



# US ARMY S1000D BREX Deliverable

## Issue Number 001

(supporting MIL-STD-3031, 30 SEP 2009)

# Overview



- US Army S1000D BREX (Issue 001)
- XSL Script
- EZ BREX Checker (v1)
- Army BREX DMC



# US Army S1000D BREX (Issue 001)

# US Army S1000D Business Rules EXchange (BREX)



- What is it?
  - An S1000D-authored XML file containing all possible machine-verifiable Army decisions (extracted from MIL-STD-3031)
- What is it used for?
  - Identifies conflicts with Army decisions in S1000D-authored data modules



# Army BREX

## DMC-USARMY-A-00-00-00-00000-022A-D

- Contains:
  - References to
    - S1000D Issue 4.0 Default BREX
      - The next higher level BREX must be referenced
      - Every DM must reference a BREX
    - MIL-STD-3031
  - Army (machine-verifiable) decisions
    - For example, prohibiting the use of the attribute vitalWarningFlag
  - NSN length validation (missing related BRs\*)
    - Verifies the attributes, which comprise the NSN, contain the applicable fixed digits (e.g., NATO codification bureau should only be two digits)
    - Verifies the element <fullNatoStockNumber> contains 13 digits, with or without hyphens

(\* To be included in MIL-STD-3031, Change 1)



# Army BREX Contents (Con't)

- Army-defined configurable attribute values (further limits the values allowed in the S1000D default BREX)

- Army-defined open ranges (further limits the S1000D open ranges)

- Projects defining new/additional values use open values allowed in the range

- 5.59.1.4 National caveat – attribute caveat.

Attribute values shall be used as defined in the following table.

Table VII. Attribute values – caveat

Allowable values	Army interpretation
cv01 - cv50	Not available for projects
cv51	For Official Use Only
cv52 - cv55	Reserved for Army
cv56 - cv99	Available for projects

ding values

# BREX Example & Log Generation



Prohibited use of:  
///`@vitalWarningFlag`

ERROR - Rule violated: (Prohibited inclusion of the attribute `vitalWarningFlag`.) The attribute `vitalWarningFlag` on the element `/warning/` shall not be used. (MIL-STD-3031 #5.17.1.7)

- The example above contains the object path ("`///@vitalWarningFlag`") and the object use as specified in the Army BREX.
  - The object use includes either an ERROR or WARNING declaration, followed by parenthetical guidance to correct the error, followed by the Army BR and parenthetical reference within MIL-STD-3031.
- Below shows an example of the output generated when the prohibited attribute is encountered.

```
-----  
ERROR - Rule violated: (Prohibited inclusion of the attribute  
vitalWarningFlag.) The attribute vitalWarningFlag on the element  
/warning/ shall not be used. (MIL-STD-3031 #5.17.1.7)  
-----
```

# Excluded from the Army BREX



- SNS rules – no equipment breakdowns apply
- Notations (additional format types)
  - Only one BR (#5.13.1.2) noted the use of JPG/GIF (BR did not limit to only those formats) – these are already allowed IAW S1000D
  - No default S1000D formats are prohibited
- Info Codes missing from Appendix B:
  - To be updated in the next release of MIL-STD-3031
    - ICs: 028, 078, ~~096~~, 0A1, 217, 236, 237, 297, 980, 981, 982, 989, 996
      - 0A1 is not an allowed DM type



# BREX Limitations



- Can check for existence of elements and attributes, but not necessarily for verbatim (boilerplate) or required content
- It will not verify compliance with every Army decision contained in MIL-STD-3031
- Unavoidable generation of (two) ‘warnings’ for every DM
  - Must verify manually, both unsupported DM types (pm and dml)



Swedish-developed XSL script

(brex2xsl)

# brex2xsl Script



- What is it?
  - A script which converts BREX-authored rules into XSL
- What isn't it?
  - Does not style or present the data module content
- What is it used for?
  - Validating S1000D-authored data modules



# Script Limitations

- The script (brex2xsl) allows for limited testing of only `<dmodule>` structures
  - Including more common types:
    - Descriptive, procedural, IPD, BREX, checklist, crew, fault, process, and maintenance planning
  - Including lesser used types:
    - Technical repository, applicability-related (ACT, PCT, CCT), container, learning, and wiring DMs
  - Excluding the following types:
    - Publication module (pm), data module list (dml), data dispatch notes (ddn), comment, and SCORM content

