How Expert Systems in Legislative Drafting Might Influence the Law

Grayfred B. GRAY[*]

Abstract

This paper reports on experience with normalized statutes enacted in Tennessee and a prototype legal expert system building program. Legislative drafters should accept Expert System Builders as new members of the legislative audience that, like accountants, prepare legal information and advice for others. Such legal expert systems seem likely to lead to more standardization of how statutes look and to greater readability. They may become part of computer-assisted legislative drafting programs. Standardization might improve the quality of statutes, at least with respect to readability.

I.   Introduction

I am honored to discuss how the use of expert systems in legislative drafting might influence the law. I was asked to supplement Ms. Daniele Bourcier's presentation by drawing on research in Tennessee where we have enacted several statutes that were standardized to facilitate development of one type of legal expert system.

For my purposes today I will define an expert system as a computer device, composed of a computer program and information, that helps a human being do or know something by providing the person information that the program, based on interaction between the person and the device, selects and manipulates in a way that looks like reasoning.[1] Greinke emphasizes the requirement of a high level of expertise to warrant the term "expert system." For this paper expertise is a relative matter to be measured against the system's audience's needs and its purposes. A legal expert system then is such a computer program coupled with legal information.

As with drafting statutes, there are times when a presentation requires the use of terms other than common ones. That is the case here where I will discuss three kinds of legal expert systems. So, I shall refer to them as follows:

1.   Drafting System[2] means a legal expert system that helps legislative drafters write statutes.

2.   Consultant System[3] means a legal expert system that informs or advises others after statutes are enacted.

3.   Expert System Builder[4] means a legal expert system program that helps build Consultant Systems.

---

[1]   This definition might be usefully compared to one attributed to Feigenbaum by Andrew Greinke, "Legal Expert Systems: A Humanistic Critique of Mechanical Legal Inference," **E Law**, at text for n. 16 (Nov. 1994), Address: Murdoch University Law School, PO Box 1014, Canning Vale, Western Australia, 6155, Phone : +61 09 360 2976, Email: elaw-editors@csuvax1.murdoch.edu.au *quoting* P. Harmon & D. King, **Expert Systems: Artificial Intelligence in Business** (John Wiley & Sons: New York, 1985) at 5.

[2]   These systems are commonly referred to as computer-assisted legislative drafting systems as they are in the conference program for this session.

[3]   These are commonly called simply legal expert systems.

[4]   These systems are commonly called expert system shells.

Drafting System, Consultant System, and Expert System Builder are not standard terms in the field of expert systems. I hope they will reduce redundancy that might otherwise be a part of my remarks and make them easier to follow. This diagram illustrates some of the connections between the kinds of systems.

Legislative drafter and Drafting System write a statute. >>>	Lawyer and Expert System Builder create a rulebase for a Consultant System >>>	Person seeks legal information from a Consultant System

I make three assumptions about legislative drafting that I want to state.

1. Legislative drafters share an obligation with the legislature to write statutes that meet the accepted standards of communication of the public they address[5] and to make the law reasonably accessible to the public.

2. A legislative drafter's most challenging tasks are creating new language and structuring substantial legislation.

3. Consultant Systems will become a common medium through which the public, business, and government employees will learn the law and be guided in applying it.

I will describe a prototype rule-based legal Expert System Builder and the normalized form of statutes that it accepts, as enacted by the legislature, to build Consultant Systems. Then I will explore implications

---

[5] Reed Dickerson, **The Interpretation and Application of Statutes** (1975) (in the United States, there is a constitutional duty to make the laws reasonably accessible to the people) pp. 10-12. See an interesting review of different views on communicating with the public in Spring Yuen Ching Fung & Anthony Watson-Brown, **The Template: A Guide for the Analysis of Complex Legislation** (Institute of Advanced Legal Studies, University of London, 1994) §§ 3.1-3.8.

 of using such Expert System Builders for legislative drafters. Finally I will talk about how the use of a Drafting System could help a legislative drafter improve the readability the drafter might have achieved by writing the statutes to put them in an Expert System Builder.

