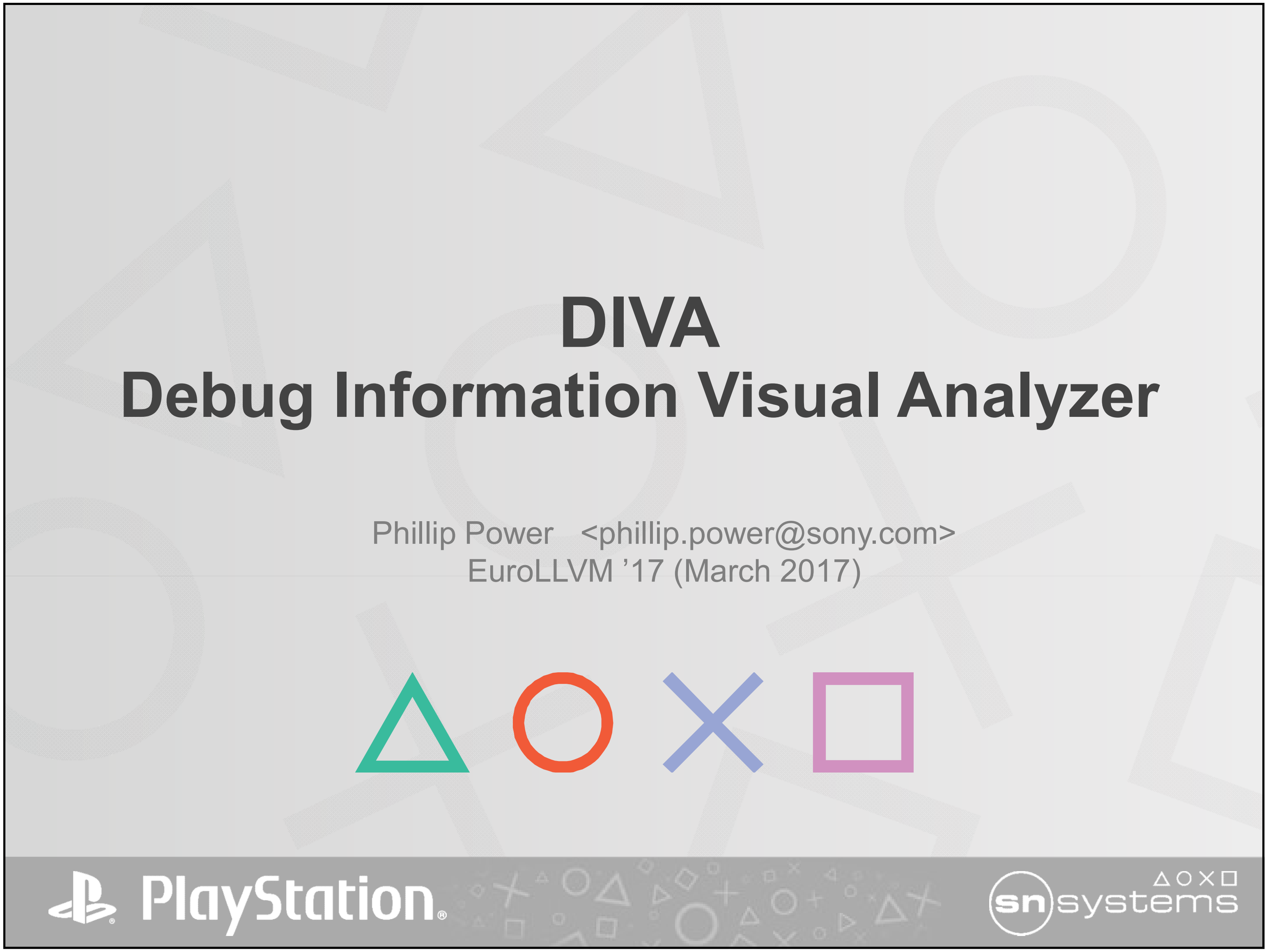
