NIST GCR 11-944

Testability Report for SCRIPT

Edmond Scientific Company

NIST National Institute of Standards and Technology • U.S. Department of Commerce

NIST GCR 11-944

Testability Report for SCRIPT

Prepared for
National Institute of Standards and Technology
Gaithersburg Md 20899-8202

By
Edmond Scientific Company
July 1 2011

U.S. Department of Commerce
John E. Bryson, Secretary

National Institute of Standards and Technology
Patrick D. Gallagher, Under Secretary for Standards and Technology and Director

This publication was produced as part of contract SB1341-10-CN-104 with the National Institute of Standards and Technology. The contents of this publication do not necessarily reflect the views or policies of the National Institute of Standards and Technology or the US Government.

Electronic Prescribing

Summary Testability and Suitability report for "NCPDP SCRIPT Implementation Guide: Version 10.6"

Written by Ioana Singureanu

Submitted September 12th, 2011

4000 Legato Road, Suite 1100
Fairfax, Virginia 22033

Document Version Control

Version	Date	Name	Change(s)
1.0	03/30/2011	Ioana Singureanu	Initial Draft
2.0	07/01/2011	Ioana Singureanu	Second Draft
3.0	8/31/2011	Ioana Singureanu	Release Version

Table of Contents

Contents

Figures:

This document summarizes the major testability issues raised by the *NCPDP SCRIPT Implementation Guide Version 10.6*. The ELR Implementation Guide is intended as a means of testing and certifying Electronic Health System (EHRS) for Meaningful Use.

During our suitability analysis of the SCRIPT Implementation Guide (IG) we uncovered several testability issues and proposed recommendations to enhance the testability of solutions based on the SCRIPT IG. Ambiguities and inconsistencies in the IG as well as its paper-only representation pose some challenges to those responsible for implementing the specification into working software. This document discusses some of the issues that could lead to an inconsistent implementation and make it challenging for the National Institute of Standards and Technology (NIST) to develop reference implementations and testing infrastructure for implementations based on the SCRIPT IG to fulfill ePrescribing and Medication Reconciliation functions consistent to the Meaningful Use criteria and test methods.

The Suitability report evaluated the SCRIPT IG from several view-points:

- Quality of the standard constraints specified for implementers. Constraints have to be applied to appropriate structures or message elements.
- Gaps in the specification, including ambiguities that could lead to multiple valid interpretations of the SCRIPT IG.
- Testability considerations and issues related to validating implementations of SCRIPT IG. These issues relate to unspecified business rules and conditional predicates that may not be testable.
- Implementability considerations based on the clarity of the specification and the strengths of the constraints specified therein.

Our analysis attempts to reconcile the contents of the Implementation Guide with the contents of the XML Schema Descriptions (XSD) for the message structures relevant to the ePrescribing and Medication Reconciliation, New Drug Prescription (NEWRX) and Prescription History Reponses (RXHRES). The overarching testability issues pertaining to these transactions, as well as generic issues related to implementing SCRIPT using XML as encoding syntax are also addressed.

Where applicable, the analysis includes a review of later versions of the SCRIPT Implementation Guide – version 10.11 – to identify best-practice preferences and issues of backwards compatibility. The analysis reconciles not only field lengths and usage, but also separates the type of a field from its other characteristics such

as presence/usage, maximum length, cardinality, and predicates for conditional elements. Descriptions of the analyzed message structures are in Appendices A and B, available as fully navigable HTML files.

The NEWRX and RXHRES transactions are shown in Figure 1. Note that each drug in a prescription is sent as a separate NEWRX transaction (e.g. if a prescription contains three drugs, then three separate NEWRX transactions are sent).