I hope to show that two things are true. First, legislative drafters should accept Expert System Builders as new members of the legislative audience that, like accountants, prepare legal information and advice for others. Second, the kinds of expert systems that I will talk about seem likely to lead to more standardization of how statutes look and to greater readability.

Such standardization might improve the quality of statutes, at least with respect to readability. Drafting Systems may also make Consultant Systems less expensive to develop and maintain while making them more reliable. Eventually legislatures might even build Consultant Systems as a direct by-product of the enactment of statutes.

By way of caution, let me note that state legislatures in the United States do not appear to use Drafting Systems of the sophistication we are discussing. In 1995, the National Conference of State Legislatures reported the results of its survey of how legislative bills were drafted in the states. Though all of the states used computers and some reported using macros in wordprocessors, the report did not appear to include anything that was a Drafting System as I have defined that term.[6]

II. An Expert System Builder prototype and normalized statutes illustrate a step toward standardization of statutes

The Natural Language Expert System Builder program (NLESB) is a prototype Expert System Builder developed in Tennessee. We designed it to enable lawyers to produce a Consultant System from legal rules expressed in normalized form[7] more easily than with other Expert System

---

[6] Table 5, Preparation of Bills, Statutes, and Journals (1995), **Guide to Legislative Information Technology**, National Conference of State Legislatures.

[7] Some of the reasons for requiring normalized input are illustrated in an article that reported on an experiment using statutes that had been adopted originally in normalized form. *See generally Preparing Enacted Normalized Statutes for an*

Builders.[8]   My co-researchers, professors Bethany K. Dumas, John E. Nolt, Bruce J. MacLennan, Donald R. Ploch of the University of Tennessee faculty and I developed the program. NLESB was given its name because it accepts rules and questions from users in natural language.[9]   Indeed, all of the user's substantive communication with NLESB or a Consultant System built with it is either natural language or menu driven.

A.   NLESB as an illustration of how expert systems may increase standardization of statutes

NLESB's operation illustrates how Drafting Systems could help drafters and how expert systems might influence the law because NLESB requires a standardized form of legal rule to work with[10] and does things that might be useful in a Drafting System. While NLESB was not developed to be a Drafting System, a drafter could use it to improve the readability of proposed legal rules and, assuming a rulebase was prepared for it, to find legal rules that a drafter might want to take into account.

NLESB has the common capacities of an Expert System Builder program developed to meet the particular needs of lawyers as creators of

---

Expert System, 4 **CC-AI: J. Integrated Study of Artificial Intelligence, Cognitive Science and Applied Epistemology** 389 (1987) (Belgium).

[8]   For a description of an effort to put the enacted normalized statutes into a general purpose Expert System Builder program that was not designed for use by lawyers, see Grayfred B. Gray, *An Experiment with Normalized Statutes in an EMYCIN Expert System*, **Computer Power and Legal Language** (Charles Walter ed.) (Quorum 1988) (proceedings of the International Conference on Law and Technology, University of Houston Law School (1985)).

[9]   Normalized statutes, while more formally structured than many other natural language sentences, are natural English with rigorously consistent and consistently signaled syntax operators between its constituent propositions. The use of item labelling and outline indentation have been common in statutes for years.

[10]   In the development of NLESB we also discovered peculiarities of statutory logic. For details, see John E. Nolt, Grayfred B. Gray, Bruce J. MacLennan, and Donald R. Ploch *The Logic of Statutory Law*, 35 **Jurimetrics Journal** 121 (1995).

Consultant Systems.[11] Capacities could be added to NLESB to make it a useful Drafting System.   A thesaurus, for example, would expand its searches for relevant rules.

NLESB builds the most rudimentary form of Consultant System, a rule-based system, one that draws inferences from rules constructed in the general form "IF (named conditions occur) THEN (these are the legal results)." [12]

An Expert System Builder's basic capacities are:

1.	To take in legal knowledge in a structured form and create a database of the domain.

2.	To help a person using the system improve the expression of legal knowledge by finding similar information, negation, and implicit information.

3.	To draw inferences from its rules in response to queries posed by a person who uses the system and to query the person to get additional information needed to draw inferences.[13]

4.	To explain, based on its rules, why it asks particular questions and how it derives its inferences.

**B.	Statutes enacted in normalized form**

---

[11]	*See, e.g.,* Grayfred B. Gray, John E. Nolt, Bruce J. MacLennan, and Donald R. Ploch*, The Logic of Statutory Law*, 35 **Jurimetrics Journal** 121 (1995).