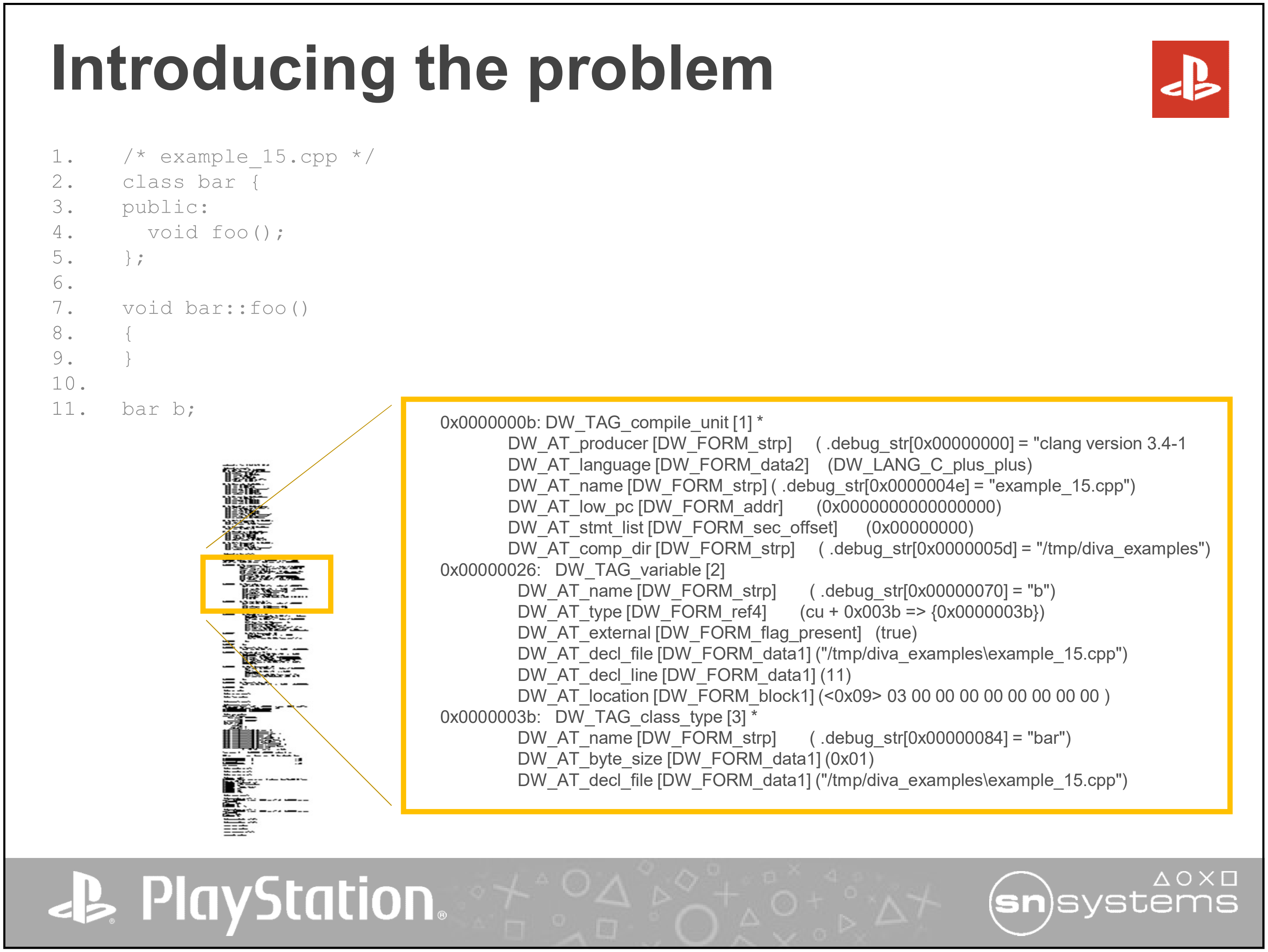