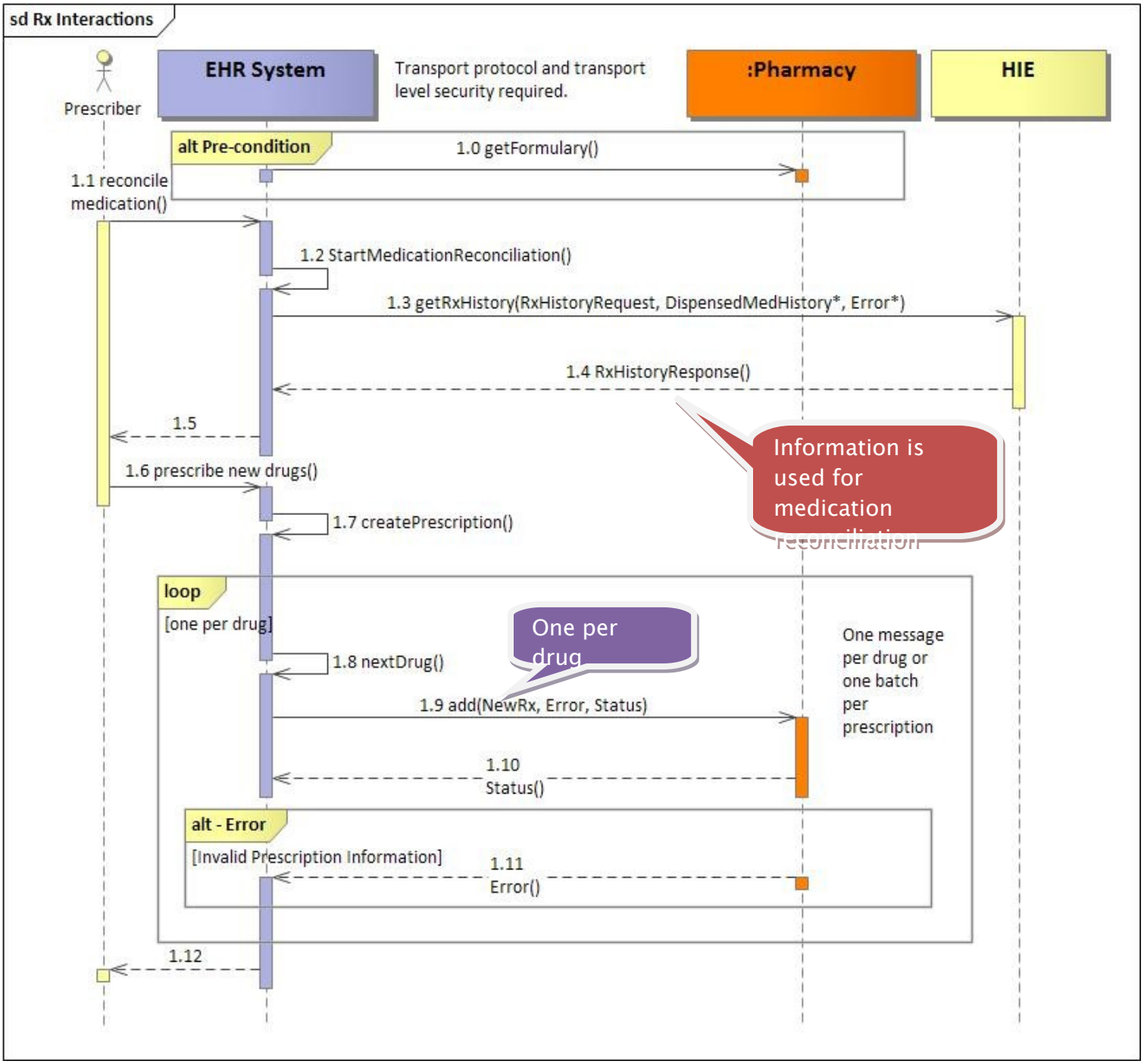


Figure 1: Overview of ePrescribing interactions

Our analysis did not include any considerations of formulary synchronization between the EHR System used by clinicians to place the drug prescription and the Pharmacy System or Health Information Exchange gateway. We assume that the formulary consistency is ensured through other means of communication.

NCPDP SCRIPT XML Testability Issues

The two NCPDP SCRIPT transactions selected for analysis are intended to support specific EHRS capabilities identified in the Meaningful Use regulation. However, since the SCRIPT standard was intended to support primarily financial transactions related to drug prescription and dispensing, the transactions are not fully aligned with the clinical workflows:

- New Drug Prescription (NEWRX) transactions support only one drug prescription at a time. Therefore, if a physician is prescribing several drugs, the EHRS system will be required to send and track one transaction per drug and track the pharmacy response for each.
- Prescription History Response (RXHRES) transactions are useful for medication reconciliation but not sufficient. These transactions allow an EHRS to determine that drugs were dispensed to a patient and provide that information to the user who is performing the reconciliation. Based on the drugs dispensed and the confirmation from the patient that those drugs are being consumed, clinicians are able to determine the active medication list for the patient. In some cases a history of past medication is useful if the physician is looking to identify the most efficacious course of treatment.

While our analysis covered these high-level suitability issues, the rest of this document focuses more precisely on the issues and discrepancies introduced as the NCPDP SCRIPT transactions are migrated from EDI to XML.

Implementation Guide Applicability to XML Format

Until very recently, the Electronic Data Interchange (EDI) syntax was the only valid representation of SCRIPT messages and is still dominant in production implementations. The industry has very little experience with XML-encoded messages and as such, little implementation guidance is available for XML implementers. Documentation such as the SCRIPT Implementation Guide (IG) is still very focused on the EDI syntax and does not provide guidance to implementers on the conventions used to represent the transactions and segments as an XML documents and elements.

Implementers may have difficulties relating the concepts described from the SCRIPT IG to the XML elements and types specified in the XML Schema Descriptions (XSD) provided by NCPDP to describe the XML encoding of .SCRIPT messages. We attempted to bridge the gap between the SCRIPT Implementation Guide and the equivalent XML Schema description in a side-by-side analysis (HTML files in Appendices A and B), but we believe a dedicated implementation guide for XML implementers is necessary if this specification is to be widely adopted by EHRS vendors who, until recently, relied on third-party clearing houses for ePrescribing using proprietary implementations of SCRIPT:

1. **script-newrx.htm** – contains a side-by-side analysis of the NEWRX message structure for Version 10.6 and *External Code List Version 201009* showing the differences between the EDI and XML representations.
2. **script-rxhres.htm** – contains a side-by-side analysis of the RXHRES message structure for Version 10.6 and *External Code List Version 201009*.