[12]	Edwina L. Rissland, *Artificial Intelligence and Law: Stepping Stones to a Model of Legal Reasoning*, 99 **Yale L.J.** 1957, 1965-68 (1990) (a brief summary of the state of legal expert systems in 1990).

[13]	NLESB, acting as a Consultant System poses queries to a user by formulating them from its rules. That capacity is a singularly important aspect of Expert System Builder as part of the legislative audience. Converting a proposition into a question often reveals ways the proposition can be expressed better. Consequently, to consider the formulation of queries in drafting statutes can, by itself, lead to statutes that are easier to read and apply. NLESB shows a drafter immediately the questions that will come from the system's use of a proposed statute.

Normalized statutes are intended to eliminate syntactic ambiguity between or among conditions and legal results expressed in legal rules. Professor Layman E. Allen pioneered the development of normalized legal drafting.[14]   For a specific method for converting existing legal rules to normalized form, see generally Layman E. Allen and C. Rudy Engholm, *Normalized Legal Drafting and the Query Method*, 29 Journal of Legal Education 380 (1978). While it can have many syntax operators, only five operators have been used in Tennessee in the interests of readability and acceptability to the legislative audience.[15]   This limitation reflects a legislative drafter's decision about how best to communicate with the legislative audience.

1.   The syntax operators

Normalized syntax operators in the enacted statutes in Tennessee are "IF," "IF AND ONLY IF," "AND," "OR," and "THEN." The syntax operators, written in all capital letters, express solely the syntax between propositions written as complete sentences that convey either conditions for a legal result or the legal result of the conditions. "IF" and "IF AND ONLY IF" have their ordinary meanings as do "AND" and "THEN." "OR" is the inclusive "or." "OR" joins only propositions that express conditions. The propositions and syntax operators of the statute are always outlined and labeled in the same manner.

2.   Examples of enacted normalized statutes

---

[14]   *See, e.g.*, Layman E. Allen, *Symbolic Logic: A Razor-Edged Tool for Drafting and Interpreting Legal Documents*, 66 **Yale L.J.** 833 (1957); Layman E. Allen, *Language, Law and Logic: Plain Legal Drafting for the Electronic Age*, *in* **Computer Science and Law** 75-100 (B. Niblett ed., 1980); Layman E. Allen, *Formalizing Hohfeldian Analysis to Clarify the Multiple Senses of 'Legal Right': A Powerful Lens for the Electronic Age*, 48 **S. Cal. L. Rev.** 428, 428-87 (1974).

[15]   For a simple introduction to normalization for legislative or other legal writing see *Reducing Unintended Ambiguity in Statutes: An Introduction to Normalization of Statutory Drafting*, 54 **Tennessee Law Review** 433 (1987).

a.   A sample IF rule

IF (1) an officer authorized to make arrests in Tennessee or a licensed physician has reason to believe that a person is subject to detention under subsection (a),

THEN

(2)   the officer or physician may take the person into custody without a civil order or warrant for immediate examination under subsection (d) for certification of need for care and treatment.[16]

b.   A sample IF AND ONLY IF rule

IF AND ONLY IF

(1)   a person is mentally ill, AND

(2)   the person poses an immediate substantial likelihood of serious harm, as defined in § 33-6-104, because of the mental illness,

THEN

(3)   the person may be detained under subsection (b) to obtain examination for certification of need for care and treatment.[17]

---

[16]   Tenn. Code Ann. § 33-6-103(b) (Supp. 1996).