DIVA
Debug Information Visual Analyzer

Phillip Power <phillip.power@sony.com>

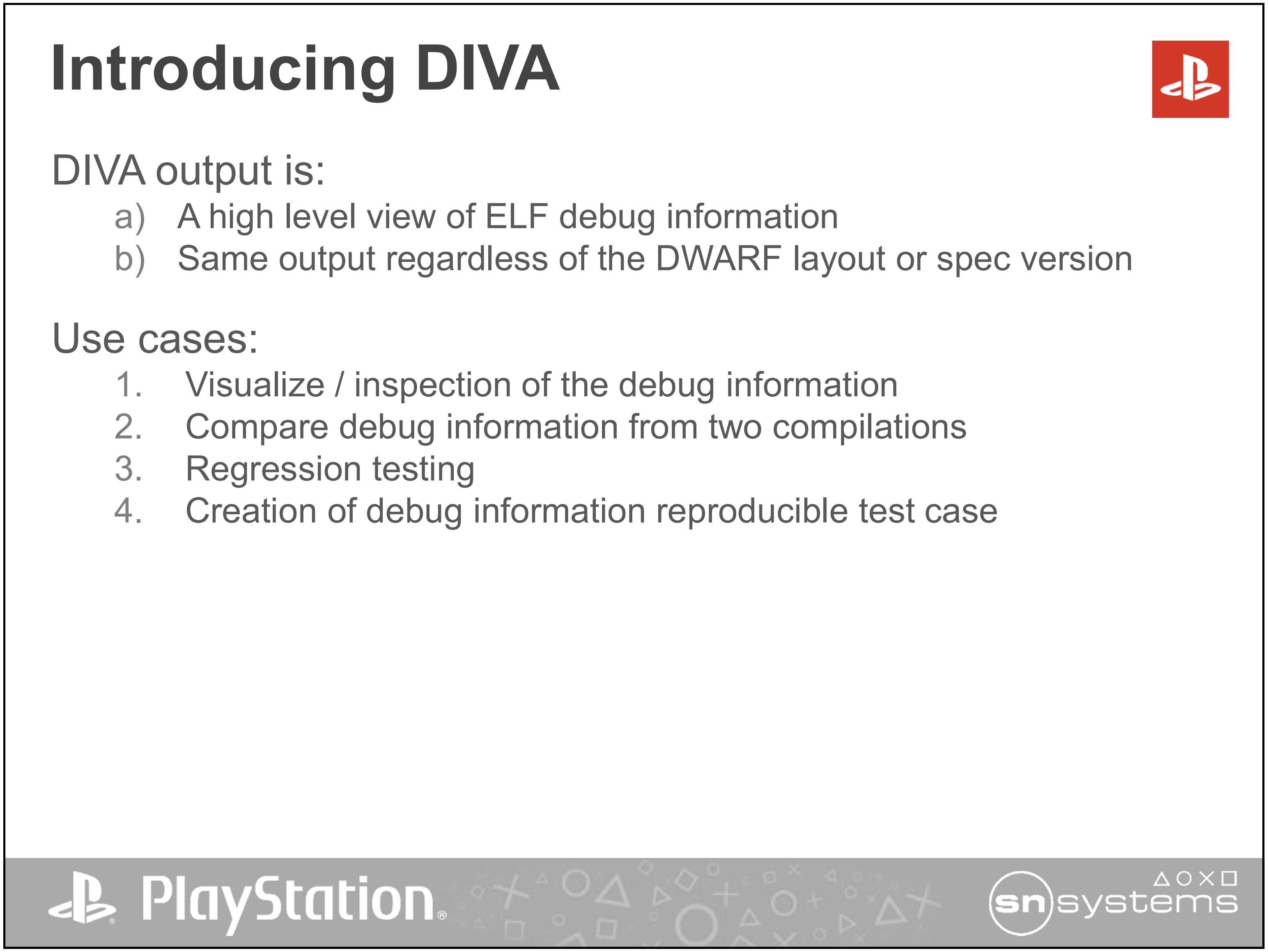
EuroLLVM '17 (March 2017)



Introducing the problem

```
1.    /* example_15.cpp */
2.    class bar {
3.    public:
4.      void foo();
5.    };
6.
7.    void bar::foo()
8.    {
9.    }
10.
11.   bar b;
```

```
0x0000000b: DW_TAG_compile_unit [1] *
        DW_AT_producer [DW_FORM_strp]    ( .debug_str[0x00000000] = "clang version 3.4-1
        DW_AT_language [DW_FORM_data2]    (DW_LANG_C_plus_plus)
        DW_AT_name [DW_FORM_strp] ( .debug_str[0x0000004e] = "example_15.cpp")
        DW_AT_low_pc [DW_FORM_addr]      (0x0000000000000000)
        DW_AT_stmt_list [DW_FORM_sec_offset]      (0x00000000)
        DW_AT_comp_dir [DW_FORM_strp]    ( .debug_str[0x0000005d] = "/tmp/diva_examples")
0x00000026:   DW_TAG_variable [2]
         DW_AT_name [DW_FORM_strp]      ( .debug_str[0x00000070] = "b")
         DW_AT_type [DW_FORM_ref4]      (cu + 0x003b => {0x0000003b})
         DW_AT_external [DW_FORM_flag_present]   (true)
         DW_AT_decl_file [DW_FORM_data1] ("/tmp/diva_examples\example_15.cpp")
         DW_AT_decl_line [DW_FORM_data1] (11)
         DW_AT_location [DW_FORM_block1] (<0x09> 03 00 00 00 00 00 00 00 00 )
0x0000003b:   DW_TAG_class_type [3] *
         DW_AT_name [DW_FORM_strp]      ( .debug_str[0x00000084] = "bar")
         DW_AT_byte_size [DW_FORM_data1] (0x01)
         DW_AT_decl_file [DW_FORM_data1] ("/tmp/diva_examples\example_15.cpp")
```



