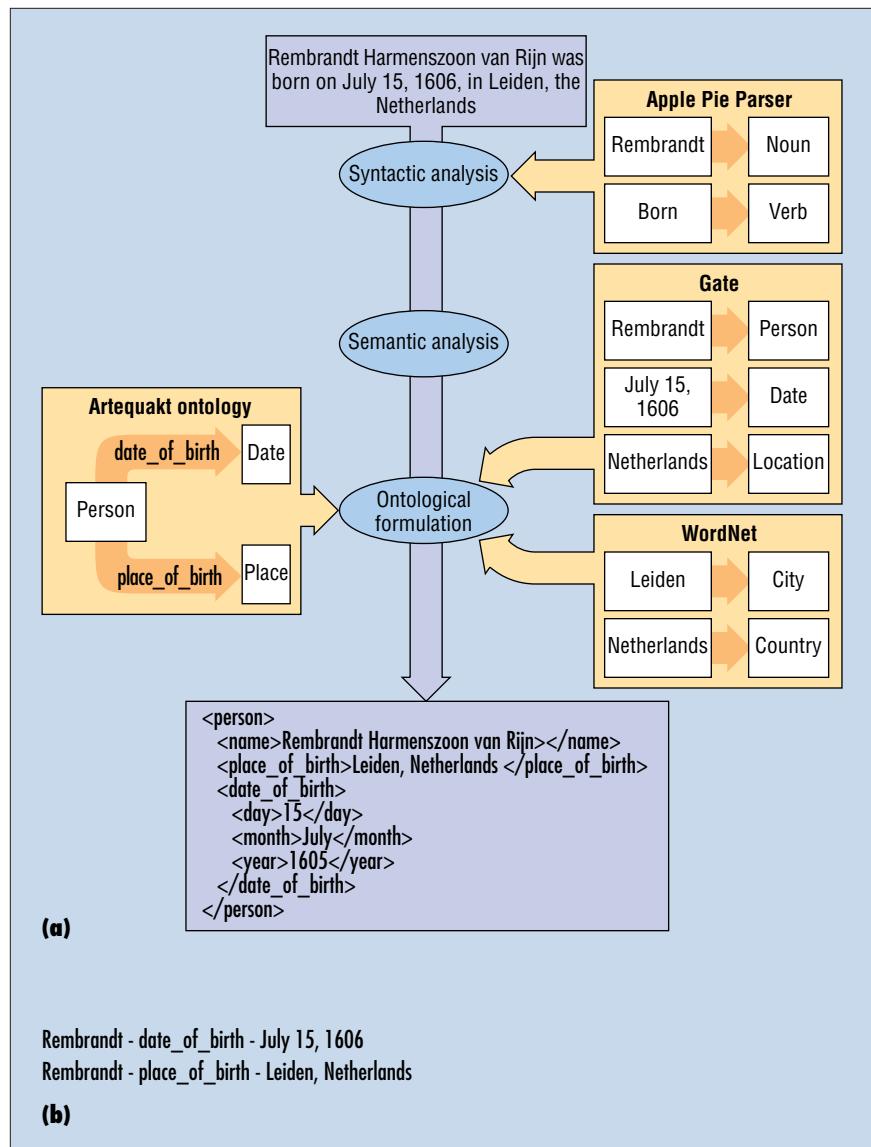


Figure 3. An example of knowledge extraction: (a) extraction results and (b) knowledge triples.

SOME KIND OF
FILL HERE?