[17]   Tenn. Code Ann. § 33-6-103(a) (Supp. 1996) (item label on (1)(A) is corrected from the way it erroneously appears in the 1996 supplement). Item (3) in the statute is expressed in passive voice and, in this context, makes the statute shorter and simpler than if it had listed those qualified to act. Query whether the drafter should have written the item in the active voice. The author, who drafted these statutes, wishes he had had the benefit of a legislative drafting expert system in writing the first normalized statutes in 1981.

  c. A sample complex IF rule illustrating OR and nested IF...THEN structures

IF

   (1) (A) the patient does not request judicial review of the discharge plan, OR

   (B) the court approves an outpatient treatment plan after a hearing under subsection (c),

THEN

   (2) IF

   (A) the patient is subject to judicial review under § 33-6-110,

THEN

   (B) the patient shall be discharged in conformity with § 33-6-110, AND

   (3) IF

   (A) the patient is not subject to judicial review under § 33-6-110,

THEN

(B) the hospital shall discharge the patient, AND

(4) The hospital shall notify the committing court that the patient has been discharged subject to the obligation to participate in the outpatient treatment.[18]

d. A sample substantive definition rule

IF AND ONLY IF

(1) (A) a person has threatened or attempted suicide or to inflict serious bodily harm on himself, OR

(B) the person has threatened or attempted homicide or other violent behavior, OR

(C) the person has placed others in reasonable fear of violent behavior and serious physical harm to them, OR

(D) the person is unable to avoid severe impairment or injury from specific risks, AND

---

[18] Tenn. Code Ann. § 33-6-201(d) (Supp. 1996).

(2)   there is a substantial likelihood that such harm will occur unless the person is placed under involuntary treatment,

THEN

(3)   the person poses a "substantial likelihood of serious harm" for purposes of § 33-6-103 and this section.[19]

3.   How readable are normalized statutes?

The research that resulted in NLESB began with an experiment to test the readability of normalized rules because that form was what NLESB would show users in operation. In the experiment, non-lawyer subjects applied laws that they were not likely to have read before to problems they were unfamiliar with. Three of the five statutes had been enacted in normalized form. Each subject in the course of the experiment saw five forms of rules: fully normalized, normalized but with the condition-result order reversed, unindented normalized, unindented normalized with lower case syntax operators, and ordinary text. We stripped the statutes of legalistic jargon and cross-references to other statutes to eliminate those threats to readability. On each statute the different forms of the law varied at most five percent on conventional tests of readability.

The experiment showed:

1.   Non-lawyers applied complex normalized statutes with the Tennessee syntax operators more accurately than statutes written in the other four ways, including traditional legislative drafting.

---

[19]   Tenn. Code Ann. § 33-6-104(a) (Supp. 1996).

2.   The non-lawyers applied less complex normalized statutes at least as well as the other four forms of the statutes.[20]

We inferred from those results that statutes expressed in normalized form were at least as easy to read as the other forms and that complex statutes were easier to read in normalized form than in the other forms.

Normalized statutes always have some features that are thought to improve readability. First, they are simple and consistent in their between proposition syntax operators. Second, they support reader expectations by signaling standard syntax operators unambiguously. Third, they have uniform indentation and a standard outline format that suggests correlation of parts. Fourth, their components, the propositions, are grammatically complete sentences. Fifth, their complexity and size are limited to six or seven propositions in a single rule.[21] In the experiment, apparently it was the combination of the features in the normalized statutes that enhanced their reability where they were complex and held them at least equally readable with the other forms where they were simple.[22]

C.   Uses of NLESB that might help a statutory drafter[23]

When NLESB takes in a statute, it first scans for formal defects in the normalization. If it finds any, it reports to the user and sometimes

---

[20]   Donald R. Ploch, Bethany K. Dumas, Grayfred B. Gray, Bruce J. MacLennan, and John E. Nolt, *Readability of the Law: Forms of Law for Building Legal Expert Systems*, 33 **Jurimetrics J.** 189 (1993).

[21]   This rule of thumb was based on comments from people who used the first enacted normalized statutes.

[22]   Donald R. Ploch, Bethany K. Dumas, Grayfred B. Gray, Bruce J. MacLennan, and John E. Nolt, *Readability of the Law: Forms of Law for Building Legal Expert Systems*, 33 **Jurimetrics J.** 189, 195 (1993).