Introducing DIVA

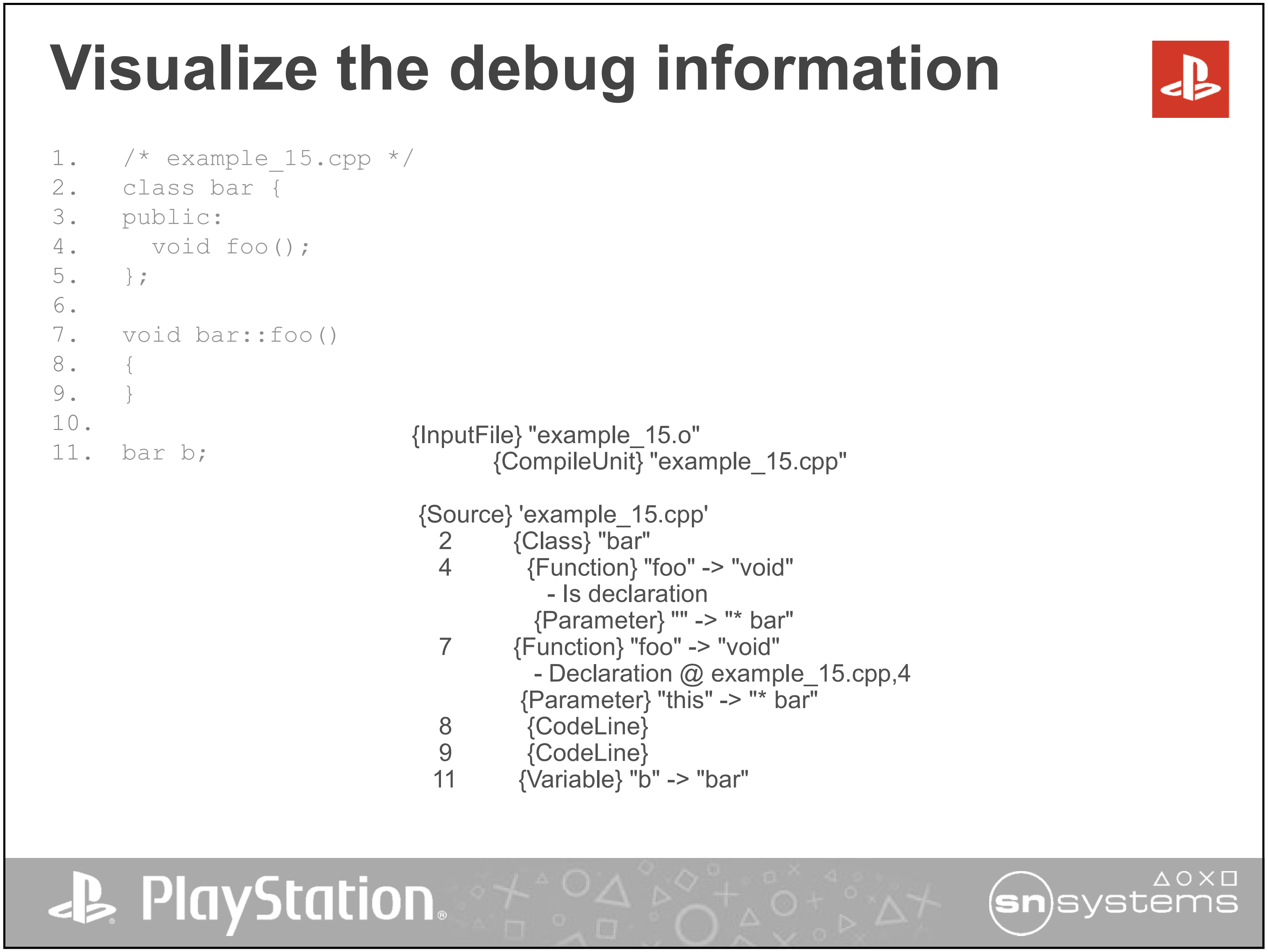
DIVA output is:

a) A high level view of ELF debug information
b) Same output regardless of the DWARF layout or spec version

Use cases:

1. Visualize / inspection of the debug information
2. Compare debug information from two compilations
3. Regression testing
4. Creation of debug information reproducible test case

