Resources for multilingual text generation in three Slavic languages

John Bateman
Linguistics and Literature,
University of Bremen
bateman@uni-bremen.de

Elke Teich
Institute for Applied Linguistics,
Translation and Interpretation,
University of the Saarland
e.teich@mx.uni-saarland.de

Geert-Jan Kruijff
Ivana Kruijff-Korbayová
Institute of Formal and Applied Linguistics,
Charles University
{gj, korbay}@ufal.mff.cuni.cz

Serge Sharoff
Russian Research Institute for Artificial Inteligence
sharoff@aha.ru

Hana Skoumalová
Institute of Theoretical and Computational Linguistics,
Charles University
hana.skoumalova@ff.cuni.cz

Abstract

The paper discusses the methods followed to re-use a large-scale, broad-coverage English grammar for constructing similar scale grammars for Bulgarian, Czech and Russian for the fast prototyping of a multilingual generation system. We present (1) the theoretical and methodological basis for resource sharing across languages, (2) the use of a corpus-based contrastive register analysis, in particular, contrastive analysis of mood and agency. Because the study concerns reuse of the grammar of a language that is typologically quite different from the languages treated, the issues addressed in this paper appear relevant to a wider range of researchers in need of large-scale grammars for less-researched languages.

1. Introduction

In this paper, we report about our experience in re-using a large-scale, broad-coverage English grammar for building similar scale grammars for Bulgarian, Czech, and Russian. These grammars have been developed for a multilingual text generation system that aims at automatic generation of instructional texts starting from a specification of their meaning.

We used a naturally occurring corpus of software manuals, from which we adapted a small number of texts as target texts on which to test the incremental development of the generators. The target texts exhibit a wide range of grammatical phenomena and made it clear that large-scale generation grammars would be required for the system to be able to produce texts appropriate for the envisioned application. For languages like Czech, Bulgarian and Russian, however, there were at the time we started no large descriptive grammars available that could readily support natural language generation. The choice was therefore made to re-use an existing large-scale grammar for a different language, and to develop on the basis of that grammar three individual grammars for the Slavic languages. This methodology has been claimed effective in several previous developments (cf. (Rayner et al.1996; Bateman et al, 1999)) and avoids the necessity of building three large-scale grammars from scratch. Our approach can therefore be seen as additional evidence for (or against) this development strategy.

More specifically, the motivation for adopting this approach lies in three rather strong constraints:

1. New resources for *three* languages had to be developed. For none of them large-scale computational grammars aimed at generation exist.
2. Resource development had to be fast. The time frame for an integrated intermediate prototype was a year, where grammar development was only one component task.
3. The resources to be built should themselves be reusable, i.e., be general-language accounts of the languages besides covering the sublanguage of the CAD/CAM domain.

Given these constraints, it was necessary to adopt a methodology of resource development which supports resource sharing with existing accounts of languages other than the ones that are treated here, and to gear resource development in such a way that a fairly general linguistic account was achieved.

The grammar we have used as our basis is the Nigel grammar. This grammar has been under development since the early 1980s, when it was used within the Penman text generation project for English. It was mainly developed by Matthiessen on the foundation of work by Halliday (Mann and Matthiessen, 1985). Many people have since contributed to various parts of its coverage, making Nigel into a large-scale grammar of English that covers a very broad range of grammatical phenomena.

The most important reason for choosing Nigel as starting point was that it has been claimed to be particularly amenable for re-use and multilingual development. According to SFL theory, the organization of the grammar separates specifications of syntactic structures from a description of the communicative functions of those syntactic structures. (Bateman et al, 1991) argue that this functional description varies less across languages than does the syntactic description and, since it is precisely the latter functional component of the description that provides the overall organization of the grammar, the result is a grammatical account that can serve as a general guideline for the grammatical description of a wide range of languages without enforcing artificial uniformity. The development of generation grammars on this basis is now also supported by an extensive grammar development environment and multilingual sentence generator: Komet-Penman Multilingual system (KPML; (Bateman, 1997)) which, like Nigel, is available free of charge (see the URL at (Bateman, 1999)).