[23]   For a fuller exposition of the processes, see Grayfred B. Gray, Bruce J. MacLennan, John E. Nolt, and Donald R. Ploch, *Legal Expert System Building: A Semi-Intelligent Computer Program Makes It Easier*, 12 **John Marshall J. of Computer & Information L.** 555 (1994).

pinpoints the area of the deficiency. For example, taking in a rule in which THEN had been omitted, NLESB advised the drafter:

DEM4CIAJ   C:\EXPERT\FILES   Matching level: 20%   Free memory: 224827 byte

Parsing revised Def. of files pro se

ERROR...

 Incorrect normalization.

Once the statute is formally adequate, NLESB looks for the use of NOT and asks whether a proposition in which it appears means its opposite if the NOT is taken out.[24]IF AND ONLY IF

> (1) The petitioner files a petition for an order of protection, AND

> (2) The petitioner does not have a lawyer,

> THEN

> (3) The petitioner files the petition pro se.

DEM4CIAJ   C:\EXPERT\FILES   Matching level: 20%   Free memory: 224730 byte

Is this proposition:

The petitioner does not have a lawyer

the negation of this one:

---

[24]   NLESB is responding to taking in this rule:

The petitioner does have a lawyer? (Y/N)

In building a Consultant System, NLESB next finds rules in the system that may be related to the new rule and then asks the person who is using NLESB to decide how they fit together. In NLESB the process is called *unification*. NLESB compares each proposition in the new rule to all propositions it already contains. It compares on the basis of the percentage of words that match in each pair of propositions. When the match is at the level that the user has prescribed, NLESB displays propositions that are sufficiently similar to a proposition in the new rule that both might mean the same thing.

NLESB in a unification procedure

DEM4CIAJ   C:\EXPERT\FILES   Matching level: 20%   Free memory: 87805 bytes

The propositions below match the proposition 'The petitioner files a petition for an order of protection'.

Choose any you may consider unifying with it:

Proposition F10: Done with list ESC: Abort unification # of matching words

The petitioner seeks to file a petition for an order of protection

The petitioner may file a petition for an order of protection*

CHOOSE   SHOW_CONTEXT

The builder then can see each in its fuller context to decide whether they mean the same thing, whether they could be expressed identically, and, if so, whether to change the language of either or to preserve both as

expressed and note that if one is true in application, the other is also true. If the builder tells NLESB that the propositions mean the same thing, the Consultant System being built knows more than merely the expression of the rules.

Seeing similarly expressed parts of statutes in isolation from other text, however, provides a legislative drafter a fresh opportunity to pursue consistency of language and structure and other features of clear drafting. For example, when the similar propositions are different only in that one is in active voice and the other in passive voice, that difference leaps out. When the active-passive distinction is present, seeing it may reveal that the passive voice proposition did not show who the actor was.

Alternatively, the isolated propositions may show undetected negation in one proposition and the absence of negation in the other. For example, where one proposition includes the word "written," the other may include "unauthorized." There may be many other negations in sentences containing words with prefixes such as "un." A drafter may be more likely to notice such negation in propositions in isolation than in the midst of paragraphs.[25]

While normalization eliminates syntactic ambiguity among the propositions it connects explicitly, it does not eliminate syntactic ambiguity between separate legal rules. NLESB reveals the potential for reducing such syntactic ambiguity in unification. For example, the drafter who sees the similar language and checks the context, may find in one case a legal result proposition and in the other a condition that represents carrying out the legal result. That raises the question of whether the drafter wants to make the connection between the rules explicit.

When NLESB reports similar propositions between a proposed statute and existing statutes, the drafter also may realize that it is possible to write the proposal to amend an existing statute rather than to write a new one.

---

[25] Among other words that communicate negation are "neither...nor," "no . . ." forms, words with the prefix "non," and exception signals such as "except," "however," "provided, however, that," "notwithstanding."

III. Expert System Builders as part of the legislative audience for legislative drafters

A. What makes Expert System Builders part of the legislative audience?

Traditionally legislative drafters have taken as their audience

1. Legislators and others who are part of the legislative process because they were the people involved in enactment,

2. The courts because they were the authoritative interpreters,

3. The public because it is governed by them,

4. The bar and government because they carry out the laws, and

5. Groups like accountants[26] because they were the primary conveyors of some law for those governed by it.

The rationale that makes accountants part of the audience fits Expert System Builders as well. Legislative drafters should recognize Expert System Builders as a new member of the legislative audience[27] and the Consultant Systems as a new medium for conveying the law to the public. Expert System Builders are important because they can help meet our needs to simplify application of the law. Consultant Systems make application of law interactive, step-by-step, and simple. They make the law more readily accessible than full text. Consultant Systems don't merely present the law to the people, they find relevant parts and help the people reason through its application.