PlayStation
sn systems

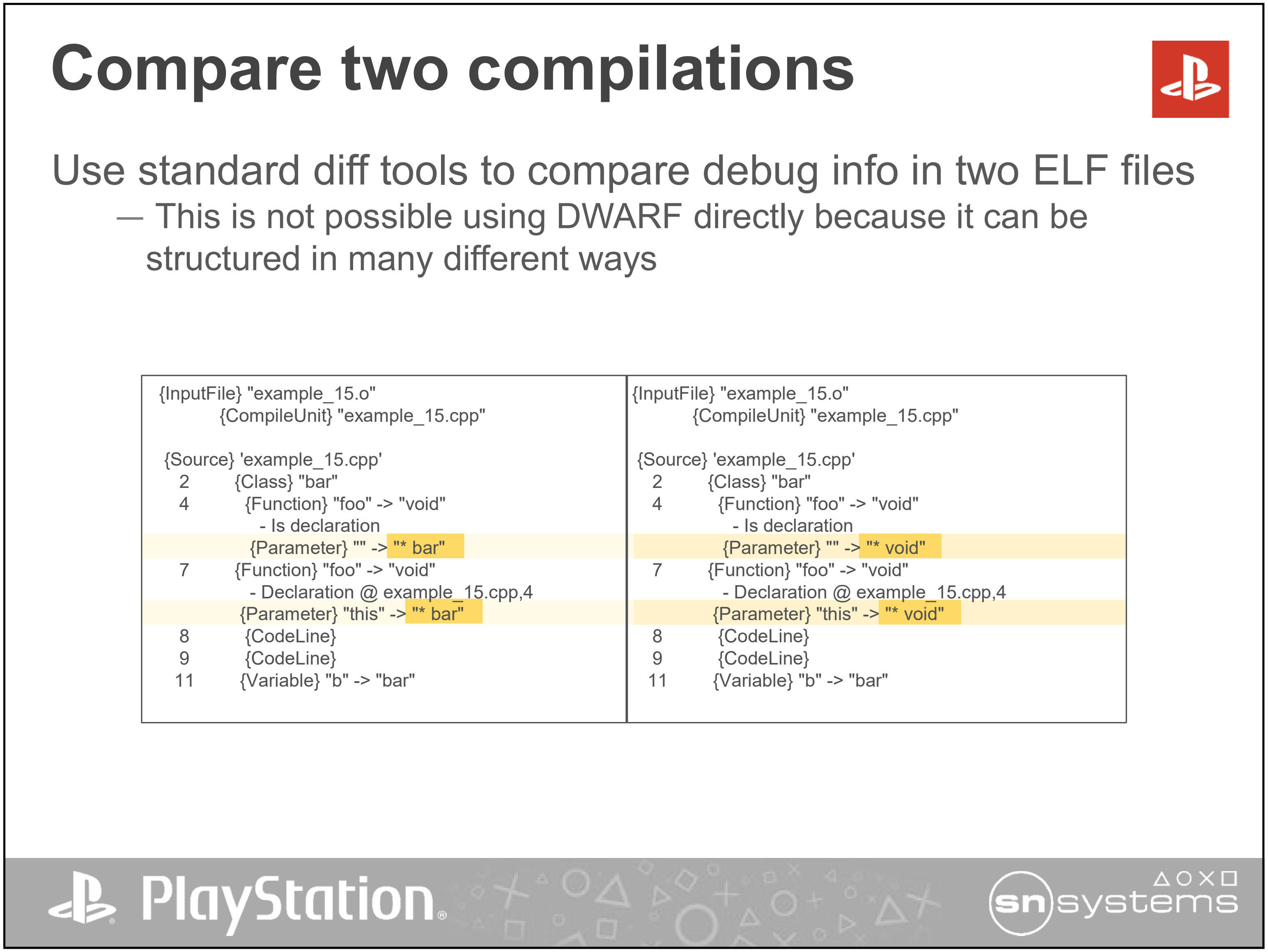


Visualize the debug information

```
1.   /* example_15.cpp */
2.   class bar {
3.   public:
4.     void foo();
5.   };
6.
7.   void bar::foo()
8.   {
9.   }
10.
11.  bar b;
```

```
{InputFile} "example_15.o"
        {CompileUnit} "example_15.cpp"

 {Source} 'example_15.cpp'
   2        {Class} "bar"
   4          {Function} "foo" -> "void"
                - Is declaration
               {Parameter} "" -> "* bar"
   7        {Function} "foo" -> "void"
               - Declaration @ example_15.cpp,4
             {Parameter} "this" -> "* bar"
   8          {CodeLine}
   9          {CodeLine}
  11         {Variable} "b" -> "bar"
```