The analysis documents were created using the *Messaging Workbench* tool. This tool allows integrators to describe the hierarchical tree structure of any type of message and to document the constraints pertaining to each message element's usage (e.g., mandatory, not supported), cardinality, maximum/minimum lengths, and other formatting rules that may be represented using regular expressions. In addition to the message profile file (.mwb file used with *Messaging Workbench*) for NEWRX and RXHRES structures, we needed to use a data type library file for all the primitive type and structures reused across the specification. A Messaging Workbench data type library contains reusable definitions for the simple and complex data types referenced by the SCRIPT IG. We discovered that a reusable library of data types is very desirable when constraining a message structure because it allows us to constraint all the data types for all the transaction and even to constrain them for each field where they apply.

As an example of how the data type library was designed and implemented, the **DRUG_CODE** data type highlighted in Figure 5 is represented as an XML complex type "**DrugCoded**" in the XSD. This data type is then used in **MedicationPrescribed** and **MedicationDispensed** to represent the drug. Introduction of reusable XML complex types is clearly an improvement over the EDI structure where coded drug information is described as a sequence of field components. Organization of these XML complex types into a reusable type library would allow for refactoring of the current XML schemas and provide a means of applying global constraints (e.g. constraining the drug code to RxNorm across both NEWRX and RXHRES).

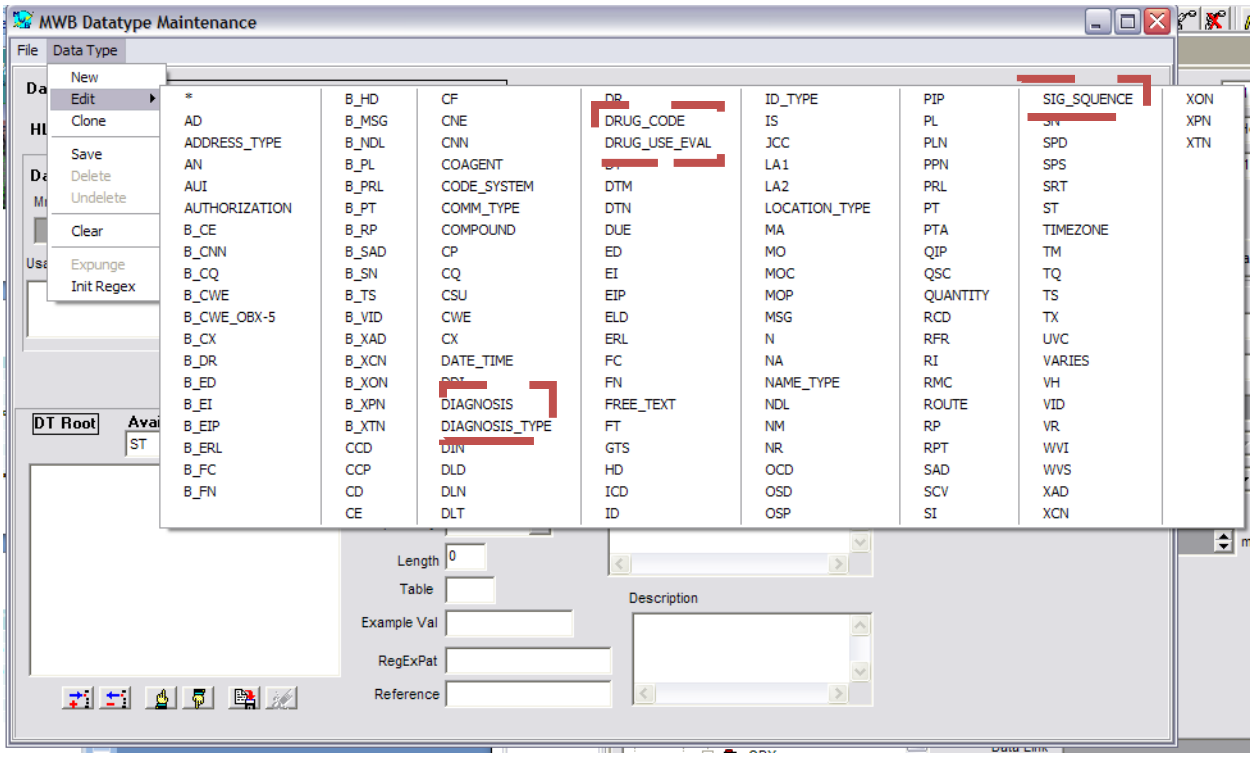
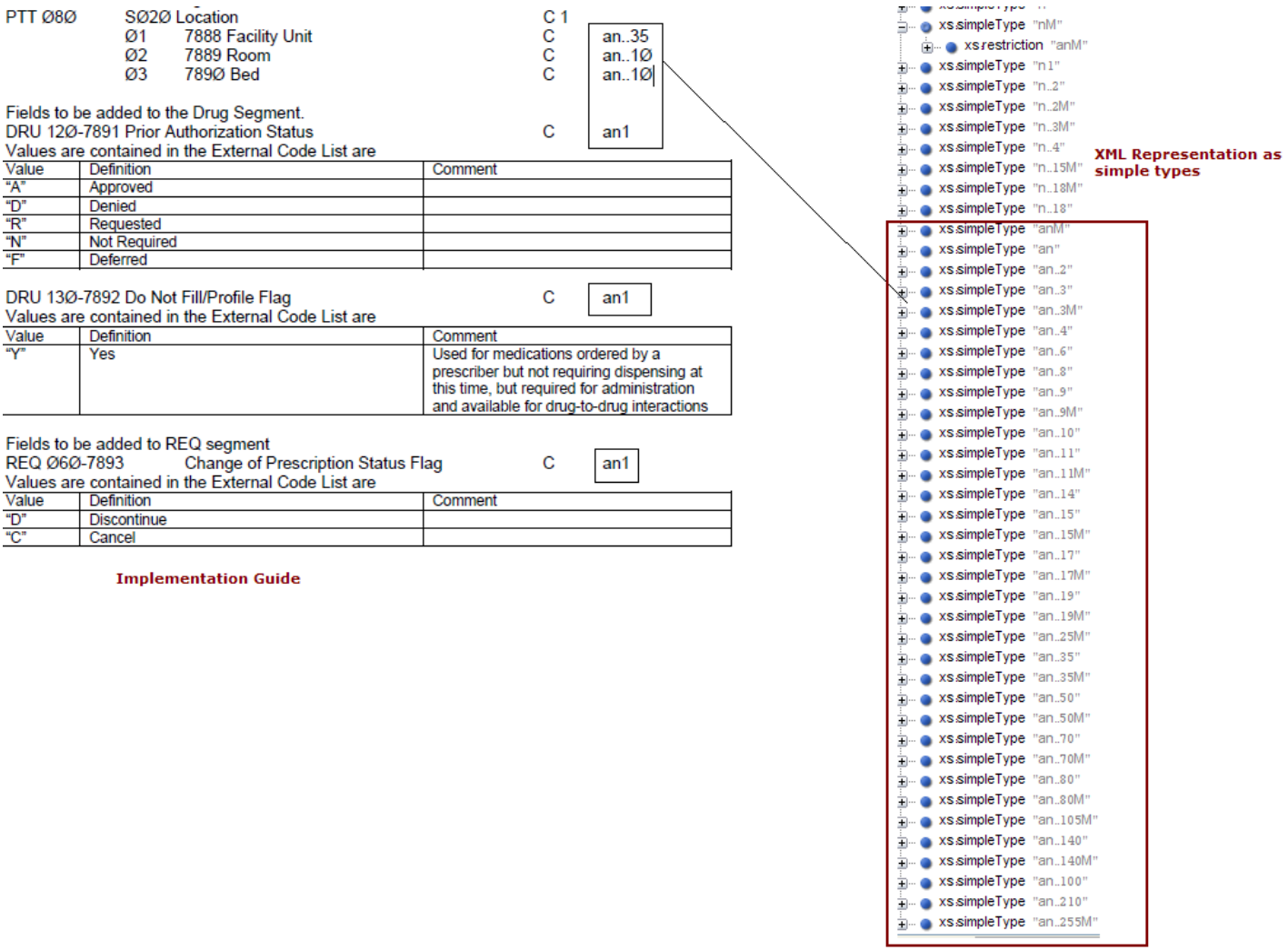


