



Lua S-100 Portrayal Processing

SPAWARSYSCEN LANT

August 31, 2016

David.Grant1@navy.mil

+1-757-541-5794

What is an Extension Language?

A scripting language
interpreter

Provides an Application
Programming Interface

Has access to the
applications data
structures

Application can usually
access the extension
language data structures

XSLT as an ECDIS Extension Language

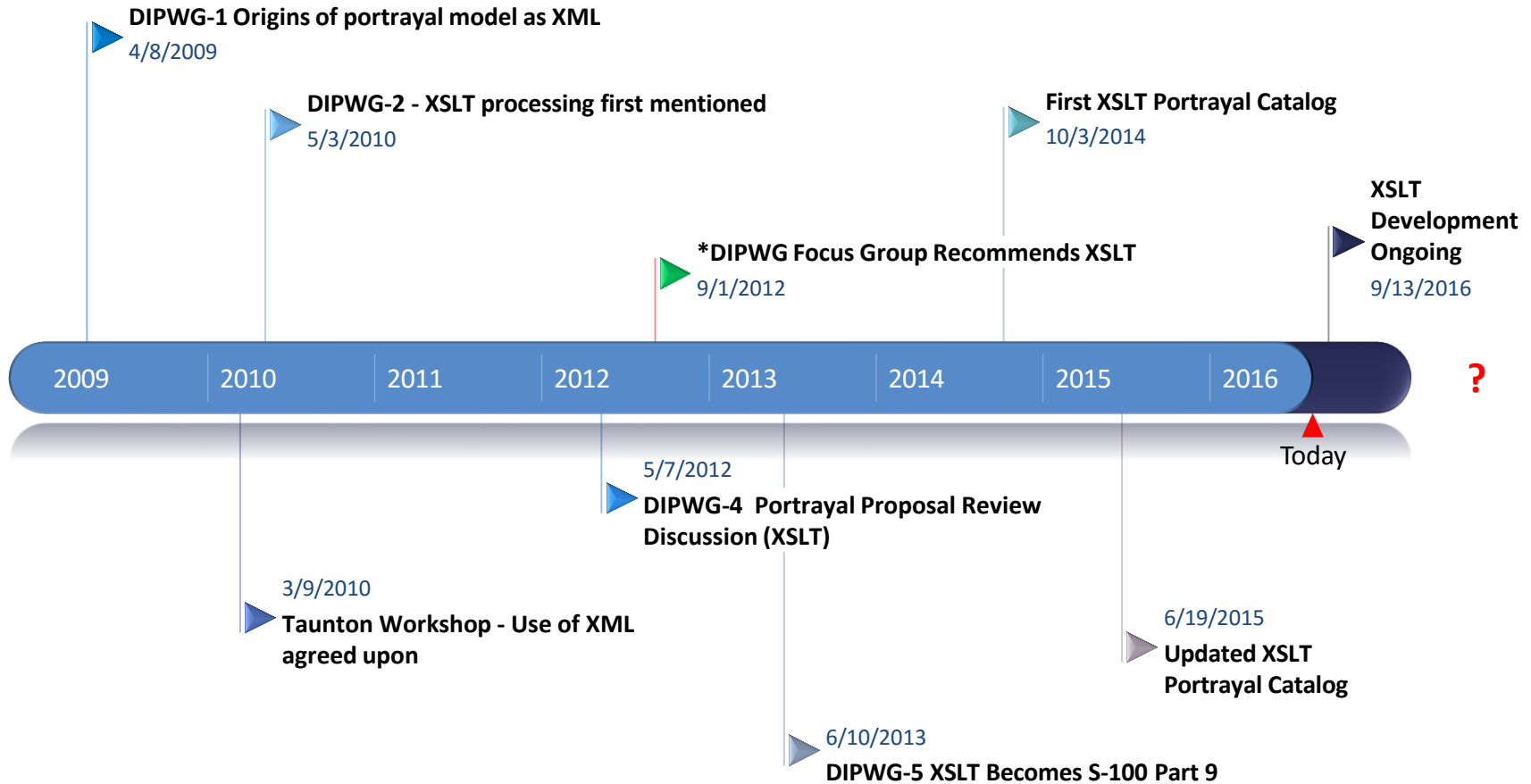
S-100 Part 9 uses XSLT 1.0 as an extension language to provide portrayal processing for an ECDIS system