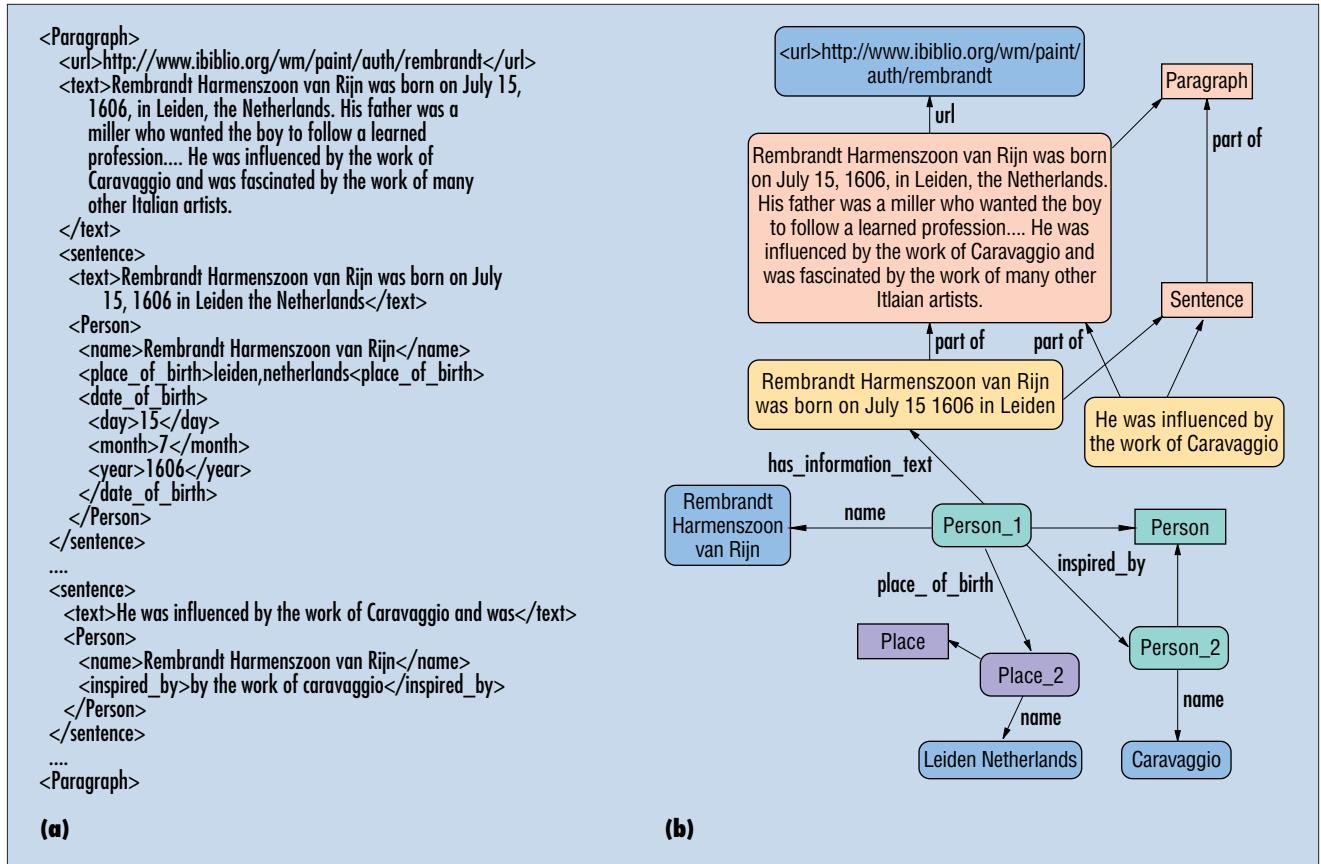


Figure 4. Automatic ontology population: (a) XML file of extracted information; (b) the corresponding instances and relationships in the ontology.

all available facts about an artist. Artequakt's ontology server sends some extracted knowledge to a relational database for quick access to frequently used information through SQL queries when generating biographies.

Narrative generation

Machines can effectively use structured ontologies to exchange information, but humans need a more intuitive interface. Storytelling offers a natural way to do this, and a wealth of critical and philosophical thought exists on using narrative to construct stories (in our case, biographies) from raw information. Narratologists frequently divide narration into *story*, the basic description of the fundamental events, and *discourse*, the techniques used to vary the presentation.

In Artequakt, the KB essentially contains our underlying story—it stores abstract pieces of information without any explicit ordering, just relationships. Strictly speaking, because the fragments consist mainly of text paragraphs harvested from Web pages, they might already contain discourse ele-

ments (focalization, tense information, and so on). To produce the eventual discourse (here, HTML pages), we must first arrange story elements into a sensible sequence and then render them into a text.