---

[26] Reed Dickerson, **The Interpretation and Application of Statutes** (1975) pp. 117-220.

[27] One could argue that the real audience is not the Expert System Builders but the people who build such systems. However, the demands that legislative drafters are to address are essentially intrinsic to the Expert System Builders, not to the people who use them.

While Consultant Systems are not commonly used in the United States, there are a growing number of Consultant Systems.[28] Lawyers have begun to make substantial use of document assembly systems that run on IF-THEN rules, and some large firms have built their own. Ronald W. Staudt, *Does the Grandmother Come With It?: Teaching and Practicing Law*, 44 Case W. Res. L. Rev. 499, 515 (1994). and building such systems is an international phenomenon.[29] Business, unions, and professional organizations want to cut the cost of legal services and reduce their risk of inadvertent violations of the law. Consultant Systems are one way they could, and legislative drafters could write legislation that would reduce the cost of developing and maintaining Consultant Systems.

B. **Special needs of the new audience**

An Expert System Builder like any other legislative audience has special needs. Fortunately, as will be illustrated in a moment, meeting its needs tends to make legislation easier for traditional legislative audiences to understand.

Expert System Builders require the elimination of syntactic ambiguity between expressions of conditions and the legal results of those conditions. The systems also need a greater level of completeness than traditional audiences require because the expert system cannot read in implications and cannot use common sense to supply information. The

---

[28] Edwina L. Rissland, *Artificial Intelligence and Law: Stepping Stones To a Model Of Legal Reasoning*, 99 **Yale L.J.** 1957, 1968 (1990).

[29] See, e.g., Voermans W. and E. Verharen, *Leda: A Semi-Intelligent Legislative Drafting-Support System*, in: J.S. Svensson, J.G.J. Wassink and B. van Buggenhout (eds.) **Legal Knowledge Based Systems: Jurix '93: Intelligent Tools for Drafting Legislation, Computer-Supported Comparison of Law** (Lelystad: Koninklijke Vermande: 1993) 81-94; *see also* Edwina L. Rissland, *Artificial Intelligence and Law: Stepping Stones to a Model of Legal Reasoning*, 99 **Yale L.J.** 1957, 1965-68 (summarizing some of the earlier work); Phillip Capper & Richard Susskind, **Latent Damage Law--the Expert System** (1988); Marek Sergot, Sadri, Kowalski, Kriwaczck, **Hammond & Cory, The British Nationality Act as a Logic Program**, 29 COMM. ACM 370 (1986).

need for completeness is not trivial for Consultant Systems, because their level of expertise is diminished without it.[30]

For Expert System Builders legislative drafters should be careful to state explicitly duties that in the past they might have left implicit and to state the consequences of breaches of duty. For example, in the following statute did the drafter (or the legislature) mean to impose a duty on the boards to adopt a definition of "suitable accommodations" or to convey a privilege if a board did adopt one?

> Except in the circumstances of an emergency under § 33-6-103, admission is subject to the availability of suitable accommodations. The admitting physician shall have on hand at all times for public inspection the definition of "suitable accommodations" which shall have been determined by the board of the institution involved.[31]

Did the drafter mean to prohibit admitting a patient when suitable accommodations were not available or to give the institution permission to create an impediment to admission by adopting a definition of "suitable accommodations?" Building a Consultant System with those statutes would require judgment to answer such questions. Better drafting could avoid raising them.

In short, legislative drafters can take into account the Expert System Builder audience and serve traditional audiences better by these steps:

1.   Normalize for syntactic clarity;

---

[30] Andrew Greinke distinguishes between "decision support systems" and "expert systems," the former merely support decisions made by humans while the latter provide advice as would a human expert. "Legal Expert Systems: A Humanistic Critique of Mechanical Legal Inference," **E Law**, section 1 (Nov. 1994), Murdoch University Law School, PO Box 1014, Canning Vale, Western Australia, 6155, Phone : +61 09 360 2976, Email: elaw-editors@csuvax1.murdoch.edu.au. What seems central to either kind of system is that it needs to be as smart as possible to be very effective, and neither is ever likely to be complete.