FIGURE 2: MWB DATA TYPE LIBRARY AND SCRIPT DATA TYPES

In an attempt to duplicate how the data types are represented and referenced in the original EDI IG, the XSDs contains a large number of types that are constrained versions of primitive data types (e.g. AN). Each time a primitive is used with minimum length, maximum length (e.g. AN 2..3), or is mandatory (AN 2..3M) (see Figure 3: Simple Types Specify min, max length and mandatory field as (IG and XSD)). Since the length and usage constraints are field-specific, it is not necessary to declare a new data type each time a type is constrained for a field. Instead, it would be better from a testing standpoint if any constraints related to field length and mandatory presence were added to the field definition itself. The field constraints would therefore include:

- data type
- minimum length
- maximum length
- mandatory indicator

Note that in the MWB library both AN (text) and N (numeric/integer) primitive types appear only once. Each numeric and text field has a minimum/maximum length, mandatory/conditional status, and other formatting constraints.



PTT Ø8Ø	SØ2Ø Location	C 1	
	Ø1 7888 Facility Unit	C	an..35
	Ø2 7889 Room	C	an..1Ø
	Ø3 789Ø Bed	C	an..1Ø

Fields to be added to the Drug Segment.
DRU 12Ø-7891 Prior Authorization Status — C — an1
Values are contained in the External Code List are

Value	Definition	Comment
"A"	Approved	
"D"	Denied	
"R"	Requested	
"N"	Not Required	
"F"	Deferred	

DRU 13Ø-7892 Do Not Fill/Profile Flag — C — an1
Values are contained in the External Code List are

Value	Definition	Comment
"Y"	Yes	Used for medications ordered by a prescriber but not requiring dispensing at this time, but required for administration and available for drug-to-drug interactions

Fields to be added to REQ segment
REQ Ø6Ø-7893 Change of Prescription Status Flag — C — an1
Values are contained in the External Code List are

Value	Definition	Comment
"D"	Discontinue	
"C"	Cancel	

Implementation Guide

FIGURE 3: SIMPLE TYPES SPECIFY MIN, MAX LENGTH AND MANDATORY FIELD AS (IG AND XSD)

Conditional Elements

Unlike other standards, NCPDP SCRIPT makes liberal use of the *conditional* designator for segments and fields to mean that a business rule may determine its presence. The use of this designator means different things for the EHRS when sending or receiving a transaction:

- When sending a message, the EHRS needs to have full knowledge of the condition and business rules governing a specific drug prescription (e.g. DEA schedule drug prescription rules)
- When receiving a query response the EHRS system is expected to make use of all information received with the understanding that each system responding may

have tracked and populated a different subset. Since it's not feasible to expect complete consistency across all pharmacy and clearing house systems, the EHRS must be able to process any subset of conditional fields.

The majority of fields in the NEWRX structure are conditional based on external business rules, rather than being based on the content of the message. This is a problem since the validity of the conditional predicates cannot be validated by the test harness without knowledge of the pre-conditions. For instance, the need for a supervisor may be determined by a DEA schedule drug prescription or by the fact that the prescriber is an intern. For the purposes of interoperability testing, the conditions must be based on predicates related to the message content. The pre-condition that a supervisor is required for a new drug prescription may be tested only as an explicit pre-condition through inspection while other conditional predicates may be verified using the contents of the NewRx. However, this means that a representative set of such pre-conditions must be identified. It is more difficult to determine whether a conditional element should be present if the condition is based on externally defined business rules. However, this type of validation is not impossible and it can be based on testing pre-conditions.

Processing a RXHRES transaction is somewhat different since this transaction provides all the known details and the receiving system should make all that information available to the end user. Therefore, the information is not specifically meant as *conditional* (i.e. its presence based on an external or explicit conditional predicate) but *optional/may be empty* (i.e. the system responding to the query may have the information and it will make it available if has it).

In conclusion, a conditional element is a very different concern for a sender or receiver:

- The sender should send it if certain conditions are met.
- The receiver must be able to process a conditional element especially if it is sent in response to a query.

Terminology

Implementing the SCRIPT standard presents both implementers and certifiers with a configuration management issue – the SCRIPT standard is maintained independently from the *External Code List* specification and the versions are not explicitly related to each other. There are other issues including:

- The XSD uses patterns instead of enumerated types; therefore, the vocabulary constraints are built into the type validation. This method of validation does not allow a separate terminology schema to be created each time the *External Code List* specification is changed.

- Coded concepts are not clearly delineated by their data type designation – they are represented as alphanumeric types with fixed lengths rather than maximum lengths. It would be beneficial to differentiate between coded and string data types (i.e., A..3).

- HITSP C80 is a good example of how terminology bindings are documented and maintained using static or dynamic terminology constraints. It specifies the source coding system and enumerates the coded values applicable to a concept.

Semantic Qualifiers

Many of the structured fields in the specification use semantic qualifiers drawn from the externally defined X12 value set. The XML representation of the SCRIPT-specified messages attempts to create specialized types corresponding to values of semantic qualifiers. For instance, the meaning of the DRU segment changes based on the value of its semantic qualifier. Thus, a DRU segment whose "Item Description Identification" qualifier is set to "P" refers to a prescribed drug and corresponds to a "**MedicationPrescribed**" XML element. Otherwise if the qualifier "Item Description Identification" is set to "D" then the DRU segment is equivalent to "**MedicationDispensed**" in XML. The XML schemas for SCRIPT 10.6 and 10.11 attempt to alleviate the ambiguities introduced by the overuse of semantic qualifiers by creating separate types for each qualifier value in some cases. In the XML representation of the SCRIPT message structure, some of the qualifiers are removed and replaced by specializations of the field (e.g. **Reference Number** and **Qualifiers** are represented by an **Identifier** type with a separate choice for each allowed qualifier value).

Identifiers

The convention used by NCPDP XSD designers is to represent each semantic qualifier as a separate specialization of that field. However, there is one significant exception to this rule – when the identifier for patient, prescriber, pharmacy, etc. is specified in XML as a choice. Each semantic qualifier corresponds to a single choice. The "Identifier" type is applied the same way to each type of object which means that it would be allowed to identify a supervisor by a Medical Record Number and patient by NPI. No efforts were made by the designers of the XSDs to create specialized identifier types that provide only the allowable choices for each type of field (e.g. licensed/billing providers, patient, unlicensed/non-billing clerk/proxy, etc).

The desire to reuse a single identifier (a type defined a choice) leads to a structure that allows invalid choices (e.g. a provider may be identified using a medical record number). This problem could be easily solved by declaring different types of identifier for the various types of participants in a prescription.