Biography templates

To arrange the story, Artequakt uses human-authored biography templates that contain data and KB queries. The templates are authored in the Fundamental Open Hypermedia Model and stored as XML files in the Auld Linky contextual structure server.⁸

Artequakt constructs any given biography from several substructures. The basic structure is the *sequence*; it represents a list of queries that must be instantiated and inserted into the biography in order. We author these queries using the terms defined in the ontology. Other structures allow more complex effects. For example, a *concept* contains several queries, any of which may be used at this point in the biography. A *level of detail* (LOD) resembles a concept but orders the queries on the basis of preference—that is,

use the highest-numbered query; if that's not possible, use the next highest, and so on. These structures can be nested (for example, a sequence of concepts).

Some queries retrieve paragraphs directly, whereas others query the KB for specific facts and construct sentences dynamically from the results. This can be useful for inferred facts (that therefore have no corresponding text sections) or when no paragraph fits the rest of the biography's literary form (for example, the biography is in the third person but all available paragraphs are in the first person).

Figure 5 shows a sample template that uses a sequence of four story fragments. These can be either database queries (resolved into an original paragraph) or KB queries (sentences that the system must construct). The fourth entry in the sequence is an XOR (exclusive OR) choice (implemented by a LOD structure): if the query finds no paragraph, the system constructs the sentence from the KB.

The templates also contain information on

appropriate biography structure elements for different contexts. When a user querying Auld Linky for the template specifies a context, the system prunes unnecessary template parts. For example, Figure 5 shows the third sequence item, artist style, marked as suitable for only those interested in this topic; otherwise the system ignores the entire third branch. This mechanism helps Artequakt tailor the biography's structure and contents to user interests.

Once the system retrieves the template from Auld Linky, it must instantiate the template by making each query in turn and then rendering the results into an HTML page for display. Figure 6 shows a biography of Rembrandt rendered from our sample template (where context specifies an interest in style). Taking the artist instance name from the KB generated the heading. Templates can include their own text—for example, the sample template's subheading *Summary Biography*. The first paragraph, extracted from the Web Museum site, concerns Rembrandt's birth. The second section uses text from different sites to describes Rembrandt's painting style.

The final sentence provides details of Rembrandt's death. No suitable paragraph existed in the database, so the system constructed a simple sentence from basic facts in the KB.

Results

We've had reasonable success producing readable biographies using Artequakt. In an initial experiment, we ran Artequakt to search and extract information from the Web on 10 impressionist artists. We set up the knowledge extraction tool to select the first 10 relevant pages for each artist, for a total of 100 Web pages to analyze. The extraction process identified over 800 paragraphs and thousands of entities and relations, many of which were duplicates. Once we fed the extracted knowledge to the KB and consolidated it, 280 unique person-entities remained, along with over 600 unique relations associating these people with various pieces of information such as their professional relationships to other artists, family members, where they lived or studied, and where and when they were born or died.

When generating the biographies, we used our main *summary* template that attempts to chronicle an artist's life and include both personal and professional information. This template contained 10 queries that the system could instantiate. We found that it typi-