It is generally agreed that if an NLG system is to produce a range of texts which appear natural, then the underlying generation grammars need to be sufficiently broad in their

coverage. It is more and more acknowledged that methodologies and tools are needed to allow for a fast development of such grammars avoiding the necessity of building them from scratch. Methods for re-using existing resources have been proposed and claimed effective in several previous developments and re-use of grammatical resources for closely related languages has been successfully explored in several frameworks (e.g., the Core Language Engine: (Rayner et al, 1996), for DATR (Cahill and Gazdar, 1995)). However, it remains to be shown that employing these frameworks for languages that are typologically quite distinct is still as effective or feasible at all. The availability of a method that can deal with a wide variety of languages, less researched among them, is therefore doubly significant. The Slavic languages dealt with here are typologically quite distinct from English, employing very different modes of organization. Assessing the extent to which the functional basis of the Nigel grammar remains appropriate in this case is therefore of potential benefit for the development of natural language technology for a large number of still under-represented languages.

(1) Definiteness

En: Now specify the multiline.
Bg: Sega zadaite multiliniata.
Now specify-imp multiline-definite

(2) Politeness

En: Click on the OK button.
Ru: a. Nazhmite knopku OK.
Click-plur-imp button-acc OK
b. Nazhmi knopku OK.
Click-sing-imp button-acc OK

(3) Headings

En: To draw a polyline
Bg: Chertane na polilinija
drawing-nom of polyline
Cz: Nakreslení křivky
drawing-nom polyline-gen
Ru: Chtoby narisovatj poliliniju
in-order-to draw-infin polyline-acc

Figure 1: Examples of different realizations.

2. Method of resource development

In the present section we describe the theoretical principles of cross-linguistic resource sharing (Section 2.1) and the contrastive-linguistic corpus analysis (Section 2.2) carried out as a preparatory to implementation (Section 2.3). The approach advocated here is a combination of a *system-oriented* method of grammar development and an *instance-oriented* one. `System-oriented' means building up a computational resource with a view to the whole language system; `instance-oriented' means being guided by a register analysis.

2.1. Resource sharing

The idea of resource sharing across languages was originally applied to English, Chinese, Japanese, German and Dutch (Bateman et al, 1991; Teich et al, 1996) and is implemented in KPML. The theoretical basis of the concept of resource sharing in KPML is Systemic Functional Linguistics (SFL) (Halliday, 1985; Halliday and Matthiessen, 1999). SFL enjoys a number of properties that make it particularly suitable for modeling multilingual resources. Comparing any two languages, one will always detect commonalities and differences. The general representational constructs employed in SFL can be used as parameters along which such commonalities and differences can be described in a principled way. Systemic Functional Linguistics is a functional theory of language, in which the concept of function is reflected in three *metafunctions*. The ideational is concerned with states-of-affairs and their circumstances (processes and participants involved, such as Actor, Goal, Medium, and adjuncts of time, space etc). The interpersonal is to do with the role relations of speaker and hearer in a discourse (grammatical mood and modality). The textual represents the patterns with which cohesive and coherent texts are created. Other organizing principles are *stratification, axiality, delicacy* and *ranking*. The strata distinguished are lexicogrammar, semantics, and context. Linguistic description, at each stratum, has two aspects, one representing linguistic *systems* (paradigmatic axis), the other the *structural realizations* of these systems (syntagmatic axis). The means used for the representation of the paradigmatic axis is the system network. Formally, a system network is like type hierarchy supporting multiple inheritance. For instance, classification of the English mood starts with the types 'indicative' vs. 'imperative'. Semantically, it corresponds to the opposition of speech acts referring to exchange of information vs. issuing commands. Furthermore, a system network adheres to a partial ordering which is called delicacy, denoting the *type-subtype* relation. For example, taking again the example of an English mood description, the subtypes of 'declarative' and 'interrogative' are more delicate than 'indicative'. The types (or: features) have attributes expressing constraints on syntagmatic realization. The type 'indicative', for example, has the attributes '+Subject', and '+Finite', while 'declarative' has a constraint that Subject is located before Finite. Thus, the functional, systemic aspects are separate from the formal, syntagmatic ones. In the formal notation, this is expressed as:

