TIPSTER-COMPATIBLE PROJECTS AT SHEFFIELD

Hamish Cunningham, Kevin Humphreys, Robert Gaizauskas, Yorick Wilks
Department of Computer Science / Institute for Language, Speech and Hearing
University of Sheffield, UK

{hamish,kwh,robertg,yorick}@dcs.shef.ac.uk

Projects currently underway at Sheffield may be more appropriately described by the term *Language Engineering* than the well-established labels of *Natural Language Processing* or *Computational Linguistics*. This reflects an increased focus on viable applications of language technology, promoting a view of the software infrastructure as central to the development process. To this end, Sheffield has produced GATE – a TIPSTER-compatible General Architecture for Text Engineering – providing an environment in which a number of Sheffield projects are currently being developed.

GATE

GATE is an *architecture* in the sense that it provides a common infrastructure for building language engineering (LE) systems. It is also a *development environment* that provides aids for the construction, testing and evaluation of LE systems (and particularly for the reuse of existing components in new systems).

GATE presents researchers and developers with an environment in which they can easily use linguistic tools and databases from a user-friendly interface. Different processes, such as tagging or parsing, can be applied to documents or collections and the results compared and analysed. System modules, or combinations of modules which form complete systems, e.g. IE, IR or MT systems, can be configured and evaluated (e.g. using the Parseval tools), then reconfigured and reevaluated, providing a kind of edit/compile/test cycle for LE components.

GATE comprises three principal elements:

- the GATE Document Manager (GDM) — a TIPSTER-compatible database for storing information about texts;
- the GATE Graphical Interface (GGI) — an interface for launching processing tools on data and viewing and evaluating the results;
- a Collection of Reusable Objects for Language Engineering (CREOLE) — a collection of wrappers for algorithmic and data resources that interoperate with the database and interface.

GDM

GDM is based on the TIPSTER document manager. It provides a central repository or server that stores all the information an LE system generates about the texts it processes. All communication between the system components goes through GDM, thereby insulating parts from each other and providing a uniform API (applications programmer interface) for manipulating the data produced by the system.[1] Benefits of this approach include the ability to exploit the maturity and efficiency of database technology, easy modelling of blackboard-type distributed control regimes (of the type proposed by [2]), and reduced interdependence of components.

GGI

GGI is a graphical launchpad for LE subsystems, and provides various facilities for testing and viewing results, and interactively assembling LE components into different system configurations. As we built Sheffield's MUC-6 entry, LaSIE [6], it was often the case that we were unsure of the implications for system performance of using tagger X instead of tagger Y, or gazetteer A instead of pattern matcher B. In the GGI interface, substitution of components is a point-and-click operation. This facility supports hybrid systems, ease of upgrading and open systems-style module interchangeability.[2]

Figure 1 shows the launchpad for a MUC-6 IE system. Colours are used to indicate the status of each component with respect to the current document/collection: dark (red) components have already been run and their results are available for viewing; light (green) components have all their required inputs available and are ready to run, and grey (amber) components require a currently unavailable input before they can become runnable.

[1] Where very large data sets need passing between modules, other external databases can be employed if necessary.
[2] Note that delivered systems can use GDM and CREOLE without GGI.

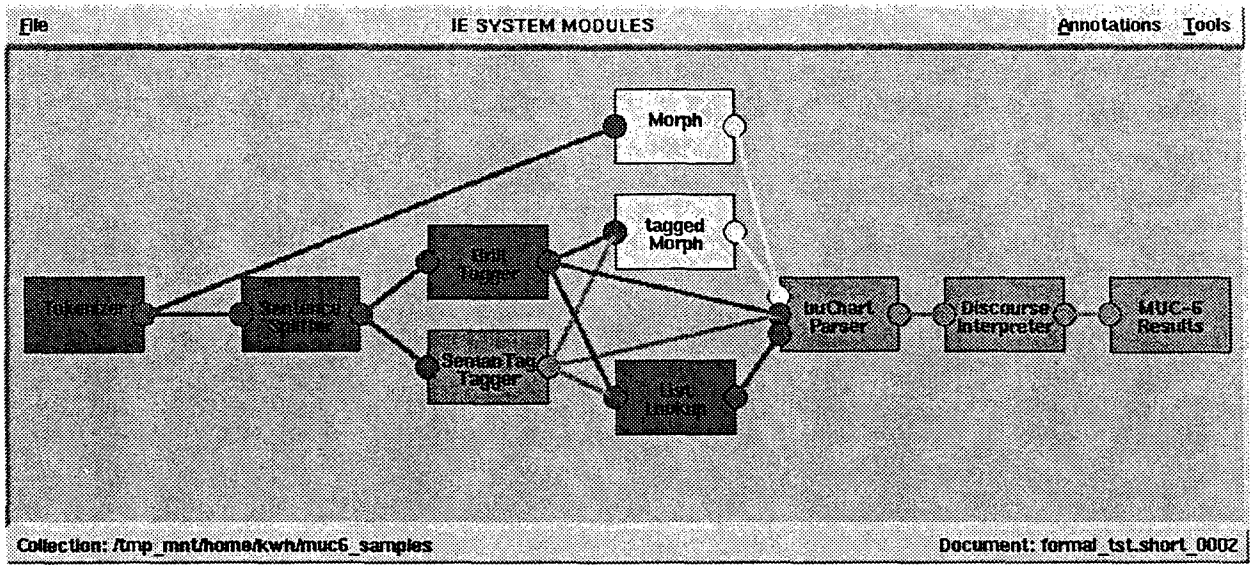

Figure 1: GGI – the GATE Graphical Interface

CREOLE

In a GATE-based system, the real work of processing texts – analysis, summarisation, translation, etc. – is done by CREOLE modules. Note that we use the terms *module* and *object* rather loosely to mean interfaces to resources which may be predominantly algorithmic or predominantly data, or a mixture of both. We exploit object-orientation for reasons of modularity, coupling and cohesion, fluency of modelling and ease of reuse (see e.g. [3]).