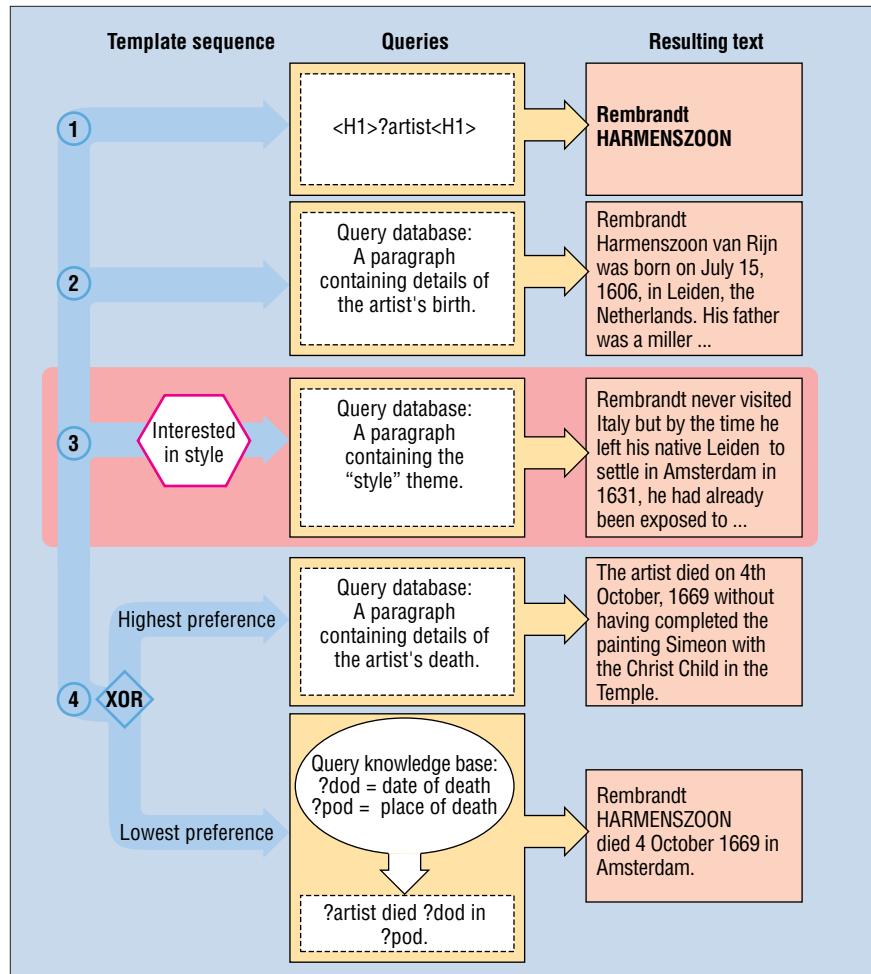


Figure 5. Templates contain queries that are resolved into the final text.

cally instantiated four of the possible 10 branches and included one paragraph for each branch.

Further challenges

The Artequakt project faces several challenges with ontology population and maintenance, knowledge extraction, and the generation of personalized artist biographies. User-driven ontology population tools must avoid duplicating information across documents and creating redundant annotations.⁹ Automatic ontology population approaches face similar issues, and, although they speed knowledge acquisition, precision and recall might decrease. Automatically populating an ontology from diverse, distributed Web resources also presents difficulties, particularly that of consolidating duplicate information that arises when extracting similar or overlapping information from different sources. We must tackle this problem to

maintain the referential integrity and results quality of any ontology-based knowledge service.¹⁰ We have made some headway on this with Artequakt.¹¹

The Artequakt extraction tool automatically retrieves metadata triplets (subject-relation-object) from the ontology to overcome the limitations of predefined fixed templates. This flexibility increases as the number of ontology class and relationship names decrease, thus avoiding, for example, compound or obscure names that might confuse concept identification and term expansion. Our approach covers semantic variations well, but coverage might decrease with limited term expansion. Also, the extraction tool cannot distinguish between ontology relations with synonymous names that connect the same classes. We avoid this by specifying appropriate synonyms for the classes and relations during ontology design to remove possible ambiguities.