MOOD-TYPE: (independent-clause-simplex) →
[indicative] (+Subject), (+Finite),
[imperative] (+Non-finite).
INDICATIVE-TYPE: (indicative) →
[declarative] (Subject^Finite),
[interrogative]

Figure 2: Mood specification in English.

System networks are set up for different *ranks* (clause, groups/phrases, words, morphemes), where the sets of features holding for each rank are pairwise disjoint. Stratification, metafunction, axiality, delicacy and rank make up the basic inventory of representational categories used in SFL. Applying these categories to cross-linguistic description, the following observations can be made.

1. Languages tend to show more similarities on the more abstract strata of linguistic organization than on the less abtract ones (i.e., they tend to express similar meanings, but cast them in different grammatical terms). For example, languages express the semantic category of speech function, i.e., make statements, ask questions, give orders. However, the grammatical

potential realizing these semantic categories may be quite different across languages.

2. Languages tend to be similar on the paradigmatic axis and less similar in terms of syntagmatic realization. A case in point is the expression of definiteness in the nominal group in Bulgarian and English (example (1) in Figure 1). In English, definiteness is expressed by the definite determiner and in Bulgarian by suffixation of the first element in the nominal group.
3. Systems of low delicacy tend to be similar across languages, and systems of higher delicacy tend to be dissimilar. An example is mood in English and Russian (example (2) in Figure 1). The basic mood options of English also apply to Russian. However, regarding imperatives, Russian further distinguishes according to politeness: there is one verb form for polite imperatives (2a) and one for nonpolite imperatives (2b). That is, Russian has more delicate options for imperative than English.
4. There may be different preferences in different languages concerning the grammatical rank at which a particular meaning is expressed. A case in point is the signalling of referents in a discourse as identifiable or nonidentifiable. While in Bulgarian and English identifiability is marked in the nominal group by determiners, in Czech and Russian it is typically marked by word order in the clause.
5. Different languages may distribute functional responsibilities differently across metafunctions. The same formal means may serve different functions. For example, the passive construction is possible in Bulgarian, Czech, Russian and English. In English the main function of the passive is to change Theme-Rheme organization, so passive carries a textual function. Czech and Russian, on the other hand, can simply change Theme-Rheme by changing word order and the passive thus carries less of a textual load.

KPML implements these five types of cross-linguistic variation and goes thus well beyond comparable approaches in multilingual computational linguistics. First, it is possible to represent both commonalities and differences between languages. In most computational approaches to multilinguality, one or the other is taken as central. For instance, most machine translation approaches would either enforce commonality at one particular level (interlingua-based approaches), or assume divergence as the basic principle (transfer-based approaches). Second, with the SFL model as a basis, we achieve a multidimensional model of cross-linguistic variation, in which that variation can be defined along the parameters presented above. The types of cross-linguistic divergences commonly considered are much less refined and do not come as close to covering the observable cross-linguistic phenomena, such as the ones presented above. [1]

2.2. Contrastive-linguistic register analysis

The second pillar in the approach to a fast prototyping of grammatical resources for generation pursued here is working from instances, i.e., from a corpus of instructional texts. The goal was to identify the repertoire of most common lexico-grammatical constructions in the three languages in this register.

Considering the results of previous works on lexical and syntactical closure of sublanguages (e.g., (Kittredge, 1987)), a small corpus appeared sufficient. The corpus (consisting of a set of parallel texts from the CAD/CAM domain and a set of non-parallel instructional texts from other domains) was analyzed following (Hartley & Paris, 1996), tagging for the type of text unit (e.g., procedure, description), plan element (e.g., goal, side-effect, step), and a set of SFL lexico-grammatical features.