A CREOLE object may be a wrapper around a pre-existing LE module or database (e.g. a tagger or parser, a lexicon or n-gram index), or may be developed from scratch to conform to the TIPSTER architecture. In either case, the object provides a standardised API to the underlying resources, with access via GGI and I/O via GDM.

MULTEXT compatibility

MULTEXT [9, 5, 1] was an EU project to produce tools for multilingual corpus annotation and sample corpora marked-up according to the same standards used to drive the tool development. Annotation tools currently developed perform text segmentation, POS tagging, morphological analysis and parallel text alignment. The project has defined an architecture centred on a model of the data passed between the various phases of processing implemented by the tools. The MULTEXT architecture is based on a commitment to TEI-style [8] SGML [7] encoding of information about text. The TEI defines standard tag sets for a range of purposes including many relevant to LE systems. Tools in a MULTEXT system communicate via interfaces specified as SGML document type definitions (DTDs – essentially tag set descriptions).

The MULTEXT tools are currently in use and are recommended by the EU. GATE plans to support conversions between TIPSTER annotations and MULTEXT SGML markup, allowing data produced by the TIPSTER-compliant CREOLE modules to be written out in SGML for processing by MULTEXT tools, and the SGML results then converted back into native format. The extensive work done on SGML processing in MULTEXT could usefully fill a gap in the current TIPSTER model, in which SGML capability is not fully specified. Integration of the results of both projects would seem to be the best of both worlds, and we hope to achieve this in GATE.

TIPSTER in EU Projects

The Sheffield NLP group is moving all its research and development work to GATE, and therefore to the TIPSTER architecture. Several pan-European projects are involved, including:

- AVENTINUS – a research and development programme set up to build a multilingual information extraction system for multinational drug enforcement;

- ECRAN – a language engineering research project designed to offer filtered access to multilingual textual information delivered via TV screen or personal computer;

with collaborators such as:

- Thompson-CSF, Paris, France;
- Intelligent Financial Systems, Berlin, Germany;
- ILSP, Athens, Greece;
- GMS, Munich, Germany (formerly Siemens Sietec);
- four European drug enforcement agencies and Europol;
- The Universities of Fribourg, Germany; Gothenburg, Sweden; Ancona, Italy.

Summary

Working with GATE, the researcher will from the outset reuse existing components, the overhead for doing so being much lower than is conventionally the case – instead of learning new methods for each module reused, the common APIs of GDM and CREOLE mean that only one integration mechanism must be learned. And as increasing numbers of CREOLE modules and databases become available, through collaboration with sites able to provide single LE components, e.g. from the MULTEXT tools, we expect GATE, and therefore the TIPSTER architecture, to become widely used in the LE research community.

Availability of GATE

GATE is freely available for research purposes. The initial release is delivered with a CREOLE set comprising a complete MUC-compatible IE system called VIE – a Vanilla IE system. Some of the objects in VIE are freely available software (e.g. the Brill part-of-speech tagger [4]), while others are derived from Sheffield's MUC-6 entry, LaSIE [6].

Please contact `gate@dcs.shef.ac.uk` for details of hardware and software requirements, and licence arrangements.

Acknowledgements

The research reported here has been supported by grants from the U.K. Department of Trade and Industry (ref.: YAE/8/5/1002) and the Engineering and Physical Science Research Council (ref.: GR/K25267).

The authors would like to thank Malcolm Crawford of ILASH, University of Sheffield, for presenting a version of this paper at the April 1996 TIPSTER workshop, and for extensive comments during the preparation of this paper.

References

[1] Ballim A. "Abstract data types for MULTEXT tool I/O". LRE 62-050 Deliverable 1.2.1, 1995.

[2] Boitet C. and Seligman M. "The "whiteboard" architecture: A way to integrate heterogeneous components of NLP systems". In Proceedings of the 16th International Conference on Computational Linguistics (COLING-94). 1994.

[3] Booch G. Object-oriented Analysis and Design. Addison Wesley, 2nd edition, 1994.

[4] Brill E. "Some advances in transformation-based part of speech tagging". In Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94). 1994.

[5] Finch S., Thompson H. and McKelvie D. "Specification of tool shell with discussion of data and process architecture". LRE 62-050 Deliverable 1.2.2, 1995.

[6] Gaizauskas R.J., Wakao T., Humphreys K., Cunningham H. and Wilks Y. "University of Sheffield: Description of the LaSIE system as used for MUC-6". In Proceedings of the Sixth Message Understanding Conference (MUC-6). Morgan Kaufmann, 1995.

[7] Goldfarb C.F. The SGML Handbook. Clarendon Press, 1990.

[8] Sperberg-McQueen C.M. and Burnard L.. "Guidelines for electronic text encoding and interchange (TEI P3)". ACH, ACL, ALLC, 1994.

[9] Thompson H. "MULTEXT workpackage 2 milestone B deliverable overview". LRE 62-050 Deliverable 2, 1995.