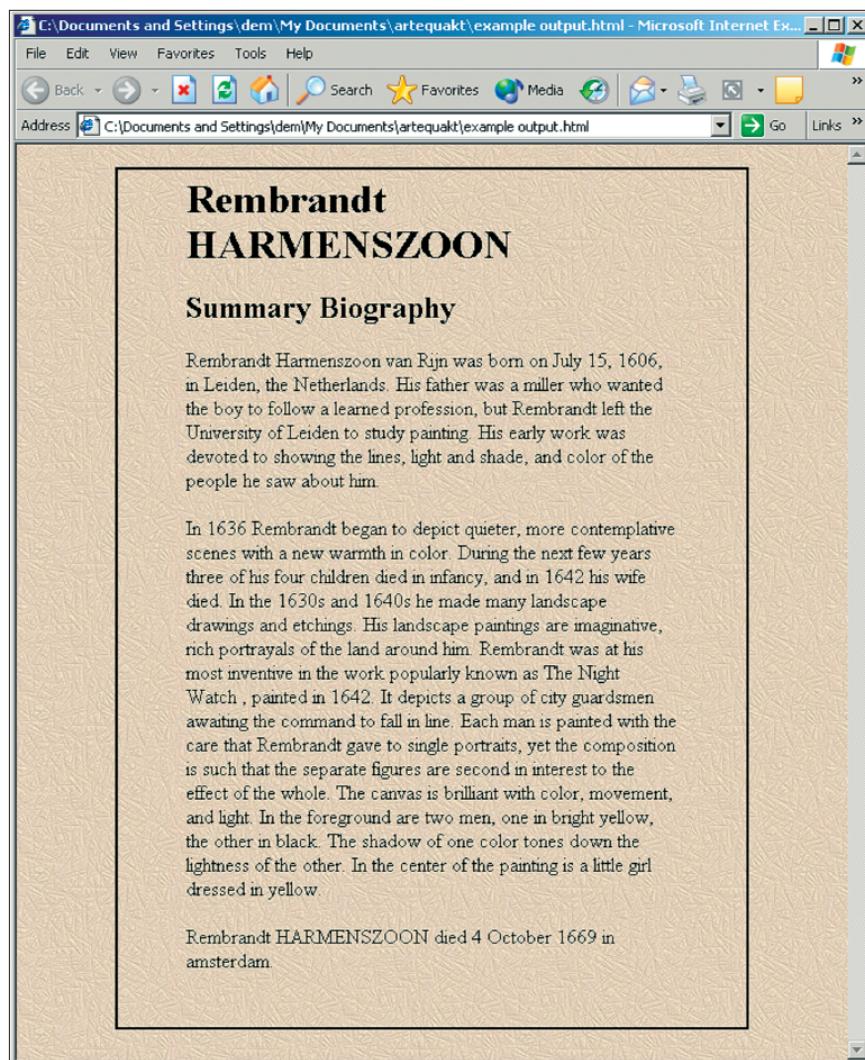


Figure 6. The final rendered biography of Rembrandt.

Specificity also presents challenges to automatic extraction. For example, while we can easily identify an entity as a person, it's harder to tell whether this person is a painter or a sculptor. We could infer that knowledge when we extract more facts about that person, such as information about paintings or sculptures produced. Likewise, generating a narrative can prove difficult if extracted sentences contain coreferences that must be resolved in the overall narrative context. For example, a sentence using the pronoun "he" makes sense only if the artist was the previous subject. Similarly, replacing artist name occurrences with appropriate pronouns might make the text more readable.

Finally, during narrative rendering, duplication might occur when selected sentences contain more information than the template

requests. A query for a sentence containing a birthdate might include information on the artist's parents, and a system that tracks this additional information could avoid later duplication. On the other hand, if we have many paragraphs about the artist's family life, determining how many of these to include in the biography will depend on such factors as the biography's desired length, interest level in that topic, and so on. Refining narrative generation techniques will improve biographies' quality and relevance to users.

We should, theoretically, be able to apply Artequakt to other domains with little technical change. For example, we can replace the current artist ontology with

an actor ontology that extracts specified information about actors. To improve entity recognition performance, domain-specific entities such as painting styles or films need specialized extraction rules that we can modify when the domain changes. We plan to further investigate and develop Artequakt's generic architecture. □

Acknowledgments

This work is part of a larger project funded in part by the EU Framework 5 Information Societies Technology project Sculpteur (IST-2001-35372), the Engineering and Physical Sciences Research Council's Interdisciplinary Research Collaborations project Equator (GR/N15986/01), and the EPSRC IRC project Advanced Knowledge Technologies (GR/N15764/01). We thank Hugh Glaser, Srinandan Dasmahapatra, and David De Roure for their contributions.