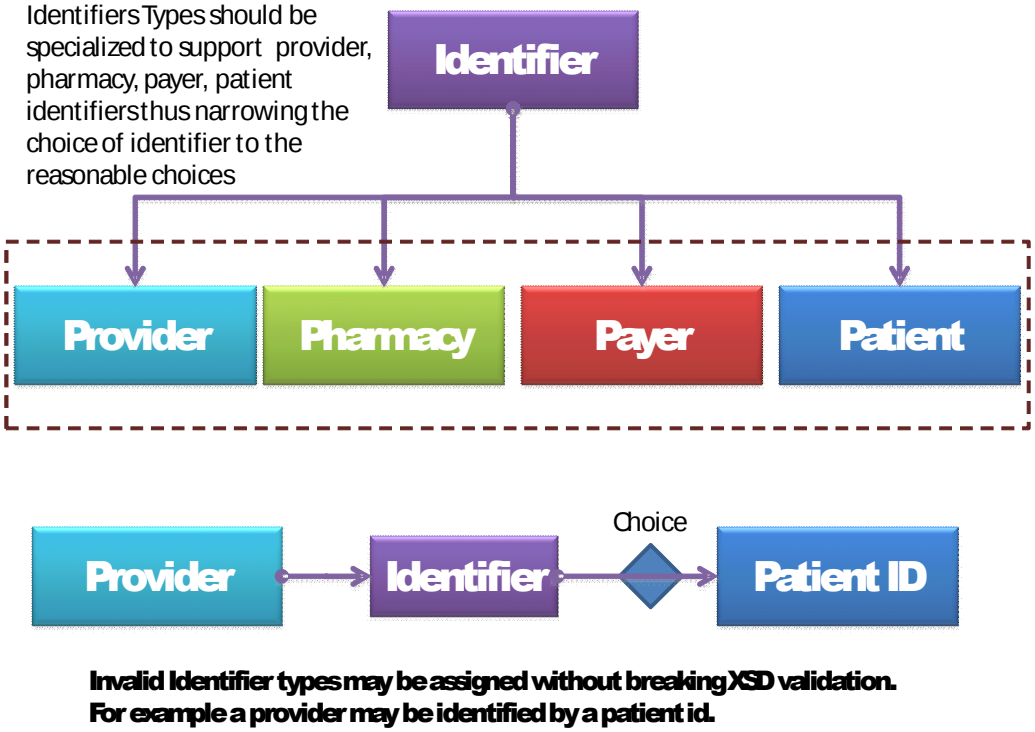


Figure 4: Identifier XSD types are ambiguous

Backwards-compatibility

Based on our analysis of versions 10.6 and 10.11, the implementation representation of the SCRIPT is not backwards compatible. Simple rules such as preserving the order of elements in a sequence have not been observed. On the other hand, version 10.11 appears to complete the transition to XML encoding and add some support for Meaningful Use by adding support for specifying the primary communication language of the patient.

Requirements Analysis Target

The SCRIPT IG is intended to support ePrescribing and Medication Reconciliation; therefore, it is important that the drug, its dose, and the timing of the dose be very precise. However, the NEWRX/NewRx transaction allows an EHRS to send a drug simply as a text description, including packaging and dose information. This lack of rigor implies that the prescriber does not need to know the drug code, specify a structured SIG, or structured dose information, thereby implying that an unstructured new drug prescription is completely acceptable. A further implication is that the drug codes are assigned in the pharmacy, based on approved formularies. Therefore the

EHRS needs to use a prescription history query to obtain the complete, structured drug record for medication reconciliation and problem list.