XSLT 1.0 does not facilitate simple implementation of the state-machines used to define the conditional symbology procedures

If conditional symbology procedures are not implemented in the portrayal, they will not be available for extension

Alerts and Indications require complex spatial processing on feature instances which are the output of conditional symbology procedures

XSLT Portrayal Development Timeline



Proposed Portrayal Changes

Augment or replace XSLT as the S-100 portrayal extension language

Eliminate the portrayal input schema. Portrayal scripts will directly access the S-100 feature model

Goals of Augmenting or Replacing XSLT

Output an XML document as currently defined in S-100 Part 9 to maintain compatibility with previously developed applications and tools

Facilitate simple portrayal implementation within a reasonable timeframe

Permissive license (MIT preferred)

Easily embedded in C, C++, C# and Java applications

Small footprint (<1MB)

Be extensible through application provided “hooks”, thereby allowing spatial and other advanced processing

Why Did We Choose Lua?

Lua, JavaScript, and Python are scripting languages which could be used as ECDIS extension languages

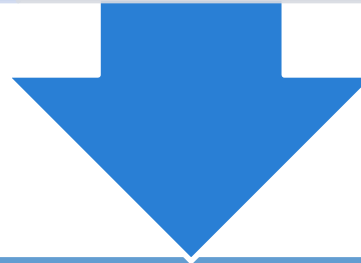
Lua is a proven, robust language

Lua is fast

Lua is portable & embeddable

Lua has a small footprint and is simple yet powerful

Lua has greater than 50% market penetration in the gaming industry as an extension language



We have chosen to implement a Lua proof-of-concept portrayal

What is Lua?

Lua is a powerful, efficient, lightweight, embeddable scripting (extension) language

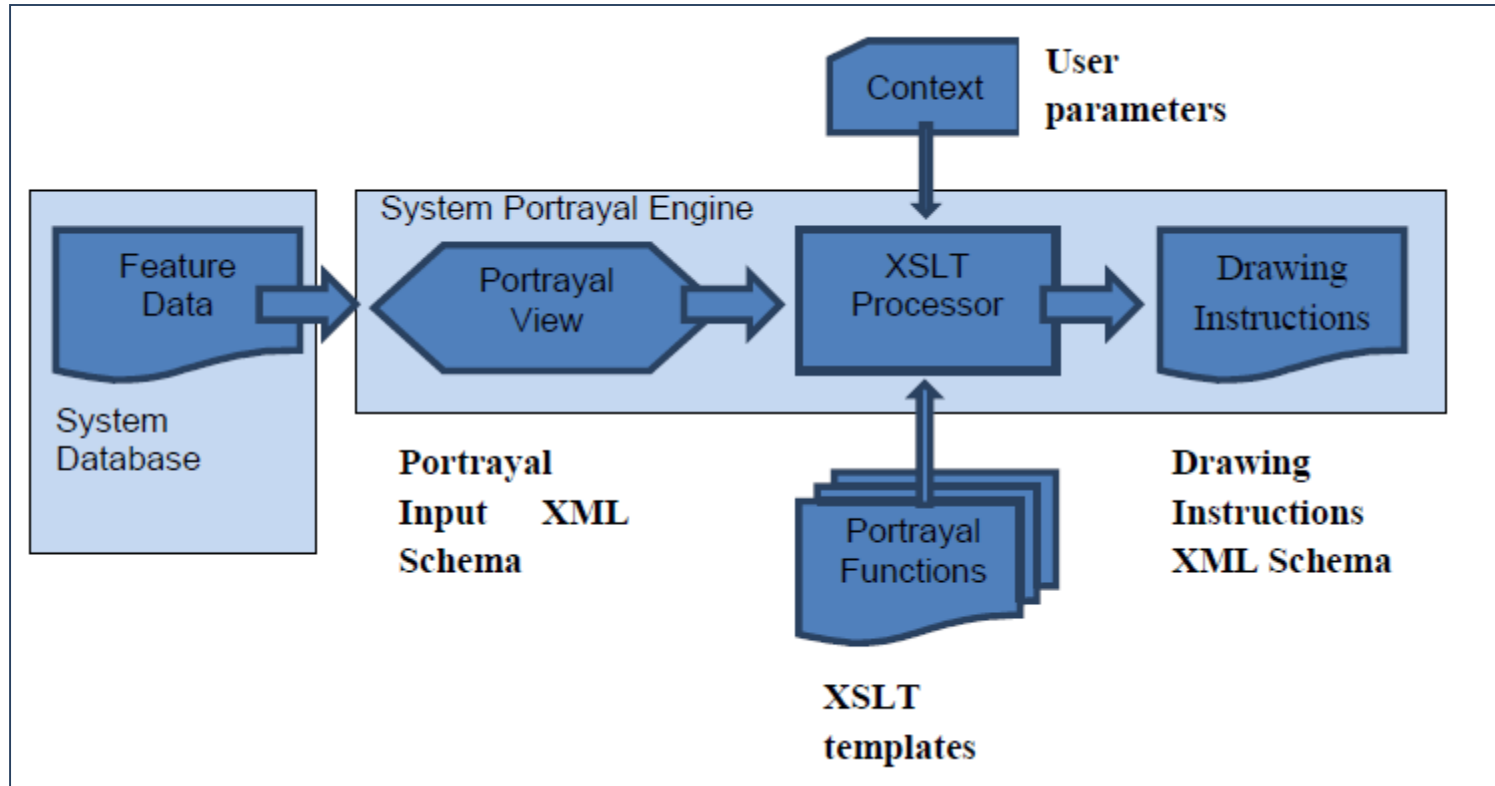
"Lua" (pronounced **LOO-ah**) means "Moon" in Portuguese