References

1. S. Handschuh, S. Staab, and F. Ciravegna, "SCREAM—Semi-Automatic Creation of Metadata," *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web—Proc. 13th Int'l Conf. Knowledge Eng. and Knowledge Management (EKAW 02)*, Lecture Notes in Artificial Intelligence, no. 2473, Springer-Verlag, Heidelberg, Germany, 2002, pp. 358–372.
2. M. Vargas-Vera et al., "Knowledge Extraction Using an Ontology-Based Annotation Tool," *Workshop on Knowledge Markup & Semantic Annotation*, ACM Press, 2001, pp. 5–12.
3. L. Rutledge et al., "Generating Presentation Constraints from Rhetorical Structure," *Proc. 11th ACM Conf. Hypertext and Hypermedia*, ACM Press, 2000, pp. 19–28.
4. B. Merialdo et al., "Automatic Construction of Personalized TV News Programs," *Proc. 7th ACM Conf. Multimedia*, ACM Press, 1999, pp. 323–331.
5. R. Yangarber and R. Grishman, "Machine Learning of Extraction Patterns from Unannotated Corpora: Position Statement," *Workshop on Machine Learning for Information Extraction* (14th Eur. Conf. Artificial Intelligence), IOS Press, 2000, pp. 76–83.
6. H. Cunningham et al., "GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications," *Proc. 40th Anniversary Meeting of the Association for Computational Linguistics (ACL 2002)*, ACL, 2002.
7. F. Ciravegna et al., "Timely and Non-Intrusive Active Document Annotation via Adaptive Information Extraction," *Workshop on Semantic Authoring, Annotation & Knowledge Markup* (15th Eur. Conf. Artificial Intelligence), IOS Press, 2002, pp. 7–13.

The Authors



Harith Alani is a research fellow in the Dept. of Electronics and Computer Science, University of Southampton, UK. He is currently working on the Advanced Knowledge Technologies project, focusing on ontology development and applications and community of practice support tools. His other research areas include ontology network analysis, knowledge services, semantic similarity measures, and spatial information retrieval. Contact him at the

Intelligence, Agents, Multimedia Research Group, Dept. of Electronics and Computer Science, Univ. of Southampton, Southampton SO17 1BJ, UK, ha@ecs.soton.ac.uk, www.ecs.soton.ac.uk/~ha.



Sanghee Kim is a research assistant with the Dept. of Electronics and Computer Science at the University of Southampton. Her research interests include text-mining systems focusing on Bayesian probability, symbolic learning, and natural language processing techniques. She is currently working on the Sculpteur (semantic and content-based multimedia exploitation for European benefit) project, which develops a sophisticated semantic layer for

distributed multimedia information management and a knowledge structure linking low- and high-level multimedia representations. She has a PhD in computer science from the University of Southampton. Contact her at the Intelligence, Agents, Multimedia Research Group, Dept. of Electronics and Computer Science, Univ. of Southampton, Southampton SO17 1BJ, UK, sk98r@ecs.soton.ac.uk, www.ecs.soton.ac.uk/~sk98r.



David Millard is a research fellow in the Intelligence, Agents, Multimedia Research Group at the University of Southampton, where he works on the Equator Interdisciplinary Research Collaboration (IRC), an interdisciplinary project exploring the boundaries between the real and digital worlds. His research interests include open and adaptive hypermedia, contextual information systems, and narrative support in digital media. Contact him at the

Intelligence, Agents, Multimedia Research Group, Dept. of Electronics and Computer Science, Univ. of Southampton, Southampton SO17 1BJ, UK, dem@ecs.soton.ac.uk, www.ecs.soton.ac.uk/~dem.



Mark Weal is a senior research fellow in the Intelligence, Agents, Multimedia Research Group at the University of Southampton. He works on the Equator Interdisciplinary Research Collaboration (IRC), focusing on the integration of physical and digital interaction. His research interests include hypermedia, orchestration tools, and narratology. Contact him at the Intelligence, Agents, Multimedia Research Group, Dept. of Electronics and Computer Science, Univ. of Southampton, Southampton SO17 1BJ, UK, mjw@ecs.soton.ac.uk, www.ecs.soton.ac.uk/~mjw.