The analysis revealed the following tendencies of similarities and differences between the linguistic resources needed for each language:

- Top-level goals for simple procedures are typically expressed by nominalizations in Bulgarian and Czech and by non-finite clauses in Russian.
- The actions to be performed are realized mostly by imperatives in Russian and Bulgarian. In Czech indicative is also used (25%).
- The actions are mainly material-directed processes, mental and relational processes also occur.
- The similarity of the three languages is very high with respect to the features positive, active and non-modal (more than 90%). Differences arise mainly in finiteness: Bulgarian and Czech tend to employ mostly finite forms, while Russian employs 32% infinite forms. In all three languages, the finite forms are predominantly in second person plural, which corresponds to the polite imperative form.
- The Czech corpus shows a strong preference for simple clauses (65%). In the Bulgarian and Russian corpora the majority of clauses are complex, mainly hypotactic, while in Czech paratactic and hypotactic relations are evenly distributed.

The corpus analysis identified the repertoire of the most common constructions in this register in each language and revealed a large overlap across the three languages. These two factors were made use of in the resource implementation. Below we describe our experience using incremental implementation of Mood and Agency features as an example.

2.3. Implementation

The implementation of grammars for Bulgarian, Czech and Russian proceeded in two stages.

Stage 1 built upon the grammar of English. The register analysis identified the grammatical systems to be dealt to cover the grammatical phenomena of the domain. Resource development was distributed according to language. Implementation by resource sharing proceeds in cycles of "copying" and "adapting" grammatical systems. For instance, the system accounting for types of predicates in systemic functional grammars is transitivity. For English, this system distinguishes between material, mental, relational and verbal predicates, each with a number of subtypes and particular syntagmatic realizational constraints. In the first step of implementation, the grammar developer has to assess whether the basic systemic options apply to the new language. In a lot of cases, this will be true. The system can then be taken over as it is. Typically, what has to be changed are the attributes of the system's types, i.e. the constraints on structural realization. For instance, the attributes of English prepositional phrase is

1 Cf. e.g., the kinds of translational divergences proposed by (Dorr, 1994).

that the minor process is lexified by a certain preposition. For Russian and Czech we have to add the constraint that the following noun phrase occurs in certain morphological case required by the preposition; for Bulgarian, we have to add the constraint that the following noun phrase can only have the short article, if it is determined. The larger part of implementation comes down to these two actions, 'copy' and 'adapt', which are supported by the KPML environment, and the new resource can be written out as the grammar of the new language, inheriting from English. The fact that the larger part of grammar implementation for a new language consists of these two actions supports the hypothesis that languages are more similar on the paradigmatic axis than in terms of the realization of paradigmatic options. These actions are repeated going depth-first, i.e., going further in delicacy, until the point of coverage required by the sublanguage is reached.

Stage 2 of grammar implementation was organized according to linguistic phenomena rather than languages. Given that at least Czech and Russian are typologically closer to each other than to English, there are common linguistic constructions that are better to treat copying from Russian to Czech or vice versa than copying from English. At this stage, the grammars become more adequate descriptions of the individual languages and stop inheriting from English. Phenomena that were not covered adequately in the English grammar because they are not so prominent in the language are treated focally in this stage. Cases in point are syntactic agreement, which is rather complex in all of Bulgarian, Czech and Russian, aspect and word order, which is primarily pragmatically determined in Czech and Russian. Also, in this stage of development, the language-specific requirements of the sublanguage are treated.

Let's consider two examples: implementation of mood and agency in Slavic languages. The functional classification of the Mood features in Nigel is inherited in target grammars as well. The main difference is in realization. In English commands are realized by imperative sentences with a nonfinite form of the verb (infinitive), whereas in Slavic languages imperatives are expressed by a finite verb form (imperative) with a more delicate choice of politeness (cf. example 2 in Figure 1). Here, according to the hypothesis that languages tend to be similar in terms of less delicate system and different in terms of more delicate ones, it becomes less likely that systems can still be shared.