- It is neither an acronym nor an abbreviation

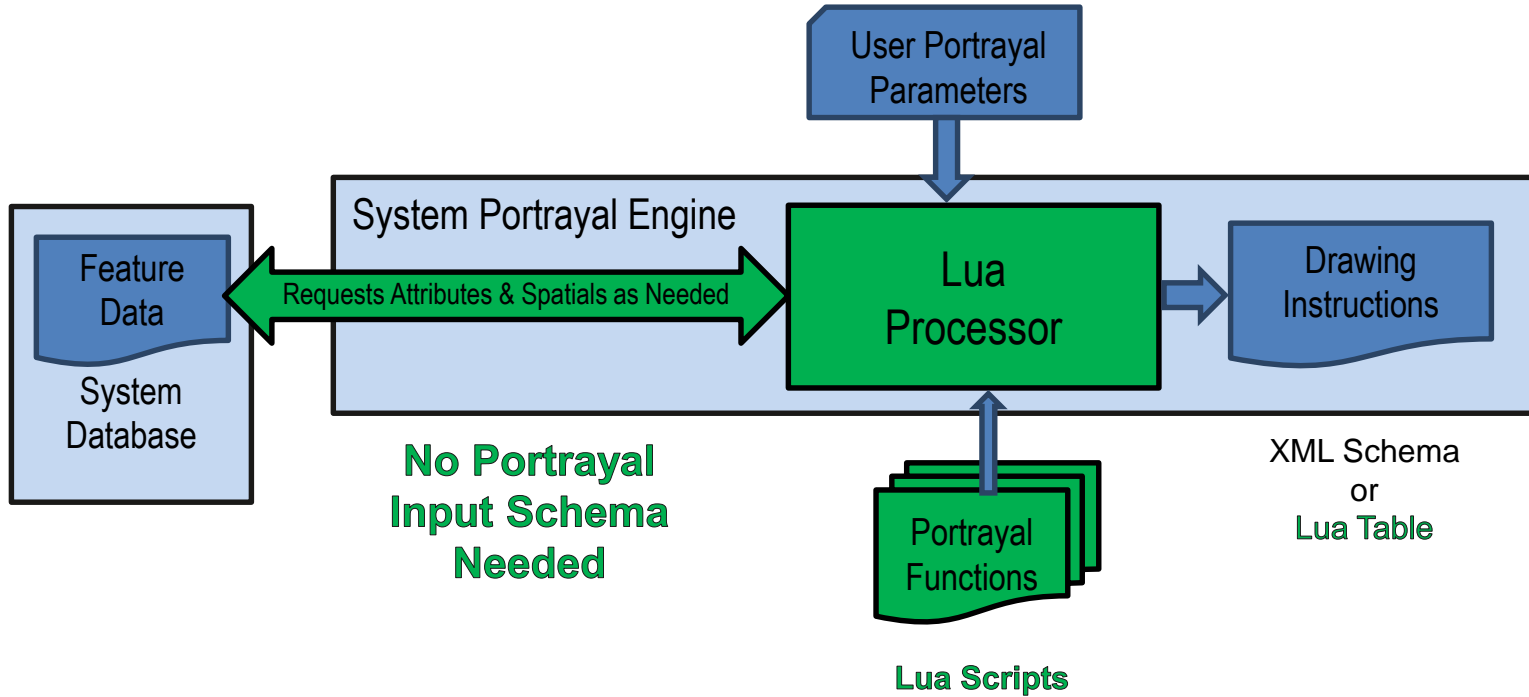
It supports procedural programming, object-oriented programming, functional programming, data-driven programming, and data description