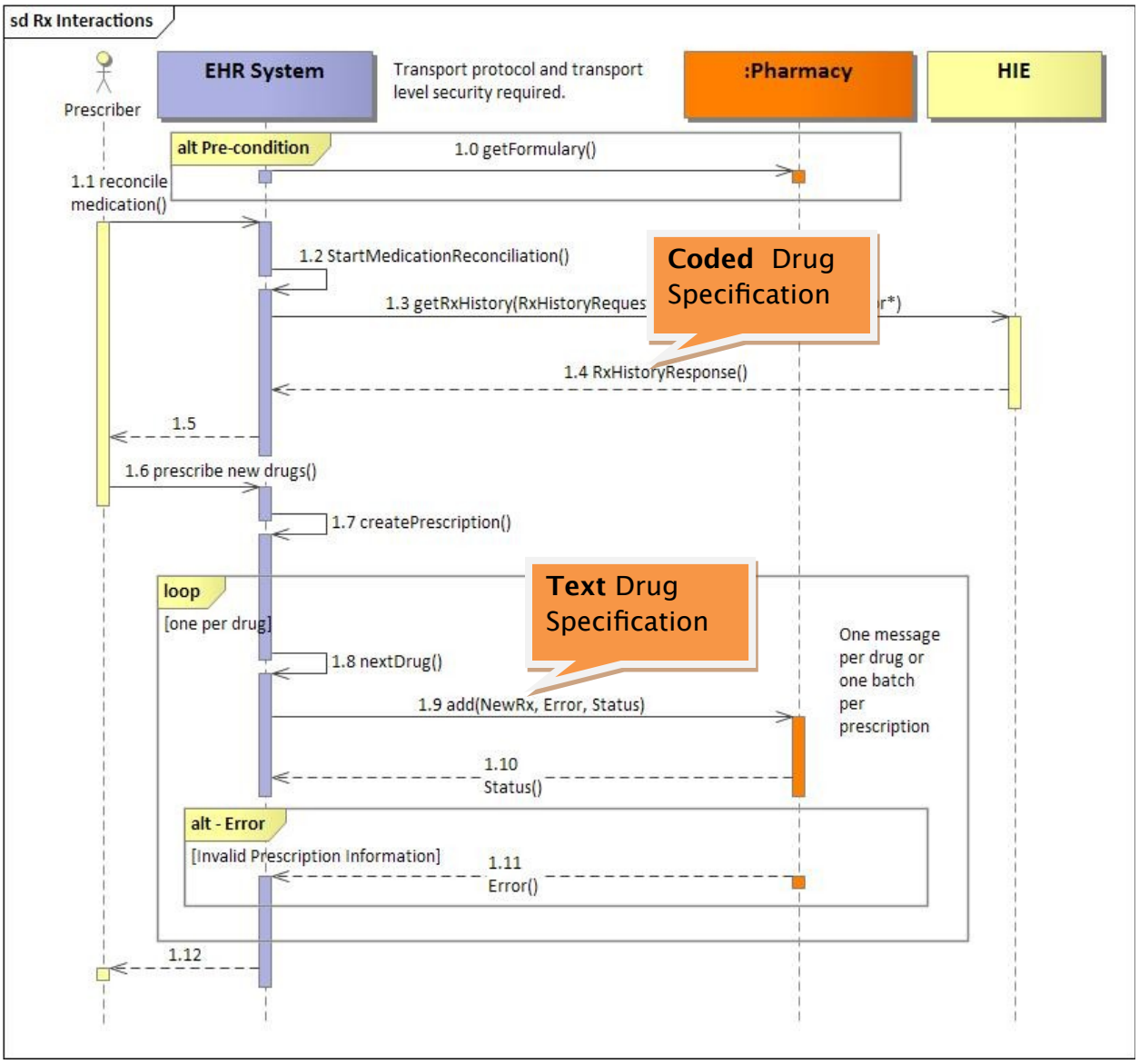


Figure 5: Unstructured Prescription and Structured Prescription History response

Loop Representation

Since the NIST testing infrastructure will validate messages encoded in XML, it will use the *NCPDP XML Schema Description* (XSD) specified for SCRIPT 10.6 and later. While the XSD is nominally based on the same implementation guide as the EDIFACT representations, the designers of the XML encoding had to make some assumption and

design decisions. Our analysis does not discuss these decisions except when those decisions appear to be ambiguous or conflict with the approach used elsewhere in the implementation guide.

1. The SCRIPT XSD for NEWRX does not support more than one medication. That is also the practice using EDI encoded SCRIPT transactions. However, the SCRIPT IG is ambiguous by specifying a DRU as a loop rather than a single occurrence:

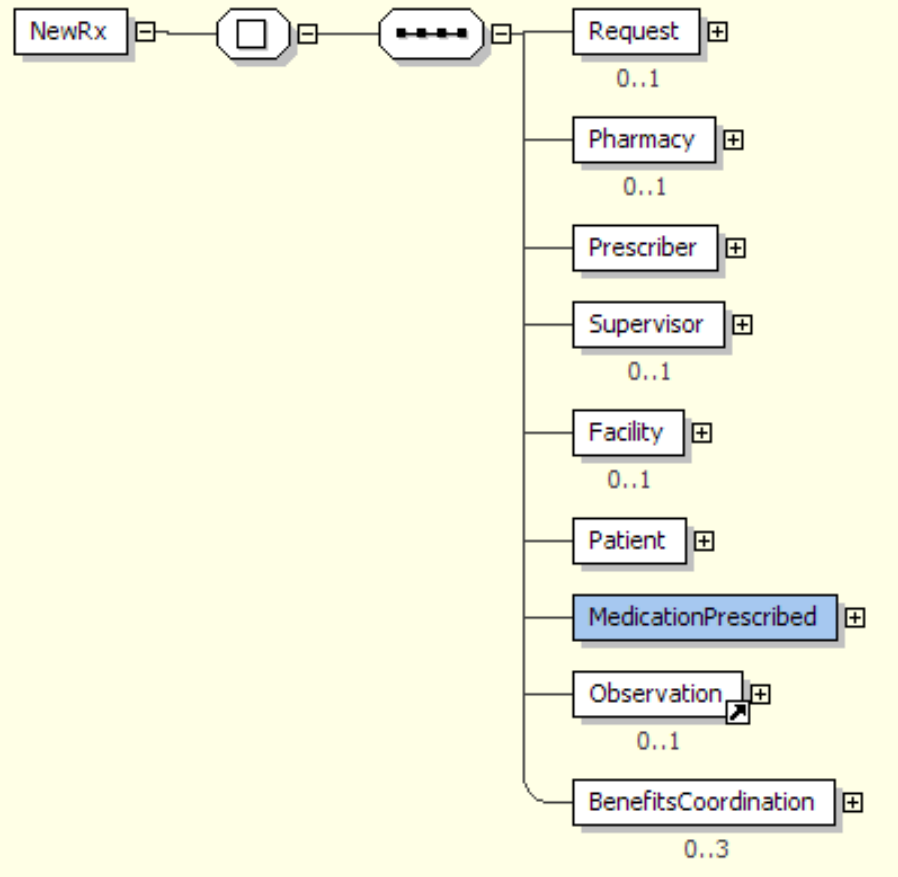


Figure 6: Version 10.6 and 10.11 MedicationPrescribed" element in a new prescription message

2. The XML definition for "NewRx" is unambiguous by specifying that the "MedicationPrescribed" element can occur exactly once in each instance. The SCRIPT IG indicates "one loop required for the drug prescribed". A loop by definition is a repeating construct, thus the IG implies repetition while the XSD explicitly prohibits it. Clearly both specifications mean to say the same thing – that only one **DRU** segment or **MedicationPrescribed** element may appear in a **NEWRX**/**NewRx** transaction. An implementation guide for the XML implementer would be useful in specifying that only one drug may be specified per NewRx message, by design.

SCRIPT Standard Implementation Guide Version 1Ø.6

SEGMENT	DEFINITION	REQUIRED	COMMENT
REQ	Request Segment	N	Only required in this transaction when a return receipt is requested.
PVD	Provider Segment	Y	One loop required for the prescriber. One loop may be needed to relay the supervisor. In long-term care settings, one loop may be the facility.
PVD	Provider Segment	N	One loop may be required for the pharmacy. In most implementations, the pharmacy PVD segment is required. In some specialized implementations, the prescriber loop is required, but the pharmacy loop is optional or not required.
PTT	Patient Segment	Y	Designates patient information.
DRU	Drug Segment	Y	One loop required for the drug prescribed.
SIG	Sig Segment	N	Segment(s) may be included to codify the SIG information.
OBS	Observation Segment	N	Segment only included if needed.
COO	Coordination of Benefits segment	N	Segment only included if needed. Might be used to convey to the pharmacy the patient's prescription program.
UIT	Interactive Message Trailer	Y	Designates the message trace number and number of segments in the message.
UIZ	Interactive Interchange Trailer	Y	Designates the interchange trace number and the number of messages in the transaction.

FIGURE 7: DRU SEGMENT IS SUPPOSED TO LOOP

Table 2 contains a snippet of the XML Schema Description specifying a mandatory but unique medication specification in each XML-encoded NewRx/New Prescription message corresponding to the Drug Segment loop above:

TABLE 1: XSD DEFINITION - MEDICATIONPRESCRIBED OR DRU LOOP

```
<xs:element name="MedicationPrescribed" type="NewRxPrescribedMedicationType">
      <xs:annotation>
            <xs:documentation>DRU-P; One loop required for the drug
prescribed. At least one loop must contain 85 = Date Issued (Written Date)
</xs:documentation>
            </xs:annotation>
      </xs:element>
</xs:element>
```

Recommendations and Conclusions

The following is a list of recommendations intended to improve the chances of successful implementation of NCPDP SCRIPT transactions required for ePrescribing and Medication Reconciliation:

- Resolve inconsistencies in the representation of commonly reused data structures. We recommend that NCPDP or NIST provide an explicit and detailed implementation guide for XML-encoded SCRIPT messages.
- We recommend that the terminology specification be part of the implementation guide and made available to EHRS vendors as one artifact in accordance with NCPDP licensing rules. According to NCPDP, its membership includes license to all billing-related codes referenced from X12, therefore, there should be no impediment to compiling a complete implementation reference and making it available as an artifact to licensed individuals.
- NCPDP primitive data types (e.g. text, numeric) are pre-coordinated with additional qualifier to indicate mandatory, minimum, and maximum length. This approach leads to complex and redundant set of data types in the XML Schema Description that are not reusable, as intended, but field-specific:
    - nM (mandatory numeric) because the field to which it is assigned is mandatory
    - n..3 (numeric, maximum three digits) because the field
    - n2..3 (numeric, between two and three digits in length).
- The usage and length attributes are logically specified by the field that is assigned the data type. There are business and semantic reasons why one numeric value may be of fixed or variable length. The semantics and business rules are specified by the field where the data type is applied.
- We recommend that a library of simple and complex data types with no mandatory usage or length constraints be created. This library could become the basis for applying global constraints for NEWRX and RXHRES transactions (e.g., constrain all numeric values to not exceed 999) and still allow for constraint of fields according to their own purpose. The data types may be constrained within the context of the field where they are assigned and made mandatory and constrained in length (minimum and/or maximum limit).

Appendix A: NEWRX Detailed Analysis

- script-newrx.htm

Appendix B: RXHRES Detailed Analysis

- script-rxhres.htm