[31] Tenn. Code Ann. § 33-6-101(a)(2) (1996 Supp.).

2.   Express negation by "not" and avoid the other ways of expressing negation;

3.   Write the complete rule and use full cross-references;

4.   In writing the constituent propositions, consider what questions the propositions will turn into;[32]

5.   Write short propositions that are not dense in clause depth;

6.   Write in the active voice and narrowly and very carefully confine the passive voice;

7.   Write in the present tense except where it would be inaccurate; and

8.   Make each statutory sentence a separate section or use a subsection numbering system that will be readily distinguished from the item labels in the bills and the codes.[33]

Authorities on legal writing, in and outside the Plain Language movement, have identified similar ways to make statutes more readable, including meeting the reader's expectations as to the location of information in the text, a notion that suggests standardization and that normalization seems responsive to.[34] Treating Expert System Builders as part of the legislative audience could make it easier for drafters to carry out some of the steps toward readability and consequently effect standardization of statutes beyond normalization.

---

[32]   Auxiliary verbs tend to be troublesome when the propositions are posed as questions. It may be useful to avoid them.

[33]   This last item may be a surprise. A number of people who read normalized statutes in the Code have told the author that where a section is broken into subsections, the subsection labels, *e.g.*, (a), tend to get lost in the item labels and make the statute hard to read.

[34]   See, e.g., George D. Gopen, *The State of Legal Writing: Res Ipsa Loquitur*, 86 **Mich. L. Rev.** 333, 347-50, 360-61 (1987), in an article that is often critical of advocates of Plain Language. Lisbeth Campbell, *Drafting Styles: Fuzzy or Fussy?*, 3 **E Law** No. 2 (July 1996) (Murdoch University Electronic Journal of Law) (http://www.murdoch.edu.au/elaw/issues/v3n2/campbell.html), discusses differences in fundamental drafting style between countries of the civilian tradition and those of the common law tradition. See para. 22, *id.,* regarding plain language under either tradition.

C.   What makes it easier to meet the Expert System Builder's needs?

Legislative drafters can address the special needs of the Expert System Builder audience more easily by using Drafting Systems designed to facilitate meeting those needs. If one thinks of NLESB's operation, despite its not being designed as a Drafting System, it would actually help meet those needs just as it is. It challenges deviations from the formality requirements, and its output meets them. In addition, NLESB processes such as unification would help the drafter obtain greater consistency and a higher level of completeness. The use of Drafting Systems can help address the needs of both the new and the old members of the legislative audience to the extent that those needs overlap.

IV.   Potential readability gains from the use of a Drafting System with Expert System Builder capacities

A Drafting System could help a drafter to simplify statutes in other ways if it were designed to review draft language for problems that are especially common in legislative writing. It should also give the drafter a high level of control over what it looks for and the sequence in which it searches for problems.

A Drafting System should permit the legislative drafter to set a limit on the length of propositions. Shorter propositions are more readable, and they are more likely to formulate questions that users can answer reliably in a Consultant System. One advantage of shorter propositions is a reduction in clause depth in each proposition. In the readability experiment described earlier,[35] one conclusion was that clause depth was high in all forms of the law that were used. That apparently characterizes legal writing and is an obstacle to achieving higher levels of readability.

---

[35]   Donald R. Ploch, Bethany K. Dumas, Grayfred B. Gray, Bruce J. MacLennan, and John E. Nolt, *Readability of the Law: Forms of Law for Building Legal Expert Systems*, 33 **Jurimetrics J.** 189 (1993).

A Drafting System should screen for pronouns to have the builder replace them with their referents. Pronouns and other reference terms, for example, "such" and "said," lose their point of reference when they appear in questions in an Expert System Builder because the referent is likely to be in another proposition that is the basis of a separate question. Unless the user keeps the earlier question fully in mind, a question containing "such" is at best uncertain and potentially worse.