Compare two compilations

Use standard diff tools to compare debug info in two ELF files

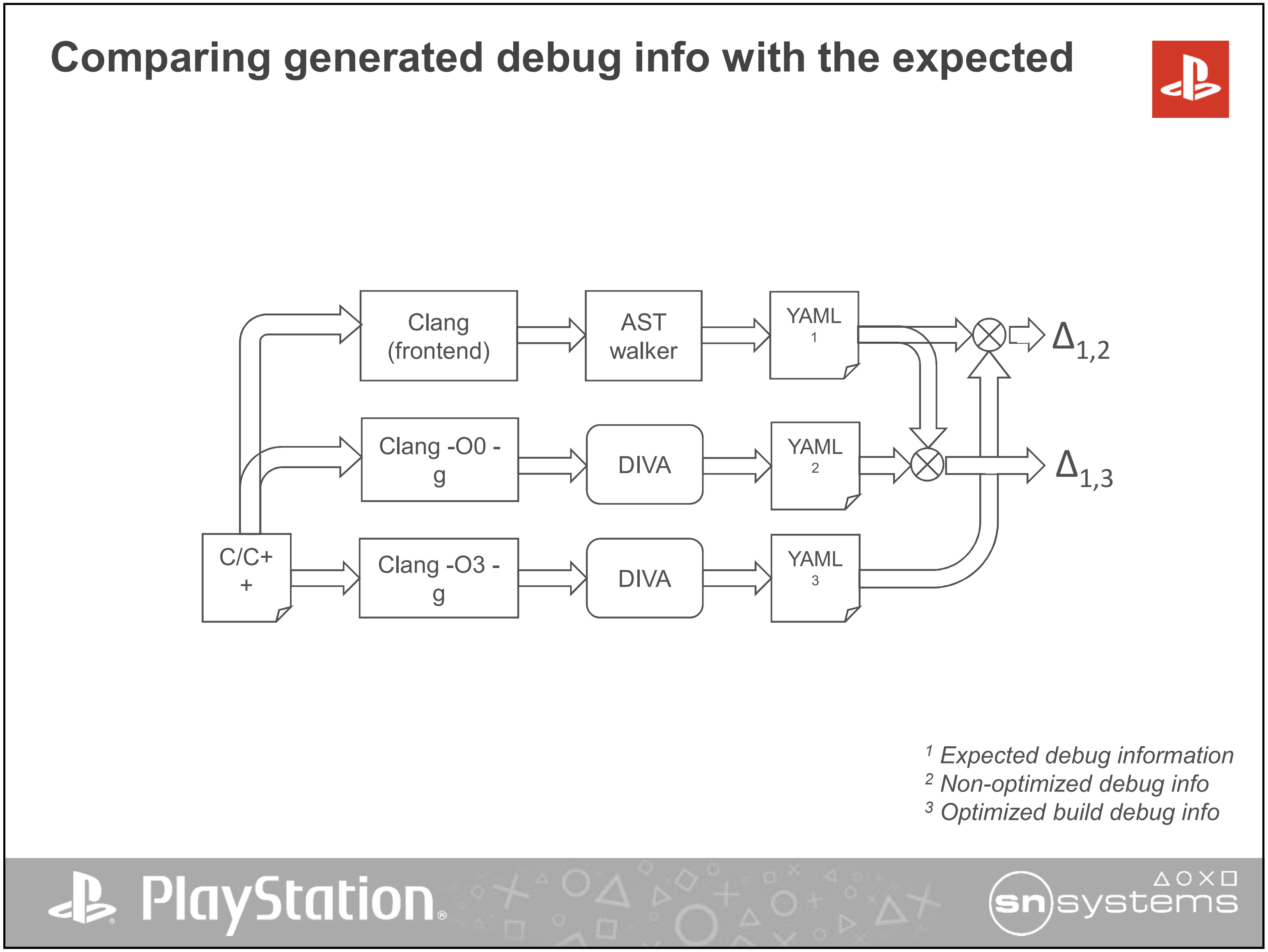
- This is not possible using DWARF directly because it can be structured in many different ways

```
{InputFile} "example_15.o"
         {CompileUnit} "example_15.cpp"

 {Source} 'example_15.cpp'
   2        {Class} "bar"
   4          {Function} "foo" -> "void"
                - Is declaration
               {Parameter} "" -> "* bar"
   7        {Function} "foo" -> "void"
              - Declaration @ example_15.cpp,4
             {Parameter} "this" -> "* bar"
   8          {CodeLine}
   9          {CodeLine}
  11         {Variable} "b" -> "bar"
```