Wendy Hall is Professor and Head of the Dept. of Electronics and Computer Science at the University of Southampton. She was the founding head of the Intelligence, Agents, Multimedia Research Group. Her research interests include multimedia, hypermedia and Web technologies, digital libraries, agent-based systems, and user interfaces. Her group developed the open hypermedia system, Microcosm, and she is a director of Active Navigation

Ltd, the company through which Microcosm and the subsequent Web-based link services were commercially exploited. Hall was awarded a CBE in the Queen's Birthday Honours list in June 2000 and became a Fellow of the Royal Academy of Engineering in the same year. She is president-elect of the British Computer Society. Contact her at the Intelligence, Agents, Multimedia Research Group, Dept. of Electronics and Computer Science, Univ. of Southampton, Southampton SO17 1BJ, UK, wh@ecs.soton.ac.uk, www.ecs.soton.ac.uk/~wh.



Paul Lewis is a senior lecturer in the Dept. of Electronics and Computer Science at the University of Southampton, UK. He is a member of the Intelligence, Agents, Multimedia Research Group and has been an investigator on numerous EPSRC and EU grants, most recently working on innovative content- and concept-based retrieval and navigation tools in multimedia environments. His research interests also include image and video analysis and

their applications to multimedia information handling. He is a member of the British Computer Society and a chartered Engineer. Contact him at the Intelligence, Agents, Multimedia Research Group, Dept. of Electronics and Computer Science, Univ. of Southampton, Southampton SO17 1BJ, UK, phl@ecs.soton.ac.uk, www.ecs.soton.ac.uk/~phl.



Nigel Shadbolt is Professor of Artificial Intelligence in the Dept. of Electronics and Computer Science at the University of Southampton. His research interests range from biologically inspired robotics to knowledge-intensive systems research. He is the director of the Advanced Knowledge Technologies (AKT) project, a UK-funded project in which five universities are pursuing basic and applied research in knowledge management technologies. He studied philosophy and psychology at the University of Newcastle upon Tyne, UK, and obtained a PhD in artificial intelligence from the University of Edinburgh, UK. He currently serves as editor in chief of *IEEE Intelligent Systems*. Contact him at the Intelligence, Agents, Multimedia Research Group, Dept. of Electronics and Computer Science, Univ. of Southampton, Southampton SO17 1BJ, UK, nrs@ecs.soton.ac.uk, www.ecs.soton.ac.uk/~nrs.

ied philosophy and psychology at the University of Newcastle upon Tyne, UK, and obtained a PhD in artificial intelligence from the University of Edinburgh, UK. He currently serves as editor in chief of *IEEE Intelligent Systems*. Contact him at the Intelligence, Agents, Multimedia Research Group, Dept. of Electronics and Computer Science, Univ. of Southampton, Southampton SO17 1BJ, UK, nrs@ecs.soton.ac.uk, www.ecs.soton.ac.uk/~nrs.

8. D.T. Michaelides et al., "Auld Leaky: A Contextual Open Hypermedia Link Server," *Hypermedia: Openness, Structural Awareness, and Adaptivity—Proc. OHS-7, SC-3, and AH-3*, Lecture Notes in Computer Science, no. 2266, Springer-Verlag, 2001, pp. 59–70.
9. S. Staab, A. Maedche, and S. Handschuh, "An Annotation Framework for the Semantic Web," *Proc. 1st Int'l Workshop MultiMedia Annotation*, 2001.
10. H. Alani et al., "Managing Reference: Ensuring Referential Integrity of Ontologies for the Semantic Web," *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web—Proc. 13th Int'l Conf. Knowledge Eng. and Knowledge Management (EKAW 02)*, Lecture Notes in Artificial Intelligence, no. 2473, Springer-Verlag, 2002, pp. 317–334.
11. S. Kim et al., "Artequakt: Generating Tailored Biographies with Automatically Annotated

Fragments from the Web," *Workshop Semantic Authoring, Annotation & Knowledge Markup* (15th Eur. Conf. Artificial Intelligence), IOS Press, pp. 1–6.

For further information on this or any other computing topic, please visit our Digital Library at <http://computer.org/publications/dlib>.