For example, NLESB would convert the following two propositions into questions that would not appear on the screen simultaneously:

"a licensed physician takes a person into custody under this section" and

"a person is brought to such a physician for examination under this section."

The Expert System Builder would present the user the question:

"A person is brought to such a physician for examination under this section?"

The user then might not know that "such a physician" means a licensed physician and that "this section" means "section 33-6-103." The example also illustrates the uncertainty created by words like "this" that convey information but do so with a significant risk of misreading when their larger context is lost.

An effective Drafting System should scan for most common forms of negation and offer the drafter the opportunity to eliminate ambiguity about the negation or to make a deliberate choice to leave it. Leaving it will mean that the expert system builder will have to resolve it or call it to the attention of the system user.

A Drafting System should scan proposed statutes for deontic operators—expressions that create a duty, permission, prohibition, or

---

See Andrew Greinke, "Legal Expert Systems: A Humanistic Critique of Mechanical Legal Inference," **E Law**, section 2.4 (Nov. 1994), Contact Address: Murdoch University Law School, PO Box 1014, Canning Vale, Western Australia, 6155, Contact Phone:+61 09 360 2976, Contact Email: elaw-editors@csuvax1.murdoch.edu.au; Layman E. Allen & Charles S. Saxon, *A-Hohfeld: A Language for Robust Structural Representation of Knowledge in the*

power (for example, "shall" or "may")—and then search for propositions that may express the performance that relates to the operator. Once the system identifies such a pair, it should offer the drafter opportunities to:

1. decide whether the differences in language between the propositions are called for by the differences in the propositions and

2. decide whether to make an explicit cross-reference in the rule that hinges on the existence of the deontic operator.

Of course, the drafter might choose instead to write out an otherwise implicit connection between the rules to avoid syntactic ambiguity.

A good legislative drafting expert system also would have features now found in grammar and style checkers in wordprocessors. However, the features should focus on the problems that bedevil legal drafting. For example, the system should screen routinely on the first pass for the passive voice, for Latinisms, and for other legalistic and generally unnecessary words that impede comprehension.

V. Conclusion

It is reasonable for those who want Consultant Systems to ask that legislative drafting minimize the problems and costs of creating them. Otherwise, private citizens, businesses, and much of government will have to pay more to have reliable Consultant Systems. Legislative drafters should write statutes clear enough that such systems can reason with them with little human preparation past enactment so long as the system's needs can be satisfied without sacrificing the human reader's comprehension. Normalized legislative drafting can achieve a large degree of that clarity now.

Drafting Systems, once available, may ultimately make legislative drafting faster and may enable legislative drafters to focus more of their time and attention on the substantive aspects of their work. A Drafting System could help assure thoroughness in important aspects of legislative drafting. The advent of the new member of the legislative audience also makes it more important to use such tools as grammar or style checkers to achieve desirable levels of uniformity.

A Drafting System also could be helpful in testing amendments against existing rules by checking how proposals integrate or compare with current rules. Such checking could be simpler than boolean searches with conventional computer-assisted legal research and could enable a legislative drafter to make judgments more easily about whether similar language means the same thing. The expert system checks not only the same words or identical phrases but also similar language. This would be especially true with tools aided by a thesaurus to find similar expressions that do not even use the same words.

There may be a tug toward more dramatic changes in the way the law is written as frames, templates, matrices, and other techniques prove fruitful in Artificial Intelligence and law. However, laws are fundamentally written for the people. We need to keep the new audience in check, to make Drafting Systems servants in our effort to serve the traditional audiences for legislative writing. Otherwise we will be subject to further challenges on making the law unreadable and may violate our obligation to make it accessible and to write it in the language of the speech community which we address. Something like normalization coupled with better implementation of traditional drafting standards is probably a level of standardization we should aim for.

Even with Drafting Systems legislative drafters, like all writers, will be able to make mistakes, to make the words too large or too small for their purpose, to choose the wrong words, to leave things out. Drafting for an audience of Expert System Builders presents a new challenge. The combination of new Drafting Systems and the new legislative audience is almost certain to result in more standardization of legislative expression and laws that are clearer to the public. It also may contribute to having more thoroughly considered legislation.