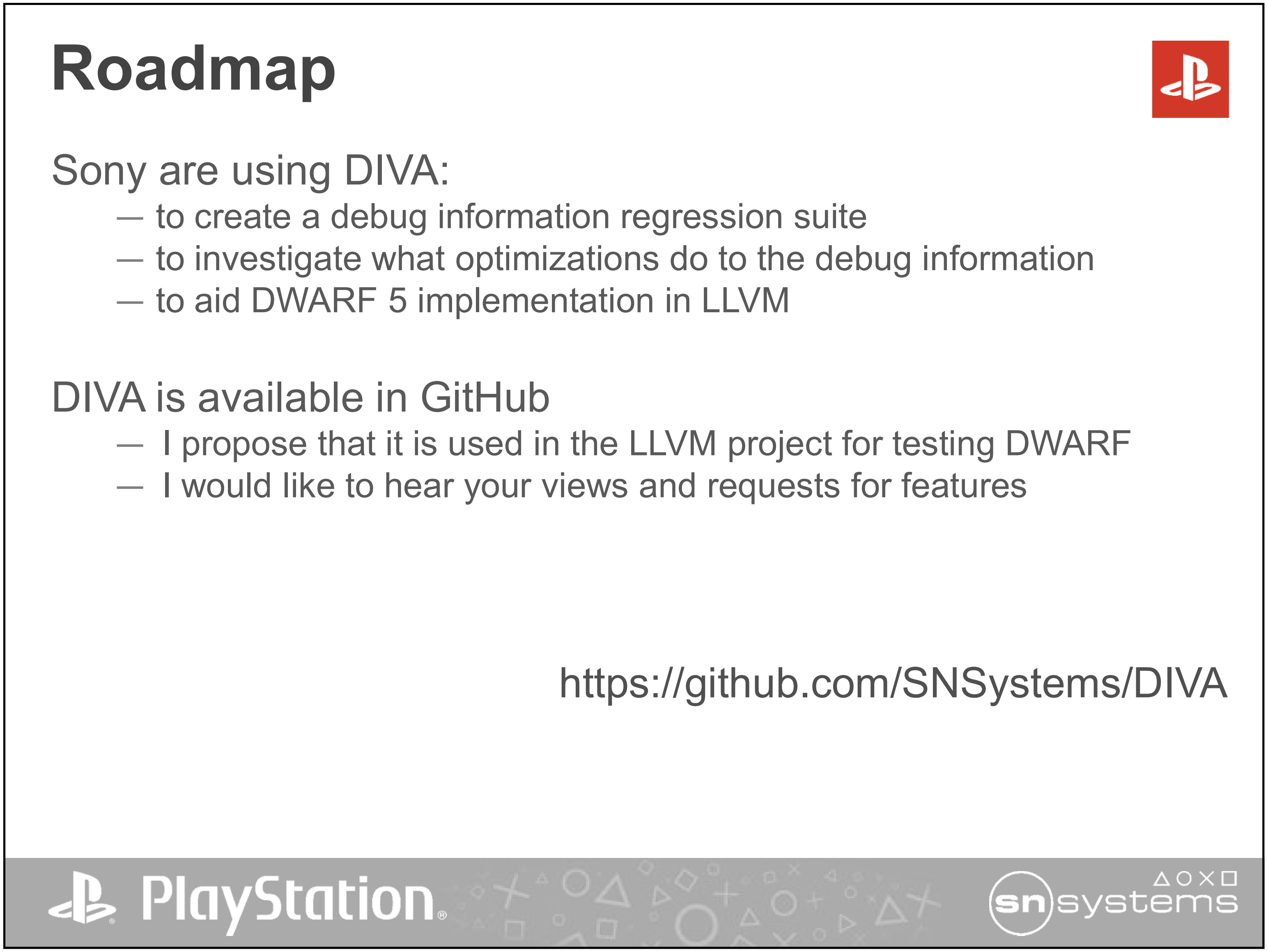
```
{InputFile} "example_15.o"
         {CompileUnit} "example_15.cpp"

 {Source} 'example_15.cpp'
   2        {Class} "bar"
   4          {Function} "foo" -> "void"
                - Is declaration
               {Parameter} "" -> "* void"
   7        {Function} "foo" -> "void"
              - Declaration @ example_15.cpp,4
             {Parameter} "this" -> "* void"
   8          {CodeLine}
   9          {CodeLine}
  11         {Variable} "b" -> "bar"
```



Comparing generated debug info with the expected

[1] *Expected debug information*
[2] *Non-optimized debug info*
[3] *Optimized build debug info*



