Latest Features in MATLAB Coder

March 2017

R2017a

Value Classes as Entry-Point Function Arguments

Generate code using value class objects as inputs/outputs of entry point functions

- Limitations
  - Objects of handle classes cannot be entry-point function inputs
  - `coder.Constant` does not support objects
  - An object cannot be an input or output argument for an extrinsic function
  - An object cannot be a global variable

```
classdef myRectangle
    properties
        length;
        width;
    end
    methods
        function obj = myRectangle(l,w)
            if nargin > 0
                obj.length = l;
                obj.width = w;
            end
        end
        function area = calcarea(obj)
            area = obj.length * obj.width;
        end
    end
end
```

```
function z = getarea(r)
%#codegen
z = calcarea(r);
end
```

```
rect_obj = myRectangle(4,5)
codegen getarea -args {rect_obj} -report
```

Nested Functions

Generate code for nested functions

- Supports nested functions, including reading and writing to variables defined in an outer function

- Limitations
  - Avoid calling `coder.varsize` on a variable referred to in a nested function
  - Avoid incrementally initializing a structure variable referred to in a nested function

```
function main(n)
    f = enumFrom(n);
    disp(f()); % n
    disp(f()); % n + 1
end
```

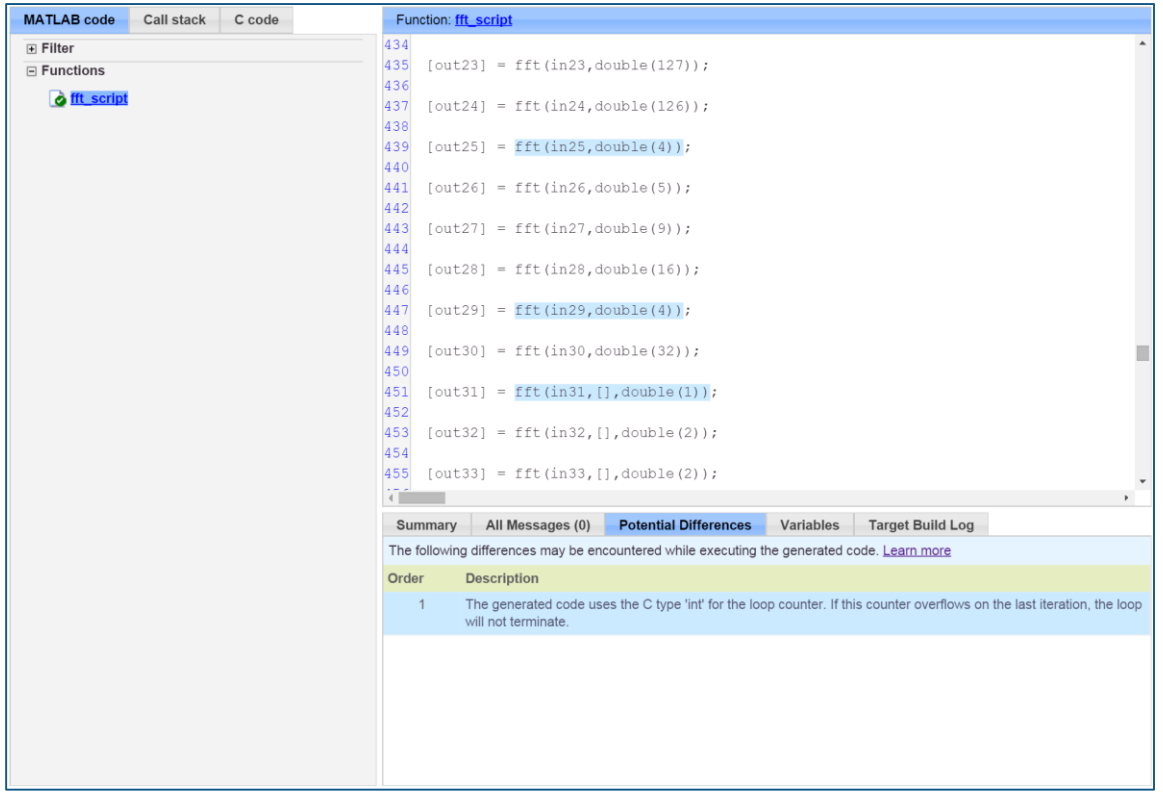
```
function f = enumFrom(n)
    function result = incr
        result = n;
        n = n + 1;
    end
    f = @incr;
end
```

```
function foo
    x = [];
    coder.varsize('x');
    function baz
        use(x);
    end
    baz;
end
```

```
function foo
    x.a = 1;
    x.b = 2;
    function baz
        use(x);
    end
    baz;
end
```

Potential Differences Reporting

Identify MATLAB code that might behave differently in generated code



Identifies potential differences such as:

- Automatic Dimension Incompatibility
- mtimes No Dynamic Scalar Expansion
- Matrix-Matrix Indexing
- Vector-Vector Indexing
- Size Mismatch

```
» c = coder.config('mex');
» c.ReportPotentialDifferences = true;
» c.GenerateReport = true;
» codegen('-config', c, ...)
```

Automated Driving System Toolbox Code Generation

Generate code for sensor fusion and tracking workflow

cameas	constvel	getTrackPositions	Initctukf	trackingEKF
cameasjac	constveljac	getTrackVelocities	Initcvekf	trackingKF
constacc	ctmeas	initcaekf	Initcvkf	trackingUKF
constaccjac	ctmeasjac	initcakf	Initcvukf	
constturn	cvmeas	initcaukf	multiObjectTracker	
constturnjac	cvmeasjac	initctekf	objectDetection	

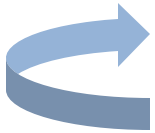
Loop Invariant Code Motion

Generate optimized code for loops

- Hoists invariant constructs out of for-loops and while-loops, including:
  - Invariant statements
  - Invariant if/else constructs
  - Invariant switch-case constructs
  - Invariant loops

R2016b

```
for (k = 0; k < 33; k++) {
    tmp[99] = 17.0;
    X[k] += 1.0 + (real_T)k;
  }
```



R2017a

```
tmp[99] = 17.0;
for (k = 0; k < 33; k++) {
    X[k] += 1.0 + (real_T)k;
  }
```