Learn more about it at:
www.lua.org/about.html

Existing Portrayal Process (XSLT)



Proposed Portrayal Process (Lua)



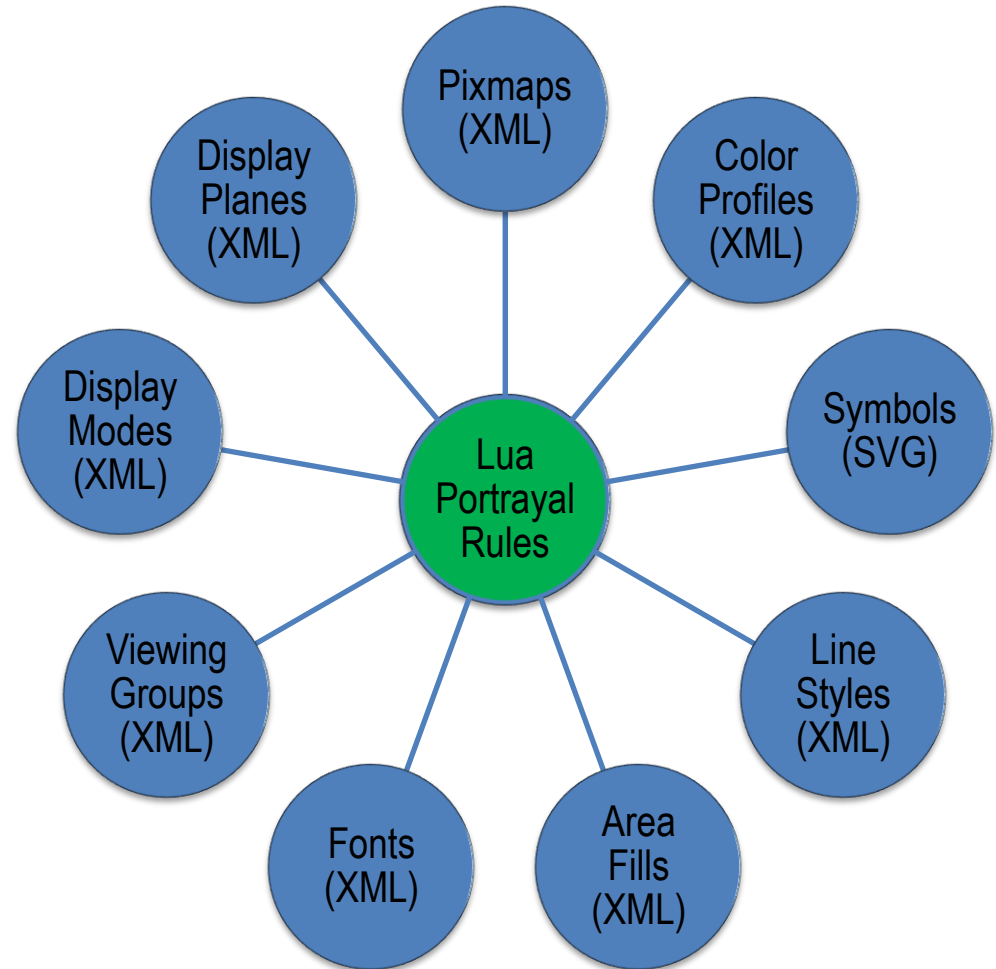
Portrayal Catalog Changes (Only Rules Change)