Roadmap

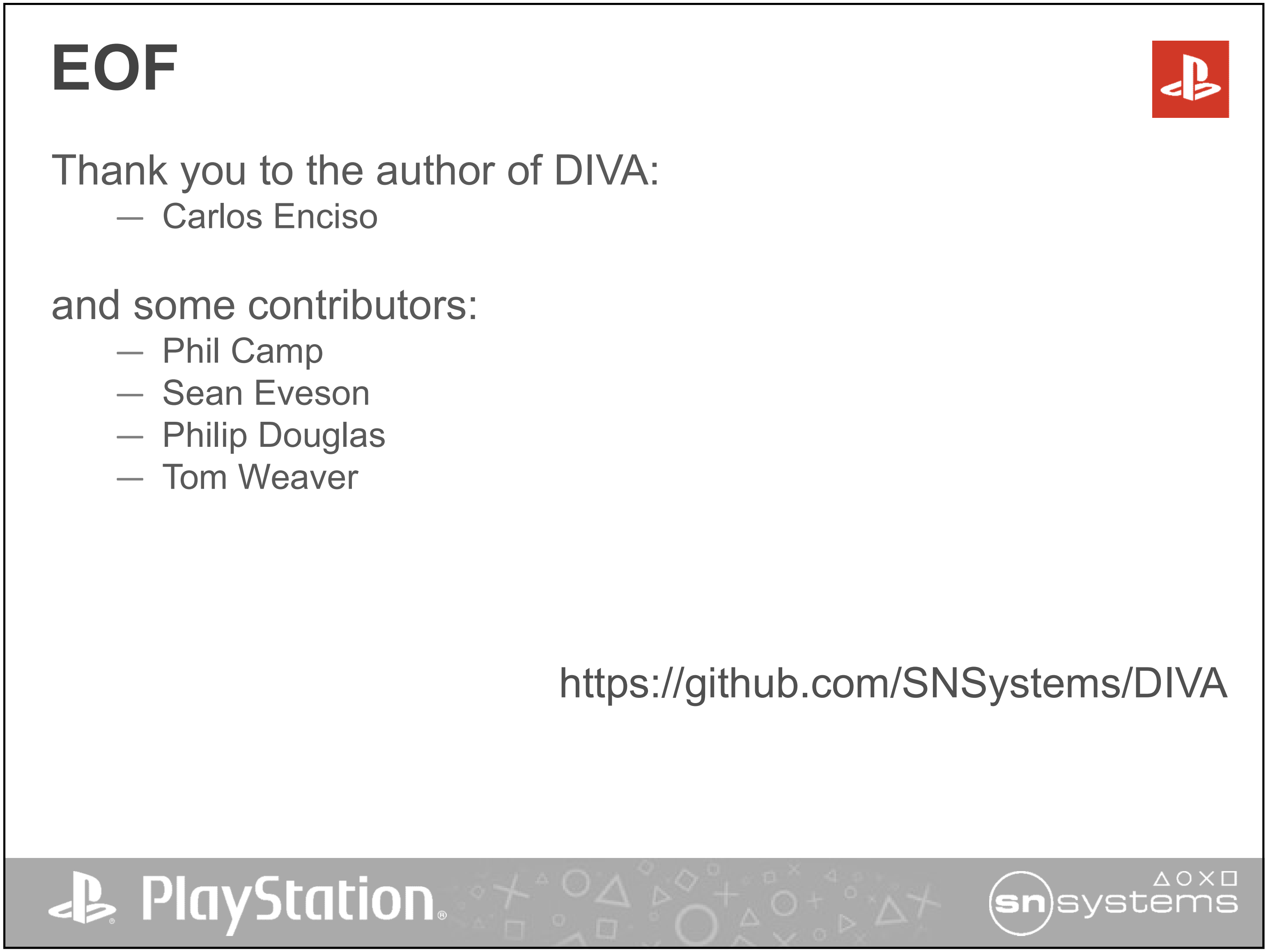
Sony are using DIVA:

- to create a debug information regression suite
- to investigate what optimizations do to the debug information
- to aid DWARF 5 implementation in LLVM

DIVA is available in GitHub

- I propose that it is used in the LLVM project for testing DWARF
- I would like to hear your views and requests for features

https://github.com/SNSystems/DIVA



EOF

Thank you to the author of DIVA:

- Carlos Enciso

and some contributors:

- Phil Camp
- Sean Eveson
- Philip Douglas
- Tom Weaver

https://github.com/SNSystems/DIVA

PlayStation
sn systems