MOOD-TYPE: (independent-clause-simplex) →
    [indicative] (+Subject), (+Finite),
    [imperative] (+Finite), (Finite:::imperative)
INDICATIVE-TYPE: (indicative) →
    [declarative],
    [interrogative].
POLITENESS-TYPE: (imperative) →
    [polite] (Finite:::plural),
    [non-polite] (Finite:::singular). [2]

Figure 3: Mood specification shared across Bulgarian, Czech and Russian.

2 In the formal notation, a three colons mark expresses constraints on morphological realization, while a two colons mark expresses constraints on semantic classification of a lexical item.

In the genre of written software instructions, there are typically no interrogative clauses. Therefore, we consider the branch of declarative clauses, which express either side effects of user's actions or user's actions themselves in the so-called non-personal style, which does not directly expresses instructions for the user, but presents a description of an artificial world of software with its own laws and possibilities for actions. The personal and non-personal versions of the same ideational content are shown in Figure 4.

This type of expressions is modeled using the notion of diathesis, which, in the SFG perspective, is described as the relation between transitivity functions (participant roles), agency functions (Agent, Medium) and syntactic relations. In English, the respective diathesis alternation is the transitive vs. ergative construction.

(1) Personal style

En: Under Name, enter the name of the style.
Bg: В полето Name въведете име на стила.
    In field Name, enter-imper name of style.
Cz: Pod Jméno zadejte název stylu.
    Under Name, enter-imper name style-gen.
Ru: В поле Name задайте имя стиля.
    In field Name enter-imper name style-gen.

(2) Non-personal style

En: Under Name, the name of the style is entered.
Bg: В полето Name се въвежда име на стила.
    In field Name, refl enter-ind name of style.
Cz: Pod Jméno se zadá název stylu.
    Under Name, refl enter-ind name style-gen.
Ru: В поле Name задается имя стиля.
    In field Name enter-refl-ind name style-gen.

Figure 4: Style examples.

In English the system of diathesis reflects two possible patterns for realization of an Agent + Medium configuration: the transitive construction, in which both participants are expressed, and the middle or ergative construction, which realizes only Medium (Halliday, 1985: 144ff), for example: *The user opens the window – The window opens.*

AGENCY: (transitivity-unit) →
    [middle] (Process::middle-verb),
    [effective] (Process::effective-verb).
AGENTIVITY: (effective) →
    [nonagentive], (Medium/Subject)
    [agentive].
EFFECTIVE-VOICE: (agentive) →
    [operative], (Medium/Directcomplement)
    [receptive] (Medium/Subject),
        (+Agentmarker::by), (Agentmarker^ Agent)

Here, the alternation is realized by putting the second argument of the transitive variant into the Subject position in the ergative variant. While the grammars of Slavic languages have the middle variant as well, its construction needs more complex morpho-syntactic means. This is formally realized in Russian morphologically by means of a reflexive verb, and in Czech and Bulgarian by insertion of a reflexive particle (a clitic). The particular diathesis realization of the non-personal style involves alteration of the verb aspect (the perfective in the personal style; the imperfective in the non-personal one) and the word order:

the subject (*name* in example (2) Figure 4) is put after the verb. Thus, the systems, for example, for Russian include:

AGENCY: (transitivity-unit) →
[middle] (Process::intransitive-verb),
[middle-transitive] (Process::middle-verb,
Process:::reflexive-form),
[effective] (Process::transitive-verb).

AGENTIVITY: (effective) →
[nonagentive], (Medium/Subject)
[agentive].

EFFECTIVE-VOICE: (agentive) →
[operative], (Medium/Directcomplement)
[receptive] (Medium/Subject), (Agent:::Instr-case)

MEDIO-PASSIVE-VOICE:
(nonagentive, middle-transitive) →
[medio-passive-process] (Finite^Subject).