Remove Input Schema

Implement XSLT rules as Lua scripts

Support files are unchanged

Output is (optionally) unchanged



What we did with Portrayal and Lua

Integrated Lua with S-100 Viewer

- Any language that supports a C level call interface can integrate with Lua.
- Lua can integrate with C, C++, C#, Java, Python, etc.

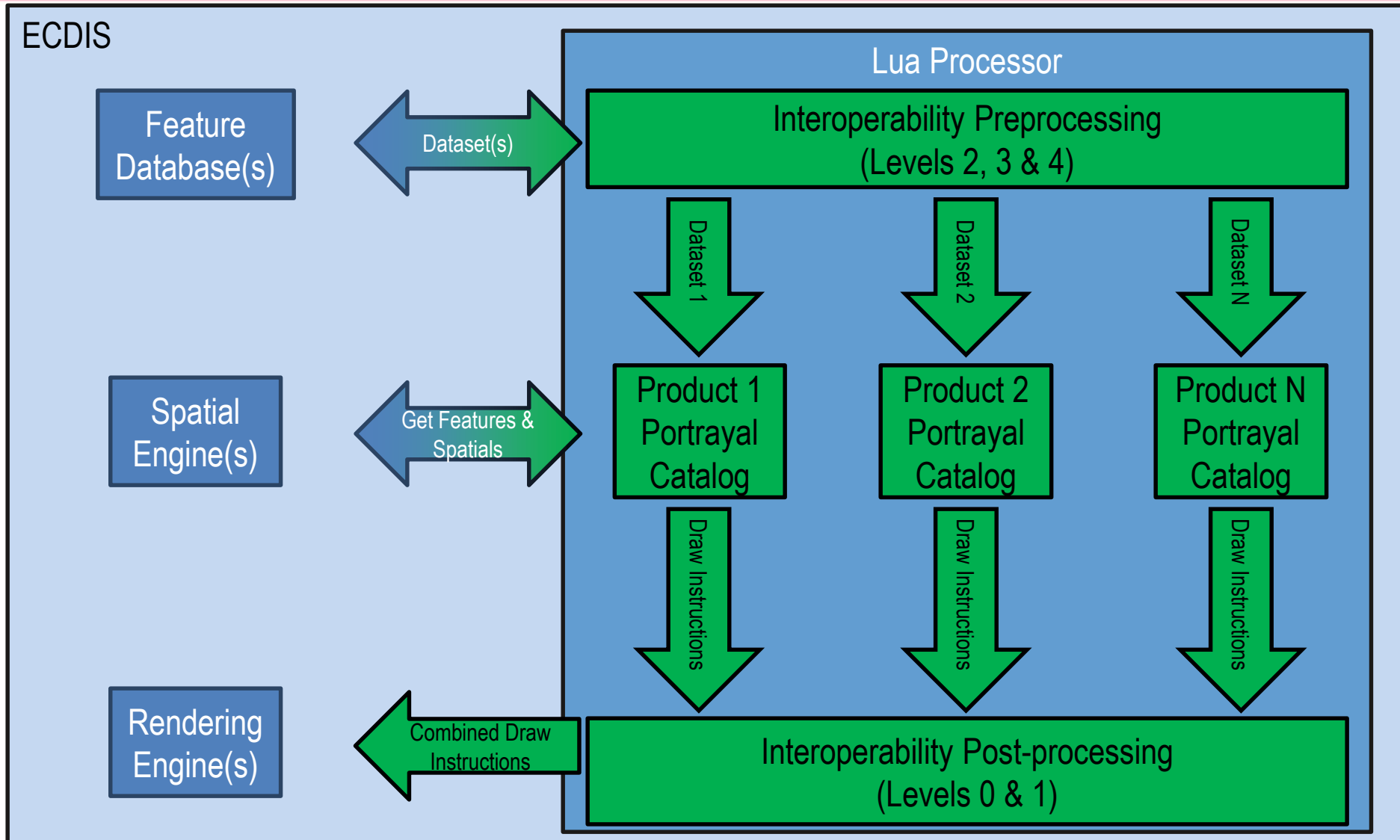
Determine attribute values and spatial associations “on the fly”