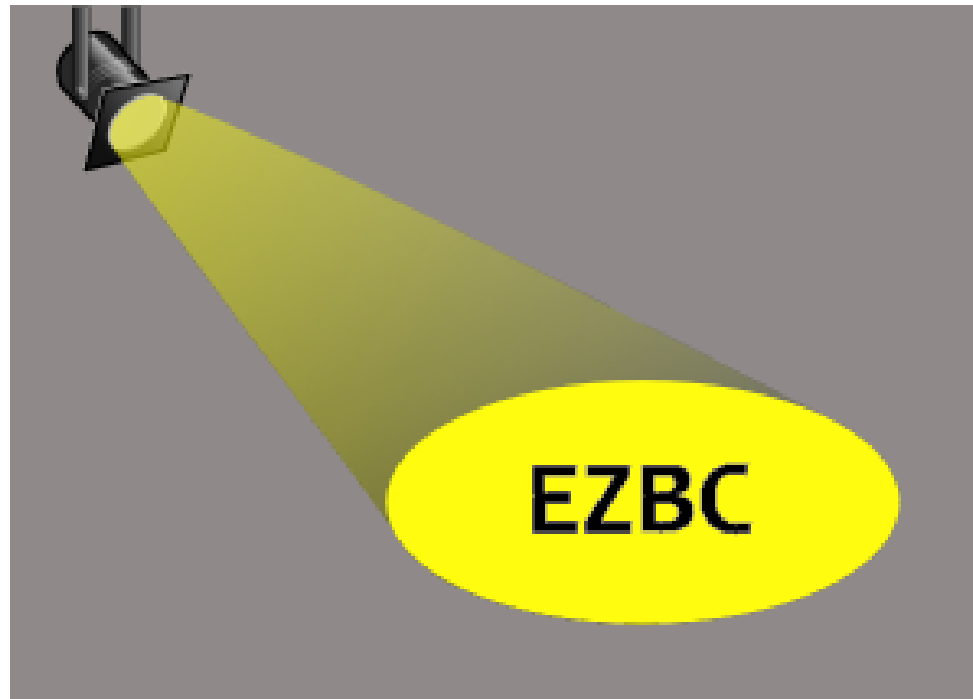


# Script Limitations (con't)

- Command line use, one file at a time
  - Time-consuming
- Risk of multiple instances of script and BREX, if used in multiple directories
- Ability to specify a schema is not supported
  - This could apply when rules for preliminary requirements differ between procedural and fault DMs
- Log output limitations
  - Quantity of like errors or location of those errors is not generated
    - Does not include “five procedural steps are missing an ID” - only that the ID is missing from the element `<proceduralStep>`
  - Set error phrases ('Rule violated' and 'Value out-of-range')
  - Text only



## EZ BREX Checker (v1)



# EZ BREX Checker (EZBC)



- What is it?
  - EZBC is a user-friendly front-end for use with the (Swedish) XSL script
- What is it used for?
  - To simplify BREX validation and provide a few extra features beyond those within the script

# EZBC



- Current Features
  - Graphical User Interface (GUI)
  - Configuration file
    - Specify location of script
    - Specify BREX
    - Specify an output directory
    - File selection location
  - Quantity and countdown of selected files!
  - Ability to ignore or disable test generation for “deleted” entries
- Benefits
  - Multi-file selection
  - Single instance of a BREX
  - Single instance of script
  - Output directory allows separate directories for each project or pub type, if desired
- Desired Features
  - Selectable XSL engine for XSL conversion
  - Support other DM types (even though `<brexDmRef>` is not included, many BRs apply)
    - Allows for validation of pm, dml, etc.
  - Project-configurable dictionary
- Desired Log Output
  - HTML output (linking to errors - BREX source and/or file location)



# EZBC



- Package Content

- Issue 2.2 XSL

- Created by BTAS using the Issue 4.0 XSL
    - Also works with Issue 3.0

- Issue 4.0 XSL (BTAS-modified)

- Instructions for

- EZBC usage, and
      - MSXML6 acquisition
      - MSXSL acquisition
      - Java Runtime Environment (JRE) acquisition
    - Default S1000D BREX acquisition
    - Bike Sample acquisition



# BREX Releases

## Initial and Subsequent

# Army BREX Data Module Code



- Issue
  - Difficult to identify the applicable S1000D-related issue and MIL-STD-3031 related version to which the BREX applies
- Solution
  - The primary policy document will be identified using the Disassembly Code (DC) and DC Variant (DCV) in the Data Module Code (DMC).
  - Army's primary policy document is MIL-STD-3031.
  - MIL-STD-3031 clearly states the applicable S1000D Issue in its Foreword and Scope.



# Army BREX Data Module Code

- DMC-USARMY-A-00-00-00-**0000**-022A-D\_001-00\_EN-US
  - Use **DC** to indicate **revisions**
  - Use **DCV** to indicate **changes**
- **0000** – Disassembly Code/DC Variant – MIL-STD-3031 version
  - Examples
    - **0000** = 3031 Base version (MIL-STD-3031)
      - DMC-USARMY-A-00-00-00-**0000**-022A-D\_001-00\_EN-US
    - **0001** = Base version, Change 1 (MIL-STD-3031, C1)
      - DMC-USARMY-A-00-00-00-**0001**-022A-D\_001-00\_EN-US
    - **0A000** = 1<sup>st</sup> revision (MIL-STD-3031A)
      - DMC-USARMY-A-00-00-00-**0A000**-022A-D\_001-00\_EN-US
    - **0B004** = 2<sup>nd</sup> Revision, Change 4 (MIL-STD-3031B, C4)
      - DMC-USARMY-A-00-00-00-**0B004**-022A-D\_001-00\_EN-US
- Any necessary corrections would increment the issue number for the applicable file



# Army BREX Updates

# Army BREX Updates



- Army BREX
  - Include all applicable changes to MIL-STD-3031 C1
    - DMC/Filename:
      - DMC-USARMY-A-00-00-00-00001-022A-D\_001-00\_EN-US
    - Still applicable to Issue 4.0/4.0.1
  - Available ~2Q 2010 (dependent on MIL-STD-3031 C1 release date)