3. Conclusions

The approach advocated here combines two methods of computational grammar implementation for the purpose of a fast prototyping of computational linguistic resources. The approach builds upon one particular interpretation of re-usable resource, which is the idea of *resource sharing across languages*. We called this a *system-oriented* method of creating computational resources for new languages. Complementary to this method, we have been guided by a corpus-based contrastive register analysis of Bulgarian, Czech and Russian instructional texts, i.e., an *instance-oriented* method of computational resource development. Proceeding this way, basic general-language grammars and sublanguage grammars for CAD/CAM instructional texts have been created. Due to the system-orientation, these grammars are less restricted than sublanguage grammars of a particular domain; and due to the instance-orientation, these grammars are adequate for the domain at hand as well. Also, this two-tiered resource development method allowed us to practice a contrastive-linguistic approach and a maximal sharing of efforts, not only working from English, but also sharing among the three Slavic languages.

Acknowledgements

The authors thank all the members of the AGILE team, especially to Donia Scott, Tony Hartley, Jiří Hana and Kamenka Staykova. The work reported in this paper has been supported by the INCO-COPERNICUS Programme of the European Commission, Grant No. PL961104.

References

Bateman J.A., Matthiessen C.M.I.M., Nanri K, Zeng L. (1991) The re-use of linguistic resources across languages in multilingual generation components. In Proceedings of the 1991 International Joint Conference on Artificial Intelligence, Sydney, Australia, volume 2, (pp. 966—971). Morgan Kaufmann Publishers.

Bateman J.A. (1997) Enabling technology for multilingual natural language generation: the KPML development environment. *Journal of Natural Language Engineering,* 3(1):15—55.

Bateman J.A. (1999) The KPML multilingual natural language generation system, development environment and tools. At: http://purl.org/net/kpml

Bateman J.A., Matthiessen C.M.I.M., Zeng L.. (1999) Multilingual natural language generation for multilingual software: a functional linguistic approach. *Applied Artificial Intelligence,* 13:607—639.

Cahill L.J. & Gazdar G. (1995) Multilingual lexicons for related languages. In Proceedings of the 2nd DTI Language Engineering Conference (pp. 169—176), London, Department of Trade and Industry.

Dorr B.J. (1994). Machine translation divergences: a formal description and proposed solution. *Computational Linguistics,* 20(4):597—634.

Halliday M.A.K. (1985). *An Introduction to Functional Grammar.* Edward Arnold, London.

Halliday M.A.K. & Matthiessen C.M.I.M. (1999) *Construing experience through meaning: a language-based approach to cognition.* Cassell Academic, London and New Work.

Hartley A., & Paris C., (1996) Two sources of control over the generation of software instructions. Information Technology Institute Technical Report Series, ITRI-96-02. Also published in Proc. ACL, Santa Cruz, California, June 1996. (pp 192—199).

Kittredge R. (1987). The significance of sublanguage for automatic translation. In: Nirenburg S. (ed.) *Machine Translation: Theoretical and Methodological Issues.*

Mann W. & Matthiessen C.M.I.M.. (1985) A demonstration of the Nigel text generation computer program. In James D. Benson and William S. Greaves, editors, *Systemic Perspectives on Discourse,* vol 1. Ablex, Norwood, N.J.

Netter, K. et al. (1998) DiET - Diagnostic and Evaluation Tools for natural language processing applications. In: Proceedings of the first international conference on Language Resources and Evaluation (pp. 573—579), Granada.

Rayner M., Carter D., Bouillon P. (1996). Adapting the Core Language Engine to French and Spanish. In: Proceedings of NLP-IA-96, Moncton, New Brunswick.

Teich E., Degand L., Bateman J.A. (1996) Multilingual textuality: Experiences from multilingual text generation. In: G. Adorni and M. Zock, editors, *Trends in Natural Language Generation: an artificial intelligence perspective,* No. 1036 in Lecture Notes in Artificial Intelligence, pages 331—349. Springer-Verlag, Berlin, New York.