- Don't need a mechanism to describe what attributes are expected by the portrayal, simplifying implementation.
- Don't have to build a large DOM (XML) to feed the portrayal.
- Only need the feature type, primitive type and unique ID; anything else is requested by the portrayal “on the fly”.

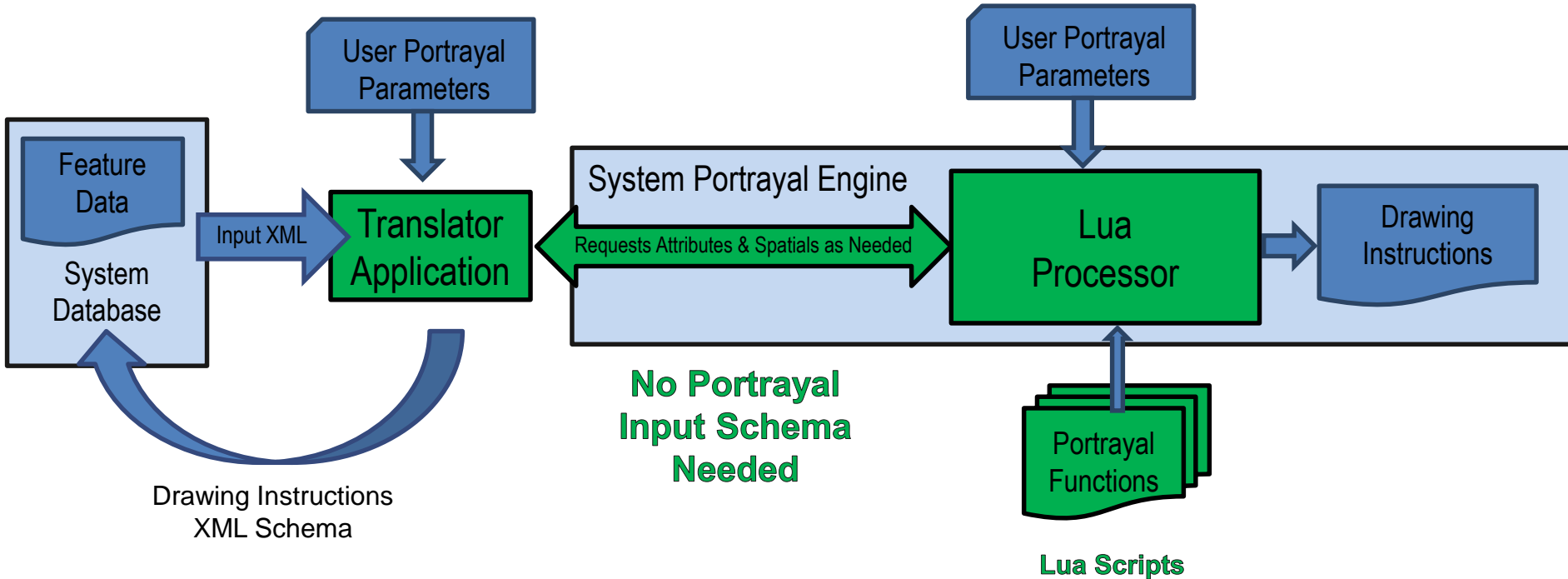
Took a “Feature Oriented” approach

- Drawing instructions are grouped by their corresponding feature.
- Allows for advanced techniques such as draw instruction caching and can be used to assist with pick reports.

Lua Rendering Pipeline - Interoperability Support



Backwards Compatibility



Lua Implementation Status

Implementing in
S-100 Viewer

No short-comings
encountered to
date

Available on
Basecamp

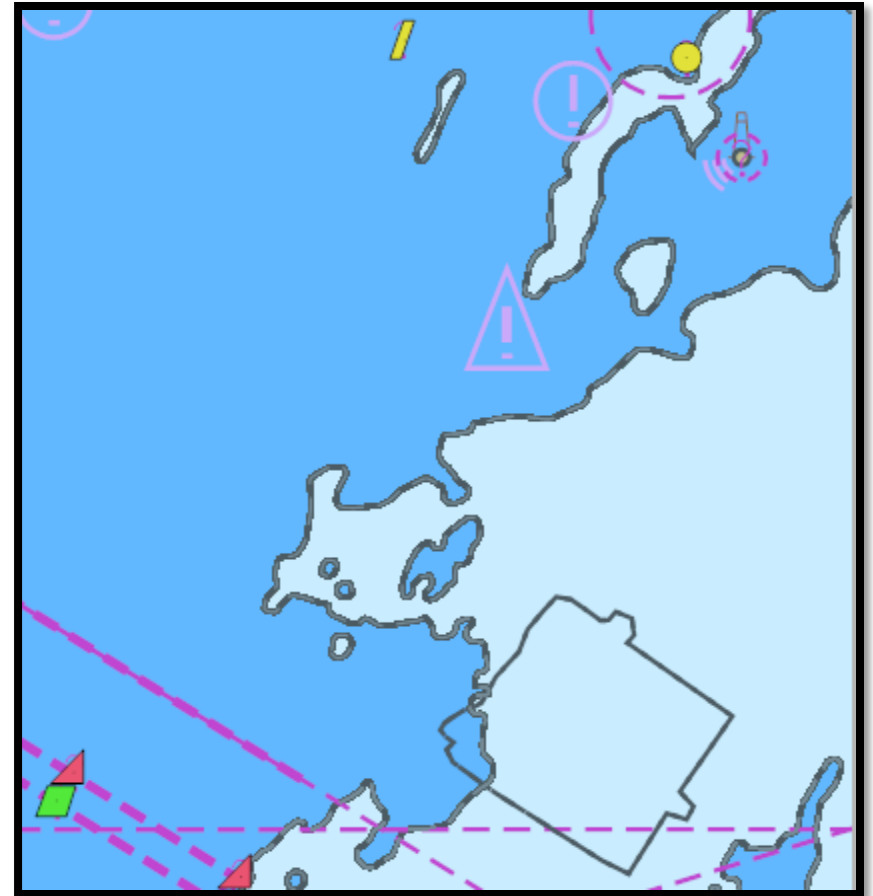
CSP Implementation Status

CSP	XSLT	* Notes	Lua	* Notes
DATCVR02	X		X*	Will be implemented as part of multi-product rendering.
DEPARE02 / 03	X*	Incorrect safety contour generated. No contour labels.	✓	
DEPCNT03	X		✓	
DEPVAL02	✓		✓	
LIGHTS05 / 06	partial*	Leg-lines incorrect, arcs incorrect	X	
OBSTRN06 / 07	✓		✓	
QUAPOS01	partial*	Spatial quality not implemented.	X	
QUALIN01	partial*	Spatial quality not implemented.	X	
QUAPNT02	X		✓	
RESARE03 / 04	X		✓	
RESTRN01	X		✓	
RESCSP02	X		✓	
SAFCON01	X		✓	
SLCONS03 / 04	partial*	Spatial quality not implemented.	partial	
SEABED01	✓		✓	
SNDFRM03 / 04	✓		✓	
SOUNDG02 / 03	X*	Implemented incorrectly.	✓	
UDWHAZ04	X		✓	
WRECKS04 / 05	✓		✓	

CSP DEPARE02 / 03



XSLT



Lua

Lua Portrayal Advantages

Lua was designed to be an extension language

S-101 CSPs are easily implemented in a timely manner

Complex portrayals which may be required for new products are easily implemented

Removal of input schema simplifies application development

Removal of input schema simplifies portrayal catalog maintenance

Facilitates implementation of Alerts and Indications

Facilitates implementation of Product Interoperability