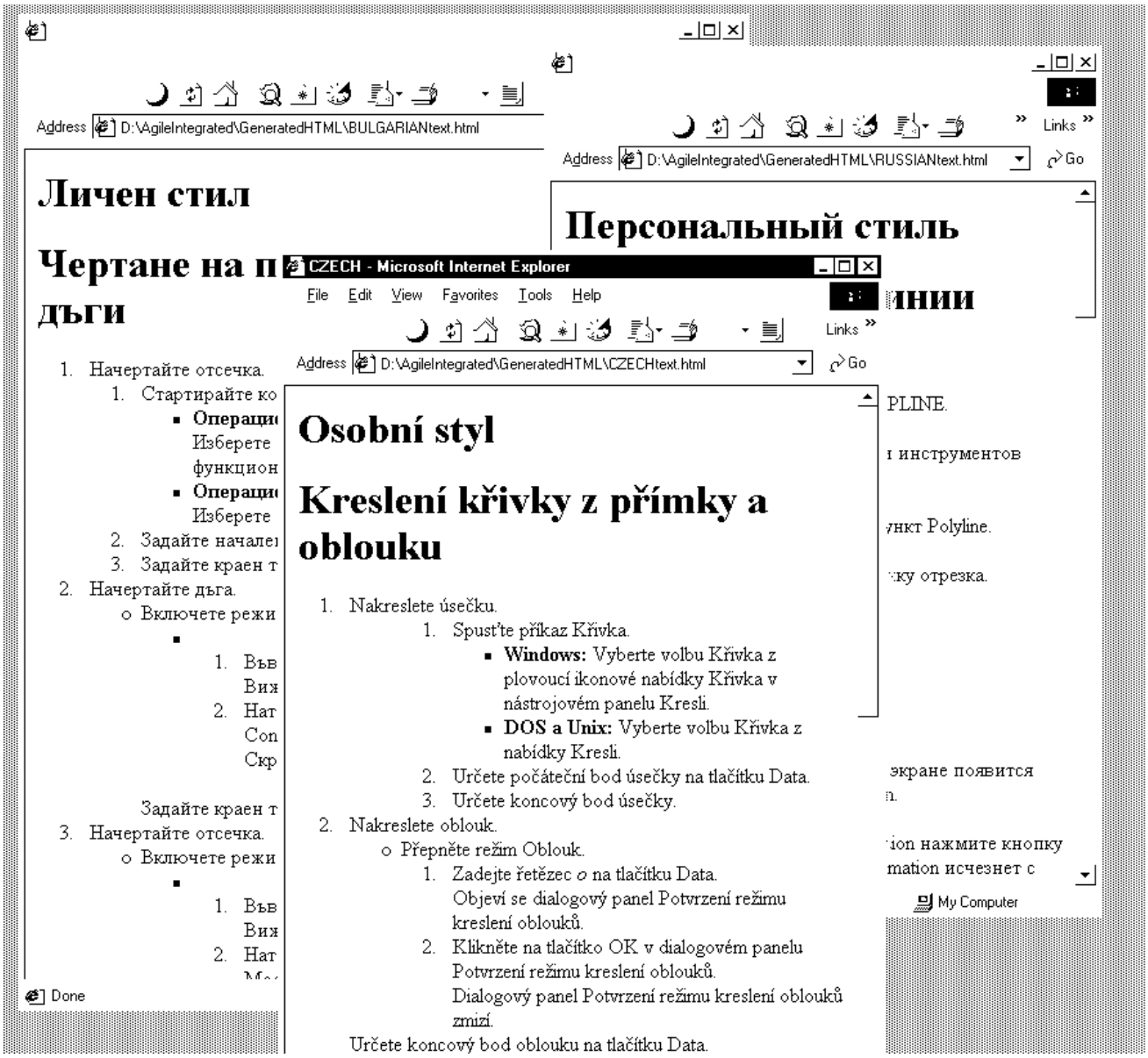

Figure 5: The output of generated text in all three languages